Dedicated to innovation in aerospace

NLR - Royal Netherlands Aerospace Centre

MASTER OF SCIENCE THESIS REPORT | SEPTEMBER 2020

Agent-based modelling and simulation for quantification of resilience in air transport

The effects of a sudden and unexpected bad weather disturbance on conventional approach operations

DEPARTMENT: **Safety Institute**

TU Delft

In collaboration with

Agent-based modelling and simulation for quantification of resilience in air transport

The effects of a sudden and unexpected bad weather disturbance on conventional approach operations

Master of Science Thesis

By

R.J.J. Groenen

Student number:
4235827

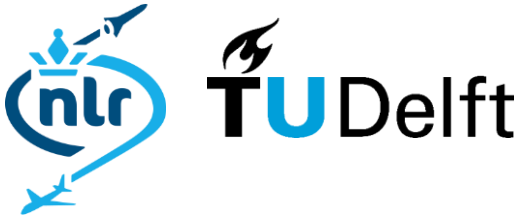
Delft University of Technology
Faculty of Aerospace Engineering
Control & Operations
Air Transport Operations

Supervised by:
Dr. S.H. Stroeve
Dr. O.A. Sharpanskykh

Committee:
Chair: Dr. ir. B.F. Santos
Supervisor (NLR): Dr. S.H. Stroeve
Supervisor (TU Delft): Dr. O.A. Sharpanskykh
External examiner: Dr.ir. E. van Kampen

For obtaining the degree of Master of Science in Aerospace Engineering at Delft University of Technology
September 12, 2020

To be defended publicly on September 28, 2020



Preface

This thesis report documents my completed research project in fulfilment of the degree of Master of Science in Aerospace Engineering at the Delft University of Technology. This research project was offered by the Netherlands Aerospace Centre (NLR) and was performed at the Aerospace Operations Safety Institute department in Amsterdam. I therefore want to thank NLR first of all for providing me the opportunity to conduct this research project in an inspiring environment full of aerospace expertise.

While I was taking the various interesting courses of the Air Transport Operations (ATO) profile I knew that my to be conducted research project had to be related to the contents of either the *Agent-Based Modelling and Simulation* course or the *Operations Optimization (i.e. Network Scheduling / Airline Planning & Optimization)* course. Fortunately, I came across, by chance, the vacancy for this MSc project, which was posted on the website of NLR. The attached project description got my attention, and until today, I am so grateful that I was given the opportunity to conduct this research project. The modelling and simulation phase of the research project can be evaluated as both real fun and challenging at the same time. Since it was an objective on its own to introduce NLR to the capabilities of the simulation software *AnyLogic*, it required quite some time to become completely familiar with the powerful functionalities of the software package. I am therefore more than proud of the extensive tool that I have developed in *AnyLogic* for the purpose of the defined research objective.

During my internship at NLR I was intensively supervised by Sybert Stroeve. I would like to thank Sybert for his critical attitude, his constructive feedback, the many supportive meetings and the good discussions. In addition I had the pleasure to be supervised by Alexei Sharpanskykh as well. The periodic meetings with Alexei at the university provided me of additional guidance, thoughts and perspectives that has further improved the quality of the conducted research. I would like to express Alexei my gratitude for sharing his knowledge and insights both during the thesis phase and during the *Agent-Based Modelling and Simulation* course.

Finally, I would also like to thank my dear friends and fellow students for spending the long days with at the university both during the pre-master and during the master. Thank you for your support and above all for the great and fun times that we had during these years.

Remco Groenen
Leende, September 2020

Abstract

Socio-technical systems consist of deeply interconnected and interdependent social entities and technical systems that collaborate to achieve a global goal. The individual characteristics and behaviours of each involved actor and their interactions define the resulting overall emergent system behaviour. Due to increased complexities in socio-technical systems traditional safety risk assessment strategies are found to become less suitable to predict, to reveal and to understand emergent system behaviour. In recent years there has therefore been a shift in the way how safety in complex socio-technical systems is perceived. The relatively new safety management paradigm called resilience engineering focusses on the ability of socio-technical systems to cope with varying conditions by applying everyday performance. Recent studies related to resilience engineering insist on the need for more structured modelling approaches for analysis and quantification of resilience in socio-technical systems. This study contributes to this need by presenting a quantitative agent-based modelling and simulation approach. The suitability of such approach for more profound analysis and quantification of resilience in socio-technical systems has been studied in the context of conventional approach operations during a sudden and unexpected bad weather disturbance. The formal agent-based model that has been developed for this resilience study especially emphasized the role of executive controllers in achieving and maintaining resilience. The adaptive strategies that are considered for this purpose are multiple vectoring strategies, the initiation of holding operations and go-arounds. The resilient capacities of conventional approach operations have been quantified using the emergent outcomes of these adaptive strategies. Considering the obtained simulation results and gained insight there can be concluded that quantitative agent-based modelling and simulation is a suitable, structured and powerful approach for more profound analysis and quantification of resilience in socio-technical systems.

Keywords: Agent-Based Modelling and Simulation, Resilience, Resilience Engineering, Air Traffic Management, AnyLogic, Quantification, Complex Socio-technical Systems.

List of figures

List of tables

List of acronyms

Acronym	Definition
ABMS	Agent-based modelling and simulation
AIP	Aeronautical Information Package
ANSP	Air Navigation Service Provider
APP	Approach
ARR	Arrival
ATC	Air Traffic Control
ATM	Air Traffic Management
BADA	Base of Aircraft DAta
CAT	Category
CNS	Communication, Navigation and Surveillance
CPDLC	Controller Pilot Data Link Communications
FAP	Final Approach Point
FMC	Flight Management Computer
FMS	Flight Management System
HST	High Speed Taxiway Turn-off
IAC	Instrument Approach Chart
IAF	Initial Approach Fix
IAP	Instrument Approach Procedure
IF	Intermediate Fix
IM	Interval Management
MCP	Mode Control Panel
NM	Nautical Mile(s)
PA	Precision Approach
RE	Resilience Engineering
RNAV	Area Navigation
ROT	Runway Occupancy Time
R/T	Radio Telephony
SA	Situation Awareness
SESAR	Single European Sky ATM Research
SRM	Safety Reference Material
STAR	Standard Terminal Arrival Route
TMA	Terminal Manoeuvring Area
TNE	Terminal East Sector
TNW	Terminal West Sector
TWR	Tower
WTC	Wake Turbulence Category

Table 1 – Acronyms

Contents

1 Introduction

1.1 The call for resilience in the ATM domain

In the past decades air traffic movements have increased significantly and continuously, both in the air and on the ground [10]. These increased traffic numbers challenge current air traffic operations and airspace infrastructure. A lot of research has therefore been devoted to increase safety, efficiency, capacity, access, flexibility, predictability and resilience in air traffic management (ATM). One of the major challenges of future air traffic growth is the lack of capacity available at airports, which is in general limited by the runway system [10]. The traffic densities at and around airports can nowadays easily be observed using a simple online flight tracking service. Especially major airports will show a relatively large number of aircraft being on arrival and approach during peak hour conditions. These specific flight phases are an example of operations that are susceptible to disturbing events. The expected future growth in the number of flights will even further increase the risk of such challenging and disrupting occurrences both in the air and on the ground. Because of increasing complexity in ATM there is a need to understand these risks and its implications on the operations in the ATM domain.

In recent years there has been a shift in the way how safety is perceived. While safety is traditionally viewed as the absence of unwanted outcomes such as errors and accidents (Safety-I), recent trends insist on the necessity to understand and support how safety is actively produced (Safety-II) [21]. From a Safety-II perspective, the purpose of safety management is to ensure that as much as possible goes right and that everyday work achieves its intended purposes. This new safety management paradigm is called *resilience engineering* and aims to enhance the ability of a complex socio-technical system to succeed under varying conditions and to perform in a way that produces acceptable outcomes.

In recent years the concepts *resilience* and *resilience engineering* have gained significant interest in a large number of domains [22, 34]. This increased interest is found to be closely related to the increased complexity and the inherent risks of modern socio-technical systems. The complexity in socio-technical systems is the result of a large number of deeply interconnected social entities and technical systems. The nonlinear and low-level correlations between the involved actors result in global system behaviours that can be described as being emergent and unpredictable. Because of this complex and emergent system behaviour it is a challenging task to identify the cause of failing or affected system behaviour using traditional risk management strategies. Resilience engineering on the other hand attempts to describe the way how complex socio-technical systems deal with failures and disturbing events and with that its resilient capacities. The concept of resilience (in socio-technical systems) is however relatively new and is therefore still under study.

In the past decade resilience studies have increasingly been conducted in the context of ATM, which are generally related to the funded research projects as conducted by *Eurocontrol* [9], *Resilience2050* [33] and *SESAR Joint Undertaking* [31]. These studies relate to the contributions of resilience engineering in the field of ATM and are found interesting because of the rapid development of ATM within the last couple of years. In the context of ATM resilience has been defined as the intrinsic ability of a system to adjust its functioning prior to, during, or following changes and disturbances, so that it can sustain required operations under both expected and unexpected conditions [9]. Resilience engineering is herein seen as a promising concept to analyse, understand and enhance the resilient capacities of the socio-technical ATM system.

The resilience study that is covered in this report builds further upon recent studies that indicated the need for more structured modelling approaches for analysis and quantification of resilience in socio-technical systems. One of the most recent studies in this context is performed by Stroeve and Everdij [43]. They presented agent-based modelling as a suitable and structured approach for analysis of resilience in socio-technical systems. They developed a qualitative agent-based model to analyse the resilient capacities of conventional approach operations during a sudden and unexpected bad weather disturbance. However, only high level analysis results can be obtained by applying such a qualitative agent-based modelling approach. Stroeve and Everdij acknowledge the uncertainties in their obtained simulation results and indicate that further research is desired. They indicate that a more profound analysis of resilience in socio-technical systems can be achieved using a quantitative agent-based modelling and simulation approach. The resilience study that is covered in this report builds further upon the need for quantification and analysis of resilience in socio-technical systems. The research activities that have been performed do all contribute to the following research objective:

"To quantify, analyse and understand the resilient capacities of conventional approach operations during a sudden and unexpected bad weather disturbance using a quantitative agent-based modelling and simulation approach"

This resilience study aims to deliver more profound results and insights in the resilient capacities of the socio-technical ATM system, which could not be delivered by the qualitative study of Stroeve and Everdij. Secondly, this study demonstrates the usefulness of quantitative agent-based modelling and simulation for resilience engineering in combination with the AnyLogic simulation software.

1.2 Report structure

The contents of this thesis report are structured as follows:

- **Section 2 – Literature review: from theory to practice**: provides all necessary background information, related research topics and current state-of-the-art with respect to the research objective;
- **Section 3 – Research objective, scope and methodology**: describes what the research is trying to achieve and what the main activities have been to answer the research questions;
- **Section 4 – Case study – Rome Fiumicino airport**: describes the operational scope of the case study by providing all related information about the used approach procedures, type of operations and involved approach sectors;
- **Section 5 – Model description**: describes the agent-based model that has been developed to study the resilient capacities of conventional approach operations;
- **Section 6 – Model verification**: describes the verification processes that have been performed during and after model implementation;
- **Section 7 – Model validation**: describes the model validation process that has been performed;
- **Section 8 – Experiments**: describes the experiment set-up, the conducted experiments and the obtained simulation results that altogether define the simulated resilient capacities of the modelled approach operations during a bad weather disturbance;
- **Section 9 – Discussion, conclusions and recommendations**: describes the relevance of the followed quantitative agent-based modelling and simulation approach in the field of resilience engineering, presents the conclusions that have been drawn regarding the research questions, and presents recommendations for future research.

2 Literature review: from theory to practice

This chapter provides all necessary background information that familiarises the reader with the related research topics and current state-of-the-art related to the field of resilience (engineering). This information serves as the basis for the resilience related research that has been conducted during this research project. The research objective that is specified in the next chapter is therefore defined by the research gaps that are introduced in this chapter.

The resilience study that is covered in this report can in short be summarized as "*the quantification of resilience in the complex socio-technical ATM system using an agent-based modelling and simulation approach*". This formalisation contains some key topics that define the contents of the research objective and the research methodology. This chapter will therefore provide some brief descriptions and related state-of-the-art to introduce the reader to the following topics:

- Complex socio-technical systems
- The concept of resilience
- Resilience engineering
- Properties and characteristics to express resilience
- Resilience assessment metrics
- Quantitative approaches for modelling resilience in complex socio-technical systems
- Agent-based modelling and simulation approach to study complex socio-technical systems
- Agent-based modelling and mental simulation for resilience engineering in air transport

The information that is provided in each section will familiarize the reader with the different topics that are considered in this research project. While reading through the descriptions of these topics the reader will notice that the contents shift from a theoretical perspective towards a practical perspective. This chapter will end with the practical field of resilience engineering, in which agent-based modelling and simulation is identified as a promising and structured approach for the analysis of resilience in socio-technical systems.

Complex socio-technical systems

Air Traffic Management (ATM) is defined as the dynamic, integrated management of air traffic and airspace including air traffic services, airspace management and air traffic flow management through the provision of facilities and seamless services in collaboration with all parties and involving airborne and ground-based functions [25]. ATM consists therefore of many interacting and interdependent actors, such as air traffic controllers, pilots, and many related systems that collaborate to achieve safe, economic and efficient flight operations. The combination of the dynamic interactions between such human operators and technical systems within air transport can be described as a *complex socio-technical system*. A *socio-technical system* is defined to be a system consisting of deeply interconnected social entities and technical systems [45]. *Complexity* on the other hand relates to the highly nonlinear correlations between different low-level system behaviours, states, dynamics and interactions, which eventually emerge in unpredictable global behaviours [5, 32].

The concept of resilience

The concept of resilience has gained significant interest over the years due to increasing complexity of socio-technical systems and its inherent risk due to this complexity [1]. Resilience is in general understood to be the degree and ability of an entity or system to return to its original state after a disturbance, i.e. the ability to withstand a disturbance and to recover from it [22]. Despite this general definition there are many other definitions of resilience as applied in various domains and disciplines, which prevents the establishment of a uniformly agreed definition [16]. The number of definitions that exist nowadays do however share many similarities. Bergström et al. found these similarities in terms of "*the ability to adapt to or absorb disturbing conditions*" and "*the ability to keep the system within its functional limits*" [1]. Erik Hollnagel, an internationally recognised specialist in the field of resilience engineering, defines a system to be resilient "*when it can sustain required operations under both expected and unexpected conditions by adjusting its functioning prior to, during, or following events*" [21].

Resilience engineering

Resilient performance is achieved and maintained by the work and results of resilience engineering. Resilience engineering (RE) aims to enhance the ability of a complex socio-technical system to adjust its functioning to sustain required operations notwithstanding disturbance, disruption and change [18, 42]. RE as part of Safety-II looks for ways to enhance the ability of systems to succeed under varying conditions and to remain productive [20]. The focus in Safety-II is therefore on the system's ability to perform in a way that produces acceptable outcomes and to ensure that things go right. In this sense Safety-II differs significantly from traditional risk management strategies (Safety-I), which focus on reducing the likelihood of disturbing events and reducing the potential adverse consequences of such events.

Properties and characteristics to express resilience

The resilient performance of socio-technical systems can be expressed by a number of properties. In general all (intelligent) properties that help a system cope with disturbances, either in advance or afterwards can be interpreted as a resilience characteristic. The most common resilience properties as found in literature seem to converge to the capacities to *absorb*, to *adapt* and to *recover*, of which the ability to adapt is seen as the most important capacity within resilience [16, 22, 44, 46, 50]. The three commonly used resilience properties or capacities can in general be described as follows:

- *absorptive capacity*: the degree to which a system can absorb disturbances to minimize the consequences;
- *adaptive capacity*: the ability of a system to adjust to disturbances and undesirable situations, which becomes required when the absorptive capacity can no longer hold the disturbance;
- *recovery/restorative capacity*: the ability of a system to return to a normal or improved state in a relatively short period, dependent on what is defined as desirable control and operations;

The adaptive capacity is also emphasized in two of the four conceptual perspectives of Woods [50]:

- *graceful extensibility*: the ability of a socio-technical system to adapt and to stretch in order to cope with surprise events. The amount of graceful extensibility determines the system performance and adaptive capacity when confronted with the boundaries of the normal and disturbed operational envelope;
- *sustained adaptability*: the ability of a socio-technical system to preserve the adaptive capacities despite changed conditions or system performance and properties over time.

Resilience as graceful extensibility and resilience as sustained adaptability are two dimensions that are still not well-understood due to the complexity of socio-technical systems nowadays. These dimensions are therefore still under study and part of state-of-the-art research [51].

Hollnagel defines the ability to *adjust* a key feature of a resilient system. These adjustments can be either reactive (something has happened; response to an occurrence) or proactive (something has to happen; be anticipatory) [19, 20]. Based on the work of Hollnagel, RE has proposed that the following four abilities (cornerstones) are necessary for resilient performance [19, 22]:

- the ability to *respond*: the system knows how to respond to a state change due to a disturbance, and knows what to do by adjusting its mode of functioning;
- the ability to *monitor*: the system can monitor those aspects within the system internally and the environment externally which can (potentially) affect the system's performance;
- the ability to *learn*: the system learns to handle disturbances based on experience;
- the ability to *anticipate*: the system knows what to expect and is able to anticipate on disturbances based on gained knowledge.

These four abilities are the minimal requirements for a system to have a resilient performance. The ability to *adapt*, in which a system is able to adjust or modify itself based on certain conditions, is seen as a combination of the abilities to respond, learn and monitor.

Resilience assessment metrics

Assessment metrics and methods have been developed that aim to analyse, assess and model the resilient capacities of complex socio-technical systems. Within the resilience assessment methodologies one can distinguish qualitative and quantitative approaches [22]. The qualitative assessment approaches seek to explain and understand resilient behaviour in words and therefore exclude numerical values. Within resilience engineering there are typically qualitative approaches used to assess resilience of complex socio-technical systems. These qualitative approaches provide however significantly less level of detail in assessing resilient properties when compared to quantitative assessment methods. Quantitative assessment approaches assess resilience in a numerical manner. In general, the majority of these quantitative assessment approaches are based on measuring and comparing the relative impact of a disturbance on system performance and the time it takes to recover. The notion for this approach is called the *resilience triangle* with the two key elements *initial impact* and *recovery time*. The *initial impact* is a measure for the amount of disturbances a socio-technical system can absorb or withstand before leading to significant changes in its KPIs. *Recovery time* is a measure for the duration of the period between the significant reduction in KPIs and the moment of recovery. The quantitative assessment approaches that are based on this resilience triangle paradigm generally assess resilient properties of socio-technical systems on the three commonly used resilience capacities to *absorb*, to *adapt* and to *recover*. Figure 1 provides a typical representation of the resilience triangle paradigm that is considered. The figure indicates the many resilience characteristics that have been identified in literature and which are used to describe system performance during and after disturbance [4, 16, 17, 44].

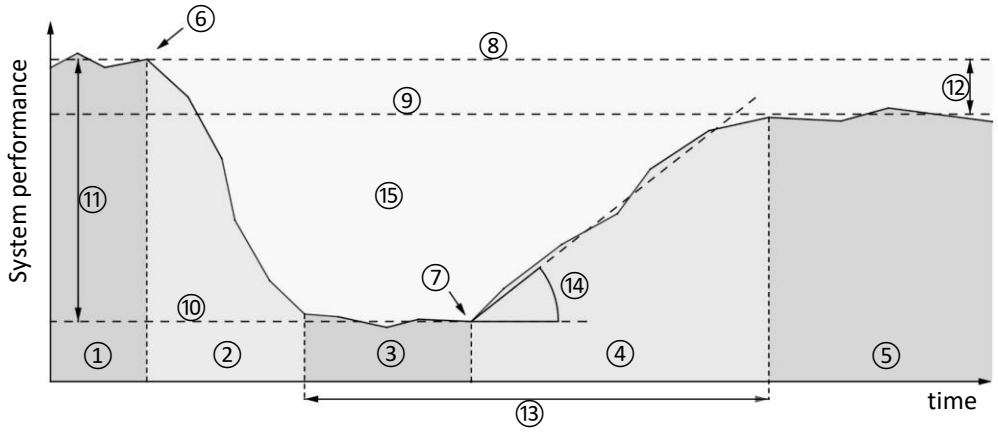


Figure 1 – Typical representation of the "resilience triangle" paradigm and corresponding properties

Number	Description
①	Stable original state [17]
②	System disruption [17], absorption
③	Disrupted state [17], adaptation
④	System recovery [17], restoration
⑤	Stable recovered state [17]
⑥	Disruptive event [17], disruption [44]
⑦	Resilience action [17]
⑧	Original stable system performance [16], desired/normal performance level [44]
⑨	Performance at a new stable level after recovery [16], recovered performance [44]
⑩	Performance level immediately post-disruption [16], minimum performance [44]
⑪	Absorptive capacity [16], absorption factor [44]
⑫	Adaptive capacity [16], recovery factor [44]
⑬	Time to recovery (used in all metrics)
⑭	Recovery rate [27]
⑮	Resilience loss [4]

Table 2 – Characteristics that are often used to assess system resilience as the ratio of recovery to the loss suffered by the system (figure 1)

Quantitative approaches for modelling resilience in complex socio-technical systems

Literature provides many deterministic and probabilistic metrics for resilience quantification. The majority of these resilience metrics are based on measuring and comparing the relative impact of a disturbance on system performance and the time it takes to recover (*resilience triangle*). These metrics require data however in order to provide quantitative results. For this reason there is need for more structured modelling approaches that are able to reveal the emergent properties of complex socio-technical systems and to quantitatively asses resilience as such [43]. Patriarca et al. [30] indicated in their literature review the significance, relevance and continuous interest of the modelling aspect within resilience engineering in current and future research. Many of their reviewed contributions are related to the development of reliable and advanced models to measure and assess resilience either in a qualitative or quantitative manner. These models and measurements are first of all required to understand the inherent complexities of socio-technical systems and secondly to fill the gap between the theoretical notion RE and its applicability in practical contexts. The need for more quantitative methods in measuring resilience, such as mathematical modelling and computer simulations is also argued by Righi et al. [34].

Agent-based modelling and simulation approach to study complex socio-technical systems

Studying complex socio-technical systems is challenging due to the many different actors involved and their corresponding behaviour, interactions and dependencies. The overall global behaviour emerges as the result of the individual processes of each actor and their interactions. In order to capture such emergent behaviour a bottom-up approach is required where only the low-level interactions between the social entities and technical systems are specified and not the system as a whole. Literature that is related to the resilience engineering topic considers agent-based modelling and simulation (ABMS) as a structured and promising approach to capture the emergent, dynamic and interdependent behaviour of complex socio-technical systems [3, 36, 43, 47].

An agent-based model is composed of a number of autonomous decision-making entities, called agents, that operate in a specific environment [28]. The overall behaviour and performance of the modelled socio-technical system emerges as the result of the individual agent processes and their interactions [45]. Agent-based approaches have therefore emerged as the key method to understand the interdependencies and interactions between system components in complex socio-technical systems. Because of this, agent-based modelling and simulation has also proven to be very suitable in modelling for instance human behaviour, decision-making and performance variability [2, 3, 28].

Agent-based modelling and mental simulation for resilience engineering in air transport

The notions *resilience* and *resilience engineering* have gained significant interest in the air transport sector as well [9, 14]. This interest resulted in the development and implementation of RE Guidance as part of the SESAR Safety Reference Material (SRM) [31]. The objective of the developed RE Guidance is to analyse and improve resilience within ATM by the use of resilience engineering principles. The proposed RE activities by SESAR were meant to examine how socio-technical system design is capable of handling and adjusting to varying conditions and based on that knowledge how such socio-technical system design can be improved. The development of the RE Guidance led to an interest for more structured means for attaining a deepened understanding of socio-technical systems and for gaining further insight in its resilient capacities. In line with this interest Stroeve and Everdij [31, 43] developed a qualitative agent-based model to demonstrate the added value of such approach for the analysis of the adaptive capacities and with that the resilient properties of the socio-technical ATM system. The analysis of these adaptive capacities was performed by reasoning in a qualitative manner how relevant states or indicators change over time due to the interactions in the agent-based model. These so-called mental simulations were related to the implications of a sudden and unexpected bad weather disturbance at the airport. A shortcoming of mental simulations is however the fact that such approach only provides high-level results. This is because of the difficulty to take into account the many low-level interactions and dynamics in complex socio-technical systems. Stroeve and Everdij acknowledge these uncertainties that still exist in their qualitative simulation results and indicate that further research is desired. They indicate that a more profound analysis of the resilient capacities of disturbed approach operations can be achieved by performing computer simulations of a formal quantitative agent-based model.

3 Research objective, scope and methodology

The previous chapter provided an overview of the relevant literature, trends and the research gaps related to resilience (engineering) and the modelling and quantification of resilience in complex socio-technical systems. Based on this obtained knowledge the objective of this resilience study can be defined. This chapter will therefore describe the defined research objective, the corresponding scope of the study, the related research questions and the followed research methodology.

3.1 Research objective

By considering the state-of-the-art research and the corresponding topics that have been covered so far the following research objective is defined for this research project:

"To quantify, analyse and understand the resilient capacities of conventional approach operations during a sudden and unexpected bad weather disturbance using a quantitative agent-based modelling and simulation approach"

Following this objective, this resilience study is expected to deliver more profound results and insights of resilience in the socio-technical ATM system, in comparison with the results and insights that are obtained using a qualitative agent-based modelling approach [43]. The followed quantitative agent-based modelling and simulation approach and its obtained results are used to contribute to the need for more profound analysis and quantification of resilience in socio-technical systems.

3.2 Research scope

The research objective of this research project aims to achieve a (more) profound analysis of the resilient capacities of conventional approach operations. In order to establish proper research questions that support the defined research objective one should be familiar with the type of operations and the emergent patterns that are of interest. This section will therefore describe the research scope that is considered for this resilience study.

Approach operations

The specific air traffic application that is considered in this study are conventional approach operations of multiple traffic streams towards a single runway. As already described in the literature review such air traffic application can be described as a complex socio-technical system where many human operators and technical systems collaborate to achieve safe and efficient approach operations. This specific system comprises a number of air traffic controllers, supervisors, the multiple aircraft on approach, the involved flight crews and various related CNS systems. Each of these actors (subsystems) has a specific responsibility and behaviour within the considered approach operations. The actions and decisions of each actor (may) influence the behaviour of the other involved and interconnected actors.

Varying condition

The specific disturbance that is considered in this study is a sudden and unexpected bad weather scenario at the airport. This disturbance is selected from a larger list of identified disturbances in ATM [37, 39 40]. Such bad weather disturbance results in a contaminated runway and reduced visibility during final approach. A contaminated runway will in turn adversely affect the braking performance of aircraft. This affected braking performance in combination with a reduced visibility will eventually cause the runway capacity to decrease. As the result of this reduced runway capacity the throughput capacities that are maintained in the multiple approach sectors have to be adjusted accordingly.

Adaptive strategies

The main strategy that a controller will apply to adjust its maintained throughput capacity is the initiation of a vectoring strategy [15]. Aircraft are herein instructed to deviate from their standard approach route until a sufficient spacing between the aircraft is achieved. Aircraft will eventually be instructed to resume their original approach operations if the spacing has been increased sufficiently. Other strategies that can be applied to reduce the throughput capacity due to the reduced runway capacity are for instance the initiation of holding operations, speed and/or altitude instructions (not common), and go-arounds.

Emergence

The main emergent phenomena that is of interest in this study is the way in which arrival and approach operations are affected and emerge as the result of changed weather conditions at the airport. More specifically, this resilience study is interested in the type of actions and strategies (i.e. vectoring, holding, go-arounds) that are applied by the controllers in both normal and disturbed conditions, the resulting trajectories that will be flown when these strategies are applied, the duration of each (to be) applied strategy, the dynamic evolution of the traffic situation in the approach sector, and the resulting workload and performance of controllers. These emergent phenomena are defined by a number of interdependent factors. These factors are for instance traffic density, size of the capacity reduction, adaptive capacities of the controller, initial spacing, the ground speed of the involved aircraft, etc. The experiments that are to be conducted aim to expose and to capture these emergent phenomena.

Resilience

The ultimate goal of this study is to gain quantitative insight in the resilient capacities of approach operations during a sudden and unexpected bad weather disturbance. These resilient capacities are defined by a number of system properties that allow to cope with the specified disturbance in an adequate manner. This study will therefore explicitly emphasize those aspects that are considered to have a significant impact on the resilient capacities of disturbed approach operations. In order to do so one should already have an initial hypothesis about the factors that affect these resilient capacities, the way in which approach operations will emerge as the result of a sudden reduction in runway capacity, and the indicators that define these resilient capacities. The resilient capacities of disturbed approach operations are in this study expressed in terms of the emergent phenomena that are discussed in the previous paragraph. Resilience in the context of approach operations is considered to be a relative property. Approach operations are defined as being less resilient when more controller actions are required to cope with the disturbance, and when aircraft in turn have to travel larger distances during their arrival/approach due to vector-, holding- and go-around operations. All the aspects that are to be modelled will have a direct (hypothetical) relation to the resilient capacities of the system under study.

3.3 Research questions

The research scope in the previous section introduced the type of insight where this resilience study is interested in and the way in which this insight is to be obtained. The following research questions are set up in order to make the objective even more concrete and to provide more guidance to what is expected and wanted as the deliverables of this research project:

1) How do conventional approach operations emerge as the result of a sudden and unexpected bad weather disturbance and to what extent can these emergent operations be described as being resilient?

This research question aims to explore and capture the type of operations that become required in order to adjust the throughput capacity in the approach sectors. It is expected that vector operations, holding operations and go-arounds will be instructed to increase the spacing between aircraft and in turn to reduce the throughput capacity. As the result of these unexpected instructions the workload of the involved controllers is expected to increase. The multiple aircraft that are currently on approach during the disturbance are likely to be in approach for an additional time period. The resilient capacities of the obtained emergent behaviours can be defined and quantified by a number of related performance indicators (to be introduced in section 8.1). The extent to which the resulting operations can be defined as being resilient is dependent on the patterns that will be observed in the obtained simulation results.

2) To what extent are executive controllers able to maintain resilient approach operations during the sudden and unexpected bad weather disturbance?
Within the complex socio-technical ATM system it is clear that the executive controllers fulfil an essential role in achieving safe and efficient air traffic movements during both expected and unexpected conditions [40, 48, 49]. The contents of the speed, altitude and/or heading instructions as provided by the executive controllers define for a large part the resilient capacities of disturbed conventional approach operations. From the perspective of resilience engineering this study should be related to the everyday performance of controllers and the corresponding outcomes of this performance. The work-as-done of controllers is for this reason considered an essential part of this resilience study. The individual skills, practices and properties of each modelled controller agent together with the cooperative and anticipative setting among the controller agents will eventually determine how the approach operations will emerge both before and during disturbance. When considering the resilient properties as proposed by Hollnagel, an executive controller is able to monitor, respond to, learn from, and anticipate on a disturbance. The controller behaviour that will be modelled should therefore take into account these four essential cornerstones. Woltjer argues that resilience from the controller's perspective can be addressed in two ways: the psychological processes which addresses the controller's ability to handle disturbances and secondly the cognitive processes that are required in the actual controlling of air traffic. One should therefore be familiar with the operator's actual performances and practices, procedures and techniques.

3) How can the socio-technical ATM system be adapted to improve the resilient capacities of conventional approach operations in the context of a bad weather disturbance?
This research question aims at finding improvements in handling conventional approach operations based on analysis of the previous two research questions. Adjusting or extending a specific strategy could have a positive effect on the resilient capacities of the socio-technical approach operations. The proposed improvements will mainly consist of high-level recommendations regarding e.g. improved adaptive strategies or control mechanisms.

4) What is the added value of a quantitative agent-based modelling and simulation approach for resilience engineering over a qualitative agent-based modelling approach?
This research question aims at comparing the obtained quantitative simulation results with the results of the qualitative analysis of Stroeve and Everdij and concluding on the usefulness of the used ABMS approach during this research project. Stroeve and Everdij provided qualitative conclusions based on their conducted mental simulations of how certain KPIs would change over time in the context of disturbed approach operations. Because quantitative agent-based models are able to provide more profound analysis, this research question is expected to provide verification of the concluding qualitative graphs of Stroeve and Everdij and is besides expected to provide more insight in the resilient capacities of disturbed approach operations.

5) How does *AnyLogic* contribute to the implementation and simulation of formal agent-based models?
AnyLogic is used as the platform for the implementation and simulation of the formal quantitative agent-based model. This research question aims to explore the beneficial features of *AnyLogic* for relatively large-scale and computationally demanding agent-based resilience studies. *AnyLogic* provides a structured architecture and a set of modelling elements that can be used to specify and implement agent properties.

3.4 Research methodology

This section provides brief descriptions of the research methodology that has been followed to find answers to the research questions that have been posed in the previous section. The descriptions below indicate the main activities that have been performed during the resilience study.

Understanding related resilience studies

The previous chapter provided the relevant literature, trends and the research gaps related to resilience (engineering). One should be familiar with these topics (i.e. resilience quantification and analysis) to make sure that this resilience study contributes to the state-of-the-art. Since this resilience study builds upon the qualitative study of Stroeve and Everdij the contents of their study are properly studied. This activity allows to familiarize with the scope of their study, their developed agent-based model, their conducted mental simulations and their obtained qualitative simulation results. This information is needed to make sure that the quantitative agent-based approach as followed in this resilience study considers a similar scope as the one that has been considered by Stroeve and Everdij. In addition, one should be familiar with the concept of resilience and its corresponding properties in order to draw proper conclusions about the resilient capacities of conventional approach operations during disturbance.

Understanding related theory and procedures

One should be familiar with the working environment, the procedures and conditions that apply during arrival and approach in order to provide valid conclusions about the resilient capacities of the socio-technical system under study. This specific phase therefore relates to the gathering of information that describes controller performance and actions, approach procedures, airspace structure, responsibilities, aircraft performance, etc. Especially the documentation of ICAO and Eurocontrol has often been used for this purpose [8, 11, 12, 23, 24, 25, 26].

Understanding the functionalities of *AnyLogic* (Java)

AnyLogic is used for the implementation and experimentation phase of this agent-based resilience study. For this reason one should be familiar with the type of modelling elements that are provided by *AnyLogic*, the functionalities and limitations of the software package, the applied Java language, the way in which simulation results are stored and exported, etc. The *AnyLogic* tutorials of professor Nathaniel Osgood have found to be very useful to become familiar with the basics of Java and the powerful features of *AnyLogic* in the context of agent-based modelling and simulation [29].

Model development

This activity concerns the development of the formal agent-based model. The quantitative agent-based model will in particular emphasize the cognitive skills of the executive controllers and the flight dynamics of the multiple aircraft that are involved. In this way the developed agent-based model is able to provide sufficient and realistic conclusions about the resulting emergent behaviour of approach operations during deteriorated weather conditions. The specification of the formal agent-based model will effectively incorporate the modelling elements that are provided by *AnyLogic*, such that an efficient implementation of the model can be achieved.

Model implementation, verification and validation

This activity involves the implementation of the formal quantitative agent-based model in *AnyLogic*, the verification of the implementation and the validation of the resulting simulation output. Efficient implementation is supported by the specification of the formal model using *AnyLogic* modelling elements. Because of these structures the development and the implementation of the agent-based model can be considered as a parallel process. Model verification has been applied as an iterative process that is performed in parallel with the implementation of the quantitative agent-based model.

Model simulation and analysis

After the model has been fully formalised, implemented and verified a number of parameter variation experiments are conducted to gain insight in the resilient capacities of approach operations during a bad weather disturbance. These type of experiments aim to capture the emergent phenomena of interest (as introduced in section 3.2) before, during and after disturbance. These obtained characteristics can then be used to express the resilient capacities of the approach operations.

4 Case study - Rome Fiumicino airport

A sufficiently large and busy airport should be considered in order to properly capture the effects of a sudden and unexpected reduction in runway capacity as the result of a bad weather disturbance. For this reason it has been decided to analyse the resilient capacities of arrival and approach operations at Rome Fiumicino airport. A second argument for considering this specific airport as the case study is the fact that multiple other related resilience studies have also been performed in the context of Rome Fiumicino operations. The specific type of procedures and the resulting operations that are considered in this study are however comparable with those that are applied at other (busy) airports, such as Frankfurt am Main Airport, Hamburg Airport and Dubai International Airport.

The developed formal agent-based model is meant to simulate the effects of a bad weather disturbance on the arrival and approach operations at Rome Fiumicino. The reader should therefore be familiar with the type of approach procedures and operations that apply at this airport to understand the aspects that are taken into account in the formal agent-based model, and secondly to be able to interpret the obtained simulation results. This chapter will for this reason describe the operational scope of the Rome Fiumicino case study by providing all related information about involved approach sectors, observed operations and considered approach procedures.

4.1 Approach sectors

The case study considers the northern approach sector of Rome Fiumicino. This northern approach sector consists of the three smaller approach sectors TNW, TNE and ARR. Figure 2 shows a schematic overview of these approach sectors in relation to the IAPs that are considered. Section 4.3 will provide more detail about the IAPs and the corresponding waypoints that are considered in the operational scope of the case study. Each airspace sector is controlled by one executive controller, i.e. the TNW, TNE and ARR approach controllers and the TWR controller. The approach controllers are herein supervised by the Supervisor APP, while the TWR controller is supervised by the Supervisor TWR.

- **NE sector**: sector positioned north of the TNW and TNE sectors. The NE sector is used to build up and pre-sequence arriving traffic, which is eventually handed over to the TNW and TNE sectors at waypoints XIBIL and RITEB respectively;
- **TNW sector**: defines the northern TMA of Rome Fiumicino between XIBIL and USIRU;
- **TNE sector:** defines the northern TMA of Rome Fiumicino between RITEB and ESALU;
- **ARR sector**: represents the airspace section (TMA) in which the sequences of arriving traffic as received from the TNW and TNE sector will further be refined using radar vectors. The ARR sector allows aircraft to prepare for final approach towards runway 16L at Rome Fiumicino. Aircraft are handed over to TWR around 6 NM before the runway threshold;
- **TWR sector**: represents the CTR of Rome Fiumicino, i.e. the relatively small circular airspace section around the airport containing aircraft that are on final approach for runway 16L at Rome Fiumicino.

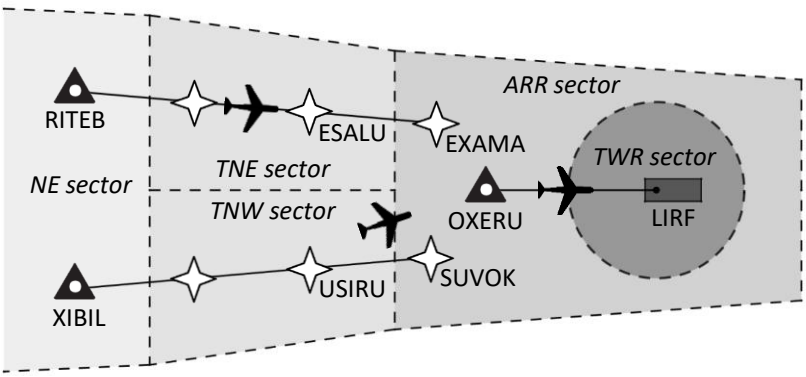


Figure 2 – Schematic overview of the northern approach sectors of Rome Fiumicino airport

4.2 Runway system

The operational scope of the case study considers two incoming streams of approaching traffic towards runway 16L. Runway 16L is the preferred runway for landing at Rome Fiumicino and is approved for CAT II/III operations (figure 3). With a length of 3902m it is of sufficient size to accommodate large wide-body aircraft. Runway 16L has multiple HSTs, of which HST DG and DH are most preferred after landing. Because of these HSTs the ROT is minimized and the runway capacity is maximized. Reduced separation procedures are applicable for runway 16L. Runway 16L does not contain any noise abatement procedures for arrival. At Rome Fiumicino there is a traffic peak three to four times a day with a maximum of 80 departing and arriving aircraft per hour. In such situations parallel runways are used.

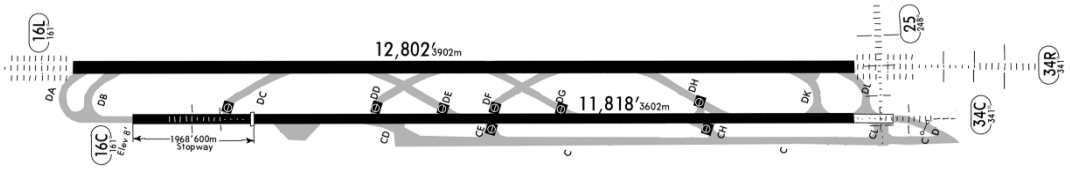


Figure 3 – Runway 16L at Rome Fiumicino

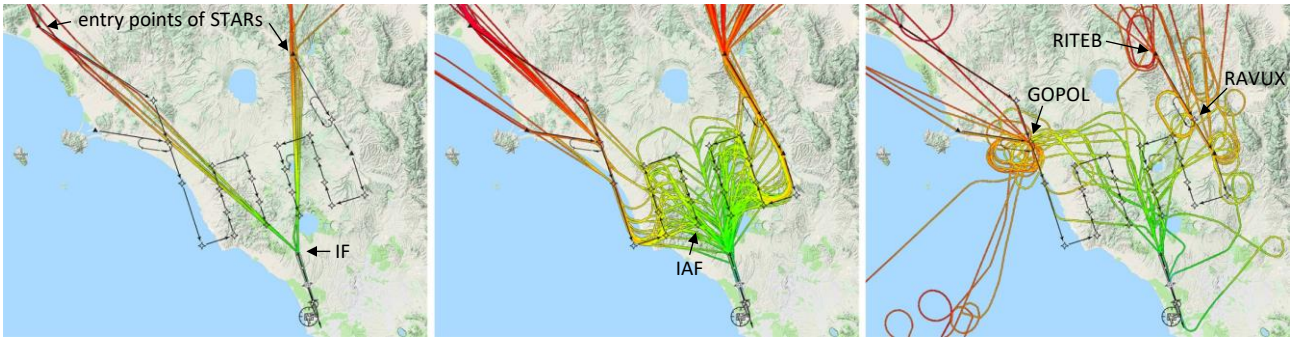
4.3 Instrument approach procedures

This section describes the instrument approach procedures (i.e. arrival, initial approach, intermediate approach and final approach) for runway 16L at Rome Fiumicino, and more specifically for aircraft that approach the airport via the northern approach sectors. These specific arrival and approach procedures are part of the operational scope of the case study.

4.3.1 Arrival segment

Aircraft that are on approach for runway 16L at Rome Fiumicino are initially arriving via the fixed profiles of multiple STAR procedures, which mark the transition between the en-route segment and the approach segment. The XIBIL2A and RITEB2A STAR procedures appear to be the default arrival procedures for aircraft that originate from the north and which are destined for runway 16L. These two STAR procedures serve therefore as the fixed and prescribed arrival procedures in the operational scope of this study. The charts of both arrival procedures can be seen in appendices A.1 and A.2. STAR procedures assist the work of approach controllers by providing a structured means in guiding aircraft towards the runway. Historic flight tracking data has been used to examine in what way these procedures assist the controllers in their work. The circa 150 obtained data files visualize the flight trajectories that aircraft have flown during their arrival and approach towards runway 16L. These data files are chosen such that they describe flight trajectories at various dates and time points. This is done to confirm that the visualized flight trajectories represent the standard and general procedures for normal arrival operations at Rome Fiumicino.

By examining the plotted trajectories, three types of observed operations can be distinguished. Two of these observed operations can be characterized as just normal. They only differ in the number of aircraft that are currently on arrival/approach. The last type of observed operations can be characterized however as less efficient and structured when compared to the first two types of observed operations. This third type of operations requires on average a longer flight time. The three types of clustered operations and the observed characteristics of each type of operation will be described below. This information is used and required to model and implement realistic approach operations that correspond with the daily operations at Rome Fiumicino, either in normal conditions or during disturbed operations.



(a) direct-to's (clearance limits) *(b) STAR procedures* *(c) vector and holding operations*

Figure 4 – Historic flight trajectories towards runway 16L at Rome Fiumicino

Arrival segment flown using direct-to's

The first type of clustered historic flight trajectories consider the operation of direct-to's, which directly connects the entry points of the STAR procedures with the IF (figure 4a). Direct-to's are in the AIP called *clearance limits*, which means that aircraft are cleared to directly fly to a certain point in space (i.e. IF). This specific type of vector is only flown when a small number of aircraft is currently on approach. The operational scope of this resilience study does therefore not consider this type of vector, since this study is only interested in the emergent behaviours as the result of a relatively large traffic density in the approach sectors.

Arrival segment flown using STAR procedures

The second type of plotted flight trajectories can be characterized by the fact that they all overlap a significant part of the corresponding STAR procedure (figure 4b). The majority of all flights that have been observed can be described by this specific type of operation. Each aircraft is herein flying the lateral profile of the STAR procedure, which prevents the necessity of providing heading instructions. Speed and altitude instructions however remain still required when these specific STAR procedures are flown. This is also clearly visible when one analyses the speed and altitude profiles of the plotted flight trajectories, which shows a relatively large variety of speeds and altitudes during arrival.

As can be seen in figure 4b the lateral profile of each STAR procedure is in general not completely flown till the IAF. Each aircraft leaves the lateral profile of the STAR procedure at a different location. This strategy is applied by the controller to merge the multiple traffic streams that are approaching Rome Fiumicino via the two (or more) STAR procedures. The relative location at which aircraft are instructed to terminate the STAR procedure provides information about the traffic density in the surrounding airspace sections of Rome Fiumicino. Aircraft are for instance required to operate a relatively larger part of the STAR procedure when these surrounding airspace sections contain a relatively large number of aircraft. This specific merging technique is called *tromboning*, which will be further explained in section 4.3.2.

Arrival segment flown using vectors, holding operations and go-arounds

The third type of clustered historic flight trajectories considers the trajectories that are typically flown during disturbed operations (figure 4c). These trajectories can be described as a combination of vector operations, holding operations and go-arounds. The trajectories that are shown in figure 4c were captured during a sudden capacity drop at Rome Fiumicino airport. The reason for this capacity reduction is however unknown. During capturing it was clearly visible that the inter-aircraft spacing had to increase. By analysing this figure there can be seen that aircraft are vectored away from the lateral profile of the STAR procedures, that the holdings at fixes GOPOL, RAVUX and RITEB are used and that go-arounds are initiated. These strategies are applied by the controller to lower the throughput capacity in the airspace sections and to lower the number of aircraft that are directed towards the runway.

The last two categories of operations (i.e. STAR, vector, holding and go-around) are considered part of the operational scope of this resilience study.

4.3.2 Initial approach segment

The initial approach segment is defined as the segment between the IAF and the IF. In this flight segment aircraft are instructed one or multiple vectors until they have intercepted the localizer of runway 16L. The specific strategy that is applied at Rome Fiumicino in the initial approach segment is called *tromboning*, which allows to properly sequence, merge and space multiple streams of approaching traffic towards the IF. The application of this tromboning technique at Rome Fiumicino can clearly be seen when looking at the flight trajectories in between the rectangular and symmetrically trombone shaped segments of the STAR procedures (figure 4b). These specific segments consist of a number of equally distanced fly-by waypoints that are used by the controller to modify the published procedures by either stretching or shortening. This technique is therefore in essence quite similar with the characteristics of the Point Merge technique [13]. Figure 5 visualizes a typical example of the tromboning sequencing technique as applied at Frankfurt am Main airport, having a similar structure as the trombones that are used at Rome Fiumicino. Aircraft are in such approach procedure cleared to proceed directly to a certain waypoint, which is in the case of Rome Fiumicino mostly the IAF or IF. These direct-to's allow the controller to guide and merge the multiple approaching aircraft in an efficient manner towards the IF (or IAF), which allows to establish a relatively high throughput capacity in the respective TMA section.

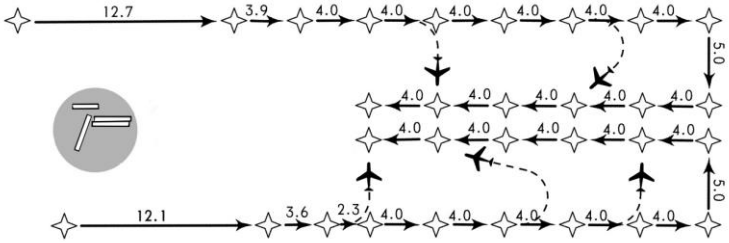


Figure 5 – Visualization of the tromboning sequencing technique as applied at Frankfurt am Main airport

There could be situations in which the trombone segments of the STAR procedures become saturated (i.e. they are completely flown). Saturation may especially occur when a large number of aircraft is currently on approach, or when relatively large separation distances are applied. The merging practice of the controller becomes more challenging once the trombone segments become saturated. This is because the required spacing between the to be merged aircraft does not increase sufficiently anymore. A controller therefore aims to enlarge the trombone segments using vector instructions such that aircraft can still be properly merged towards the IF. These specific vectors are also visualized in figure 4b, i.e. the flight trajectories that are located north of the trombone segments. A more formal definition of the term *saturation* is provided in the next chapter.

4.3.3 Intermediate approach segment and final approach segment

The last part of the IAP, i.e. the intermediate and final approach segments can be described by a straight trajectory between the IF and the runway threshold. The chart of the used approach procedure can be seen in appendix A.3. The intermediate approach segment marks herein the horizontal flight segment of the IAP between the IF and the FAP. Aircraft are in this segment adjusting their airspeed and configuration to prepare for final approach. The horizontal orientation of aircraft during the intermediate approach segment allows aircraft to intercept the glide path of the ILS from below, from where the final approach begins. The final approach segment at last can be described as the segment in which the alignment and descent towards runway 16L are made. The final approach towards this runway is in general performed using PA.

4.4 RNAV

The IAPs for Rome Fiumicino are based on the functionalities of RNAV. RNAV is a method of navigation that enables to fly any desired trajectory within the coverage of navigation aids. These trajectories are flown relative to waypoints, i.e. geographical locations of points that define a RNAV route. The IAPs for Rome Fiumicino are shaped by a number of interconnected fly-by waypoints. The navigation computers onboard modern commercial aircraft are able to determine the current position of the aircraft relative to these fly-by waypoints. The autopilot system in turn uses this situational knowledge to automatically fly the lateral profile of the IAPs. RNAV functionalities may also be used by the controllers to instruct aircraft to fly a modified arrival or approach procedure. The most common instruction in this context is the so-called *direct-to*, which allows aircraft to fly directly towards a desired fly-by waypoint instead of flying the complete published procedure. The use of RNAV in the approach sectors of Rome Fiumicino can clearly be seen in the shape of the plotted trajectories in figure 4. The many fly-by waypoints in these airspace sections can be considered a kind of grid that may be used by the controllers to guide aircraft with an efficient routing towards the runway threshold. RNAV is because of the essential functionalities as mentioned above therefore considered part of the operational scope of this resilience study.

4.5 Separation minima

Each executive controller is responsible for maintaining sufficient spacing between aircraft in its respective airspace section. This separation is maintained by the provision of heading-, altitude- and/or speed instructions. Within the approach sectors of Rome Fiumicino traditional distance based separation is applicable, where aircraft should be separated at least 3NM horizontally and 1000ft vertically. This minimum longitudinal separation distance is defined by the ICAO wake turbulence separation minima, which is a function of the WTCs of the preceding and succeeding aircraft. Table 3 lists these wake turbulence separation minima for the WTCs HEAVY and MEDIUM. These two WTCs represent the type of aircraft that are operating at Rome Fiumicino.

Aircraft WTC		
Preceding aircraft	**Succeeding aircraft**	**Separation minima**
HEAVY	HEAVY	4 NM
HEAVY	MEDIUM	5 NM
MEDIUM	HEAVY	3 NM
MEDIUM	MEDIUM	3 NM

Table 3 – Distance-based wake turbulence separation minima (only presented partly) [25]

5 Model description

This chapter will describe the formal agent-based model that has been developed to study the resilient capacities of conventional approach operations in the context of Rome Fiumicino operations. Section 5.1 will first provide an overview of the identified agents and the (type of) interactions that are considered in the model. Section 5.2 will thereafter describe the simulation environment in which the aircraft agents operate. All the remaining sections will describe the modelled properties (states and behaviour) of each identified agent and the interactions in and between the identified agents.

The identified agents, and especially the controller and aircraft agents can be described using a large number of (cognitive) properties and (internal) interactions. These modelled properties and interactions are needed to cope with the very unconstrained and dynamic characteristics of approach operations both before and during disturbance. The formal agent-based model that has been developed can therefore be described as being extensive and complex due to the many modelled dynamic processes and interdependencies. This chapter will because of these complexities and for readability reasons only describe the main concepts that are considered in the developed agent-based model. A complete formal specification of the developed agent-based model can be found in appendix C. The sections in this chapter will often refer to a particular section in this appendix for more formal context about a modelled feature.

This chapter contains a number of terms that are used to express and to refer to a modelled concept, instruction, state, or environmental property. These terms are written in italics and are used at multiple locations in this chapter and the upcoming chapters.

5.1 System identification and decomposition

In order to draw proper conclusions about the resilient capacities of the socio-technical ATM system the agent-based model should sufficiently resemble the operations that are of interest. This section therefore describes the agents and the (type of) interactions that are considered in the scope of this resilience study.

5.1.1 Agents

The following agents are identified in the formal agent-based model, together with a brief description of the most relevant behaviours and properties:

Feeder controller

- Responsible for feeding aircraft agents into the simulation environment according to a specific traffic distribution and specified generation rate (throughput capacity), such that it represents the (real) delivery of aircraft from the NE sector to the TNW/TNE sectors (figure 2).

TNW controller (Executive controller c_1)

- Responsible for one single traffic stream in the TNW sector.
- Provide speed, altitude and/or heading instructions such that a desired throughput capacity can be maintained.
- Apply vectoring strategy if separation between aircraft is considered insufficient.
- Initiate/terminate holding operations if the traffic situation in the ARR sector requires to do so.

TNE controller (Executive controller c_2)

- Responsible for one single traffic stream in the TNE sector.
- Provide speed, altitude and/or heading instructions such that a desired throughput capacity can be maintained.
- Apply vectoring strategy if separation between aircraft is considered insufficient.
- Initiate/terminate holding operations if the traffic situation in the ARR sector requires to do so.

ARR controller (Executive controller c_3)

- Responsible for two merging traffic streams in the ARR sector.
- Provide speed, altitude and/or heading instructions such that a desired throughput capacity can be maintained.
- Apply vectoring strategy if separation between aircraft is considered insufficient.
- Merge two incoming traffic streams at the intermediate fix.

TWR controller (Executive controller c_4)
- Responsible for aircraft that are on final approach towards runway 16L at Rome Fiumicino.
- Provide landing clearance.
- Instruct go-around if separation is considered insufficient.

Aircraft a_i (multiple)
- Dynamically changing position, speed, altitude and heading when flying through the simulation environment.
- Descent and separation characteristics that correspond to HEAVY/MEDIUM aircraft types.

MCP a_i (multiple)
- Responsible for the operation of the autopilot of the aircraft a_i agent.
- Defined as the collection of systems that assist the flight crew a_i agent in (automatically) controlling the trajectory of the aircraft a_i agent.

Flight crew a_i (multiple)
- Provide the MCP a_i agent with the correct input settings, in accordance with the instructions that are received by the active controller agent.

Meteo Office
- Monitor weather conditions.
- Inform Supervisor TWR when weather conditions have changed.

Supervisor TWR
- Define runway configuration and capacity.
- Inform Supervisor APP and TWR controller about a changed runway capacity after having been informed about changed weather conditions.

Supervisor APP
- Define airspace (throughput) capacity in approach sectors.
- Inform TNW, TNE, ARR and feeder controllers about a changed throughput capacity after having been informed about a changed runway capacity.

The upcoming sections will provide more detail about the complete set of (behavioural) properties that have been identified for each agent.

Communication, navigation and surveillance (CNS) systems enable within ATM the exchange of information between aircraft and controllers (communication), determines the position, orientation and airspeed of the aircraft (navigation) and allows the controller to observe this specific information (surveillance). Such systems can be considered part of the scope when one wants to take into account the effects of different working modes or when one wants to describe the dynamics and stochastics that are involved in the corresponding system processes. The CNS systems are however not considered part of the formal agent-based model because of the following assumptions:
- *The involved CNS systems are at all time working and functioning properly;*
- *The exchange of CNS information between the controller and aircraft occurs without any delay and noise;*
- *The aircraft is at all time aware of its current and exact position, heading, airspeed and altitude;*
- *The radar screen provides at all time highly accurate surveillance data of each aircraft;*

5.1.2 Interactions

Figure 6 provides a schematic overview of the agents that have been identified and the interactions between these agents, and between the agents and the environment. Note that each aircraft agent can be decomposed in this figure into a MCP and flight crew agent, instead of only aircraft a_1. Table 4 provides basic descriptions of the agent interactions that are considered in the scope of the formal agent-based model. The numbers in this table refer to the encircled numbers in figure 6. The interactions in table 4 describe the ways through which agents affect each other during either normal approach operations or in the situation when the weather conditions and the resulting runway/throughput capacity are changing. These basic descriptions aim to provide the reader with an idea of which type of interactions are considered, and therefore also the type of corresponding agent behaviours and states. Note that all communication between the controller agents and the flight crew agents can be described by the interaction that is visualized between aircraft a_1 and the TWR controller.

Nr	Interaction between agents, or between agents and the environment
1	Meteo Office notifies normalized/deteriorated weather conditions at the airport
2	Meteo Office informs Supervisor TWR about changed weather conditions at the airport
3	Supervisor TWR informs TWR controller about a changed runway capacity due to changed weather conditions
4	Supervisor TWR informs Supervisor APP about a changed runway capacity due to changed weather conditions
5	Supervisor APP informs the ARR, TNW, TNE and feeder controllers separately about a changed throughput capacity due to a changed runway capacity
6	TWR controller monitors traffic situation in the TWR sector and communicates with the flight crew of aircraft a_i if actions (instructions) are identified as necessary, such that safe and efficient approach operations can be maintained in accordance with the applied runway/throughput capacity
7	Similar context as interaction 6, but applied to the activities of the ARR controller in the ARR sector
8	Similar context as interaction 6, but applied to the activities of the TNE controller in the TNE sector
9	Similar context as interaction 6, but applied to the activities of the TNW controller in the TNW sector
10	Feeder controller monitors traffic situation at the boundary of the NE sector and generates aircraft according to the specified criteria
11	Flight crew a_i agent provides the MCP a_i agent with new input settings after having received a speed, altitude and/or heading instruction
12	The simulation environment consists of a number of significant points that are observed by the multiple aircraft and controllers to (primarily) update their situation awareness
13	Weather conditions (including wind conditions) provide an extra dimension to the simulation environment and can be perceived (in)directly by the Meteo Office, aircraft and controllers

Table 4 – Descriptions of agent interactions that are considered in the defined scope (high-level only)

Capacity modes

The majority of the interactions in table 4 describe the communication between agents about so-called capacity updates as the result of either deteriorated or normalized weather conditions at the airport. The communication of capacity updates has been incorporated into the model to simulate the temporary differences between the agents' awareness about the current runway capacity. The controller and supervisor agents can be aware of two different capacity modes: the *normal capacity mode* and the *reduced capacity mode*. The *normal capacity mode* describes the awareness of an agent about a maximum runway capacity. An agent that is acting in the *normal capacity mode* is modelled to maintain the nominal throughput/runway capacity. The *reduced capacity mode* on the other hand describes the awareness of an agent about a reduced runway capacity. An agent that is switching to a *reduced capacity mode* will therefore in reaction start to lower the throughput/runway capacity in its sector. The names of these two defined capacity modes will often be used in the upcoming sections.

Communication between controller and flight crew

The controller and flight crew agents will often be in contact with each other during approach. The time duration of this contact is modelled with a lognormal probability distribution, which makes that each contact has a different time duration. The lognormal probability distribution is kept fixed throughout the complete simulation duration. The contact between the two agents is at all time initiated by the controller agent, i.e. the flight crew agents will only react to incoming messages and not seek contact with the controller.

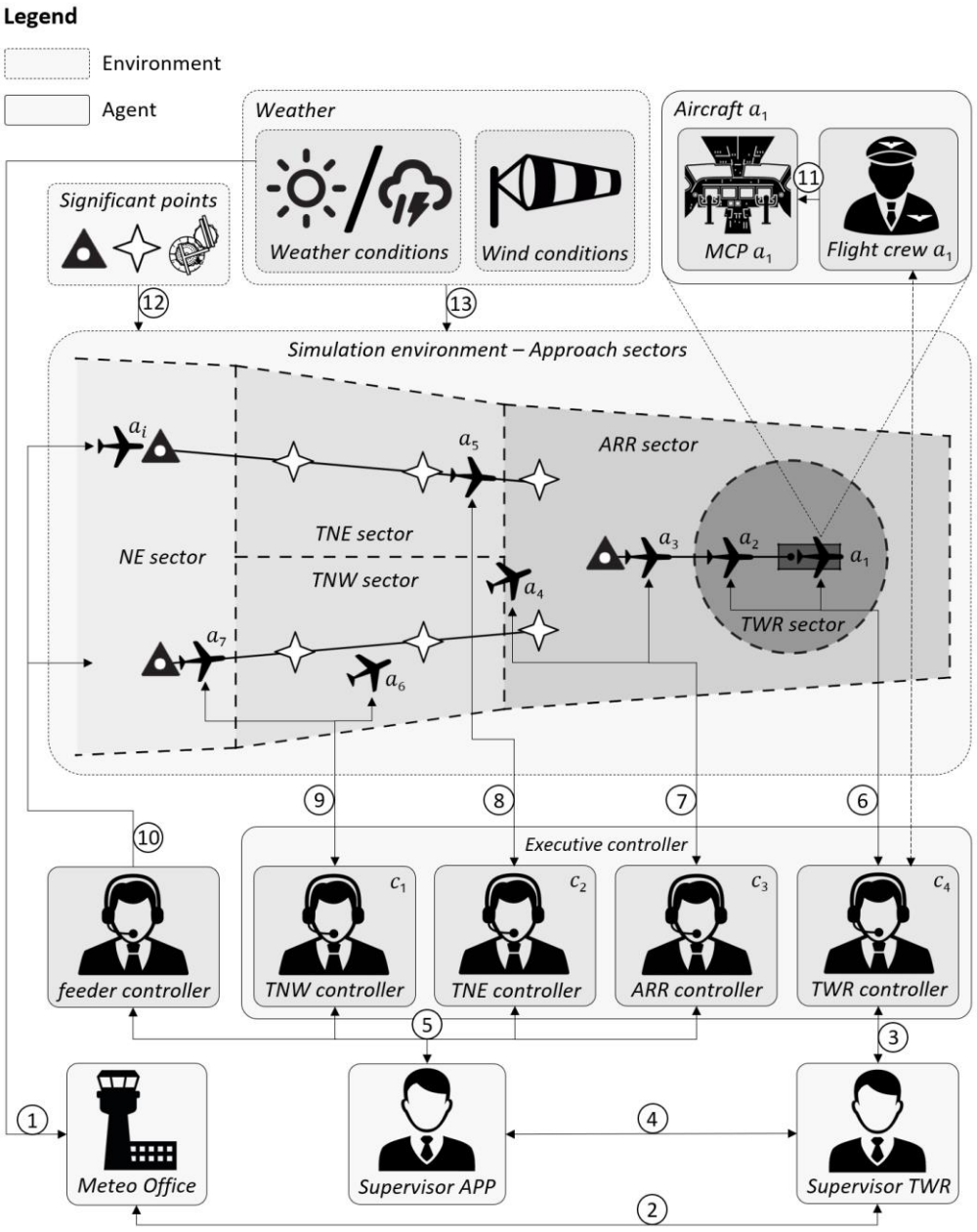


Figure 6 – Schematic overview of the agents, the environment and the interactions as considered in the formal quantitative agent-based model

5.2 Environment

Approach sectors and STAR procedures

The simulation environment comprises the northern approach sectors of Rome Fiumicino, which have been introduced in the previous chapter (i.e. TNE, TNW and ARR). By considering these three sectors the simulation environment is of sufficient size to model the complete arrival and approach segments, which start at the entry point of each modelled STAR procedure (i.e. XIBIL and RITEB) and end at the threshold of runway 16L. This means that the NE sector is not considered part of the simulation environment. The dynamics of the arrival operations in the NE sector are therefore simulated by the modelled behaviour of the feeder controller. The STAR procedures that have been introduced in the previous chapter serve as the baseline operations during arrival and connect the TNW/TNE sectors with the ARR sector. Both STAR procedures are collected in the set $\mathbb{S}$, where $\mathbb{S} = \{S_1, S_2\}$. Here S_1 represents the XIBIL2A procedure and S_2 the RITEB2A procedure. Within each STAR procedure there is one holding pattern considered, i.e. $\mathbb{H}_{S_1} = \{H_1\}$ and $\mathbb{H}_{S_2} = \{H_2\}$.

w_{index}	Description
$w_{S,k}$	Sequence of interconnected fly-by waypoints defining STAR procedure S
$w_{S,\mathrm{MO}}$	Begin point of the *base segment* (*outbound*) of STAR procedure S (*XIBIL2A: RF424, RITEB2A: RF444*)
$w_{S,\mathrm{MI}}$	End point of the *base segment* (*inbound*) of STAR procedure S (*XIBIL2A: RF426, RITEB2A: RF446*)
$w_{S,\mathrm{R}}$	Waypoint at which aircraft can re-join and re-operate STAR procedure S (*XIBIL2A: GIPAP, RITEB2A: VAKAB*)
$w_{S,\mathrm{HO}}$	Desired handover point from TNW/TNE to ARR (*XIBIL2A: USIRU, RITEB2A: ESALU*)
$w_{S,\mathrm{IAF}}$	Initial Approach Fix (IAF) (*XIBIL2A: SUVOK, RITEB2A: EXAMA*)
w_{IF}	Intermediate Fix (IF) (*OXERU*)
w_{THR}	Runway threshold (*LIRF*)
w_{HST1}	First HST (*DG of RWY 16L*)
w_{HST2}	First HST (*DH of RWY 16L*)
$w_{I_2,S,k}$	Vector points that are used to resolve conflicting situations near the lateral profile of the STAR procedure
$w_{I_4,S}$	Vector points that are used to resolve conflicting situations near the *base segment* of the STAR procedure
$w_{I_6,S,N}$	Vector points that are used to resolve conflicting situations during the merging operations relative to w_{IF}
$w_{I_8,S}$	Vector points that are used to resolve conflicting situations near w_{IF}
$w_{S,H,\mathrm{I}}$	Holding inbound fix (*XIBIL2A: GOPOL, RITEB2A: RAVUX*)
$w_{S,H,\mathrm{O}}$	Holding outbound fix

Table 5 – Descriptions of the multiple significant points that are considered in the model

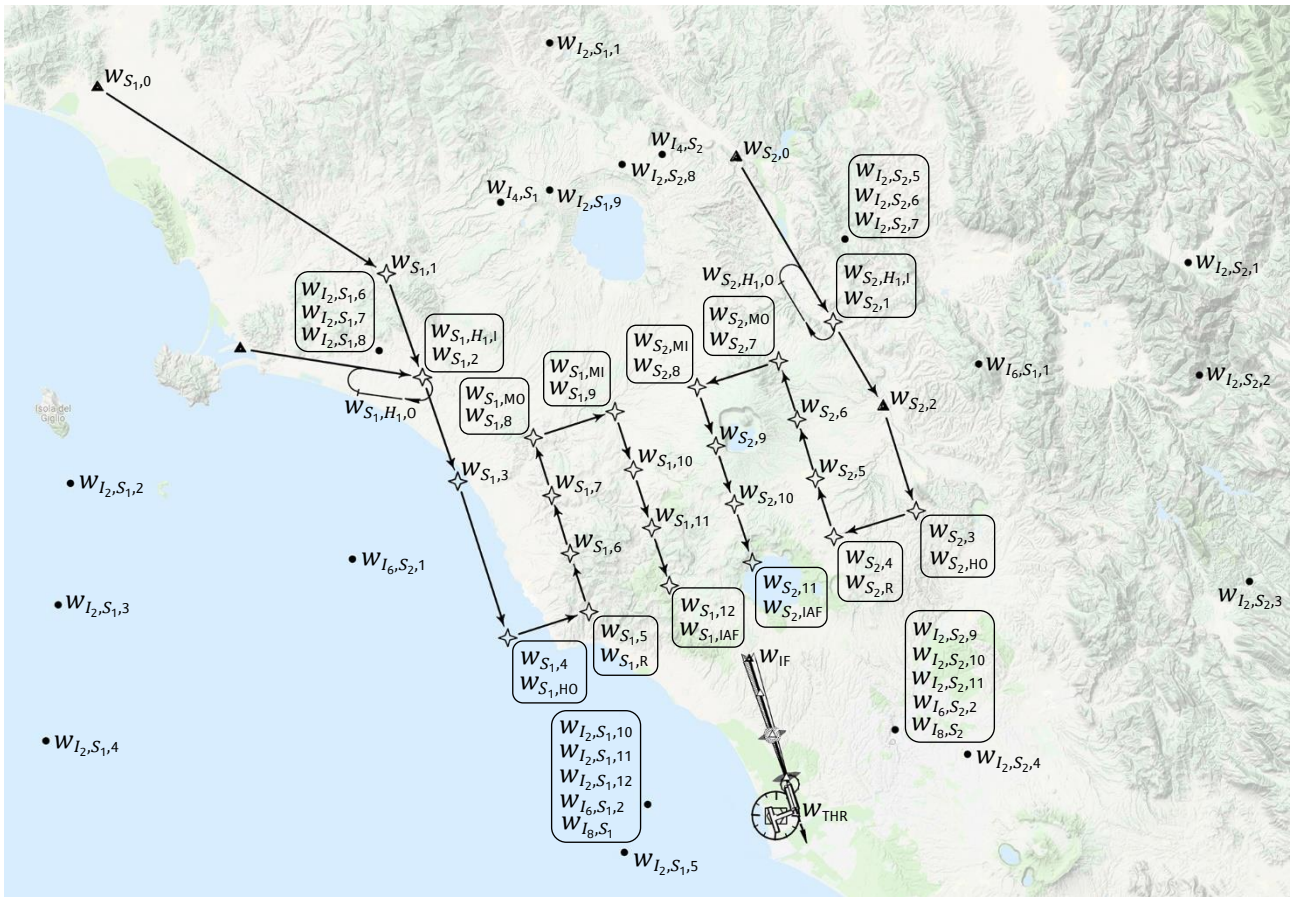


Figure 7 – Visualization of the multiple significant points that are considered in the model

Significant points

The airspace sections that are taken into account contain many (three dimensional) points in space that are used to define the position and orientation of aircraft agents within the simulation environment. All the points that are used to mark essential locations within the simulation environment are called *significant points*, denoted by "w_{index}". This notation in combination with some specified "$index$" is used to indicate the specific function of the significant point that is considered. Table 5 provides brief descriptions of the significant points that are considered in this resilience study (where $k \in \mathbb{N}_0$, $S \in \mathbb{S}$, $H \in \mathbb{H}_S$ and $N \in \{1, 2\}$). More specific detail of these significant points can be found in appendix C.3.2. The four different types of vector points in table 5 are further clarified in section 5.3.3.4. The locations of the modelled significant points are visualized in figure 7. This figure shows the map that is used in the simulation environment, containing the various, merged and on scale arrival procedures.

Trombone segments

The previous chapter introduced the tromboning technique that is applied at Rome Fiumicino to assist the ARR controller in establishing a proper sequence of merging traffic towards the IF. This merging practice is facilitated by the shape and the structure of the so-called trombone segments in the XIBIL2A and RITEB2A STAR procedures. The structure and the functionalities of the trombone segments as applied in the approach procedures for Rome Fiumicino are for this reason also considered in the formal agent-based model. Each STAR procedures is herein decomposed into smaller segments, where each separate segment has its own functionality, characteristics and resulting approach behaviour. The controller agent is able to observe the current location of an aircraft relative to each of these separate segments. These observations will eventually determine what type of instructions are or become required.

The rectangular shaped trombone segments can be decomposed into the *downwind segment*, the *base segment* and the *final segment* (figure 8). The *downwind segment* consists of the legs between $w_{S,\mathrm{R}}$ and $w_{S,\mathrm{MO}}$, which is used by aircraft to gain the required spacing before the merging practice becomes feasible. The *base segment* is defined as the leg between $w_{S,\mathrm{MO}}$ and $w_{S,\mathrm{MI}}$. Aircraft that are positioned on the *base segment* are therefore no longer flying explicitly away from w_{IF} and w_{THR}. The *final segment* at last is defined to be the part of the STAR procedure between $w_{S,\mathrm{MI}}$ and $w_{S,\mathrm{IAF}}$, i.e. the legs that are oriented again in the direction of the airport.

Figure 8 also indicates the term *saturation* and the corresponding boundary. As already described in section 4.3.2 *saturation* is defined as the situation when one or more aircraft are positioned on or near the *base*- and/or *final segment* of the STAR procedure. This term is used in the model to indicate the situation when the required spacing between the to be merged aircraft is more difficult to gain. This is the result of the orientations of the *base*- and *final segments* relative to w_{IF}. Some of the modelled properties of the controller agents are defined by their ability to observe the so-called *saturated* trombone segments. The (ARR) controller agent is for instance able to fictitiously extent the trombone segments using vector instructions once it notifies *saturation* (figure 8). The next section will elaborate further on the modelled controller actions that relate to *saturation*.

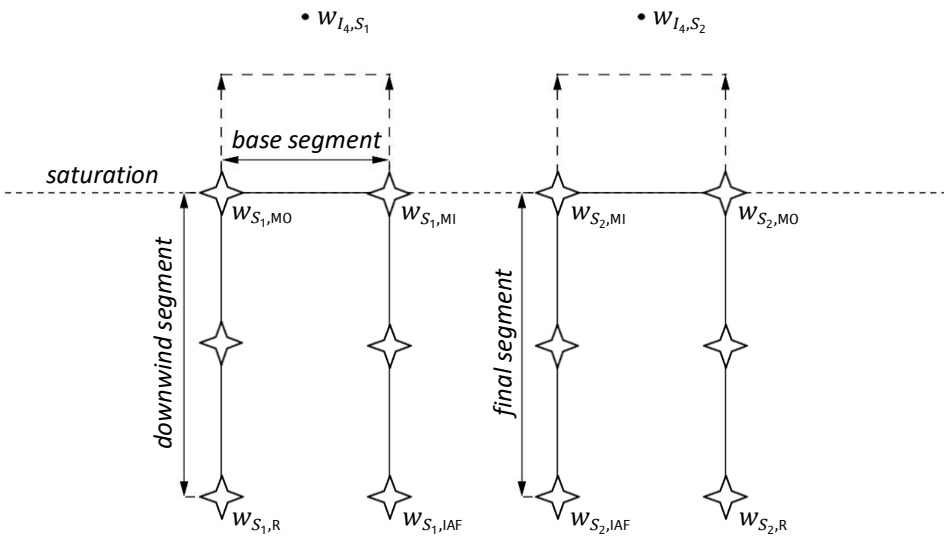


Figure 8 – Visualization of the defined downwind-, base- and final segments in the trombone segments

Wind conditions

A wind model has been incorporated to take into account the effects of varying wind conditions (i.e. direction and speed) on the groundspeed of aircraft during approach. Forecasted wind directions and wind speeds for Rome Fiumicino have been obtained from *windy.com* due to the large overall detail and wind data density. The data is chosen such that the wind direction at sea level enables headwind landings. Each altitude layer has a unique and 'random' wind speed and direction. The wind conditions are updated at the beginning of each simulation run. The formal specification of the incorporated wind model can be found in appendix C.3.3.

Weather conditions

The model considers two types of weather conditions at the airport, either normal weather conditions or bad weather conditions. The normal weather conditions represent calm weather with good visibility, i.e. aircraft approach and landing performance is not affected by the weather condition at all. The bad weather conditions can be contextualized by for instance local thunderstorms, storm cells, snowstorms etc. These bad weather conditions result in a decreased runway capacity due to contamination and reduced visibility on the runway system. The modelled bad weather disturbance is only applicable at the airport. The weather conditions during arrival and approach towards Rome Fiumicino can therefore be described by normal weather conditions.

The two discrete weather states allow to model a sudden bad weather disturbance (and recovery) at the airport (appendix C.8.1.1). The specific time point at which the weather conditions will deteriorate is chosen such that a stabilized sequence of approaching traffic can be, and has been established within the simulation environment. A reduced runway and throughput capacity is applicable during the bad weather disturbance. After some time the weather conditions will normalize again, and with that the runway capacity. The specific time point at which this will occur is chosen such that the dynamics and the resilient capacities of the overall socio-technical system due to the bad weather disturbance can be measured. The normalized weather conditions allow to measure the restorative capacities of the arrival and approach operations at Rome Fiumicino.

5.3 Executive controller

The model considers four different executive controller agents, i.e. $\mathbb{C} = \{c_1, c_2, c_3, c_4\}$. Three of those controller agents can be described as approach controllers (APP), which are the TNW (c_1), TNE (c_2) and ARR (c_3) controllers. Each of these approach controllers is responsible for the safe, orderly and expeditious flow of traffic within a specific TMA airspace section. The fourth identified controller agent is the TWR (c_4) controller, which is responsible for aircraft within a specific area around the airport. The modelled behaviour and properties of the executive controller agents are described using the following structure:

- Section 5.3.1: describes the number of tasks and responsibilities that have been assigned for each agent;
- Section 5.3.2: describes the model constructs that have been used to contextualize the properties of the controller agent;
- Section 5.3.3: describes the generic strategies and concepts that have been applied to allow the modelling of realistic and functional controller actions;
- Section 5.3.4: describes the specific controller instructions that have been modelled;
- Section 5.3.5: describes the specific operational states where aircraft may operate in as the result of the modelled instructions and which can be perceived by the controller agent;

5.3.1 Tasks and responsibilities

Each controller has its own specific responsibilities as defined by the airspace section where it is responsible for. This section will describe the (basic) tasks have been identified and modelled for each executive controller agent.

TNW and TNE controllers

The TNW and TNE controllers are responsible for air traffic within the TNW and TNE sectors respectively. Both controllers share the same set of tasks, since they are both responsible for the same type of airspace sectors. The set of identified tasks for the TNW and TNE controllers are:

- Monitor aircraft spacing in sequence within the TNW/TNE sector
- Monitor traffic situation in general
- Manage throughput capacity, as delivered to the ARR sector
- Coordinate with Supervisor APP about desired throughput capacity
- Handover to ARR controller
- Apply vectoring to adjust spacing between aircraft
- Provide vector back to route
- Provide navigation clearance to route or waypoint
- Provide heading (vector), speed and/or altitude instruction
- Initiate/terminate holding operations
- Manage altitude of aircraft within holding stack

ARR controller

The ARR controller is responsible for air traffic within the ARR sector. The set of identified tasks for the ARR controller are:

- Monitor aircraft spacing in sequence and during merging within the ARR sector
- Monitor traffic situation in general
- Manage throughput capacity, as delivered to the TWR sector
- Coordinate with Supervisor APP about desired throughput capacity
- Handover from TNE and TNW controllers
- Handover to TWR controller
- Establish arriving traffic on the final approach for runway 16L
- Merge incoming traffic from the TNW and TNE sectors
- Apply vectoring to adjust separation between aircraft
- Provide vector back to route
- Provide navigation clearance to route or waypoint
- Provide heading (vector), speed and/or altitude instruction

TWR controller

The TWR controller is responsible for air traffic within the TWR sector, i.e. for the operations on runway 16L and for airborne aircraft within the area of responsibility. The set of identified tasks for the TWR controller are:

- Monitor aircraft separation in sequence during final approach
- Manage runway capacity, as delivered at the runway threshold
- Coordinate with Supervisor TWR about desired runway capacity
- Handover from ARR controller
- Handover to GND controller
- Provide landing clearance
- Instruct go-around

5.3.2 Model constructs

The tasks and responsibilities as identified in the previous section can be contextualized by the contents of a number of interconnected and interdependent aspects and processes. Each of these specific aspects and processes can be described by so-called model constructs. This section will describe the model constructs that have been used to specify the controller agents, and which allow to model the set of identified tasks and responsibilities. The model constructs that have been used to specify the controller agent and the interactions between them are visualized in figure 9 below. The structure of these visualized model constructs is based on the situation awareness model of Endsley [1] and the human operator agent model of Stroeve and Everdij [2]. Input from other agents occurs via the situation awareness model construct, while output to other agents occurs via the task execution model construct.

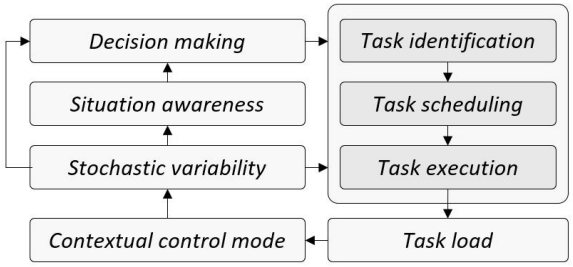


Figure 9 – Model constructs of the controller agent and their interactions

The subsections below will provide brief descriptions of the functionalities of each used model construct and the coherence between these model constructs. More practical context of these model constructs is provided hereafter in section 5.3.3. This practical context is not provided in this section, since it relates to multiple combinations of model constructs. A more formal and mathematical description of each model construct is provided in the corresponding sections in appendix C.4. The modelling elements that are used in this appendix resemble the structure and interactions as visualized in figure 9.

Situation awareness

The situation awareness of the controller agent describes the perceived traffic situation in its airspace sector, i.e. the situational information related to a_i. The situation awareness of the controller agent is modelled using a set of statecharts that each describe a specific situation or situational element in terms of observation, reasoning and memory (appendix C.4.5.1). The updating process of the situation awareness of each controller is modelled with a stochastic periodic event (appendix C.4.4.1).

The situation awareness of each controller is fed by the information that is provided on its radar screen. A radar screen provides for each aircraft in a specific airspace section its corresponding position, flight direction, altitude, airspeed, and aircraft type. This information allows the controller to observe and to determine:

- the separation between aircraft;
- the position and orientation of each aircraft relative to other aircraft and the various significant points;
- the operational state of each aircraft;
- the correct sequence of aircraft;
- the current progress of an aircraft in its flown arrival and approach procedures.

Decision making

The situation awareness of a controller allows it to comprehend the observed situation and to decide what actions are required. The controller's decision making is in the model therefore mostly related to the determination of the feasibility and necessity of the set of identified instructions, which will be described in section 5.3.4. The specific contents of these instructions are defined by the reasoning and intelligence of the controller agent. Decision making is for this reason also related to the ability of the controller agent to derive appropriate plans and actions as a reaction to incoming beliefs. The modelled controller agent is able to determine for each considered instruction (if applicable):

- its necessity (e.g. the need for a vector instruction to resolve a conflicting situation);
- its feasibility (e.g. the ability to determine if an aircraft has to be handed over);
- its contents (e.g. the specific and desired heading direction, altitude and airspeed);

See appendix C.4.4.1 for a formal description of how the necessity and feasibility of each instruction are defined.

Task identification

Task identification represents the evaluation phase in which the controller periodically checks a number of criteria that should hold in order to identify a specific task or instruction. Task identification is therefore dependent on the results of the situation awareness and decision making model constructs. See appendix C.4.5.2 for a formal description of the modelled task identification process, i.e. the complete set of conditions that should be evaluated true in order to identify each modelled instruction.

Task scheduling

Task scheduling represents the phase in which the controller evaluates all identified tasks by priority. Task scheduling follows therefore task identification and is meant to select the task/instruction with the highest priority/urgency from the complete list of identified tasks. See appendix C.4.5.2 for a formal description of the modelled task scheduling process.

Task execution

Task execution represents the phase in which the controller is executing (i.e. instructing) the identified and scheduled task. The controller is herein contacting the flight crew of the aircraft where the instruction belongs to, after which the specific contents of the instruction are communicated. The contents of each instruction are defined in the decision making model construct. See appendix C.4.5.3 for a formal description of the modelled communication between the controller and the flight crew agent, and for a description about the contents of each modelled instruction.

Task load

The task load model construct describes the current task (work) load of the controller. The number or instructions that a controller has provided in the recent past is used in the model to quantify the workload of the controller. A relatively large number of recently provided instructions in a short time period represents a high workload, whereas a relatively small number of instructions represents a less intense or just a normal workload. The task (work) load of the controller is used to define the specific contextual control mode in which the controller is operating. See appendix C.4.4.2 for a formal description of how the workload of each controller is defined.

Contextual control mode

Each controller is modelled to act in two different control modes, either in the tactical- or in the opportunistic control mode. The tactical control mode represents the state of the controller in which it has a relatively large planning horizon to act, which allows for normal operating and acting performances. The opportunistic control mode on the other hand represents the state of the controller in which workload of the controller is more intense, which leads to a relatively small planning horizon. The operating and acting performances of the controller can be considered more rapid, chaotic and spontaneous and less efficient and accurate when compared to those in a tactical control mode.

The model takes into account different performance characteristics for each control mode. These different characteristics are expressed in terms of available time to act, recognise and decide, and in terms of the accuracy of the vectoring practice. See appendix C.4.5.4 for a formal description of the implications of the tactical- and opportunistic control mode. The controller agent is modelled to operate in an opportunistic control mode if it has provided more than 20 instructions in the past 10 minutes. The controller agent may again operate in a tactical control mode if less than 15 instructions have been provided in the past 10 minutes. The used gap between both trigger values ensures that a controller will not be operating in a specific control mode for an unrealistic short period of time.

Stochastic variability

The stochastic variability in the performance of the controller agent is modelled with normal and lognormal probability density functions. These functions define the amount of time that is required by the controller agent to identify, schedule and execute tasks and to update its situation awareness. Secondly, the accuracy of the vectoring practice is also described by stochastic variability.

5.3.3 Applied concepts to model basic controller techniques

Each of the four controller agents is assigned its own set of tasks and responsibilities (section 5.3.1). These tasks and responsibilities are in general meant to manage the complex and dynamic approach operations in a safe, efficient and expeditious manner. The strategies that are applied by controllers in the context of the number of assigned tasks are however not very evident. This is because of the many (situational) factors that affect the decision-making process. Safe and efficient approach operations are the result of the controllers' ability to comprehend the observed traffic situation (spatial awareness) and its ability to anticipate on these observations in the right way. The specific contents of the instructions (i.e. heading, speed and altitude) are therefore highly dependent on the actual traffic situation. For this reason a number of robust strategies and practices have been applied in the model that allow the controller agent to perform the number of assigned tasks in a realistic manner and under all type of settings and circumstances. The subsections below will each address a specific strategy or practice that is considered for the controller agent.

5.3.3.1 Aircraft referencing

The simulation environment contains many aircraft agents. Each aircraft has herein its own unique position in space, a specific operational state and a varying airspeed, altitude and heading direction. The controller agent uses this situational information to decide if actions are required, and if yes, what actions to undertake. However, with so many aircraft present in the simulation environment it is difficult to model actions for each individual aircraft taken into account the various orientations and future states of the other surrounding aircraft. The majority of the modelled instructions are directly or indirectly related to separation and spacing. The controller agent should therefore be able to identify the various aircraft pairings in the simulation environment in order to decide if the corresponding spacing is appropriate (again). In order to facilitate this type of intelligence the model considers a number of sets and so-called *reference aircraft*. These sets and *reference aircraft* are used to define the multiple sequences of aircraft within the simulation environment.

$a_{j,i}$	Description
$a_{1,i}$	the aircraft that is flying in front of a_i while operating the STAR procedure or other type of operations near the lateral profile of the STAR procedure.
$a_{2,i}$	the aircraft that is flying in front of a_i during the merging and sequencing practice towards w_{IF}.
$a_{3,i}$	the aircraft that is flying in front of a_i while operating the intermediate- or final approach segment.
$a_{4,i}$	the aircraft where a_i is to be sequenced behind relative to $w_{S,\text{R}}$.
$a_{5,i}$	the aircraft where a_i is to be sequenced in front of relative to $w_{S,\text{R}}$.

Table 6 – Descriptions of the so-called reference aircraft

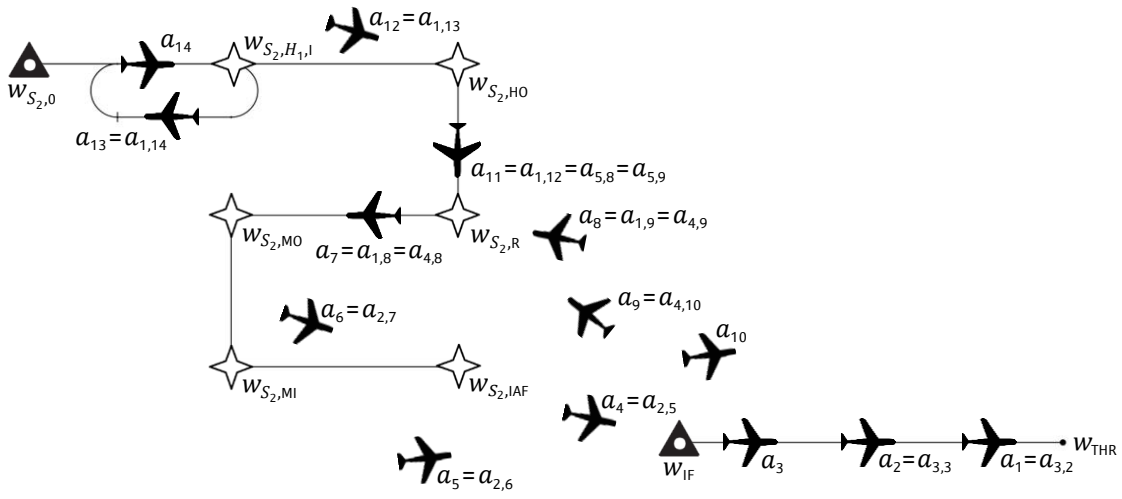


Figure 10 – Visualization of the applied "aircraft referencing" strategy

The various sets that are used for this purpose are formally described in appendix C.4.1. Each of these sets contains a dynamic number of ordered aircraft agents that are used to define the specific sequence of aircraft agents in a certain flight operation or -phase. The applied *aircraft referencing* strategy makes use of these ordered sets to determine the multiple aircraft pairings. The principle of this applied strategy is that each aircraft agent (i.e. a_i) is referenced ("connected") to a fixed set of surrounding aircraft (if applicable) (i.e. $a_{j,i}$, where $j \in \{1,2,3,4,5\}$), where each *reference aircraft* is related to a specific type of operation. The *aircraft referencing* strategy allows to define the necessity and feasibility of the modelled instructions in an efficient manner, and allows to control at all times the very diverse and dynamic flight trajectories that emerge during simulation. Table 6 explains the purpose of each of the five considered *reference aircraft*. A typical example of the used *aircraft referencing* strategy is visualized in figure 10. The upcoming sections will further clarify the use of this strategy and the purpose of each *reference aircraft*.

5.3.3.2 Determine aircraft positioning

The ability to observe and to interpret the current traffic situation in terms of the multiple aircraft positions and orientations is a key property of the controller agent. This position information is required to define the necessity and feasibility of the identified tasks and to define the contents of the instructions. The multiple interconnected waypoints of both STAR procedures (i.e. $w_{S,k}$) have been used to define the situation awareness of the controller agent about the position and orientation of each aircraft agent in the simulation environment. This situation awareness is defined by the current waypoint number of each aircraft agent, i.e. the k^{th} waypoint of STAR procedure S where each aircraft is referenced to. The waypoint number where an aircraft is referenced to is dynamic and will change over time while the respective aircraft is moving through the simulation environment. A large part of the arrival operations of aircraft in the simulation environment can therefore be described by the waypoint numbers, which define the progress of an aircraft along the STAR profile. Figure 11 visualizes the usage of these dynamically changing waypoint numbers to specify the position and orientation of an aircraft agent in the simulation environment. Each aircraft is herein referenced to a specific waypoint.

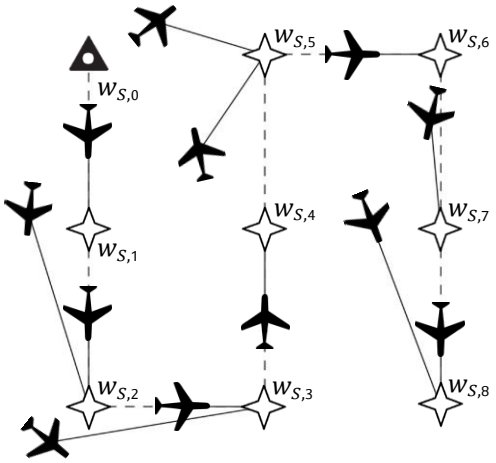


Figure 11 – Visualization of the usage of dynamically changing waypoint numbers to relate the position and orientation of aircraft agents to

5.3.3.3 Separation practice

An air traffic control service is defined as a service provided for the purpose of preventing collisions between aircraft, and expediting and maintaining an orderly flow of air traffic [26]. Both responsibilities are managed by maintaining a sufficient and safe separation between aircraft. One of the core properties of the controller agent is therefore its ability to observe the (longitudinal) spacing between the many aircraft and to detect conflicts as the result of loss of separation. The *aircraft referencing* strategy is for this reason applied to keep track of the separation distances between the multiple aircraft.

Separation distances

Five different types of separation distances are considered in the model that allow the controller agent to detect separation conflicts and to maintain the desired throughput capacity (figure 12). The applied separation strategy is identical for each *reference aircraft*. The five different types of separation distances can be described as follows:

- *Actual spacing*: the actual (observed) longitudinal spacing between a_i and $a_{j,i}$;
- *Separation minima*: the ICAO wake turbulence separation minima between a_i and $a_{j,i}$;
- *Separation buffer (type I)*: additional separation distance that is added to the *separation minima* to steer upon a desired throughput capacity (see section 5.3.3.6), and to model the reduced separation minima that apply during final approach;
- *Desired spacing*: the (minimum) spacing between a_i and $a_{j,i}$ that should be maintained to comply with the desired throughput capacity;
- *Separation buffer (type II)*: additional separation distance that is added to the *desired spacing* to model the inaccuracies of the controller's vectoring strategy (see section 5.3.3.5).

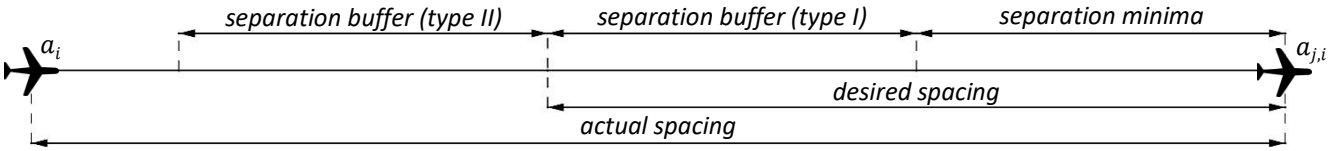


Figure 12 – Visualization of the separation distances that are used to model the separation practice of the controller

Conflicts

The first four separation distances allow the controller agent to detect and to resolve the following types of conflicts between a_i and $a_{j,i}$ (if applicable):

- *Separation minima conflict*: the situation in which the *actual spacing* is smaller than the *separation minima*.
- *Desired spacing conflict*: the situation in which the *actual spacing* is smaller than the *desired spacing*.

Both type of conflicts are resolved using an *outbound vector* (see next section) or go-around instruction. The *separation minima conflict* is assigned a higher priority than the *desired spacing conflict*.

5.3.3.4 Vectoring strategy (outbound)

The type of vector that is used to enlarge the longitudinal spacing between aircraft is in this report defined as an *outbound vector*. These type of vectors become required when the controller agent observes either a *separation minima conflict* or a *desired spacing conflict*. A controller will in such situation instruct the conflicting (succeeding) aircraft a vector. The direction of this vector should be such that the required spacing can be gained efficiently and such that the resulting flight path of the (to be) vectored aircraft will not interfere with the arrival and approach operations of other nearby aircraft. The specific direction of an *outbound vector* is in practice defined by the controller's situation awareness, reasoning and experience. A controller takes herein a lot of related factors into consideration, such as the current trajectories of nearby aircraft, the estimated future states of the nearby aircraft in terms of position and type of operation, the structure of the approach procedures that apply, etc. This makes that the vectoring practice of a controller cannot be explained by just one single and general description. The modelling of an efficient, realistic and feasible vectoring strategy can therefore be considered a challenging task.

In order to allow for realistic and feasible vector operations the model makes use of strategically placed *vector points*. These fictitious points are used by the controller agent to vector aircraft to. The locations of these *vector points* have been determined manually and are chosen such that the vector operations to these points are not likely to conflict with the flown trajectories of other aircraft. This is achieved by allocating the multiple *vector points* such that the resulting vector operation will guide aircraft away from the dense airspace sections and away from the profiles of the modelled IAPs.

The specific *vector point* that is to be used by the controller agent is dependent on the specific operation and location of the aircraft that is going to be vectored. Table 7 below provides descriptions of the purpose and the features of the four types of vector points that have been considered in the model. The locations of these vector points have already been visualized in figure 7.

w_{index}	Purpose and features
$w_{I_2,S,k}$	Vector points that are used to resolve conflicting situations near the lateral profile of the STAR procedure. They have been assigned by taking into account the heading directions of each leg within the STAR procedure, the locations of the waypoints ($w_{S,k}$) and the structure of the procedure in total. Each leg/waypoint within the modelled STAR procedures has therefore its own vector point assigned, i.e. $w_{I_2,S,k}$. All these specific type of vector points have default (fixed) coordinates at the start of each simulation run (as shown in figure 7). However, the coordinates of these vector points are dynamic and will be updated once and after the controller has vectored an aircraft towards the respective point. These dynamically changing coordinates allow for so-called "*opening*" vectors to achieve the desired spacing even quicker. See appendix C.4.3.2 for a formal specification of this dynamic vectoring practice.
$w_{I_4,S}$	Vector points that are used to resolve conflicting situations near the *base segment* of the STAR procedure. They have been assigned such that they serve as an extension of the trombone segment by fictitiously shifting the *base segment* towards the north and in a direction parallel to the *downwind segment*.
$w_{I_6,S,N}$	Vector points that are used to resolve conflicting situations during the merging operations relative to w_{IF}. They have been assigned by taking into account the wide variety of operations and the corresponding flight trajectories in the ARR sector. Two vector points have been assigned for each STAR procedure (i.e. $w_{I_6,S,N}$) to allow for realistic vector operations in this dense airspace sector. The position of the conflicting aircraft determines which of the two vector points is to be used.
$w_{I_8,S}$	Vector points that are used to resolve conflicting situations near w_{IF}. They have been assigned such that the conflicting aircraft is able to leave the dense ARR sector away from both $w_{S,IAF}$ and w_{IF} and in a direction that is somewhat parallel to the direction of the localizer of runway 16L.

Table 7 – Purpose and features of the four types of vector points

5.3.3.5 Vectoring strategy (inbound)

The type of vector that is used to resume/recover the standard arrival and approach operations once the controller considers the longitudinal spacing between aircraft sufficient (again) is in this report defined as an *inbound vector*. All the modelled *inbound vectors* are directed towards one of the fly-by waypoints that are contained in the simulation environment (i.e. $w_{S,k}$ and w_{IF}). These so-called direct-to vectors are considered because of the application of RNAV in the approach sectors of Rome Fiumicino.

Where to instruct an *inbound vector* to?

An *inbound vector* is in general the result and continuation of an *outbound vector*. Most of the modelled *outbound vectors* originate from one of the legs of the modelled STAR procedures. Aircraft are herein vectored away from the STAR profile towards a specific vector point (i.e. $w_{I_2,S,k}$). These *outbound vectors* result in very dynamic flight behaviour due to the many possible directions and trajectories that can be flown. Each vectored aircraft is herein flying a unique trajectory with a continuously changing position and orientation relative to the various significant points. As the result of this continuously changing position and orientation the most 'logical' waypoint to direct an *inbound vector* to may change over time as well. This means that a controller will not necessarily need to stay focused on one particular waypoint to decide upon the feasibility of an *inbound vector*. The controller agent is therefore modelled with some intelligence that defines what waypoint the controller should consider to assess the feasibility on and to direct the *inbound vector* to. The output of this modelled intelligence is visualized in figure 13. This figure shows how the reasoning of the controller in terms of preferred vector direction changes over time as the result of a changing position of a_i. This is because a controller will always opt to vector an aircraft towards a waypoint that allows aircraft to resume their originally flown STAR procedure in preferably the most fluent and (relatively) fastest way. See the "Determine the waypoint number $k^{a_i,c}$ during *vector outbound STAR* operations" phase in appendix C.4.4.1 for a mathematical description of how this specific type of reasoning has been modelled. The used algorithm evaluates the position and orientation of the to be vectored aircraft relative to the multiple $w_{S,k}$ to decide which waypoint (i.e. $w_{S,k}$) is the most logical for an *inbound vector*. The algorithm selects the waypoint that allows for acceptable interception angles and a relatively short *inbound vector* distance to resume the originally flown STAR procedure.

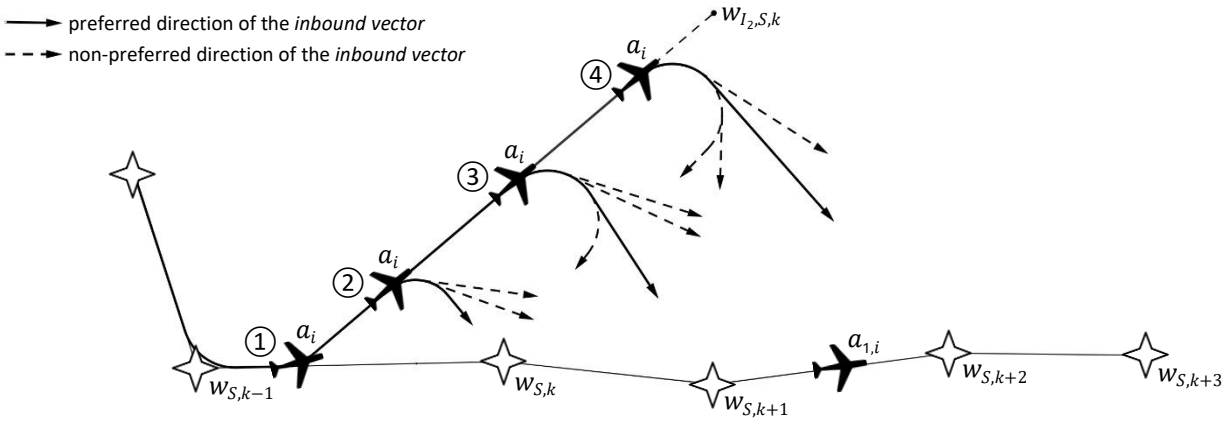


Figure 13 – Visualization of how the preferred direction of an inbound vector changes over time due to a changing position of a_i

When to instruct an *inbound vector*?

An aircraft is only instructed an *inbound vector* towards one of the fly-by waypoints once the controller estimates the observed spacing between the respective aircraft pairing as sufficient. Secondly, an *inbound vector* is only instructed if the to be vectored aircraft is not expected to be in conflict with other aircraft once it will intercept the waypoint. The feasibility of an *inbound vector* is therefore also dependent on the ability of the controller to estimate the future positions of aircraft relative to the waypoint of interest. The provision of well-timed vector instructions is because of these many related factors and uncertainties a challenging task. The varying wind conditions and the relative differences between the dynamic states of the nearby aircraft (i.e. airspeed, altitude) do even further contribute to these challenges. The model does therefore concern the implications of vectoring accuracy, i.e. the performance of the controller agent in vectoring aircraft towards a specific waypoint and in sequencing aircraft relative to the respective waypoint.

The vectoring accuracy is modelled as a function of the distance between the to be vectored aircraft and the waypoint where the aircraft is to be vectored to, and secondly the current control mode of the controller. See appendix C.4.3.1 for a formal description of this specific function. This function is executed each time when the controller agent is updating its situation awareness. It calculates and returns the so-called *separation buffer (type II)* using a normal probability density function. The mean of the used normal distribution is kept slightly negative to take into account the distance that is flown additionally in the time period between defining the feasibility of an *inbound vector* and communicating the *inbound vector* to the flight crew. The exact size of the *separation buffer (type II)* is defined by the size of the used standard deviation in the normal probability density function. The size of the standard deviation is defined by the two function arguments that are described above, i.e. control mode and vector distance. The function in appendix C.4.3.1 applies larger standard deviation values in the situation when the performance of the controller can be described by an opportunistic control mode and when large vector distances apply, which in turn will result in (on average) larger *separation buffers (type II)*. Note that the calculated *separation buffer (type II)* can both be positive and negative because of the used values for the mean and the standard deviation. The controller agent considers an aircraft eventually available for an *inbound vector* once the *actual spacing* between the aircraft pairing is observed to be larger than the *desired spacing + separation buffer (type II)* (figure 14). The specific value of the standard deviation defines therefore the accuracy of the to be instructed *inbound vector*, i.e. the size determines the precision at which aircraft will intercept the fly-by waypoint taken into account the positions of other aircraft. The size of the *separation buffer (type II)* can affect the vectoring accuracy in two ways. A too large (positive) *separation buffer (type II)* will negatively affect the maintained and desired throughput capacity. A too small (negative) *separation buffer (type II)* will on the other hand increase the risk of a *desired spacing conflict* or a *separation minima conflict* while operating the *inbound vector* or after waypoint interception.

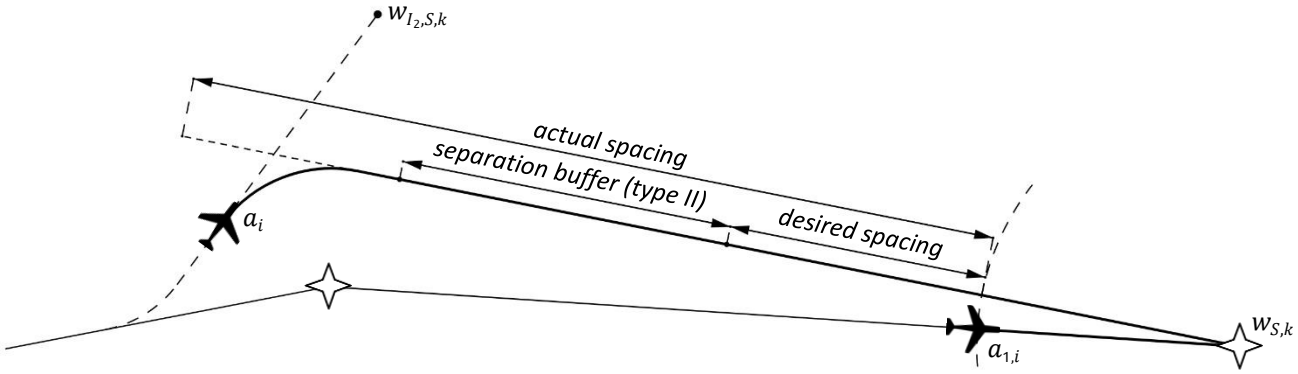


Figure 14 – Visualization of the application of the separation buffer (type II) to take into account the inaccuracies and uncertainties of the controller agent in deciding upon the feasibility of inbound vectors

Each time when the controller agent has instructed an *inbound* (or *outbound*) *vector* to a_i the model imposes a time period in which the controller cannot instruct a new vector operation to a_i. This time period makes sure that a controller will not immediately 1) instruct an *inbound vector* when it has just instructed an *outbound vector*, and 2) intervene when it has just instructed a_i a too tight *inbound vector* towards one of the fly-by waypoints. This modelled memory prevents therefore the provision of too many vector instructions to the same aircraft in a relatively short time period. See appendix C.4.5.1.3 for a formal description of the statechart that is used for this purpose.

5.3.3.6 Throughput capacity

Each approach controller is responsible for establishing and maintaining a stable sequence of approaching traffic that is eventually delivered with a certain rate to the next controller. This rate can be described as the throughput capacity, which is defined as the number of aircraft that are delivered at a desired location in a given time period. The specific throughput capacity that is to be maintained in the approach sectors is dependent on the current runway capacity at the airport, and is defined by the Supervisor TWR and Supervisor APP. A controller is able to change the throughput capacity in its airspace sector by reducing or enlarging the *actual spacing*.

The ability of a controller to steer upon a certain desired throughput capacity is modelled using specific *time buffers*. A *time buffer* is defined as a specific time-based separation that is added to the *separation minima* between a_i and $a_{j,i}$. These *time buffers* define therefore the size of the to be added *separation buffer (type I)*, and with that the size of the *desired spacing* that should be maintained. By varying these *time buffers* the model can affect the approach behaviour and the flown trajectories in the multiple approach sectors. The specific values of these *time buffers* determine for instance the necessity of vector operations, define the achieved runway capacity, shape the merging operations and determine the relative use of the trombone segments. The *time buffers* are thus also used to change the throughput capacity in reaction to the changed runway capacity.

Throughput capacity is modelled using fixed *time buffers* instead of fixed separation buffers to take into account the effects of varying airspeeds. The *separation buffer (type I)* is therefore defined as the multiplication of a specified *time buffer* with the groundspeed of the succeeding aircraft. This definition makes sure that despite the varying airspeeds each aircraft has to fly the same time period before a *separation minima* conflict takes place. This means that the controller is provided a normalized time period in which it can detect and resolve conflicts. A relatively larger speed difference will therefore not result in a relatively higher risk of conflicts. Figure 15 visualizes an example scenario of how the *separation buffer (type I)* is calculated and defined using *time buffers*.

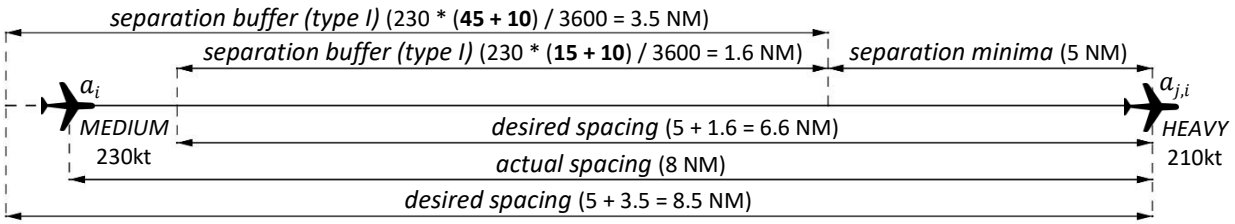


Figure 15 – Visualization of the application of time buffers to steer upon a desired throughput capacity

The model considers two different types of *time buffers* to define the *separation buffer (type I)*. The first type of *time buffer* can be described as the *default time buffer*. This specific *time buffer* is the default time-based separation that is to be added to the *separation minima*. The *default time buffer* has the largest effect on the maintained throughput/runway capacity and is defined by the Supervisor TWR and Supervisor ARR based on the current runway capacity. Each of the two *capacity modes* has its own specific set of *default time buffer* values. Each controller will apply a different set of *time buffer* values once it is informed by the respective supervisor agent about a changed throughput/runway capacity. Figure 15 visualizes a typical example of the use of two different *default time buffers* to steer upon a desired throughput capacity in a certain *capacity mode*. The "15 seconds" *default time buffer* corresponds to the *normal capacity mode* and the "45 seconds" *default time buffer* corresponds to the *reduced capacity mode*. The figure shows that a_i is not having any conflict with $a_{j,i}$ when the controller is operating in a *normal capacity mode*. However, a_i is having a *desired spacing conflict* with $a_{j,i}$ when the controller is operating in a *reduced capacity mode.* In this latter situation the controller will need to enlarge the spacing between a_i and $a_{j,i}$ in order to comply with the desired throughput capacity.

The second type of *time buffer* can be described as the *additional time buffer*. This specific type of *time buffer* is used to account for the effects of compression between aircraft during approach. Compression, i.e. the reduction of separation distance over time is the natural result of the decelerating and descending motions during approach. Separation conflicts may eventually occur if aircraft have not been separated enough initially. Conflicts as the result of compression are even more likely for aircraft of different type and WTC. These varying flight performances during arrival and approach result in either "closing" or "opening" separation distances, depending on the WTC of both the preceding and succeeding aircraft. The *additional time buffer* is therefore meant to reserve some separation distance that will be gradually consumed due to varying descending and decelerating motions of aircraft during approach. The model considers for each combination of WTCs a specific value for the *additional time buffer*. The "10 seconds" *time buffer* in figure 15 represents the *additional time buffer* that corresponds to the WTCs of a_i and $a_{j,i}$. The *default time buffer* and the *additional time buffer* combined will eventually define the size of the *separation buffer (type I)*, and thus the size of the *desired spacing* between a_i and $a_{j,i}$. See appendix C.4.1 for the multiple (*default*/*additional*) *time buffer* variables and parameters that are used in the model.

Each controller applies its own set of *time buffers*. This means that each approach sector has its own specific and resulting throughput capacity. Moreover each controller may apply different *time buffers* to each of the five *reference aircraft.* The use of unique *time buffers* for each aircraft pairing allows to apply different separation criteria to the type of operation where the aircraft pairing corresponds to. The values of the multiple *time buffers* are defined such that the *separation buffer (type I)* corresponding to a_i and $a_{2,i}$ is generally larger when compared to other *reference aircraft*. These settings allow for fluent arrival operations towards the ARR sector, after which the incoming aircraft will be merged with the set merging rate. Section 5.10 will elaborate further on how the overall throughput capacity is managed among the multiple approach sectors.

5.3.3.7 Merging practice

The model considers two types of merging and sequencing practices. The first practice is meant to sequence a_i in between $a_{4,i}$ and $a_{5,i}$ at $w_{S,\mathrm{R}}$ to let a_i re-join and re-operate its originally flown STAR procedure. The second practice is meant to merge the two incoming traffic streams via the XIBIL2A and RITEB2A STAR procedures into one traffic stream at w_{IF} using the tromboning technique. Both practices are based on the same principle that is applied when instructing *inbound vectors*. Figure 16 visualizes a representative traffic situation in the ARR sector as the result of both merging and sequencing practices.

The sequencing practice at $w_{S,\mathrm{R}}$ is the result of unsuccessful merging operations at and near w_{IF}, mostly due to conflicting situations in this dense airspace section. Aircraft that are causing this loss of separation are vectored (*outbound*) towards $w_{I_8,S}$. These type of vectors are required since vectors are in practice no longer instructed once aircraft have already intercepted the localizer course (i.e. after interception of w_{IF}). Separation conflicts during the intermediate- and final approach are generally resolved using go-around instructions only. While operating these specific vectors towards $w_{I_8,S}$ aircraft may be instructed a direct-to towards $w_{S,\mathrm{R}}$ to re-operate the STAR procedure. In this way aircraft are provided a new opportunity for a successful approach towards w_{IF}. This type of *inbound vector* is meant to enhance the resilient capacities of the arrival and approach operations that are considered in this study. Note that these type of operations are in general only flown when an inaccurate merging practice is applied relative to

w_{IF} or when the throughput capacity in the ARR sector is suddenly decreased and a relatively large number of aircraft are soon intercepting w_{IF}.

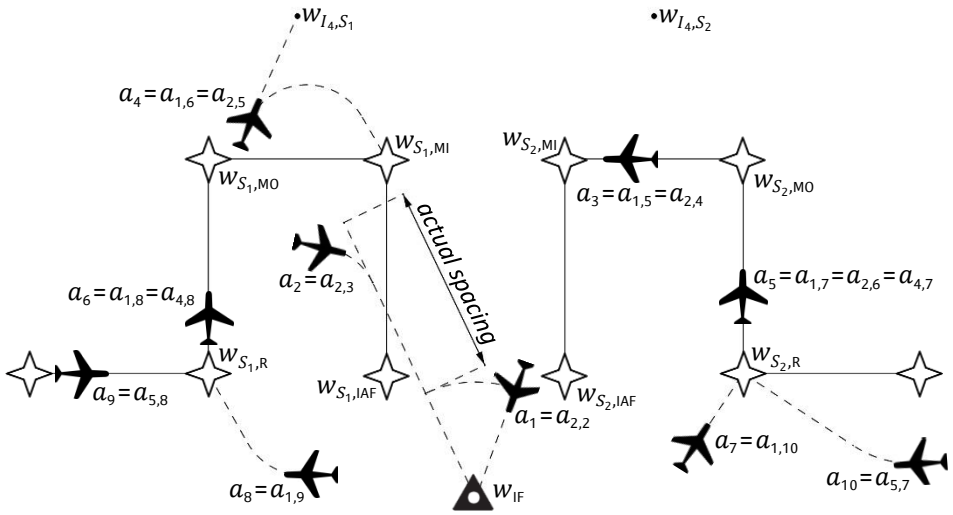


Figure 16 – Visualization of the merging and sequencing practices in the ARR sector

The second type of merging practice as applied by the ARR controller relates to tromboning. The tromboning merging technique is only applicable to aircraft that have already passed $w_{S,R}$. This condition is considered: to make sure that aircraft can intercept the localizer of runway 16L in a realistic manner, to prevent conflicting situations with the sequencing operations that take place at $w_{S,R}$, and at last because $w_{S,R}$ marks the begin point of the trombone segments within the XIBIL2A and RITEB2A STAR procedures. The sequence in which aircraft are to be merged is fixed and defined by the sequence in which they pass $w_{S,R}$, i.e. based on a "first come first serve" principle. The merging practice of the controller is based on the provision of well-timed *inbound vectors* towards the merging point w_{IF}. Aircraft that are near w_{IF} are generally instructed only one direct-to towards w_{IF}. This specific vector will be instructed once the *actual spacing* between a_i and $a_{2,i}$ is considered sufficient for a feasible merging operation (e.g. a_1 and a_2 in figure 16). The spacing that is required for a feasible merging operation is especially gained in the situation when $a_{2,i}$ is operating an *inbound vector* towards w_{IF} and when a_i is flying the *downwind segment* of the STAR procedure. The merging practice of the controller becomes however more challenging once aircraft are positioned on the *base-* or *final segment* (e.g. a_3). As a reaction to these saturated trombone segments the ARR controller will start to vector aircraft towards $w_{I_4,S}$. The direction of this vector allows the vectored aircraft (e.g. a_4) to further increase the required spacing with the surrounding aircraft (e.g. a_2 and a_3) to facilitate a feasible merging operation. The *outbound vector* towards $w_{I_4,S}$ is thereafter first followed by an *inbound vector* towards $w_{S,MI}$. This is required because aircraft are only considered available for an *inbound vector* towards w_{IF} once they are positioned on or inside the trombone segments of the STAR procedure. This condition is meant to limit the distance that is to be travelled while operating the direct-to towards w_{IF}, which in turn enhances the predictability of the arrival time of the merged aircraft at w_{IF}. The vector towards $w_{S,MI}$ anticipates already on the merging practice that is still to be performed, which results in the delivery of pre-merged aircraft pairings at $w_{S,MI}$.

5.3.3.8 Holding operations

Two holding patterns are considered in the model that are used by the controller agent as a delaying tactic due to *saturation* of the trombone segments in the used STAR procedures. Holding operations are initiated by the TNW/TNE controllers once they observe this *saturation* due to the presence of vectored aircraft north of the *base segments*. These vector operations will rapidly increase over time if the same amount of traffic is still being delivered to the ARR sector. The initiation of holding operations by the TNW/TNE controllers enables the ARR controller to eliminate the number of (vectored) aircraft in its sector. Aircraft will again be released from the holding stack once the trombone segments are observed to be no longer saturated. Aircraft that have been operating the holding operations the longest period of time are positioned at the bottom of the holding stack and will be released first.

Release rates

The model considers two different rates at which aircraft are released from the holding. The first default rate is simply determined by the *desired spacing* between a_i and $a_{1,i}$, which makes that aircraft are exiting the holding at a maximum possible rate. The second release rate however takes the presence of aircraft near the *base segment* into account. Aircraft that are near the *base segment* are likely to saturate the trombone segments of the STAR procedures (again). The TNW and TNE controller agents are therefore modelled to release aircraft from the holding with an adjusted (lower) rate to prevent a new possible *saturation* in the near future. This adjusted rate is modelled using a multiplier (factor) by which the *desired spacing* between a_i and $a_{1,i}$ is to be multiplied with. This increased *desired spacing* results in relatively less aircraft that are delivered to the ARR sector. The release rate will eventually increase when no aircraft are near the *base segments* anymore.

Altitude layers

Each holding stack can contain multiple aircraft that are flying at different altitude layers. The controller agent is modelled to maintain two aircraft at each altitude layer, where each layer is vertically separated by 1000ft. Each time when an aircraft is released from the holding the controller needs to instruct a number of altitude instructions to move down the multiple aircraft in the holding stack. Vertical separation in holding stacks is for this reason considered in the model because of its implications on the workload of the TNW/TNE controllers. Longitudinal separation criteria are however not considered in the modelled holding stack.

5.3.3.9 Speed and altitude instructions

Aircraft that are operating the XIBIL2A and RITEB2A STAR procedures do still require speed and altitude instructions. These instructions allow aircraft to descend and decelerate towards a proper altitude and airspeed to initiate the approach phase. The controller agent is modelled to provide speed and altitude instructions in a structured and iterative manner. This approach allows aircraft to decelerate and to descend in a stepwise manner along the lateral profile of the STAR procedure and such that conflicts between aircraft as the result of different speed and altitude profiles are not likely to occur. The specific contents of the speed and altitude instructions are defined by the known speed and altitude constraints for each waypoint in the XIBIL2A and RITEB2A STAR procedures. The controller agent is able to relate the position of each aircraft along the STAR procedure to these “desired” speed and altitude constraints. The controller will eventually issue a speed and/or altitude instruction if the observed airspeed and/or altitude of an aircraft are not in proportion with the desired speed and altitude.

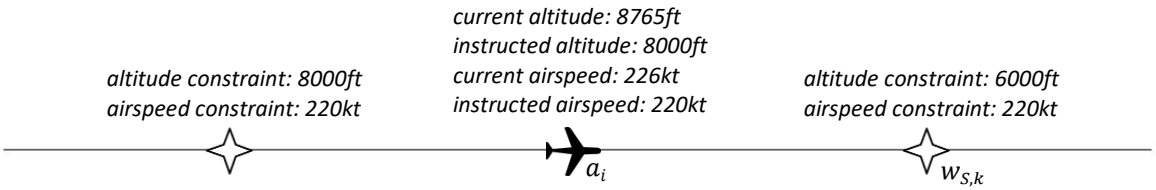


Figure 17 – Visualization of a scenario in which the controller agent would need to instruct an altitude instruction

A typical scenario in which a controller would need to instruct an altitude instruction is visualized in figure 17. This example scenario is meant to demonstrate the logic that has been modelled in defining the necessity of speed and/or altitude instructions and the contents of such instructions. The used scenario considers an aircraft (i.e. a_i) that is currently descending and decelerating towards its instructed altitude and airspeed. Both the instructed altitude and airspeed relate to the altitude and airspeed constraints of the previous fly-waypoint (i.e. 8000ft and 220kt). The controller agent is able to compare both the actual observed and instructed altitude and airspeed of a_i with the altitude and airspeed that a_i is desired to fly when considering the current position of a_i along the STAR procedure or within the arrival phase in general. In the scenario of figure 17 the controller agent is aware of the waypoint where a_i is heading to (i.e. $w_{S,k}$) and the altitude and airspeed constraints that correspond to that specific waypoint. The controller agent combines this information to decide if an airspeed and/or altitude instruction is required. The scenario as visualized in figure 17 does not require a speed instruction, since a_i is already decelerating towards the airspeed constraint of the next waypoint (i.e. $w_{S,k}$). An altitude instruction is on the other hand required, since a_i is currently descending towards an ‘old’ altitude constraint. The controller agent is modelled not to wait in providing speed and/or altitude instructions until an aircraft is again flying in a steady state condition. In the scenario of figure 17 this means that the controller agent will instruct a_i to proceed descending to the new desired altitude of 6000ft.

The appendices C.4.5.1.5 to C.4.5.1.8 provide the formal description of how the controller agent is modelled to define the necessity of speed and/or altitude instructions.

5.3.4 Modelled instructions

A number of instructions have been modelled that are used by the controller agents to guide aircraft in an efficient and safe manner from the entry point of the STAR towards the runway threshold in both normal and disturbed conditions. These instructions are closely related to and based on the number of tasks that each controller agent has been assigned (section 5.3.1) and the type of strategies and practices that are considered by the controller agent (section 5.3.3). Table 8 below provides the instructions that are considered in the model, the controller agents where each instruction is to be provided by, the index of each instruction (used in the model specification appendix), and brief descriptions of each instruction. These descriptions contain definitions and theory that has been covered in the previous section. The next section provides a visualization of typical trajectories that correspond to the different vector instructions. The modelled instructions are collected in the set $\mathbb{I}$.

Instruction	$c \in$	I_{index}	Description
STAR speed and/or altitude	$\{c_1, c_2, c_3\}$	I_1	Speed and/or altitude instruction to prepare a_i for the initial approach by guiding a_i in a gradual manner along the STAR profile
vector outbound STAR	$\{c_1, c_2, c_3\}$	$I_{2,1}$, $I_{2,2}$	Vector instruction towards $w_{I_2,S,k}$ due to insufficient spacing between a_i and $a_{1,i}$ ($I_{2,1}$: *vector outbound STAR* instruction due to a *separation minima conflict*, $I_{2,2}$: *vector outbound STAR* instruction due to a *desired spacing conflict*)
vector inbound STAR	$\{c_1, c_2, c_3\}$	I_3	Vector instruction towards $w_{S,k}$ due to sufficient spacing between a_i and $a_{1,i}$
vector outbound merge	$\{c_3\}$	I_4	Vector instruction towards $w_{I_4,S}$ due to saturated trombone segments and because of an infeasible merging practice due to the relative locations of a_i, $a_{1,i}$ and $a_{2,i}$
vector inbound merge	$\{c_3\}$	I_5	Vector instruction towards $w_{S,\text{MI}}$ when the merging practice towards w_{IF} is expected to be feasible in the near future due to sufficient spacing between a_i and $a_{1,i}$, and between a_i and $a_{2,i}$
vector outbound IF	$\{c_3\}$	$I_{6,1}$, $I_{6,2}$	Vector instruction towards $w_{I_6,S,N}$ due to insufficient spacing between a_i and $a_{2,i}$. ($I_{6,1}$: *vector outbound IF* instruction due to a *separation minima conflict*, $I_{6,2}$: *vector outbound IF* instruction due to a *desired spacing conflict*)
vector inbound IF	$\{c_3\}$	I_7	Vector instruction towards w_{IF} due to sufficient spacing between a_i and $a_{2,i}$ for a feasible merging operation at w_{IF}
vector outbound trombone	$\{c_3\}$	I_8	Vector instruction towards $w_{I_8,S}$ due to insufficient spacing between a_i and $a_{2,i}$ and/or a too large interception angle of the localizer, while a_i is nearing w_{IF}
vector inbound trombone	$\{c_3\}$	I_9	Vector instruction towards $w_{S,\text{R}}$ due to sufficient spacing between a_i and $a_{4,i}$, and between a_i and $a_{5,i}$, and meant to let a_i re-join and re-operate the trombone segment of its originally flown STAR procedure as a sequel to the unsuccessful merging operation(s) at w_{IF}
handover to ARR	$\{c_1, c_2\}$	I_{10}	Handover of a_i from the TNW/TNE controller to the ARR controller at or near $w_{S,\text{HO}}$
handover to TWR	$\{c_3\}$	I_{11}	Handover of a_i from the ARR controller to the TWR controller when a_i is positioned on final approach
handover to GND	$\{c_4\}$	I_{12}	Handover of a_i from the TWR controller to the GND controller after a_i has landed
landing clearance	$\{c_4\}$	I_{13}	Landing clearance instruction when a_i is on final approach and $a_{3,i}$ has already landed
go-around	$\{c_4\}$	I_{14}	Go-around instruction due to insufficient spacing between a_i and $a_{3,i}$

			while a_i is on final approach
holding entry	$\{c_1, c_2\}$	I_{15}	Instruction for a_i to enter the holding at $w_{S,H,\text{I}}$ due to saturated trombone segments and high traffic densities in the ARR sector
holding exit	$\{c_1, c_2\}$	I_{16}	Instruction for a_i to exit the holding when the traffic situation in the ARR sector is observed to be balanced again
holding altitude	$\{c_1, c_2\}$	I_{17}	Instruction for a_i to change altitude in the holding stack

Table 8 – List of modelled instructions

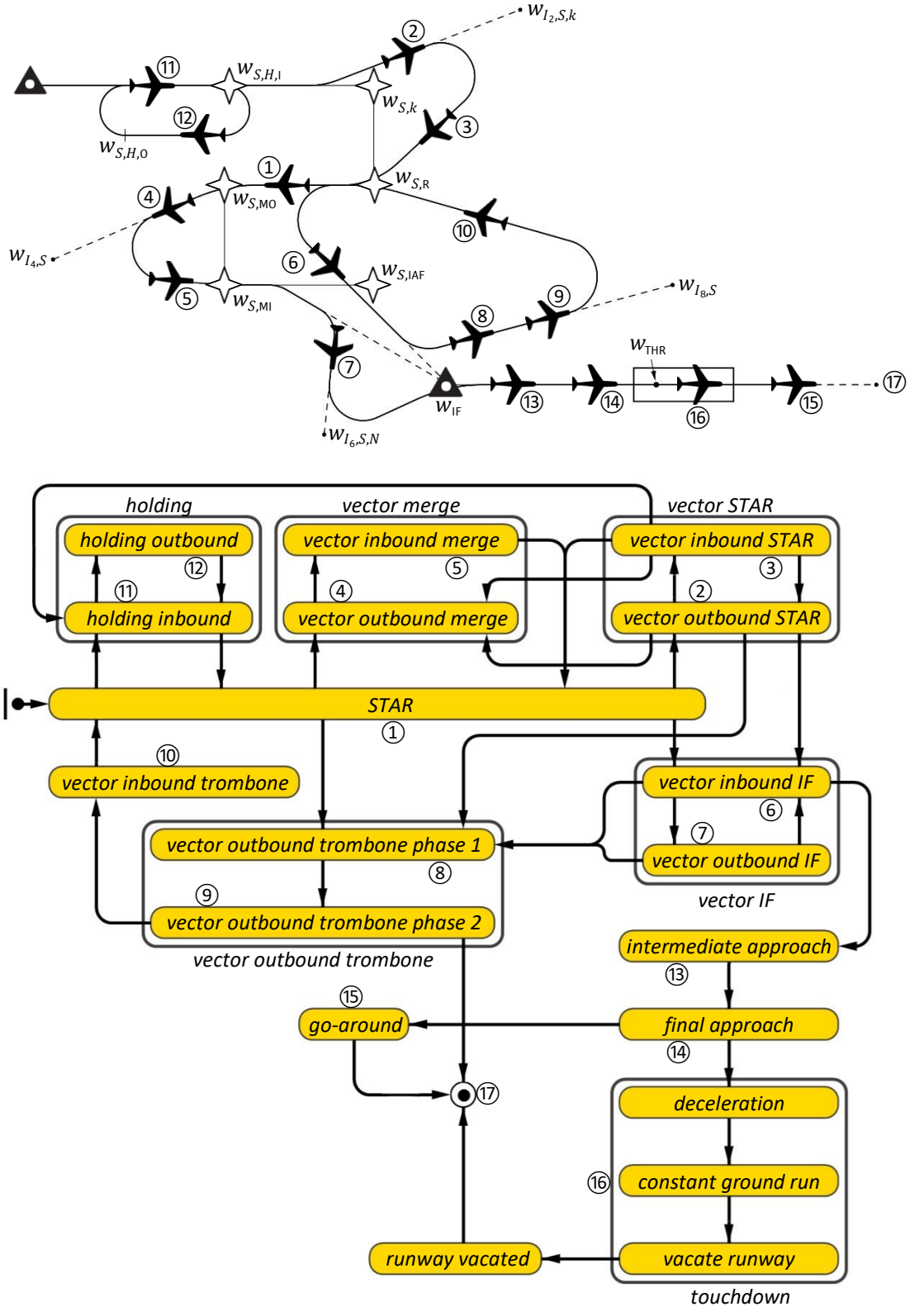


Figure 18 – Visualization of the modelled operational states of aircraft agents, which can be observed by the controller agent

5.3.5 Operational states

The traffic situation that is observed by the controller agent is contextualized by the operational state of each aircraft, i.e. the specific flight phase or flight operation that the aircraft is operating. Each aircraft operates in a series of operational states in the period between generation and landing. The exact series of states is dependent on many factors that emerge during simulation, such as traffic density, the spacing between aircraft, wind conditions, accuracy of the predictive capacities of controllers, etc. The various operational states that are considered in the model and the possible transitions between those states are collected in a statechart. This statechart is visualized in figure 18 below, together with a visualization of typical trajectories that correspond to the modelled operational states. This statechart is used in appendix C.4.5.1.1 to model the situation awareness of the controller agent about the operational state of a_i. This section is meant to clarify the type of operations that have been modelled, the coherence between these operational states, and the resulting trajectories that can be expected and which correspond to each type of operation. The specific operational state of each aircraft (i.e. type of operation, heading, altitude, speed) is herein determined and managed by the four executive controllers. Some of the performance indicators that have been measured during the parameter variation experiments relate to a number of operational states that are depicted in figure 18. Note the similarities between the names of the operational states and the names of the modelled instructions.

The use of STAR procedures as the baseline arrival operations is clearly shown by the number or transitions that enter and leave the STAR state (figure 18). Many of the modelled operational states (such as the vectoring practice and the holding operations) originate from the lateral profile of the STAR procedure (*outbound*). Other operational states, however, do connect again to the profile of the STAR after interception of one of its fly-by waypoints (*inbound*). As can be seen by the number of transitions in figure 18 there are a lot of possible state switches modelled between different type of operational modes. This large number of transitions is required to facilitate the various type of operations that may emerge during simulation.

The *vector outbound trombone* operational state consists of two phases. The first phase represents the time period that is used to allow a_i to leave the dense ARR sector. The second phase represents the time period that the controller may use to decide upon the feasibility of a *vector inbound trombone* instruction. a_i is removed from the simulation environment if the *vector inbound trombone* instruction is not found to be feasible within this period of time. The operational states that describe the intermediate approach, final approach and touchdown follow a known and standard pattern.

5.4 Feeder controller

The feeder (controller) agent (NE controller) is in the model responsible for the generation of aircraft agents at the entry points of the XIBIL2A and RITEB2A STAR procedures. The modelled generation practice is such that it resembles the sequencing practices at the entry points as they would have occurred when the en-route NE sector was taken into account. The modelled generation practice can be described by the following characteristics:

- The generation rate approaches the desired throughput capacity at the entry points of both STAR procedures;
- The generation rate results at all time in realistic and feasible aircraft generations, i.e. there will be no *desired spacing conflicts* just after generation. The feeder agent will only generate an aircraft at the entry point once the separation with the preceding aircraft is conform the *desired spacing* (as maintained by the feeder controller);
- The generation rate is adjusted accordingly and instantaneously when the feeder agent is informed by the Supervisor APP about a changed throughput capacity in the approach sectors;
- The generation rate takes into account the inaccuracies of the feeder agent in delivering aircraft at the entry point. This spread in inter-arrival times is modelled with a truncated normal probability distribution. The mean of this distribution is in line with the current maintained generation rate. The distribution is truncated to make sure that aircraft are not delivered at the entry points with a deviation of more than 40 seconds before or after the desired generation time point (i.e. mean).

See appendix C.5.5 for a formal specification of the two events that model the generation practice of the feeder agent. These events make sure that aircraft are generated conform the desired throughput capacity and such that initial conflicts at the entry points are avoided. As will be seen in chapter 8 the generation rate (i.e. throughput capacity) is varied during the conducted parameter variation experiments. The experiments consider different settings

for the generation rate during *normal capacity mode* and only one fixed setting during *reduced capacity mode* (i.e. 21 ac·h^{-1}).

In addition to these modelled generation characteristics the feeder agent is also responsible for defining the initial flight conditions and properties of the (to be) generated aircraft agents. These initial conditions and properties relate to the following characteristics:

- 65% of the aircraft agents will be generated at entry point RITEB, which is based on the distribution that has been obtained after analysis of the historic flight data. The distribution of the aircraft generations among the entry points is managed by a uniform probability distribution;
- 80% of the generated aircraft are of type MEDIUM, the remaining 20% are of type HEAVY. The exact WTC is determined by a uniform probability distribution;
- Each aircraft is generated exactly on top of the entry point with a fixed altitude, indicated airspeed and heading. The analysed historic flight data supports the use of these fixed initial conditions;
- Each aircraft will upon generation immediately start to operate the respective STAR procedure;
- Each aircraft is upon generation immediately handed over to the TNW or TNE controller.

5.5 Aircraft

Each aircraft a_i agent (with $i \in \mathbb{N}_0$ and $a_i \in \mathbb{A}$) is modelled as a composite agent that consists of the two deeply related and interconnected subagents: Flight crew a_i and MCP a_i. The flight operations of the aircraft a_i agent are fully defined by the behaviour and actions of the flight crew a_i agent and the functionalities of the MCP a_i agent. The flight crew a_i (human) agent is responsible for the operation of aircraft a_i and the execution of one of the modelled instructions ($I \in \mathbb{I}$) as instructed by one of the executive controller agents. The MCP a_i (system) agent is used in the model to control the autopilot of aircraft a_i, which is able to automatically control (e.g.) the speed, altitude and heading of the aircraft. (The next two sections will often omit the index "a_i" for readability reasons)

The model considers two different types of WTCs: HEAVY (W_H) and MEDIUM aircraft (W_M). Both WTCs are collected in the set $\mathbb{W}$. By considering these two WTCs the controller agent has to apply different *separation minima*, which in turn will affect the to be established throughput capacity. Because of these different WTCs the model has also to include a minimum of two different aircraft types. The B738 (D_1) and B744 (D_2) aircraft types are therefore used, which allow to incorporate different flight dynamics and performances. Both aircraft types are collected in the set $\mathbb{D}$.

5.6 MCP

The Mode Control Panel (MCP) a_i agent is responsible for the operation of the autopilot of aircraft a_i. The MCP is an instrument panel with switches, knobs and pushbuttons that allow the flight crew agent to select which parts of the aircraft's flight are to be controlled automatically. The MCP agent is therefore defined as the collection of systems that assist the flight crew agent in (automatically) controlling the trajectory of the aircraft. The MCP agent is used and operated by the flight crew agent when the aircraft is instructed to change speed, altitude and/or heading. The MCP agent can however also be set automatically by the FMS when the aircraft is flying the lateral profile of the STAR. The aircraft agent is during the complete arrival and approach segments flown by the autopilot functionalities as controlled by the MCP agent. This means that the flight performance of aircraft and the resulting flight trajectories are fully defined by the MCP agent. The three upcoming sections will provide more detail about the modelled flight performance, the application of the various significant points, and the modelled functionalities of the MCP agent.

5.6.1 Flight performance

The MCP agent controls the autopilot of the aircraft agent and therefore also its resulting flight performance. The modelled flight performance of each aircraft can be described by the following characteristics:

- Aircraft are modelled as point masses, i.e. the reduction of aircraft weight due to fuel burn is not taken into account, as well as (e.g.) drag and thrust forces;
- Aircraft movement is modelled with the following differential equations: $\dot{x} = v \cdot \cos(\psi)\cos(\gamma)$, $\dot{y} = v \cdot \sin(\psi)\cos(\gamma)$, $\dot{z} = v \cdot \sin(\gamma)$, $v = \omega \cdot r$, $\dot{\psi} = \omega$ and $\dot{v} = a$;

- The acceleration of aircraft is modelled using a fixed set of constant acceleration values. Each of these acceleration values relates to a specific aircraft type, operational state and descending/accelerating state of the aircraft. Deceleration is only considered during descent and level flight, while acceleration is only considered during climb (i.e. go-around);
- Each aircraft reaches a stabilized airspeed before approaching the 1000ft above the elevation of the runway;
- The descent performance in terms of vertical speed profiles for different flight levels is modelled using the BADA performance files. The vertical speed for a given altitude layer is obtained using linear interpolation between the discrete set of given data points. Different descent performances are considered for the B738 and B744 aircraft types (appendix C.6.3.1);
- The vertical speed during climb (i.e. go-around) has a constant value for both aircraft types;
- All turning movements have a constant turn radius, and can therefore be described by a circular shape. Circular shaped turning movements have been modelled to take into account the important effects of time and distance during such maneuvers, which is especially true for vector operations that require relatively large heading changes;
- The angular velocity of the aircraft is defined by its ground speed (due to the fixed turn radius);
- The rate of change of displacement of aircraft agents through the simulation environment is defined by their groundspeed (i.e. the true airspeed of the aircraft plus the wind speed component at the given altitude layer);
- The wind component that determines the groundspeed of the aircraft is defined as the projection of the wind vector onto the air vector representing the direction of the aircraft through the airmass (appendix C.2.2.8);
- The heading angle equals the course angle, i.e. drift angles as the result of cross wind are not considered;
- Each aircraft type has its own range of final approach speeds and threshold speeds. The specific final approach speed and threshold speed is for each aircraft agent defined by a uniform probability distribution;
- It is assumed that aircraft have sufficient fuel to operate the various type of controller instructions.

See appendix C.6.5 for a formal description of how each specific flight performance variable of the aircraft agent is updated, and therefore how aircraft movement is modelled.

5.6.2 Navigation system

The navigation system as part of the FMS is considered an important subsystem of the MCP agent. The navigation database of this system contains the geographical locations of the waypoints within the IAPs (i.e. $w_{S,k}$ and w_{IF}), the locations of the holding patterns (i.e. $w_{S,H,\text{I}}$ and $w_{S,H,\text{O}}$) and the location of the runway threshold (i.e. w_{THR}). The MCP agent uses these geographical locations to determine the position and orientation of the aircraft agent in the approach sector. This position data is secondly also used to fly the lateral profile of the STARs, the holding patterns and the final approach towards the runway threshold. The information that is contained in the navigation database enables the functionalities that are described in the next section.

5.6.3 Functionalities

The MCP agent is responsible for the operation of the autopilot system of the aircraft agent. This autopilot system is able to automatically control the aircraft agent during the complete arrival and approach segments. The following functionalities of the autopilot system are considered in the model:

Speed control

The ability to hold a specific (indicated) airspeed, or to change airspeed with a specific acceleration/deceleration until the aircraft reaches the set airspeed. All speeds are set manually by the flight crew agent after having been instructed by the controller agent. (appendix C.6.4.1)

Altitude control

The ability to hold a specific altitude, or to change altitude with a specific vertical speed until the aircraft reaches the set altitude. All altitudes are set manually by the flight crew agent after having been instructed by the controller agent. (appendix C.6.4.2)

Heading control
The ability to hold a specific heading, or to turn to a new heading or desired point in space with a certain rate of turn. When a new heading direction is required and/or desired, the MCP agent is dependent on either manual input from the flight crew agent or automated input from the MCP agent. The new heading is set manually when the flight crew agent is instructed by the controller agent to do so. The heading of the aircraft is set automatically by the aircraft's FMS when the aircraft is flying the fixed profile of the STAR procedure. The modelled heading control functionality will always steer an aircraft towards the direction (left/right) that will result in the fastest completion of the required turning movement. (appendix C.6.4.3)

Interception of fly-by waypoints
The ability to (re-)operate a published procedure or track, such as the STAR procedure or the intermediate approach by intercepting the respective fly-by waypoint (i.e. $w_{S,k}$ and w_{IF}). This functionality is considered because of the usage of fly-by waypoints in the arrival and approach operations at Rome Fiumicino (instead of fly-over waypoints). A fly-by waypoint requires an anticipated turning movement before actually passing the waypoint. Such turn anticipation is required in order to tangentially intercept the next segment of the route or procedure. The point at which an aircraft should initiate the turning movement to intercept a specific fly-by waypoint is calculated by the MCP agent. This interception functionality allows the aircraft to automatically fly the lateral profile of a STAR procedure via the multiple interconnected fly-by waypoints. (appendix C.6.4.4)

Interception of the glideslope
The ability to intercept the glideslope and to follow the glide path of the glideslope along the localizer course until touchdown at the runway threshold. (appendix C.6.4.5)

Holding procedure
The ability to automatically enter, operate, and exit the holding pattern. The MCP agent is able to automatically fly and maintain a holding procedure over the inbound leg towards $w_{S,H,\text{I}}$ and the outbound leg towards $w_{S,H,\text{O}}$. (appendix C.6.4.4)

5.7 Flight crew

The flight crew a_i agent can be described as the two pilots that are operating aircraft agent a_i. During the arrival and approach phases the flight crew agent will be periodically in contact with the controller agent. Within such contact the flight crew agent is instructed one of the modelled instructions. Most of these instructions are related to adjustments in the speed, altitude and/or heading of the aircraft. All these type of instructions that changes the aircraft's state in terms of position and orientation are performed and controlled automatically by the MCP agent. Upon reception of such instruction the flight crew agent is thus only responsible for setting the correct input on the MCP. The flight crew agent is therefore modelled rather simplistic because of the considered autopilot functionalities. All received instructions are immediately processed by the flight crew agent while being in contact with the controller agent. As described in section 5.1.2, the duration of this contact is modelled with a fixed lognormal distribution. See appendix C.7.3.2 for a formal description about the communication between the flight crew and the controller agents and the way how incoming instructions are processed.

5.8 Meteo Office

The Meteo Office agent is in the model responsible for monitoring the current weather conditions at the airport. When changing weather conditions have been detected the Meteo Office will immediately inform the Supervisor TWR. The Meteo Office agent does therefore only serve as a trigger after which a number of related agents will need to adjust their maintained runway/throughput capacity.

5.9 Supervisor TWR

The main task of the Supervisor TWR agent is to define the runway capacity. The runway capacity is determined by the current weather conditions as reported by the Meteo Office. The Supervisor TWR will reduce the maintained runway capacity when it is informed about deteriorated weather conditions. The runway capacity will be recovered again when the weather conditions are improving. The Supervisor TWR will inform both the TWR controller and the Supervisor APP when it has decided to change the runway capacity. The TWR controller is always informed first about a changed runway capacity.

The model considers for both weather conditions one specific and desired runway capacity. The notion 'desired' is used since the to be achieved runway capacity is limited and dependent on many factors, such as wind conditions, the accuracy of the merging practices, runway occupancy times, and the type and category of aircraft that are on approach. Because of these dependencies it is difficult to steer upon (i.e. model) an exact runway capacity that should be achieved and maintained for a given period of time. In other words, the achieved runway capacity will be the result of the factors that are described above and is therefore difficult to predict exactly. One of the factors that can be changed to influence the achieved runway capacity is the *desired spacing* that is maintained between aircraft. The size of this separation distance is determined by the variable time-based *separation buffer (type I)*. The Supervisor TWR is responsible for defining and communicating these time-based *separation buffers (type I)*.

5.10 Supervisor APP

The Supervisor APP agent is responsible for defining the capacity of the multiple approach sectors and for the coordination of the traffic flow within these sectors. More specifically it makes sure that the throughput capacity in and between these sectors is in proportion with the current runway capacity. The Supervisor APP therefore monitors the traffic situation in the approach sector to decide if the achieved throughput capacity is still appropriate. In the situation of a sudden drop (or recovery) in runway capacity it is obvious that the throughput capacity in the approach sectors has to be adjusted accordingly. The Supervisor APP will be informed by the Supervisor TWR when the runway capacity has been adjusted. The Supervisor APP will in reaction to this message change the capacity of the approach sectors. The changed throughput capacity is thereafter communicated to the feeder agent and the approach controller agents. These four agents will upon reception of these capacity updates adjust the maintained throughput capacity in their sectors.

Traffic flow

The traffic flow in the approach sectors is coordinated by the Supervisor APP. The Supervisor APP oversees that the maintained *desired spacings* in each approach sector are such that they allow for controlled and expeditious arrival and approach operations. The Supervisor APP is therefore (first of all) modelled to take into account the effects of compression between aircraft during approach. In order to anticipate on the effects of compression the controller agents gradually apply larger separation distances (*desired spacings*) for aircraft that are positioned further away from the airport. These increased separation distances are modelled by applying larger *time buffers* (i.e. *separation buffers (type I)*). The *time buffers* that are applied by the feeder agent are for instance larger than the ones that are applied by the TNW/TNE controllers, and so on. The specific values of the *time buffers* have been assigned empirically by evaluating the effects of the assigned *time buffers* on the achieved runway/throughput capacity. A typical traffic situation that may arise as the result of compression and the use of different *time buffer* sizes is visualized in figure 19. This figure shows that the separation distances between aircraft are gradually becoming smaller while aircraft are approaching the runway threshold. This is first of all due to the natural effects of compression during arrival and approach and secondly due to the use of different *time buffer* values by the multiple approach controllers and the feeder controller.

Figure 19 – Visualization of typical spacings during arrival and approach as the result of compression and the application of different time buffer sizes

The Supervisor APP is thus responsible for defining the (qualitative) *desired spacings* that should be maintained between aircraft in the approach sectors. These separation distances are used by the Supervisor APP to manage the capacity in the approach sectors. When the Supervisor APP is informed about the changed runway capacity it will as a reaction adjust the capacity of the approach sectors. This desired capacity change is modelled by applying different values for the *time buffers* that define the size of the *separation buffer (type I)* between the multiple aircraft pairings. The shape of the traffic flow and the size of the traffic density and throughput capacity in the approach sectors are therefore defined by the multiple time-based separation buffers that are applied in the model. Note that the set of *time buffers*, as defined by the Supervisor APP, is formally specified in appendix C.4.1 instead of appendix C.10.1. This is done deliberately to combine all information that is related to the modelled separation practice of the controller agent in one appendix, which enhances readability and comprehensibility.

Dynamics related to capacity updates

The Supervisor APP will inform the approach controllers and the feeder agent when it has decided to change the capacity of the approach sector. Each of the these agents will be instructed to increase or to reduce the separation (i.e. *desired spacing*) between its aircraft, depending on how the throughput capacity has been adjusted. The four agents will be informed separately and therefore at a different point in time about a capacity update. Because of these dynamics there will be a time period in which the feeder agent and the controller agents have a different awareness about the *capacity mode*. The sequence in which the controllers are informed about capacity updates will eventually thus define the way in which the traffic flow evolves within the approach sectors. For this reason the model contains logic that determines when each agent will be informed about a capacity update. The ARR controller agent is considered by default the agent that will be informed first about a reduced or recovered capacity update. This is because traffic that is located in the ARR sector is closest to the runway threshold when compared to traffic in the TNW/TNE sectors. The TNW/TNE controller agents will be informed next. The controller (i.e. TNW/TNE) with the most aircraft in its sector will be informed first. The feeder agent will eventually be informed last. In this way the throughput capacity is gradually changed among the sectors and in a direction away from the runway threshold.

6 Model verification

Model verification has been applied as an iterative process that is performed in parallel with the implementation of the quantitative agent-based model. The verification involves both single-agent testing and multi-agent testing to verify the behaviour of agents and the interactions among agents. The graphical features within *AnyLogic* have proven to be very useful during these verification processes. These visualizations allow for instance to verify aircraft movement over time, to observe how variables evolve over time, to observe the state of each agent at a specific time point, etc. In this way one can quickly verify if the observed emergent behaviour is in line with the behaviour that is expected.

Figure 20 shows the main window of the model while running the simulations in *AnyLogic*. This window can be considered as a kind of dashboard that combines and visualizes the emergent behaviour as the result of the low-level agent specifications. By observing this main window the following behaviours and interactions can be verified:

- the total flown trajectories of the multiple aircraft agents in the simulation environment;
- the travelled distance of an aircraft agent over time;
- the flown trajectory of an aircraft agent in relation to the instruction that is received from the active controller;
- the type of operations that each aircraft is flying in relation to the traffic situation in each airspace sector;
- the state of each aircraft during arrival and approach in terms of speed, altitude and heading;
- the sequence in which the Supervisor TWR/APP, the controllers and the feeder controller are informed;
- the communication between agents by keeping track of the time points at which messages are sent;
- the type of instructions that are provided;
- the specific executive controller agent by which an instruction has been provided;
- the specific content of instructions;
- the observation by the Meteo Office of the normalized/deteriorated weather conditions;
- the distribution in which, and the rate at which aircraft agents are generated at the entry points.

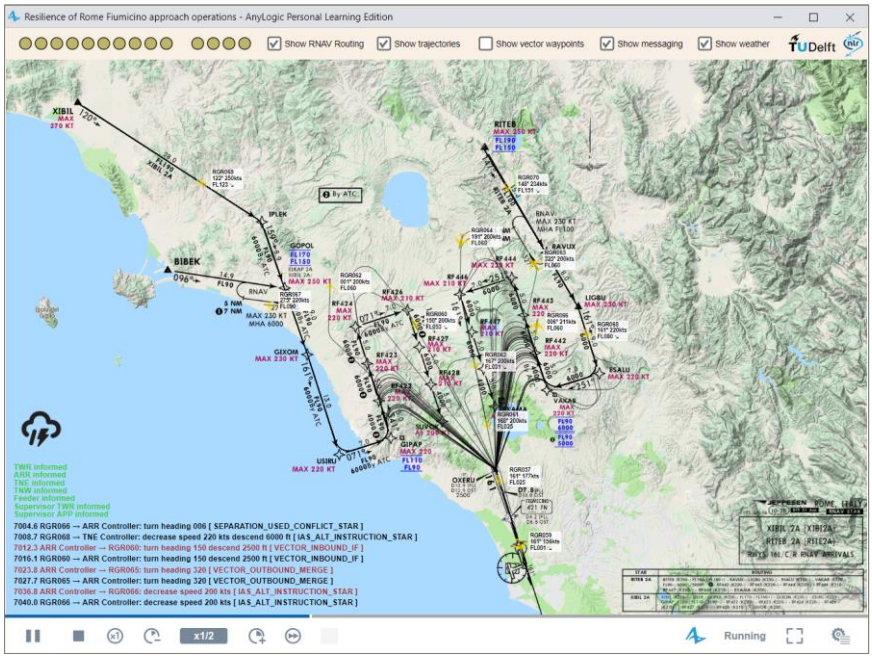


Figure 20 – Model (main) view in AnyLogic during the simulation experiment

In addition to this main window each individual agent has also its own specific window associated with it. These windows display for each individual agent the set of modelling elements that have been used to specify their modelled properties and interactions. In this way one can track the state of each agent over time (i.e. thus also each individual aircraft and controller agent). The graphical features of these used modelling elements have proven to be very suitable for verification of the dynamics in the model. In order to verify these observed dynamics the following features of the used modelling elements have been applied:

- Variables: *AnyLogic* allows to monitor the values of each assigned (class) variable at model runtime, and the history of values using plots and charts. In this way one can verify if the monitored values of each variable and the way in which each variable is changing make sense in relation to for instance the behaviour and operation that is considered. This type of verification is therefore used to verify the ground speed of the aircraft agents as the result of the incorporated wind model, the multiple bearings that are considered, the *aircraft referencing*, etc.;
- Sets: *AnyLogic* allows to monitor the contents of each set at model runtime. In this way one can verify for instance the type and number of tasks that are identified by a specific controller agent, the correct sequence of aircraft in the various related sets, etc.;
- Statecharts: *AnyLogic* highlights the active states of each statechart during model runtime. In this way one can easily verify for each agent if the set of observed active states makes sense given the current situation, operation etc. In addition to the highlighted states *AnyLogic* also highlights (some of) the transitions when they are either taken or when the transition is scheduled to be taken in the near future. Both type of highlights are mainly used to verify if statecharts are not infinitely looping in order to prevent unrealistic behaviour from occurring.
- Functions: the set of statements in functions are in itself static and are only performed when for instance the actions in an event or state require to do so. In order to verify the return values of functions and/or its corresponding actions a set of interactive controls (buttons/edit box) have been used. These controls are provided by *AnyLogic* and allow to easily verify the functionalities of the various functions that have been modelled.
- Events: all type of events are highlighted once they occur during model runtime. The timeout triggered events do also display the remaining time period before the event is scheduled to occur. The highlighted event occurrences and corresponding countdowns are used to verify if the rate at which events occur is according to the expectations.

Another method that has often be applied to verify the functioning of the modelled statecharts, functions and events is the implementation of a "*pause*" command in the respective action fields. This command pauses the running model at a desired action or given condition, and can thus be used to verify if the corresponding situation is according to what is expected.

Other general applied verification steps are related to:

- Verification of agent and model functioning using a range of (extreme) parameter values. This verification step makes sure that the model is also functioning properly when for instance: a relatively large generation rate is applied by the feeder agent (i.e. large traffic density in approach sector), when the controller agent is acting relatively slow (e.g. relatively late conflict detection), when the controller agent is applying a relatively bad vectoring accuracy (e.g. snowball effects), when a relatively large capacity drop is considered (e.g. the need for relatively many adaptive strategies), etc.;
- Execution of many simulation replications (≈100) with random variables to verify the agent and model functioning in different scenarios.

In parallel with the model development an extensive number of verification steps have eventually been performed. Verification has been performed by observing the implemented modelling elements during model runtime and secondly by statistical analysis of the output data. Based on the obtained results there can be concluded that the implemented agent-based model in *AnyLogic* is functioning properly and as expected.

7 Model validation

The developed and implemented quantitative agent-based model is used to explore and to quantify the resilient capacities of approach operations during disturbance. As already described, these resilient capacities are primarily defined by the skills and the resulting actions of controllers. The model considers therefore a number of instructions that may be provided by each of the controllers to guide aircraft in a fluent and safe manner towards the runway. The specific contents of these instructions and the resulting flight operations should be such that they resemble to an acceptable extent the approach operations that are flown in real life. This chapter will therefore describe briefly what steps have been performed to determine the validity of the developed agent-based model and the obtained simulation results.

Historic flight data

Historic flight tracking data has been used to specify the number of controller instructions and the corresponding contents of such instructions. Such data is of great value to model realistic and representative approach operations. Model validation is therefore for a considerable part related to the extent to which the simulated flight trajectories during the experiments resemble the historic flight trajectories as shown in figure 4. When analysing the obtained simulation results the following key validations can be observed in both normal and disturbed conditions:

- STAR procedures serve as the baseline operations during arrival;
- similar merging trajectories towards the intermediate fix;
- similar traffic density and achieved separation on final and intermediate approach;
- similar shape and operations of holding patterns;
- similar speed profiles during arrival and approach;

ATC broadcasts

Live ATC broadcasts and historic ATC recordings are considered useful to validate the workload of controllers by keeping track of the number of conversations in combination with the observed traffic situation using an online flight tracking service. There are however no ATC broadcasts available for Rome Fiumicino airport unfortunately. ATC broadcasts of other major airports with comparable approach operations have therefore been used.

Expert validation

The parameter values that are considered in the model are validated using the expertise at NLR. Employees that are experienced in the field of ATM were asked to validate the specified parameter values, especially those related to controller performance and flight dynamics during arrival and approach. In addition they were also asked to validate the flight trajectories that emerged during simulation. This expert validation allowed to fine-tune the implemented model by modifying the performance of the modelled approach operations.

The results of these validation steps in combination with the performed verification steps indicate that the developed model has sufficient accuracy to provide a realistic and profound analysis of the resilient capacities of conventional approach operations.

8 Experiments

The model that has been formalised, implemented and verified is used to conduct a number of parameter variation experiments with. These experiments are set up to explore and to quantify the resilient capacities of approach operations during a sudden bad weather disturbance. This chapter describes the experiments that have been conducted and the simulation results that have been obtained. Section 8.1 will first describe all important background information that is needed to understand and to interpret the obtained simulation results. Section 8.2 will thereafter provide a general analysis of the obtained simulation results by describing the general (resilience) characteristics and dynamic patterns that are observed. Section 8.3 at last will describe three specific parameter variation experiments that have been conducted.

8.1 Experiment set-up and considerations

This section will provide all background information that is needed to understand and to interpret the obtained simulation results, which are covered in the upcoming sections. The different topics below describe in a stepwise manner what kind of data is measured, how this data is measured, and how the dynamic patterns in the obtained data are visualized.

Type of experiments to conduct

The research objective aims towards exploration, i.e. to examine in what way certain factors affect the resilient capacities of disturbed approach operations. The experiments that have been conducted therefore aim to examine the significance of a number of factors in relation to the resilient capacities of disturbed approach operations. For this reason parameter variation is the type of experiment that has been applied to gain insight in the resilient capacities of approach operations during a sudden and unexpected bad weather disturbance. Parameter variation offers the opportunity to run the model multiple times with different parameter settings to analyse how these parameter settings affect the model behaviour and with that the resilient capacities of the system under study.

A number of processes and conditions are in the model specified using probability density functions, such as the wind model, the various time periods, the aircraft generation process, etc. The random variables that describe these processes and conditions will be reset at the initiation of each replication (simulation) during the parameter variation experiments. In this way one can describe the type of experiments that have been conducted as a combination of the parameter variation and Monte Carlo experiments.

What parameters to vary

See section 8.3.

What to measure

As described in the beginning of this report resilience can in general be understood as the ability of a (socio-technical) system to sustain operations within its functional limits under disturbing conditions. The extent to which a socio-technical system is able to cope with disturbing events can be assessed with resilience assessment metrics. The majority of these metrics are based on measuring and comparing the relative impact of a disturbance on system performance and the time it takes to recover. This *resilience triangle* paradigm is also applied to assess the resilient capacities of the disturbed approach operations. The following performance indicators have been used to assess and to express the resilient capacities of the simulated approach operations as the result of controller instructions (per time interval):

Performance indicators related to: Capacity

- *Time between landing*: the time period between two successive landings, used to express indirectly the separation between aircraft at touchdown (logged at landing);
- *Number of aircraft in approach*: the number of aircraft that are on arrival/approach for runway 16L, used to express traffic density in the approach sectors (logged every time interval);
- *Number of landings*: the number of landings at runway 16L, used to express the dynamically changing runway capacity (logged every time interval);

- *Runway throughput*: number of landings per hour, used to express the runway capacity in terms of "before and after disturbance" and "during disturbance" (logged before and after disturbance/during disturbance);

Performance indicators related to: Operational states

- *Percentage of aircraft flying "a specific operational state"*: the relative distribution of the active operational states for each aircraft agent, used to express the emergent and dynamic traffic situation in the approach sector in terms of type of operations that are flown and adaptive strategies that are applied. Note that these percentages only relate to the operations that are flown prior to interception of the IF, i.e. the percentages do not take into account aircraft that are operating the intermediate- or final approach segment (logged every time interval):
  - *Percentage of aircraft flying STAR*
  - *Percentage of aircraft flying vector STAR*
  - *Percentage of aircraft flying vector merge*
  - *Percentage of aircraft flying vector IF*
  - *Percentage of aircraft flying holding*
- *Time in approach*: the arrival/approach duration of an aircraft agent, i.e. the time period between generation and landing, used to express indirectly throughput capacity and traffic density in the approach sector (logged at landing);
- *Time flying "a specific operational state"*: the total time period that an aircraft has operated a specific operational state during its arrival/approach towards runway 16L, used to express to which extent a certain type of operation is flown and an adaptive strategy is applied (logged at landing):
  - *Time flying STAR*
  - *Time flying vector STAR*
  - *Time flying vector merge*
  - *Time flying vector IF*
  - *Time flying holding*
- *Number of "a specific instruction"*: the number of *go-around* and *vector trombone* instructions that are provided by the controller agents, used to express the number of times in which specific type of conflicting and challenging situations occur (logged before and after disturbance/during disturbance):
  - *Number of go-arounds*
  - *Number of vector inbound trombone*: relates to aircraft that have successfully re-operated the STAR procedure
  - *Number of vector outbound trombone*: relates to aircraft that are removed from the simulation environment

Performance indicators related to: Controller workload

- *Percentage in tactical mode*: the relative time period in which the controller is acting in the tactical control mode, as compared to the opportunistic control mode, used to express the workload of the controller (logged every time interval);
- *Number of instructions*: the number of contacts made in which the flight crew is instructed one of the modelled instructions, used to express the workload of the controller (logged every time interval);

Trajectory plots

This 'measure' at last is used to visualize the many flight trajectories that emerge during the number of simulations (replications). These plots present the flown trajectories for a given scenario either before disturbance and during disturbance, and therefore allow to quickly compare the operations that have been flown during both conditions. The coordinates of each aircraft agent are logged every 60 simulated seconds. The density of the logged coordinates (dots) in the trajectory plots are therefore an indication for the extent to which a certain operation is flown and an adaptive strategy is applied in a specific scenario before and during disturbance.

The number of identified performance indicators are used to express how the approach operations emerge as the result of controller actions. These obtained characteristics can then be used to express the resilient capacities of the approach operations. However, one has to note that within this quantitative study resilience is considered to be a relative property. This means that the performance of the system under study does not necessarily have to stay within some prescribed boundaries. These boundaries are not required, since safe operations can be maintained 'at all time'

as the result of controller actions. Quantification of resilience is in this study therefore more related to exploring how certain parameter settings contribute (relatively seen) to the resilient capacities of approach operations.

How to measure

The modelled approach operations, and more specifically the way in which the involved agents behave can be described by many dynamic, stochastic and interacting processes. The overall emergent behaviour of the modelled approach operations is shaped by these low-level interactions and processes. Since many of these processes are modelled by stochastics each simulation run will result in unique emergent behaviour. Each simulation run will therefore also result in different values for the performance indicators that have been defined. The specific values of these performance indicators are however time dependent due to the dynamics that are caused by the bad weather disturbance. In order to properly assess the resilient capacities of approach operations one should incorporate the effects of time. These time periods should be of sufficient length to capture the emergent behaviour, to allow for stabilized approach operations and to measure the resilient capacities. The time periods that are considered in the parameter variation experiments are visualized in figure 21 below. The measurement of the performance indicators will start when a stabilized and condensed flow of approaching traffic has been established in the simulation environment.

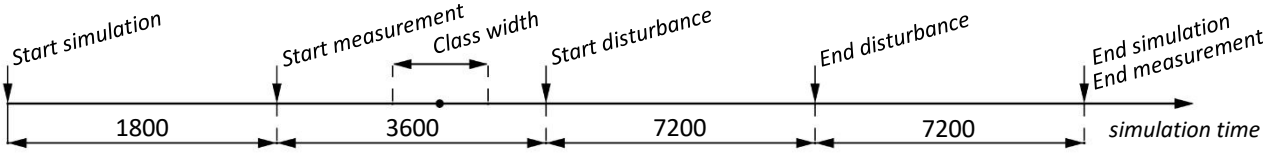


Figure 21 – Schematic timeline of the conducted parameter variation experiments, expressed in simulation seconds

The scope of this resilience study considers a number of aircraft agents that are dynamically added and removed from the simulation environment over time. Such dynamically changing aircraft population causes difficulties when quantifying the number of performance indicators, which becomes especially true when conducting a large number of simulations (i.e. replications). Therefore, to allow for useful and meaningful simulation results the timeline of the parameter variation experiments has been split in a number of time intervals (time classes). This method allows to log the time dependent performance indicators in time intervals, which eventually allows to quantify the set of performance indicators after a number of experiments.
The values of the performance indicators are logged in the following three ways:

- *Logged at landing*: this category addresses all aircraft related data that is measured in the time period between generation and landing. The values of these performance indicators are logged in the time class that corresponds with the time of landing.
- *Logged every time interval*: this category addresses the performance indicators that are measured and logged every time interval (i.e. every 10 simulated minutes). The values of these performance indicators are cleared after every log.
- *Before and after disturbance/during disturbance*: this category addresses the performance indicators whose values are measured in the total period “before and after disturbance” and in the period “during disturbance”.

The way in which the values of each performance indicator are logged have been described in the previous section.

Note that the obtained simulation results do not consider the time period that is used to create a stabilized sequence of approaching traffic, i.e. the first 1800 simulated seconds are not covered in the boxplots, the median and standard deviation.

Presentation of simulation results and analysis

Boxplots

As described in the previous section the simulation results that are obtained during the parameter variation experiments are expressed in a series of time intervals. Boxplots have been used to visualize these time-dependent simulation results. These plots allow to visualize the interesting dynamic patterns that are measured for each performance indicator. The size of the boxes and the whiskers, and the number of outliers provide information of the dispersion in the obtained data. The spacings between the different parts of the box indicate the skewness of the obtained data.

Histogram
In addition, histograms are used to visualize the distribution of the data points in a specific time interval. Each measured performance indicator is extended by two histograms that provide more detail about the distribution of the collected data points. The first histogram will present the distribution of the measured data points in a (specific) time interval before disturbance, i.e. during steady-state approach operations. The second histogram will present the distribution of the obtained data points in a time interval during disturbance. The specific time intervals that are used in each histogram are determined by the visualized patterns of each performance indicator and the shape of the boxplots. The histograms do only consider the measured data points within the 95% confidence interval.

Confidence intervals
The used boxplots take into account all the measured data points that have been logged during the parameter variation experiments. This measured data is therefore filtered with a 95% confidence interval to ignore the effects of potential outliers. The statistical analysis (median and standard deviation) of the experiment results is performed using data out of the 95% confidence interval only. By excluding the upper and lower 2.5% of the measured data the obtained simulation results do better describe the outcomes of the model in the situations before, during and after disturbance.

Median
The median ($\tilde{x}$) is used to indicate the middle value in the ordered set of measured data points for a given time interval. The reason for choosing the median is the observed distribution of the data points in each time interval. The median approaches in some sense the function of the mode. The mean is not considered because of the misleading skewing effects of outliers. The median is found to be a suitable measure to describe the dynamics that have been observed in the measured performance indicators.

Standard deviation
The standard deviation (σ) at last is used to indicate the dispersion in the obtained data points for a given time interval.

The main method for the analysis of the simulation results has been the use of the boxplots over a series of time intervals. This method allows for a relatively quick understanding of the dynamics in the measured performance indicators, and allows to examine which parameter settings enable viable and resilient approach operations during disturbance. However, the comparison of different scenarios in the parameter variation experiments can only be performed in a qualitative manner when using boxplots over a number of time intervals. The simulation results of each experiment are therefore expressed in tables, which allows for a quantitative comparison of the different simulated scenarios. These tables contain the median ($\tilde{x}$) and standard deviation (σ) of each measured performance indicator for a specific time interval before disturbance and during disturbance. The time interval before disturbance is used to quantify the steady-state values of each performance indicator, which is therefore kept fixed at time interval 0:30. The specific time interval that is used to quantify a performance indicator during disturbance is on the other hand variable. The measured median and standard deviation of a performance indicator during disturbance correspond to the time interval having the largest deviation between its own median value and the median value of the respective steady-state time interval. This means that the time intervals between 1:20 and 2:00 are in general used to express the measurements of the performance indicators during disturbance. The median and standard deviation values that show significant and interesting variations between the simulated scenarios are highlighted in bold.

Number of simulations
Each scenario in each experiment should be simulated a sufficient number of times in order to obtain statistically significant simulation results. The specific number of replications is defined by comparing the simulation results of relatively many simulations (i.e. 600x) with those as obtained after relatively less simulations (i.e. 300x). The comparison showed that roughly the same simulation results and corresponding statistics are already obtained after 300x replications, which indicates that the simulation results have stabilized and converged. Therefore, considering the specifications and performance of the used hardware each experiment is performed using 300 simulations.

8.2 General analysis of simulation results

The experimentation phase of this study can be described by the many simulations that are performed to visualize and to capture the emergent and resilient behaviours of approach operations during a bad weather disturbance. The set of identified performance indicators allows to measure and to understand this emergent and resilient behaviour. Simulations have been performed to explore how the modelled approach operations emerge before, during and after disturbance using default parameter settings (as specified in appendix C). This section will provide the general (resilience) characteristics and dynamic patterns that can be observed in the obtained (default) simulation results. These observed characteristics and patterns are meant to support the simulation results of the conducted parameter variation experiments, which are described in section 8.3.

8.2.1 Common characteristics and patterns in obtained simulation results

Each of the measured performance indicators reveals the dynamic emergent behaviour that is captured during the number of simulations. These performance indicators describe how the modelled approach operations behave before, during and after the bad weather disturbance. When analysing the results of the conducted experiments there can be seen that the performance indicators share similar type of patterns in each experiment. This section will therefore introduce the reader to the common characteristics and dynamic patterns that have been recognized when analysing the measured performance indicators. These dynamic patterns reveal how the measured data points of each performance indicator change over time as the result of a changed runway and throughput capacity. The specific way in which a performance indicator is affected by the disturbance is however always dependent on the specific scenario (i.e. parameter settings) and experiment that is considered. Appendix B provides therefore the dynamic simulation results for each experiment and scenario separately. The type of patterns that are observed in all simulation results are however comparable with the ones that will be shown below. The next section will elaborate more on the resilience characteristics that can be observed in the obtained simulation results. Section 8.2.3 will elaborate more on the factors that shape the dynamic patterns in each measured performance indicator.

Figure 22 shows two characteristic trajectory plots of simulated arrival and approach operations in the situation before and during disturbance. The densities of the logged coordinates in figure 22a show that the simulated arrival and approach operations in the situation before disturbance are in general flown using the fixed STAR procedures, the tromboning merging technique and the straight trajectory of the intermediate- and final approach segments. Only a few replications require vector and holding operations as the result of *saturation*, or as the result of conflicts on the STAR profile. The trajectory plot in figure 22b clearly shows the execution of the various modelled adaptive strategies that become required to lower the throughput capacity in the approach sectors. The obtained (common) simulation results of the multiple measured performance indicators will be used to further clarify the operations and the corresponding trajectories that are shown in figure 22.

(a) Characteristic simulated trajectories before disturbance

(b) Characteristic simulated trajectories during disturbance

Figure 22 – Characteristic plots of trajectories that emerge during simulation

The sections below will discuss the common characteristics and patterns that have been observed in the simulation results of each performance indicator. These common characteristics and patterns will be discussed for each performance indicator by means of a numbered list of observations. Each of these numbered observations refers to a specific observed characteristic or pattern in the simulation results of the respective performance indicator. These characteristics and patterns can often be related to specific aspects in the formalised and implemented model, such as the various considered adaptive strategies, controller actions, applied parameter settings or the considered arrival and approach procedures. Each observation is therefore followed by an explanation about the (possible) model properties and aspects that are (considered) responsible for the respective observed characteristic or pattern in the simulation results. These explanatory descriptions do deliberately contain information that has already been covered in chapter 5 to remind the reader of the modelled aspects, instructions, dynamics, etc. The numbers in front of each observation refer to the encircled numbers in the boxplot figure of the corresponding performance indicator. Note that the bad weather disturbance starts in time interval 1:10 and ends in time interval 3:10.

Number of landings (figure 23)

1. *Constant median values before, during and after disturbance*: The total number of landings that can be achieved is dependent on a number of factors, such as flight dynamics, wind conditions, the applied merging rate, the current runway capacity and the traffic density in the approach sector. The last three factors are found to be important for defining the specific size of the number of landings both before and after disturbance, and during disturbance. As can be seen in figure 23 the median values do generally not vary more than 1 landing per time interval, no matter what type of experiment is considered. The median values during disturbance are representative for each type of conducted experiment, since the model considers a fixed reduced runway capacity (i.e. 21 ac·h^{-1}) and a fixed corresponding merging rate. The relatively small spread in obtained data points for each time interval indicates that a relatively constant number of aircraft are positioned on the intermediate- and/or final approach segments during both weather conditions.
2. *Rapid decrease in median values just after the start of the disturbance*: The TWR controller is modelled to maintain a different *desired spacing* when it is informed about (i.e. observes) changed weather conditions at the airport. In reaction to this message (and observation) the TWR controller will adjust the spacing between its aircraft, which are all positioned on final approach. The initiation of go-around instructions is generally the only strategy that a TWR controller can apply for this purpose, since its aircraft are already close to the runway threshold. This means that in the situation of a relatively large reduction in runway capacity in combination with a relatively large initial throughput capacity at least one go-around instruction is likely to be instructed. A go-around removes, as it were, an aircraft from final approach. This 'removal' is shown in figure 23 by a rapid decrease in the *number of landings*, instead of a more gradual reduction in runway capacity. The increase in the number of landings can on the other hand be characterized by a more gradual course (i.e. the transition phase as denoted by ③).

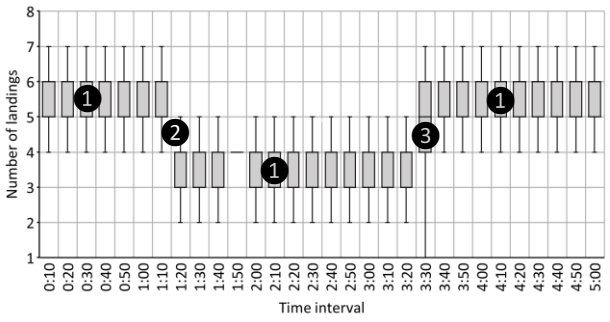


Figure 23 – Characteristic pattern in obtained simulation results of performance indicator "Number of landings"

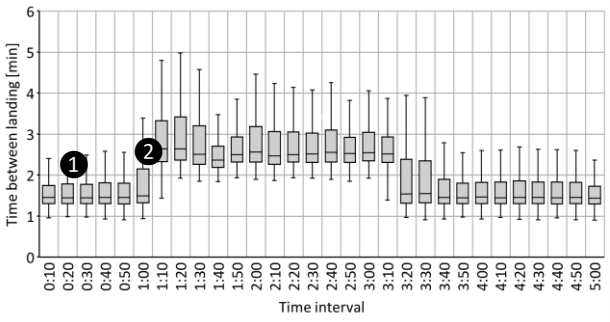


Figure 24 – Characteristic pattern in obtained simulation results of performance indicator "Time between landing"

Time between landing (figure 24)

1. *Spread and minimum values in each time interval:* This explanation extends the explanations as provided for the *number of landings* performance indicator. Figure 24 shows that the minimum time period between two successive landings equals around one minute. This time period is the result of the specific set of *time buffers* that are applied by the ARR and TWR controllers to define the *desired spacing* between aircraft. The maximum *time*

between landing is unbounded and defined by the traffic density in the approach sector. The spread in the obtained simulation results will increase when the generation rate as applied by the feeder controller is decreased.

2. *Rapid increase in median values just after the start of the disturbance*: See the explanation that is provided for observation 2 in the simulation results of the *number of landings* performance indicator. The rapid increase in the *time between landing* just after the start of the disturbance can therefore also be explained by the initiation of *go-arounds*.

Number of go-arounds (figure 25)

1. *Go-arounds are generally only instructed just after the start of the disturbance*: As described in the model description the TWR controller is modelled to maintain a specific spacing between the aircraft on final approach for each *capacity mode*. When the TWR controller is operating in a *normal capacity mode* it will try to separate these aircraft according the reduced separation minima to maximize runway capacity. In a *reduced capacity mode* the TWR controller will try to separate aircraft according the enlarged *desired spacing* to comply with the reduced runway capacity. *Go-arounds* will be instructed by the TWR controller in either of the two situations when insufficient spacing is observed. Figure 25 shows that go-arounds are on average not flown in the *normal capacity mode*, i.e. before and after disturbance. The few *go-arounds* that are instructed in the *normal capacity mode* are the result of high traffic densities in the approach sector in combination with a relatively high merging rate as applied by the ARR controller. Most *go-arounds* are however instructed just after the start of the disturbance. These go-arounds become required as the result of a reduced runway capacity. The simulation results show that a maximum of one to two *go-arounds* need to be instructed as the result of a sudden reduction in runway capacity.

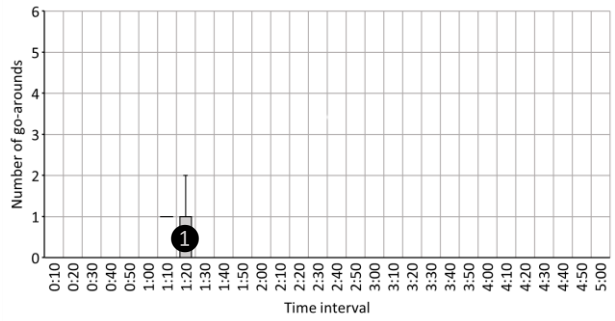


Figure 25 – Characteristic pattern in obtained simulation results of performance indicator "Number of go-arounds"

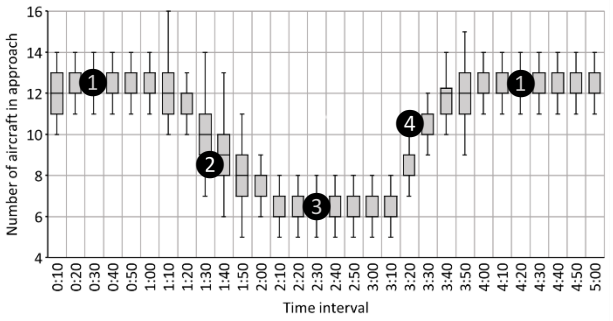


Figure 26 – Characteristic pattern in obtained simulation results of performance indicator "Number of aircraft in approach"

Number of aircraft in approach (figure 26)

1. *Constant median values before and after disturbance*: The obtained simulation results show that the approach sector contains an on average constant number of aircraft in the time period before and after disturbance. This constant *number of aircraft in approach* indicates that a stabilized and condensed flow of approaching traffic has been established in the simulation environment when the measurements start. The on average constant *number of aircraft in approach* is defined by the ratio between the maintained throughput capacity at the entry points (i.e. generation rate) and the achieved runway capacity (i.e. landing rate). The constant values indicate that the used parameter settings do not result in an undesired build-up of traffic in the approach sectors.
2. *Gradual decrease in median values just after the start of the disturbance*: The gradual decrease in the *number of aircraft in approach* just after the start of the disturbance can be related to a number of factors. The main factor in this context is the rate at which aircraft are delivered by the feeder controller to the approach sector. The feeder controller is modelled to maintain a different throughput capacity (i.e. delivery rate) for each *capacity mode*. Less aircraft will be delivered to the approach sector once the feeder controller is informed about a reduced runway/throughput capacity. This lower delivery rate will in turn result in a decrease in the *number of aircraft in approach*. Other (minor) factors that contribute to a decreased *number of aircraft in approach* are the initiation of *go-around* and *vector outbound trombone* instructions. Aircraft that are operating a *go-around* will eventually be removed from the simulation environment. Aircraft that are not considered available for a *vector inbound trombone* instruction while operating the *vector outbound trombone* instruction will also eventually be removed from the simulation environment. (Figure 18 visualizes the three ways in which aircraft may be removed from the simulation environment)

3. *Constant median values during disturbance*: A new stable *number of aircraft in approach* can be achieved during disturbance once a new stable equilibrium has been established between the rates at which aircraft enter and leave the approach sector. Such equilibrium may however not always be established. Section 8.2.3 will further elaborate on the effect of throughput capacity (i.e. generation rate) and runway capacity in establishing a new equilibrium.
4. *Gradual increase in median values just after the end of the disturbance*: The gradual increase in the *number of aircraft in approach* can be explained by the increased rate at which aircraft are delivered to the approach sector once the feeder controller is informed about a recovered runway capacity. This increase will continue until a new equilibrium is established between the maintained delivery rate and achieved landing rate.

Percentage STAR (figure 27)

1. *Constant median values before, during and after disturbance*: Figure 27 shows that around 60-80% of the total number of aircraft in the approach sector are operating the STAR procedures. This percentage is only applicable in the situation when all arrival and approach operations are flown using standard procedures (this will become clear in the analysis of the other performance indicators). The measured percentages indicate that a major part of the modelled arrival operations are flown via the fixed profiles of the STAR procedures, which is also supported by the trajectory plot in figure 22a.
2. *Rapid decrease in median values just after the start of the disturbance*: Most of the aircraft in the approach sector are operating the STAR procedure before disturbance (figure 27). Each approach controller is modelled to keep these aircraft separated according a specific minimum *desired spacing* such that the resulting sequence of aircraft complies with the maintained throughput capacity. The *actual spacings* between these aircraft do generally approach the respective *desired spacing*. Due to the relatively large number of aircraft that are operating the STAR procedure before disturbance the *actual spacings* between these aircraft are generally small. The *actual spacings* that are achieved before disturbance do therefore generally not satisfy the reduced throughput capacity that applies when the respective approach controller is informed about a reduced runway capacity. The rapid decrease in the STAR operations can therefore be explained by the relatively large number of *vector outbound STAR* instructions that need to be provided to satisfy the increased *desired spacings*.
3. *Increase in median values just after the start of the disturbance*: An increasing number of aircraft satisfy the enlarged *desired spacing* that is achieved by operating the *vector outbound STAR*, which allows them to re-operate the STAR procedure.
4. *Jump in median values just after the end of the disturbance:* The feeder controller is modelled to deliver aircraft at the entry points according the initial throughput capacity once it is operating in the *normal capacity mode* again. This recovered delivery rate results in an increase in the number of aircraft that are operating the arrival segment via the STAR procedures. The *actual spacings* between these newly (to be) generated aircraft are all smaller than the *actual spacings* that can be observed in the ARR sector just after the end of the disturbance. This is because the controllers require a certain time period in which the traffic situation in each approach sector can be adapted to the recovered throughput capacities. Aircraft are for this reason still merged with a relatively small rate just after the end of the disturbance. This implies that the achieved throughput capacity near the entry points and in the time frame just after the end of the disturbance is larger when compared to the throughput capacity that is maintained and achieved in the airspace sections closer to the airport. This temporary difference between both achieved capacities is expressed in the simulation results by a relatively small increase in the number of aircraft that are operating the STAR procedures.

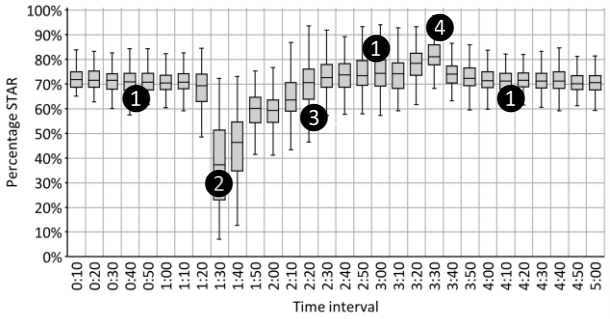


Figure 27 – Characteristic pattern in obtained simulation results of performance indicator "Percentage STAR"

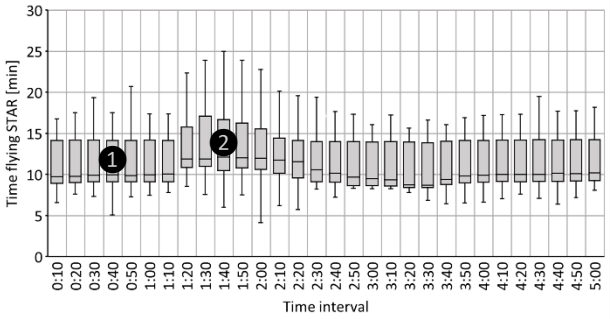


Figure 28 – Characteristic pattern in obtained simulation results of performance indicator "Time flying STAR"

Time flying STAR (figure 28)

1. *Constant median values before and after disturbance*: Each aircraft needs to operate a fixed distance of the STAR procedure to reach the *downwind segment* of the trombone segment, after which the merging operations towards the IF can be initiated using *vector inbound IF* instructions. The on average constant median values before and after disturbance can therefore be explained by the (minimum) time period that aircraft require to reach the trombone segment. Note however that the obtained simulation results of the *time flying STAR* performance indicator relate to the operating times of both STAR procedures. The relatively large overall spread in the measured data points can therefore be explained by the different lengths of both STAR procedures, and thus the different required operating times.
2. *Increase in median values during disturbance*: Aircraft have to operate a larger part of the trombone segments in order to gain the required (enlarged) *desired spacing* for a feasible merging operation relative to the IF.

Percentage vector STAR (figure 29)

1. *Increase in median values just after the start of the disturbance*: Aircraft are instructed to operate the *vector outbound STAR* to lower the throughput capacity in the approach sectors as the result of the bad weather disturbance, which means that they no longer operate the fixed profile of the STAR procedure (i.e. ② in figure 27).
2. *Decrease in median values during disturbance:* Aircraft are instructed to operate the *vector inbound STAR* once the achieved spacing as the result of the *vector outbound STAR* is considered to be in accordance with the reduced throughput/runway capacity, which means that they are allowed to resume their arrival operations via the fixed STAR procedure (i.e. ③ in figure 27).

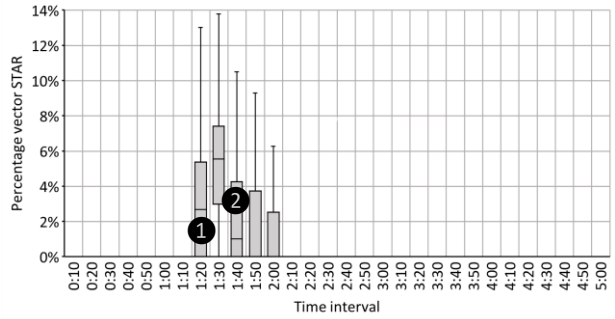


Figure 29 – Characteristic pattern in obtained simulation results of performance indicator "Percentage vector STAR"

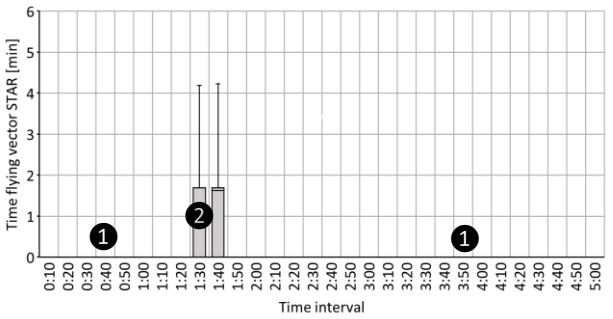


Figure 30 – Characteristic pattern in obtained simulation results of performance indicator "Time flying vector STAR"

Time flying vector STAR (figure 30)

1. *Vector STAR operations are (generally) not flown before and after disturbance*: A *vector outbound STAR* is instructed once a *desired spacing conflict* or *separation minima conflict* is detected relative to the fixed profile of the STAR procedure. These specific type of conflicts are generally not likely to occur if aircraft are already properly delivered (i.e. generated) at the entry points, i.e. the average spacing between aircraft satisfies the maintained throughput capacity. However, there may emerge some rare situations where aircraft are sequenced too close behind each other at the entry points because of a relatively high maintained throughput capacity. This initially small separation distance in combination with the effects of compression could eventually result in a conflicting situation while aircraft are operating the STAR, even without considering the bad weather disturbance. Such type of conflicting situation is resolved by the controller using a *vector outbound STAR* instruction. The probability of this specific type of conflict in undisturbed conditions increases in proportion with the generation rate as applied by the feeder controller (NE controller). V*ector outbound STAR* operations are however generally not flown before and after disturbance, which can be seen visualized by the small number of logged coordinates near the profile of the STAR procedure in the trajectory plot of figure 22a, and by the absence of boxes in the time intervals before and after disturbance in figure 30.
2. *Vector STAR operations are (generally) only flown during disturbance*: See the explanation that is provided for observation 2 in the simulation results of the *percentage STAR* performance indicator.

Percentage vector merge (figure 31)

1. *Increase in median values just after the start of the disturbance*: *Vector merge* operations become required once the trombone segments of the STAR procedures are no longer able to facilitate feasible merging operations towards the IF as the result of *saturation*. As described in section 4.3.2, *saturation* of the trombone segments may especially occur in the situation of relatively large traffic densities in the ARR sector and/or when the ARR controller is applying relatively large *desired spacings*. Both conditions hold just after the start of the disturbance, which explains the increase in the number of *vector merge* operations just after disturbance.
2. *Decrease in median values during disturbance*: *Vector merge* operations are instructed to fictitiously extent the trombone segments. These instructions are especially required just after the start of the disturbance, since the ARR controller will start to merge incoming traffic according the enlarged *desired spacings*. This implies that aircraft have to operate initially a larger part of the trombone segments, often leading to an increase in *vector merge* operations. The number of *vector merge* operations will decrease again once all aircraft in the ARR sector are separated according the enlarged *desired spacings* and when the throughput capacity at $w_{S,\mathrm{R}}$ has been reduced accordingly.

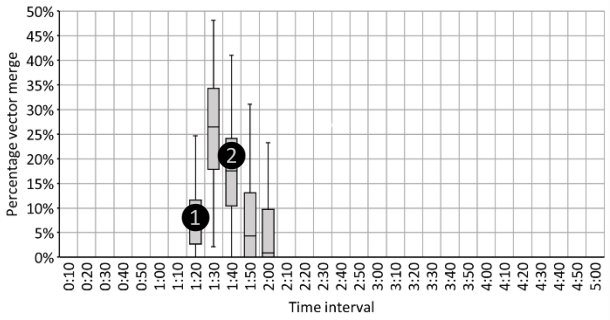


Figure 31 – Characteristic pattern in obtained simulation results of performance indicator "Percentage vector merge"

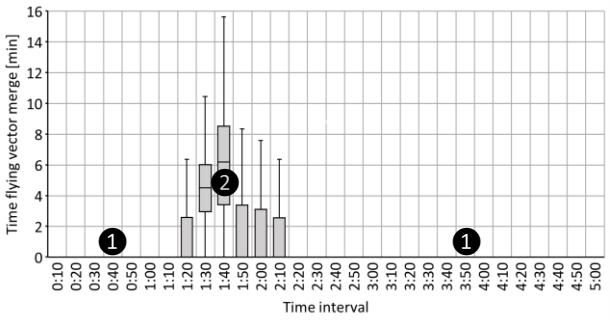


Figure 32 – Characteristic pattern in obtained simulation results of performance indicator "Time flying vector merge"

Time flying vector merge (figure 32)

1. *Vector merge operations are (generally) not flown before and after disturbance*: In undisturbed conditions the trombone segments are generally of sufficient size to merge two incoming streams of high density traffic relative to the IF and with a rate that is in accordance with the maintained runway capacity. This means that *vector merge* operations are not very common before and after disturbance. The obtained simulation results show that *vector merge* operations are only flown occasionally before and after disturbance when the ARR sector is containing a relatively large number of aircraft as the result of a relatively large maintained and achieved throughput capacity.
2. *Vector merge operations are (generally) only flown during disturbance*: As described, *vector merge* operations are instructed by the ARR controller when its merging practice is challenged due to *saturated* trombone segments. *Saturation* of the trombone segments is likely to occur in the situation when the ARR sector is containing a relatively large number of aircraft and when the corresponding *desired spacings* have to be increased. The *vector merge* operations that are flown during disturbance can therefore be explained by these specific conditions. The extent to which *vector merge* operations will be flown is determined by the size of the throughput capacities that are maintained before disturbance.

Percentage vector IF (figure 33)

1. *Constant median values before, during and after disturbance*: Figure 33 shows that around 25-30% of the total number of aircraft in the approach sector are operating the *vector IF*. The *vector IF* relates to the tromboning merging technique that is applied by the ARR controller in the initial approach segment. This specific type of modelled vector is used to connect the arrival segment (i.e. STAR procedures) with the intermediate approach segment (i.e. interception of IF). The measured percentages in the *percentage STAR* (① in figure 27) and *percentage vector IF* performance indicators show that the modelled arrival segment and initial approach segment are generally only flown using STAR and *vector (inbound) IF* operations before and after disturbance.
2. *Rapid decrease in median values just after the start of the disturbance*: The ARR controller will start to enlarge the maintained *desired spacings* in its sector when it is informed about a reduced runway capacity. These spacings will be enlarged first for aircraft that are already operating the *vector (inbound) IF*. This implies that the

maintained merging rate is reduced and that the to be merged aircraft have to wait longer before they are instructed to operate a *vector (inbound) IF* (② in figure 28). The decrease in the number of *vector IF* operations just after the start of the disturbance can therefore be explained by this reduced merging rate and by an increase in the number of *vector STAR*, *vector merge* and *holding* operations.

3. *Increase in median values during disturbance*: The ARR controller is modelled to maintain a constant reduced merging rate during disturbance. The obtained simulation results seem to suggest however that an increasing number of aircraft is operating the *vector IF* during disturbance. This increase in the significance of *vector IF* operations can be explained by the reduced throughput capacity that is maintained by the feeder controller in the time period during disturbance, which results in a reduction of the *number of aircraft in approach* (② in figure 26).
4. *Drop in median values just after the end of the disturbance*: This description extends the explanation that is provided for observation 4 in the simulation results of the *percentage STAR* performance indicator. These simulation results showed an increase in the number of aircraft that are operating the STAR procedure just after the end of the disturbance. As described, this increase can be explained by the increased rate at which aircraft are delivered to the approach sector. This recovered delivery/generation rate exceeds initially the rate at which aircraft are merged relative to the IF using *vector (inbound) IF* instructions. The relatively small merging rate is the result of the reduced throughput capacity that is maintained in the approach sectors during disturbance. After the end of the disturbance the ARR controller needs time to increase the throughput capacity in its sector, and with that the merging rate. The relatively lower number of *vector IF* operations can be explained by this to be recovered merging rate. Note that the reduction in *vector IF* operations is cancelled out by an increase in the number of STAR operations (④ in figure 27). The simulation results of the *percentage STAR* and *percentage vector IF* performance indicators show that a new equilibrium is established in the number of STAR and *vector IF* operations (① in figures 27 and 33) once the throughput capacity in the ARR sector (and thus the merging rate) has been recovered.

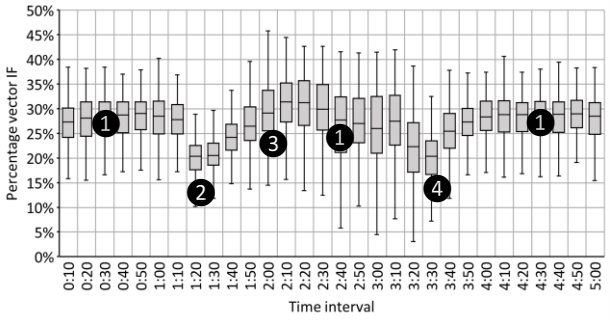


Figure 33 – Characteristic pattern in obtained simulation results of performance indicator "Percentage vector IF"

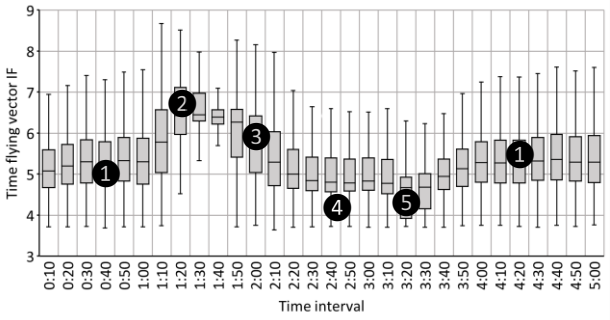


Figure 34 – Characteristic pattern in obtained simulation results of performance indicator "Time flying vector IF"

Time flying vector IF (figure 34)

1. *Constant median values before and after disturbance*: Each aircraft can only be instructed the *vector IF* once it is positioned on the trombone segment of the STAR procedure. The *time flying vector IF* is defined as the time period that is required by an aircraft to intercept the IF after leaving the fixed profile of the trombone segment. The obtained simulation results show therefore that each aircraft is on average leaving the trombone segment (i.e. *downwind segment*) at a fixed location before and after disturbance. This constant can be explained by the rather constant throughput capacity that is applied by the feeder controller before and after disturbance, which results in a constant flow of approaching traffic entering the trombone segments.
2. *Increase in median values just after the start of the disturbance*: The observed increase in the *time flying vector IF* performance indicator has two types of explanations. The first explanation relates to the aircraft that are already operating the *vector (inbound) IF* when the ARR controller is informed about a reduced runway/throughput capacity. The ARR controller will upon reception of this capacity update first enlarge the spacing between the aircraft that are already operating the *vector inbound IF* instruction. The spacing between these aircraft is enlarged using a *vector outbound IF* instruction. The direction of this vector is always such that it sends aircraft away from the IF, which allows the aircraft to gain the required spacing. A *vector inbound IF* will be instructed again once the respective aircraft is separated according the reduced throughput capacity. The need for these temporary *outbound vectors* does therefore increase the duration of the *vector IF* operations just after

disturbance. The second explanation relates to the aircraft that have not been yet instructed a *vector (inbound) IF*, i.e. the aircraft that are still positioned on the trombone segment of the STAR procedure. As already described, the ARR controller will apply a reduced merging rate when it is operating in a *reduced capacity mode*. This means that aircraft have to operate a larger part of the trombone segment (i.e. *downwind segment*) before they are considered available for a *vector inbound IF* instruction. When considering the orientation of the trombone segment this means that the distance that has to be covered with the *vector inbound IF* instruction will increase as well.

3. *Decrease in median values during disturbance*: This decrease can primarily be explained by the reduced rate at which aircraft are delivered to the approach sector and after some time to the ARR sector. This reduced delivery rate results in a lower traffic density within the ARR sector, which means that most separation distances are already or almost in accordance with the reduced merging rate as applied by the ARR controller. In such situation aircraft do generally not need to operate a relatively large part of the trombone segment in order to gain the required spacing for a feasible merging operation. The decrease in the length and duration of the *vector IF* operations during disturbance can therefore be explained by this reduced need to operate large parts of the trombone segments.
4. *Constant median values during disturbance:* The TNW/TNE controllers are modelled to maintain a reduced throughput capacity in their approach sectors when they are operating in a *reduced capacity mode*. As the result of these reduced and maintained throughput capacities the rate at which aircraft are handed over to the ARR sector will be reduced. Due to a lower number of aircraft entering the ARR sector the traffic density in the ARR sector will reduce as well. Because of these smaller traffic densities in the ARR sector aircraft do not need to operate a large part of the trombone segment in order to be considered available for a *vector inbound IF* instruction. The reduced constant median values in the *time flying vector IF* performance indicator can therefore be explained by the lower number of aircraft in the ARR sector during disturbance.
5. *Drop in median values just after the end of the disturbance*: Once the ARR controller is informed about a recovered runway capacity it will start to merge the incoming streams of traffic according the initial merging rate again. This initial merging rate is achieved by using smaller *desired spacings*. The relatively small number of aircraft in and near the ARR sector do however already satisfy these separation criteria at the end of the disturbance because of the larger *desired spacings* that were maintained during disturbance. This means that these specific aircraft can be instructed the *vector inbound IF* in a relatively quick succession in this specific time frame. The drop in the median values just after the end of the disturbance can be related to this increased and applied merging rate. The measured values of the *time flying vector IF* performance indicator will increase thereafter again because of the recovered throughput capacities that are maintained in the approach sectors.

Percentage holding (figure 35)

1. *Increase in median values during disturbance*: The TNW/TNE controllers are modelled to initiate *holding* operations when the trombone segments of both STAR procedures become *saturated* as the result of large traffic densities in the ARR sector. The simulated arrival and approach operations before and after disturbance can in particular be described by these large traffic densities due to the relatively high throughput capacities that are applied. Such high density traffic situation is likely to saturate the trombone segment(s) when the maintained throughput capacity has to be adjusted in reaction to the reduced runway capacity. This so-called *saturation* is initially resolved using *vector merge* instructions only (i.e. figures 31 and 32). *Holding* operations are eventually instructed once too many aircraft are operating the *vector merge* simultaneously. These *holding* instructions are required to prevent that an excessive number of aircraft is being vectored towards the airspace section north of the trombone segments. When comparing the simulation results of the *percentage vector merge* and *percentage holding* performance indicators there can be seen that *holding* operations are initiated to (successfully) bring back the number of *vector merge* operations in the ARR sector. The increase in the number of *holding* operations as observed in the simulation results can therefore be explained by the relatively large number of aircraft that are currently operating the *vector merge* instruction.
2. *Decrease in median values during disturbance*: *Holding* operations are thus initiated to enable the ARR controller to eliminate the number of vector operations (i.e. *vector STAR* and *vector merge*) in the ARR sector such that standard approach operations can be resumed. Aircraft will be released again from the holding stack once the trombone segments are observed to be no longer *saturated*, i.e. all *vector merge* operations are eliminated. The observed decrease in the number of *holding* operations during disturbance can therefore be explained by the fact that *vector merge* operations are no longer flown or are drastically reduced (i.e. ② in figure 31).

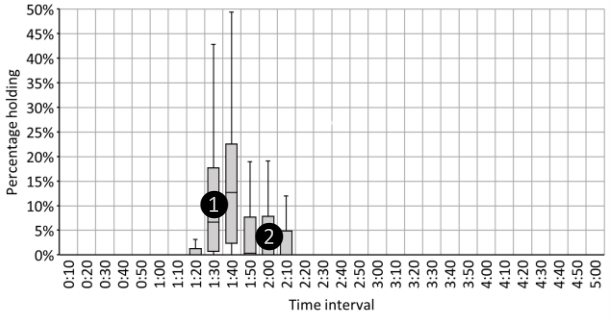


Figure 35 – Characteristic pattern in obtained simulation results of performance indicator "Percentage holding"

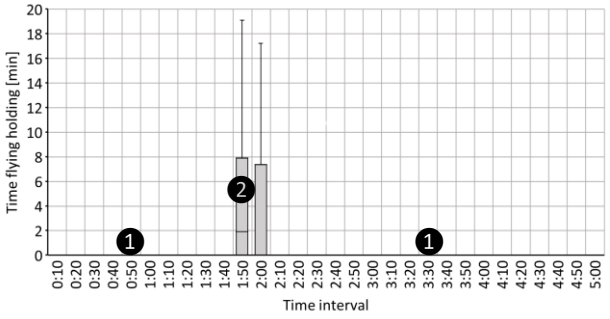


Figure 36 – Characteristic pattern in obtained simulation results of performance indicator "Time flying holding"

Time flying holding (figure 36)

1. *Holding operations are (generally) not flown before and after disturbance*: As described in the analysis of the *percentage holding* performance indicator *holding* operations are initiated to eliminate the number of *vector merge* operations in the ARR sector. The obtained simulation results of the *percentage vector merge* and *time flying vector merge* performance indicators (figures 31 and 32) indicate that *vector merge* operations are on average not flown before and after disturbance. The absence of *holding* operations before and after disturbance can therefore be explained and supported by the absence of *vector merge* operations in these same time periods. However, the trajectory plot in figure 22a shows that a few *holding* operations are already flown before disturbance. These rare occurrences can be explained by the high traffic densities that sometimes may emerge in the ARR sector as the result of the used parameter settings.
2. *Holding operations are (generally) only flown during disturbance*: The obtained simulation results show that vector operations are generally only instructed during disturbance to comply with the reduced runway/throughput capacity. *Holding* operations are for this same reason also (generally) only applied during disturbance. The durations of these *holding* operations, and with that the number of aircraft in the holding stack are defined by the size of the capacity reduction, the traffic density in the ARR sector just before disturbance, the rate at which aircraft are delivered by the feeder controller to the approach sector and the rate at which aircraft are released from the holding stack to enter the ARR sector.

Note that the simulation results of the *number of vector outbound trombone* and *number of vector inbound trombone* performance indicators as shown below are expressed by mean values instead of median values (figures 37 and 38). This is done since all the measured median values for these performance indicators are equal to zero. The shape of the mean line does however still provide some information about the use of *vector trombone* operations before, during and after disturbance.

Number of vector outbound trombone (figure 37)

1. *Vector outbound trombone operations are (generally) only flown during disturbance*: The *vector outbound trombone* instruction is used by the ARR controller to resolve conflicting situations near the IF or is used when an aircraft is not considered available (anymore) for a *vector inbound IF* instruction due to its position relative to the IF. Both purposes are related to the merging practice as applied by the ARR controller using *vector inbound IF* instructions. Conflicts between aircraft that are operating the *vector inbound IF* are mostly resolved at an early stage using *vector outbound IF* instructions only. These early interventions ensure that in general all aircraft can be properly merged relative to the IF, which prevents the necessity of *vector outbound trombone* instructions. However, conflicts between aircraft that are operating the *vector inbound IF* can also occur while they are already close to the IF. In such situation the respective aircraft is instructed the *vector outbound trombone*, since the aircraft is no longer able to intercept the IF once it is operating such instruction. As can be seen in the simulation results the *vector outbound trombone* operations are in particular instructed just after the start of the disturbance. This increase in the number of *vector outbound trombone* instructions can be related to the aircraft that are about to intercept the IF when the ARR controller is informed about a reduced runway capacity. Since these aircraft are already close to the IF the ARR controller has insufficient time to enlarge their spacing using *outbound vectors*. In these typical situations the *vector outbound trombone* instructions become required. The trajectory plot in figure 22b clearly shows the flown *vector outbound trombone* operations during disturbance.

2. *Low overall number of vector outbound trombone instructions*: The measured mean values indicate that the number of *vector outbound trombone* instructions can be attributed to the adapted traffic situation during disturbance. The measured values also indicate that these type of vectors are not often instructed, which can be explained by the low number of conflicts that emerge near the IF while aircraft are operating the *vector inbound IF*. The low number of conflicts near the IF can be related to the quite accurate merging practice of the ARR controller.

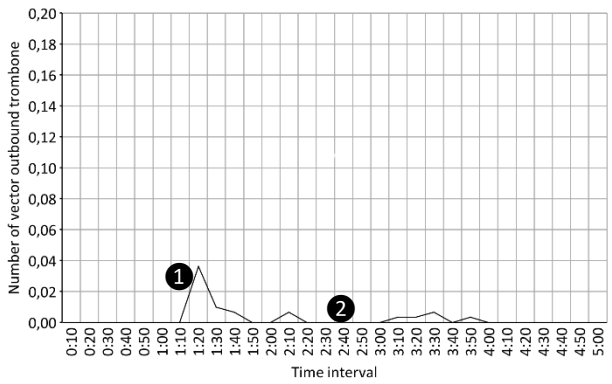


Figure 37 – Characteristic pattern in obtained simulation results of performance indicator "Number of vector outbound trombone"

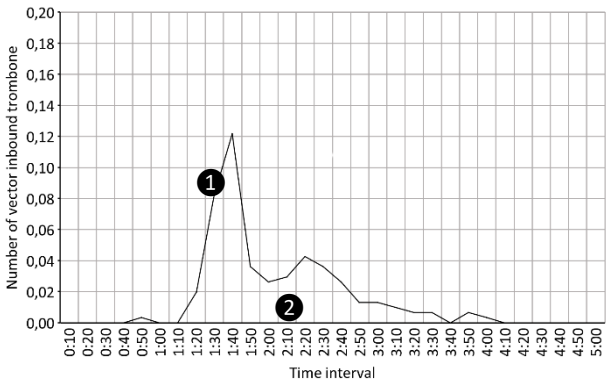


Figure 38 – Characteristic pattern in obtained simulation results of performance indicator "Number of vector inbound trombone"

Number of vector inbound trombone (figure 38)

1. *Most vector outbound trombone instructions are followed by a vector inbound trombone instruction*: The direction of the *vector outbound trombone* instruction is always such that it guides aircraft away from the dense ARR sector, which makes that these vectored aircraft cannot easily be integrated again in the 'old' merging sequence. The *vector inbound trombone* instruction is therefore used as a sequel to the *vector outbound trombone* instruction to integrate aircraft again in the merging sequence. The sequence in which aircraft are to be merged is defined once the approaching aircraft have passed the entry point of the trombone segment. The *vector inbound trombone* instruction is for this reason directed towards this specific point. The *vector inbound trombone* instruction is however only provided if the to be vectored aircraft is to be sequenced in between the regular stream of approaching aircraft. The simulation results of the *number of vector outbound trombone* and *number of vector inbound trombone* performance indicators indicate that most *vector outbound trombone* instructions are indeed followed by a *vector inbound trombone* instruction. This means that there is only a very small probability that aircraft will be removed from the simulation environment as the result of infeasible *vector inbound trombone* instructions. In this way the majority of the small number of aircraft that are instructed the *vector outbound trombone* will be able to re-join and re-operate the trombone segment in order to be considered available again for a merging operation towards the IF.
2. *Low overall number of vector inbound trombone instructions*: This description extends the explanation that is provided for observation 2 in the simulation results of the *number of vector outbound trombone* performance indicator. The low number of *vector inbound trombone* instructions can therefore also be explained by the low number of conflicts that emerge near the IF while aircraft are operating the *vector inbound IF*. As described in the previous explanation the number of *vector inbound trombone* instructions exceeds on average the number of *vector outbound instructions*, which implies that in general all *vector outbound trombone* instructions are followed by a *vector inbound trombone* instruction.

Time in approach (figure 39)

1. *Constant median values before and after disturbance*: The simulation results indicate that each aircraft requires on average the same amount of time to complete its arrival and approach operations before and after disturbance. The constant median values in the *time in approach* performance indicator before and after disturbance are generally a summation of the *time flying STAR* (figure 28), *time flying vector IF* (figure 34) and the remaining time period that is required to complete the intermediate- and final approach segments. The measured durations in these undisturbed conditions approach the durations of real approach operations via XIBIL2A and RITEB2A. The exact size of the measured data points is dependent on the various capacities that are applied in the approach sectors. Note however that the *time in approach* performance indicator combines the results of all

aircraft, which means that no distinction is made between aircraft that enter via XIBIL2A or via RITEB2A. The relatively large overall spread in the measured data points can therefore be explained by the different lengths of both STAR procedures, and thus the different required operating times.

2. *Increase in median values at the start of the disturbance*: In reaction to the decreased runway capacity as the result of a bad weather disturbance the controllers will start to adjust the maintained throughput capacity in their airspace sectors by applying a specific modelled adaptive strategy. The increased durations of the *time in approach* can therefore be explained by the multiple *vector* (i.e. *vector outbound STAR*, *-merge*, *-IF* and *-trombone*) and *holding* operations that are instructed at the start of the disturbance.
3. *Decrease in median values during disturbance*: Aircraft that are operating an *outbound vector* will be instructed the corresponding *inbound vector* once the gained spacing is observed to be in accordance with the reduced runway/throughput capacity. Secondly, aircraft will be released from the holding stack once the approach operations in the ARR sector have been properly adapted to the reduced capacities. The decreased durations of the *time in approach* can therefore be explained by the decreased number of (necessary) *vector* and *holding* operations.
4. *Constant median values during disturbance*: The constant median values during disturbance indicate that the flown arrival and approach operations can be described again by standard operations only (i.e. the STAR procedures, the tromboning merging technique and the intermediate- and final approach segments). These constant values imply therefore that *vector* and *holding* operations are no longer flown in this specific phase during disturbance. When comparing the size of the median values before and after disturbance with the ones during disturbance there can be seen that the flown arrival and approach operations during disturbance take (eventually) less time, which can be explained by the lower traffic densities that apply during disturbance. The specific decrease in the *time in approach* is defined by the extent to which the maintained throughput capacities have to be adjusted in each approach sector.
5. *Drop in median values just after the end of the disturbance*: The small drop in the median values just after the end of the disturbance is caused by the increased rate at which the ARR controller is merging aircraft towards the IF once it is informed about a recovered runway capacity. See the explanation that is provided for observation 5 in the simulation results of the *time flying vector IF* performance indicator for a more detailed explanation.

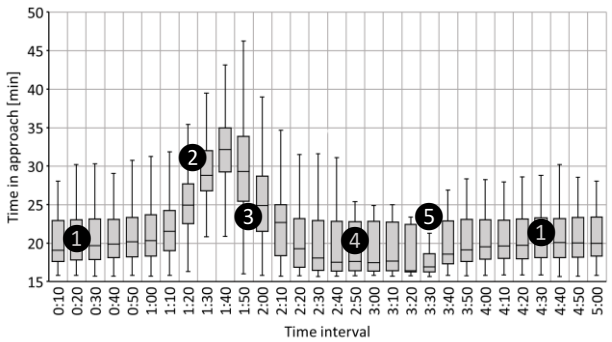


Figure 39 – Characteristic pattern in obtained simulation results of performance indicator "Time in approach"

Number of instructions (figure 40)

1. *Constant median values before and after disturbance*: These constant median values indicate that each controller provides on average a constant number of instructions per time interval before and after disturbance. The specific number of instructions that each controller has provided is determined by the specific set of instructions that each controller has been assigned and is determined by the number of aircraft where each controller is responsible for.
2. *On average unaffected median values just after the start of the disturbance (TNW/TNE controllers only)*: When the TNW/TNE controllers are informed about a reduced runway capacity they will start to reduce the throughput capacity in their sectors using *vector outbound STAR* instructions. As the result of these actions an increase in the number of provided instructions is expected in the time period just after the start of the disturbance. The simulation results indicate however that the workload of the TNW/TNE controllers in terms of the number of provided instructions is not really affected by the disturbance (i.e. time intervals 1:10 – 1:20). The on average unaffected workload of both controllers just after the start of the disturbance can be explained by the fact that

the contents of the *vector outbound STAR* and *STAR speed and/or altitude* instructions can be combined, if applicable.

3. *Decrease in median values during disturbance (TNW/TNE controllers only)*: While instructing the number of *vector outbound STAR* instructions the throughput capacities in the TNE/TNW sectors are gradually adapted to comply with the reduced runway capacity. A gradual adaptation relates herein to the time period that is needed by the vectored aircraft to gain the enlarged *desired spacing*. This 'settling' time in combination with an increasing number of aircraft that have already been instructed the *vector outbound STAR* explains the decrease in the number of provided instructions during disturbance.
4. *Increase in median values during disturbance (TNW/TNE controllers only)*: At a given point in time during disturbance the observed *actual spacings* between an increasing number of vectored aircraft will again be in accordance with the reduced throughput capacity. The TNW/TNE controllers will therefore start to provide *vector inbound STAR* instructions such that aircraft can resume their standard arrival procedures. The increase in the number of provided instructions during disturbance can therefore be explained by the increased number of *vector inbound STAR* instructions. A second reason for the increase in the *number of instructions* performance indicator during disturbance is the initiation of *holding* operations as the result of *saturation* in the ARR sector.
5. *Constant median values during disturbance*: The modelled adaptive strategies are primarily applied by the approach controllers to adapt the traffic situations in each sector to the reduced runway capacity. The traffic situation in a specific approach sector is considered to be properly adapted to the bad weather disturbance once the established throughput capacity in the respective approach sector complies with the reduced runway capacity and when all arrival and approach operations are again flown using standard procedures. The constant number of provided instructions during disturbance indicate therefore that the arrival and approach operations are properly adapted.
6. *Increase in median values at the start of the disturbance (ARR controller only)*: The two streams of simulated traffic that initially arrive via the TNW/TNE sectors will eventually both enter the ARR sector. The ARR controller is therefore responsible for the largest number of aircraft compared to the other controllers. Because of this larger number of aircraft a relatively large number of instructions is required to adapt the traffic situation in the ARR sector to the reduced runway capacity. The increased *number of instructions* just after the start of the disturbance can therefore be explained by the various *outbound vectors* (i.e. *vector outbound STAR*, *-merge*, *-IF* and *-trombone*) that are instructed by the ARR controller.
7. *Decrease in median values during disturbance (ARR controller only)*: The ARR controller applies a vectoring strategy (i.e. *vector STAR*, *-merge*, *-IF* and *-trombone*) to gradually adapt the traffic situation in the ARR sector to comply with the reduced runway capacity. After initiation of this vectoring strategy an increasing number of aircraft will eventually be separated conform the enlarged *desired spacing*. Aircraft that have gained sufficient spacing will thereafter be instructed to resume standard approach operations by the provision of *vector inbound STAR*, *-merge*, *-IF* and/or *-trombone* instructions. This gradual recovery of the standard approach operations explains the gradual decrease in the number of provided instructions during disturbance.
8. *Rapid decrease in median values just after the start of the disturbance (TWR controller only)*: The TWR controller is responsible for the provision of landing clearances and go-arounds. The rate at which both instructions are provided by the TWR controller is defined by the achieved runway capacity. The (to be) achieved runway capacity is in turn defined by the *actual spacings* on the intermediate- and final approach segments. These *actual spacings* are however again defined by the rate at which the incoming traffic streams are merged by the ARR controller relative to the IF. As described, the maintained merging rate is immediately reduced once the ARR controller is informed about a reduced runway capacity. The rapid decrease in the number of provided instructions just after the start of the disturbance can therefore be explained by this reduced merging rate.
9. *Increase in median values just after the end of the disturbance*: Aircraft will again be delivered to the approach sector according the initial delivery rate once the feeder controller is informed about a recovered runway capacity. Furthermore, each controller is modelled to achieve and to maintain the initial throughput capacity again when it is informed about a recovered runway capacity. Because of the multiple recovered capacities the number of aircraft in each approach sector will gradually increase again. The increase in the number of provided instructions just after the end of the disturbance can therefore be explained by these increased traffic densities.
10. *Large overall spread in simulation results (TNW/TNE controllers only)*: All aircraft enter the approach sector via entry points XIBIL and RITEB, which mark the northern boundaries of the TNW and TNE sectors respectively. The specific entry point where each aircraft will be generated and the rate at which aircraft will be generated are determined by probability distributions. These probability distributions ensure that each simulation (i.e.

replication of the parameter variation experiment) will generate different streams of arriving traffic. Because of these involved stochastics the resulting traffic densities in the TNW/TNE sectors will be different for each simulation, which explains the overall spread in the number of instructions that have been provided by the TNW/TNE controllers.

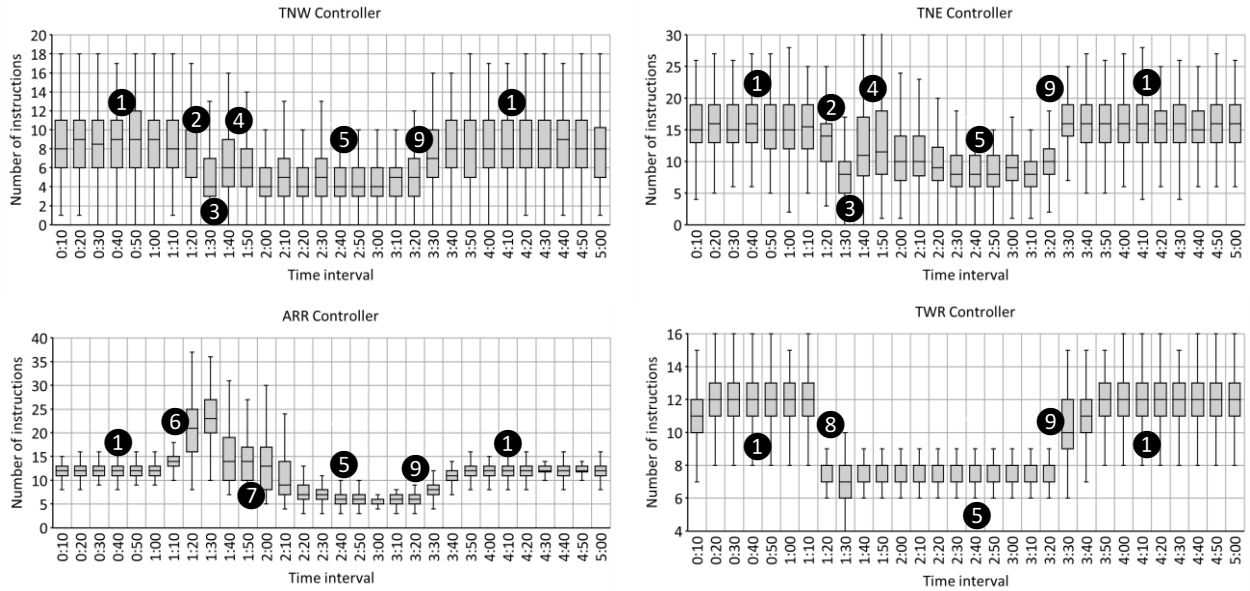


Figure 40 – Characteristic pattern in obtained simulation results of performance indicator "Number of instructions"

Percentage in tactical mode (figure 41)

1. *The TNW controller is always operating in a tactical control mode before, during and after disturbance*: The model assumes that 35% of the aircraft arrive via the TNW sector. This percentage in combination with a relatively high overall throughput capacity (e.g. 36 ac·h^{-1}) does however not result in a sufficiently high workload to make the TNW controller operate in an opportunistic control mode.
2. *The TNE controller is sometimes operating in an opportunistic control mode before and after disturbance*: A relatively large part of the total number of necessary speed and altitude adjustments take place while aircraft are arriving via the TNW/TNE sectors using STAR procedures (i.e. the *STAR speed and/or altitude* instructions). Since 65% of the arrivals originate from the TNE sector most of the provided *STAR speed and/or altitude* instructions can be ascribed to the TNE controller. Especially before and after disturbance a relatively large number of *STAR speed and/or altitude* instructions need to be provided by the TNE controller because of the relatively large number of aircraft that are operating the STAR procedure in the TNE sector. This large number of provided *STAR speed and/or altitude* instructions does therefore explain the operations in the opportunistic control mode before and after disturbance.
3. *The TNE controller is on average operating in a tactical control mode during disturbance*: As explained in the previous explanation, the TNE controller needs to instruct a relatively large number of *STAR speed and/or altitude* instructions before and after disturbance because of the relatively large number of aircraft that are operating the STAR procedure in the TNE sector. The significance of these instructions will however be less during disturbance due to the lower number of aircraft that are operating the STAR procedure. The increased workload of the TNE controller during disturbance in terms of the number of provided *vector STAR* and the various *holding* instructions is apparently not sufficiently high to let the TNE controller operate in an opportunistic control mode.
4. *The ARR controller is always operating in a tactical control mode before and after disturbance*: The workload of the ARR controller before and after disturbance in terms of the number of provided speed and altitude instructions, the various handovers and the applied tromboning merging technique is not sufficiently high to let the ARR controller operate in an opportunistic control mode.
5. *The ARR controller switches to an opportunistic control mode just after the start of the disturbance*: See the explanation that is provided for observation 6 in the simulation results of the *number of instructions* performance indicator. The switch to an opportunistic control mode just after the start of the disturbance can also be explained by the various *outbound vectors* that need to be instructed by the ARR controller.

6. *The ARR controller switches to a tactical control mode during disturbance:* See the explanation that is provided for observation 7 in the simulation results of the *number of instructions* performance indicator. The switch to a tactical control mode can also be explained by a gradual recovery of the standard approach operations in the ARR sector.
7. *The TWR controller is always operating in a tactical control mode before, during and after disturbance:* The TWR controller is responsible for the provision of landing clearances and go-arounds. The frequency at which both type of instructions are provided is however insufficiently high to let the TWR controller operate in an opportunistic control mode.

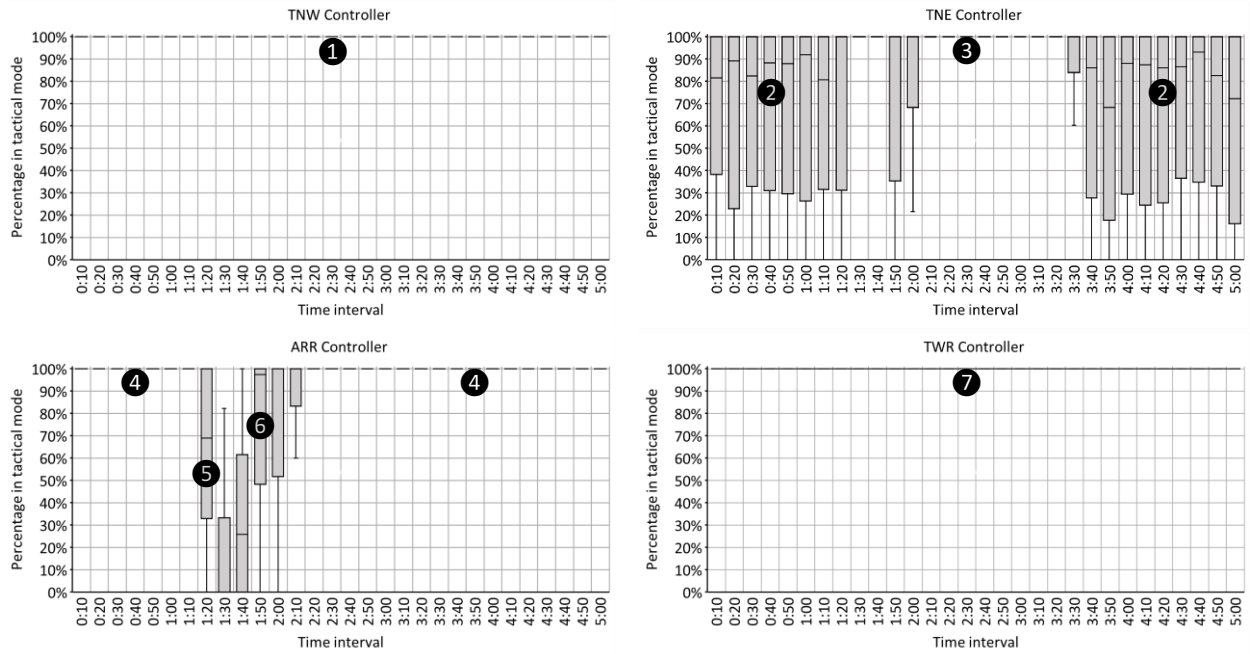
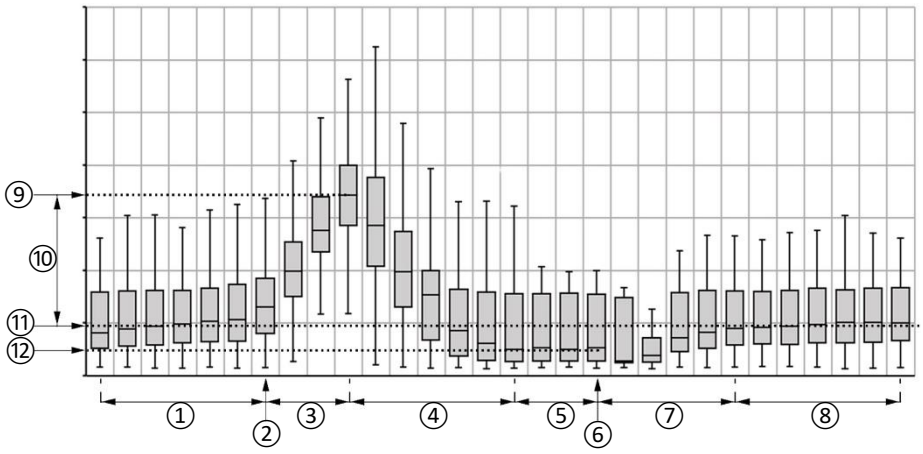


Figure 41 – Characteristic pattern in obtained simulation results of performance indicator "Percentage in tactical mode"

8.2.2 Resilience characteristics and patterns in simulation results

The previous section visualized the type of dynamic patterns that can be observed in the obtained simulation results for each specific performance indicator. Each of these dynamic patterns can be decomposed into smaller segments, where each segment describes a specific resilient characteristic and property (e.g. absorption, adaptation, recovery). The characteristics and properties of the *resilience triangle* paradigm (figure 1) have been used to discuss the resilience characteristics and patterns that can be observed in the simulation results of each performance indicator. Figure 42 visualizes a typical example in which the various characteristics and properties of the *resilience triangle* paradigm are applied to analyse the observed dynamic pattern in the obtained simulation results of the *time in approach* performance indicator. This specific performance indicator is used, since the corresponding dynamic pattern is found to be representative for many other performance indicators as well. This section is meant to indicate and to contextualize the general resilience characteristics that are observed in the obtained simulation results. These resilience characteristics are discussed using brief summaries of the related information that has been provided in the previous section. The next section will elaborate further on the factors that are found to determine these resilient characteristics.



Nr	Description	Nr	Description	Nr	Description
①	Stable original state	⑤	Stable recovered state (I)	⑨	Minimum performance
②	Start disturbance	⑥	End disturbance	⑩	Performance loss
③	Absorption phase	⑦	Recovery phase (II)	⑪	Original performance
④	Recovery phase (I)	⑧	Stable recovered state (II)	⑫	Recovered performance

Figure 42 – Resilience ("triangle") characteristics and properties that can be observed in the obtained simulation results

Stable original state

Each measured performance indicator indicates a constant and steady state performance level in the period before the start of the bad weather disturbance. These constant values indicate that all arrival and approach operations are flown using standard procedures, which means that *vector* operations, *holding* operations and *go-arounds* are generally not flown before disturbance. The constant workload of the controllers before disturbance can therefore also be explained by these normal and steady arrival and approach operations and by the steady traffic densities in each approach sector. Since all arrival and approach operations are flown using standard procedures all aircraft require on average the same time period to complete the various approach segments. The achieved runway capacity can be described as being constant before disturbance as the result of the relatively large traffic densities in the approach sectors. The specific size of the achieved runway capacity is dependent on the parameter settings that have been used for a specific experiment. All conducted experiments do however show that a runway capacity can be achieved of 28 to 36 ac·h^{-1} before disturbance.

Absorption phase

The absorption phase is initiated just after the start of the bad weather disturbance. The measured characteristics in this phase can be related to the *vector* instructions, *holding* operations and *go-arounds* that are instructed to lower the throughput capacity in the approach sectors. When analysing the simulation results two absorption phases can be identified: the *initial absorption phase* and the *secondary absorption phase*. The *initial absorption phase* comprises the (immediate) controller actions that are initiated and which become required just after the start of the disturbance. These actions relate to the provision of *outbound vectors* (i.e. *vector outbound STAR*, *-IF*, and/or *-trombone*) and *go-around* instructions in order to gradually reduce the maintained throughput capacity in each approach sector. It takes on average 10 to 20 minutes to complete the *initial absorption phase*, no matter what scenario or experiment is considered. The *secondary absorption phase* comprises the (later) controller actions that become required to cope with the *saturated* trombone segments, if applicable. Instructions that are typically provided in this *secondary absorption phase* are therefore related to *vector merge* and *holding* operations.

Recovery phase (I)

The recovery phase (I) is initiated after the performance indicators have suffered their maximum performance loss. This is typically the time point at which the throughput capacity in the approach sectors has been adjusted successfully, i.e. each aircraft in the approach sector has gained sufficient enlarged spacing with the other aircraft to comply with the new maintained *desired spacings*. Aircraft will in this situation gradually resume their original stable arrival/approach operations after the reception of a specific *inbound vector* or *holding exit* instruction. The simulation

results show that the recovery phase is on average initiated at or near time interval 1:40. After this time point one can observe a reduction in *vector merge* and *holding* operations, which indicate that the flown approach operations are recovering again. At a given point in time the trombone segments of the STAR procedures will not be *saturated* anymore as the result of a recovered traffic situation in the ARR sector. This recovered traffic situation can be seen visualized in the simulation results by relatively shorter STAR and *vector inbound IF* operations. After a certain period of time the approach operations can be described again by stabilized and constant performance indicators, which indicate stable and recovered approach operations in *reduced capacity mode*. A successful recovery of approach operations during a capacity reduction is however not obvious since it is defined by specific conditions. The next section will elaborate further on the specific conditions that determine such successful recovery.

Stable recovered state (I)

The stable recovered state (I) represents the phase in which all aircraft have resumed their original steady arrival/approach operations in *reduced capacity mode*. This means that each aircraft is separated according to the enlarged *desired spacing* to comply with the reduced throughput capacity that is maintained in the approach sectors. The stable recovered state (I) can be seen visualized in the obtained simulation results by stabilized and constant median values for the number of measured performance indicators. The stable recovered state (I) is characterized by the absence of *vector* operations, *holding* operations and *go-arounds*.

Recovery phase (II)

The recovery phase (II) is initiated just after the end of the bad weather disturbance. A normalized and recovered runway capacity applies during this phase, which means that the controllers are gradually recovering the throughput capacity in the approach sectors by applying lower *desired spacings*. The feeder controller will during the recovery phase (II) increase the generation rate, which in turn leads to an increase in the number of aircraft that are on approach. The workload of each controller will increase again as the result of an increased traffic density and throughput capacity. The increased traffic density in the ARR sector causes aircraft to operate (again) a larger part of the *downwind segment* of the STAR procedure in order to be instructed a feasible *vector inbound IF* instruction towards the IF. Since aircraft need to operate again a larger part of the trombone segments the size of the merging operations in between the trombone segments will increase as well. The increase in the number of sequenced and merged aircraft relative to the intermediate fix is represented in the simulation results by the increase in *vector IF* operations. The recovery phase (II) is completed once a stable sequence of approaching traffic has been established that complies with the original throughput capacity again.

Stable recovered state (II)

The stable recovered state (II) represents the phase after the end of the bad weather disturbance where the simulation results of the measured performance indicators show a stable and recovered state again, i.e. the flown arrival and approach operations and the corresponding throughput capacity and traffic density are similar with those before the bad weather disturbance.

8.2.3 General factors that define the resilience characteristics and patterns

During the experimentation and simulation phase there has been noticed that the resilient capacities of the modelled approach operations are highly affected by and dependent on the ratio between the generation rate (i.e. throughput capacity) as applied by the feeder controller and the merging rate (i.e. runway capacity) as applied by the ARR controller. The duration of the absorption and recovery phases and the size of the performance loss are defined by these two factors. The median and dispersion of the measured data points are besides also defined by these two factors. This section will therefore elaborate a bit further on the effects of throughput capacity and runway capacity on the observed characteristics and patterns in the measured performance indicators.

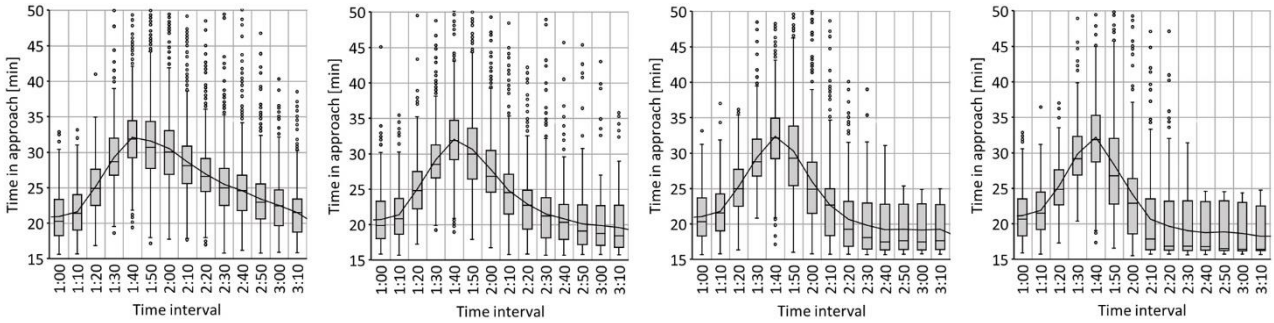
The effects of throughput capacity and runway capacity on the resilient capacities of approach operations will be discussed using the simulation results of the *time in approach* performance indicator. This specific performance indicator is considered since the corresponding resilient characteristics are found to be representative for other performance indicators as well. The obtained simulation results of the *time in approach* performance indicator (figure 39) indicated that its maximum performance loss is suffered in time interval 1:40 and that a new stable recovered state is achieved at around time intervals 2:20 – 2:30. These absorption and recovery characteristics can also be

identified in the simulation results of the other performance indicators that have been discussed in section 8.2.1. For this reason the resilience characteristics and patterns as observed in the *time in approach* performance indicator will be used in the two sections below.

Generation rate (throughput capacity)

The rate at which aircraft are delivered to the approach sectors should be in balance with the current runway capacity. This may sound very self-evident, but is also found to be an absolute requirement to enable stable approach operations over a longer period of time with relatively large traffic volumes. Let's therefore introduce the symbol Δ as the relative difference between the generation rate (i.e. throughput capacity) as applied by the feeder controller and the merging rate (i.e. runway capacity) as applied by the ARR controller. A negative Δ represents herein the situation where the applied generation rate is on average lower than the maintained merging rate. Conducted experiments have shown that a positive Δ will result over time in a build-up of approaching traffic, leading to unstable approach operations. Therefore, in order to guarantee stable approach operations the condition $\Delta \leq 0$ should on average apply.

The obtained simulation results indicate that the resilient capacities of approach operations during disturbance are defined by the specific value of Δ. The size of Δ defines for each performance indicator the time period that is required to return to stable operations again. The effects of different values for Δ on the recovery phase of approach operations (in terms of the *time in approach* performance indicator) are visualized in figure 43. The graphs in this figure show how the duration of the recovery phase and the achieved recovery rate benefit from the situation when the feeder controller is maintaining a lower generation rate in comparison with the achieved runway capacity. Note that this specific factor does only affect the recovery rate of a performance indicator and not the absorption phase and maximum performance loss.

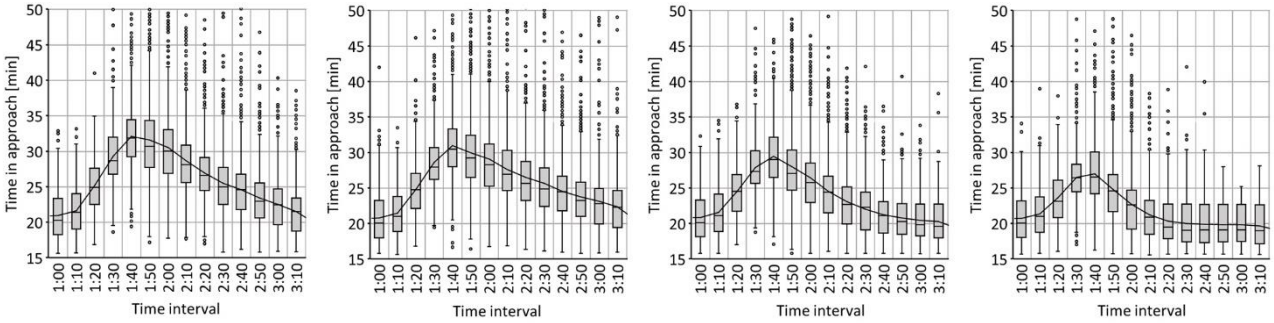


(a) Δ = 0 ac·h^{-1} *(b)* Δ = -2 ac·h^{-1} *(c)* Δ = -4 ac·h^{-1} *(d)* Δ = -6 ac·h^{-1}

Figure 43 – Effects of relative difference between maintained throughput capacity (i.e. generation rate) and achieved runway capacity on recovery phase

Merging rate and achieved runway capacity

The obtained simulation results show that the applied generation rate does not affect the course of the absorption phase and therefore also not the maximum performance loss. It is found that the absorption phase and the maximum performance loss are defined by the reduced rate at which the two incoming streams of traffic will be merged relative to the IF when the ARR controller is informed about a reduced runway capacity. In other words the absorption phase and the maximum performance loss are defined by the specific reduction in the to be achieved runway capacity during disturbance. The effects of different merging rates on the absorption phase and the maximum performance loss are visualized in figure 44. Each graph in this figure corresponds to a different merging rate as applied by the ARR controller in *reduced capacity mode*. This merging rate is modelled using *time buffers* that define the *desired spacing* between α_i and $\alpha_{2,i}$ (section 5.3.3.7). As can be seen in figure 44 the different *time buffer* values do not affect the duration of the absorption phase, but do affect the amount of performance loss. This beneficial resilient property is the result of maintaining a larger merging rate. A larger merging rate results in shorter (and less) *vector* and *holding* operations and therefore more resilient performance. This enhanced resilient performance can also be expressed in terms of a faster recovery time (figure 44). A higher merging rate does however also result in a higher achieved runway capacity, which may not always be desired. The to be maintained merging rate is therefore dependent on the maximum runway capacity that can be achieved during the bad weather disturbance taking into account the reduced runway conditions.



(a) time buffer = 75 s *(b) time buffer* = 65 s *(c) time buffer* = 55 s *(d) time buffer* = 45 s

Figure 44 – Effects of different merging rates (time buffers) on absorption phase and maximum performance loss

8.2.4 Comparison between qualitative and quantitative simulation results

Figure 45 below visualizes the mental simulation results that have been obtained by Stroeve and Everdij in their qualitative agent-based resilience study [43]. These mental simulation results are obtained by reasoning in a qualitative manner how relevant states and indicators change over time as the result of a bad weather disturbance. This section describes how these qualitative results of Stroeve and Everdij relate to the simulation results that are obtained in this quantitative study. The comparison between both simulation results is presented in table 9.

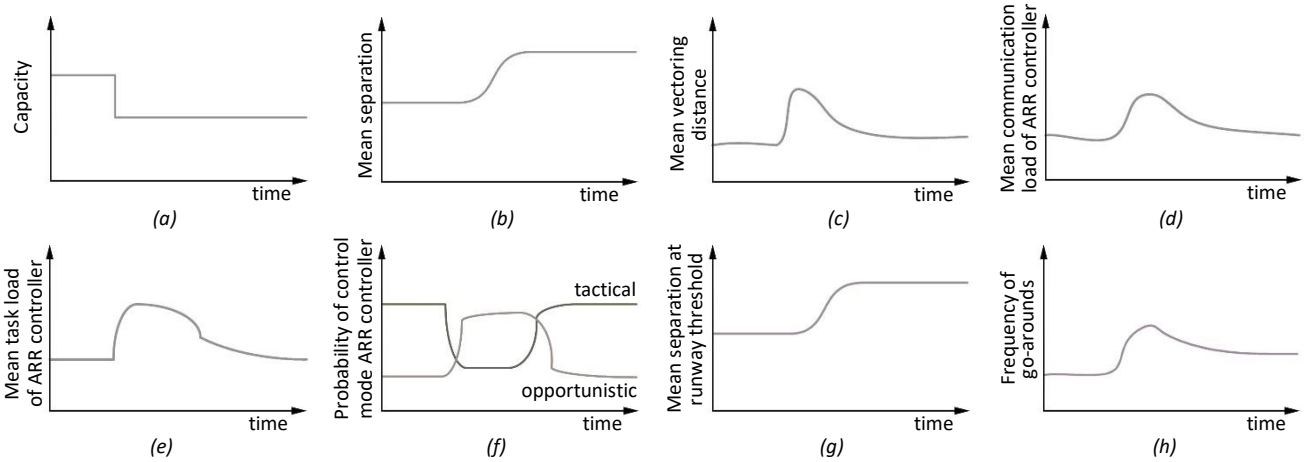


Figure 45 – Mental simulation results of indicators that describe conventional approach operations during a sudden bad weather condition at the airport [43]

Performance indicator	Differences between qualitative and quantitative simulation results
Capacity	The quantitative study does also consider an instantaneous reduction in runway capacity as the result of a bad weather disturbance. All conducted experiments considered a fixed reduced runway capacity of 21 ac·h^{-1}, while the initial runway capacity varies between 28 and 36 ac·h^{-1}. The qualitative study of Stroeve and Everdij does however not consider specific values for the runway capacity before and during disturbance.
Mean separation	The mean separation is not measured as a function of time in the quantitative study, because of the wide range in *actual spacings* at each time point as the result of the many aircraft in approach, the results of compression, different flight performances, different *separation minima*, and the random generations at each entry point. The *mean separation* performance indicator would therefore not deliver any useful results after the large number of simulations. The *actual spacing* between aircraft will however gradually increase during disturbance as the result of the increased *desired spacings*. The separation increase is completed once the measured performance indicators indicate a stable recovered state.

Mean vectoring distance	This plot can be compared with the obtained simulation results of the *time flying vector* STAR (figure 28), *time flying vector IF* (figure 34) and *time flying vector merge* (figure 32) performance indicators. The obtained quantitative simulation results indicate that aircraft are only temporarily operating these type of vectors just after disturbance, which is in contrast with the qualitative results. The quantitative results do however show the same fast increase in vector operations just after the start of the disturbance.
Mean communication load of ARR controller	This plot can be compared with the obtained simulation results of the *number of instructions* performance indicator (figure 40). Both the qualitative and quantitative simulation results share a similar type and rate of increase and decrease in the communication load. The quantitative simulation results indicate a lower communication load after disturbance in comparison with the communication load before disturbance as the result of a lower maintained throughput capacity and a lower traffic density. This is in contrast with the qualitative simulation results, which indicate that the recovered communication load is just as intense as the communication load before disturbance.
Mean task load of ARR controller	Same explanation as used for the *mean communication load of ARR controller* performance indicator.
Probability of control mode ARR controller	This plot can be compared with the obtained simulation results of the *percentage in tactical mode* performance indicator (figure 41). Both the qualitative and quantitative simulation results indicate that the ARR controller will rapidly switch to an opportunistic control mode just after the start of the disturbance. Both simulation results differ however in the way in which the ARR controller switches back again to the tactical control mode. The quantitative simulation results show that the ARR controller is gradually switching to a tactical control mode again as the result of a gradually decreasing workload. The gradually decreasing workload can be explained by an increasing number of aircraft in the ARR sector that have resumed their standard approach operations. The qualitative simulation results indicate however that during disturbance the ARR controller remains operating in the opportunistic control mode for a specific period of time, after which the ARR controller is quickly switching back to a tactical control mode again. The qualitative simulation results do therefore not really take into account a kind of transition phase between the opportunistic and tactical control mode.
Mean separation at runway threshold	This plot can be compared with the obtained simulation results of the *time between landing* performance indicator (figure 24). The obtained quantitative simulation results do however not really indicate the similar smooth increase as can be seen in the qualitative results, which can be related to the initiation of go-arounds.
Frequency of go-arounds	This plot can be compared with the obtained simulation results of the *number of go-arounds* performance indicator (figure 25). The obtained quantitative simulation results indicate however that *go-arounds* are on average only initiated just after disturbance, which is in contrast with the qualitative simulation results.

Table 9 – Comparison between qualitative and quantitative simulation results

8.3 Experiments

The previous section highlighted the general characteristics and patterns that are observed in the obtained simulation results. The contents of this section extend this general analysis by describing the simulation results of the conducted parameter variation experiments. These experiments have been performed to study the effects of specific parameters on the resilient capacities of disturbed approach operations. The specific selection of these parameters is established by reasoning about their possible effects on disturbed approach operations, their relation to the field of resilience and their relation to the emergent phenomena of interest. The size of the scenario space is chosen such that a proper trade-off is made between the level of variation that can be examined and the number of required experiments that has to be performed. The specific parameters that have been used in the parameter variation experiments are listed in table 10. These selected parameters are specified in appendix C. See the corresponding sections for more specific information about each parameter.

Parameter	Appendix	Defines:
C_{I}	C.5.2	the generation rate (throughput capacity) of the feeder controller in *normal capacity mode*
$t_{\mathrm{B},C_{\mathrm{D}}}^{a_{1,i},c_1}$	C.4.1	the *time buffer* that is maintained by the TNW controller in *reduced capacity mode*
$t_{\mathrm{B},C_{\mathrm{D}}}^{a_{1,i},c_2}$	C.4.1	the *time buffer* that is maintained by the TNE controller in *reduced capacity mode*
$\sigma_{s_{\mathrm{B}},\mathrm{L,O}}$	C.4.3.1	the size of the *separation buffer (type II)* (large vector distance/opportunistic control mode)
$\sigma_{s_{\mathrm{B}},\mathrm{L,T}}$	C.4.3.1	the size of the *separation buffer (type II)* (large vector distance/tactical control mode)
$\sigma_{s_{\mathrm{B}},\mathrm{S,O}}$	C.4.3.1	the size of the *separation buffer (type II)* (small vector distance/opportunistic control mode)
$\sigma_{s_{\mathrm{B}},\mathrm{S,T}}$	C.4.3.1	the size of the *separation buffer (type II)* (small vector distance/tactical control mode)

Table 10 – Parameters that are used in the parameter variation experiments

The set of parameters in table 10 allows to examine the effects of the following factors on the resilient capacities of approach operations:

- the size of the total throughput capacity in the approach sector, using different parameter settings of C_{I};
- the coordination between controllers about the maintained throughput capacity, using different parameter settings of $t_{\mathrm{B},C_{\mathrm{D}}}^{a_{1,i},c_1}$ and $t_{\mathrm{B},C_{\mathrm{D}}}^{a_{1,i},c_2}$;
- the skills of the controller in maintaining and adjusting the throughput capacity, using different parameter settings of $\sigma_{s_{\mathrm{B}},\mathrm{L,O}}$, $\sigma_{s_{\mathrm{B}},\mathrm{L,T}}$, $\sigma_{s_{\mathrm{B}},\mathrm{S,O}}$, $\sigma_{s_{\mathrm{B}},\mathrm{S,T}}$.

Each of these three factors relates to a specific conducted parameter variation experiment. The three conducted parameter variation experiments will be described in more detail in the subsections below. Each experiment is discussed by using the following structure: research question, hypothesis, parameter settings, results, analysis. As described in section 8.1 the obtained simulation results are expressed in trajectory plots and tables, where each table contains the median ($\tilde{x}$) and standard deviation (σ) values of the measured performance indicators for a specific time interval before disturbance and during disturbance. Note however that the measured and presented median and standard deviation values relate to the data points out of the 95% confidence intervals only, while the obtained trajectory plots contain all logged data points (section 8.1). One should be aware of this difference, otherwise it may lead to some confusion when analysing the simulation results. Appendix B provides the more detailed simulation results of each conducted parameter variation experiment.

8.3.1 Experiment 1

Research question

To what extent does the maintained (initial) throughput capacity contribute to the resilient capacities of approach operations during a sudden and unexpected bad weather disturbance?

Hypothesis

The relative size of the reduction in runway capacity is expected to define the type of operations that become required in order to cope with such capacity loss. A relatively large reduction in runway capacity is more likely to cause the initiation of *holding* operations, *go-arounds*, and *vector merge* operations. A relatively small reduction in runway capacity on the other hand is expected to be solved using *vector STAR* operations only. The difference between the initial throughput capacity and the reduced runway capacity will define the extra time period that each aircraft needs to spend during its approach operations.

Parameter settings

Experiment 1 considers three different values for the generation rate of the feeder controller in *normal capacity mode*. All remaining parameters are kept fixed for each scenario in experiment 1. Table 11 lists the parameter settings for the three simulated scenarios in experiment 1. The maximum generation rate (throughput capacity) that is simulated equals 36 aircraft per hour. This maximum is established since it is found empirically that a higher throughput capacity will lead to *saturation* of the trombone segments already before disturbance. All three simulated scenarios consider a constant reduced generation rate of 21 ac·h^{-1}, as maintained by the feeder controller in *reduced capacity mode*.

Parameter	Scenario 1	Scenario 2	Scenario 3
C_{I}	36 ac·h^{-1}	32 ac·h^{-1}	28 ac·h^{-1}
$t_{\mathrm{B},C_{\mathrm{D}}}^{a_{1,i},c_1}$	20 s	20 s	20 s
$t_{\mathrm{B},C_{\mathrm{D}}}^{a_{1,i},c_2}$	20 s	20 s	20 s
$\sigma_{s_{\mathrm{B}},\mathrm{L,O}}$	0.15 NM	0.15 NM	0.15 NM
$\sigma_{s_{\mathrm{B}},\mathrm{L,T}}$	0.10 NM	0.10 NM	0.10 NM
$\sigma_{s_{\mathrm{B}},\mathrm{S,O}}$	0.10 NM	0.10 NM	0.10 NM
$\sigma_{s_{\mathrm{B}},\mathrm{S,T}}$	0.05 NM	0.05 NM	0.05 NM

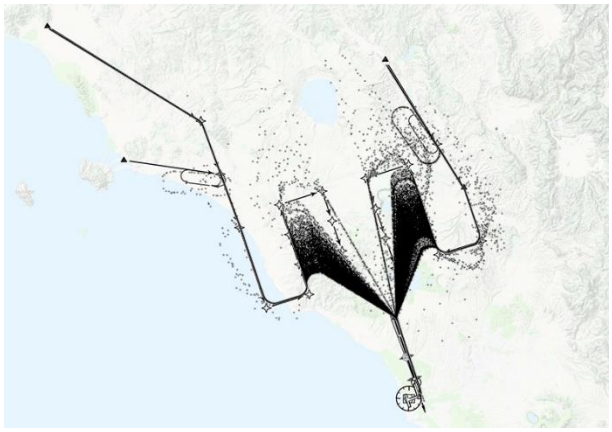
Table 11 – Scenario space for experiment 1

Results

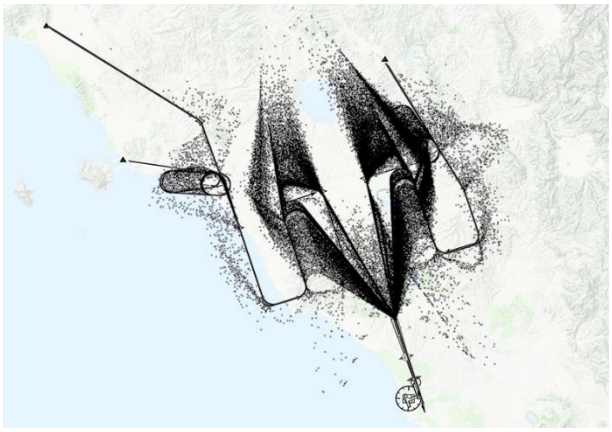
Table 12 provides the simulation results that correspond to experiment 1. The dynamics in these simulation results are visualized by the boxplots and histograms in appendix B.1. The trajectory plots that correspond to experiment 1 can be seen in figure 46.

	Before disturbance						**During disturbance**					
	Scenario 1		**Scenario 2**		**Scenario 3**		**Scenario 1**		**Scenario 2**		**Scenario 3**	
	$\tilde{x}$	σ	$\tilde{x}$	σ	$\tilde{x}$	σ	$\tilde{x}$	σ	$\tilde{x}$	σ	$\tilde{x}$	σ
Time between landing [min]	1.4	0.5	1.6	0.7	1.8	0.9	2.5	0.5	2.6	0.6	2.5	0.5
Number of aircraft in approach	12	0.8	11	0.8	9	0.8	10	1.2	9	1.0	8	0.8
Number of landings	6	0.8	5	0.9	5	0.9	4	0.5	4	0.5	4	0.5
Percentage STAR	71	4	72	5	74	5	40	15	55	12	65	7
Percentage vector STAR	0	0	0	0	0	0	**6**	**3**	**3**	**3**	**0**	**2**
Percentage vector merge	0	0	0	0	0	0	**24**	**8**	**14**	**7**	**4**	**7**
Percentage vector IF	29	4	28	5	26	5	19	3	23	4	25	5
Percentage holding	0	0	0	0	0	0	**20**	**13**	**0**	**9**	**0**	**1**
Time in approach [min]	19.7	2.8	19.1	2.6	18.0	2.8	**32.1**	**4.4**	**28.2**	**2.0**	**24.8**	**3.1**
Time flying STAR [min]	9.9	2.5	9.6	2.4	9.1	2.5	12.7	2.7	12.0	2.8	11.9	2.5
Time flying vector STAR [min]	0.0	0.0	0.0	0.0	0.0	0.0	**1.6**	**1.1**	**0.0**	**0.8**	**0.0**	**0.4**
Time flying vector merge [min]	0.0	0.0	0.0	0.0	0.0	0.0	**6.7**	**3.5**	**3.5**	**2.5**	**0.0**	**1.8**
Time flying vector IF [min]	5.3	0.7	5.0	0.5	4.7	0.5	6.5	0.8	6.5	0.5	6.5	0.7
Time flying holding [min]	0.0	0.0	0.0	0.0	0.0	0.0	**2.4**	**5.0**	**0.0**	**2.8**	**0.0**	**0.1**
Percentage in tactical mode - TNW	100	0	100	0	100	0	100	4	100	0	100	0
Percentage in tactical mode - TNE	86	39	100	32	100	12	96	39	100	26	100	0
Percentage in tactical mode - ARR	100	0	100	0	100	0	**0**	**25**	**30**	**38**	**100**	**34**
Percentage in tactical mode - TWR	100	0	100	0	100	0	100	0	100	0	100	0
Number of instructions - TNW	8	3.1	8	3.1	7	2.7	5	2.7	5	2.6	5	2.5
Number of instructions - TNE	16	3.7	14	3.7	12	3.3	8	3.0	9	2.9	9	2.8
Number of instructions - ARR	12	1.1	11	1.3	9	1.2	**24**	**4.4**	**20**	**4.4**	**11**	**4.5**
Number of instructions - TWR	12	1.4	11	1.5	10	1.8	7	0.8	7	0.9	7	1.1
Runway throughput	35.4	0.9	31.7	0.7	27.9	0.7	22.3	0.6	22.0	0.6	21.6	0.5
Number of go-arounds	0	0.5	0	0.4	0	0.3	**2**	**0.7**	**1**	**0.8**	**1**	**0.7**
Number of vector inbound trombone	0	0.4	0	0.3	0	0.1	**0**	**1.2**	**0**	**0.9**	**0**	**0.4**
Number of vector outbound trombone	0	0.2	0	0.1	0	0.0	**0**	**0.5**	**0**	**0.3**	**0**	**0.2**

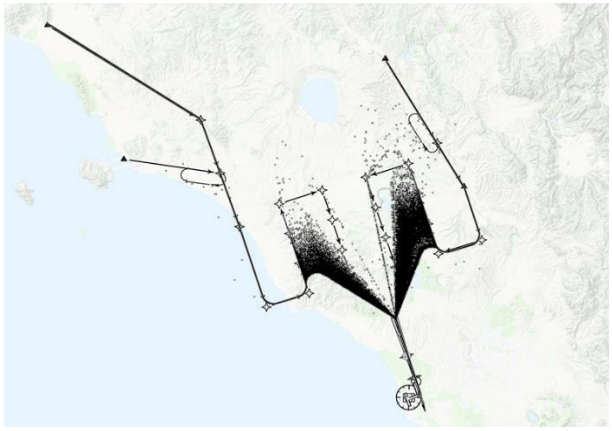
Table 12 – Simulation results of experiment 1



(a) Scenario 1 before disturbance

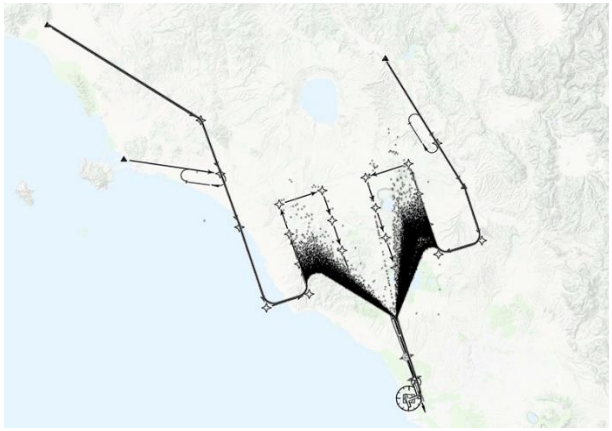


(b) Scenario 1 during disturbance

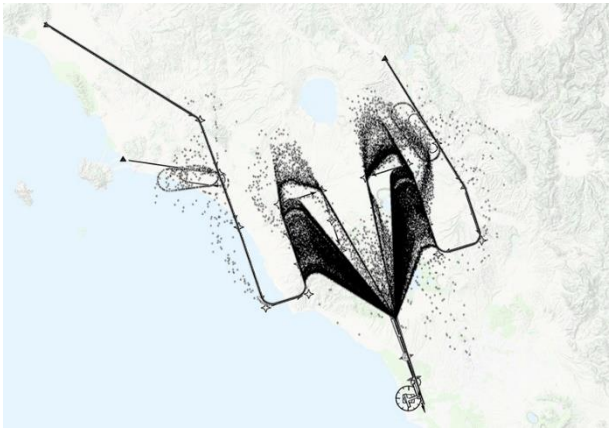


(c) Scenario 2 before disturbance

(d) Scenario 2 during disturbance



(e) Scenario 3 before disturbance



(f) Scenario 3 during disturbance

Figure 46 – Trajectory plots visualizing the emergent patterns as the result of the simulated scenarios in experiment 1

Analysis

The following comments apply when analysing the results of experiment 1:

- A lower overall throughput capacity results in less risk to saturate the trombone segments of the STAR procedures, which can clearly be seen in the trajectory plots in figure 46. The *saturation* of these trombone segments would result in the initiation of *vector merge* and *holding* operations, which in turn reduces the stability of the approach operations. The results in table 12 show that scenario 1 required more *vector* and *holding* operations when compared to scenario 3, which required almost no additional actions.
- A lower overall throughput capacity, and with that less *vector* and *holding* operations, does positively affect the durations of the total arrival/approach operations during disturbance. The *time in approach* performance indicator shows that arrival/approach operations during disturbance take on average (maximal) 7 minutes shorter in scenario 3 when compared to scenario 1.
- A lower throughput capacity has a significant and beneficial impact on the workload of the ARR controller when confronted with a sudden and unexpected capacity loss. The maximum *number of instructions* that have been provided by the ARR controller in a specific time interval during disturbance equals 24 in scenario 1, against only 11 instructions in scenario 3.
- *Vector trombone* operations are on average flown (slightly) more in scenario 1 when compared to the other scenarios as the result of relatively higher traffic densities near the IF.

8.3.2 Experiment 2

Research question

To what extent do the vector (in-)accuracies of controllers contribute to the resilient capacities of approach operations during a sudden and unexpected bad weather disturbance?

Hypothesis

Vector accuracy is considered an important property to enlarge the separation distances in a structured and fluent way, and such that the desired throughput capacity can be properly approached. Large deviations in vector accuracy are likely to cause separation distances that are eventually assessed as too large or too short. Relatively large separation distances will herein further build up the vectored traffic situation and negatively affect the desired throughput capacity. Relatively small separation distances on the other hand increase the risk of conflicts in the near future, causing a second (or more) vector instruction to become required. These vectoring operations as the result of too short vectoring practices can eventually affect multiple aircraft in a row, each having to adjust their separation distance with the preceding aircraft as the result of a conflict somewhere in the beginning. An improper vectoring strategy will therefore also affect controller workload in a negative manner. For this reason vectoring accuracy is considered an interesting parameter that can be used to examine the balance between controller workload and throughput capacity.

Parameter settings

Experiment 2 considers three sets of vector accuracies that are used to simulate the vectoring practices of controllers. All remaining parameters are kept fixed for each scenario in experiment 2. Table 13 lists the parameter settings that are used in experiment 2. Scenario 3 describes the worst vectoring accuracy when compared to scenarios 1 and 2. It is found empirically that a larger variation in vector accuracy will lead to a significant and unrealistic increase in required vector instructions, causing a number of performance indicators to explode.

Parameter	Scenario 1	Scenario 2	Scenario 3
C_{I}	34 ac·h^{-1}	34 ac·h^{-1}	34 ac·h^{-1}
$t_{\mathrm{B},C_{\mathrm{D}}}^{a_{1,i},c_1}$	20 s	20 s	20 s
$t_{\mathrm{B},C_{\mathrm{D}}}^{a_{1,i},c_2}$	20 s	20 s	20 s
$\sigma_{s_{\mathrm{B}},\mathrm{L},\mathrm{O}}$	0.25 NM	0.75 NM	1.25 NM
$\sigma_{s_{\mathrm{B}},\mathrm{L},\mathrm{T}}$	0.10 NM	0.60 NM	1.10 NM
$\sigma_{s_{\mathrm{B}},\mathrm{S},\mathrm{O}}$	0.15 NM	0.65 NM	1.15 NM
$\sigma_{s_{\mathrm{B}},\mathrm{S},\mathrm{T}}$	0.05 NM	0.50 NM	1.00 NM

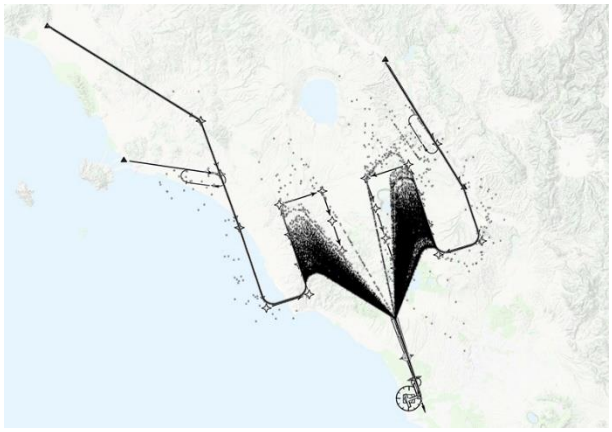
Table 13 – Scenario space for experiment 2

Results

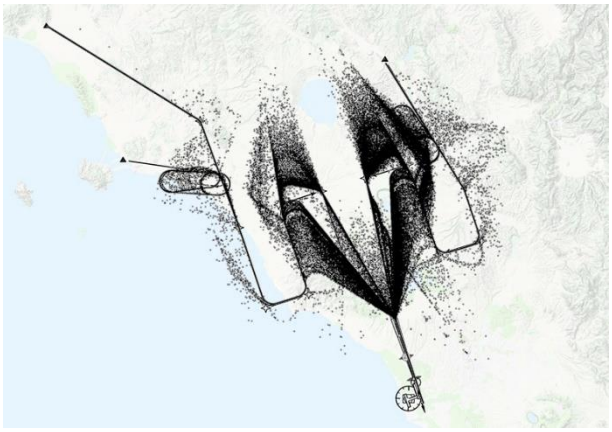
Table 14 provides the simulation results that correspond to experiment 2. The dynamics of some of these simulation results are visualized by boxplots and histograms in appendix B.2. The trajectory plots that correspond to experiment 2 can be seen in figure 47.

	Before disturbance						**During disturbance**					
	Scenario 1		**Scenario 2**		**Scenario 3**		**Scenario 1**		**Scenario 2**		**Scenario 3**	
	$\tilde{x}$	σ	$\tilde{x}$	σ	$\tilde{x}$	σ	$\tilde{x}$	σ	$\tilde{x}$	σ	$\tilde{x}$	σ
Time between landing [min]	1.5	0.5	1.6	0.6	1.6	0.6	2.6	0.6	2.5	0.6	2.5	0.5
Number of aircraft in approach	11	0.8	11	0.9	12	0.9	9	1.1	9	0.3	9	1.5
Number of landings	6	0.8	6	0.8	5	0.8	4	0.5	4	0.5	4	0.5
Percentage flying STAR	72	4	72	4	71	4	48	13	52	14	51	13
Percentage flying vector STAR	0	0	0	0	0	0	5	3	5	3	5	3
Percentage flying vector merge	0	0	0	0	0	0	18	8	18	8	18	9
Percentage flying vector IF	28	4	28	4	29	4	20	3	22	3	22	4
Percentage flying holding	0	0	0	0	0	0	6	11	3	12	4	13
Time in approach [min]	**19.2**	**2.8**	**19.4**	**2.8**	**20.2**	**2.9**	**29.9**	**3.2**	**30.2**	**4.5**	**30.3**	**5.3**
Time flying STAR [min]	9.7	2.5	9.7	2.5	10.0	2.5	12.1	2.6	12.1	2.7	12.5	2.8
Time flying vector STAR [min]	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.5	0.0	1.4	0.0	1.2
Time flying vector merge [min]	0.0	0.0	0.0	0.0	0.0	0.0	5.0	2.7	4.7	2.9	4.5	2.8
Time flying vector IF [min]	**5.0**	**0.6**	**5.2**	**0.7**	**5.4**	**0.8**	6.5	0.8	6.5	0.8	6.6	1.2
Time flying holding [min]	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.6	0.0	3.8	0.0	4.3
Percentage in tactical mode - TNW	100	0	100	0	100	0	100	1	100	0	100	4
Percentage in tactical mode - TNE	100	32	100	35	100	34	100	33	100	34	100	33
Percentage in tactical mode - ARR	100	0	100	13	100	39	8	33	0	29	0	16
Percentage in tactical mode - TWR	100	0	100	0	100	0	100	0	100	0	100	0
Number of instructions - TNW	8	2.9	8	3.3	8	3.0	5	2.7	5	2.7	5	2.7
Number of instructions - TNE	15	3.5	15	3.6	15	3.9	9	3.0	9	3.1	9	3.2
Number of instructions - ARR	**11**	**1.2**	**12**	**2.5**	**15**	**4.5**	**22**	**3.9**	**24**	**5.3**	**28**	**5.8**
Number of instructions - TWR	11	1.5	11	1.6	11	1.5	7	0.8	7	0.9	7	0.9
Runway throughput	**33.5**	**0.7**	**33.1**	**0.9**	**32.8**	**1.1**	22.2	0.6	22.2	0.7	22.3	0.7
Number of go-arounds	**0**	**0.4**	**1**	**0.8**	**1**	**1.0**	1	0.8	1	0.8	1	0.9
Number of vector inbound trombone	**0**	**0.3**	**0**	**0.6**	**0**	**0.9**	**0**	**1.0**	**1**	**1.2**	**1**	**1.7**
Number of vector outbound trombone	**0**	**0.1**	**0**	**0.7**	**0**	**0.9**	0	0.4	0	0.4	0	0.6

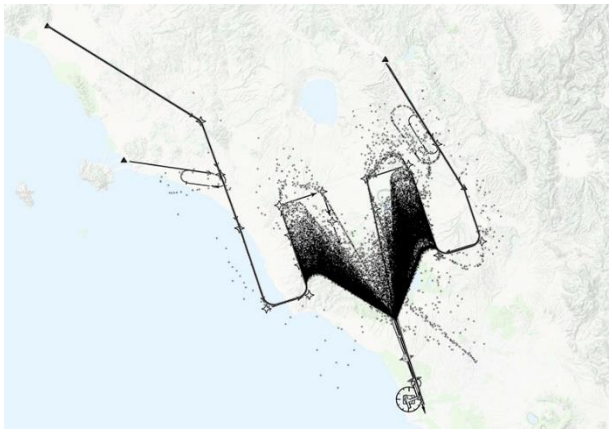
Table 14 – Simulation results of experiment 2



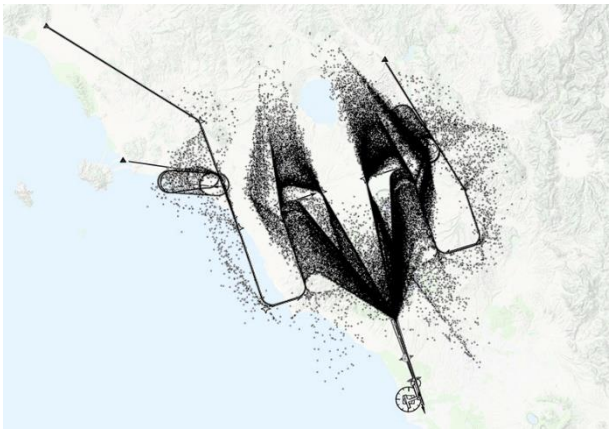
(a) Scenario 1 before disturbance



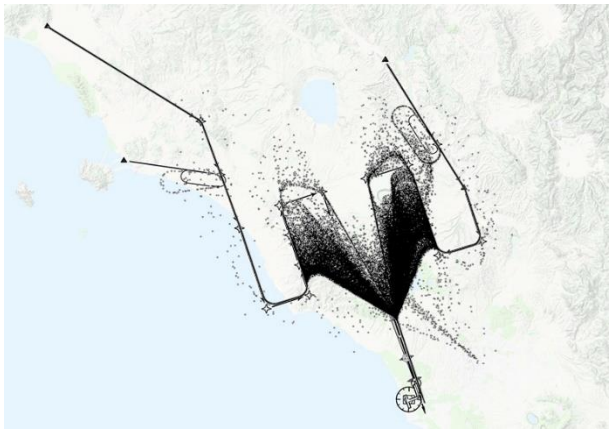
(b) Scenario 1 during disturbance



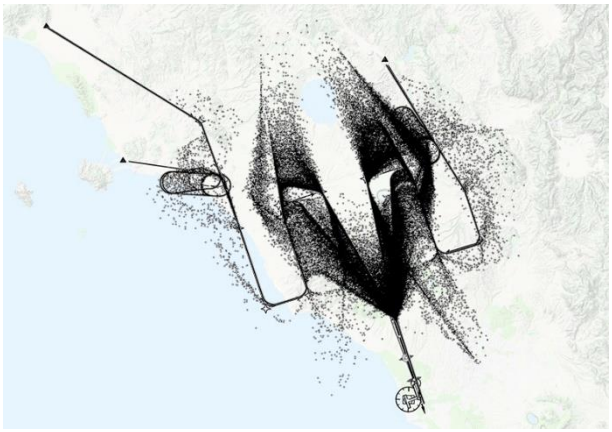
(c) Scenario 2 before disturbance



(d) Scenario 2 during disturbance



(e) Scenario 3 before disturbance



(f) Scenario 3 during disturbance

Figure 47 – Trajectory plots visualizing the emergent patterns as the result of the simulated scenarios in experiment 2

Analysis

The following comments apply when analysing the results of experiment 2:

- The effects of vector (in-)accuracies on the flown trajectories can already clearly be seen in the trajectory plots in figure 47. When observing the densities of the logged coordinates near the intermediate fix (before disturbance) one can for instance see that aircraft in scenario 3 required additional vectors when compared to scenario 1.
- The vector (in-)accuracies have no effect on the operational state of aircraft, i.e. the type of operations that are (and should be) flown.
- Larger vector inaccuracies result in more go-arounds due to larger variability in separation at the IF, which can also be seen in the achieved runway capacity (before disturbance).
- The merging practice of the ARR controller is negatively affected as the result of larger vector inaccuracies. This means that conflicts are more likely to occur near the IF. This increased risk in conflicts is reflected in the simulation results by a larger number of *vector trombone* operations in scenario 3 when compared to the ones that have occurred in scenario 1.
- The workload of the ARR controller increases as the result of the inaccurate vectoring practice. This increased workload is expressed by a larger overall number of instructions (i.e. 6 instructions per time interval), more dispersion in the measured data points, and a relatively larger operation in the opportunistic control mode.
- Larger vector inaccuracies have only a slightly negative effect on runway throughput capacity, causing the duration of the total arrival/approach operations to increase a bit.

8.3.3 Experiment 3

Research question

To what extent does coordination between the approach controllers contribute to the resilient capacities of approach operations during a sudden and unexpected bad weather disturbance?

Hypothesis

A sudden and unexpected reduction in runway capacity asks for an increase in separation to reduce the throughput capacity accordingly. Coordination with respect to this research question is related to the ability of each approach controller to reduce the throughput capacity in their respective airspace sectors such that the overall approach operations will benefit from the combined actions of each controller. More specifically, the considered coordination mechanism can be defined as the ability of the TNW/TNE controllers to anticipate on the fact that the workload of the ARR controller will be affected most significantly during disturbance. In practical terms this means that the TNW/TNE controllers aim to deliver aircraft to the ARR sector according a lower maintained throughput capacity such that the ARR controller is given the opportunity to adapt the traffic situation in its approach sector with a relatively lower increased workload. Such cooperative setting is expected to spread the increased workload as the result of the sudden reduction in throughput capacity over the multiple approach controllers. Secondly, it is expected that the risk of *saturation* in the ARR sector will be positively affected as the result of a lower maintained throughput capacity in the TNW/TNE sectors. A lower maintained throughput capacity in the TNW/TNE sectors is in turn however also expected to increase the number of necessary *vector* and *holding* operations.

Parameter settings

Experiment 3 considers three sets of *time buffers* that are used by the TNW/TNE controllers to steer upon a desired and reduced throughput capacity. All remaining parameters are kept fixed for each scenario in experiment 3. Table 15 lists the parameter settings that are used in experiment 3. The specific values of the used *time buffers* in the scenario space are chosen such that they exceed the *time buffers* that are applied by the feeder controller. Scenario 3 considers the maximum *time buffer* that has been simulated. It is found empirically that larger *time buffers* will result in unrealistically large and long vector operations due to the large maintained *desired spacings*. Note again that the *time buffers* define the size of the to be added *separation buffer (type I)* (section 5.3.3.3).

Parameter	Scenario 1	Scenario 2	Scenario 3
C_{I}	36 ac·h^{-1}	36 ac·h^{-1}	36 ac·h^{-1}
$t_{\mathrm{B},C_{\mathrm{D}}}^{a_{1,i},c_1}$	70 s	100 s	130 s
$t_{\mathrm{B},C_{\mathrm{D}}}^{a_{1,i},c_2}$	70 s	100 s	130 s
$\sigma_{s_{\mathrm{B}},\mathrm{L},\mathrm{O}}$	0.25 NM	0.25 NM	0.25 NM
$\sigma_{s_{\mathrm{B}},\mathrm{L},\mathrm{T}}$	0.10 NM	0.10 NM	0.10 NM
$\sigma_{s_{\mathrm{B}},\mathrm{S},\mathrm{O}}$	0.15 NM	0.15 NM	0.15 NM
$\sigma_{s_{\mathrm{B}},\mathrm{S},\mathrm{T}}$	0.05 NM	0.05 NM	0.05 NM

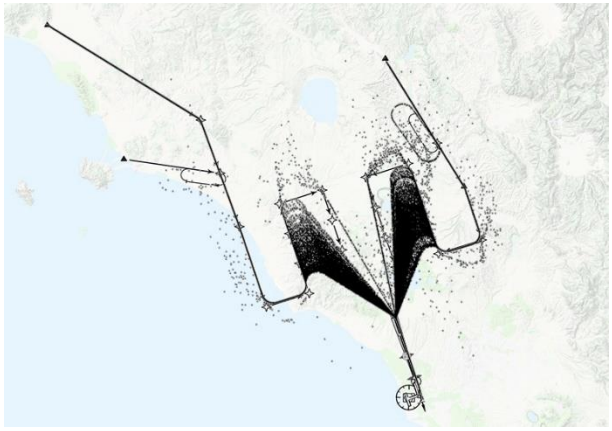
Table 15 – Scenario space for experiment 3

Results

Table 16 provides the simulation results that correspond to experiment 3. The dynamics in some of these simulation results are visualized by boxplots and histograms in appendix B.3. The trajectory plots that correspond to experiment 3 can be seen in figure 48.

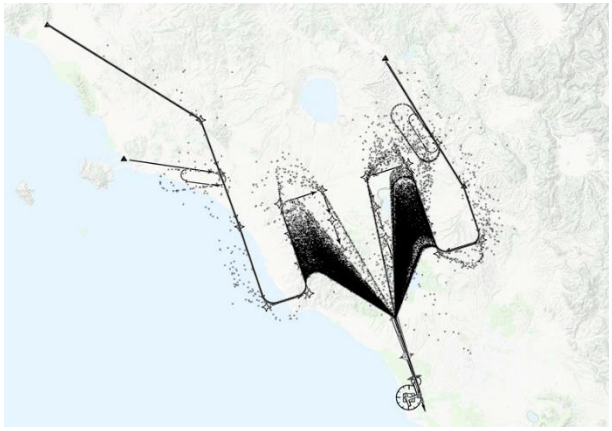
	Before disturbance						**During disturbance**					
	Scenario 1		**Scenario 2**		**Scenario 3**		**Scenario 1**		**Scenario 2**		**Scenario 3**	
	$\tilde{x}$	σ	$\tilde{x}$	σ	$\tilde{x}$	σ	$\tilde{x}$	σ	$\tilde{x}$	σ	$\tilde{x}$	σ
Time between landing [min]	1.5	0.5	1.5	0.5	1.5	0.5	2.4	0.5	2.5	0.5	2.5	0.5
Number of aircraft in approach	12	0.9	12	0.8	12	0.8	10	1.2	10	1.3	10	1.3
Number of landings	6	0.8	6	0.8	6	0.8	4	0.4	4	0.5	4	0.5
Percentage flying STAR	71	4	71	4	72	4	35	14	36	15	37	12
Percentage flying vector STAR	0	0	0	0	0	1	**17**	**10**	**25**	**12**	**32**	**9**
Percentage flying vector merge	0	0	0	0	0	0	**21**	**8**	**20**	**8**	**17**	**7**
Percentage flying vector IF	29	4	29	4	28	4	20	3	20	4	20	3
Percentage flying holding	0	0	0	0	0	0	**20**	**14**	**12**	**14**	**1**	**12**
Time in approach [min]	19.9	2.8	19.9	2.8	19.8	2.8	31.9	3.3	32.0	3.9	32.2	3.2
Time flying STAR [min]	10.0	2.5	10.0	2.5	10.0	2.5	12.3	2.7	11.9	3.3	10.6	4.0
Time flying vector STAR [min]	0.0	0.0	0.0	0.0	0.0	0.0	**1.8**	**3.3**	**5.5**	**4.7**	**7.0**	**5.0**
Time flying vector merge [min]	0.0	0.0	0.0	0.0	0.0	0.0	**6.2**	**2.4**	**5.3**	**2.7**	**4.5**	**2.1**
Time flying vector IF [min]	5.3	0.7	5.3	0.7	5.3	0.7	6.6	0.9	6.5	1.0	6.5	0.8
Time flying holding [min]	0.0	0.0	0.0	0.0	0.0	0.0	**3.0**	**5.4**	**0.0**	**5.4**	**0.0**	**3.8**
Percentage in tactical mode - TNW	100	0	100	0	100	0	100	18	100	19	100	14
Percentage in tactical mode - TNE	82	39	80	38	93	41	28	39	32	41	40	42
Percentage in tactical mode - ARR	100	0	100	0	100	0	0	24	0	28	0	33
Percentage in tactical mode - TWR	100	0	100	0	100	0	100	0	100	0	100	0
Number of instructions - TNW	8	3.0	8	3.0	8	3.1	8	6.5	10	6.0	10	5.9
Number of instructions - TNE	16	3.8	16	3.7	16	3.8	20	7.8	20	6.7	20	6.2
Number of instructions - ARR	12	1.2	12	1.2	12	1.2	**25**	**4.9**	**23**	**4.4**	**21**	**4.6**
Number of instructions - TWR	12	1.5	12	1.4	12	1.4	7	0.9	7	0.9	7	0.9
Runway throughput	35.1	1.0	35.1	0.9	35.4	1.0	22.4	0.7	22.4	0.6	22.1	0.6
Number of go-arounds	0	0.6	0	0.5	0	0.5	1	0.7	2	0.7	2	0.7
Number of vector inbound trombone	0	0.4	0	0.3	0	0.4	0	1.1	0	1.0	0	1.1
Number of vector outbound trombone	0	0.3	0	0.3	0	0.3	0	0.5	0	0.5	0	0.4

Table 16 – Simulation results of experiment 3

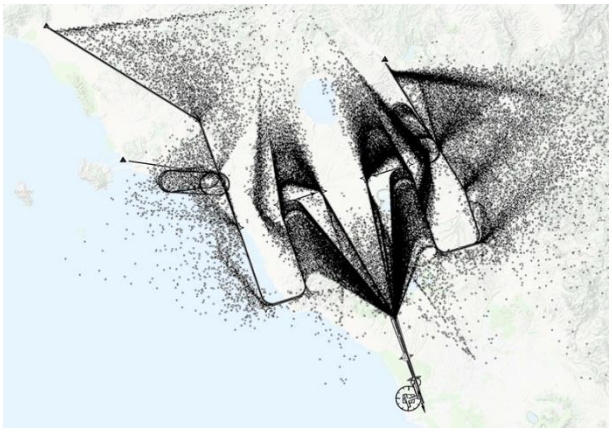


(a) Scenario 1 before disturbance

(b) Scenario 1 during disturbance



(c) Scenario 2 before disturbance



(d) Scenario 2 during disturbance

(e) Scenario 3 before disturbance

(f) Scenario 3 during disturbance

Figure 48 – Trajectory plots visualizing the emergent patterns as the result of the simulated scenarios in experiment 3

Analysis

The following comments apply when analysing the results of experiment 3 (and scenario 1 of experiment 1):

- The simulated coordination mechanism does not affect the total duration of the arrival/approach operations, resulting in a constant runway capacity for each simulated scenario.
- The application of larger *time buffers* by the TNW/TNE controllers during disturbance results in more *vector STAR* operations and leads to a reduction in *holding* and *vector merge* operations.
- The applied coordination mechanism does not prevent the trombone segments from being saturated with traffic. This can be seen in the relative number of aircraft that are still operating the *vector merge* operation during disturbance.
- The absorption phase in the measured performance indicators is affected positively by the application of larger *time buffers*. The recovery phase on the other hand is not really affected by these enlarged *time buffers*.
- The workload of the ARR controller will only benefit negligibly from the applied coordination mechanism, while the workload of the TNW/TNE controllers will increase slightly due to increased vector operations.

9 Discussion, conclusions and recommendations

This chapter presents the conclusions that have been drawn regarding the research questions that are posed in section 3.3. Section 9.1 will first briefly describe the relevance of the followed quantitative agent-based modelling and simulation approach in the field of resilience engineering. Section 9.2 will thereafter present the conclusions in terms of the posed research questions. Section 9.3 at last marks the end of this report by presenting recommendations for future research.

9.1 Discussion

This specific resilience study aims to provide insight in how approach operations behave and how controller workload is affected during disturbing conditions by quantifying a number of related performance indicators. The simulation results provide an indication of how sensitive such operations are with respect to a bad weather disturbance and how such operations evolve as a function of time. When considering the research gaps that have been identified this study contributes to the need for more structured modelling approaches in the field of resilience engineering. The conducted experiments and obtained results demonstrate how and in which detail a quantitative agent-based modelling and simulation approach can support resilience engineering. This quantitative agent-based modelling and simulation approach completes the resilience engineering cycle that has been proposed by Stroeve and Everdij.

The resilient capacities of conventional approach operations are mainly defined by the actions and the performance of the executive controllers. The controllers' situation awareness, reasoning and decision-making are in this context essential processes that are covered in the developed agent-based model. Human cognition modelling is however a challenging task, so one could wonder to what extent one should model the controllers' cognitive skills before meaningful results can be obtained. The controller's situation awareness, reasoning and decision-making are so complex that a model will never be able to fully simulate such activities. Nevertheless the extent to which controller behaviour has been modelled is considered to be adequate and such that a sufficient level of adaptive strategies have been incorporated. One has to understand however that the flown operations as the result of controller instructions are still an approximation of the operations that would have been flown in reality. The obtained simulation results are therefore indicative and could deviate slightly from the results that would have been obtained when real approach operations are considered.

Resilience in the context of approach operations can be described as a relative property. This is because controllers are generally in all type of situations capable to maintain safe air traffic operations. The considered arrival and approach operations are therefore not necessarily bounded by some prescribed limitations, mostly because of the unlimited intervention possibilities that are provided by the three-dimensional airspace. For this reason unlimited vectoring and holding operations are considered and assumed in the model. The responsible ANSP should eventually determine to what extent certain flight operations are still considered as acceptable. In other words the ANSP is responsible for defining the maximum allowed performance loss and the maximum durations of the absorption and recovery phases. Quantification of resilience is in this study therefore more related to exploring how certain parameter settings contribute (relatively seen) to the resilient capacities of approach operations. The simulation results that are discussed in the report do therefore only provide an indication of the type of flight operations that can be expected to emerge as the result of specific parameter settings.

The simulation results of this resilience study showed in which situations, i.e. with what parameter settings the trombone segments of the modelled STAR procedures become saturated. Such saturation marks a turning point in the achieved performance, since more vector instructions become necessary, making the flown approach operations less efficient and robust. The developed model and the obtained simulation results can therefore be used to expose bottlenecks in the used procedures. The design process of similarly shaped approach operations could therefore benefit from the insight that has been obtained in this resilience study and the type of operations that have been considered.

9.2 Conclusion

In this section conclusions are drawn by answering the research questions that have been posed in the beginning of this report (section 3.3). These research questions were set up to contextualize the following research objective:

"To quantify, analyse and understand the resilient capacities of conventional approach operations during a sudden and unexpected bad weather disturbance using a quantitative agent-based modelling and simulation approach"

Following this research objective, the conducted resilience study contributes to the need for more structured modelling approaches for analysis and quantification of resilience in complex socio-technical systems. This study has applied a quantitative agent-based modelling and simulation approach to examine its suitability for this need. The suitability of such approach for more profound analysis and quantification of resilience in complex socio-technical systems has been studied in the context of disturbed approach operations at Rome Fiumicino. The answers that are presented for each of the five research questions below are therefore based on the obtained simulation results of this specific case study.

1) How do conventional approach operations emerge as the result of a sudden and unexpected bad weather disturbance and to what extent can these emergent operations be described as being resilient?
This research question was posed to explore and to capture the way in which arrival and approach operations are affected and emerge as the result of a bad weather disturbance. The multiple parameter variation experiments that have been conducted for this purpose considered an initial achieved runway capacity of around 36 ac·h^{-1} and a reduced to be achieved runway capacity of around 21 ac·h^{-1}. The obtained simulation results showed that various vector operations, holding operations and go-arounds are initiated by the approach/tower controllers as the result of this relatively large capacity reduction. The extent to which these adaptive strategies need to be applied is found highly dependent on the capacities that are maintained before disturbance and the size of the capacity reduction after disturbance. The obtained simulation results showed that more and longer vector and holding operations are required in the situation with a relatively high maintained initial throughput capacity and a relatively high reduced runway capacity. In other words, these capacities are found to increase the time period that is required by the controllers to properly adjust the traffic situation in the approach sectors as the result of a bad weather disturbance. The disturbed approach operations are found to recover again, i.e. aircraft start to resume their original stable arrival/approach operations again once the rate at which aircraft are delivered to the approach sector is (again) equal to or smaller than the achieved runway capacity. The resilient capacities of approach operations as the result of a bad weather disturbance are therefore for a large part defined by the various maintained capacities before and during disturbance.

2) To what extent are executive controllers able to maintain resilient approach operations during the sudden and unexpected bad weather disturbance?
This resilience study has in particular emphasized the role of executive controllers in achieving and maintaining resilience during both normal and disturbed approach operations. Each of the considered controllers has been assigned its own specific set of standard instructions that enable the controller to (re-)act in an appropriate and realistic manner if the observed traffic situation in its airspace sector requires to do so. The simulation results, and in particular the obtained trajectory plots showed that these modelled instructions are adequate to maintain safe, efficient and resilient arrival and approach operations, either before and during the bad weather disturbance. The provision of instructions as a means to maintain resilience during disturbed approach operations is found highly dependent on the ability of the controller to decide upon the necessity, feasibility and contents of a specific instruction. Therefore, the extent to which executive controllers are able to maintain resilience during disturbed approach operations by the provision of instructions is primarily defined by their skills, experience and creativity. This statement is supported by the fact that the inclusion of additional adaptive strategies (e.g. *holding* and *vector trombone* operations) and cognitive skills (e.g. situation awareness and decision-making) has proven to further enhance the resilient properties of disturbed approach operations. Resilience in the context of conventional approach operations is therefore a relative property, where controllers are 'at all time' able to maintain a certain minimum level of resilient performance. The specific resilient characteristics of conventional approach operations (i.e. absorption, adaptation, recovery) are however eventually defined by the number of adaptive strategies that are considered (e.g. operational concepts, controller intelligence and reasoning, etc.).

3) How can the socio-technical ATM system be adapted to improve the resilient capacities of conventional approach operations in the context of a bad weather disturbance?
Conventional approach operations are considered to be more resilient if relatively less controller actions are required to cope with the bad weather disturbance, and when aircraft are able to resume their standard arrival and approach operations in a relatively fast and fluent manner after having operated a specific type of adaptive strategy (e.g. vectoring strategy). When combining this definition with the type of trajectories that are visualized in the obtained trajectory plots there can be seen that the characteristics of the considered trombones have potential to further improve the resilient capacities of approach operations in the context of a bad weather disturbance. The shape of the trajectories in the obtained trajectory plots do clearly show the significant use of the trombone segments in the modelled approach operations both before and during disturbance. Approach procedures that incorporate the concept of tromboning can be made more resilient for a bad weather disturbance by simply applying larger trombones (i.e. by enlarging the *downwind segment*). These additional leg distances can be used ('consumed') by aircraft when the approach operations are confronted with a sudden capacity reduction. The workload of the approach controllers will not be affected in such configuration since aircraft can just remain operating the fixed profile of the enlarged trombone segment (up to a certain point).

4) What is the added value of a quantitative agent-based modelling and simulation approach for resilience engineering over a qualitative agent-based modelling approach?
Following the identified research gaps in the field of resilience engineering this study examined the suitability of a quantitative agent-based modelling and simulation approach for more profound analysis and quantification of resilience in socio-technical systems. The obtained quantitative results are meant to further support and to verify the qualitative mental simulation results that have been obtained during the qualitative agent-based modelling phase.

When comparing the simulation results and features of both approaches there can be concluded that a quantitative agent-based modelling and simulation approach is able to provide more detail and insight about the resilient capacities of socio-technical systems. This increased detail is first of all found in the obtained quantitative simulation results. These results showed that, if an appropriate selection of performance indicators is considered, that the quantitative approach is able to provide detailed predictions about (e.g.) the extent to which a socio-technical system has to adapt to a specific disturbance, the time period that is required to recover from a disturbance, the maximum performance loss that will be suffered, etc. The validity of these predictions is however always dependent on the level of detail that has been incorporated in the agent-based model. The dynamic patterns that can be observed in the obtained quantitative simulation results do however show similar shapes as the ones that are considered in the qualitative mental simulation results. This means that a quantitative agent-based modelling and simulation approach can be used to support and to verify the results obtained during the qualitative agent-based modelling phase.

The results that are obtained by applying a quantitative agent-based modelling and simulation approach are completely defined by (e.g.) the modelled aspects, interactions, behaviours and states. This means that the (to be) measured resilient capacities are completely defined by the contents of the developed agent-based model. In this way quantitative agent-based modelling and simulation may thus also be used to trace back the cause of specific resilient behaviour in complex socio-technical systems, which is not possible when applying a qualitative agent-based modelling approach. The answer that is provided for the next research question will further elaborate on this ability by describing the specific functionalities of *AnyLogic* that are found useful for this purpose.

5) How does *AnyLogic* contribute to the implementation and simulation of formal agent-based models?
The developed formal agent-based model has been implemented and simulated in *AnyLogic*. *AnyLogic* is found to be a very suitable and powerful software tool to study the emergent and resilient behaviours of large-scale and complex agent-based models. *AnyLogic* provides a number of modelling elements and modelling methods that allow to specify the internal states and properties of each agent and the interactions in and between agents in a very efficient, structured and readable manner. The implementation of an agent-based model into *AnyLogic* is fully mapped into Java code, which allows for unlimited and flexible extension possibilities. Because of the many graphical features and its structured architecture *AnyLogic* allows to visually trace e.g. how each agent changes state during simulation, how agents interact with each other, how the overall modelled socio-technical system evolves over time, etc. *AnyLogic* is found to support a large set of powerful experiments that are relatively easy to create, such as the parameter variation experiments and the Monte Carlo experiments. The only real shortcoming in *AnyLogic* is found to be its

limited ability to internally process the obtained data while running experiments, and the fact that *AnyLogic* is only able to export the obtained data to Excel.

9.3 Recommendations for further research

The experiments that have been conducted in this study do only consider the implications of a few parameter settings. The model that has been developed for this resilience study provides however also a stable basis for further research and analysis of arrival and approach operations. The developed software tool in *AnyLogic* allows to analyse in a user friendly manner the implications of other parameter settings on arrival and approach operations. Further research in this specific operational field could therefore be related to the implications of: wind conditions, the distribution of wake turbulence categories, the distribution of the entry points at which aircraft enter the approach sectors, the structure of the used STAR procedures and the corresponding trombone segments, speed and altitude profiles and runway occupancy times (ROT). The model can of course also be extended by adding more features, detail and type of controller actions, or by implementing new operational concepts. The tool that has been developed in *AnyLogic* can therefore be used as a powerful and structured starting point for further research and analysis of arrival and approach operations and for resilience engineering in general.

It should be noted that the qualitative study of Stroeve and Everdij also considered a second operational concept in addition to the conventional approach operations. This second type of operation is called Interval Management (IM), which is defined as “the overall system that enables the improved means for managing traffic flows and aircraft spacing through automated inter-aircraft spacing” [7]. It is the answer to a demand of future ATM for increased capacity and flight efficiency while maintaining flight safety. IM is a dynamic process between flight crew, air traffic control and CNS systems where aircraft spacing is automatically achieved and maintained by an on-board IM system, leading to a reduction in workload of both controllers and flight crews. Studies have shown that IM results in improved inter-aircraft spacing precision, which allows aircraft to be spaced relatively closer in comparison with conventional approach operations. More precise inter-aircraft spacing will therefore result in improved airspace capacity and reduced delays. IM operations differ however significantly from conventional approach operations. This is because aircraft spacing (separation) is during conventional operations manually controlled and maintained by air traffic control and the flight crew. During IM operations however an onboard IM system becomes responsible for automatically maintaining (and achieving) a specific spacing with some target aircraft. Because of this different philosophy IM operations are expected to behave significantly different during a bad weather disturbance when compared to conventional approach operations. This is especially due to the larger traffic densities and a shift in responsibilities when IM is active. A sudden and unexpected bad weather disturbance will therefore most likely have a different effect on both type of operations in terms of for instance controller workload and required vectoring. Future research could therefore be devoted to the implications of a bad weather disturbance on IM operations, since this type of operation is still considered as a promising innovation in near future ATM.

Bibliography

[1] Bergström, J., van Winsen, R., Henriqson, E. (2015). *On the rationale of resilience in the domain of safety: A literature review.* Reliability Engineering & System Safety. https://doi.org/10.1016/j.ress.2015.03.008

[2] Bonabeau, E. (2002). *Agent-based modelling: methods and techniques for simulating human systems*. Proceedings of the National Academy of Sciences. https://doi.org/10.1073/pnas.082080899

[3] Bouarfa, S., Blom, H.A.P., Curran, R., Everdij, M.H.C. (2013). *Agent-based modelling and simulation of emergent behaviour in air transportation*. Complex Adaptive Systems Modeling. https://doi.org/10.1186/2194-3206-1-15

[4] Bruneau, M., Chang, S.E., Eguchi, R.T., Lee, G.C., O'Rourke T.D., Reinhorn, A., Shinozuka, M., Tierney, K., Wallace, W., Winterfeldt, D. (2003). *A framework to quantitatively assess and enhance the seismic resilience of communities*. Earthquake Spectra. https://doi.org/10.1193/1.1623497

[5] Darley V. (1994). *Emergent phenomena and complexity*. in Artificial Life IV. Proceedings of the Fourth International Workshop on the Synthesis and Simulation of Living Systems

[6] Endsley, M.R. (1995). *Toward a theory of situation awareness in dynamic systems*. Texas Tech University. Human Factors. https://doi.org/10.1518/001872095779049543

[7] Eurocae. (2015). *Safety, Performance and Interoperability Requirements Document for Airborne Spacing Flight-deck Interval Management (ASPA-FIM)*

[8] Eurocontrol. (1997). *Model of the cognitive aspects of air traffic control*

[9] Eurocontrol. (2009). *A white paper on resilience engineering for ATM*. https://www.eurocontrol.int/archive_download/all/node/11591

[10] Eurocontrol. (2018). *European aviation in 2040 – Challenges of growth*. https://www.eurocontrol.int/sites/default/files/2019-07/challenges-of-growth-2018-annex1_0.pdf

[11] Eurocontrol. (2020). *AIP Italy*. https://www.ead.eurocontrol.int/

[12] Eurocontrol. (2020). *BADA Aircraft Performance Model*. https://simulations.eurocontrol.int/

[13] Eurocontrol (2020). *Point Merge implementation – A quick guide*. Edition 1.2. https://www.eurocontrol.int/sites/default/files/2020-03/eurocontrol-point-merge-guide-v1.2.pdf

[14] European Commission (2011). *Flightpath 2050 Europe's Vision for Aviation*. https://doi.org/10.2777/50266

[15] Everdij, M.H.C., Stroeve, S.H., Bottone, M., Diaz Dominguez, C., De Gelder, N., Paino, M. (2016). *ASAS - Interim Report 2014*. Project 16.06.01b D03-001. SESAR Joint Undertaking

[16] Francis, R., Bekera, B. (2014). *A metric and frameworks for resilience analysis of engineered and infrastructure systems*. Reliability Engineering & System Safety. https://doi.org/10.1016/j.ress.2013.07.004

[17] Henry, D., Ramirez-Marquez, H. (2012). *Generic metrics and quantitative approaches for system resilience as a function of time*. Reliability Engineering & System Safety. https://doi.org/10.1016/j.ress.2011.09.002

[18] Hollnagel, E., Woods, D.D. (2006). *Epilogue: Resilience Engineering Precepts.*

[19] Hollnagel, E., Pariès, J., Woods, D.D., Wreathall, J. (2011). *Resilience engineering in practice - A guidebook*. Ashgate Publishing Limited A. ISBN 978-1-4094-1035-5

[20] Hollnagel, E. (2015). *RAG – Resilience Analysis Grid*.

[21] Hollnagel, E. (2016). *Resilience engineering: a new understanding of safety*. http://dx.doi.org/10.5143/JESK.2016.35.3.185

[22] Hosseini, S., Barker, K., Ramirez-Marquez (2016). *A review of definitions and measures of system resilience*. Reliability Engineering & System Safety. https://doi.org/10.1016/j.ress.2015.08.006

[23] ICAO. (2007). *RNAV Training for ATC*. https://www.icao.int/safety/pbn/Documentation/States/Japan_RNAV%20Training%20for%20ATC.pdf

[24] ICAO. (2010). *Doc 8168 Aircraft Operations Volumes I and II* (5th edition)

[25] ICAO (2016). *Doc 4444 PANS-ATM* (16th edition)

[26] ICAO. (2018). *Annex 11 Air Traffic Services* (15th edition)

[27] Ingrisch, J., Bahn, M. (2018). *Towards a comparable quantification of resilience*. Trends in Ecology & Evolution. https://doi.org/10.1016/j.tree.2018.01.013

[28] Macal, C.M., North, M.J. (2010). *Tutorial on agent-based modelling and simulation*. Journal of Simulation. https://doi.org/10.1057/jos.2010.3

[29] Osgood, N. (2020). *AnyLogic Tutorials*. https://www.cs.usask.ca/faculty/ndo885/Classes/ConsensusABM/Lectures.html

[30] Patriarca, R., Bergström, J., Di Gravio, G., Constantino, F. (2018). *Resilience engineering: current status of the research and future challenges*. Safety Science. https://doi.org/10.1016/j.ssci.2017.10.005

[31] Pinska-Chauvin, E., Josefsson, B., Branlat, M. Everdij, M.H.C., Stroeve, S.H., Laursen, T., Herrera, I. (2016). *T4 Final report*. Project 16.06.01b D04 SESAR Joint Undertaking

[32] Privosnik, M., Marolt, M., Kavcic, A., Divjak, S. (2002). *Evolutionary construction of emergent properties in multi-agent systems*. in 11th IEEE Mediterranean Electrotechnical Conference (IEEE Cat. No.02CH37379)
[33] Resilience2050. (2012). https://resilience2050.org/
[34] Righi, A.W., Saurin, T.A., Wachs, P. (2015). *A systematic literature review of resilience engineering: research areas and a research agenda proposal*. Reliability Engineering & System Safety. https://doi.org/10.1016/j.ress.2015.03.007
[35] Romano, M. et al. (2015). *VP-708 Participant Handbook*. SESAR
[36] Sharpanskykh, A. (2011). *Agent-based modeling and analysis of socio-technical systems*. Cybernetics and Systems. https://doi.org/10.1080/01969722.2011.595332
[37] Stroeve, S.H., Everdij, M.H.C., Blom, H.A.P. (2011). *Hazards in ATM: model constructs, coverage and human responses*. E.02.10-MAREA-D1.2. SESAR Joint Undertaking
[38] Stroeve, S.H., Bosse, T., Blom, H.A.P., Sharpanskykh, A., Everdij, M.H.C. (2013). *Agent-based modelling for analysis of resilience in ATM*. Third SESAR Innovation days. Stockholm. Sweden
[39] Stroeve, S.H., Van Doorn, B.A., Everdij, M.H.C., (2013). *The human contribution – analysis of the human role in resilience in ATM.* Deliverable D1.2. Resilience2050.eu https://doi.org/10.13140/2.1.3527.3287
[40] Stroeve, S.H., Van Doorn, B.A., Everdij, M.H.C. (2015). *Analysis of the roles of pilots and controllers in the resilience of air traffic management*. Safety Science. https://doi.org/10.1016/j.ssci.2015.02.023
[41] Stroeve, S.H, Everdij, M.H.C., De Gelder, N., Bottone, M., Diaz Dominguez, C., Paino, M. (2016). *ASAS - Interim Report 2015*. Project 16.06.01b D03-002. SESAR Joint Undertaking
[42] Stroeve, S.H., Everdij, M.H.C. (2017). *Agent-based modelling for analysis of resilience in ATM*. Safety Science https://doi.org/10.1016/j.ssci.2016.11.003
[43] Stroeve, S.H., Everdij, M.H.C. (2017). *Agent-based modelling and mental simulation for resilience engineering in air transport*. Safety Science. https://doi.org/10.1016/j.ssci.2016.11.003
[44] Tran, H.T., Balchanos, M., Domercant, J.C., Mavris, D.N. (2017). *A framework for the quantitative assessment of performance-based system resilience*. Reliability Engineering & System Safety. https://doi.org/10.1016/j.ress.2016.10.014
[45] Van Dam, K.H., Nikolic, I., Lukszo, Z. (2013). *Agent-based modelling of socio-technical systems*. https://doi.org/10.1007/978-94-007-4933-7. ISBN 978-94-007-4933-7
[46] Vugrin, E. D., Warren, D.E., Ehlen, M. (2011). *A resilience assessment framework for infrastructure and economic systems: quantitative and qualitative resilience analysis of petrochemical supply chains to a hurricane*. Process Safety Progress. https://doi.org/10.1002/prs.10437
[47] Wang, J., Zuo, W., Rhodebarbarigos, L., Lu, X., Wang, J., Lin, Y. (2019). *Literature review on modelling and simulation of energy infrastructures from a resilience perspective*. Reliability Engineering & System Safety. https://doi.org/10.1016/j.ress.2018.11.029
[48] Woltjer, R., Pinska-Chauvin, E., Laursen, T., Josefsson, B. (2015). *Towards understanding work-as-done in air traffic management safety assessment and design*. Reliability Engineering & System Safety. https://doi.org/10.1016/j.ress.2015.03.010
[49] Woltjer, R. (2019). *Exploring resilience at interconnected system levels in air traffic management*. Exploring Resilience. https://doi.org/10.1007/978-3-030-03189-3_13
[50] Woods, D.D. (2015). *Four concepts for resilience and the implications for the future of resilience engineering*. Reliability Engineering & System Safety. https://doi.org/10.1016/j.ress.2015.03.018
[51] Woods, D.D. (2018). *The theory of graceful extensibility: basic rules that govern adaptive systems*. Environment Systems and Decisions. https://doi.org/10.1007/s10669-018-9708-3
[52] Yang, Q., Tian, J., Zhao, T. (2017). *Safety is an emergent property: illustrating functional resonance in air traffic management with formal verification*. Safety Science. https://doi.org/10.1016/j.ssci.2016.12.006

Appendix A Arrival and approach procedures

This appendix shows the arrival and approach procedures for runway 16L at Rome Fiumicino. The information on these charts has been used to shape the environment and to control the flight operations of aircraft during arrival and approach in terms of airspeed, altitude and heading.

A.1 XIBIL2A STAR procedure

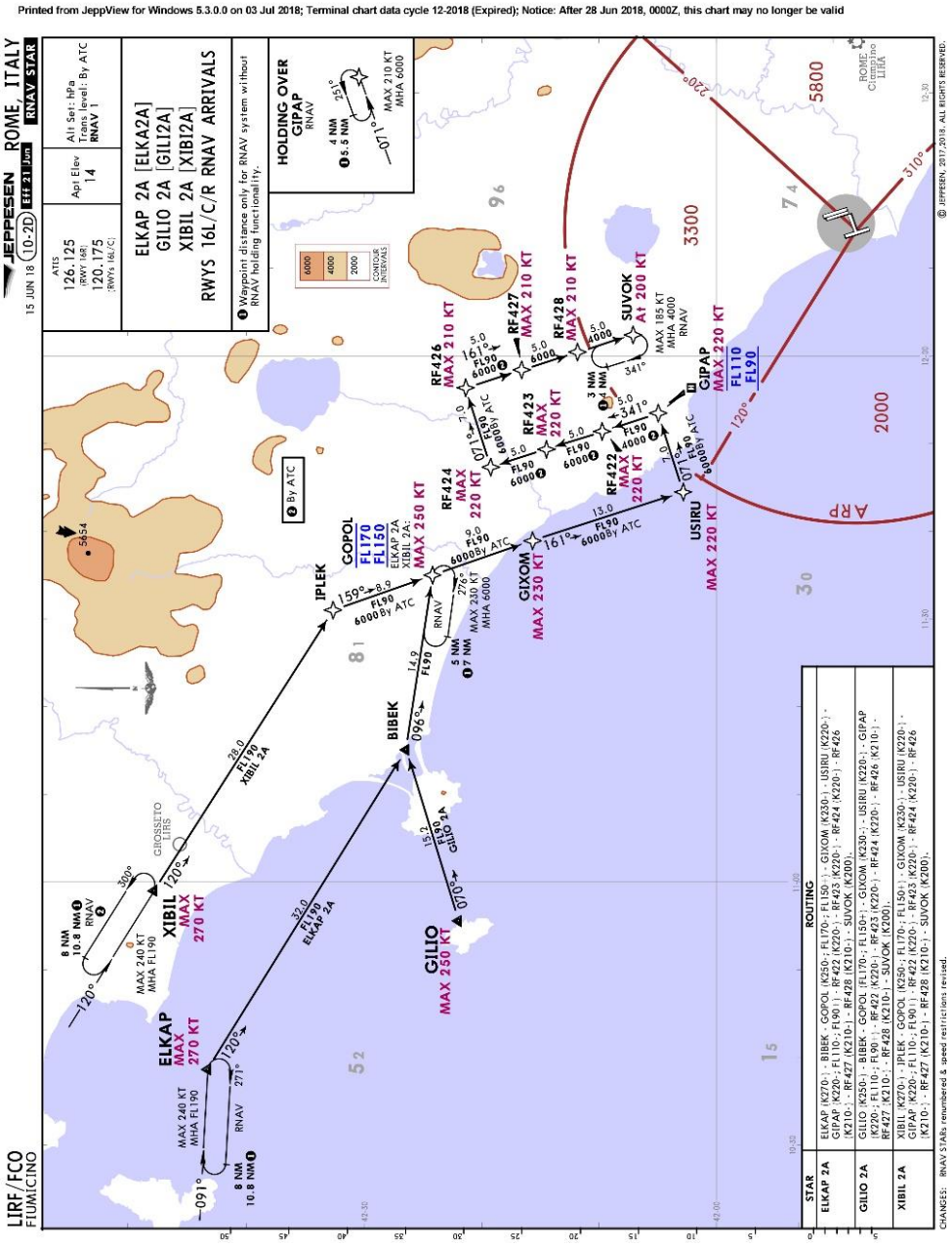


Figure 49 – XIBIL2A RNAV STAR procedure

A.2 RITEB2A STAR procedure

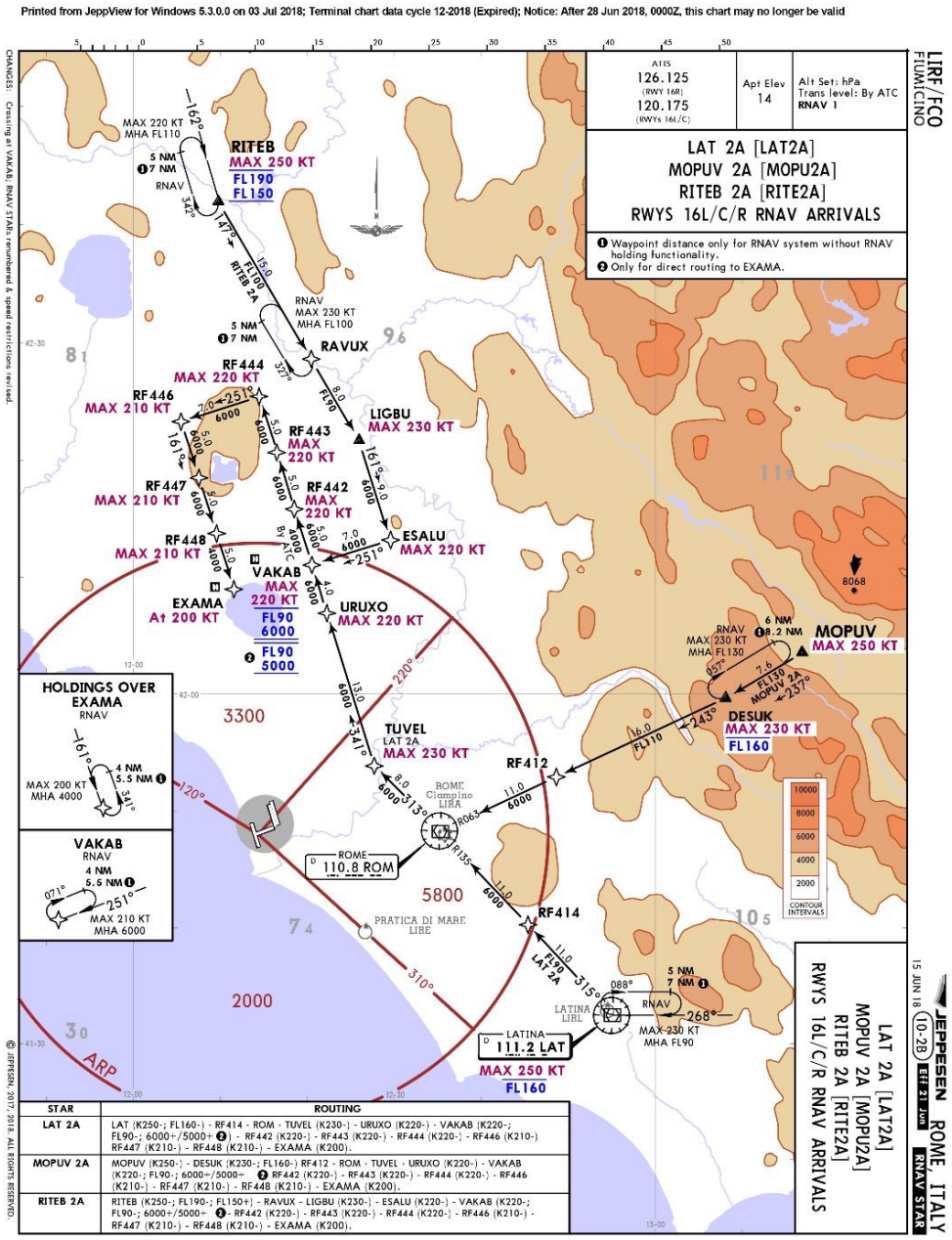


Figure 50 – RITEB2A RNAV STAR procedure

A.3 Instrument Approach Chart Runway 16L

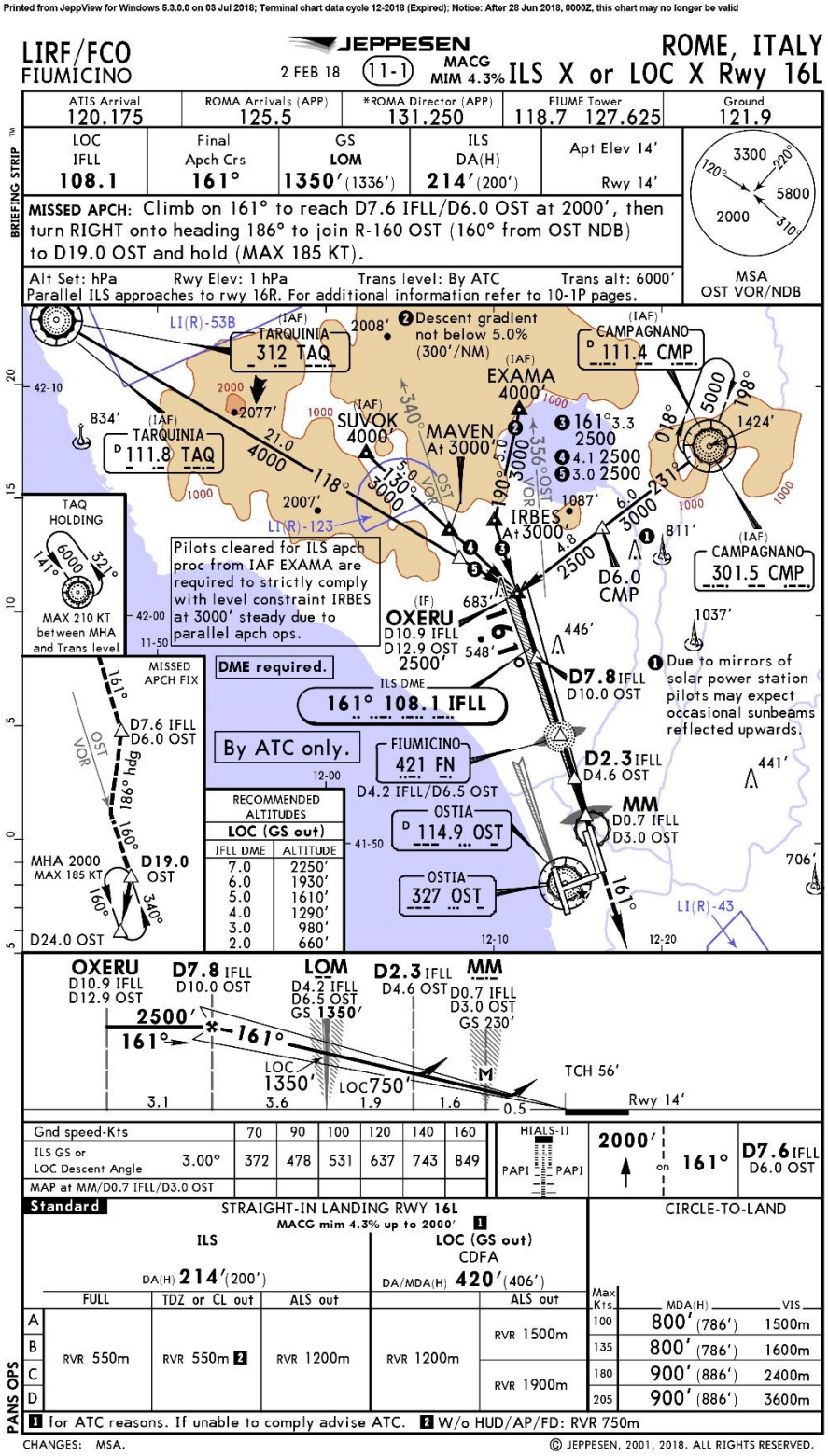


Figure 51 – Instrument Approach Chart Runway 16L

Appendix B Dynamic patterns in measured performance indicators

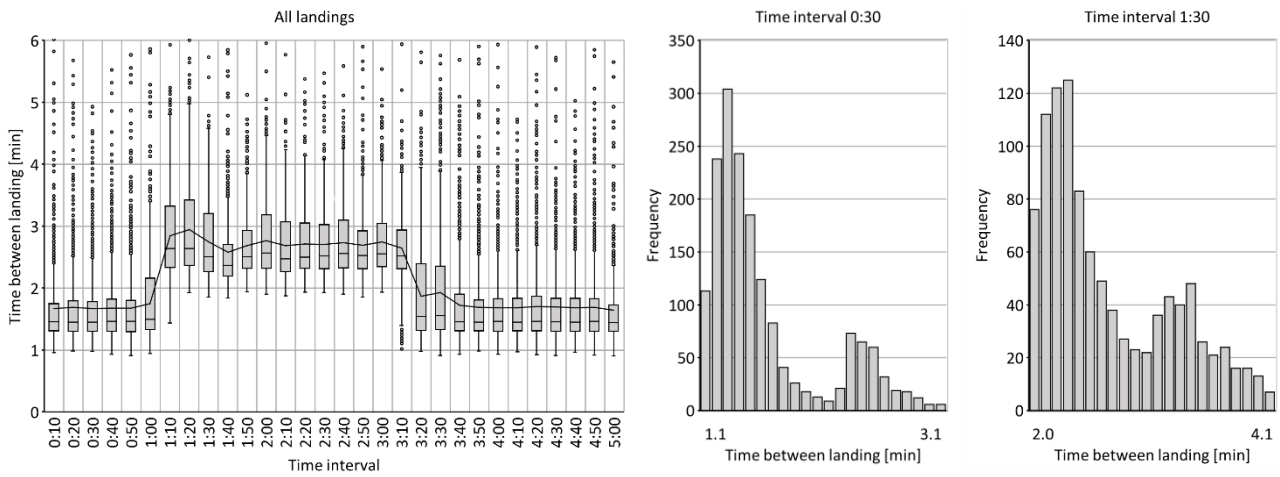
This appendix presents in more detail the dynamic simulation results that have been obtained after the multiple parameter variation experiments. These results are presented for each defined performance indicator with a series of boxplots and histograms. These diagrams are used to visualize the dynamic patterns of each measured performance indicator as a function of time. These captured dynamics are meant to provide more context and detail to the "summarized" simulation results as provided in section 8.3. Appendix B.1 provides a full set of dynamic simulation results of the measured performance indicators that correspond to experiment 1. Appendices B.2 and B.3 do only provide partial simulation results of experiments 2 and 3 respectively, since the omitted simulation results contain similar type of dynamic patterns as the ones that are shown in appendix B.1.

B.1 Experiment 1

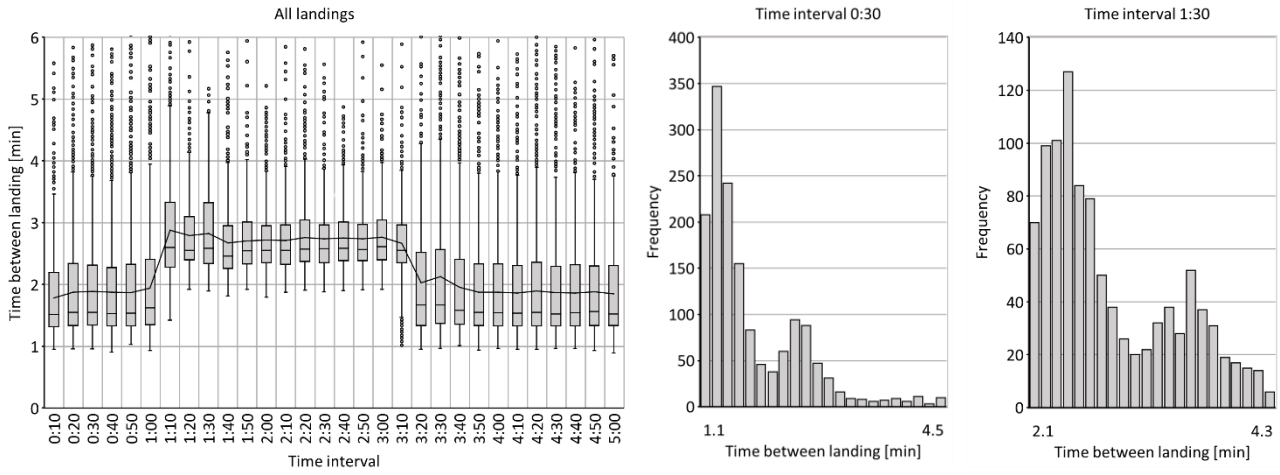
This appendix is meant to provide further context to the simulation results that are presented in table 12 by providing the measured dynamics of the following performance indicators:

- *Time between landing*
- *Number of aircraft in approach*
- *Number of landings*
- *Number of go-arounds*
- *Number of vector inbound trombone*
- *Number of vector outbound trombone*
- *Percentage flying STAR*
- *Percentage flying vector STAR*
- *Percentage flying vector merge*
- *Percentage flying vector IF*
- *Percentage flying holding*
- *Time in approach*
- *Time flying STAR*
- *Time flying vector STAR*
- *Time flying vector merge*
- *Time flying vector IF*
- *Time flying holding*
- *Percentage in tactical mode*
- *Number of instructions*

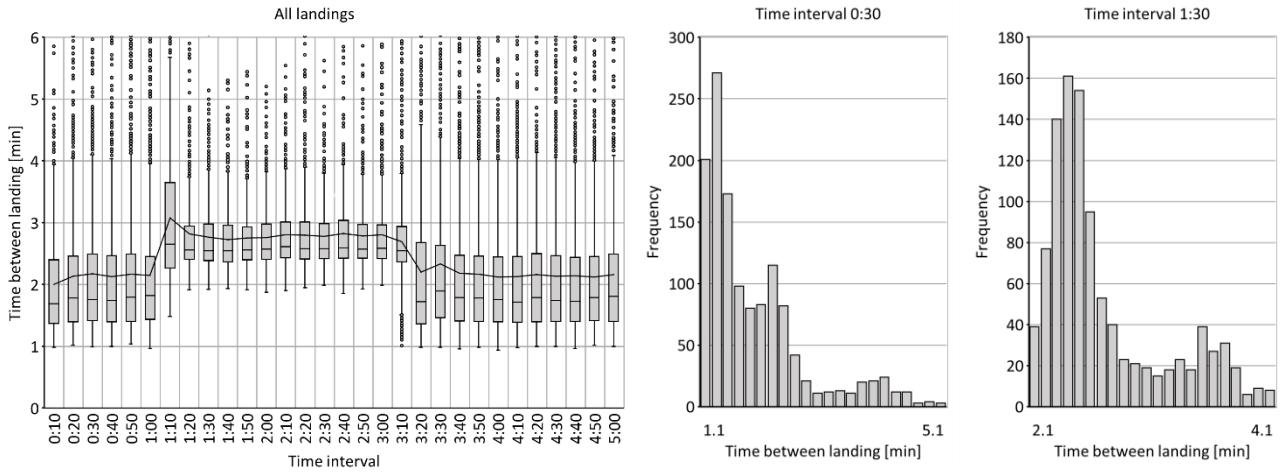
Time between landing – Experiment 1 – Scenarios 1, 2, 3



Simulation results of performance indicator "Time between landing", corresponding to scenario 1 of experiment 1

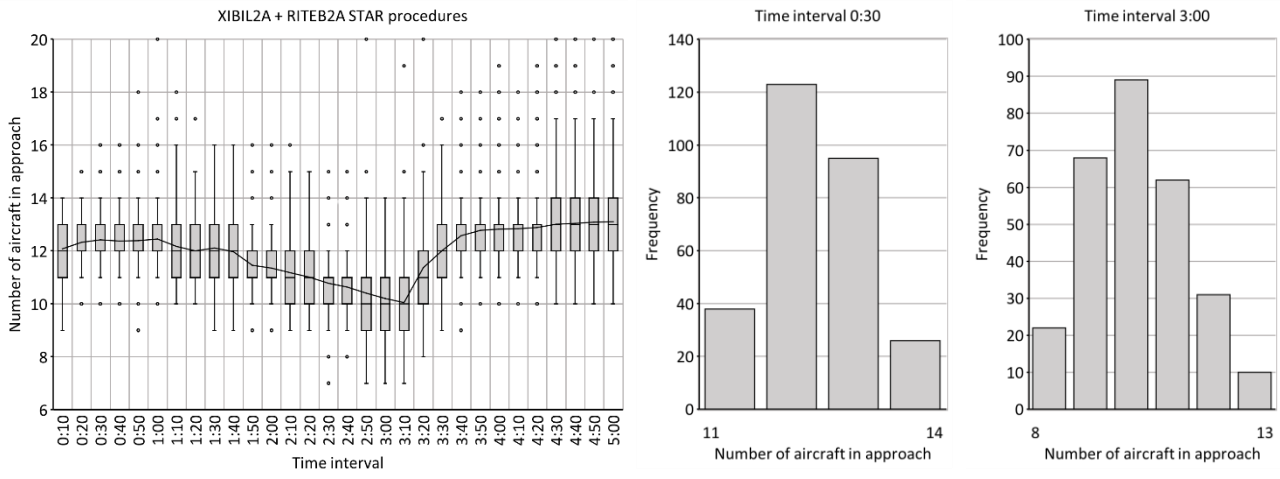


Simulation results of performance indicator "Time between landing", corresponding to scenario 2 of experiment 1

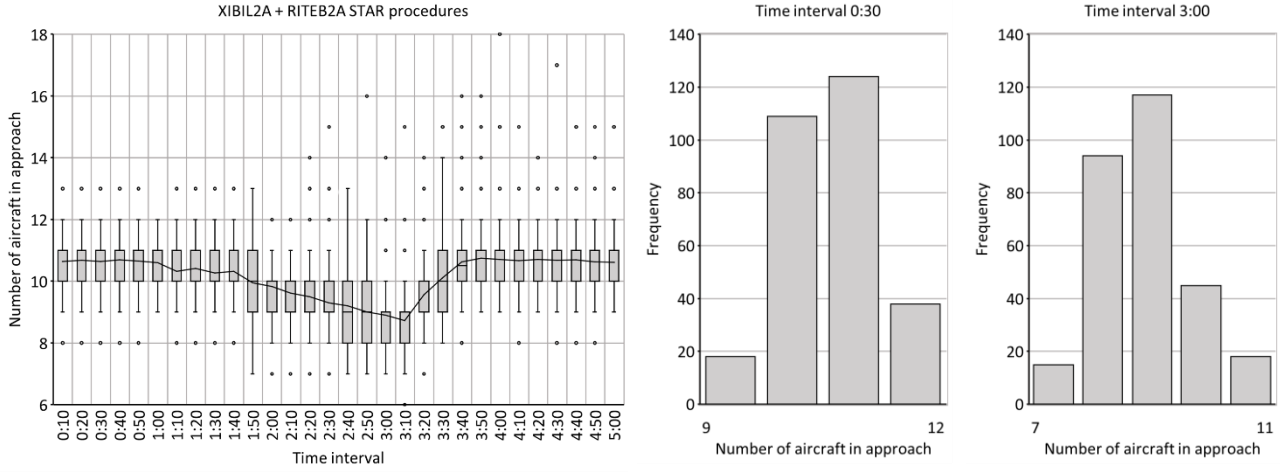


Simulation results of performance indicator "Time between landing", corresponding to scenario 3 of experiment 1

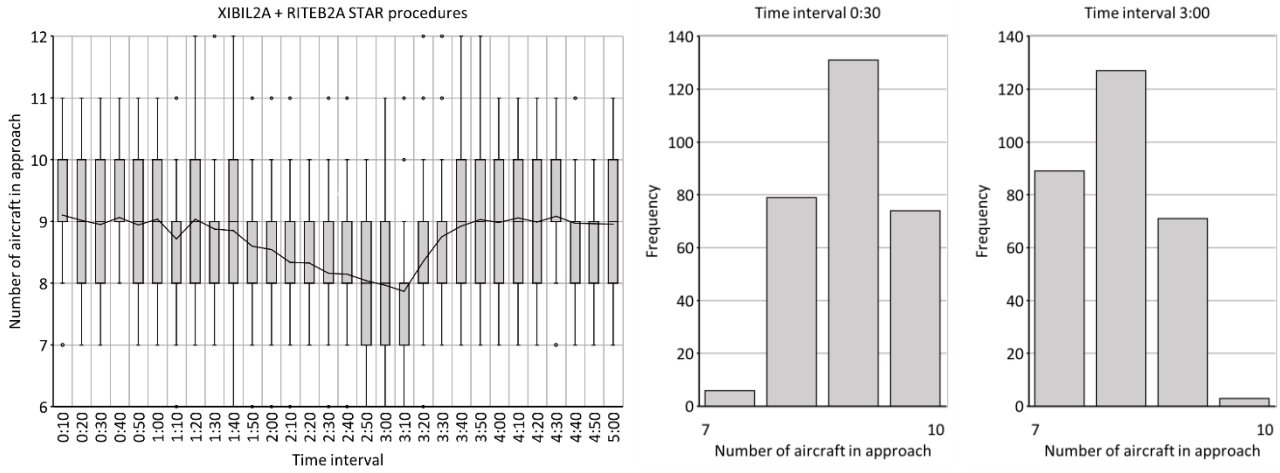
Number of aircraft in approach – Experiment 1 – Scenarios 1, 2, 3



Simulation results of performance indicator "Number of aircraft in approach", corresponding to scenario 1 of experiment 1

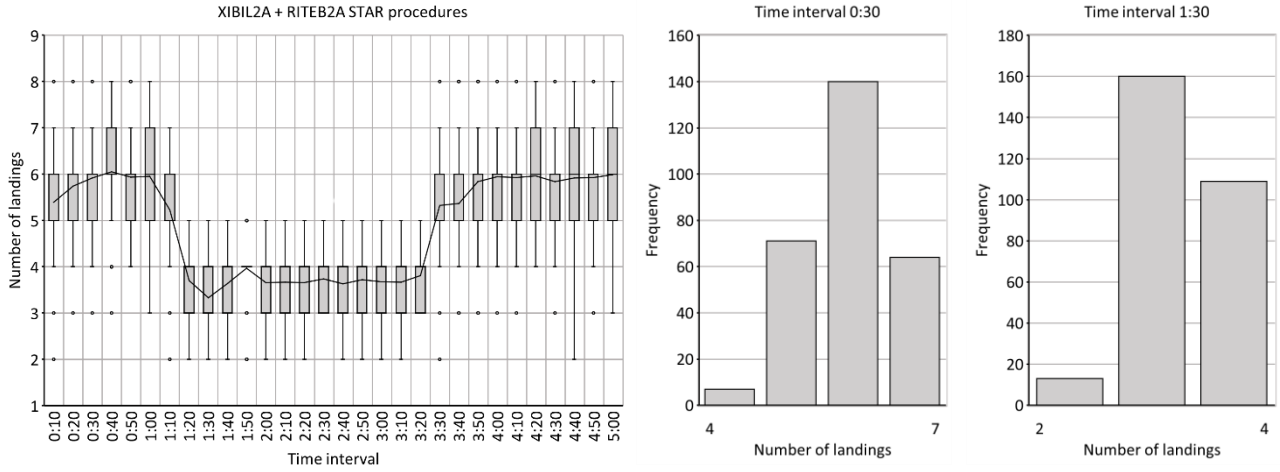


Simulation results of performance indicator "Number of aircraft in approach", corresponding to scenario 2 of experiment 1

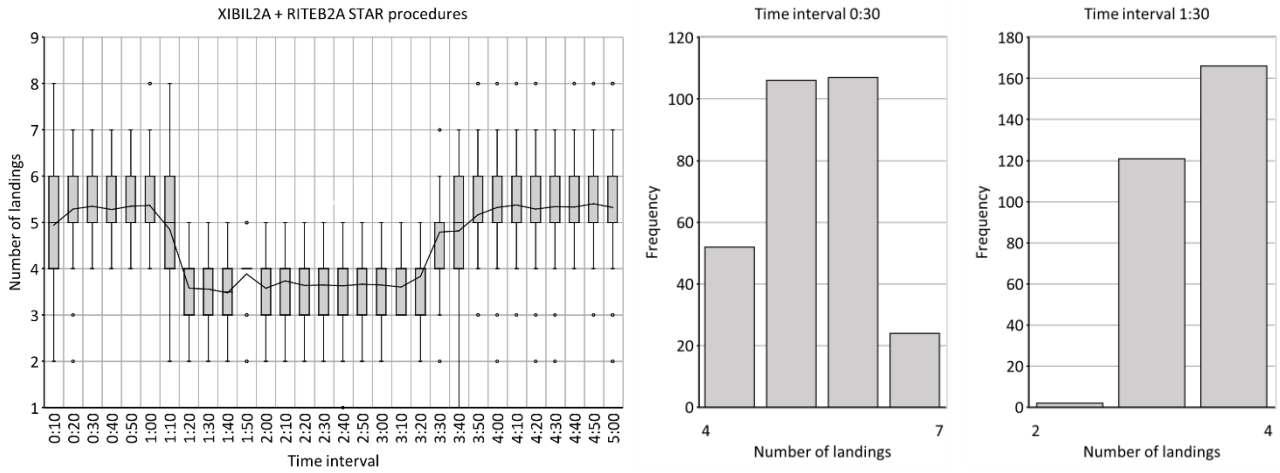


Simulation results of performance indicator "Number of aircraft in approach", corresponding to scenario 3 of experiment 1

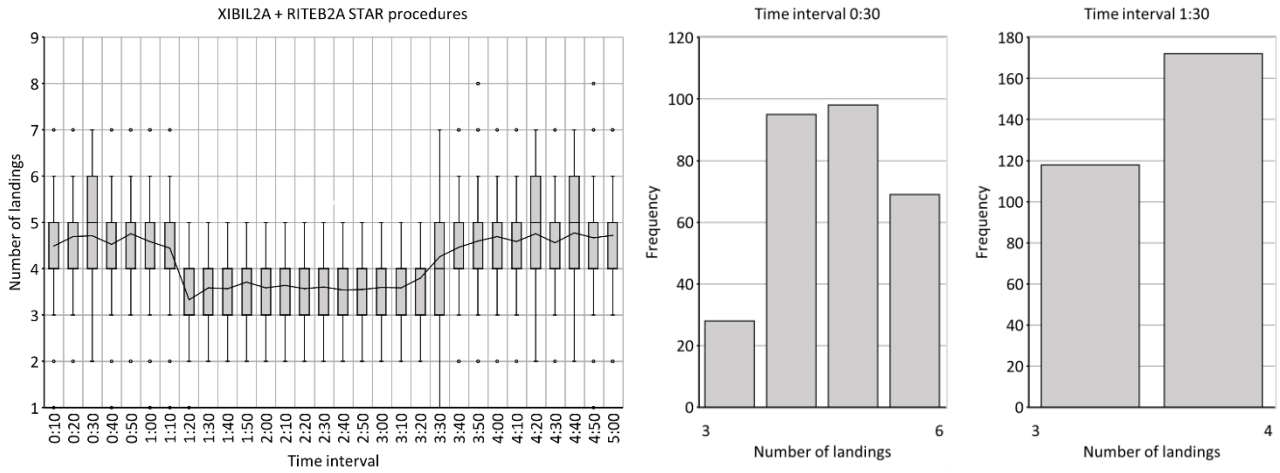
Number of landings – Experiment 1 – Scenarios 1, 2, 3



Simulation results of performance indicator "Number of landings", corresponding to scenario 1 of experiment 1

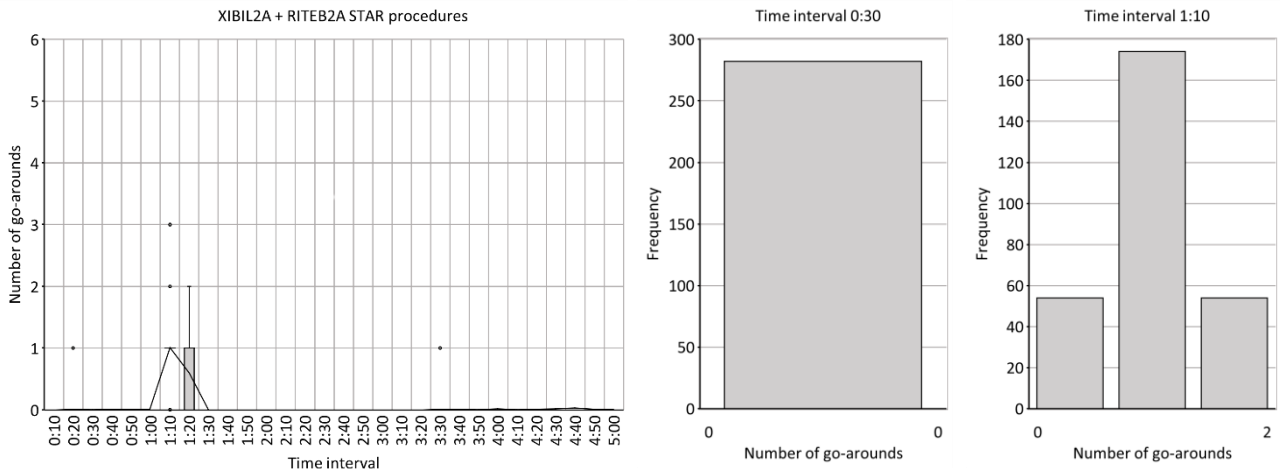


Simulation results of performance indicator "Number of landings", corresponding to scenario 2 of experiment 1

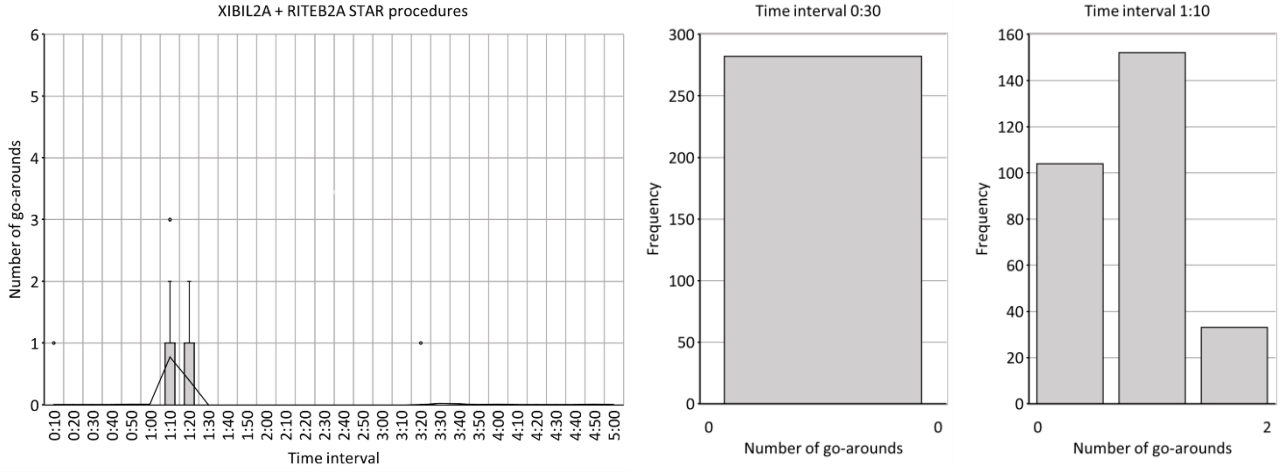


Simulation results of performance indicator "Number of landings", corresponding to scenario 3 of experiment 1

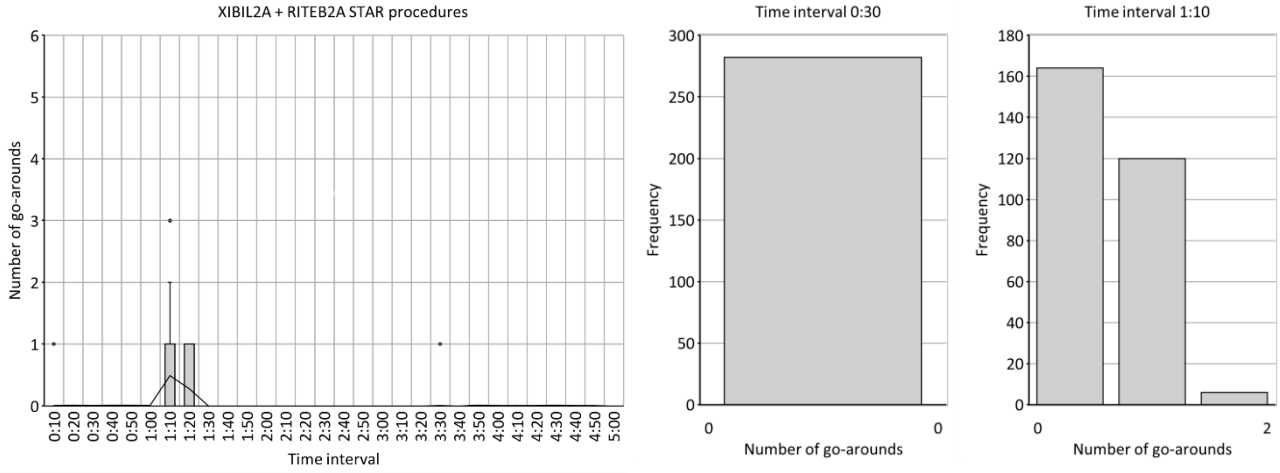
Number of go-arounds – Experiment 1 – Scenarios 1, 2, 3



Simulation results of performance indicator "Number of go-arounds", corresponding to scenario 1 of experiment 1

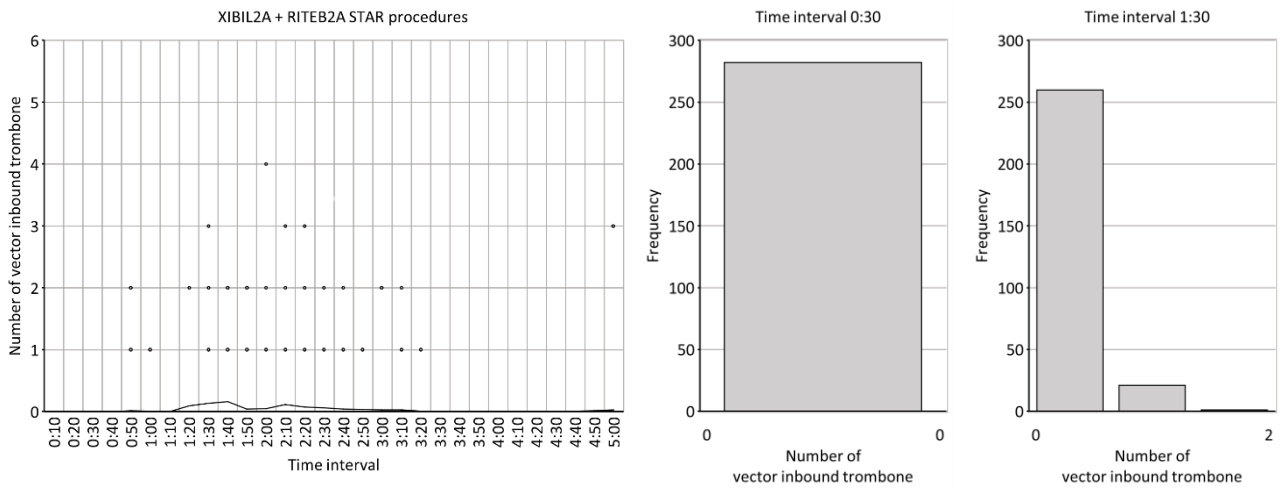


Simulation results of performance indicator "Number of go-arounds", corresponding to scenario 2 of experiment 1

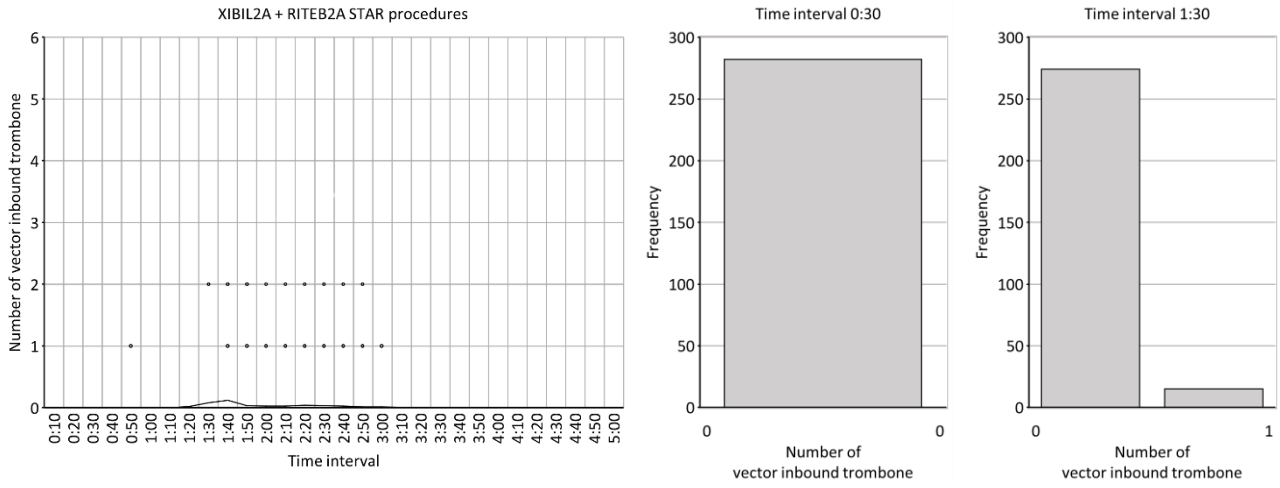


Simulation results of performance indicator "Number of go-arounds", corresponding to scenario 3 of experiment 1

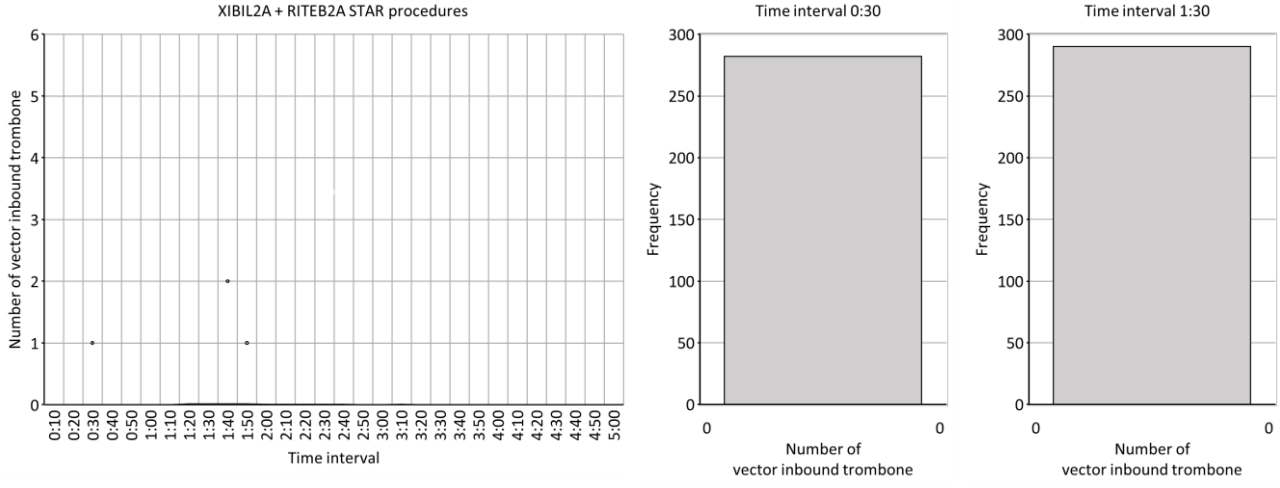
Number of vector inbound trombone – Experiment 1 – Scenarios 1, 2, 3



Simulation results of performance indicator "Number of vector inbound trombone", corresponding to scenario 1 of experiment 1

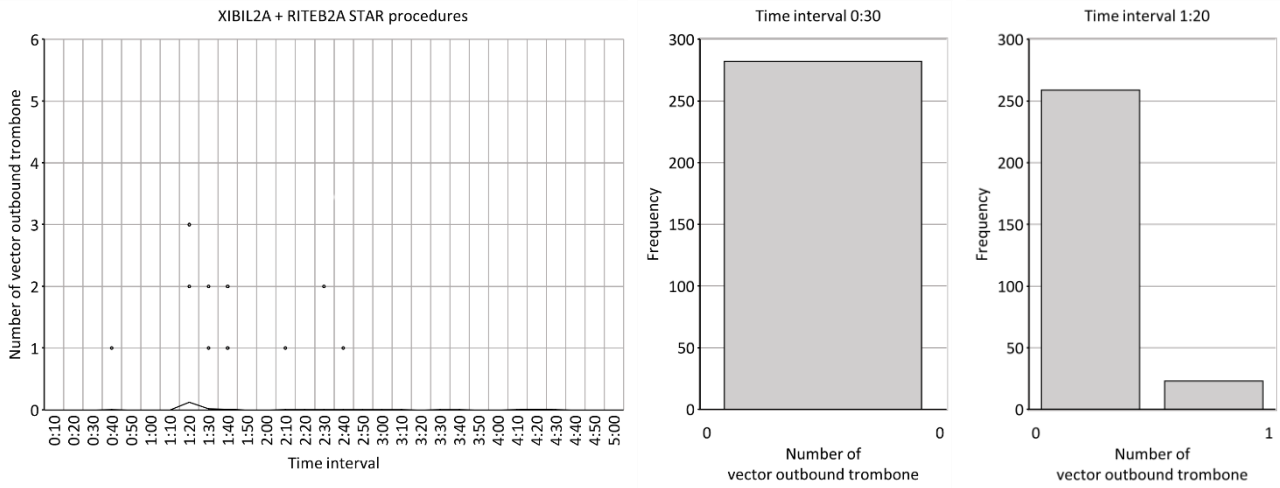


Simulation results of performance indicator "Number of vector inbound trombone", corresponding to scenario 2 of experiment 1

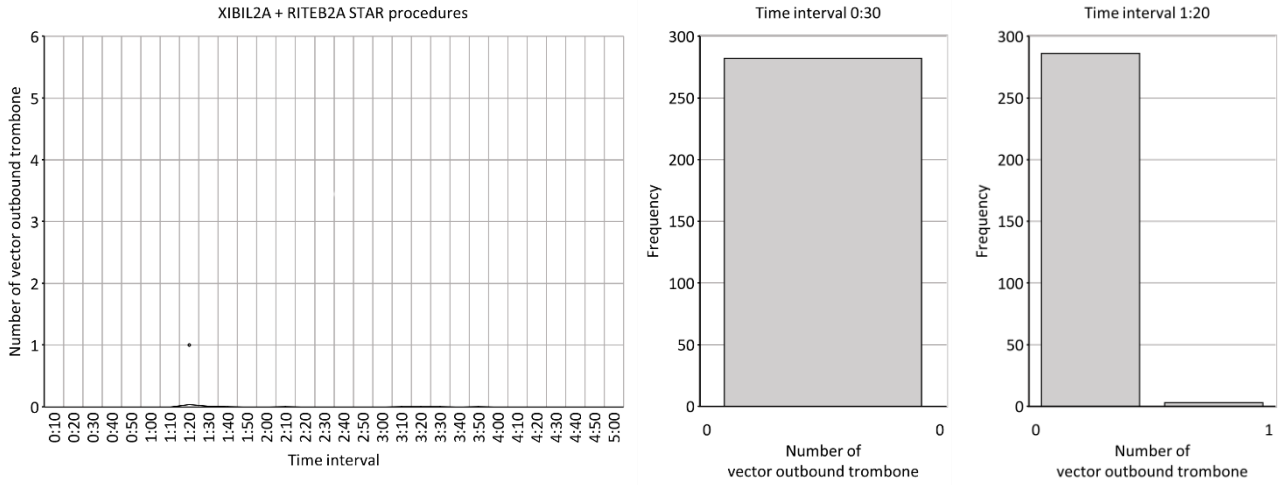


Simulation results of performance indicator "Number of vector inbound trombone", corresponding to scenario 3 of experiment 1

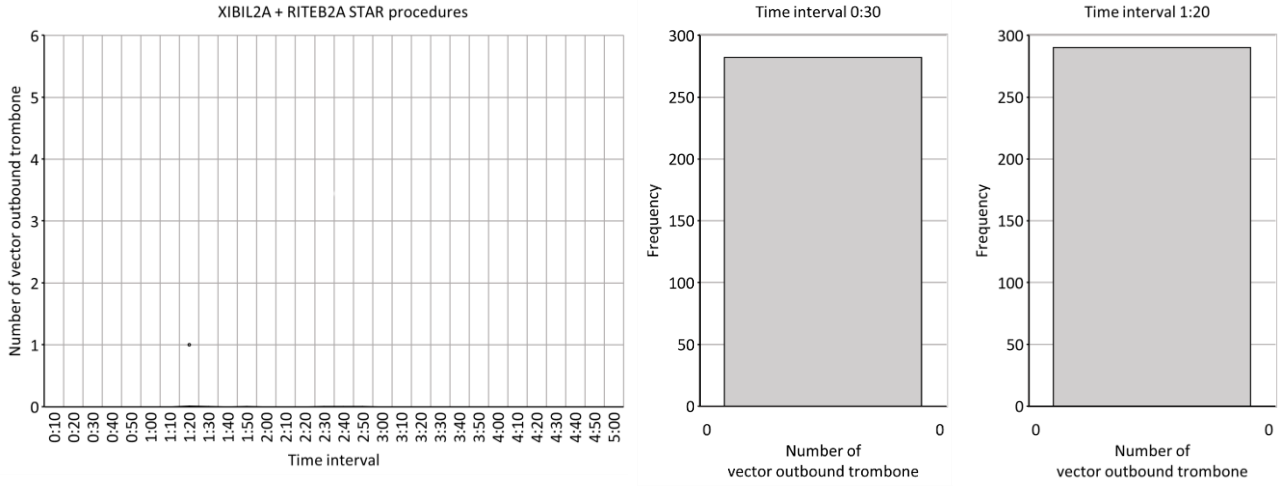
Number of vector outbound trombone – Experiment 1 – Scenarios 1, 2, 3



Simulation results of performance indicator "Number of vector outbound trombone", corresponding to scenario 1 of experiment 1

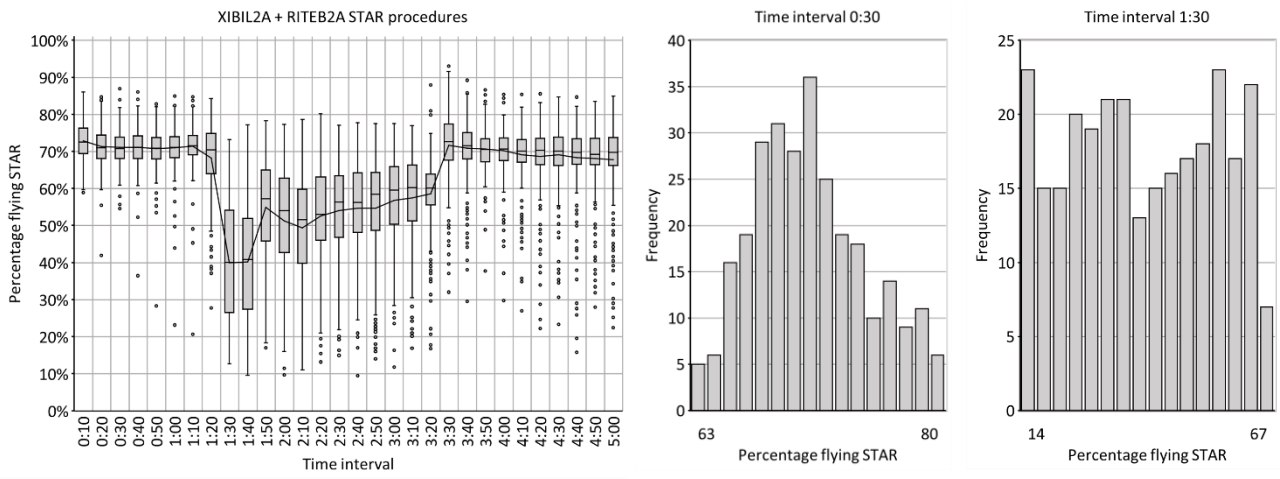


Simulation results of performance indicator "Number of vector outbound trombone", corresponding to scenario 2 of experiment 1

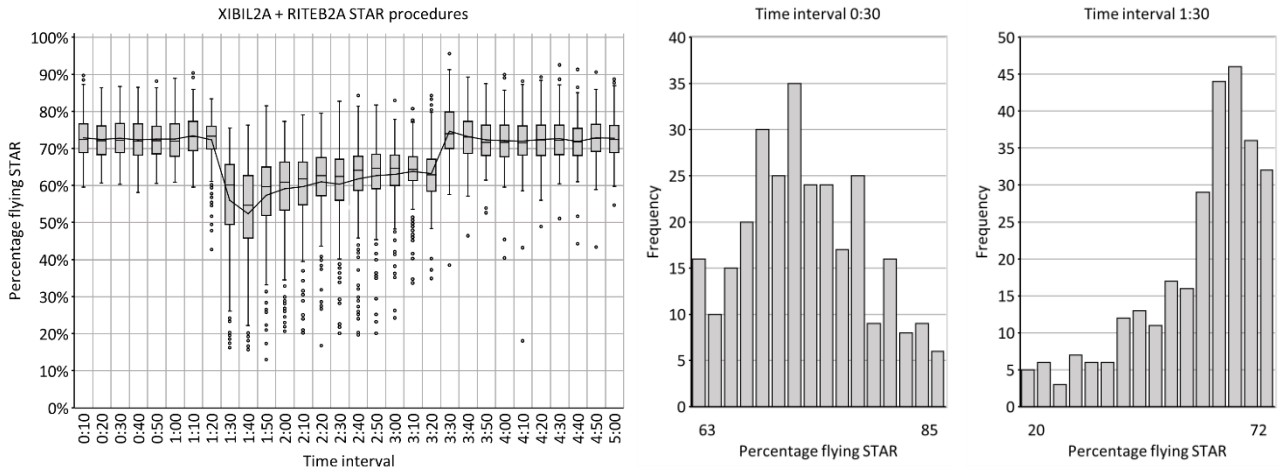


Simulation results of performance indicator "Number of vector outbound trombone", corresponding to scenario 3 of experiment 1

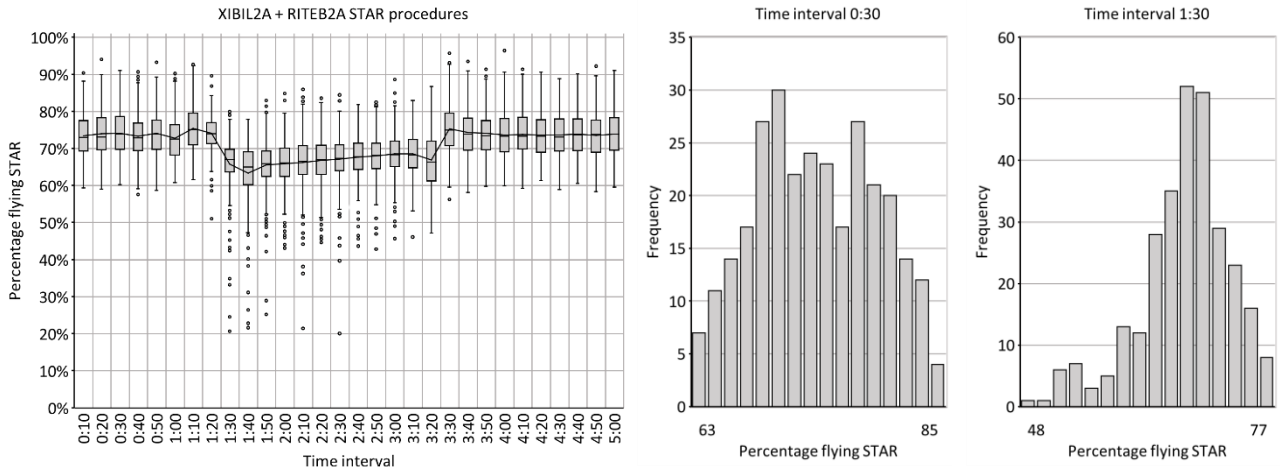
Percentage flying STAR – Experiment 1 – Scenarios 1, 2, 3



Simulation results of performance indicator "Percentage flying STAR", corresponding to scenario 1 of experiment 1

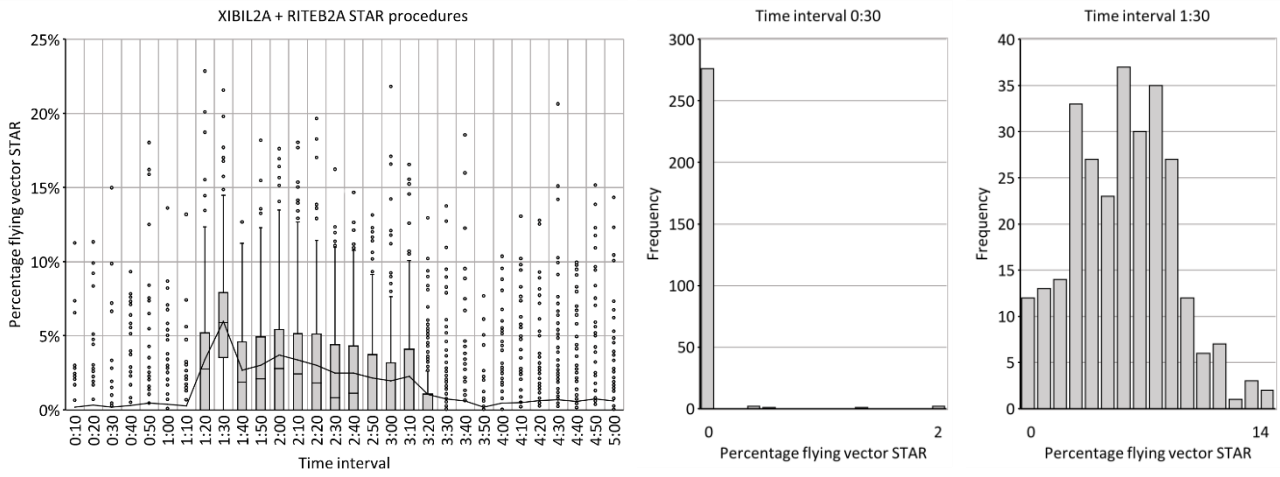


Simulation results of performance indicator "Percentage flying STAR", corresponding to scenario 2 of experiment 1

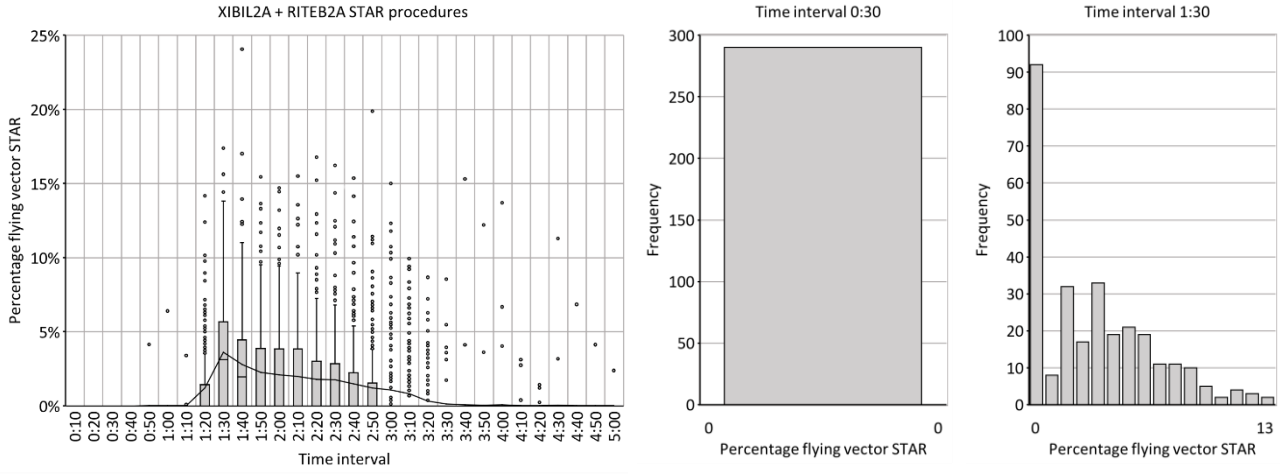


Simulation results of performance indicator "Percentage flying STAR", corresponding to scenario 3 of experiment 1

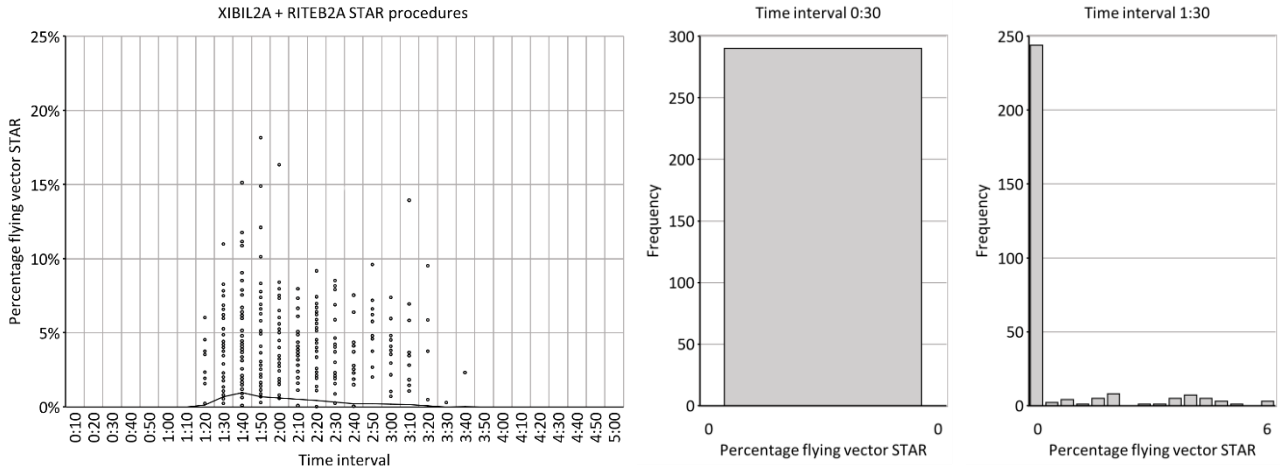
Percentage flying vector STAR – Experiment 1 – Scenarios 1, 2, 3



Simulation results of performance indicator "Percentage flying vector STAR", corresponding to scenario 1 of experiment 1

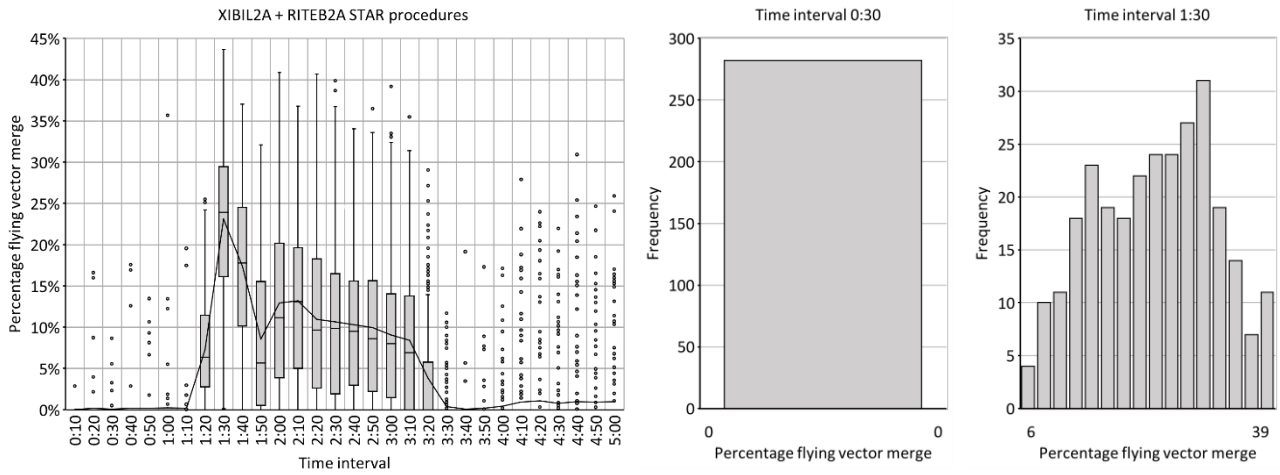


Simulation results of performance indicator "Percentage flying vector STAR", corresponding to scenario 2 of experiment 1

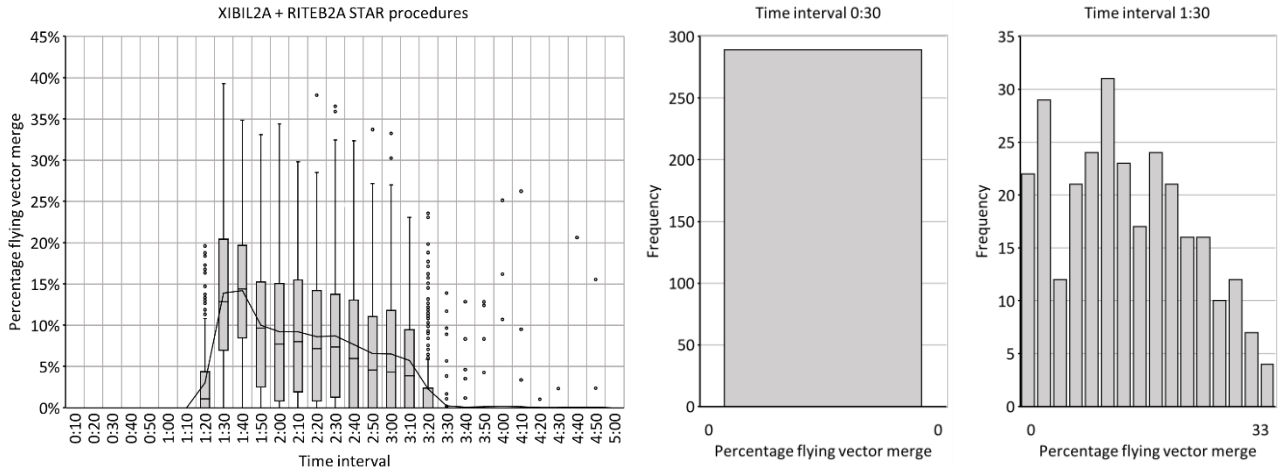


Simulation results of performance indicator "Percentage flying vector STAR", corresponding to scenario 3 of experiment 1

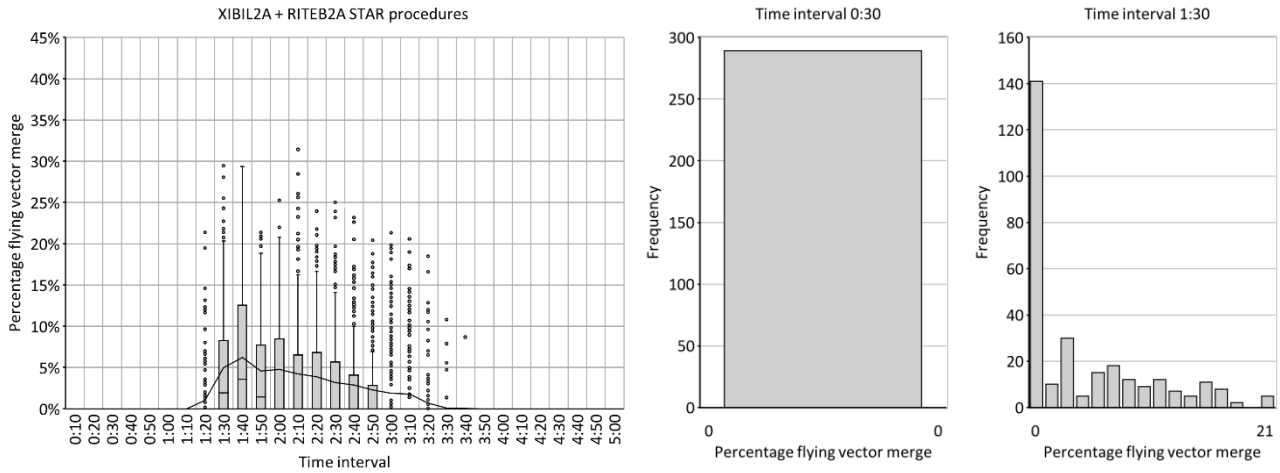
Percentage flying vector merge – Experiment 1 – Scenarios 1, 2, 3



Simulation results of performance indicator "Percentage flying vector merge", corresponding to scenario 1 of experiment 1

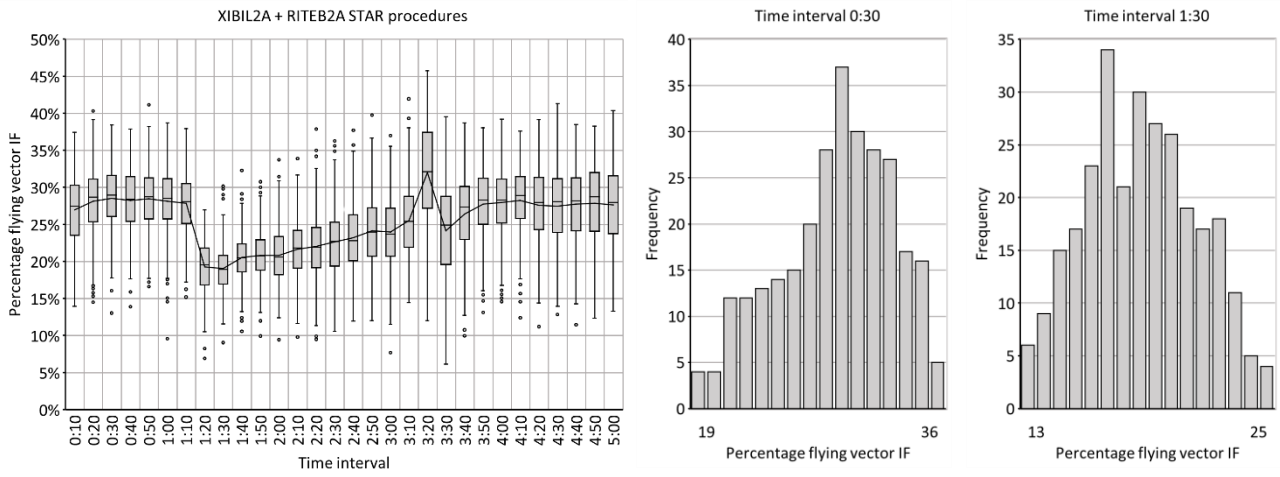


Simulation results of performance indicator "Percentage flying vector merge", corresponding to scenario 2 of experiment 1

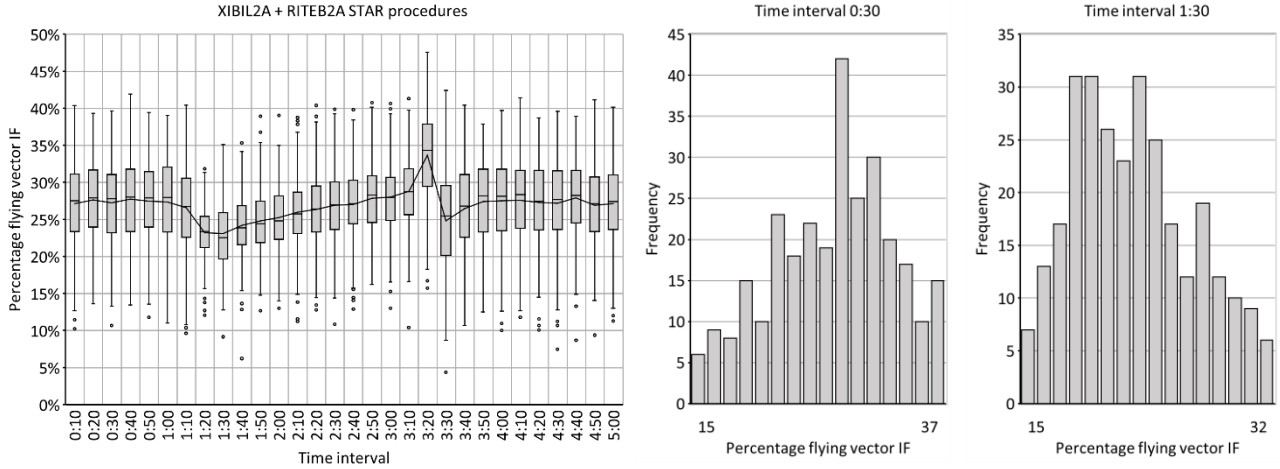


Simulation results of performance indicator "Percentage flying vector merge", corresponding to scenario 3 of experiment 1

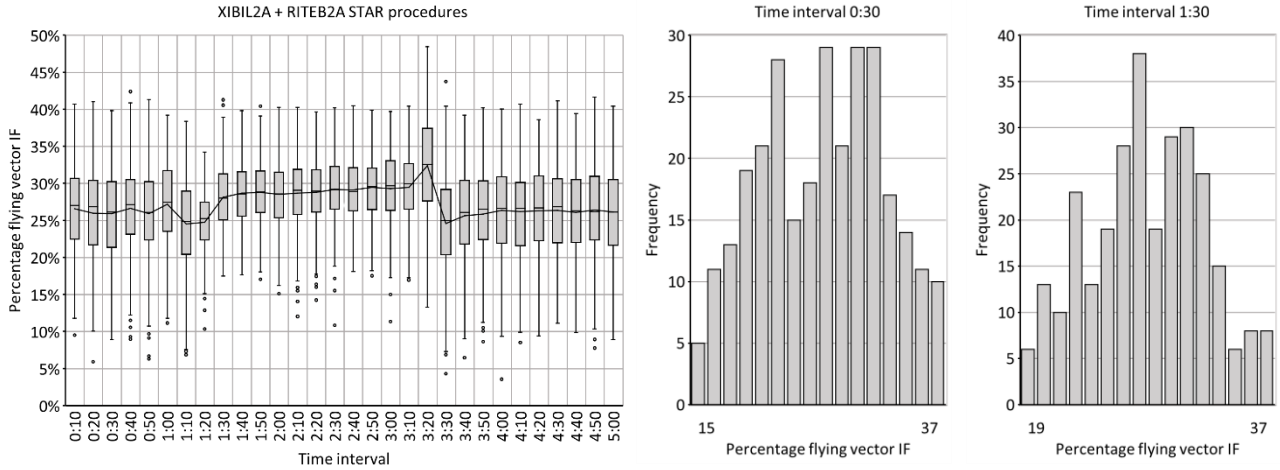
Percentage flying vector IF – Experiment 1 – Scenarios 1, 2, 3



Simulation results of performance indicator "Percentage flying vector IF", corresponding to scenario 1 of experiment 1

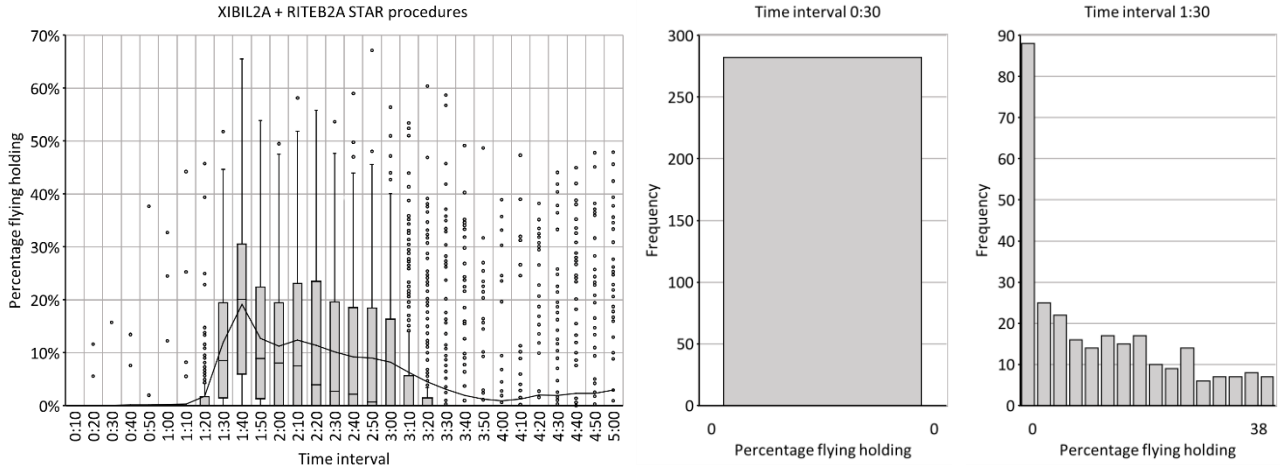


Simulation results of performance indicator "Percentage flying vector IF", corresponding to scenario 2 of experiment 1

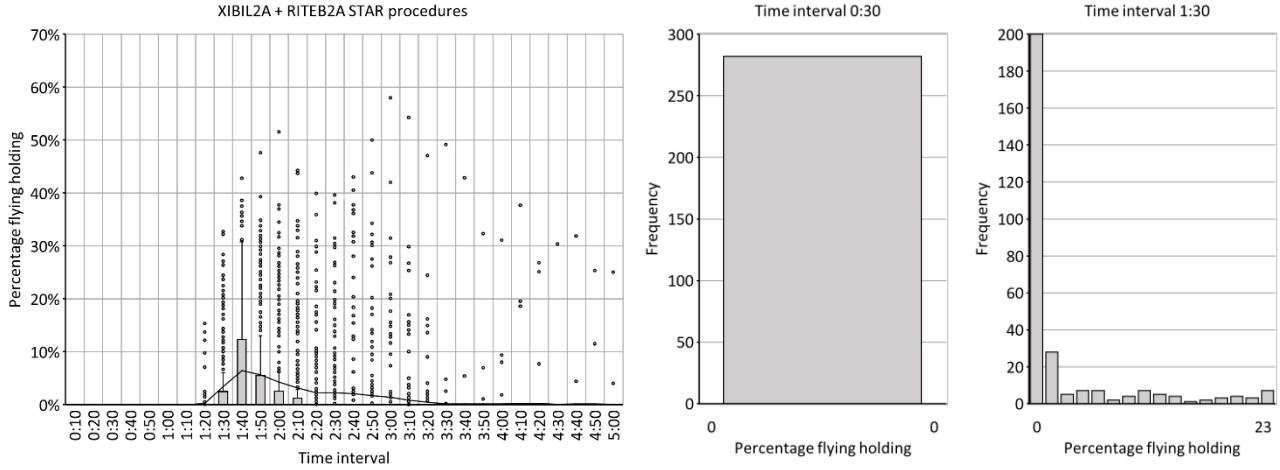


Simulation results of performance indicator "Percentage flying vector IF", corresponding to scenario 3 of experiment 1

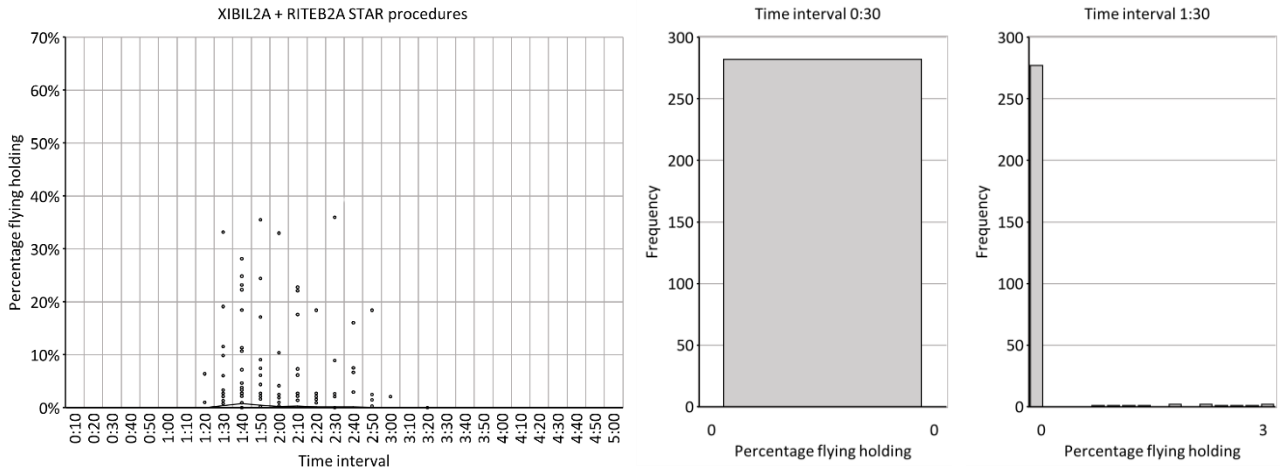
Percentage flying holding – Experiment 1 – Scenarios 1, 2, 3



Simulation results of performance indicator "Percentage flying holding", corresponding to scenario 1 of experiment 1

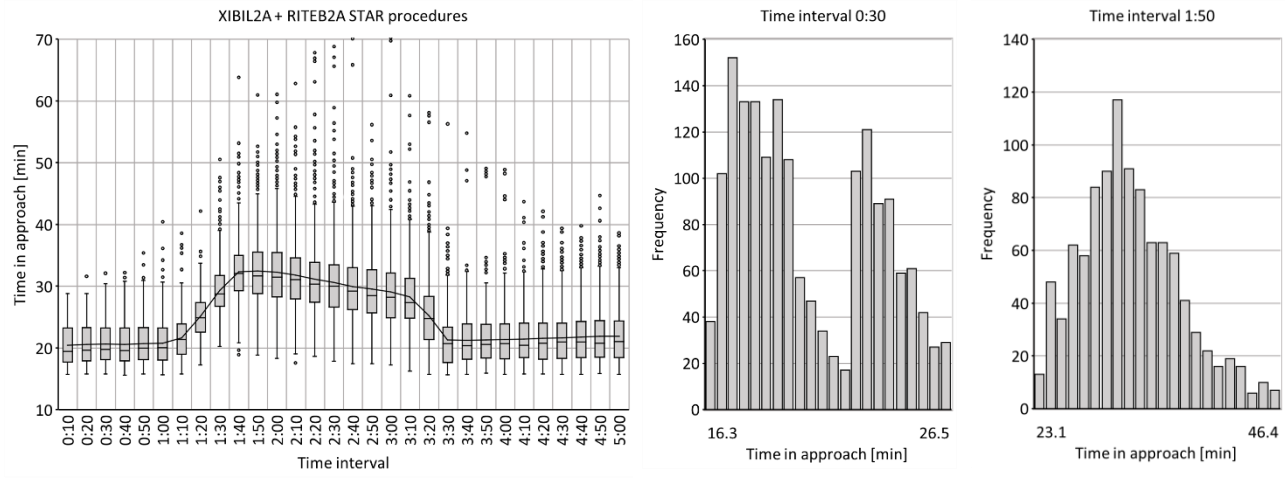


Simulation results of performance indicator "Percentage flying holding", corresponding to scenario 2 of experiment 1

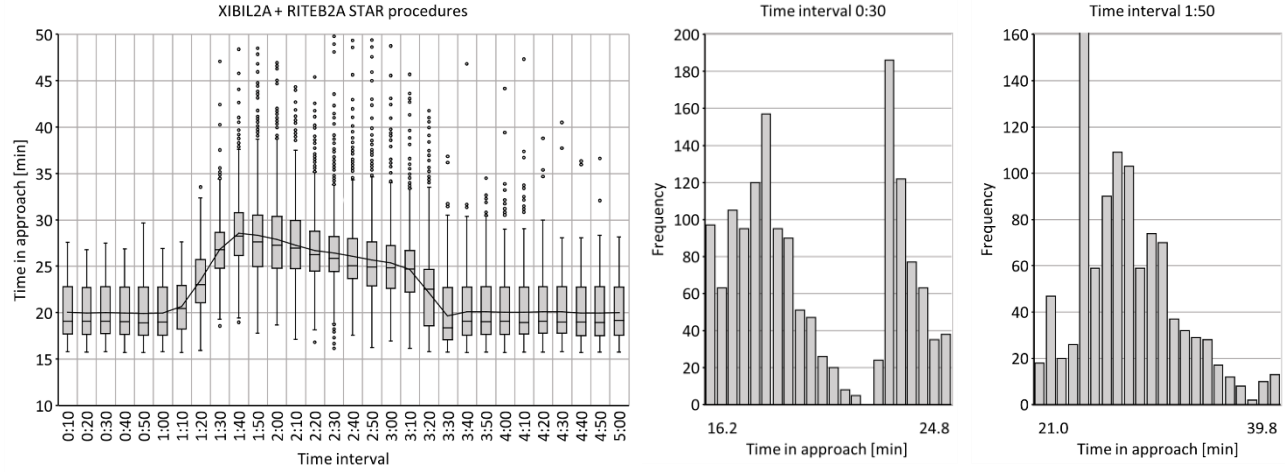


Simulation results of performance indicator "Percentage flying holding", corresponding to scenario 3 of experiment 1

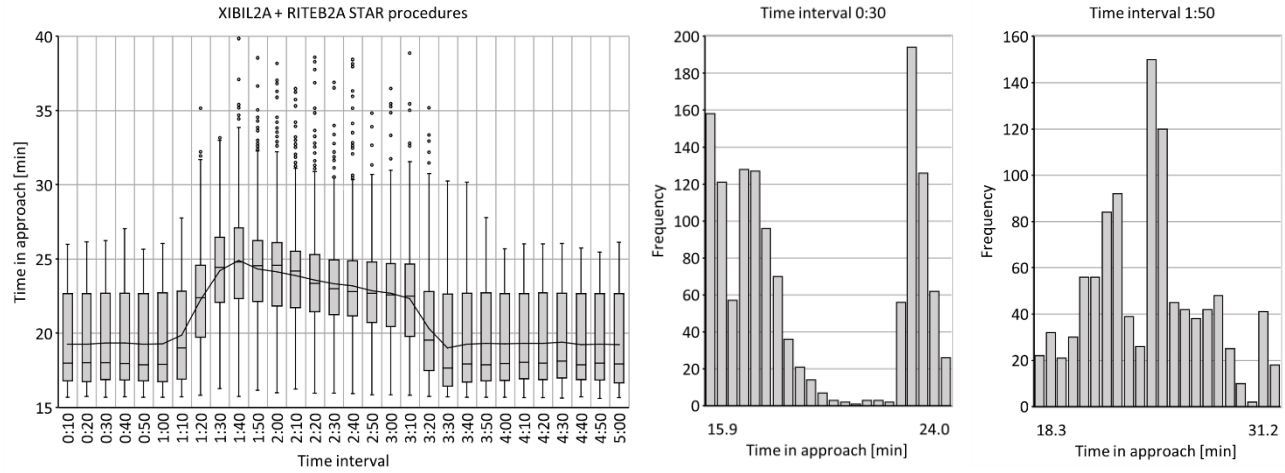
Time in approach – Experiment 1 – Scenarios 1, 2, 3



Simulation results of performance indicator "Time in approach", corresponding to scenario 1 of experiment 1

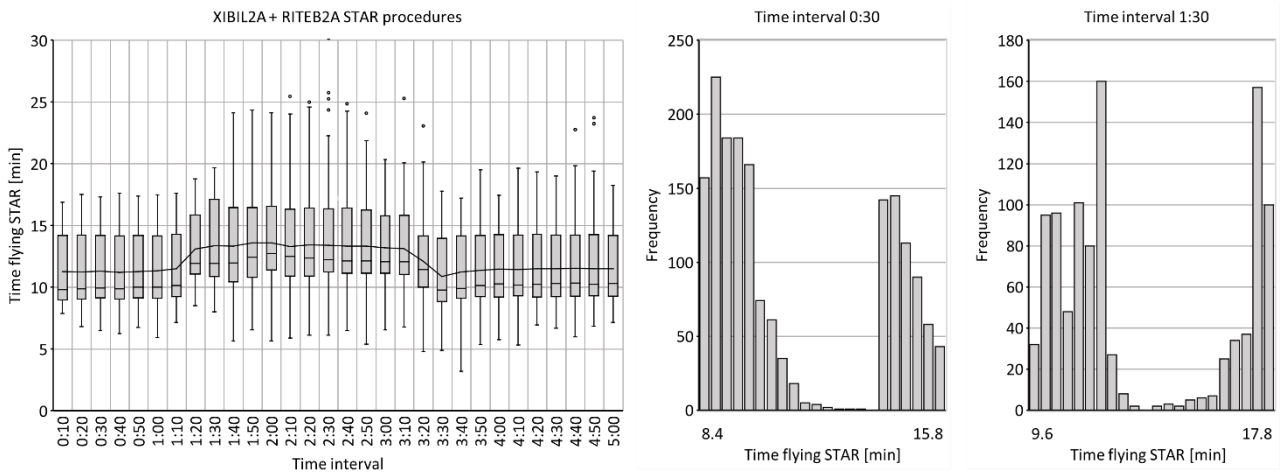


Simulation results of performance indicator "Time in approach", corresponding to scenario 2 of experiment 1

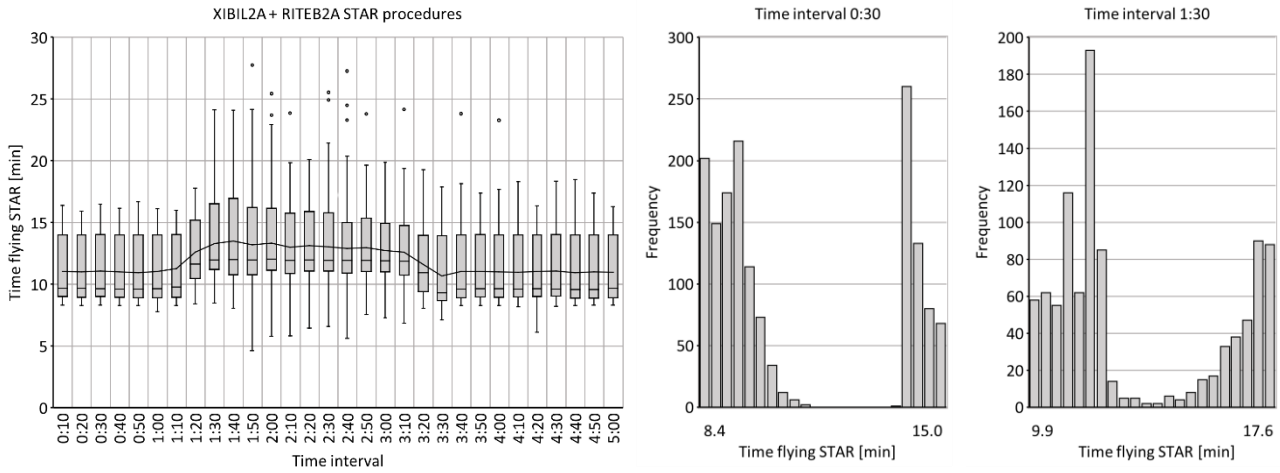


Simulation results of performance indicator "Time in approach", corresponding to scenario 3 of experiment 1

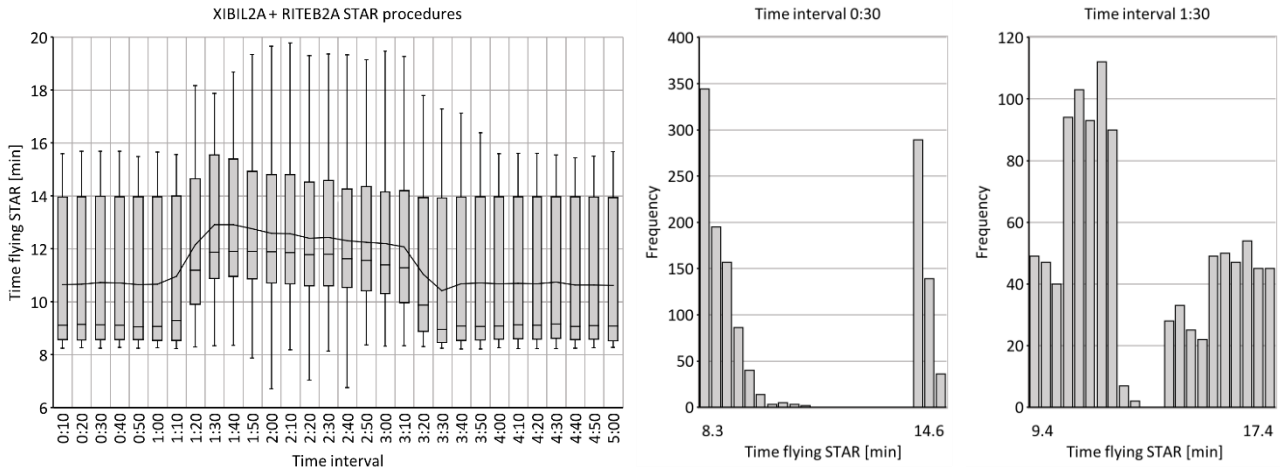
Time flying STAR – Experiment 1 – Scenarios 1, 2, 3



Simulation results of performance indicator "Time flying STAR", corresponding to scenario 1 of experiment 1

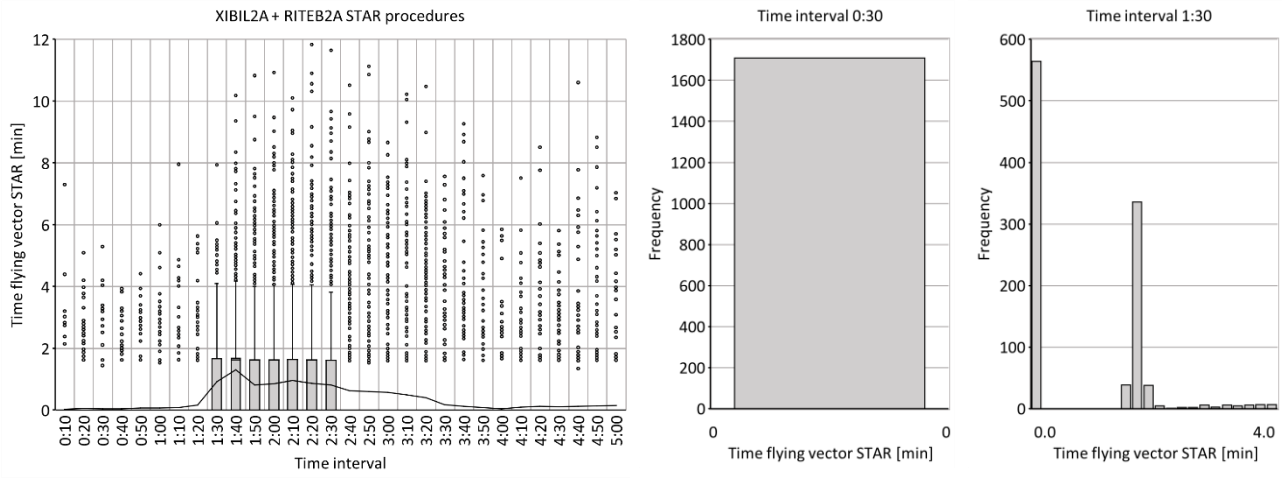


Simulation results of performance indicator "Time flying STAR", corresponding to scenario 2 of experiment 1

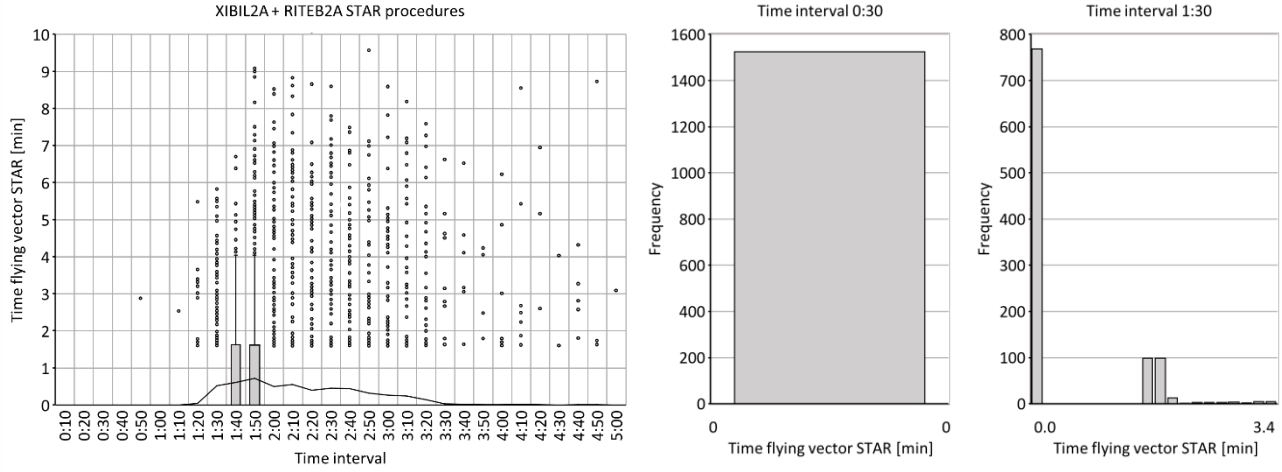


Simulation results of performance indicator "Time flying STAR", corresponding to scenario 3 of experiment 1

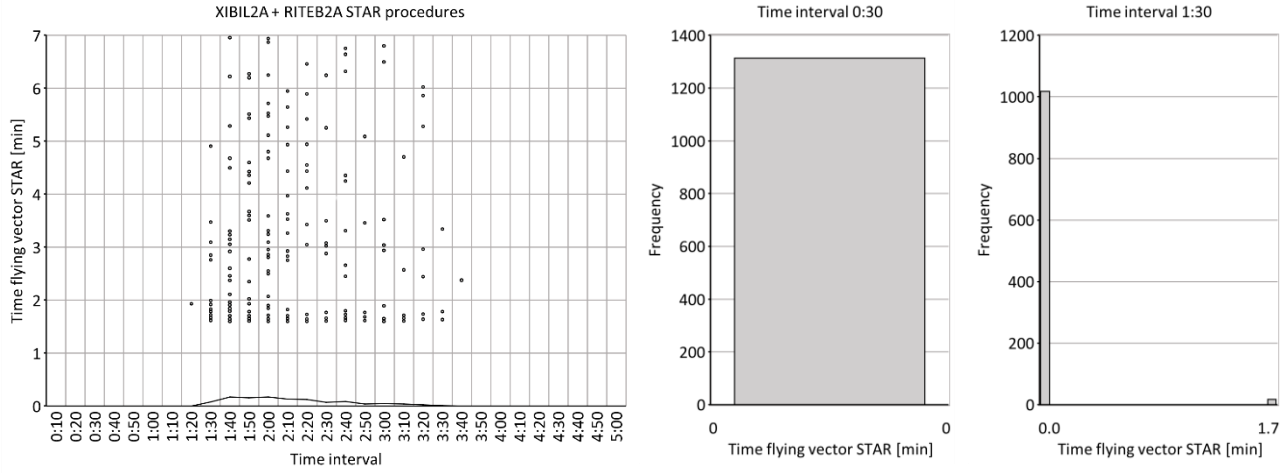
Time flying vector STAR – Experiment 1 – Scenarios 1, 2, 3



Simulation results of performance indicator "Time flying vector STAR", corresponding to scenario 1 of experiment 1

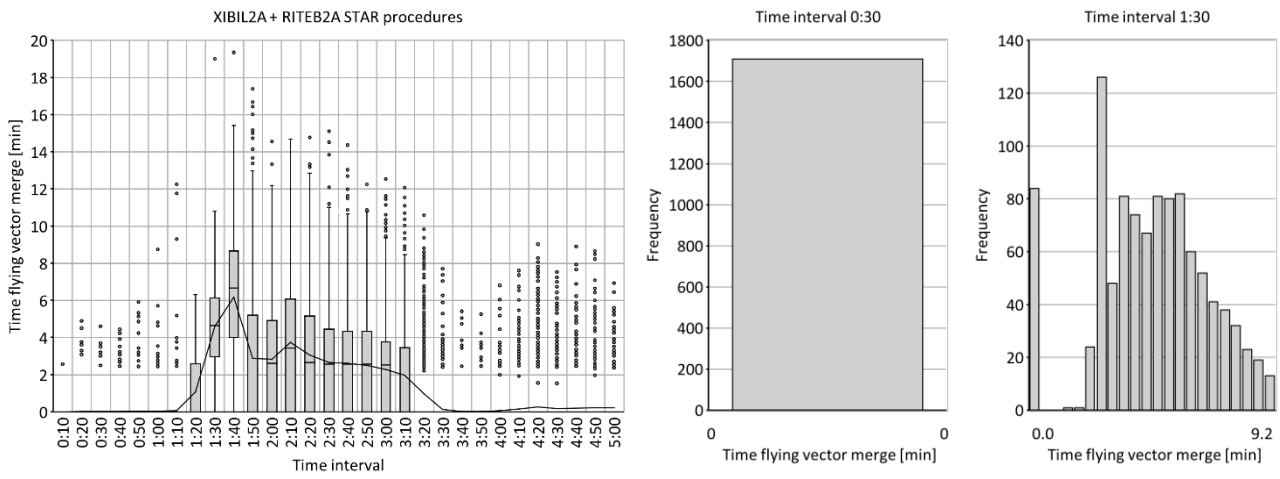


Simulation results of performance indicator "Time flying vector STAR", corresponding to scenario 2 of experiment 1

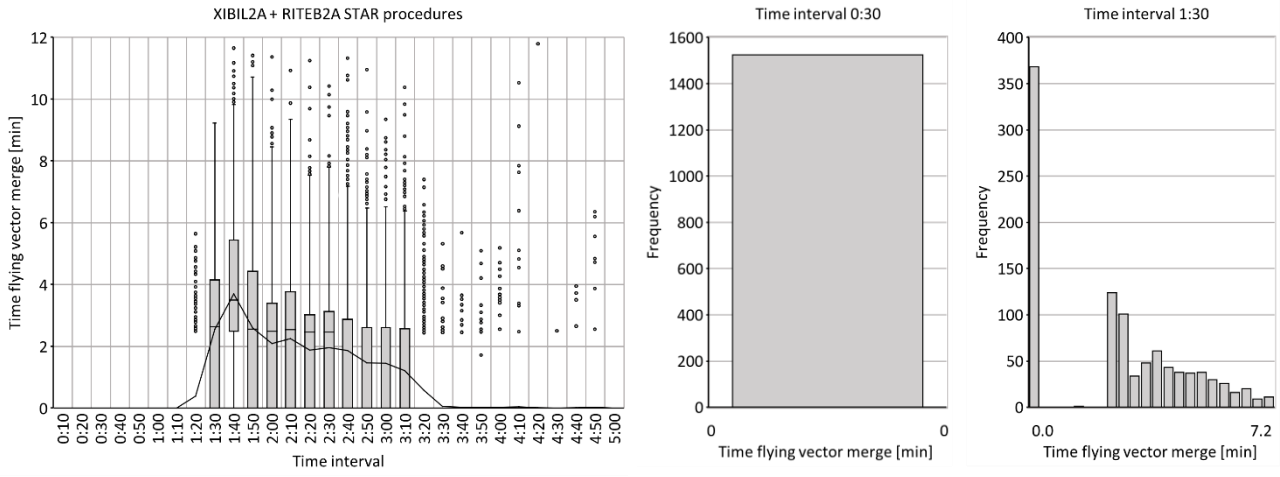


Simulation results of performance indicator "Time flying vector STAR", corresponding to scenario 3 of experiment 1

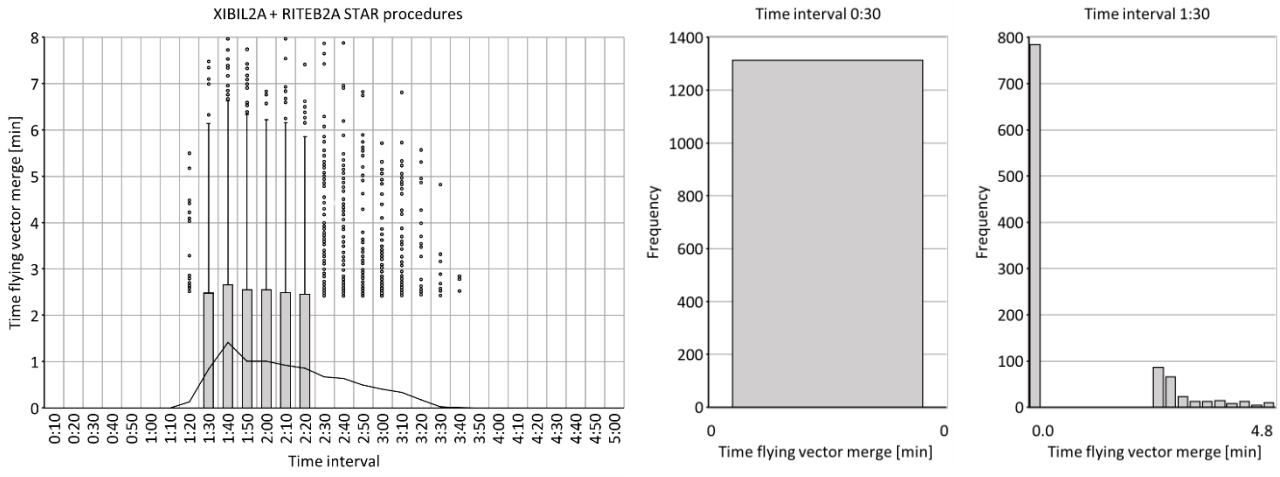
Time flying vector merge – Experiment 1 – Scenarios 1, 2, 3



Simulation results of performance indicator "Time flying vector merge", corresponding to scenario 1 of experiment 1

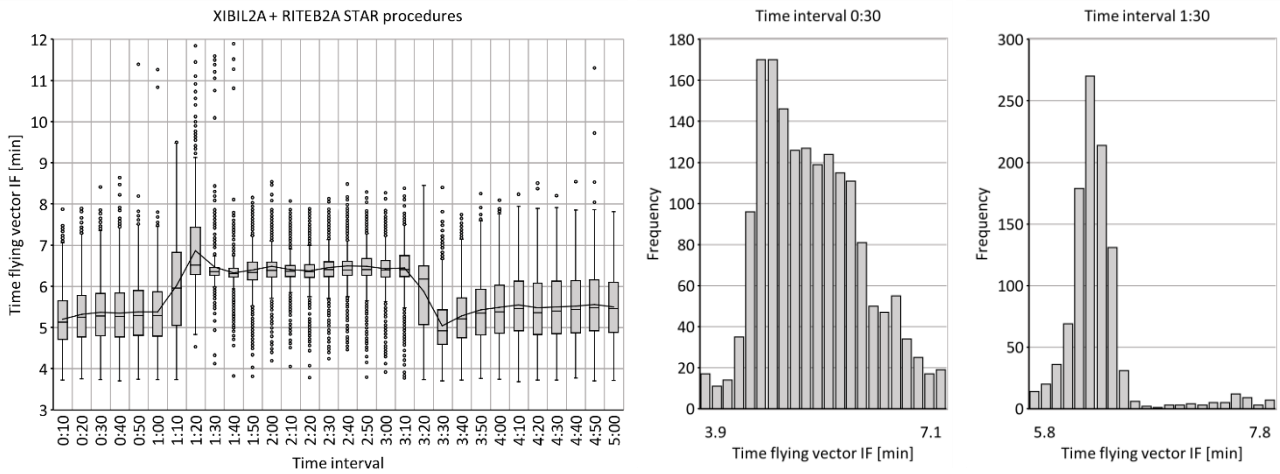


Simulation results of performance indicator "Time flying vector merge", corresponding to scenario 2 of experiment 1

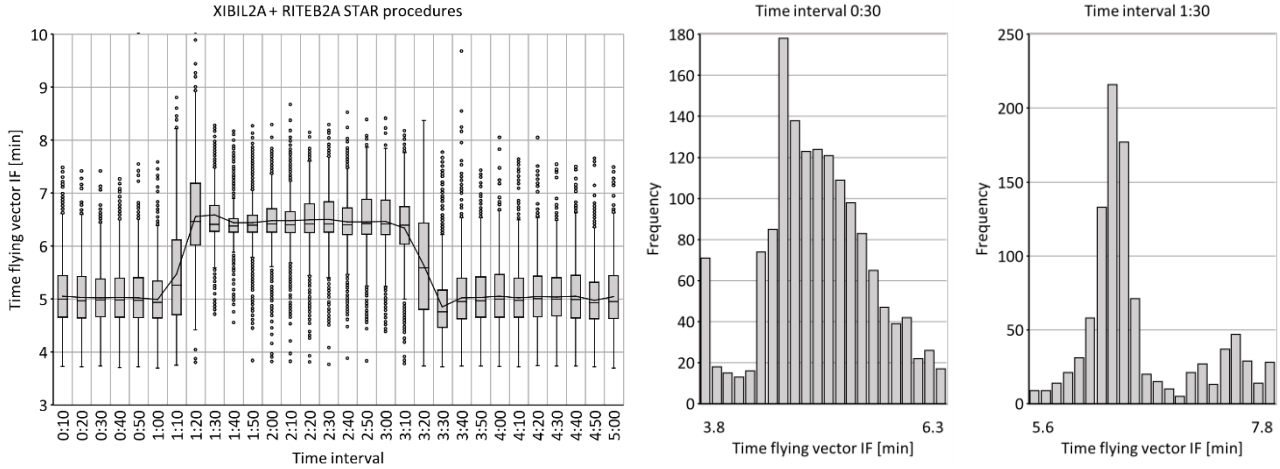


Simulation results of performance indicator "Time flying vector merge", corresponding to scenario 3 of experiment 1

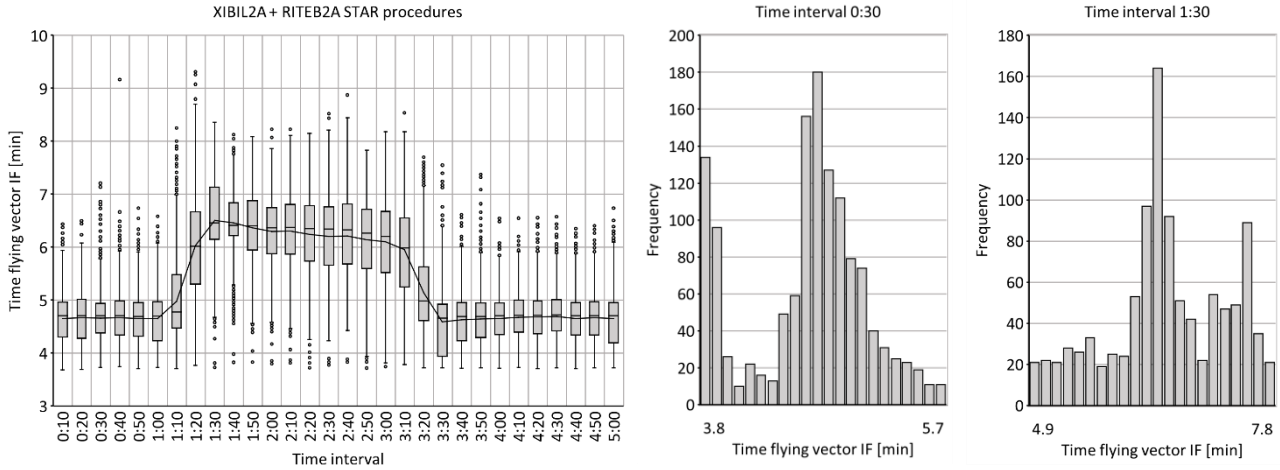
Time flying vector IF – Experiment 1 – Scenarios 1, 2, 3



Simulation results of performance indicator "Time flying vector IF", corresponding to scenario 1 of experiment 1

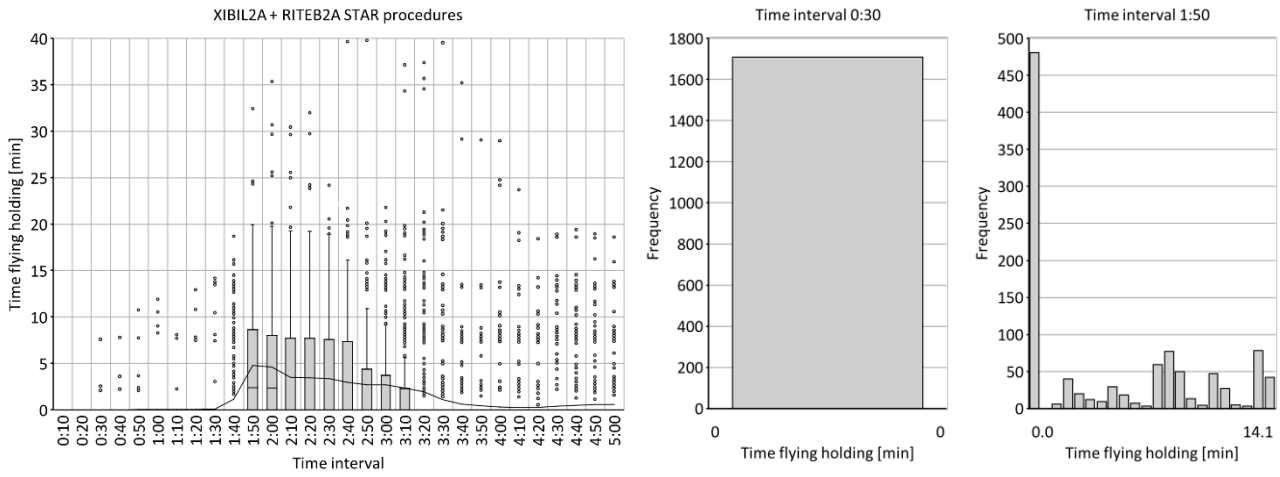


Simulation results of performance indicator "Time flying vector IF", corresponding to scenario 2 of experiment 1

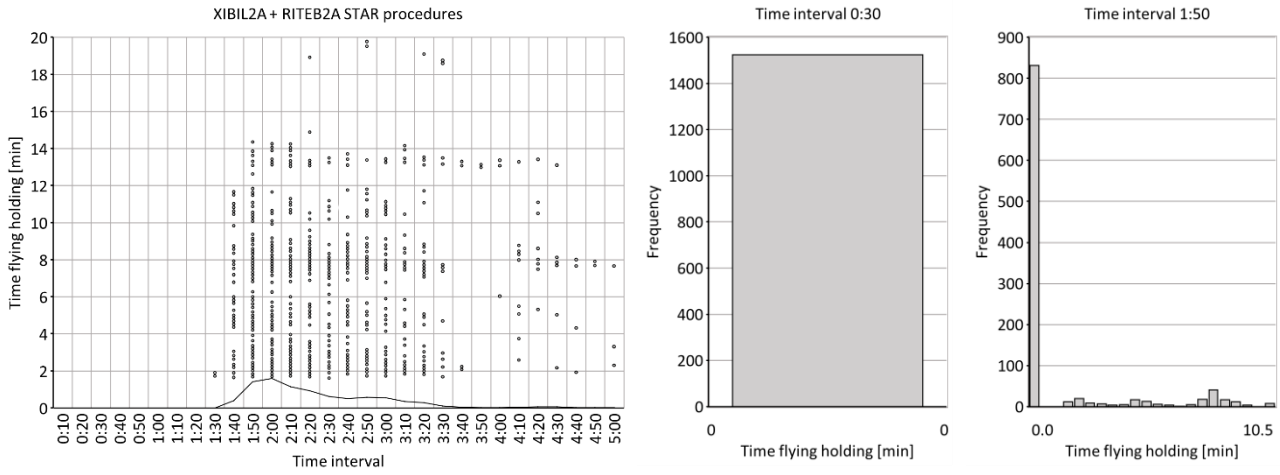


Simulation results of performance indicator "Time flying vector IF", corresponding to scenario 3 of experiment 1

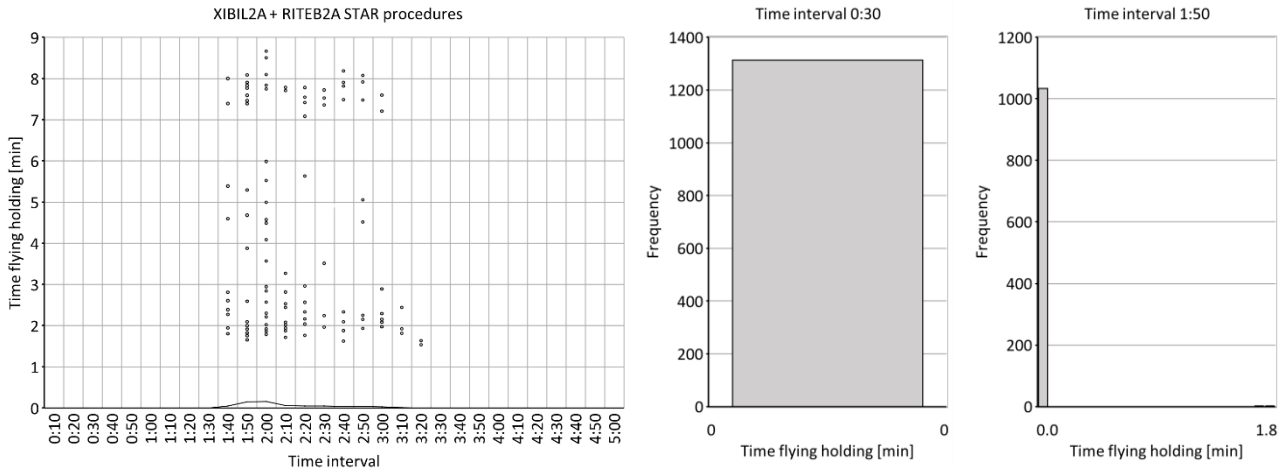
Time flying holding – Experiment 1 – Scenarios 1, 2, 3



Simulation results of performance indicator "Time flying holding", corresponding to scenario 1 of experiment 1

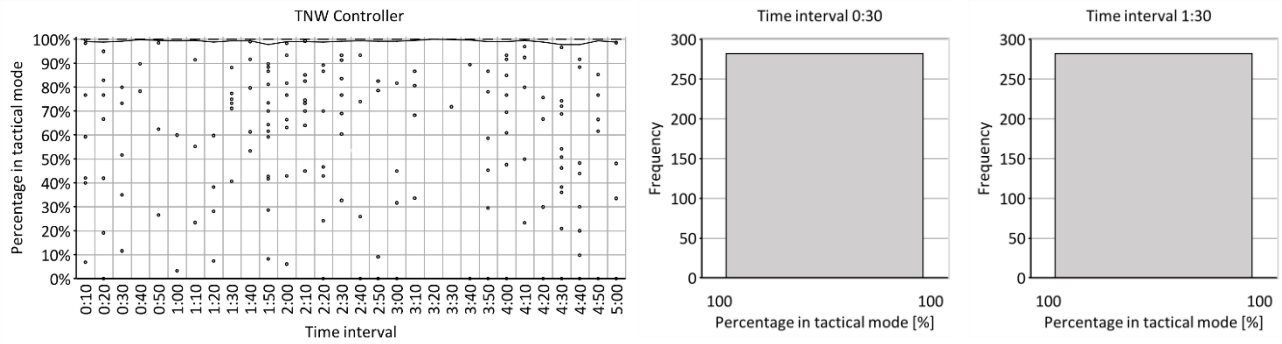


Simulation results of performance indicator "Time flying holding", corresponding to scenario 2 of experiment 1

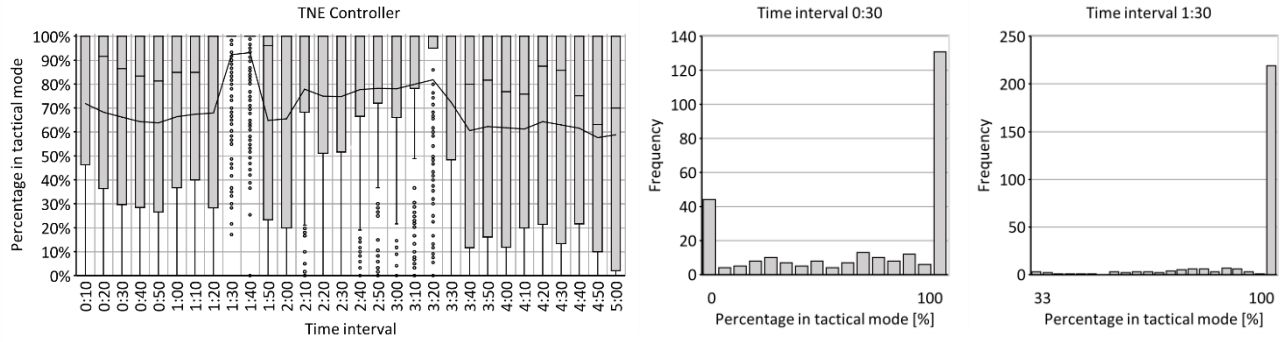


Simulation results of performance indicator "Time flying holding", corresponding to scenario 3 of experiment 1

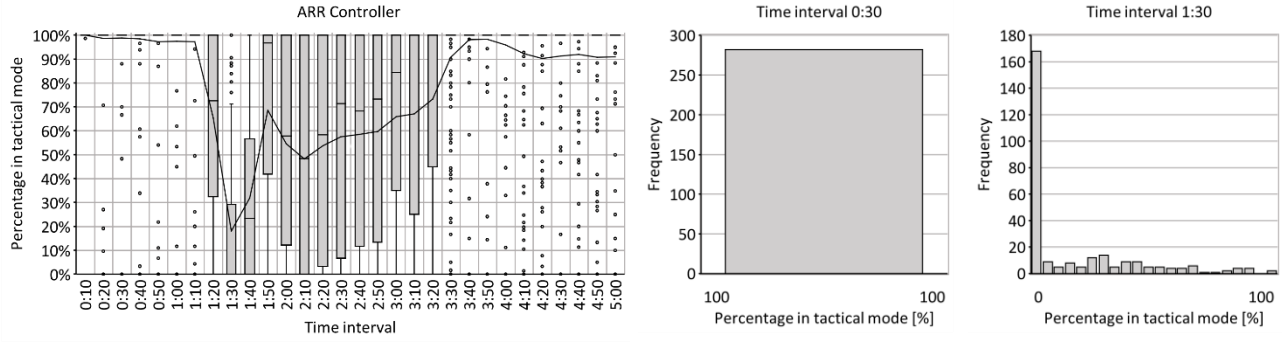
Percentage in tactical mode – Experiment 1 – Scenario 1



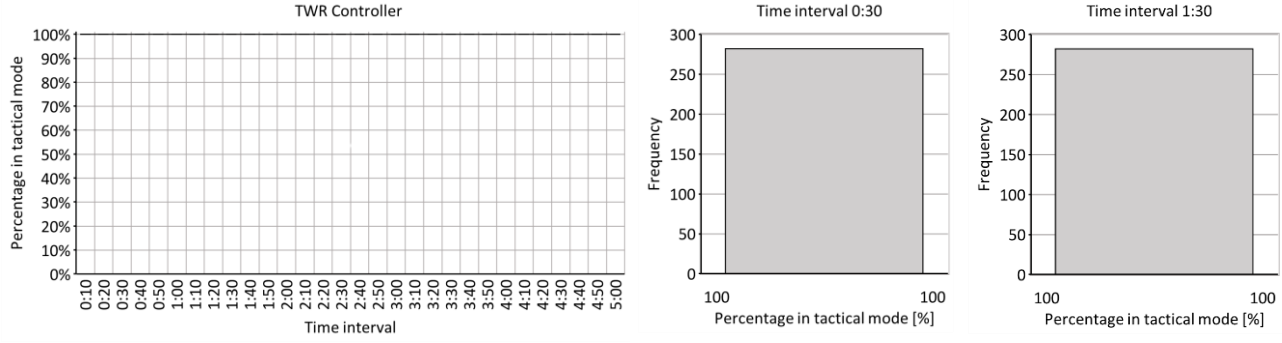
Simulation results of performance indicator "Percentage in tactical mode (TNW controller)", corresponding to scenario 1 of experiment 1



Simulation results of performance indicator "Percentage in tactical mode (TNE controller)", corresponding to scenario 1 of experiment 1

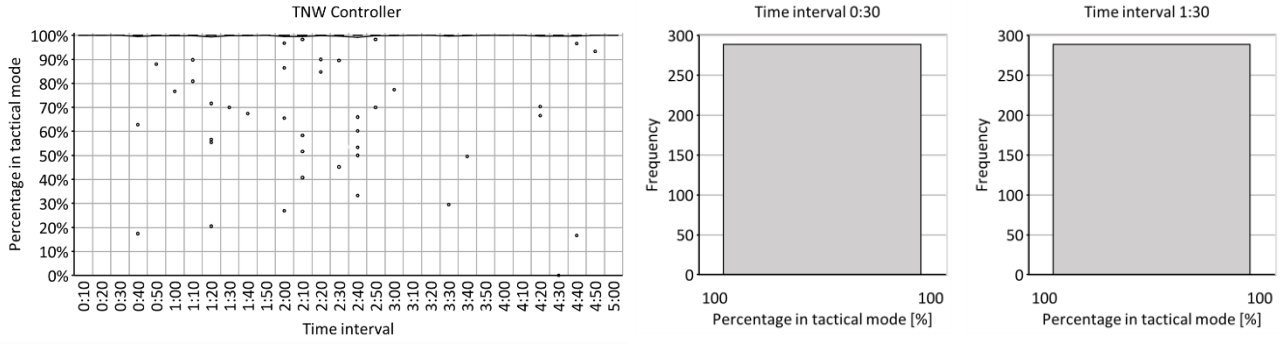


Simulation results of performance indicator "Percentage in tactical mode (ARR controller)", corresponding to scenario 1 of experiment 1

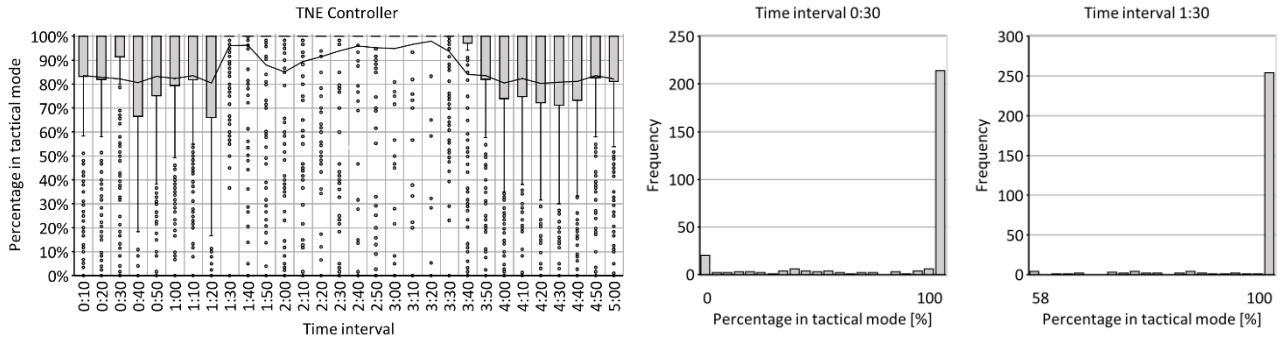


Simulation results of performance indicator "Percentage in tactical mode (TWR controller)", corresponding to scenario 1 of experiment 1

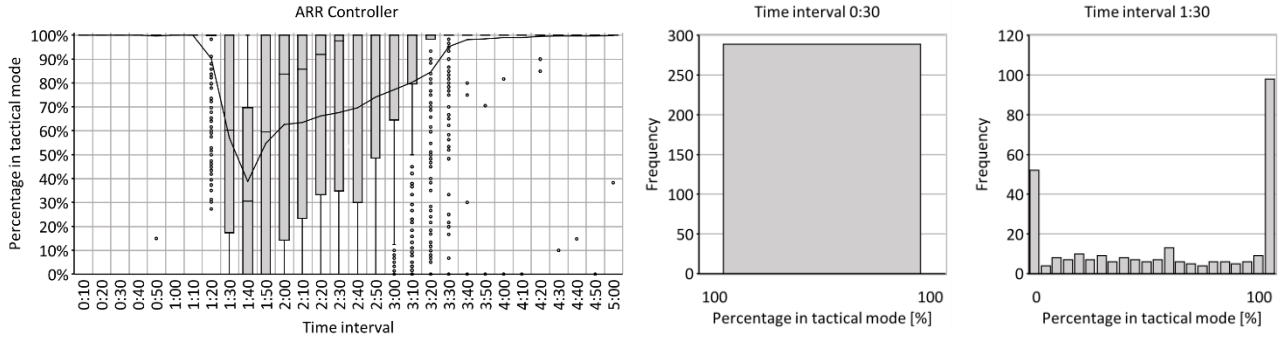
Percentage in tactical mode – Experiment 1 – Scenario 2



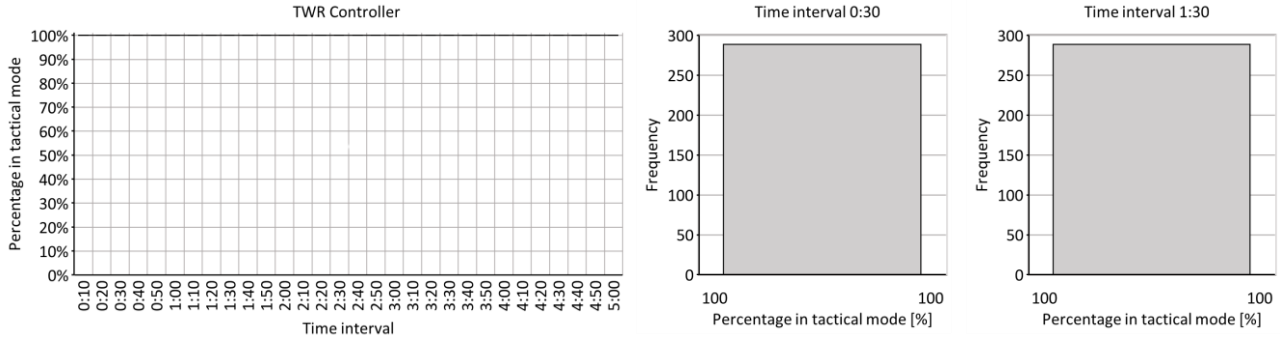
Simulation results of performance indicator "Percentage in tactical mode (TNW controller)", corresponding to scenario 2 of experiment 1



Simulation results of performance indicator "Percentage in tactical mode (TNE controller)", corresponding to scenario 2 of experiment 1

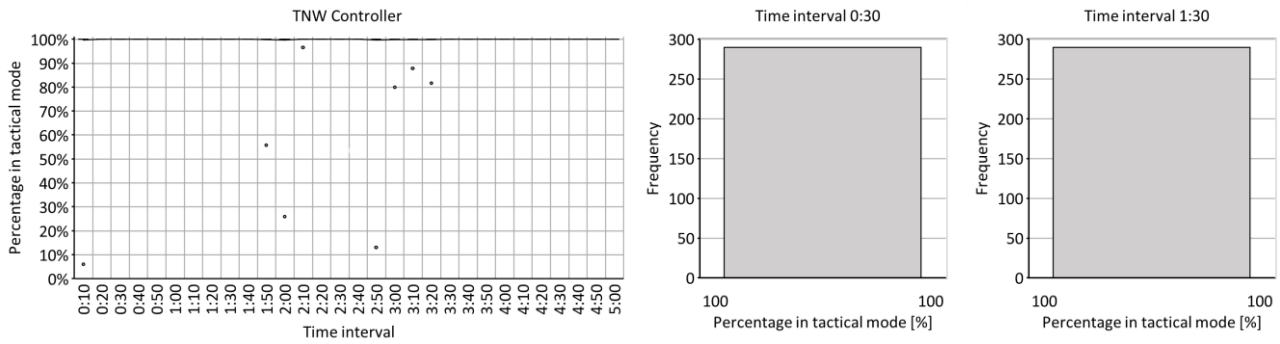


Simulation results of performance indicator "Percentage in tactical mode (ARR controller)", corresponding to scenario 2 of experiment 1

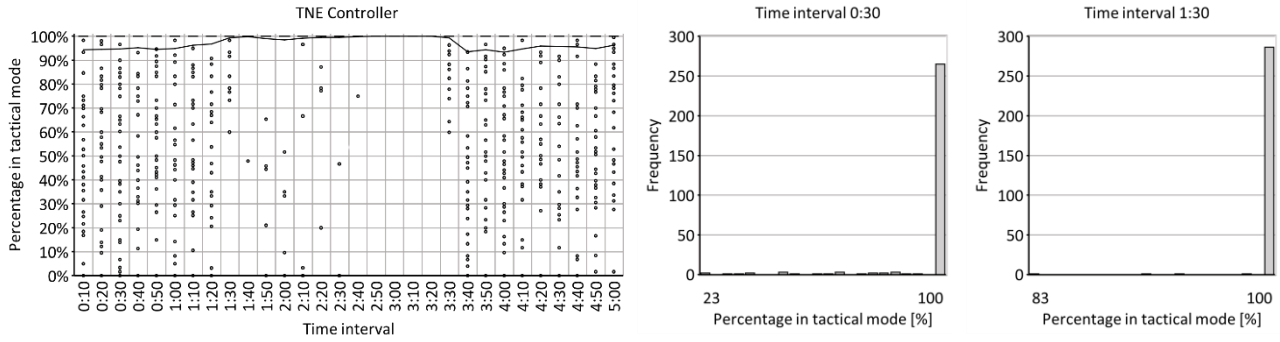


Simulation results of performance indicator "Percentage in tactical mode (TWR controller)", corresponding to scenario 2 of experiment 1

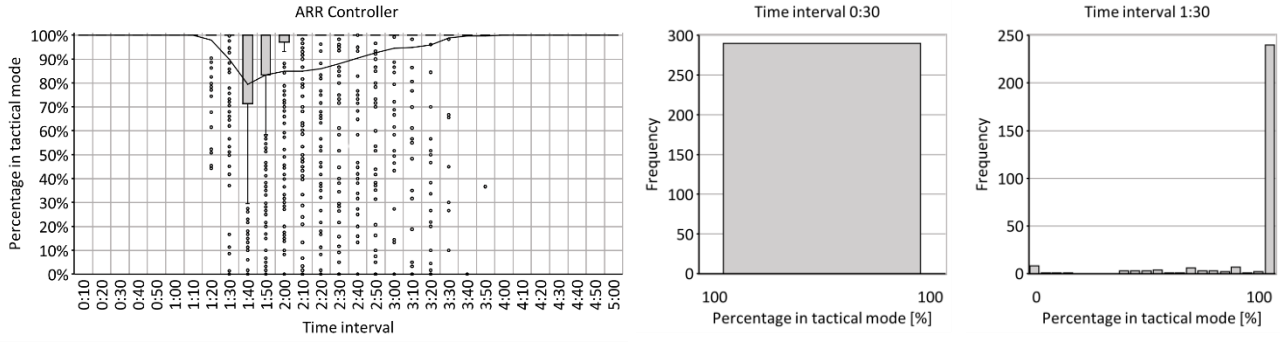
Percentage in tactical mode – Experiment 1 – Scenario 3



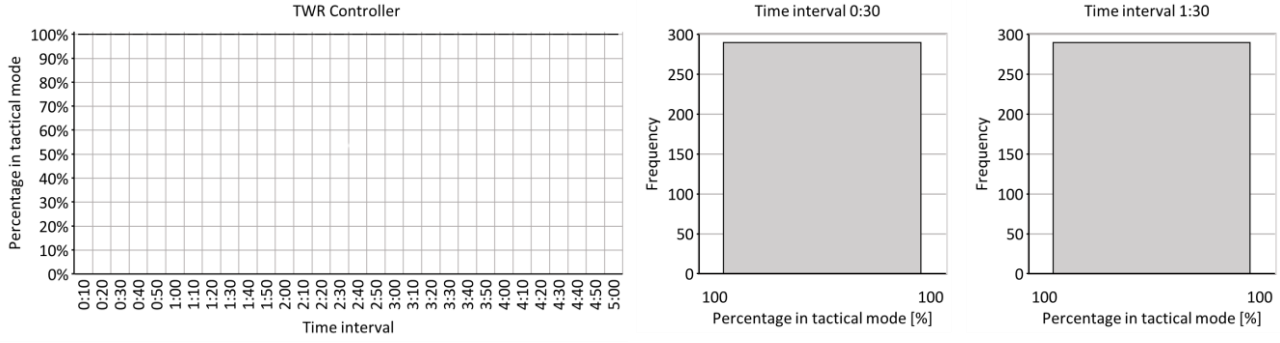
Simulation results of performance indicator "Percentage in tactical mode (TNW controller)", corresponding to scenario 3 of experiment 1



Simulation results of performance indicator "Percentage in tactical mode (TNE controller)", corresponding to scenario 3 of experiment 1

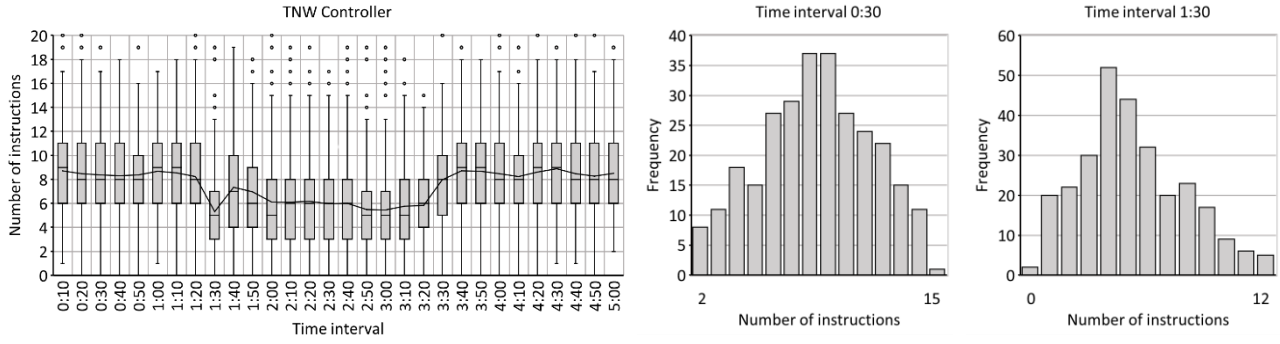


Simulation results of performance indicator "Percentage in tactical mode (ARR controller)", corresponding to scenario 3 of experiment 1

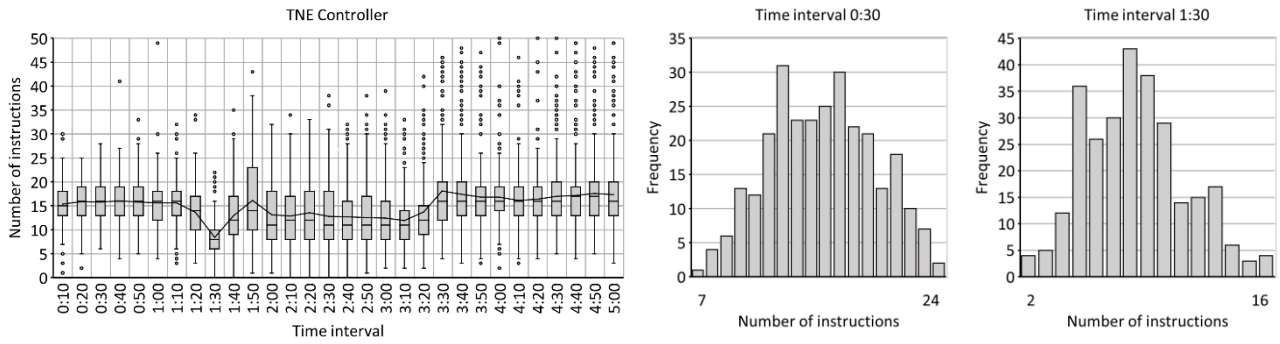


Simulation results of performance indicator "Percentage in tactical mode (TWR controller)", corresponding to scenario 3 of experiment 1

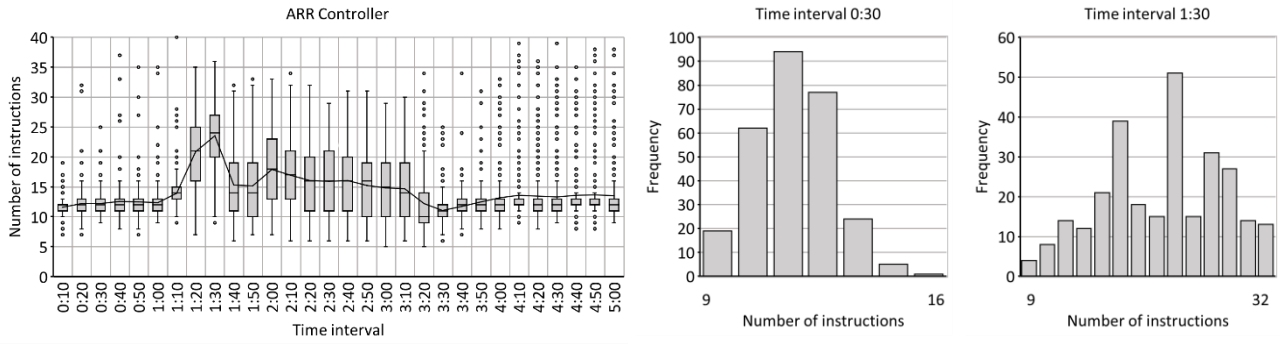
Number of instructions – Experiment 1 – Scenario 1



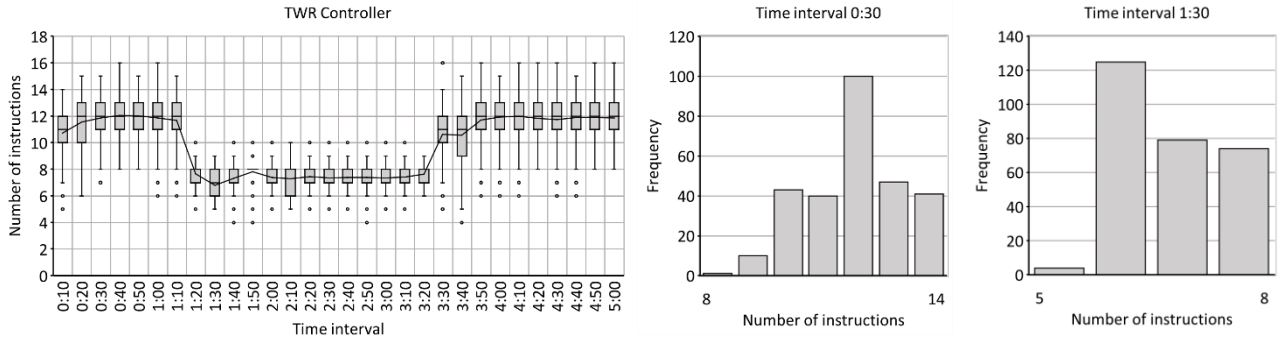
Simulation results of performance indicator "Number of instructions (TNW controller)", corresponding to scenario 1 of experiment 1



Simulation results of performance indicator "Number of instructions (TNE controller)", corresponding to scenario 1 of experiment 1

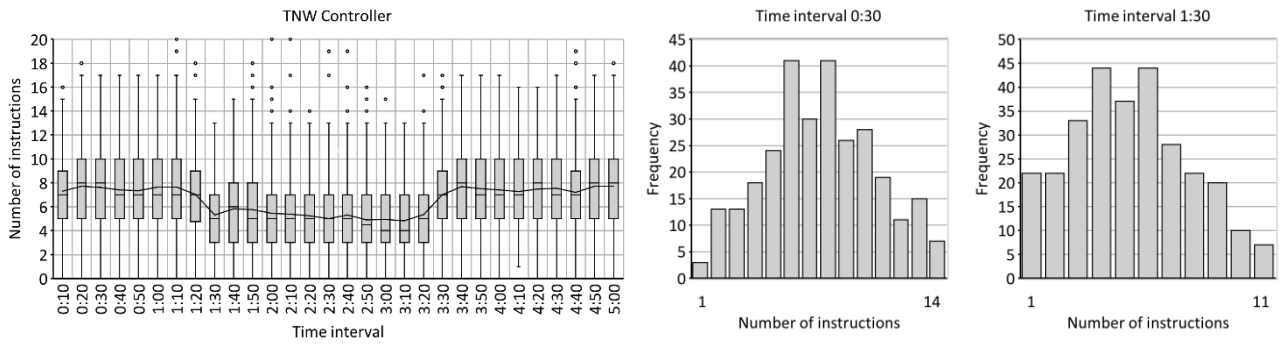


Simulation results of performance indicator "Number of instructions (ARR controller)", corresponding to scenario 1 of experiment 1

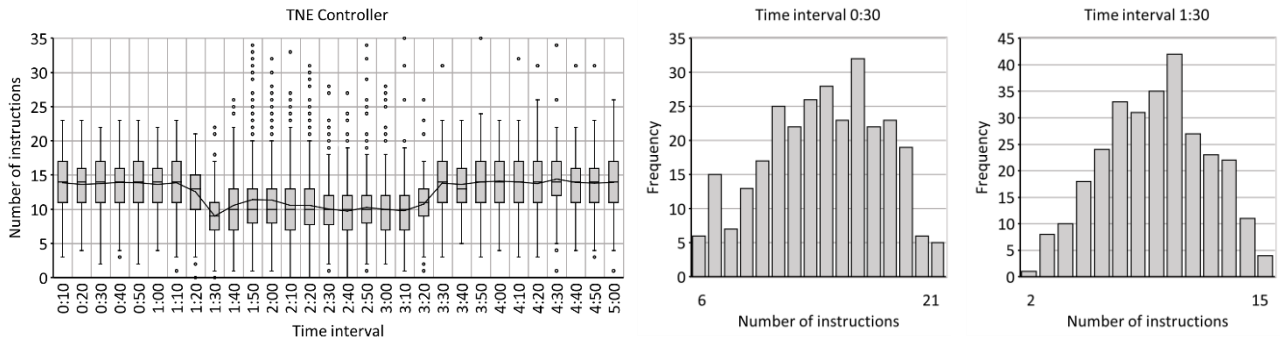


Simulation results of performance indicator "Number of instructions (TWR controller)", corresponding to scenario 1 of experiment 1

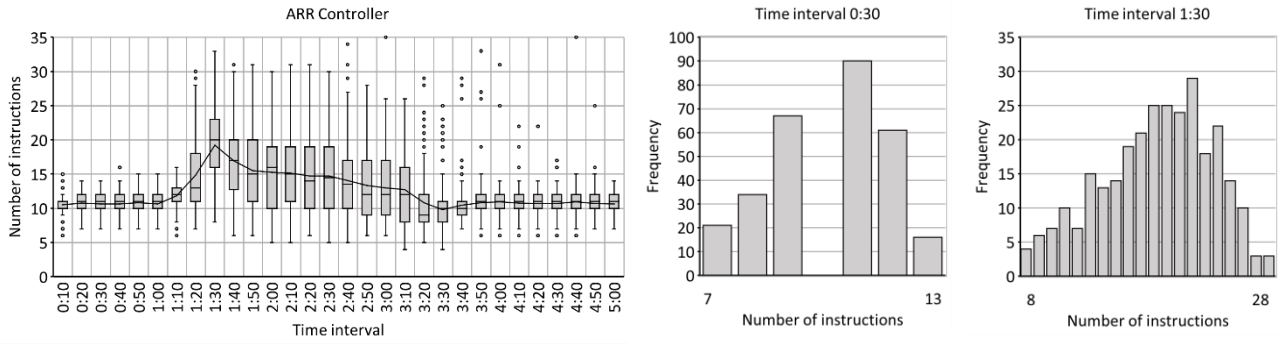
Number of instructions – Experiment 1 – Scenario 2



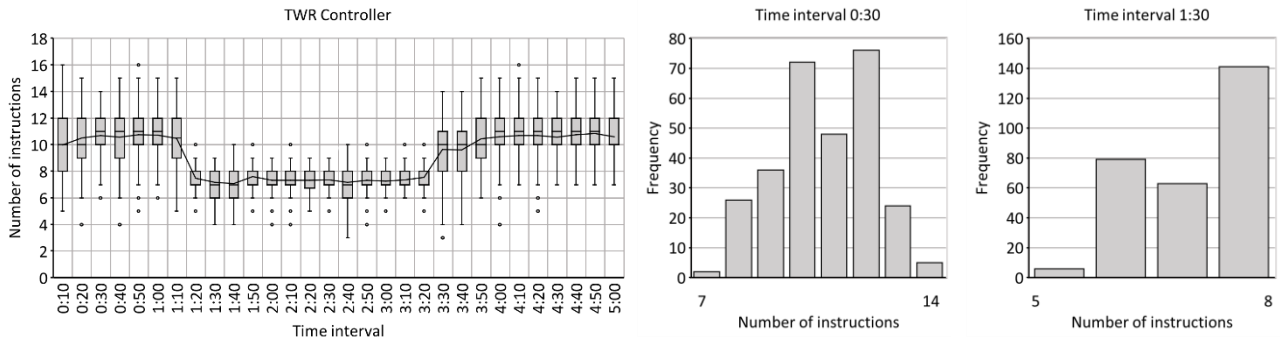
Simulation results of performance indicator "Number of instructions (TNW controller)", corresponding to scenario 2 of experiment 1



Simulation results of performance indicator "Number of instructions (TNE controller)", corresponding to scenario 2 of experiment 1

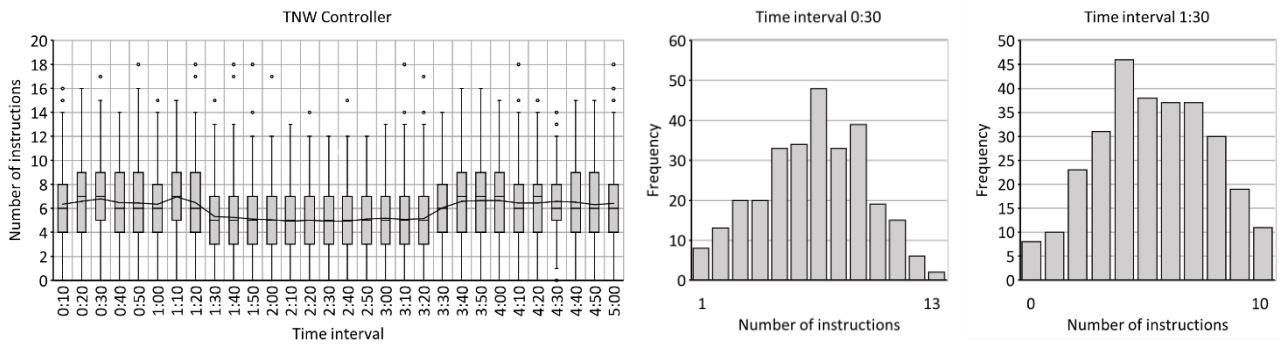


Simulation results of performance indicator "Number of instructions (ARR controller)", corresponding to scenario 2 of experiment 1

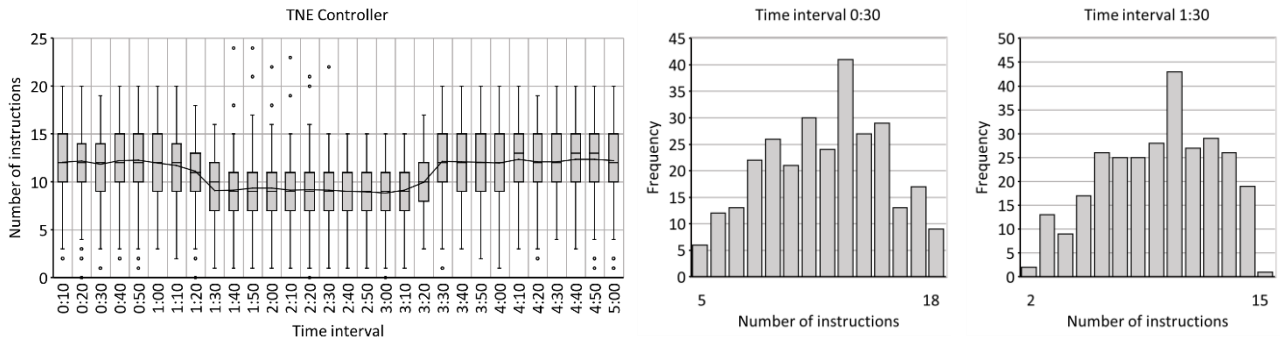


Simulation results of performance indicator "Number of instructions (TWR controller)", corresponding to scenario 2 of experiment 1

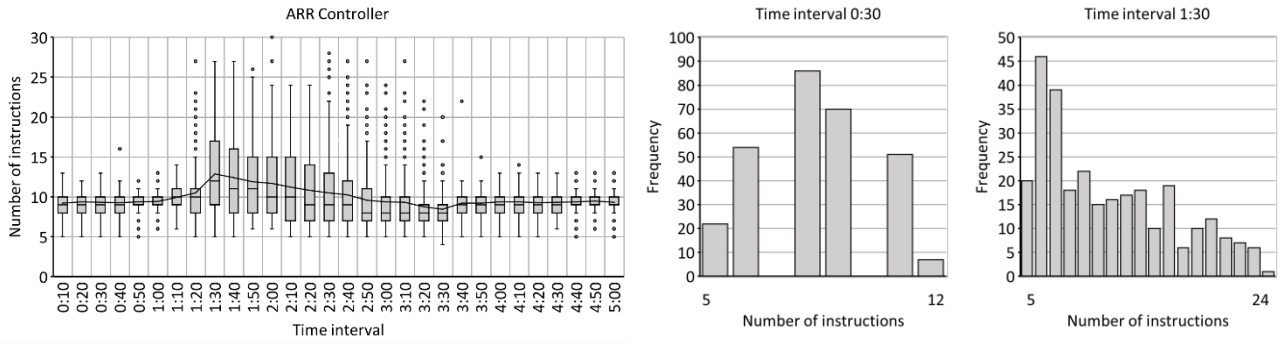
Number of instructions – Experiment 1 – Scenario 3



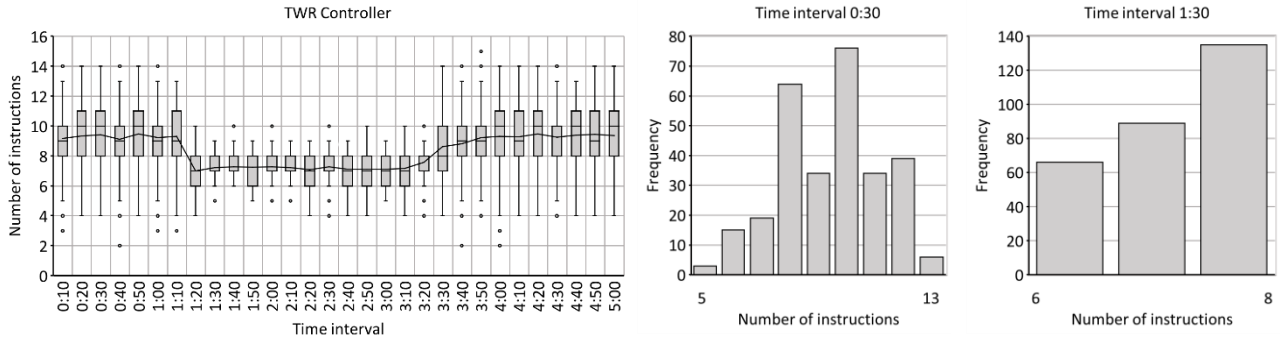
Simulation results of performance indicator "Number of instructions (TNW controller)", corresponding to scenario 3 of experiment 1



Simulation results of performance indicator "Number of instructions (TNE controller)", corresponding to scenario 3 of experiment 1



Simulation results of performance indicator "Number of instructions (ARR controller)", corresponding to scenario 3 of experiment 1



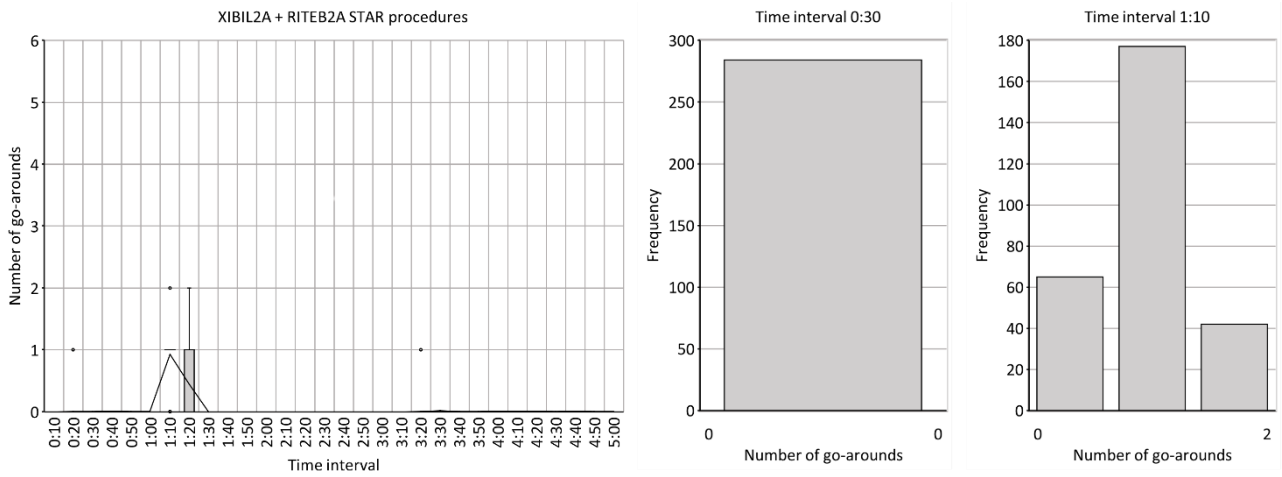
Simulation results of performance indicator "Number of instructions (TWR controller)", corresponding to scenario 3 of experiment 1

B.2 Experiment 2

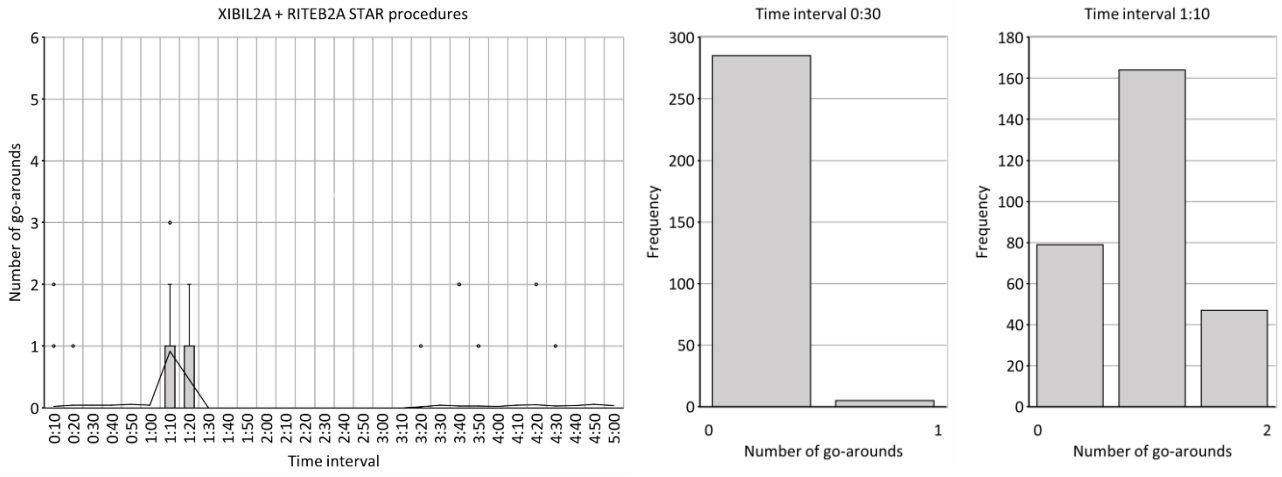
This appendix is meant to provide further context to the simulation results that are presented in table 14 by providing the measured dynamics of the following performance indicators:

- *Number of go-around*
- *Number of vector inbound trombone*
- *Number of vector outbound trombone*
- *Percentage in tactical mode (ARR controller)*
- *Number of instructions (ARR controller)*

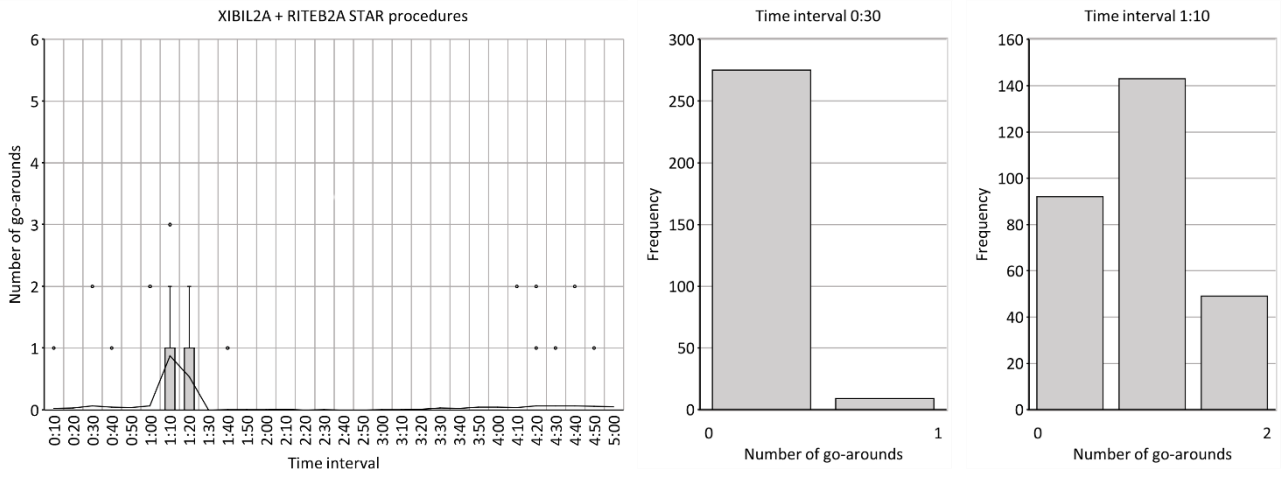
Number of go-arounds – Experiment 2 – Scenarios 1, 2, 3



Simulation results of performance indicator "Number of go-arounds", corresponding to scenario 1 of experiment 2

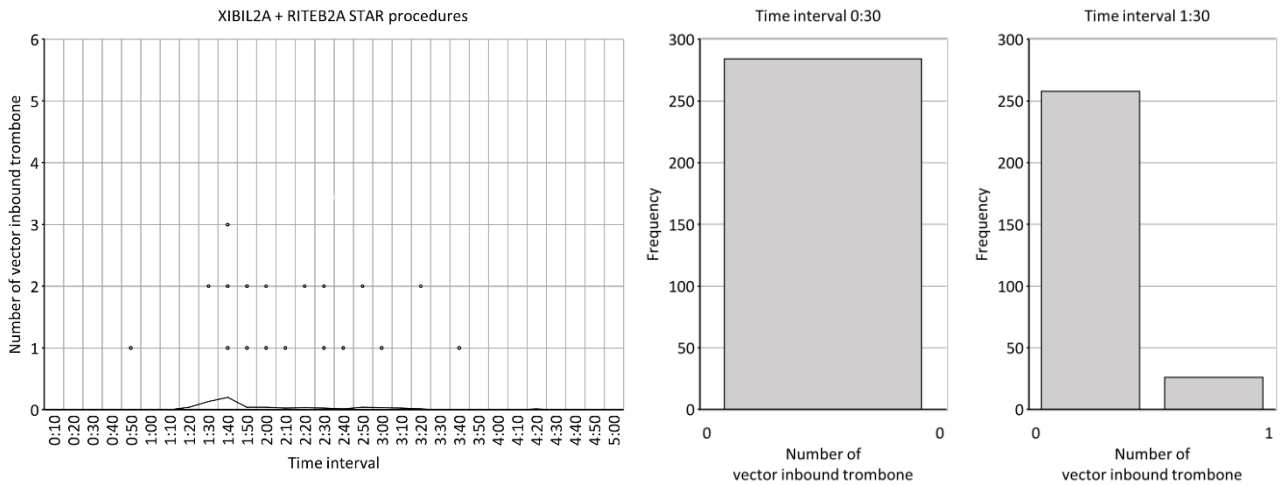


Simulation results of performance indicator "Number of go-arounds", corresponding to scenario 2 of experiment 2

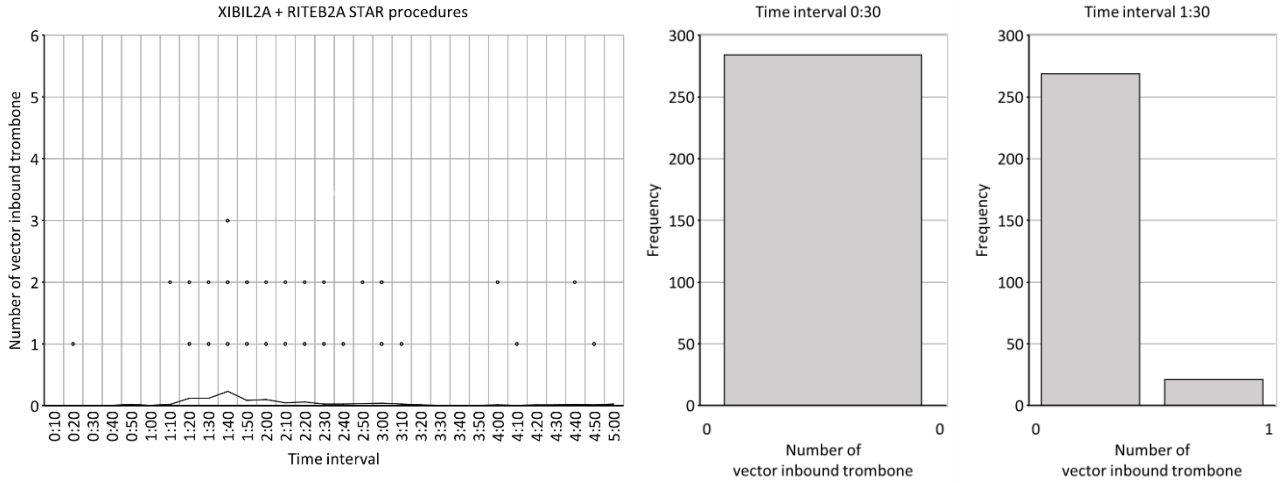


Simulation results of performance indicator "Number of go-arounds", corresponding to scenario 3 of experiment 2

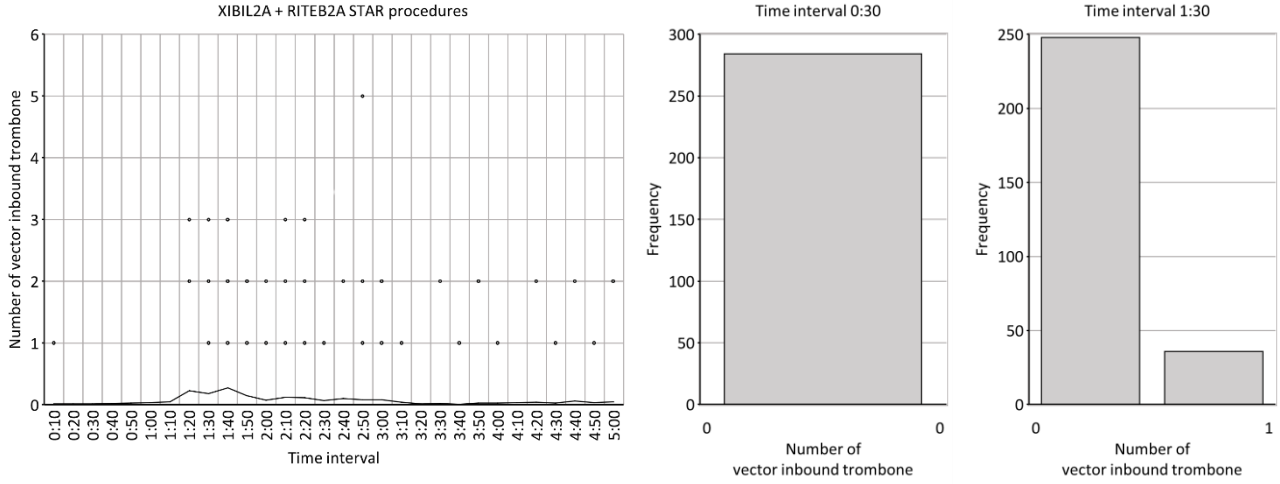
Number of vector inbound trombone – Experiment 2 – Scenarios 1, 2, 3



Simulation results of performance indicator "Number of vector inbound trombone", corresponding to scenario 1 of experiment 2

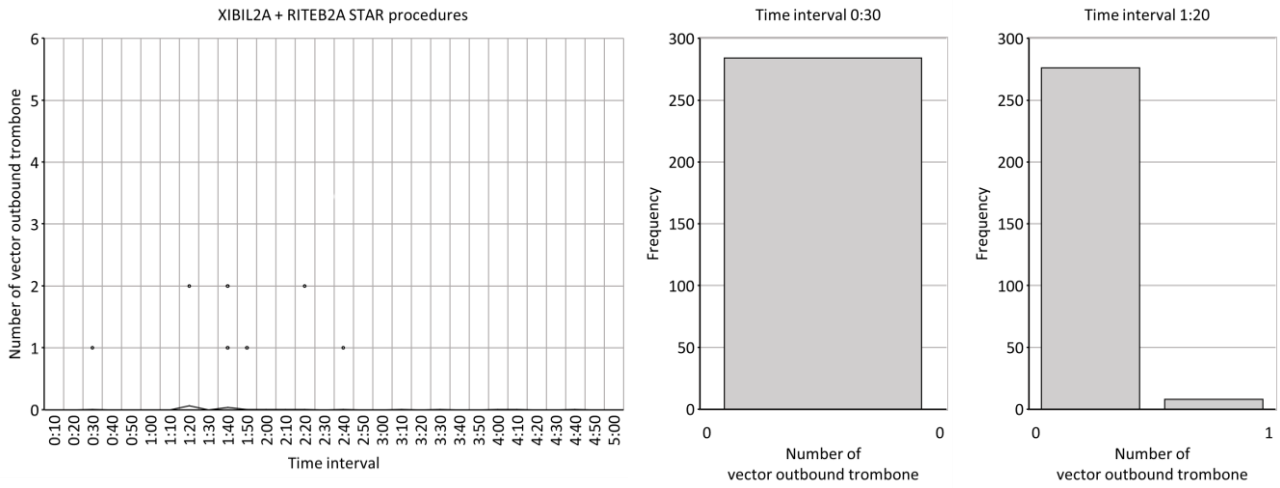


Simulation results of performance indicator "Number of vector inbound trombone", corresponding to scenario 2 of experiment 2

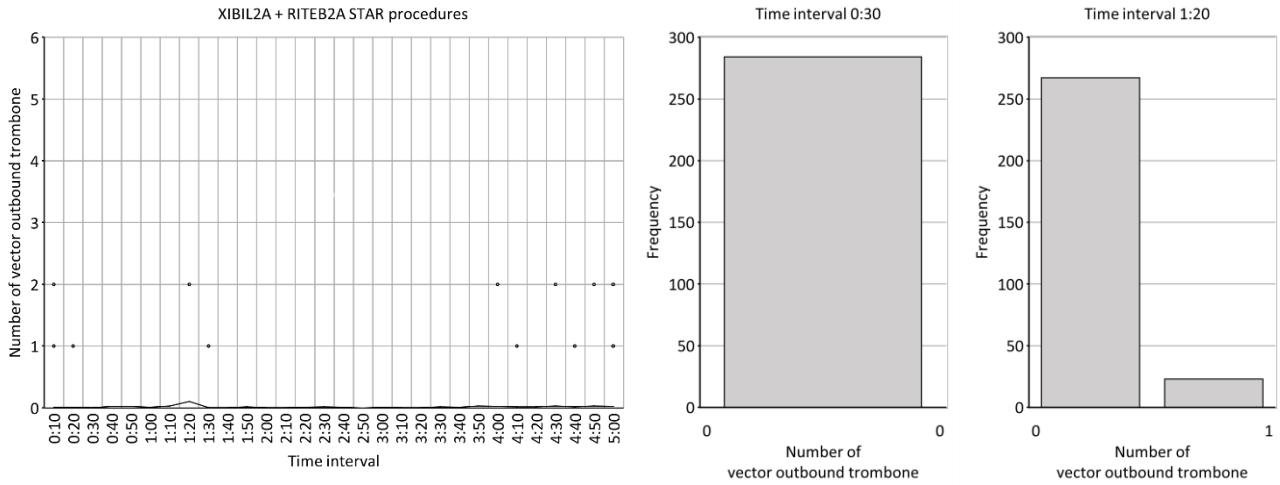


Simulation results of performance indicator "Number of vector inbound trombone", corresponding to scenario 3 of experiment 2

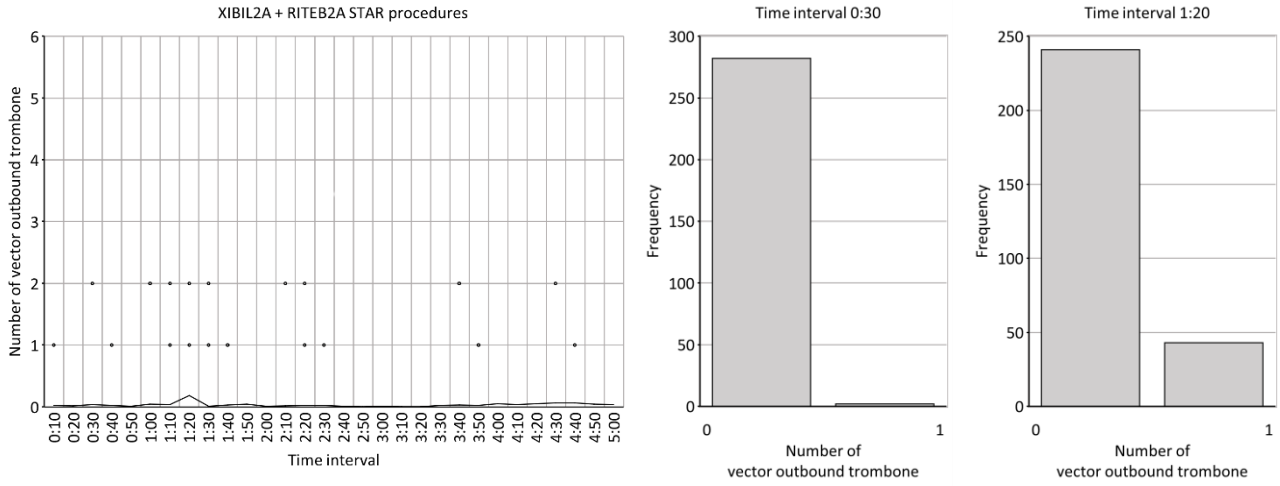
Number of vector outbound trombone – Experiment 2 – Scenarios 1, 2, 3



Simulation results of performance indicator "Number of vector outbound trombone", corresponding to scenario 1 of experiment 2

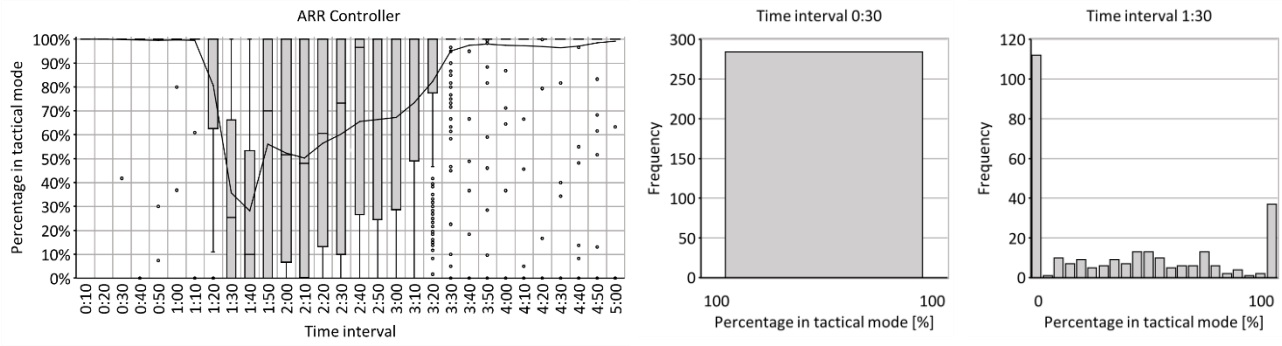


Simulation results of performance indicator "Number of vector outbound trombone", corresponding to scenario 2 of experiment 2

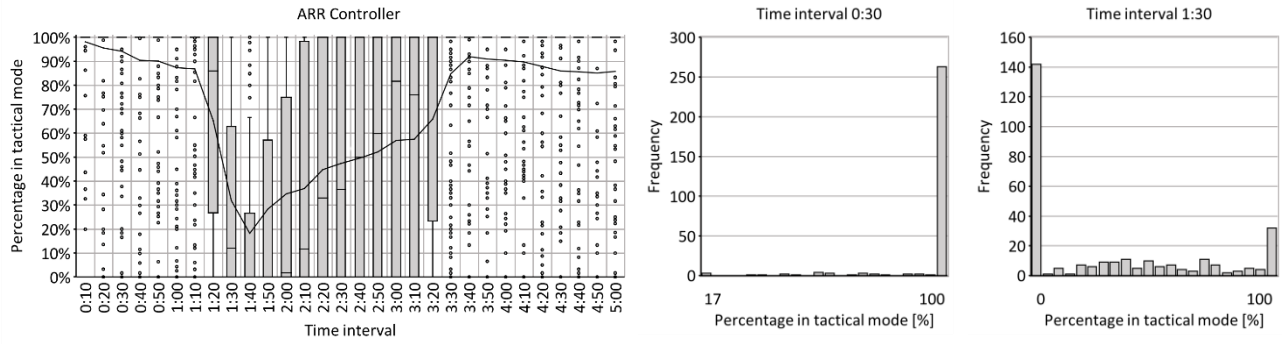


Simulation results of performance indicator "Number of vector outbound trombone", corresponding to scenario 3 of experiment 2

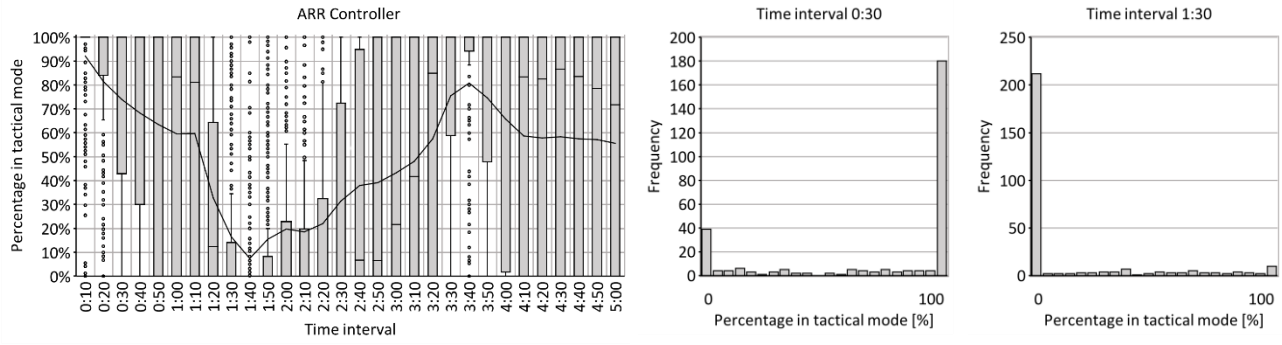
Percentage in tactical mode (ARR controller) – Experiment 2 – Scenarios 1, 2, 3



Simulation results of performance indicator "Percentage in tactical mode (ARR controller)", corresponding to scenario 1 of experiment 2

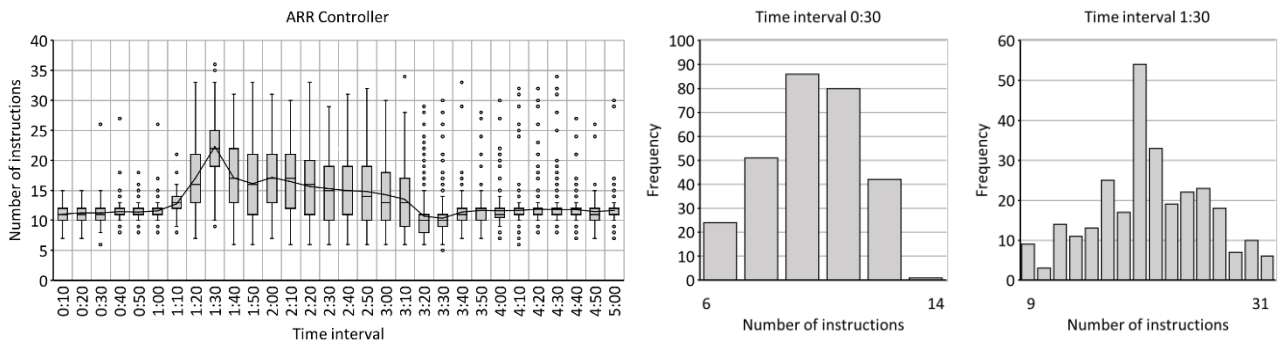


Simulation results of performance indicator "Percentage in tactical mode (ARR controller)", corresponding to scenario 2 of experiment 2

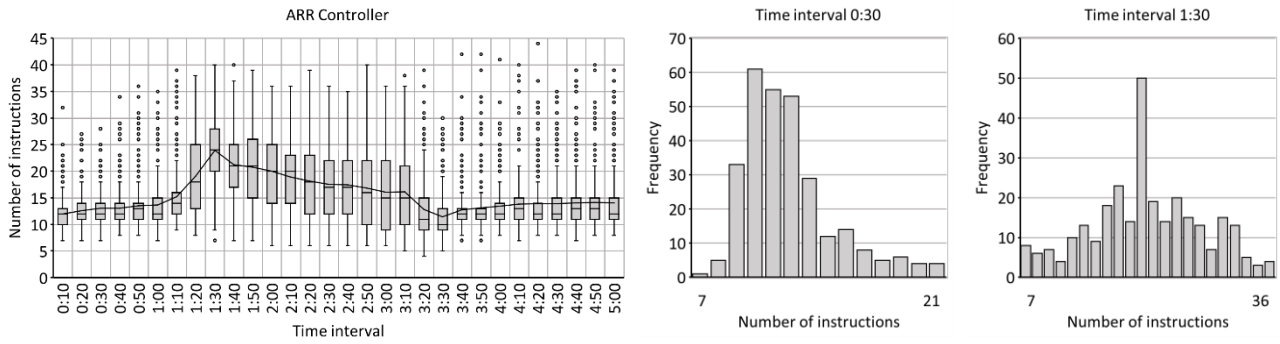


Simulation results of performance indicator "Percentage in tactical mode (ARR controller)", corresponding to scenario 3 of experiment 2

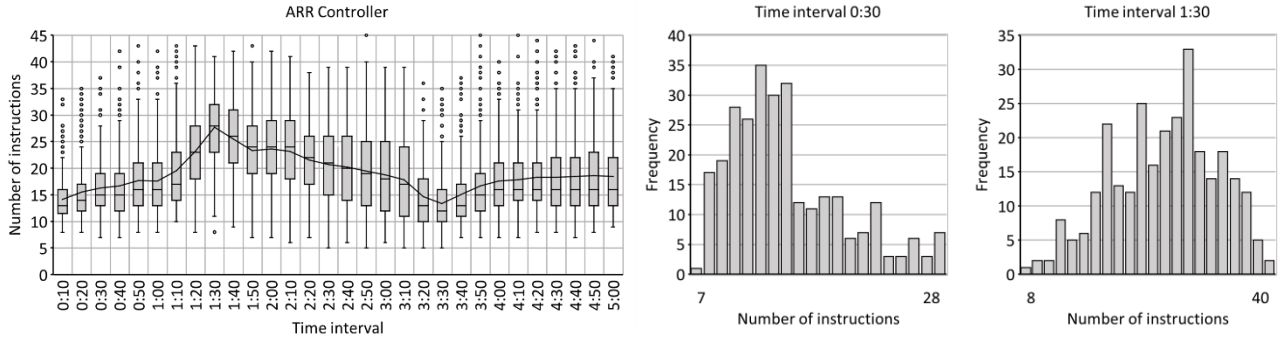
Number of instructions (ARR controller) – Experiment 2 – Scenarios 1, 2, 3



Simulation results of performance indicator "Number of instructions (ARR controller)", corresponding to scenario 1 of experiment 2



Simulation results of performance indicator "Number of instructions (ARR controller)", corresponding to scenario 2 of experiment 2



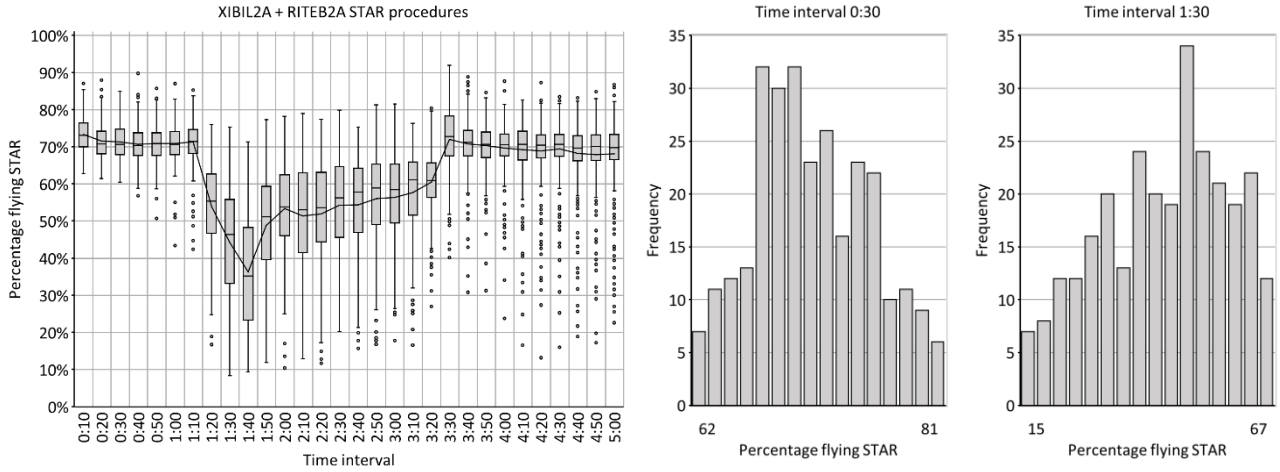
Simulation results of performance indicator "Number of instructions (ARR controller)", corresponding to scenario 3 of experiment 2

B.3 Experiment 3

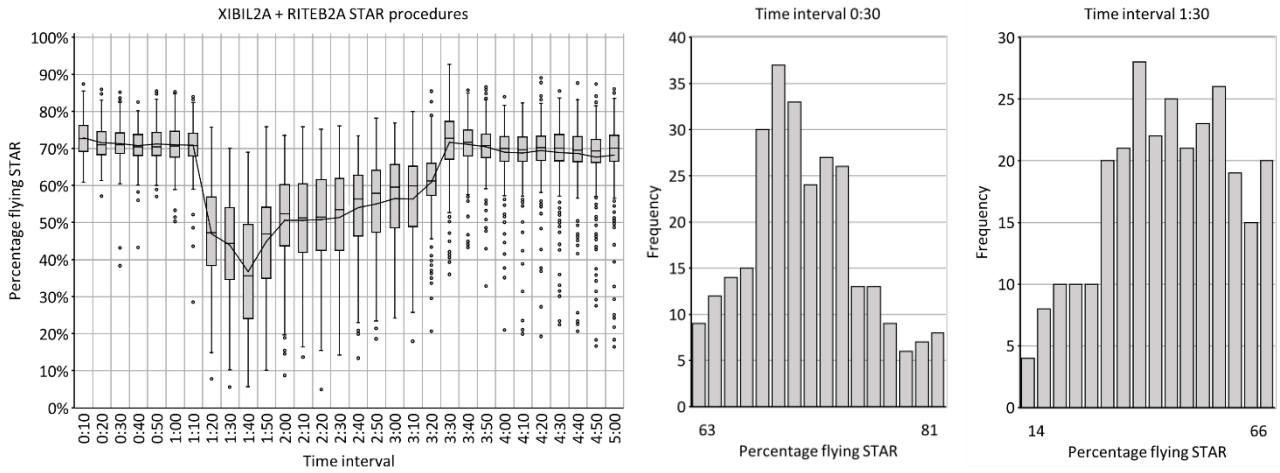
This appendix is meant to provide further context to the simulation results that are presented in table 16 by providing the measured dynamics of the following performance indicators:

- *Percentage flying STAR*
- *Percentage flying vector STAR*
- *Percentage flying vector merge*
- *Percentage flying holding*
- *Number of instructions (TNW controller)*
- *Number of instructions (TNE controller)*
- *Number of instructions (ARR controller)*

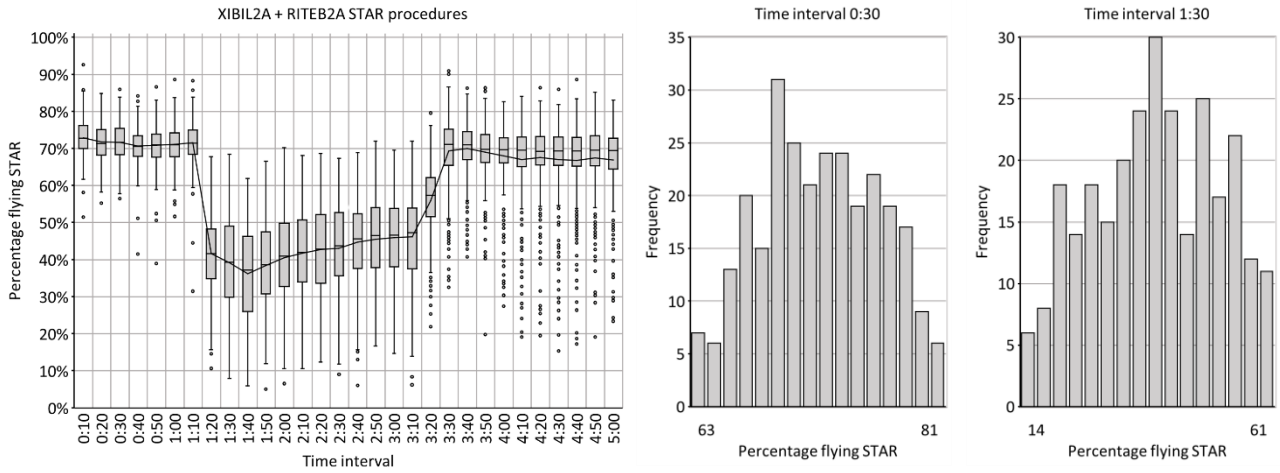
Percentage flying STAR – Experiment 3 – Scenarios 1, 2, 3



Simulation results of performance indicator "Percentage flying STAR", corresponding to scenario 1 of experiment 3

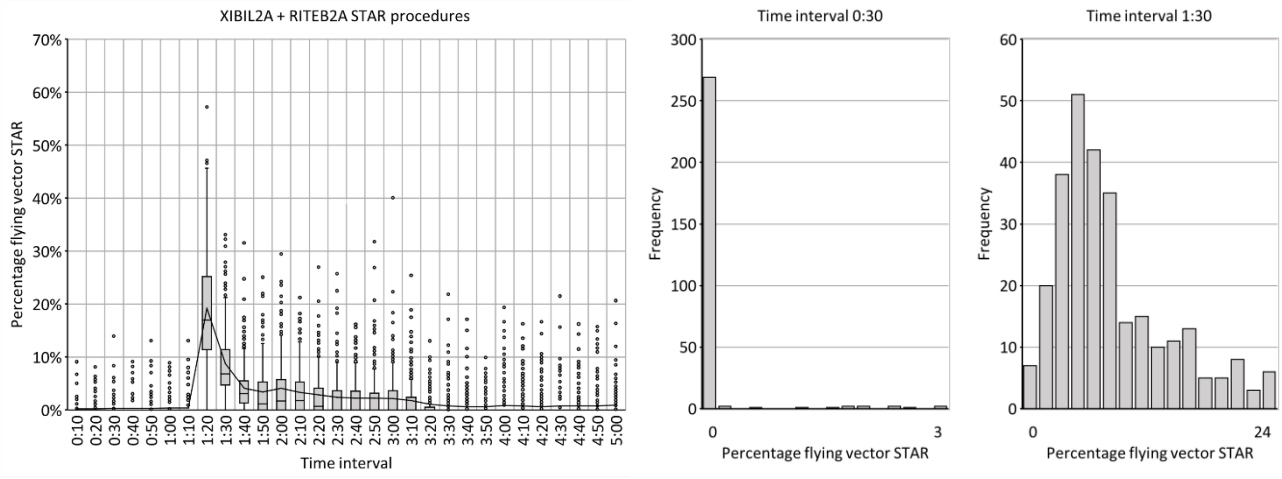


Simulation results of performance indicator "Percentage flying STAR", corresponding to scenario 2 of experiment 3

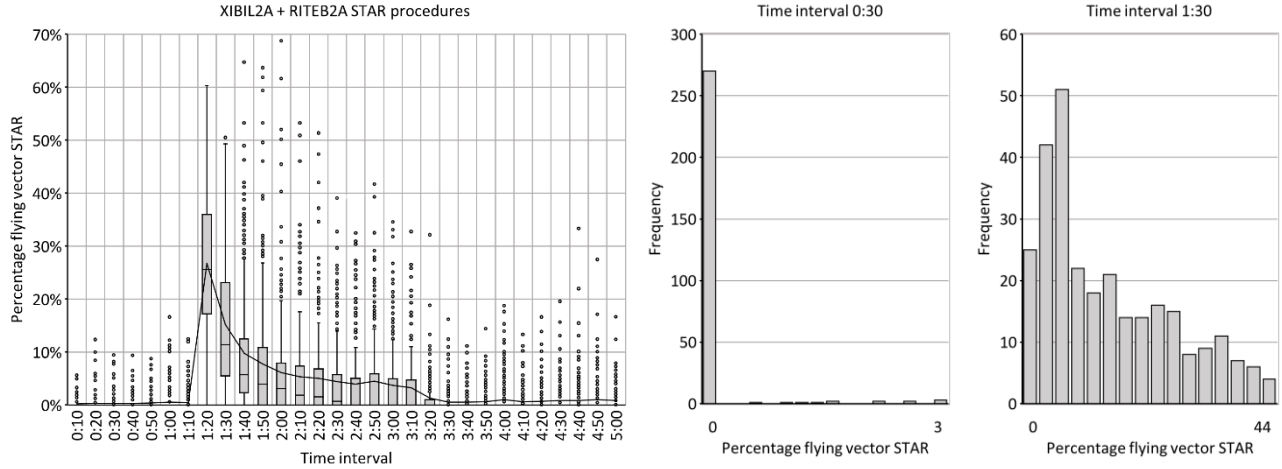


Simulation results of performance indicator "Percentage flying STAR", corresponding to scenario 3 of experiment 3

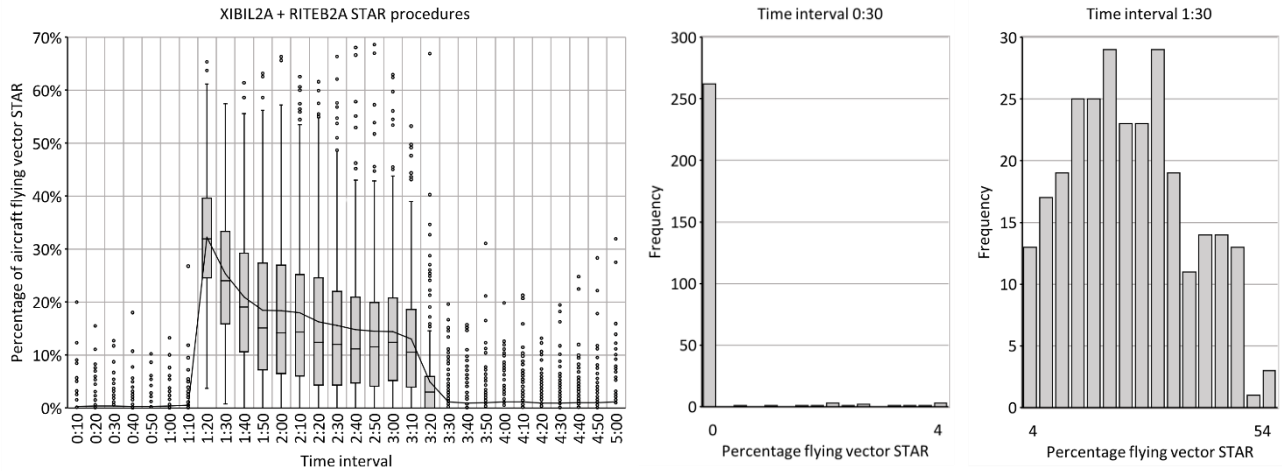
Percentage flying vector STAR – Experiment 3 – Scenarios 1, 2, 3



Simulation results of performance indicator "Percentage flying vector STAR", corresponding to scenario 1 of experiment 3

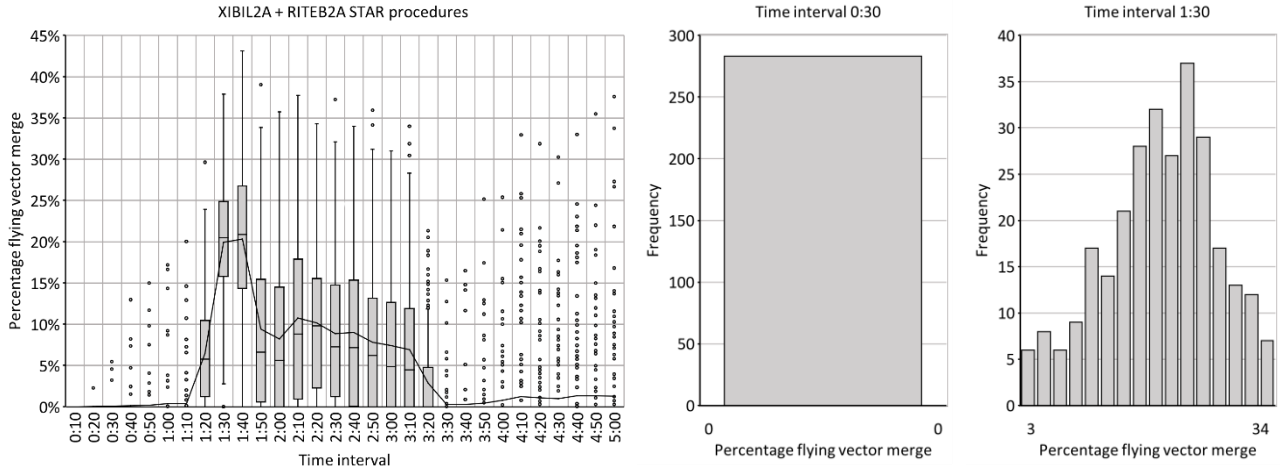


Simulation results of performance indicator "Percentage flying vector STAR", corresponding to scenario 2 of experiment 3

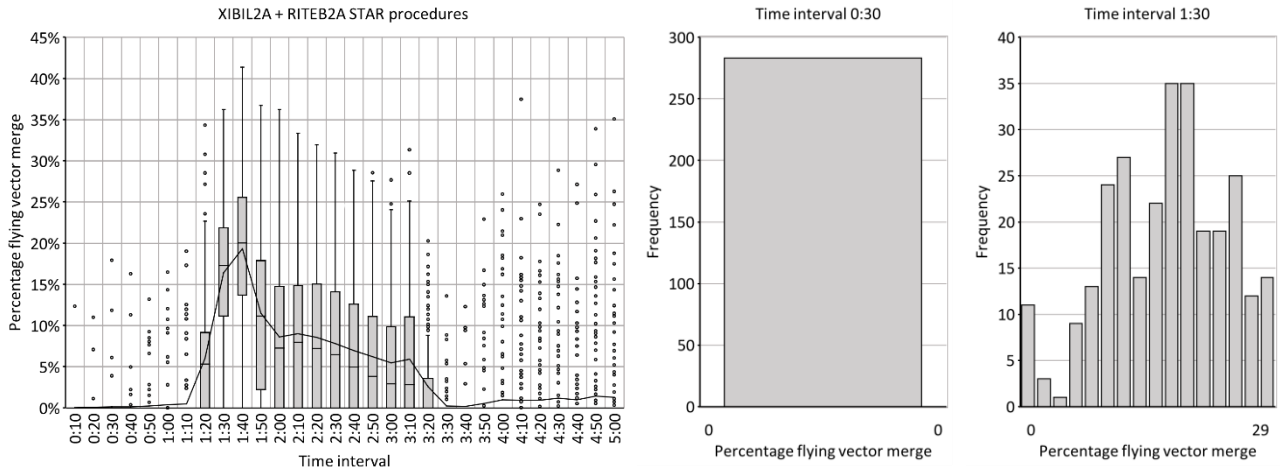


Simulation results of performance indicator "Percentage flying vector STAR", corresponding to scenario 3 of experiment 3

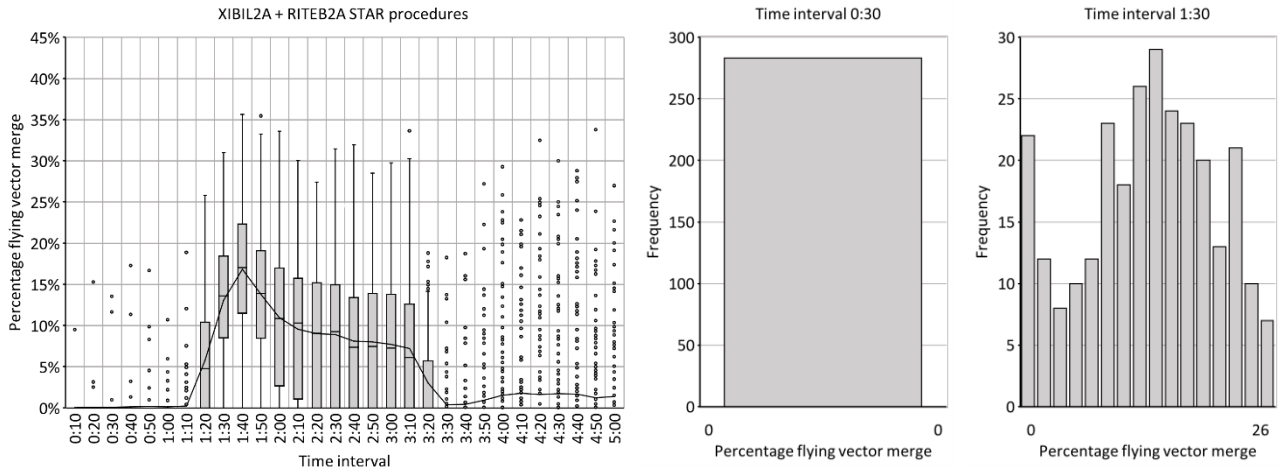
Percentage flying vector merge – Experiment 3 – Scenarios 1, 2, 3



Simulation results of performance indicator "Percentage flying vector merge", corresponding to scenario 1 of experiment 3

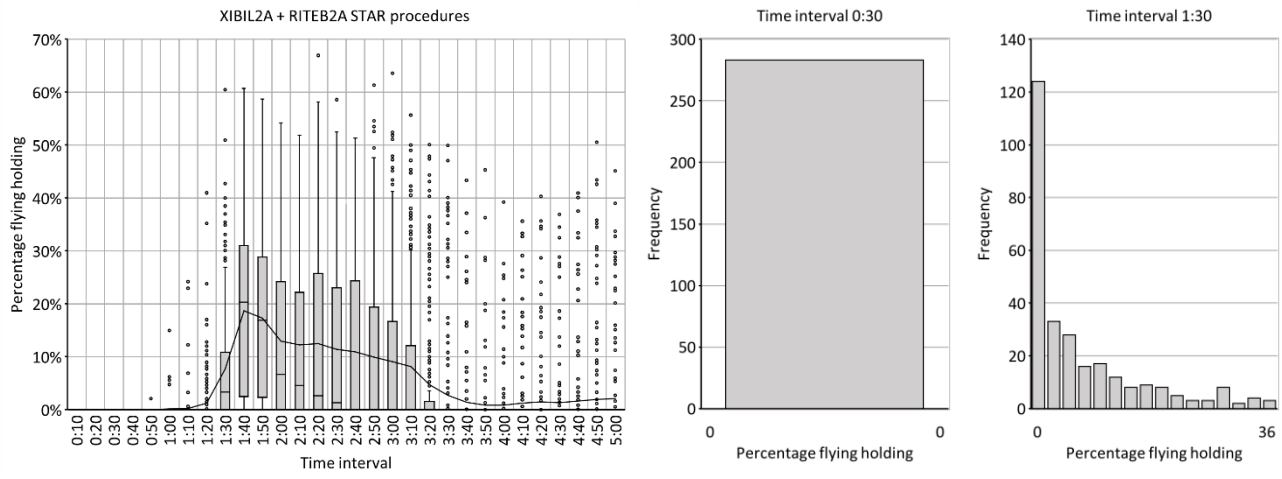


Simulation results of performance indicator "Percentage flying vector merge", corresponding to scenario 2 of experiment 3

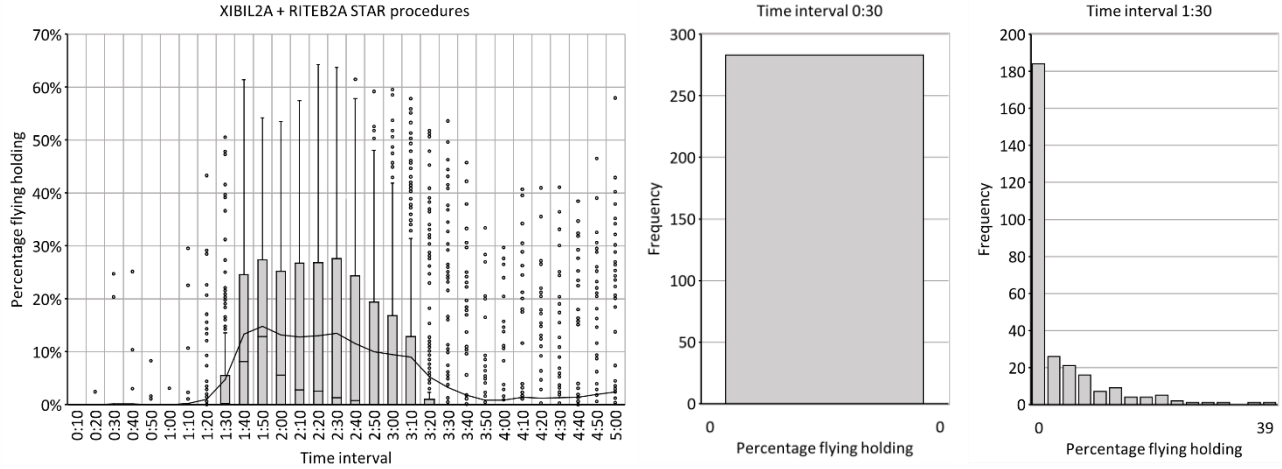


Simulation results of performance indicator "Percentage flying vector merge", corresponding to scenario 3 of experiment 3

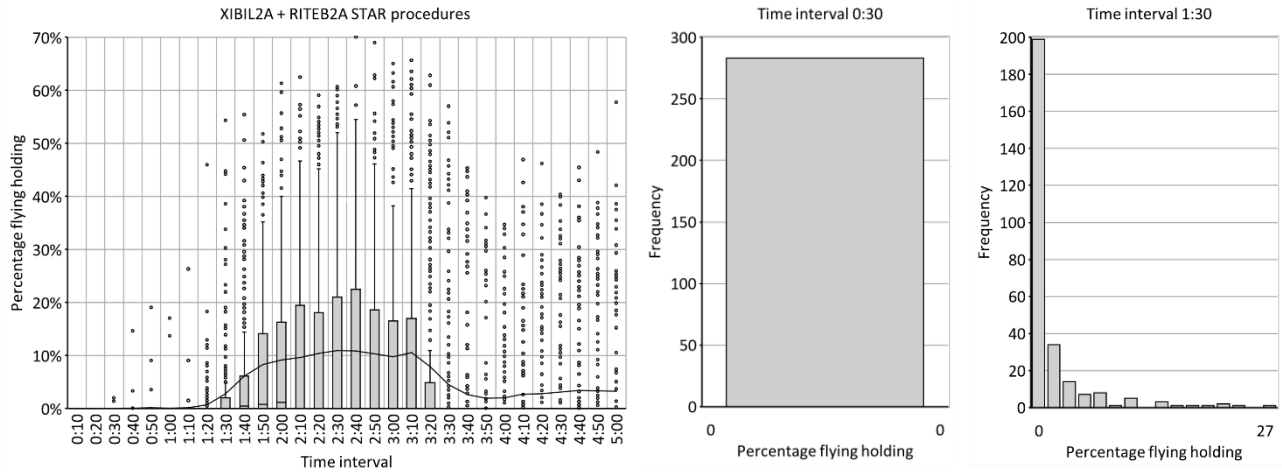
Percentage flying holding – Experiment 3 – Scenarios 1, 2, 3



Simulation results of performance indicator "Percentage flying holding", corresponding to scenario 1 of experiment 3

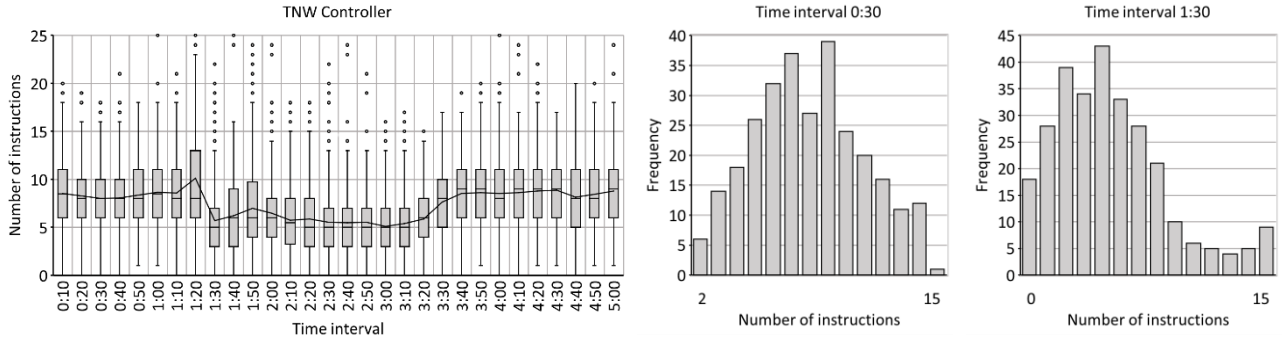


Simulation results of performance indicator "Percentage flying holding", corresponding to scenario 2 of experiment 3

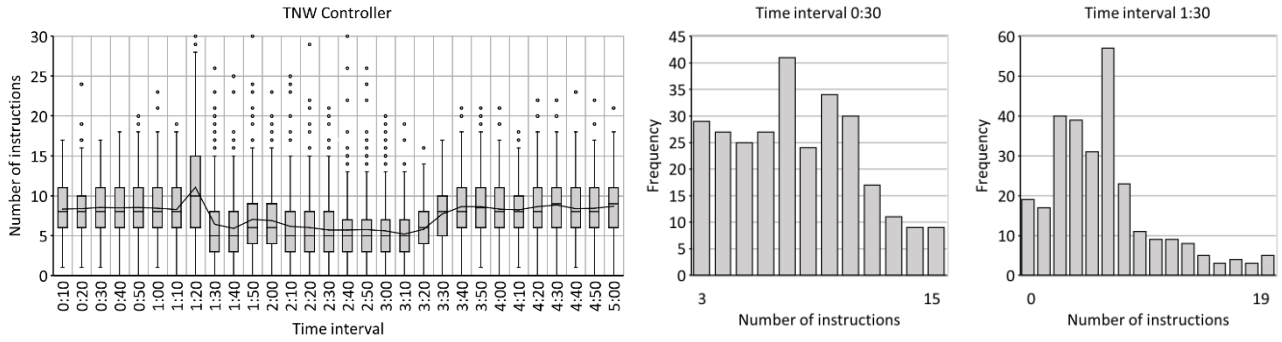


Simulation results of performance indicator "Percentage flying holding", corresponding to scenario 3 of experiment 3

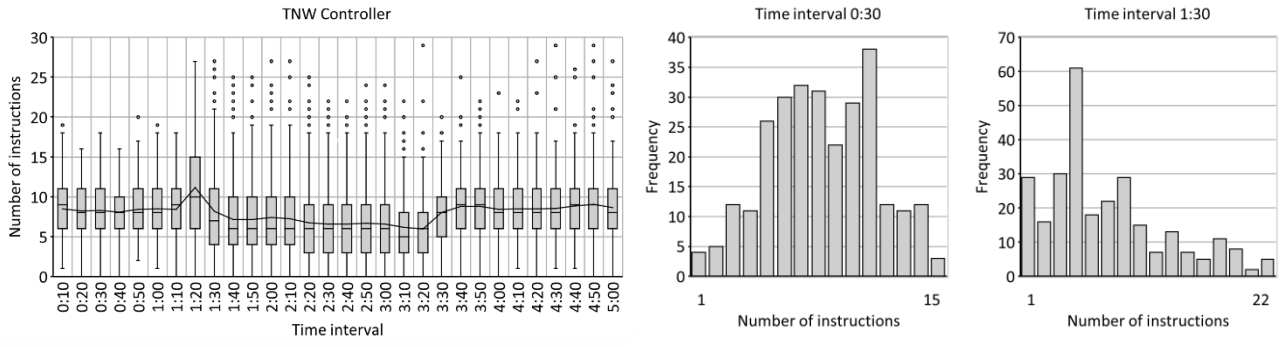
Number of instructions (TNW controller) – Experiment 3 – Scenarios 1, 2, 3



Simulation results of performance indicator "Number of instructions (TNW controller)", corresponding to scenario 1 of experiment 3

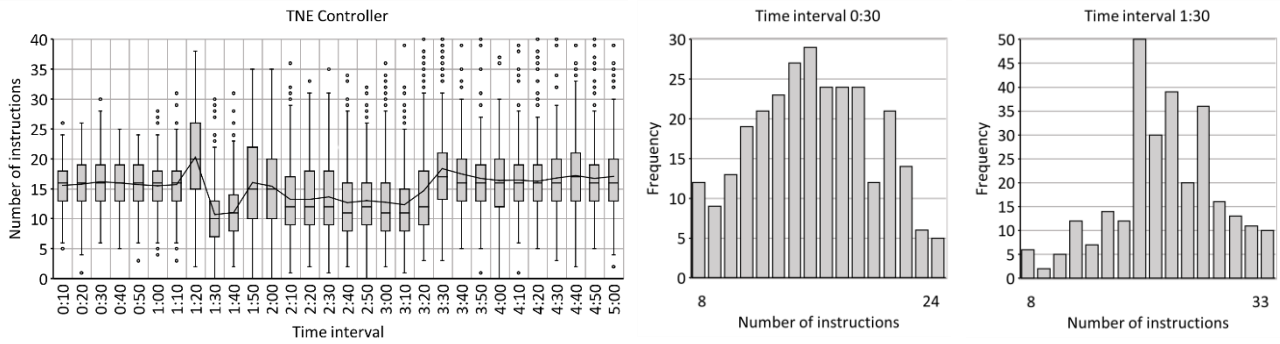


Simulation results of performance indicator "Number of instructions (TNW controller)", corresponding to scenario 2 of experiment 3

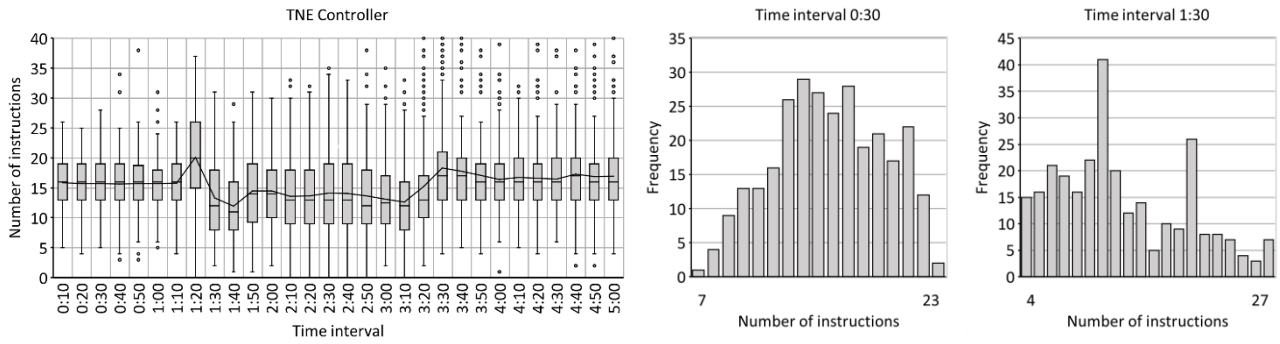


Simulation results of performance indicator "Number of instructions (TNW controller)", corresponding to scenario 3 of experiment 3

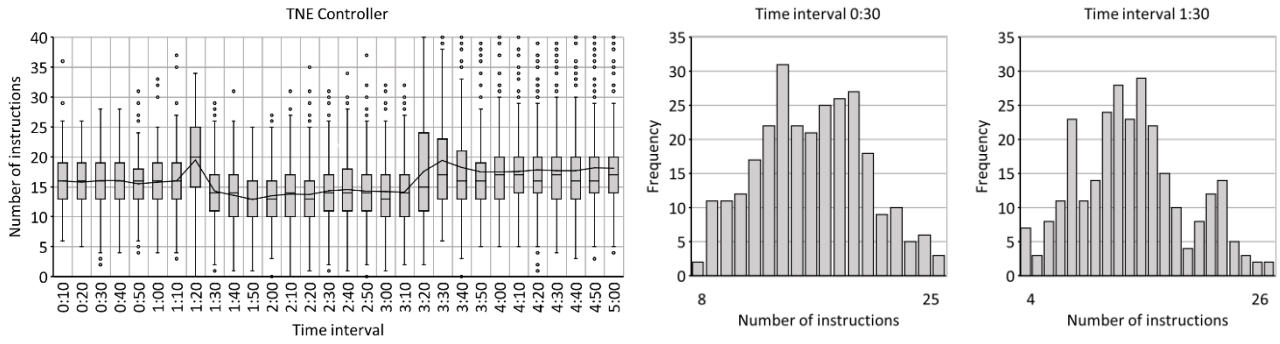
Number of instructions (TNE controller) – Experiment 3 – Scenarios 1, 2, 3



Simulation results of performance indicator "Number of instructions (TNE controller)", corresponding to scenario 1 of experiment 3

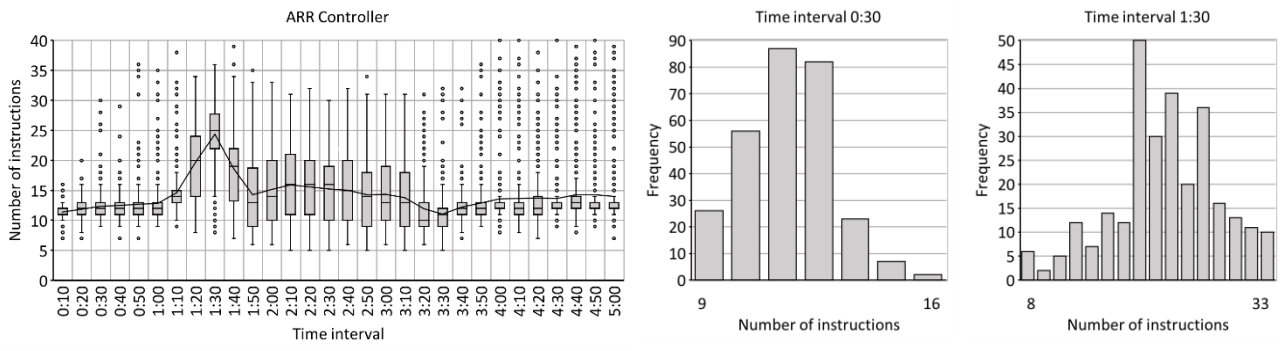


Simulation results of performance indicator "Number of instructions (TNE controller)", corresponding to scenario 2 of experiment 3

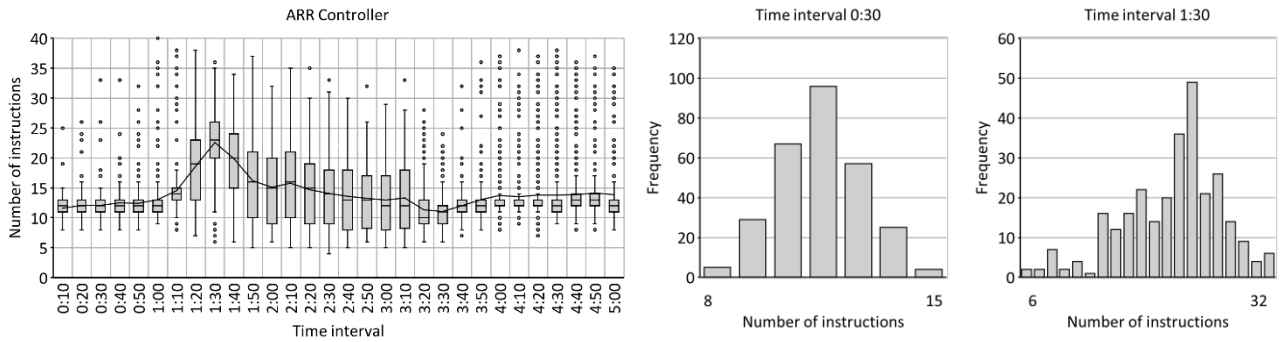


Simulation results of performance indicator "Number of instructions (TNE controller)", corresponding to scenario 3 of experiment 3

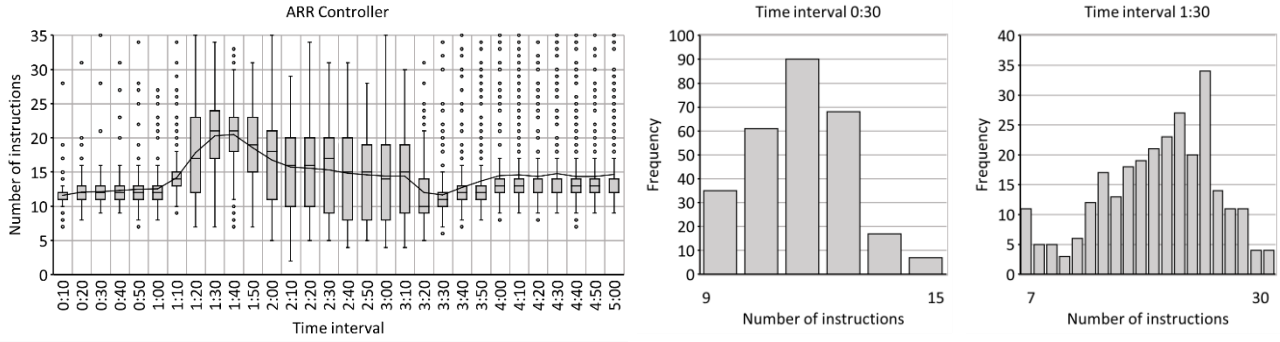
Number of instructions (ARR controller) – Experiment 3 – Scenarios 1, 2, 3



Simulation results of performance indicator "Number of instructions (ARR controller)", corresponding to scenario 1 of experiment 3



Simulation results of performance indicator "Number of instructions (ARR controller)", corresponding to scenario 2 of experiment 3



Simulation results of performance indicator "Number of instructions (ARR controller)", corresponding to scenario 3 of experiment 3

Appendix C Model specification

This appendix provides the complete formal specification of the model as implemented in *AnyLogic*. This specification appendix is meant to contextualize the more abstract model description as presented in chapter 5. The structure and the contents of this appendix closely resemble the architecture of the modelling elements that are provided by *AnyLogic*. Appendix C.1 will therefore first provide a description of the modelling elements that have been used to specify (and to implement) the model, together with the corresponding semantics. All the remaining sections within this appendix relate to the formal specification of the simulation environment and each identified agent. More specifically they mathematically specify the incorporated dynamics, stochastics, interactions, behaviours, states, data sources, reasoning, processes, features, conditions, etc.

C.1 Specification in AnyLogic format

AnyLogic provides a number of modelling elements and modelling methods that allow to specify the internal states of each agent and the interactions in and between agents in a very efficient, structured and readable manner. The developed formal agent-based model is fully defined by using only the following modelling elements: variables, parameters, sets, statecharts, events and functions. The combination of these modelling elements will eventually define the emergent behaviour and properties of the modelled socio-technical system. This appendix describes the modelling elements and the corresponding semantics that have been used to specify the developed agent-based model.

C.1.1 Variables and parameters

The model contains a lot of variables and parameters that each describe a specific and essential element of an agent. The amount of variables becomes even larger when one or more populations of the same agent type are alive, for instance the set of aircraft- and controller agents. Behaviour of a certain agent may be determined by the value of a variable that is contained by another agent. A readable and systematic way of variable specification should be applied because of these large sets of variables being present in the model. For this reason, many variables are extended with an sub- and superscript to distinguish the specific agent, holding, STAR procedure, instruction and significant point where the variable corresponds and applies to. Sub- and superscripts are added to these variables to make them more specific and to add more context, e.g. $s_{\mathrm{S}}^{a_{1,i},c}$, $v_{\mathrm{GS}}^{a_i}$. The descriptions of the used superscripts and subscripts are provided in table 17 and 18 respectively.

Superscript	Description
c	Variable that corresponds to controller agent c, where $c \in \mathbb{C}$
a_i	Variable that corresponds to aircraft agent a_i, where $a_i \in \mathbb{A}$
a_i, c	Variable that describes a specific situation awareness component/element of aircraft agent a_i as observed by controller agent c, where $a_i \in \mathbb{A}^c$ and $c \in \mathbb{C}$
$a_{j,i}, c$	Variable that describes a specific situation awareness component/element of aircraft agent $a_{j,i}$ as observed by controller agent c, where $a_{j,i} \in \mathbb{A}^c$ and $c \in \mathbb{C}$

Table 17 – Descriptions of the superscripts that are used in the model specification

Subscript	Description
D	Variable that is related to aircraft type D, where $D \in \mathbb{D}$
H	Variable that is related to a specific holding H, where $H \in \mathbb{H}_S$
I	Variable that is related to a specific instruction I, where $I \in \mathbb{I}$
k	Variable that is related to the k^{th} waypoint of STAR procedure S
S	Variable that is related to a specific STAR procedure S, where $S \in \mathbb{S}$
w_{index}	Variable that is related to one of the identified significant points

Table 18 – Descriptions of the subscripts that are used in the model specification

The type of variables that have been described above can be defined as 'class variables'. The class variables are constantly present, i.e. the variable's lifetime is the same as the respective agents' lifetime. Class variables therefore provide information about the agents' state. The value of class variables can be obtained, or changed as long as the respective agent is alive. All the class variables have been given a unique notation and are extended with the superscripts of table 17 if applicable.

The other type of variables that are considered in the model are defined as 'local variables'. The local variable is an auxiliary temporary variable that only exists while a particular function, action or block of statements is executed. Local variables are initialized when a function, etc. is executed and are removed when the execution is finished. Since the model contains a lot of functions and other code, it automatically contains local variables as well. Such local variables may often have the same meaning and sometimes even a similar notation with that of a class variable. A local variable is therefore extended with the breve (˘) to clarify the difference between class and local variables. Do note that local variables are not stored and that they do not change and influence the agent's state.

Parameters are (mostly) used to describe static values or some characteristics of the modelled agent. Multiple agents may require the same parameter for their calculations, i.e. functions, statecharts, etc. Parameters are therefore mostly not extended with superscripts, since these parameter values are not agent specific. Some of the parameter tables contain parameter values that are described by "*varying*". This means that these parameters are varied in the multiple experiments that have been conducted. See chapter 8 for the exact values that have been used throughout the multiple experiments.

C.1.2 Sets

Sets are in particular used to collect the multiple agent populations in to. Each agent that corresponds to a population is located at a specific index within the set. All sets are considered to be ordered, i.e. the indices of the elements/agents are fixed and only change when elements/agents are added or subtracted. Set notation is an appropriate method to formalize the operations that can be performed in sets. This section will therefore describe the properties and symbols of set notation that are relevant and useful with respect to specific type of sets and associated operations that are used in the model. The mathematical representation of sets therefore takes into account the features and characteristics of sets as processed by *AnyLogic*.
Consider the following four sets of elements a_i, where $i \in \mathbb{N}_0$.

$\mathbb{A} = \{a_1, a_2, a_3, a_4, a_5\}$ $\mathbb{A}_1 = \{a_3, a_1\}$ $\mathbb{A}_2 = \{a_1, a_4, a_6, a_2\}$ $\mathbb{A}_3 = \{\}$

By considering these example sets the following statements hold:

Statement	Description
$\{a_2\} \in \mathbb{A}$	a_2 is an element of set $\mathbb{A}$
$\{a_2\} \notin \mathbb{A}_1$	a_2 is not an element of set $\mathbb{A}_1$
$\mathbb{A}_1 \subseteq \mathbb{A}$	$\mathbb{A}_1$ is a subset of $\mathbb{A}$, i.e. every element of $\mathbb{A}_1$ is an element of $\mathbb{A}$
$\lvert\mathbb{A}_2\rvert = 4$	$\mathbb{A}_2$ contains 4 elements
$\mathbb{A}_3 = \emptyset$	$\mathbb{A}_3$ does not contain any elements
$\mathbb{A}_2[2] = a_6$	The third element of $\mathbb{A}_2$ is a_6 (the first element is located at index 0)
$(i\{a_4\} \in \mathbb{A}) = 3$	The index of a_4 within $\mathbb{A}$ is 3 (the first element is located at index 0)
$\{a_2\} \cup \mathbb{A}_1 = \{a_3, a_1, a_2\}$	Addition of element a_2 to $\mathbb{A}_1$
$\{a_4\} \setminus \mathbb{A}_2 = \{a_1, a_6, a_2\}$	Subtraction of element a_4 from $\mathbb{A}_2$

These basic concepts, notations and operations are often applied in the action fields of functions, statecharts and events.

C.1.3 Statecharts

The model contains many statecharts that each describe a specific operation, process, behaviour or specific situation awareness component of an agent. The statecharts that are used in the model specification (may) consist of the following constructs (figure 52):

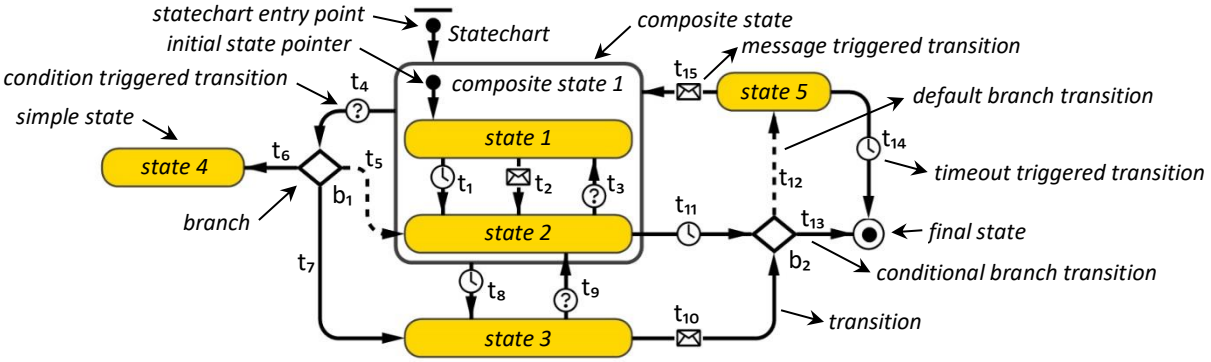


Figure 52 – Example statechart structure consisting of various statechart elements

Statechart Entry Point

A statechart entry point is used to indicate the initial state of the statechart. Every statechart has therefore exactly one statechart entry point.

States

A state defines a specific location, condition or mode of operation of the agent. A state on its own is called a *simple state*. When a (outer) state contains other (inner) states, the (outer) state is called a *composite state*. The activity of a specific state in a statechart may influence other state(s) in different statechart(s) within the same agent or within another interconnected agent. In order to describe every state orderly and readable, the following notation style is used to index a specific state in one of the agent's statecharts: ⟨*state*⟩⌊*statechart*⌋[*agent*]. This notation style simplifies the referencing as used in the condition triggered transitions.

Table 19 contains some of the agent notations that are used within this specification chapter. These notations are required, because the model contains duplicate statecharts due to agent populations. Each agent type is build up and modelled with a fixed set of statecharts. This means that a different set of states might be active among the many agents that are alive within a certain agent population. The notations as presented in table 19 are therefore necessary, since an agent may react upon a certain state that belongs to another agent.

[agent]	Description
$[c]$	Statechart that corresponds to controller agent c, where $c \in \mathbb{C}$
[*flight crew* a_i]	Statechart that corresponds to the flight crew a_i agent, where $a_i \in \mathbb{A}$
[*MCP* a_i]	Statechart that corresponds to the MCP a_i agent, where $a_i \in \mathbb{A}$
$[a_i, c]$	Statechart that describes a specific operational state of a_i as observed by c, where $a_i \in \mathbb{A}^c$. The combination of all these specific type of (SA-)statecharts will define the situation awareness of c about a_i
$[a_{j,i}, c]$	Statechart that describes a specific operational state of $a_{j,i}$ as observed by c, where $a_{j,i} \in \mathbb{A}^c$. The combination of all these specific type of (SA-)statecharts will define the situation awareness of c about $a_{j,i}$

Table 19 – Agent notations that are used in the statecharts in the model specification

Each state within a statechart may contain entry- and/or exit actions. Entry actions are executed when the statechart enters the respective state, exit actions are executed when the statechart exits the state. Entry and/or exit actions are in the specification sections only defined and written down when the state contains any. These actions are specified in the following format:

- State entry actions:

1: ..the to be executed action/algorithm..

- State exit actions:

1: ..the to be executed action/algorithm..

Initial state pointer

The initial state pointer represents the initial state of a composite state. In the situation when a (entry) transition is taken that is connected to the border of the composite state, the simple state within the composite that is connected with the initial state pointer will become active first (e.g. t_{15} and *state 1* in figure 52). However, when a (entry) transition is taken that is directly connected to a simply state within a composite state, then that specific simple state will become active first (e.g. t_5 and *state 2* in figure 52).

State transitions

Each statechart contains one or more transitions where each transition is assigned a different name (t_1, t_2, etc.). Transitions allow statecharts to switch state, which cause for instance agent behaviour to change, or result in changed operational modes. Switching states occurs according to a specified trigger event. The transitions that are contained in the model can be triggered by a timeout, a specified condition, or a received and matching triggering message. See table 20 below for a description of each specific trigger type.

Trigger type	Transition triggered:
Timeout	after a specified time interval, this trigger type can therefore be interpreted as a delay transition. A state is stayed in for a given time, and left when the specified time period is passed. The timeout transitions start counting from the moment that the statechart enters the source state of the transition, i.e. the state on whose boundary the transition start point is located.
Condition	once a given condition becomes true. This condition is an arbitrary Boolean expression that depends on any state, either with continuous or discrete dynamics. This trigger type is continuously evaluated when the transition is active.
Message	upon reception of a triggering message that matches the message as specified in the properties. The statechart receives messages and reacts to it. This trigger type is therefore used to model communication or commands.

Table 20 – Descriptions of the different transition trigger types

All transitions are specified in one of the formats shown below. The specific structure of the transition specification depends on the transition type that is applicable. Note that the upcoming specification sections will only describe and define the fields (bullets) that are applicable. The *Guard* or *Action* fields for instance will not always be specified. All type of transitions may incorporate specified actions that are performed when the respective transition is taken.

Transition $t_{(index)}$

- Timeout: the timeout value after which the transition is and should be taken when the timeout elapses
- Guard: Boolean expression that should be evaluated true to allow the transition to be taken
- Action:

1: ..the to be executed action/algorithm..

Transition $t_{(index)}$

- Condition: Boolean expression that should be evaluated true to allow the transition to be taken. Such expression often includes (multiple) states that have to be (in-)active to allow a certain transition to be taken. The introduced state notation style is used for this purpose, i.e. ⟨*state*⟩⌊*statechart*⌋[*agent*] for states that have be active, and ¬⟨*state*⟩⌊*statechart*⌋[*agent*] for states that have to be inactive
- Action:

1: ..the to be executed action/algorithm..

(the conditional branch transitions that exit the branches are also specified using this conditional format)

Transition $t_{(index)}$

Message triggered transitions are specified using a description that describes which agent, state(chart), action and/or transition causes this specific transition to be triggered. Such triggering messages model in general updates of the agents' situation awareness or model the communication between agents. Message triggered transitions are in general taken when a specific related state becomes active. It is therefore that this triggering message is included in the Action field of the agent, state(chart), action and/or transition that causes this message triggered transition to be taken. Each triggering message is formalised using the format: "Trigger t_{index} of ⌊*statechart*⌋[*agent*]". The message triggered transition is in itself formalized by describing the specific state, transition, function or event that has caused the transition to be taken, e.g. "Triggered in ⟨*state*⟩⌊*statechart*⌋[*agent*]".

Branch

A branch is used to merge multiple incoming transitions into one outgoing transition or is used to split one incoming transition into multiple outgoing transitions for multiple destinations. A branch can be interpreted as an evaluation point in time in between the incoming and outgoing transitions and its corresponding transition actions. When the action of the incoming transition is executed, the branch is evaluating the guards of the transitions exiting the branch. The outgoing transition whose guard is evaluated positively (true) is taken.

Branch transitions

A second type of transition are the transitions that exit the branch state. These exiting transitions can only be triggered by either a specified condition, or by default. There can only be one default transition that exits the branch state. This default condition is only taken if the conditions for all other transitions that exit the branch state are false. These types of transitions do not have triggers, since they are immediately fired after the branch has been passed and the condition of the outgoing transition is satisfied.

Final state

The final state represents the termination point of a statechart. It is therefore that a final state can only be assigned entry actions and that transitions may not exit a final state. The final state can be described as a simple state.

C.1.4 Events

Events are used to model repeated processes or actions and are denoted by the symbol $\mathcal{E}_{index}^{agent}$. Here $\mathcal{E}$ represents the symbol for *event*, $agent$ indicates the agent where the event corresponds to (only applicable for agent populations), and $index$ denotes the specific type of event. The superscripts use the same notations as used in the variables and parameters. Within the model specification there are two types of events considered.

The first type of event is the *timeout triggered event* and is used when one wants to model and schedule periodic events. The timeout triggered events are formalised using the following format:

- First occurrence time: the time point at which the event should occur for the first time
- Recurrence time period: the time period between two successive events

1: ..the to be executed action/algorithm..

The second type of event is the *condition triggered event* and is used when one wants to model (periodic) action(s) after a certain specified condition becomes true. This type of event will stop working when the specified condition is evaluated to be true. The condition triggered event can be made periodic by using a so-called *restart* command. Once the condition triggered event is restarted the specified condition will be evaluated again. By applying this method the condition triggered event keeps evaluating a specific condition. The condition triggered events are formalised using the following format:

- Condition: Boolean expression that should be evaluated true to let the event occur

1: ..the to be executed action/algorithm..

C.1.5 Functions and actions

Each state, transition, event and function can often be described with an associated action. Such actions are specified using a block of statements that are related to for instance agent behaviour or a specific process. These statements are executed sequentially one after another in a top-down order. They can either be specified as a (re-usable) function or as a unique set of statements that are associated with a single state, transition, event, etc.

A function that calculates and returns a certain output is throughout this formalisation chapter generally denoted with the symbol $\mathcal{F}$ and extended with a sub- or/and superscript to clarify the meaning of the function. A function generally requires argument values that are used to perform a certain action or to return some value. These function arguments are put between parentheses, i.e. $\mathcal{F}(\text{arg}_1, \text{arg}_2, ..)$.

C.2 General

This appendix provides the general variables, parameters, sets, functions and (type of) statecharts that are used by and which relate to multiple agents in the model.

C.2.1 Variables, parameters and sets

Variable	State space	Unit	Description
t_A	$\mathbb{R}_+$	s	Current model time, i.e. time point in the simulation, starting at $t_A = 0$
t_{INF}	$\mathbb{R}_+$	s	Time period required by the agent to inform another agent about a certain situation or condition

Table 21 – List of general variables that are used by the multiple agents in the model

The controller and aircraft agents are modelled with some intelligence that is able to define:

- the specific direction of turning;
- the distance that has to be travelled to reach a specific point in space taking into account the circular shaped trajectory that is flown during the turning movement;
- the heading direction at which an aircraft should stop turning to reach a specific point in space.

This intelligence is modelled with the variables that are listed in table 22 and which are visualized in figure 53. These variables describe two types of distances and two types of heading directions and are used to define the position and orientation of an aircraft (i.e. a_i) relative to a specific significant point (i.e. w). The first category of variables consider the aircraft already pointed towards the significant point that it wants to fly to, or consider the aircraft's heading angle already parallel to the desired heading direction. These variables contain the subscript "D", which stands for *direct*. These type of variables do not take into account the turning movement that is required to arrive at the desired significant point or desired heading direction. The other category of variables do take into account the effects of turning movements that are required to arrive at this desired point or heading direction. These variables contain the subscript "A", which stands for *arc/tangent*. Both the notions *direct* and *tangent* are often used in the model specification.

Variable	State space	Unit	Description
$\breve{d}_{A,w}$	$\mathbb{R}_+$	NM	Distance (*tangent*) between a_i and w
$\breve{d}_{D,w}$	$\mathbb{R}_+$	NM	Distance (*direct*) between a_i and w
$\breve{\psi}_{A,w}$	⟨0, 360]	deg	Bearing (*tangent*) of w seen from a_i
$\breve{\psi}_{D,w}$	⟨0, 360]	deg	Bearing (*direct*) of w seen from a_i

Table 22 – Variables that are used to define the position and orientation of a_i relative to w

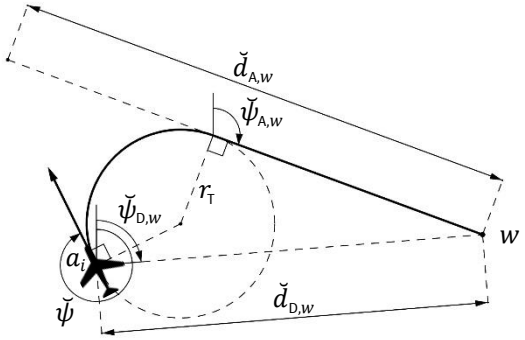


Figure 53 – Visualization of the type of variables that are used to define the position and orientation of a_i relative to w

Parameter	Value	Unit	Description
$l_{t_{INF}}$	53	s	Minimum value of t_{INF}
r_T	1.75	NM	Turn radius of aircraft in the simulation environment
t_{SM}	1800	s	Time point at which the model will begin to measure and collect simulation output, i.e. the time period that is required to create a stabilized and condensed flow of approaching traffic within the simulation environment
$v_{IAS,\varepsilon}$	2	kt	Allowed speed error
z_ε	50	ft	Allowed altitude error
$\mu_{t_{INF}}$	7.6	s	Mean value of t_{INF}
$\sigma_{t_{INF}}$	0.3	s	Standard deviation of t_{INF}

Table 23 – List op general parameters that are used by multiple agents in the model

Set	Contents	Description
$\mathbb{A}$	-	Collection of all aircraft agents that are currently alive in the simulation environment
$\mathbb{C}$	$\{c_1, c_2, c_3, c_4\}$	Set of executive controller agents that are considered in the model, where c_1 represents the TNW controller, where c_2 represents the TNE controller, where c_3 represents the ARR controller and where c_4 represents the TWR controller
$\mathbb{D}$	$\{D_1, D_2\}$	Set of aircraft types that are considered in the model, where D_1 represents the B738 aircraft type and where D_2 represents the B744 aircraft type
$\mathbb{H}_S$	$\{H_1\}$	Set of holding patterns for each S. Both S_1 and S_2 have however only one holding pattern modelled, i.e. $\mathbb{H}_{S_1} = \{H_1\}$ and $\mathbb{H}_{S_2} = \{H_1\}$
$\mathbb{I}$	$\{I_1, I_2, .., I_{17}\}$	Set of instructions that are considered in the model.
$\mathbb{J}$	{1,2,3,4,5}	Set of indices that relate to the five identified *reference aircraft*
$\mathbb{S}$	$\{S_1, S_2\}$	Set of STAR procedures that are used in the model, where S_1 represents the XIBIL2A procedure and S_2 the RITEB2A procedure
$\mathbb{W}$	$\{W_M, W_H\}$	Collection of wake vortex categories that are considered in the model, where W_M represents the MEDIUM category and where W_H represents the HEAVY category

Table 24 – List of sets that are used in multiple sections in the model specification

C.2.2 Functions

This appendix provides the set of functions that are used by multiple agents. These functions are related to probability distributions, conversions, wind conditions, geometry, bearings and distances.

C.2.2.1 Uniform distribution

Function $\mathcal{U}(\check{l}, \check{u})$

Argument	State space	Unit	Description
$\check{l}$	$-\infty < \check{l} < \check{u}$	-	Minimum value of $\check{X}$
$\check{u}$	$\check{l} < \check{u} < \infty$	-	Maximum value of $\check{X}$
Variable	**State space**	**Unit**	**Description**
$\check{X}$	$[\check{l}, \check{u}]$	-	Random variable

1: $\check{X} = \mathcal{U}(\check{l}, \check{u})$ *(distribution function supported in and provided by AnyLogic)*
2: **return:** $\check{X}$

Function 1 – Uniform distribution

C.2.2.2 Normal distribution

Function $\mathcal{N}(\check{\mu}, \check{\sigma})$

Argument	State space	Unit	Description
$\check{\mu}$	$\mathbb{R}$	-	Mean value of the normal distribution $\mathcal{N}$
$\check{\sigma}$	$\mathbb{R}_+$	-	Standard deviation of the normal distribution $\mathcal{N}$
Variable	**State space**	**Unit**	**Description**
$\check{X}$	$\mathbb{R}$	-	Random variable

1: $\check{X} = \mathcal{N}(\check{\mu}, \check{\sigma})$ *(distribution function supported in and provided by AnyLogic)*
2: **return:** $\check{X}$

Function 2 – Normal distribution

C.2.2.3 Truncated normal distribution

Function $\mathcal{N}_t(\check{\mu}, \check{\sigma}, \check{l}, \check{u})$

Argument	State space	Unit	Description
$\check{\mu}$	$\mathbb{R}$	-	Mean value of the normal distribution $\mathcal{N}$
$\check{\sigma}$	$\mathbb{R}_+$	-	Standard deviation of the normal distribution $\mathcal{N}$
$\check{l}$	$-\infty < \check{l} < \check{u}$	-	Minimum value of $\check{X}$
$\check{u}$	$\check{l} < \check{u} < \infty$	-	Maximum value of $\check{X}$
Variable	**State space**	**Unit**	**Description**
$\check{X}$	$[\check{l}, \check{u}]$	-	Random variable

1: $\check{X} = \mathcal{N}(\check{\mu}, \check{\sigma})$
2: **while** $(\check{X} < \check{l} \vee \check{X} > \check{u})$ **then**
3: $\quad \check{X} = \mathcal{N}(\check{\mu}, \check{\sigma})$
4: **return:** $\check{X}$

Function 3 - Truncated normal distribution

C.2.2.4 Lognormal distribution

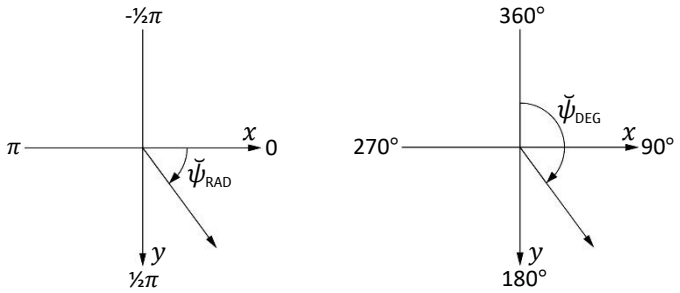
Function $\mathcal{L}(\breve{\mu}_n, \breve{\sigma}_n, \breve{l})$

Argument	State space	Unit	Description
$\breve{\mu}_n$	$\mathbb{R}_{>0}$	-	Mean value of the normally distributed logarithm of $\breve{X}$
$\breve{\sigma}_n$	$\mathbb{R}_+$	-	Standard deviation of the normally distributed logarithm of $\breve{X}$
$\breve{l}$	$\mathbb{R}_+$	-	Minimum value of $\breve{X}$, shifts the minimum from 0 to $\breve{l}$
Variable	**State space**	**Unit**	**Description**
$\breve{\sigma}_l$	$\mathbb{R}_+$	-	Standard deviation of the lognormal distribution
$\breve{\mu}_l$	$\mathbb{R}_+$	-	Mean value of the lognormal distribution
$\breve{X}$	$\mathbb{R}$	-	Random variable

1: $\breve{\sigma}_l = \sqrt{(\ln(\breve{\sigma}_n^2 / \breve{\mu}_n^2 + 1))}$
2: $\breve{\mu}_l = \ln(\breve{\mu}_n) - ½\breve{\sigma}_l^2$
3: $\breve{X} = \mathcal{L}(\breve{\mu}_l, \breve{\sigma}_l, \breve{l})$ *(distribution function supported in and provided by AnyLogic)*
4: **return:** $\breve{X}$

Function 4 - Lognormal distribution

C.2.2.5 Conversion from radians to degrees and vice versa



(a) Heading angles expressed in radians *(b) Heading angles expressed in degrees*

Figure 54 – Visualization of the two different ways in which heading angles can be expressed

Function $\mathcal{C}_{DEG}^{RAD}(\breve{\psi}_{RAD})$

Argument	State space	Unit	Description
$\breve{\psi}_{RAD}$	$\langle -\pi, \pi]$	rad	Heading angle expressed in radians, as applied in *AnyLogic*
Variable	**State space**	**Unit**	**Description**
$\breve{\psi}_{DEG}$	$\langle 0, 360]$	deg	Heading angle expressed in degrees, with respect to magnetic north
Parameter	**Value**	**Unit**	**Description**
V	1.3	deg	Magnetic variance at Rome Fiumicino

1: **if** ($\breve{\psi}_{RAD} \geq 0$) **then**
2: $\quad \breve{\psi}_{DEG} = \breve{\psi}_{RAD} \cdot 180 / \pi + 90$
3: **else if** ($\breve{\psi}_{RAD} \leq -½\pi$) **then**
4: $\quad \breve{\psi}_{DEG} = (\breve{\psi}_{RAD} + \pi) \cdot 180 / \pi + 270$
5: **else if** ($\breve{\psi}_{RAD} < 0 \wedge \breve{\psi}_{RAD} > -½\pi$) **then**
6: $\quad \breve{\psi}_{DEG} = (\breve{\psi}_{RAD} + ½\pi) \cdot 180 / \pi$
7: **end if**
8: $\breve{\psi}_{DEG}$ -= V
9: **if** ($\breve{\psi}_{DEG} \leq 0$) **then**
10: $\quad \breve{\psi}_{DEG}$ += 360
11: **end if**
12: **return:** $\breve{\psi}_{DEG}$

Function 5 – Conversion from (AnyLogic) radians to (magnetic) degrees

Function $\mathcal{C}_{\text{RAD}}^{\text{DEG}}(\breve{\psi}_{\text{DEG}})$

Argument	State space	Unit	Description
$\breve{\psi}_{\text{DEG}}$	$\langle 0, 360]$	deg	Heading angle expressed in degrees, with respect to magnetic north
Variable	**State space**	**Unit**	**Description**
$\breve{\psi}_{\text{RAD}}$	$\langle -\pi, \pi]$	rad	Heading angle expressed in radians, as applied in *AnyLogic*
Parameter	**Value**	**Unit**	**Description**
V	1.3	deg	Magnetic variance at Rome Fiumicino

1: $\breve{\psi}_{\text{RAD}} = (\breve{\psi}_{\text{DEG}} - 90 + V) \cdot \pi \,/\, 180$

2: **return:** $\breve{\psi}_{\text{RAD}}$

Function 6 – Conversion from (magnetic) degrees to (AnyLogic) radians

C.2.2.6 Conversion from IAS to TAS and vice versa

Variable	State space	Unit	Description
$\breve{P}$	$\mathbb{R}_+$	Pa	Pressure at $\breve{z}$
$\breve{T}$	$\mathbb{R}$	K	Temperature at $\breve{z}$
$\breve{z}_{\text{C}}$	$\mathbb{R}_+$	m	Altitude
$\breve{\rho}$	$\mathbb{R}_{>0}$	kg·m^{-3}	Density at $\breve{z}$
Parameter	**Value**	**Unit**	**Description**
f_{M}^{FT}	0.3048	-	[ft] to [m] conversion factor
$f_{\text{MPS}}^{\text{KTS}}$	0.5144444	-	[kt] to [m·s^{-1}] conversion factor
$f_{\text{KTS}}^{\text{MPS}}$	1.94384449	-	[m·s^{-1}] to [kt] conversion factor
g_0	9.80665	m·s^{-2}	Acceleration of gravity at sea level
P_0	101325	Pa	Pressure at sea level
R	287.06	J·kg^{-1}·K^{-1}	Specific gas constant
T_0	288.15	K	Temperature at sea level
γ	1.4	-	Ratio of specific heats of air
λ	-0.0065	K·m^{-1}	Lapse rate (temperature with height increase)
ρ_0	1.225	kg·m^{-3}	Density at sea level

Table 25 – Variables and parameters corresponding to functions 7 and 8

Function $\mathcal{C}_{\text{TAS}}^{\text{IAS}}(\breve{v}_{\text{IAS}}, \breve{z})$

Argument	State space	Unit	Description
$\breve{v}_{\text{IAS}}$	$\mathbb{R}_+$	kt	Indicated (calibrated) airspeed
$\breve{z}$	$\mathbb{R}_+$	ft	Altitude
Variable	**State space**	**Unit**	**Description**
$\breve{v}_{\text{IAS,C}}$	$\mathbb{R}_+$	m·s^{-1}	Indicated (calibrated) airspeed
$\breve{v}_{\text{TAS}}$	$\mathbb{R}_+$	kt	True airspeed

1: $\breve{v}_{\text{IAS,C}} = \breve{v}_{\text{IAS}} \cdot f_{\text{MPS}}^{\text{KTS}}$

2: $\breve{z}_{\text{C}} = \breve{z} \cdot f_{\text{M}}^{\text{FT}}$

3: $\breve{T} = T_0 + \lambda \breve{z}$

4: $\breve{P} = P_0 \left(\frac{\breve{T}}{T_0}\right)^{\frac{-g_0}{\lambda R}}$

5: $\breve{\rho} = \frac{\breve{P}}{R \cdot \breve{T}}$

6: $\breve{v}_{\text{TAS}} = \sqrt{\frac{2\lambda}{\lambda - 1} \frac{\breve{P}}{\breve{\rho}} \left[\left[1 + \frac{P_0}{\breve{P}} \left[\left(1 + \frac{\lambda - 1}{2\lambda} \frac{\rho_0}{P_0} \breve{v}_{\text{IAS,C}}^2 \right)^{\frac{\lambda}{\lambda - 1}} - 1 \right] \right]^{\frac{\lambda - 1}{\lambda}} - 1 \right]} \cdot f_{\text{KTS}}^{\text{MPS}}$

7: **return:** $\breve{v}_{TAS}$

Function 7 – Conversion from calibrated airspeed (IAS) to true airspeed (TAS)

Function $\mathcal{C}_{IAS}^{TAS}(\breve{v}_{TAS}, \breve{z})$

Argument	State space	Unit	Description
$\breve{v}_{TAS}$	$\mathbb{R}_+$	kt	True airspeed
$\breve{z}$	$\mathbb{R}_+$	ft	Altitude
Variable	**State space**	**Unit**	**Description**
$\breve{v}_{IAS}$	$\mathbb{R}_+$	kt	Indicated (calibrated) airspeed
$\breve{v}_{TAS,C}$	$\mathbb{R}_+$	m·s^{-1}	True airspeed

1: $\breve{v}_{TAS,C} = \breve{v}_{TAS} \cdot f_{MPS}^{KTS}$

2: $\breve{z}_C = \breve{z} \cdot f_M^{FT}$

3: $\breve{T} = T_0 + \lambda \breve{z}$

4: $\breve{P} = P_0 \left(\frac{\breve{T}}{T_0}\right)^{\frac{-g_0}{\lambda R}}$

5: $\breve{\rho} = \frac{\breve{P}}{R \cdot \breve{T}}$

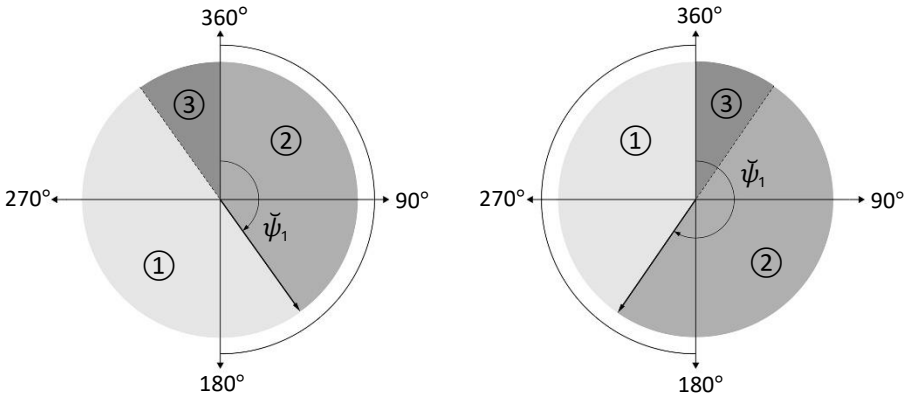
6: $\breve{v}_{IAS} = \sqrt{\frac{2\lambda}{\lambda-1}\frac{P_0}{\rho_0}\left[\left[1 + \frac{\breve{P}}{P_0}\left[\left(1 + \frac{\lambda-1}{2\lambda}\frac{\breve{\rho}}{\breve{P}}\breve{v}_{TAS,C}^2\right)^{\frac{\lambda}{\lambda-1}} - 1\right]\right]^{\frac{\lambda-1}{\lambda}} - 1\right]} \cdot f_{KTS}^{MPS}$

7: **return:** $\breve{v}_{IAS}$

Function 8 – Conversion from true airspeed (TAS) to calibrated airspeed (IAS)

C.2.2.7 Calculate angle between two bearings or radials

Function 9 calculates the angle between two specified heading directions $\breve{\psi}_1$ and $\breve{\psi}_2$, i.e. the direction and magnitude of change between those directions. It always calculates the smallest angle between two heading directions, i.e. the angle that results in the fastest completion of the desired turning movement.



(a) $\breve{\psi}_1 \leq 180$ *(b)* $180 < \breve{\psi}_1$

Figure 55 - Two orientations with different calculations of heading change

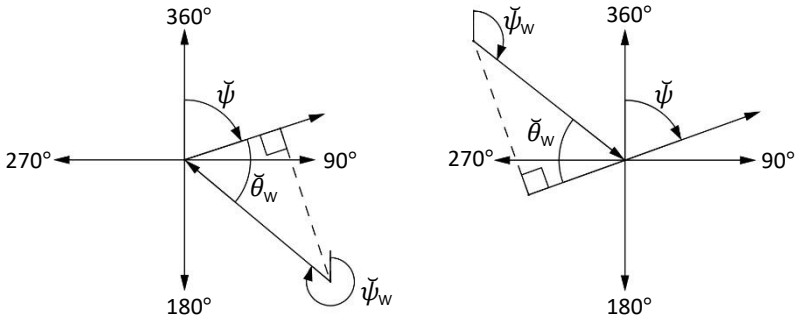
Function $\mathcal{F}_\theta(\breve{\psi}_1, \breve{\psi}_2)$

Argument	State space	Unit	Description
$\breve{\psi}_1$	⟨0, 360]	deg	Initial radial or bearing, i.e. reference radial
$\breve{\psi}_2$	⟨0, 360]	deg	Desired radial or bearing
Variable	**State space**	**Unit**	**Description**
$\breve{\theta}$	[-180, 180]	deg	Angle between $\breve{\psi}_1$ and $\breve{\psi}_2$, $\breve{\theta} > 0$ denotes a clockwise direction of change and $\breve{\theta} < 0$ denotes a counter-clockwise direction of change

1: **if** ($\breve{\psi}_1 == \breve{\psi}_2$) **then**
2: $\breve{\theta} = 0$
3: **else if** ($\breve{\psi}_1 \leq 180 \wedge \breve{\psi}_1 < \breve{\psi}_2 < \breve{\psi}_1 + 180$) **then** *//corresponding to area 1 in figure 55a*
4: $\breve{\theta} = |\breve{\psi}_2 - \breve{\psi}_1|$
5: **else if** ($\breve{\psi}_1 \leq 180 \wedge \breve{\psi}_2 \leq \breve{\psi}_1$) **then** *//corresponding to area 2 in figure 55a*
6: $\breve{\theta} = -|\breve{\psi}_2 - \breve{\psi}_1|$
7: **else if** ($\breve{\psi}_1 \leq 180 \wedge \breve{\psi}_1 + 180 \leq \breve{\psi}_2$) **then** *//corresponding to area 3 in figure 55a*
8: $\breve{\theta} = -|(\breve{\psi}_2 - 180) - (\breve{\psi}_1 + 180)|$
9: **else if** ($180 < \breve{\psi}_1 \wedge \breve{\psi}_1 < \breve{\psi}_2$) **then** *//corresponding to area 1 in figure 55b*
10: $\breve{\theta} = |\breve{\psi}_2 - \breve{\psi}_1|$
11: **else if** ($180 < \breve{\psi}_1 \wedge \breve{\psi}_1 - 180 < \breve{\psi}_2 < \breve{\psi}_1$) **then** *//corresponding to area 2 in figure 55b*
12: $\breve{\theta} = -|\breve{\psi}_2 - \breve{\psi}_1|$
13: **else if** ($180 < \breve{\psi}_1 \wedge \breve{\psi}_2 < \breve{\psi}_1 - 180$) **then** *//corresponding to area 3 in figure 55b*
14: $\breve{\theta} = |(\breve{\psi}_2 + 180) - (\breve{\psi}_1 - 180)|$
15: **end if**
16: **return:** $\breve{\theta}$

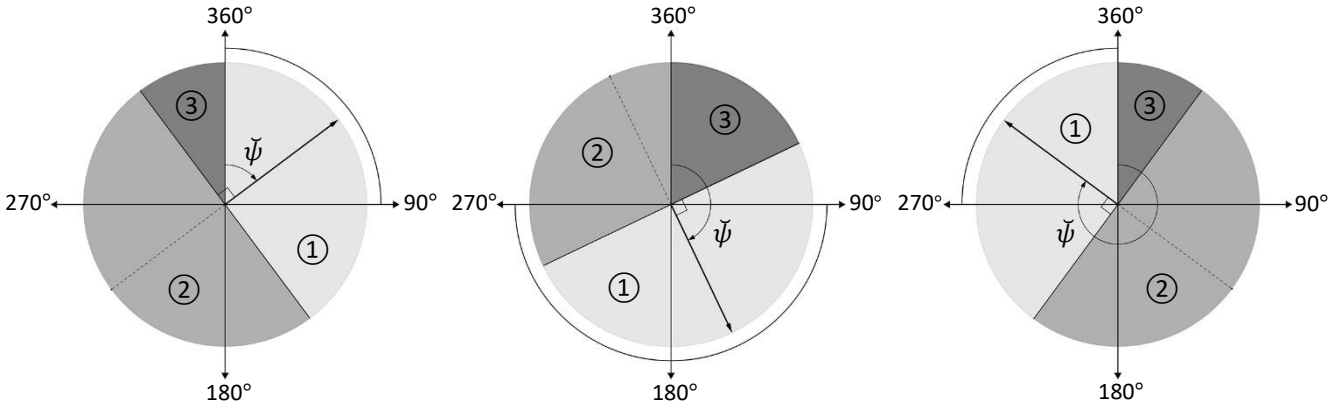
Function 9 - Calculate heading change and direction of change

C.2.2.8 Calculate the wind component



(a) Headwind *(b) Tailwind*

Figure 56 – Two different wind configurations



(a) $\breve{\psi} < 90$ *(b)* $90 \leq \breve{\psi} \leq 270$ *(c)* $270 < \breve{\psi}$

Figure 57 - Three orientations with different wind calculations

Function $\mathcal{F}_{\theta_w}(\breve{\psi}_w, \breve{\psi})$

Argument	State space	Unit	Description
$\breve{\psi}_w$	⟨0, 360]	deg	Direction from which the wind originates, with respect to true north
$\breve{\psi}$	⟨0, 360]	deg	Magnetic direction in which the longitudinal axis of the aircraft is pointed

Variable	State space	Unit	Description
$\breve{\theta}_w$	[-90, 90]	deg	Angle between $\breve{\psi}$ and $\breve{\psi}_w$, i.e. the wind component angle. $\breve{\theta}_w \geq 0$ denotes tailwind and $\breve{\theta}_w < 0$ denotes headwind
Parameter	**Value**	**Unit**	**Description**
V	1.3	deg	Magnetic variance at Rome Fiumicino

$\breve{\psi}$ += V
if ($\breve{\psi} < 90 \wedge 90 + \breve{\psi} \leq \breve{\psi}_w \leq 270 + \breve{\psi}$) **then** *//corresponding to area 2 in figure 57a*
    $\breve{\theta}_w = |(\breve{\psi} + 180) - \breve{\psi}_w|$
else if ($90 \leq \breve{\psi} \leq 270 \wedge \breve{\psi}_w \geq \breve{\psi} + 90$) **then** *//corresponding to area 2 in figure 57b*
    $\breve{\theta}_w = |\breve{\psi} - (\breve{\psi}_w - 180)|$
else if ($90 \leq \breve{\psi} \leq 270 \wedge \breve{\psi}_w \leq \breve{\psi} - 90$) **then** *//corresponding to area 3 in figure 57b*
    $\breve{\theta}_w = |\breve{\psi} - (\breve{\psi}_w + 180)|$
else if ($270 < \breve{\psi} \wedge \breve{\psi} - 270 \leq \breve{\psi}_w \leq \breve{\psi} - 90$) **then** *//corresponding to area 2 in figure 57c*
    $\breve{\theta}_w = |\breve{\psi} - (\breve{\psi}_w + 180)|$
else if ($\breve{\psi} < 90 \wedge \breve{\psi}_w < 90 + \breve{\psi}$) **then** *//corresponding to area 1 in figure 57a*
    $\breve{\theta}_w = -|\breve{\psi} - \breve{\psi}_w|$
else if ($\breve{\psi} < 90 \wedge \breve{\psi}_w > 270 + \breve{\psi}$) **then** *//corresponding to area 3 in figure 57a*
    $\breve{\theta}_w = -|(\breve{\psi} + 180) - (\breve{\psi}_w - 180)|$
else if ($90 \leq \breve{\psi} \leq 270 \wedge |\breve{\psi} - \breve{\psi}_w| < 90$) **then** *//corresponding to area 1 in figure 57b*
    $\breve{\theta}_w = -|\breve{\psi} - \breve{\psi}_w|$
else if ($270 < \breve{\psi} \wedge \breve{\psi}_w < \breve{\psi} - 270$) **then** *//corresponding to area 3 in figure 57c*
    $\breve{\theta}_w = -|(\breve{\psi} - 180) - (\breve{\psi}_w + 180)|$
else if ($270 < \breve{\psi} \wedge \breve{\psi}_w > \breve{\psi} - 90$) **then** *//corresponding to area 1 in figure 57c*
    $\breve{\theta}_w = -|\breve{\psi} - \breve{\psi}_w|$
end if
return: $\breve{\theta}_w$

Function 10 – Calculate wind component angle

C.2.2.9 Get wind speed at specific altitude

Function $\mathcal{F}_{v_w}(\breve{z})$

Argument	State space	Unit	Description
$\breve{z}$	$\mathbb{R}_+$	ft	Specific altitude layer to return the wind speed for
Variable	**State space**	**Unit**	**Description**
v_w^z	$\mathbb{R}_+$	kt	Set of wind speed variables that were assigned for multiple altitudes in action 1 in appendix C.3.3
$\breve{v}_w$	$\mathbb{R}_+$	kt	Wind speed at a specific altitude $\breve{z}$, obtained after applying linear interpolation

if ($0 \leq \breve{z} < 2000$) **then**
    $\breve{v}_w = v_w^{FL000} + (\breve{z} - 0) \cdot (v_w^{FL020} - v_w^{FL000}) / (2000 - 0)$
else if ($2000 \leq \breve{z} < 3000$) **then**
    $\breve{v}_w = v_w^{FL020} + (\breve{z} - 2000) \cdot (v_w^{FL030} - v_w^{FL020}) / (3000 - 2000)$
continue likewise
else if ($30000 \leq \breve{z}$) **then**
    $\breve{v}_w = v_w^{FL300}$
end if
return: $\breve{v}_w$

Function 11 – Return wind speed at specific altitude

C.2.2.10 Get wind direction at specific altitude

Function $\mathcal{F}_{\psi_W}(\check{z})$

Argument	State space	Unit	Description
$\check{z}$	$\mathbb{R}_+$	ft	Specific altitude layer to return the wind direction for
Variable	**State space**	**Unit**	**Description**
ψ_W^z	⟨0, 360]	deg	Set of wind direction variables that were assigned for multiple altitudes in action 1 in appendix C.3.3
$\check{\psi}_W$	⟨0, 360]	deg	Wind direction at a specific altitude $\check{z}$, obtained after applying linear interpolation

if $(0 \leq \check{z} < 2000)$ **then**
    $\check{\psi}_W = \psi_W^{FL000} + (\check{z} - 0) \cdot \mathcal{F}_\theta(\psi_W^{FL000}, \psi_W^{FL020}) \,/\, (2000 - 0)$
else if $(2000 \leq \check{z} < 3000)$ **then**
    $\check{\psi}_W = \psi_W^{FL020} + (\check{z} - 2000) \cdot \mathcal{F}_\theta(\psi_W^{FL020}, \psi_W^{FL030}) \,/\, (3000 - 2000)$
continue likewise
else if $(30000 \leq \check{z})$ **then**
    $\check{\psi}_W = \psi_W^{FL300}$
end if
if $(\check{\psi}_W > 360)$ **then**
    $\check{\psi}_W$ -= 360
else if $(\check{\psi}_W \leq 0)$ **then**
    $\check{\psi}_W$ += 360
end if
return: $\check{\psi}_W$

Function 12 – Return wind direction at specific altitude

C.2.2.11 Calculate ground speed

Function $\mathcal{F}_{v_{GS}}(\check{z}, \check{v}_{IAS}, \check{\psi})$

Argument	State space	Unit	Description
$\check{z}$	$\mathbb{R}_+$	ft	Altitude of aircraft
$\check{v}_{IAS}$	$\mathbb{R}_+$	kt	Indicated (calibrated) airspeed of aircraft
$\check{\psi}$	⟨0, 360]	deg	Heading of aircraft
Variable	**State space**	**Unit**	**Description**
$\check{v}_{GS}$	$\mathbb{R}_+$	kt	Ground speed
$\check{v}_{TAS}$	$\mathbb{R}_+$	kt	True airspeed at altitude $\check{z}$
$\check{v}_W$	$\mathbb{R}_+$	kt	Wind speed at altitude $\check{z}$
$\check{v}_{W,C}$	$\mathbb{R}$	kt	Wind speed component of $\check{v}_W$, parallel to $\check{\psi}$
$\check{\theta}_W$	[-90, 90]	deg	Wind component angle (function 10)
$\check{\psi}_W$	⟨0, 360]	deg	Wind direction at altitude $\check{z}$

$\check{v}_{TAS} = \mathcal{C}_{TAS}^{IAS}(\check{v}_{IAS}, \check{z})$
$\check{v}_W = \mathcal{F}_{v_W}(\check{z})$
$\check{\psi}_W = \mathcal{F}_{\psi_W}(\check{z})$
$\check{\theta}_W = \mathcal{F}_{\theta_W}(\check{\psi}_W, \check{\psi})$
$\check{v}_{W,C} = \check{v}_W \cdot \cos(\check{\theta}_W \cdot \pi \,/\, 180)$
$\check{v}_{GS} = \check{v}_{TAS} + \check{v}_{W,C}$
return: $\check{v}_{GS}$

Function 13 – Calculate and return the ground speed of an aircraft

C.2.2.12 Calculate "direct" bearing

Function atan2($\breve{y}$, $\breve{x}$)

Argument	State space	Unit	Description
$\breve{y}$	$\mathbb{R}$	NM	$\breve{y}$-position of a certain point in the simulation environment
$\breve{x}$	$\mathbb{R}$	NM	$\breve{x}$-position of a certain point in the simulation environment
Variable	**State space**	**Unit**	**Description**
$\breve{\theta}$	$\langle -\pi, \pi]$	rad	Bearing of $(\breve{x}, \breve{y})$ seen from (0, 0) or relative to a different coordinate

1: **if** ($\breve{x} > 0$) **then**
2: $\quad\breve{\theta}$ = arctan($\breve{y}/\breve{x}$)
3: **else if** ($\breve{x} < 0 \wedge \breve{y} \geq 0$) **then**
4: $\quad\breve{\theta}$ = arctan($\breve{y}/\breve{x}$) + π
5: **else if** ($\breve{x} < 0 \wedge \breve{y} < 0$) **then**
6: $\quad\breve{\theta}$ = arctan($\breve{y}/\breve{x}$) – π
7: **else if** ($\breve{x} == 0 \wedge \breve{y} > 0$) **then**
8: $\quad\breve{\theta} = ½\pi$
9: **else if** ($\breve{x} == 0 \wedge \breve{y} < 0$) **then**
10: $\quad\breve{\theta} = -½\pi$
11: **else if** ($\breve{x} == 0 \wedge \breve{y} == 0$) **then**
12: $\quad\breve{\theta}$ = undefined
13: **end if**
14: **return:** $\breve{\theta}$

Function 14 – Calculate the "direct" bearing between two points in the simulation environment

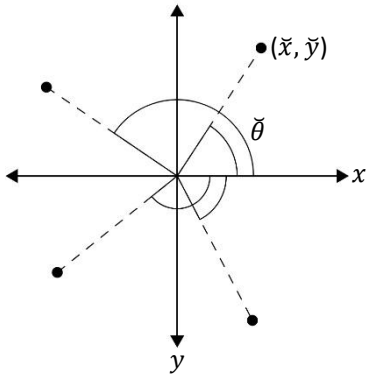


Figure 58 – atan2($\breve{y}$, $\breve{x}$) returning angle $\breve{\theta}$

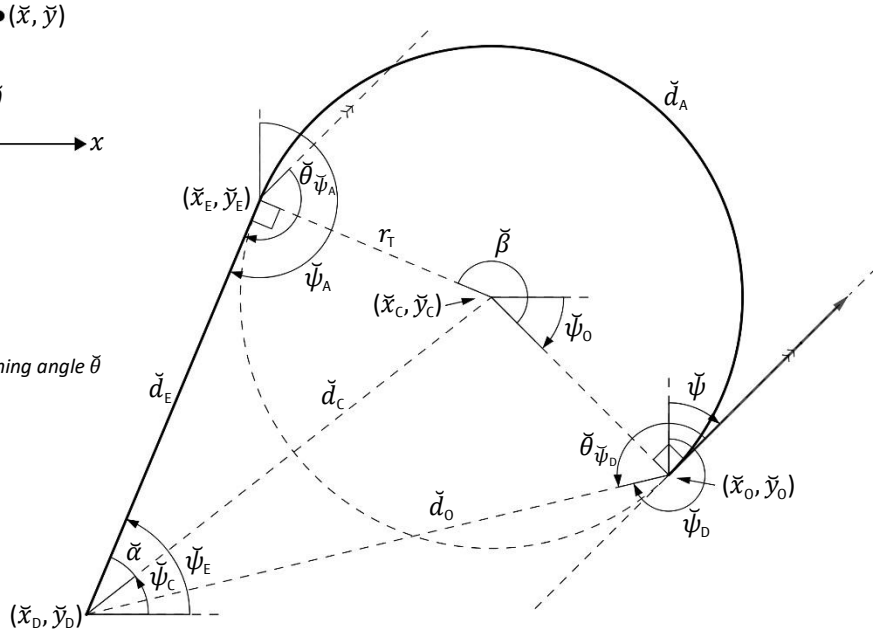


Figure 59 – Overview of the arguments, variables and parameters that correspond to functions 15 and 16

C.2.2.13 Calculate "tangent" bearing

Function $\mathcal{F}_{\psi_A}(\breve{\psi}, \breve{x}_O, \breve{y}_O, \breve{x}_D, \breve{y}_D, r_T)$

Argument	State space	Unit	Description
$\breve{\psi}$	$\langle 0, 360]$	deg	Initial reference heading to start turning from
$\breve{x}_O$	$\mathbb{R}$	NM	Initial x-position, i.e. point of origin
$\breve{y}_O$	$\mathbb{R}$	NM	Initial y-position, i.e. point of origin
$\breve{x}_D$	$\mathbb{R}$	NM	Desired x-position, i.e. point of destination
$\breve{y}_D$	$\mathbb{R}$	NM	Desired y-position, i.e. point of destination
Variable	**State space**	**Unit**	**Description**
$\breve{d}$	$\mathbb{R}_+$	NM	Distance (*tangent*) between $(\breve{x}_D, \breve{y}_D)$ and $(\breve{x}_O, \breve{y}_O)$
$\breve{d}_A$	$\mathbb{R}_+$	NM	Distance that has been travelled over the circular arc
$\breve{d}_C$	$\mathbb{R}_+$	NM	Distance (*direct*) between $(\breve{x}_D, \breve{y}_D)$ and $(\breve{x}_C, \breve{y}_C)$
$\breve{d}_E$	$\mathbb{R}_+$	NM	Distance (*direct*) between $(\breve{x}_D, \breve{y}_D)$ and $(\breve{x}_E, \breve{y}_E)$
$\breve{d}_O$	$\mathbb{R}_+$	NM	Distance (*direct*) between $(\breve{x}_D, \breve{y}_D)$ and $(\breve{x}_O, \breve{y}_O)$
$\breve{x}_C$	$\mathbb{R}$	NM	x-position of the centre of the turning circle
$\breve{x}_E$	$\mathbb{R}$	NM	x-position at which the turning movement should be terminated
$\breve{y}_C$	$\mathbb{R}$	NM	y-position of the centre of the turning circle
$\breve{y}_E$	$\mathbb{R}$	NM	y-position at which the turning movement should be terminated
$\breve{\alpha}$	$\mathbb{R}$	rad	Angle between $\breve{\psi}_C$ and $\breve{\psi}_E$
$\breve{\beta}$	$\langle 0, 360]$	deg	Angle that the aircraft has turned during the turning movement
$\breve{\theta}_{\breve{\psi}_A}$	[-180, 180]	deg	Angle between $\breve{\psi}$ and $\breve{\psi}_A$
$\breve{\theta}_{\breve{\psi}_D}$	[-180, 180]	deg	Angle between $\breve{\psi}$ and $\breve{\psi}_D$
$\breve{\psi}_A$	$\langle 0, 360]$	deg	Bearing (*direct*) of $(\breve{x}_D, \breve{y}_D)$ seen from $(\breve{x}_E, \breve{y}_E)$
$\breve{\psi}_C$	$\langle -\pi, \pi]$	rad	Bearing (*direct*) of $(\breve{x}_C, \breve{y}_C)$ seen from $(\breve{x}_D, \breve{y}_D)$
$\breve{\psi}_D$	$\langle 0, 360]$	deg	Bearing (*direct*) of $(\breve{x}_D, \breve{y}_D)$ seen from $(\breve{x}_O, \breve{y}_O)$
$\breve{\psi}_E$	$\langle -\pi, \pi]$	rad	Bearing (*direct*) of $(\breve{x}_E, \breve{y}_E)$ seen from $(\breve{x}_D, \breve{y}_D)$
$\breve{\psi}_O$	$\mathbb{R}$	rad	Bearing (*direct*) of $(\breve{x}_O, \breve{y}_O)$ seen from $(\breve{x}_C, \breve{y}_C)$
Parameter	**Value**	**Unit**	**Description**
$d_{D,b}$	1	NM	Minimum distance that is required between $(\breve{x}_D, \breve{y}_D)$ and $(\breve{x}_O, \breve{y}_O)$ before a *tangent* distance is calculated, otherwise a *direct* distance is considered
θ_b	1	deg	Minimum angle that is required between $\breve{\psi}$ and $\breve{\psi}_D$ before a *tangent* distance is calculated, otherwise a *direct* distance is considered

1: $\breve{\psi}_D = \mathcal{C}^{RAD}_{DEG}(\text{atan2}(\breve{y}_D - \breve{y}_O, \breve{x}_D - \breve{x}_O))$
2: $\breve{\theta}_{\breve{\psi}_D} = \mathcal{F}_\theta(\breve{\psi}, \breve{\psi}_D)$
3: $\breve{d} = \sqrt{((\breve{x}_D - \breve{x}_O)^2 + (\breve{y}_D - \breve{y}_O)^2)}$
4: $\breve{\psi}_A = \breve{\psi}_D$
5: **if** $(|\breve{\theta}_{\breve{\psi}_D}| > \theta_b \wedge \breve{d} > d_{D,b})$ **then**
6: **if** $(\breve{\theta}_{\breve{\psi}_D} \geq 0)$ **then**
7: $\breve{\psi}_O = \mathcal{C}^{DEG}_{RAD}(\breve{\psi}) - \pi / 2$
8: **else if** $(\breve{\theta}_{\breve{\psi}_D} < 0)$ **then**
9: $\breve{\psi}_O = \mathcal{C}^{DEG}_{RAD}(\breve{\psi}) + \pi / 2$
10: **end if**
11: $\breve{x}_C = \breve{x}_O - r_T \cdot \cos(\breve{\psi}_O)$
12: $\breve{y}_C = \breve{y}_O - r_T \cdot \sin(\breve{\psi}_O)$
13: $\breve{d}_C = \sqrt{((\breve{x}_D - \breve{x}_C)^2 + (\breve{y}_D - \breve{y}_C)^2)}$
14: $\breve{d}_E = \sqrt{(\breve{d}_C^2 - r_T^2)}$
15: $\breve{\psi}_C = \text{atan2}(\breve{y}_C - \breve{y}_D, \breve{x}_C - \breve{x}_D)$
16: $\breve{\alpha} = \text{asin}(r_T / \breve{d}_C)$
17: $\breve{\psi}_E = \breve{\psi}_C$
18: **if** $(\breve{\theta}_{\breve{\psi}_D} \geq 0)$ **then**

19: $\breve{\psi}_E$ += $\breve{\alpha}$
20: **else if** ($\breve{\theta}_{\breve{\psi}_D} < 0$) **then**
21: $\breve{\psi}_E$ -= $\breve{\alpha}$
22: **end if**
23: $\breve{x}_E = \breve{x}_D + \breve{d}_E \cdot \cos(\breve{\psi}_E)$
24: $\breve{y}_E = \breve{y}_D + \breve{d}_E \cdot \sin(\breve{\psi}_E)$
25: $\breve{\psi}_A = \mathcal{C}_{DEG}^{RAD}(\text{atan2}(\breve{y}_D - \breve{y}_E, \breve{x}_D - \breve{x}_E))$
26: $\breve{\theta}_{\breve{\psi}_A} = \mathcal{F}_\theta(\breve{\psi}, \breve{\psi}_A)$
27: **if** $((\breve{\theta}_{\breve{\psi}_D} \geq 0 \wedge \breve{\theta}_{\breve{\psi}_A} \geq 0) \vee (\breve{\theta}_{\breve{\psi}_D} < 0 \wedge \breve{\theta}_{\breve{\psi}_A} < 0))$ **then**
28: $\breve{\beta} = |\breve{\theta}_{\breve{\psi}_A}|$
29: **else if** $((\breve{\theta}_{\breve{\psi}_D} \geq 0 \wedge \breve{\theta}_{\breve{\psi}_A} < 0) \vee (\breve{\theta}_{\breve{\psi}_D} < 0 \wedge \breve{\theta}_{\breve{\psi}_A} \geq 0))$ **then**
30: $\breve{\beta} = 360 - |\breve{\theta}_{\breve{\psi}_A}|$
31: **end if**
32: $\breve{d}_A = 2 \cdot \pi \cdot r_T \cdot \breve{\beta} \,/\, 360$
33: $\breve{d} = \breve{d}_E + \breve{d}_A$
34: **end if**
35: **return:** $\breve{\psi}_A$

Function 15 – Calculate "tangent" bearing between two points in the simulation environment

C.2.2.14 Calculate "tangent" distance

Function $\mathcal{F}_d(\breve{\psi}, \breve{x}_O, \breve{y}_O, \breve{x}_D, \breve{y}_D, r_T)$

See function 15 for information about the arguments, variables and parameters used

1: $\mathcal{F}_{\psi_A}(\breve{\psi}, \breve{x}_O, \breve{y}_O, \breve{x}_D, \breve{y}_D, r_T)$
2: **Return:** $\breve{d}$

Function 16 – Calculate "tangent" distance between two points in the simulation environment

C.2.2.15 Determine the required wake vortex separation distance

Function $\mathcal{F}_{s_S}(\breve{W}_1, \breve{W}_2)$

Argument	State space	Unit	Description
$\breve{W}_1$	$\mathbb{W}$	-	The WTC of the preceding aircraft
$\breve{W}_2$	$\mathbb{W}$	-	The WTC of the succeeding aircraft
Variable	**State space**	**Unit**	**Description**
$\breve{s}_S$	$\mathbb{R}_+$	NM	Required wake vortex separation distance between $\breve{W}_1$ and $\breve{W}_2$
Parameter	**Value**	**Unit**	**Description**
s_{MM}	3	NM	Wake vortex separation minima between MEDIUM and MEDIUM WTCs
s_{HM}	5	NM	Wake vortex separation minima between HEAVY and MEDIUM WTCs
s_{MH}	3	NM	Wake vortex separation minima between MEDIUM and HEAVY WTCs
s_{HH}	4	NM	Wake vortex separation minima between HEAVY and HEAVY WTCs

1: **if** ($\breve{W}_1 == W_M \wedge \breve{W}_2 == W_M$) **then**
2: $\breve{s}_S = s_{MM}$
3: **else if** ($\breve{W}_1 == W_H \wedge \breve{W}_2 == W_M$) **then**
4: $\breve{s}_S = s_{HM}$
5: **else if** ($\breve{W}_1 == W_M \wedge \breve{W}_2 == W_H$) **then**
6: $\breve{s}_S = s_{MH}$
7: **else if** ($\breve{W}_1 == W_H \wedge \breve{W}_2 == W_H$) **then**
8: $\breve{s}_S = s_{HH}$
9: **end if**
10: **return:** $\breve{s}_S$

Function 17 – Return ICAO wake vortex separation distance between aircraft pair

C.2.2.16 Floor function

The floor() function returns the largest integer less than or equal to a given number. The floor() can be mathematically described as $\max\{m \in \mathbb{Z} |\ m \leq x\}$, where $x \in \mathbb{R}$.

C.2.3 Statechart: communication of capacity updates

The communication between agents about capacity updates has been modelled similarly. The concerned agents can either send or receive capacity updates. This section provides a general description of how this specific communication has been modelled and is meant to prevent duplicate descriptions in the model specification.

9.3.1.1 Sending a capacity update

The sending of a capacity update is modelled using the structure of the ⌊*Contact with agent about capacity update*⌋ ⌊*sending agent*⌋ statechart (figure 60). All the statecharts of the multiple agents that are related to the sending of a capacity update have a similar structure as that of the ⌊*Contact with agent about capacity update*⌋⌊*sending agent*⌋ statechart.

⟨no contact required with agent⟩
Represents the situation in which there is no need to send a capacity update to another agent, i.e. the sending agent is not informed about a capacity change or has not observed a change in the weather conditions at the airport.

⟨contact required with agent⟩
Represents the situation in which the situation awareness of the agent has been updated with the changed weather conditions. This update can be achieved either by observation or by communication. The notified agent will immediately try to contact the agent(s) that should also be informed about the changed capacity at the airport.

⟨agent has been informed⟩
Indicates that the capacity update has been sent to the receiving agent, which marks a finished and irreversible communication between the two concerning agents about the capacity update.

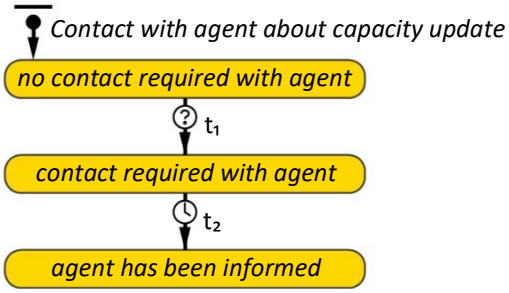


Figure 60 – ⌊Contact with agent about capacity update⌋⌊sending agent⌋

Transition t_1
Transition t_1 is taken once the sending agent has been informed about either changed weather conditions or a capacity update

Transition t_2
Transition t_2 models the time period that is required by the sending agent to make contact and to inform the receiving agent about the capacity update. The durations of all capacity updates have been modelled with the same distribution. This time period comprises both the sending of the capacity update and the reception of a confirmation from the receiving agent.

- Timeout: t_{INF}
- Action:

1: $t_{INF} = \mathcal{L}(\mu_{t_{INF}}, \sigma_{t_{INF}}, l_{t_{INF}})$

9.3.1.2 Receiving a capacity update

The receiving of a capacity update is modelled using the structure of the ⌊*Contact with agent about capacity update*⌋ ⌊*receiving agent*⌋ statechart (figure 61). All the statecharts of the multiple agents that are related to the reception of a capacity update have a similar structure as that of the ⌊*Contact with agent about capacity update*⌋⌊*receiving agent*⌋ statechart.

⟨on stand-by for capacity update⟩
Represents the situation in which the receiving agent waits for an incoming message from the sending agent about a capacity update.

⟨capacity update received⟩
Indicates that the receiving agent is informed about the reduced runway capacity at the airport, which marks the finished and irreversible contact between the receiving and sending agent.

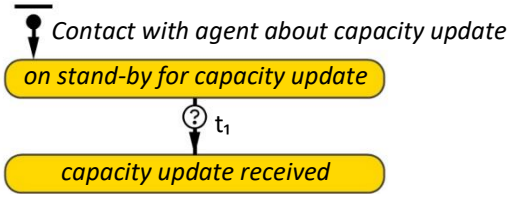


Figure 61 – ⌊Contact with agent about capacity update⌋⌊receiving agent⌋

Transition t_1
Transition t_1 is taken once the sending agent has informed the receiving agent about a capacity update.

C.3 Environment

C.3.1 Coordinate system

The simulation environment is built on a local tangent plane, which prevents the necessity to take into account the curvature of the earth. This assumption is valid, since the considered approach sectors are relatively small. All positions/coordinates are in the model specified according to the coordinate system as applied in *AnyLogic* (figure 62) to make the specification and implementation more consistent. The positive direction of rotation is clockwise in the coordinate system as applied in *AnyLogic*. The axis origin, i.e. the origin of the simulation environment is defined at coordinate 42°53'22"N 10°46'41"E.

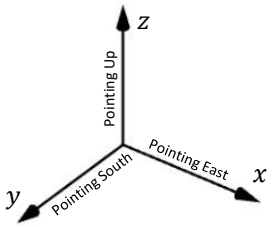


Figure 62 – Coordinate system used in AnyLogic

C.3.2 Parameters

Tables 26-27 provide all the relevant parameter values of the multiple significant points. The coordinates of each significant point in the xy-plane relative to the axis origin are defined by the Haversine formula.

Haversine formula

Argument	State space	Unit	Description
$\check{\lambda}_w$	[-180, 180]	deg	Longitude of the significant point
$\check{\varphi}_w$	[-90, 90]	deg	Latitude of the significant point
Variable	**State space**	**Unit**	**Description**
$\check{x}_w$	$\mathbb{R}$	NM	x-position of the significant point relative to the origin (0,0)
$\check{y}_w$	$\mathbb{R}$	NM	y-position of the significant point relative to the origin (0,0)

Parameter	Value	Unit	Description
$f_{\text{NM}}^{\text{KM}}$	0.539956	-	[km] to [NM] conversion factor
r_{E}	6371	km	Radius of the earth
λ_{R}	10.77806	deg	Longitude of the reference coordinate, i.e. the specified origin
φ_{R}	42.88944	deg	Latitude of the reference coordinate, i.e. the specified origin

1: $\breve{x}_w = r_{\text{E}} \cdot 2\text{asin}(\sqrt{(\cos(\breve{\varphi}_w) \cdot \cos(\varphi_{\text{R}}) \cdot \sin^2(½\breve{\lambda}_w - ½\lambda_{\text{R}}))}) \cdot f_{\text{NM}}^{\text{KM}}$ *//arcsin is in radians*

2: $\breve{y}_w = r_{\text{E}} \cdot 2\text{asin}(\sqrt{(\sin^2(½\breve{\varphi}_w - ½\varphi_{\text{R}}))}) \cdot f_{\text{NM}}^{\text{KM}}$ *//arcsin is in radians*

3: **return:** $\breve{x}_w$, $\breve{y}_w$

Function 18 – Adjusted Haversine formula to calculate the coordinates of each significant point within the xy-plane relative to the origin (0,0)

Parameter	Value	Unit	Description
$d_{w_{\text{THR}},w_{S,k}}$	Table 29	NM	Distance between w_{THR} and $w_{S,k}$ along the legs of the STAR procedure
$d_{w_{\text{THR}},w_{\text{IF}}}$	10.7	NM	Distance between w_{THR} and w_{IF}
$k_{I_4,S}$	Table 28	-	Waypoint (number) that specifies $w_{S,\text{MO}}$
$k_{I_5,S}$	Table 28	-	Waypoint (number) that specifies $w_{S,\text{MI}}$
$k_{I_9,S}$	Table 28	-	Waypoint (number) that specifies $w_{S,\text{R}}$
$k_{I_{10},S}$	Table 28	-	Waypoint (number) that specifies $w_{S,\text{HO}}$
$k_{I_{15},S}$	Table 28	-	Waypoint (number) that specifies $w_{S,H,\text{I}}$
K_{S_1}	13	-	Number of waypoints in S_1, including the entry point
K_{S_2}	12	-	Number of waypoints in S_2, including the entry point
$n_{w_{S,k}}$	Table 29	-	Name of $w_{S,k}$ (only used for clarification in the table 29)
$n_{w_{S,H}}$	Table 27	-	Name of $w_{S,H}$ (only used for clarification in the table 27)
$v_{\text{IAS},w_{S,k}}$	Table 29	kt	Speed limit (IAS) at $w_{S,k}$
$v_{\text{IAS},w_{S,H}}$	Table 27	kt	Speed limit (IAS) in holding stack
$v_{\text{IAS},w_{\text{IF}}}$	200	kt	Speed limit (IAS) at w_{IF}
$v_{\text{IAS},I_{14}}$	185	kt	Go-around speed
$x_{w_{S,k}}$	Table 29	NM	x-position of $w_{S,k}$
$x_{w_{S,\text{MI}}}$	$x_{w_{S,k}}$	NM	x-position of $w_{S,\text{MI}}$, where k == $k_{I_5,S}$
$x_{w_{I_2,S,k}}$	Table 29	NM	x-position of $w_{I_2,S,k}$
$x_{w_{I_4,S_1}}$	42.0	NM	x-position of w_{I_4,S_1}
$x_{w_{I_4,S_2}}$	55.0	NM	x-position of w_{I_4,S_2}
$x_{w_{I_6,S_1,1}}$	81.0	NM	x-position of $w_{I_6,S_1,1}$
$x_{w_{I_6,S_1,2}}$	54.0	NM	x-position of $w_{I_6,S_1,2}$
$x_{w_{I_6,S_2,1}}$	30.0	NM	x-position of $w_{I_6,S_2,1}$
$x_{w_{I_6,S_2,2}}$	74.0	NM	x-position of $w_{I_6,S_2,2}$
$x_{w_{I_8,S_1}}$	54.0	NM	x-position of w_{I_8,S_1}
$x_{w_{I_8,S_2}}$	74.0	NM	x-position of w_{I_8,S_2}
$x_{w_{S,H,\text{I}}}$	Table 27	NM	x-position of $w_{S,H,\text{I}}$
$x_{w_{S,H,\text{O}}}$	Table 27	NM	x-position of $w_{S,H,\text{O}}$
$x_{w_{\text{IF}}}$	62.6	NM	x-position of w_{IF}
$x_{w_{\text{THR}}}$	65.8	NM	x-position of w_{THR}
$x_{w_{\text{HST1}}}$	66.1	NM	x-position of w_{HST1}
$x_{w_{\text{HST2}}}$	66.2	NM	x-position of w_{HST2}
$y_{w_{S,k}}$	Table 29	NM	y-position of $w_{S,k}$
$y_{w_{S,\text{MI}}}$	$y_{w_{S,k}}$	NM	y-position of $w_{S,\text{MI}}$, where k == $k_{I_5,S}$
$y_{w_{I_2,S,k}}$	Table 29	NM	y-position of $w_{I_2,S,k}$
$y_{w_{I_4,S_1}}$	19.0	NM	y-position of w_{I_4,S_1}
$y_{w_{I_4,S_2}}$	15.0	NM	y-position of w_{I_4,S_2}

$y_{w_{I_6,S_1,1}}$	32.0	NM	y-position of $w_{I_6,S_1,1}$
$y_{w_{I_6,S_1,2}}$	68.0	NM	y-position of $w_{I_6,S_1,2}$
$y_{w_{I_6,S_2,1}}$	48.0	NM	y-position of $w_{I_6,S_2,1}$
$y_{w_{I_6,S_2,2}}$	74.0	NM	y-position of $w_{I_6,S_2,2}$
$y_{w_{I_8,S_1}}$	68.0	NM	y-position of w_{I_8,S_1}
$y_{w_{I_8,S_2}}$	62.0	NM	y-position of w_{I_8,S_2}
$y_{w_{S,H,\mathrm{I}}}$	Table 27	NM	y-position of $w_{S,H,\mathrm{I}}$
$y_{w_{S,H,\mathrm{O}}}$	Table 27	NM	y-position of $w_{S,H,\mathrm{O}}$
$y_{w_{\mathrm{IF}}}$	56.5	NM	y-position of w_{IF}
$y_{w_{\mathrm{THR}}}$	66.7	NM	y-position of w_{THR}
$y_{w_{\mathrm{HST1}}}$	67.8	NM	y-position of w_{HST1}
$y_{w_{\mathrm{HST2}}}$	68.1	NM	y-position of w_{HST2}
$z_{w_{S,k}}$	Table 29	ft	Altitude constraint at $w_{S,k}$
$z_{w_{S,H}}$	Table 27	ft	Altitude (minimum) in holding stack
$z_{w_{\mathrm{IF}}}$	3000	ft	Altitude constraint at w_{IF}
$z_{I_{14}}$	2000	ft	Go-around altitude
$\lambda_{w_{S,k}}$	Table 29	deg	Longitude of $w_{S,k}$
$\lambda_{w_{\mathrm{IF}}}$	12.1903	deg	Longitude of w_{IF}
$\lambda_{w_{\mathrm{THR}}}$	12.2614	deg	Longitude of w_{THR}
$\lambda_{w_{\mathrm{HST1}}}$	12.2692	deg	Longitude of w_{HST1}
$\lambda_{w_{\mathrm{HST2}}}$	12.2717	deg	Longitude of w_{HST2}
$\varphi_{w_{S,k}}$	Table 29	deg	Latitude of $w_{S,k}$
$\varphi_{w_{\mathrm{IF}}}$	42.0158	deg	Latitude of w_{IF}
$\varphi_{w_{\mathrm{THR}}}$	41.8458	deg	Latitude of w_{THR}
$\varphi_{w_{\mathrm{HST1}}}$	41.8275	deg	Latitude of w_{HST1}
$\varphi_{w_{\mathrm{HST2}}}$	41.8217	deg	Latitude of w_{HST2}
$\psi_{w_{S,k}}$	Table 29	deg	Magnetic track of $w_{S,k}$, i.e. bearing of $w_{S,k}$ seen from $w_{S,k-1}$
$\psi_{w_{S,H,\mathrm{I}}}$	Table 27	deg	Magnetic track of the inbound leg towards $w_{S,H,\mathrm{I}}$
$\psi_{w_{S,H,\mathrm{O}}}$	Table 27	deg	Magnetic track of the outbound leg towards $w_{S,H,\mathrm{O}}$
$\psi_{w_{\mathrm{THR}}}$	161	deg	Magnetic track of w_{THR}, i.e. bearing of w_{THR} seen from w_{IF}
$\psi_{w_{\mathrm{HST}}}$	191	deg	Magnetic direction of both w_{HST1} and w_{HST2}

Table 26 – Multiple parameter values that describe essential information about each significant point in the model.

		$n_{w_{S,H}}$		$x_{w_{S,H,\mathrm{I}}}$		$x_{w_{S,H,\mathrm{O}}}$		$y_{w_{S,H,\mathrm{I}}}$		$y_{w_{S,H,\mathrm{O}}}$		$z_{w_{S,H}}$		$v_{\mathrm{IAS},w_{S,H}}$		$\psi_{w_{S,H,\mathrm{I}}}$		$\psi_{w_{S,H,\mathrm{O}}}$	
		S_1	S_2	S_1	S_2	S_1	S_2	S_1	S_2	S_1	S_2	S_1	S_2	S_1	S_2	S_1	S_2	S_1	S_2
H	H_1	GOPOL	RAVUX	35.5	69.0	28.3	62.2	33.2	28.8	36.2	25.0	9000	10000	220	220	96	276	148	328

Table 27 – Parameter values that define holding H ($H \in \mathbb{H}_S$) of S ($S \in \mathbb{S}$), i.e. $w_{S,H,\mathrm{I}}$ and $w_{S,H,\mathrm{O}}$

$k_{I_4,S}$		$k_{I_5,S}$		$k_{I_9,S}$		$k_{I_{10},S}$		$k_{I_{15},S}$	
S_1	S_2	S_1	S_2	S_1	S_2	S_1	S_2	S_1	S_2
8	7	9	8	5	4	4	3	2	1

Table 28 – Waypoint numbers of the STAR procedure that define a specific instruction or airspace section

	$n_{w_{S,k}}$		$\varphi_{w_{S,k}}$		$\lambda_{w_{S,k}}$		$x_{w_{S,k}}$		$y_{w_{S,k}}$		
	S_1	S_2	S_1	S_2	S_1	S_2	S_1	S_2	S_1	S_2	
0	XIBIL	RITEB	42.7969	42.6986	10.9839	12.1636	9.1	61.0	9.6	15.5	
1	IPLEK	RAVUX	42.5442	42.4764	11.5167	12.3411	32.6	69.0	24.7	28.8	
2	GOPOL	LIGBU	42.4036	42. 3614	11.5828	12.4325	35.5	73.1	33.2	35.7	
3	GIXOM	ESALU	42.2619	42.2197	11.6489	12.4922	38.5	75.8	41.7	44.2	
4	USIRU	VAKAB	42.0475	42.1850	11.7411	12.3422	42.7	69.2	54.6	46.3	
5	GIPAP	RF442	42.0831	42.2644	11.8906	12.3086	49.3	67.7	52.4	41.5	
k 6	RF422	RF443	42.1625	42.3439	11.8567	12.2753	47.7	66.2	47.6	36.8	
7	RF423	RF444	42.2419	42.4236	11.8228	12.2417	46.2	64.6	42.9	32.0	
8	RF424	RF446	42.3211	42.3886	11.7886	12.0911	44.7	58.0	38.1	34.1	
9	RF426	RF447	42.3567	42.3089	11.9386	12.1250	51.3	59.5	36.0	38.9	
10	RF427	RF448	42.2772	42.2294	11.9725	12.1586	52.8	61.1	40.8	43.6	
11	RF428	EXAMA	42.1978	42.1500	12.0064	12.1922	54.3	62.6	45.5	48.4	
12	SUVOK	N/A	42.1183	N/A	12.0403	N/A	55.9	N/A	50.3	N/A	

	$z_{w_{S,k}}$		$v_{\text{IAS},w_{S,k}}$		$\psi_{w_{S,k}}$		$d_{w_{\text{THR}},w_{S,k}}$		$x_{w_{I_2,S,k}}$		$y_{w_{I_2,S,k}}$	
	S_1	S_2	S_1	S_2	S_1	S_2	S_1	S_2	S_1	S_2	S_1	S_2
0	19000	17000	250	250	N/A	N/A	123.2	95.2	N/A	N/A	N/A	N/A
1	9000	10000	250	230	122	148	95.2	79.7	46	98	6	24
2	9000	9000	250	230	159	148	86.3	71.7	7	99	42	33
3	9000	6000	230	220	159	161	77.3	62.7	6	103	52	50
4	6000	6000	220	220	161	251	63.7	55.8	5	80	63	64
5	6000	6000	220	220	71	341	56.8	50.8	52	70	72	22
k 6	6000	6000	220	220	341	341	51.8	45.8	32	70	31	22
7	6000	6000	220	200	341	341	46.8	40.8	32	70	31	22
8	6000	6000	200	200	341	251	41.8	33.8	32	52	31	16
9	6000	6000	200	200	71	161	34.8	28.8	46	74	18	62
10	6000	6000	200	200	161	161	29.8	23.8	54	74	68	62
11	6000	4000	200	200	161	161	24.8	18.8	54	74	68	62
12	4000	N/A	200	N/A	161	N/A	19.8	N/A	54	N/A	68	N/A

Table 29 – Multiple parameter values that describe essential information about $w_{S,k}$

C.3.3 Wind model

Variable	State space	Unit	Description
v_{W}^{z}	$\mathbb{R}_{+}$	kt	Wind speed random variable at altitude z *
ψ_{W}^{z}	⟨0, 360]	deg	Wind direction random variable at altitude z *
Parameter	**Value**	**Unit**	**Description**
Δv_{W}	5	kt	Wind speed margin that is both added to and subtracted from the mean $\mu_{v_{\text{W}}^{z}}$, where the total range is used to pick a random sample from
$\Delta\psi_{\text{W}}$	20	deg	Wind direction margin that is both added to and subtracted from the mean $\mu_{\psi_{\text{W}}^{z}}$, where the total range is used to pick a random sample from
$\mu_{v_{\text{W}}^{z}}$	Table 31	kt	Mean (i.e. baseline) wind speed at altitude layer z *
$\mu_{\psi_{\text{W}}^{z}}$	Table 31	deg	Mean (i.e. baseline) wind direction at altitude layer z *

* where z is expressed in flight levels, corresponding to the flight levels that are provided in table 31.

Table 30 – Parameters that describe the default wind speed and -direction at a specific altitude layer

	$\mu_{v_w^z}$	$\mu_{\psi_w^z}$
FL000	6	220
FL020	7	210
FL030	6	210
FL050	3	135
FL064	3	340
FL100	16	340
FL140	20	350
FL180	20	10
FL240	28	360
FL300	18	10

Table 31 – Wind speed and -direction data that corresponds to a specific altitude layer (flight level), serving as default input for the wind model

1: $v_w^{FL000} = |\mathcal{U}(\mu_{v_w^{FL000}} - \Delta v_w,\ \mu_{v_w^{FL000}} + \Delta v_w)|$
2: $v_w^{FL020} = |\mathcal{U}(\mu_{v_w^{FL020}} - \Delta v_w,\ \mu_{v_w^{FL020}} + \Delta v_w)|$
3: *continue likewise*
4: $v_w^{FL300} = |\mathcal{U}(\mu_{v_w^{FL300}} - \Delta v_w,\ \mu_{v_w^{FL300}} + \Delta v_w)|$
5: $\psi_w^{FL000} = \mathcal{F}_{\psi_{w,c}}(\mathcal{U}(\mu_{\psi_w^{FL000}} - \Delta\psi_w,\ \mu_{\psi_w^{FL000}} + \Delta\psi_w))$
6: $\psi_w^{FL020} = \mathcal{F}_{\psi_{w,c}}(\mathcal{U}(\mu_{\psi_w^{FL020}} - \Delta\psi_w,\ \mu_{\psi_w^{FL020}} + \Delta\psi_w))$
7: *continue likewise*
8: $\psi_w^{FL300} = \mathcal{F}_{\psi_{w,c}}(\mathcal{U}(\mu_{\psi_w^{FL300}} - \Delta\psi_w,\ \mu_{\psi_w^{FL300}} + \Delta\psi_w))$

Action 1 – Assigning values to the wind speed (v_w^z) and wind direction (ψ_w^z) variables, chosen from a uniform distribution $\mathcal{U}(\check{l}, \check{u})$

Function $\mathcal{F}_{\psi_{w,c}}(\check{\psi}_w)$

Argument	State space	Unit	Description
$\check{\psi}_w$	$\mathbb{R}$	deg	Wind direction that will be corrected if found to be outside the range ⟨0, 360]
Variable	**State space**	**Unit**	**Description**
$\check{\psi}_{w,C}$	⟨0, 360]	deg	Corrected wind direction

1: $\check{\psi}_{w,C} = \check{\psi}_w$
2: **if** ($\check{\psi}_w > 360$) **then**
3: $\quad\check{\psi}_{w,C}$ -= 360
4: **else if** ($\check{\psi}_w \leq 0$) **then**
5: $\quad\check{\psi}_{w,C}$ += 360
6: **end if**
7: **return:** $\check{\psi}_{w,C}$

Function 19 – Correct wind direction when being outside of the viable range of ⟨0, 360]

C.4 Executive controller

C.4.1 Variables, parameters and sets

Some of the variables in table 32 below are dependent on and defined by the STAR procedure that a_i is/has been operating (i.e. $S^{a_i,c}$), by the specific waypoint where a_i is referenced to (i.e. $k^{a_i,c}$) or by a specific index that defines the significant point where a_i will be vectored to when flying the *vector inbound IF* instruction (i.e. $N^{a_i,c}$). These dependent variables contain the subscripts "S", "k", "N" respectively, where $S = S^{a_i,c}$, $k = k^{a_i,c}$ and $N = N^{a_i,c}$.

Variable	State space	Unit	Description
a_{C}^{c}	$\mathbb{A}^{c}$	-	Aircraft where c is in contact with
a_{I}^{c}	$\mathbb{A}^{c}$	-	Aircraft that is available/required to receive I ($I \in \mathbb{I}^{c}$), as identified by c
$b_{\mathrm{A},I_3}^{a_i,c}$	*Boolean*	-	Boolean that indicates if a_i is available for a *vector inbound STAR* instruction
$b_{\mathrm{A},I_5}^{a_i,c}$	*Boolean*	-	Boolean that indicates if a_i is available for a *vector inbound merge* instruction
$b_{\mathrm{A},I_7}^{a_i,c}$	*Boolean*	-	Boolean that indicates if a_i is available for a *vector inbound IF* instruction
$b_{\mathrm{A},I_9}^{a_i,c}$	*Boolean*	-	Boolean that indicates if a_i is available for a *vector inbound trombone* instruction with respect to both $a_{4,i}$ and $a_{5,i}$
$b_{\mathrm{A},I_9}^{a_{4,i},c}$	*Boolean*	-	Boolean that indicates if a_i is available for a *vector inbound trombone* instruction with respect to $a_{4,i}$
$b_{\mathrm{A},I_9}^{a_{5,i},c}$	*Boolean*	-	Boolean that indicates if a_i is available for a *vector inbound trombone* instruction with respect to $a_{5,i}$
$b_{\mathrm{A},I_{10}}^{a_i,c}$	*Boolean*	-	Boolean that indicates if a_i is available for the *handover to ARR* instruction
$b_{\mathrm{A},I_{11}}^{a_i,c}$	*Boolean*	-	Boolean that indicates if a_i is available for the *handover to TWR* instruction
$b_{\mathrm{A},I_{12}}^{a_i,c}$	*Boolean*	-	Boolean that indicates if a_i is available for the *handover to GND* instruction
$b_{\mathrm{A},I_{13}}^{a_i,c}$	*Boolean*	-	Boolean that indicates if a_i is available for the *landing clearance* instruction
$b_{\mathrm{A},I_{15}}^{a_i,c}$	*Boolean*	-	Boolean that indicates if a_i is available for the *holding entry* instruction
$b_{\mathrm{A},I_{16}}^{a_i,c}$	*Boolean*	-	Boolean that indicates if a_i is available for the *holding exit* instruction
$b_{\mathrm{R},I_{2,1}}^{a_i,c}$	*Boolean*	-	Boolean that indicates if a_i has a *separation minima* conflict with $a_{1,i}$
$b_{\mathrm{R},I_{2,2}}^{a_i,c}$	*Boolean*	-	Boolean that indicates if a_i has a *desired spacing* conflict with $a_{1,i}$
$b_{\mathrm{R},I_4}^{a_i,c}$	*Boolean*	-	Boolean that indicates if a_i requires a *vector outbound merge* instruction
$b_{\mathrm{R},I_{6,1}}^{a_i,c}$	*Boolean*	-	Boolean that indicates if a_i has a *separation minima* conflict with $a_{2,i}$
$b_{\mathrm{R},I_{6,2}}^{a_i,c}$	*Boolean*	-	Boolean that indicates if a_i has a *desired spacing* conflict with $a_{2,i}$
$b_{\mathrm{R},I_8}^{a_i,c}$	*Boolean*	-	Boolean that indicates if a_i requires a *vector outbound trombone* instruction
$b_{\mathrm{R},I_{14}}^{a_i,c}$	*Boolean*	-	Boolean that indicates if a_i requires a *go-around* instruction
$b_{\mathrm{R},I_{15}}^{a_i,c}$	*Boolean*	-	Boolean that indicates if a_i requires a *holding entry* instruction
$b_{\mathrm{R},I_{17}}^{a_i,c}$	*Boolean*	-	Boolean that indicates if a_i requires a *holding altitude* instruction
$d_{\mathrm{A},w_{S,k}}^{a_i,c}$	$\mathbb{R}_+$	NM	Distance (*tangent*) between a_i and $w_{S,k}$
$d_{\mathrm{D},w_{S,H,\mathrm{I}}}^{a_i,c}$	$\mathbb{R}_+$	NM	Distance (*direct*) between a_i and $w_{S,H,\mathrm{I}}$
$d_{\mathrm{D},w_{\mathrm{IF}}}^{a_i,c}$	$\mathbb{R}_+$	NM	Distance (*direct*) between a_i and w_{IF}
$d_{\mathrm{V},w_{\mathrm{IF}}}^{a_i,c}$	$\mathbb{R}_+$	NM	Distance (*via*) between a_i and w_{IF} via $w_{S,\mathrm{MI}}$
$d_{\mathrm{D},w_{\mathrm{IF}}}^{a_{2,i},c}$	$\mathbb{R}_+$	NM	Similar description as $d_{\mathrm{D},w_{\mathrm{IF}}}^{a_i,c}$, but applied to $a_{2,i}$
$d_{\mathrm{V},w_{\mathrm{IF}}}^{a_{2,i},c}$	$\mathbb{R}_+$	NM	Similar description as $d_{\mathrm{V},w_{\mathrm{IF}}}^{a_i,c}$, but applied to $a_{2,i}$
$d_{\mathrm{V},w_{\mathrm{THR}}}^{a_i,c}$	$\mathbb{R}_+$	NM	Distance between a_i and w_{THR}, defined by the operational state of a_i
$d_{\mathrm{V},w_{\mathrm{THR}}}^{a_{1,i},c}$	$\mathbb{R}_+$	NM	Similar description as $d_{\mathrm{V},w_{\mathrm{THR}}}^{a_i,c}$, but applied to $a_{1,i}$
$d_{\mathrm{V},w_{\mathrm{THR}}}^{a_{2,i},c}$	$\mathbb{R}_+$	NM	Similar description as $d_{\mathrm{V},w_{\mathrm{THR}}}^{a_i,c}$, but applied to $a_{2,i}$
$d_{\mathrm{V},w_{\mathrm{THR}}}^{a_{3,i},c}$	$\mathbb{R}_+$	NM	Similar description as $d_{\mathrm{V},w_{\mathrm{THR}}}^{a_i,c}$, but applied to $a_{3,i}$
$d_{\mathrm{V},w_{\mathrm{THR}}}^{a_{4,i},c}$	$\mathbb{R}_+$	NM	Similar description as $d_{\mathrm{V},w_{\mathrm{THR}}}^{a_i,c}$, but applied to $a_{4,i}$
$d_{\mathrm{V},w_{\mathrm{THR}}}^{a_{5,i},c}$	$\mathbb{R}_+$	NM	Similar description as $d_{\mathrm{V},w_{\mathrm{THR}}}^{a_i,c}$, but applied to $a_{5,i}$
$d_{w_{\mathrm{THR}},w_{S,k}}^{a_i,c}$	$\mathbb{R}_+$	NM	Distance between w_{THR} and $w_{S,k}$ along the legs of the STAR procedure
I^{c}	$\mathbb{I}^{c}$	-	Instruction that is ("*currently*") instructed by c to a_i
$I^{a_i,c}$	$\mathbb{I}^{c}$	-	Last instruction that has been instructed by c to a_i
$j_{\mathbb{A}_{\mathrm{FA}}}^{a_i,c}$	$\mathbb{N}_0$	-	Index of a_i in $\mathbb{A}_{\mathrm{FA}}$
$j_{\mathbb{A}_{\mathrm{IF}}}^{a_i,c}$	$\mathbb{N}_0$	-	Index of a_i in $\mathbb{A}_{\mathrm{IF}}$
$j_{\mathbb{A}_S}^{a_i,c}$	$\mathbb{N}_0$	-	Index of a_i in $\mathbb{A}_S$
$j_{\mathbb{A}_{S,H}}^{a_i,c}$	$\mathbb{N}_0$	-	Index of a_i in $\mathbb{A}_{S,H}$
$K^{a_i,c}$	$\mathbb{N}_0$	-	Number of waypoints in $S^{a_i,c}$
$k^{a_i,c}$	$[0, K^{a_i,c}]$	-	Current waypoint number of $S^{a_i,c}$

$k_{I_4,S}^{a_i,c}$	$[0, K^{a_i,c}]$	-	Waypoint (number) that corresponds to $w_{S,\text{MO}}$
$k_{I_5,S}^{a_i,c}$	$[0, K^{a_i,c}]$	-	Waypoint (number) that corresponds to $w_{S,\text{MI}}$
$k_{I_9,S}^{a_i,c}$	$[0, K^{a_i,c}]$	-	Waypoint (number) that corresponds to $w_{S,\text{R}}$
$k_{I_{10},S}^{a_i,c}$	$[0, K^{a_i,c}]$	-	Waypoint (number) that corresponds to $w_{S,\text{HO}}$
$k_{I_{15},S}^{a_i,c}$	$[0, K^{a_i,c}]$	-	Waypoint (number) that corresponds to $w_{S,H,\text{I}}$
$l_{t_{\text{IDE}}}^{c}$	$\mathbb{R}_+$	s	Minimum value of t_{IDE}^{c} as defined by M^c
$l_{t_{\text{INS}}}^{c}$	$\mathbb{R}_+$	s	Minimum value of t_{INS}^{c} as defined by M^c
$l_{t_{\text{SCN}}}^{c}$	$\mathbb{R}_+$	s	Minimum value of t_{SCN}^{c} as defined by M^c
$l_{t_{\text{SHD}}}^{c}$	$\mathbb{R}_+$	s	Minimum value of t_{SHD}^{c} as defined by M^c
M^c	$\mathbb{M}$	-	Current contextual control mode of c
$N^{a_i,c}$	$\mathbb{N}_0$	-	Index that defines to which $w_{I_6,S,N}$ a_i will be vectored
Q_{a,I_5}	$\mathbb{N}_0$	-	Number of aircraft that are currently flying *vector outbound merge*
$s^{a_{1,i},c}$	$\mathbb{R}_+$	NM	*Actual spacing* between a_i and $a_{1,i}$
$s_{\text{D}}^{a_{2,i},c}$	$\mathbb{R}_+$	NM	*Actual spacing* (*direct*) between a_i and $a_{2,i}$
$s_{\text{V}}^{a_{2,i},c}$	$\mathbb{R}_+$	NM	*Actual spacing* (*via*) between a_i and $a_{2,i}$
$s^{a_{3,i},c}$	$\mathbb{R}_+$	NM	*Actual spacing* between a_i and $a_{3,i}$
$s^{a_{4,i},c}$	$\mathbb{R}_+$	NM	*Actual spacing* between a_i and $a_{4,i}$
$s^{a_{5,i},c}$	$\mathbb{R}_+$	NM	*Actual spacing* between a_i and $a_{5,i}$
$s_{\text{S}}^{a_{1,i},c}$	$\mathbb{R}_+$	NM	*Separation minima* between a_i and $a_{1,i}$
$s_{\text{S}}^{a_{2,i},c}$	$\mathbb{R}_+$	NM	*Separation minima* between a_i and $a_{2,i}$
$s_{\text{S}}^{a_{3,i},c}$	$\mathbb{R}_+$	NM	*Separation minima* between a_i and $a_{3,i}$
$s_{\text{S}}^{a_{4,i},c}$	$\mathbb{R}_+$	NM	*Separation minima* between a_i and $a_{4,i}$
$s_{\text{S}}^{a_{5,i},c}$	$\mathbb{R}_+$	NM	*Separation minima* between a_i and $a_{5,i}$
$s_{\text{U}}^{a_{1,i},c}$	$\mathbb{R}_+$	NM	*Desired spacing* between a_i and $a_{1,i}$
$s_{\text{U}}^{a_{2,i},c}$	$\mathbb{R}_+$	NM	*Desired spacing* between a_i and $a_{2,i}$
$s_{\text{U}}^{a_{3,i},c}$	$\mathbb{R}_+$	NM	*Desired spacing* between a_i and $a_{3,i}$
$s_{\text{U}}^{a_{4,i},c}$	$\mathbb{R}_+$	NM	*Desired spacing* between a_i and $a_{4,i}$
$s_{\text{U}}^{a_{5,i},c}$	$\mathbb{R}_+$	NM	*Desired spacing* between a_i and $a_{5,i}$
$S^{a_i,c}$	$\mathbb{S}$	-	STAR procedure that is/has been operated by a_i
$t_{\text{B}}^{a_{1,i},c_1}$	$\mathbb{R}_+$	s	*Default time buffer* between a_i and $a_{1,i}$ as applied by c_1
$t_{\text{B}}^{a_{1,i},c_2}$	$\mathbb{R}_+$	s	*Default time buffer* between a_i and $a_{1,i}$ as applied by c_2
$t_{\text{B}}^{a_{1,i},c_3}$	$\mathbb{R}_+$	s	*Default time buffer* between a_i and $a_{1,i}$ as applied by c_3
$t_{\text{B}}^{a_{2,i},c_3}$	$\mathbb{R}_+$	s	*Default time buffer* between a_i and $a_{2,i}$ as applied by c_3
$t_{\text{B}}^{a_{3,i},c_4}$	$\mathbb{R}_+$	s	*Default time buffer* between a_i and $a_{3,i}$ as applied by c_4
t_{IDE}^{c}	$\mathbb{R}_+$	s	Time period required by c to identify tasks
t_{INS}^{c}	$\mathbb{R}_+$	s	Time period required by c to send instruction to a_i
t_{SCN}^{c}	$\mathbb{R}_+$	s	Recurrence time period of the scanning practice of c, i.e. event $\mathcal{E}_{\text{SA}}^{c}$
t_{SHD}^{c}	$\mathbb{R}_+$	s	Time period required by c to schedule the identified tasks
$v_{\text{GS}}^{a_i,c}$	$\mathbb{R}_+$	kt	Ground speed of a_i
$v_{\text{GS}}^{a_{5,i},c}$	$\mathbb{R}_+$	kt	Ground speed of $a_{5,i}$
$v_{\text{IAS}}^{a_i,c}$	$\mathbb{R}_+$	kt	Indicated (calibrated) airspeed of a_i
$v_{\text{IAS,I}}^{a_i}$	$\mathbb{R}_+$	kt	Instructed indicated airspeed
$v_{\text{IAS},w_{S,k}}^{a_i,c}$	$\mathbb{R}_+$	kt	Speed limit (IAS) at $w_{S,k}$
$v_{\text{IAS},w_{S,H}}^{a_i,c}$	$\mathbb{R}_+$	kt	Speed limit (IAS) in holding stack
$W^{a_i,c}$	$\mathbb{W}$	-	WTC of a_i
$W^{a_{1,i},c}$	$\mathbb{W}$	-	WTC of $a_{1,i}$
$W^{a_{2,i},c}$	$\mathbb{W}$	-	WTC of $a_{2,i}$
$W^{a_{3,i},c}$	$\mathbb{W}$	-	WTC of $a_{3,i}$
$W^{a_{4,i},c}$	$\mathbb{W}$	-	WTC of $a_{4,i}$
$W^{a_{5,i},c}$	$\mathbb{W}$	-	WTC of $a_{5,i}$

$x^{a_i,c}$	$\mathbb{R}$	NM	x-position of a_i
$x^{a_i,c}_{w_{S,k}}$	$\mathbb{R}$	NM	x-position of $w_{S,k}$
$x^{a_i,c}_{w_{S,H,\mathrm{I}}}$	$\mathbb{R}$	NM	x-position of $w_{S,H,\mathrm{I}}$
$x^{a_i,c}_{w_{I_2,S,k}}$	$\mathbb{R}$	NM	x-position of $w_{I_2,S,k}$
$x^{a_i,c}_{w_{I_4,S}}$	$\mathbb{R}$	NM	x-position of $w_{I_4,S}$
$x^{a_i,c}_{w_{I_6,S,N}}$	$\mathbb{R}$	NM	x-position of $w_{I_6,S,N}$
$x^{a_i,c}_{w_{I_8,S}}$	$\mathbb{R}$	NM	x-position of $w_{I_8,S}$
$x^{a_i,c}_{w_{S,\mathrm{MI}}}$	$\mathbb{R}$	NM	x-position of $w_{S,\mathrm{MI}}$
$y^{a_i,c}$	$\mathbb{R}$	NM	y-position of a_i
$y^{a_i,c}_{w_{S,k}}$	$\mathbb{R}$	NM	y-position of $w_{S,k}$
$y^{a_i,c}_{w_{S,H,\mathrm{I}}}$	$\mathbb{R}$	NM	y-position of $w_{S,H,\mathrm{I}}$
$y^{a_i,c}_{w_{I_2,S,k}}$	$\mathbb{R}$	NM	y-position of $w_{I_2,S,k}$
$y^{a_i,c}_{w_{I_4,S}}$	$\mathbb{R}$	NM	y-position of $w_{I_4,S}$
$y^{a_i,c}_{w_{I_6,S,N}}$	$\mathbb{R}$	NM	y-position of $w_{I_6,S,N}$
$y^{a_i,c}_{w_{I_8,S}}$	$\mathbb{R}$	NM	y-position of $w_{I_8,S}$
$y^{a_i,c}_{w_{S,\mathrm{MI}}}$	$\mathbb{R}$	NM	y-position of $w_{S,\mathrm{MI}}$
$z^{a_i,c}$	$\mathbb{R}_+$	ft	Altitude of a_i
$z^{a_i,c}_{\mathrm{I}}$	$\mathbb{R}_+$	ft	Instructed altitude
$z^{a_i,c}_{w_{S,k}}$	$\mathbb{R}_+$	ft	Altitude constraint at $w_{S,k}$
$z^{a_i,c}_{w_{S,H}}$	$\mathbb{R}_+$	ft	Altitude (minimum) in holding stack
$z^{a_i,c}_{\mathrm{R},w_{S,H}}$	$\mathbb{R}_+$	ft	Required altitude in holding stack
$\beta^{a_i,c}$	[-180, 180]	deg	Angle between $\psi^{a_i,c}_{\mathrm{A},w_{\mathrm{IF}}}$ and $\psi_{w_{\mathrm{THR}}}$, i.e. interception angle at w_{IF}
$\mu^c_{t_{\mathrm{IDE}}}$	$\mathbb{R}_{>0}$	s	Mean value of t^c_{IDE} as defined by M^c
$\mu^c_{t_{\mathrm{INS}}}$	$\mathbb{R}_{>0}$	s	Mean value of t^c_{INS} as defined by M^c
$\mu^c_{t_{\mathrm{SCN}}}$	$\mathbb{R}_{>0}$	s	Mean value of t^c_{SCN} as defined by M^c
$\mu^c_{t_{\mathrm{SHD}}}$	$\mathbb{R}_{>0}$	s	Mean value of t^c_{SHD} as defined by M^c
$\sigma^c_{t_{\mathrm{IDE}}}$	$\mathbb{R}_+$	s	Standard deviation of t^c_{IDE} as defined by M^c
$\sigma^c_{t_{\mathrm{INS}}}$	$\mathbb{R}_+$	s	Standard deviation of t^c_{INS} as defined by M^c
$\sigma^c_{t_{\mathrm{SCN}}}$	$\mathbb{R}_+$	s	Standard deviation of t^c_{SCN} as defined by M^c
$\sigma^c_{t_{\mathrm{SHD}}}$	$\mathbb{R}_+$	s	Standard deviation of t^c_{SHD} as defined by M^c
$\psi^{a_i,c}$	⟨0, 360]	deg	Heading direction of a_i with respect to magnetic north
$\psi^{a_i,c}_{w_{S,k}}$	⟨0, 360]	deg	Magnetic track of $w_{S,k}$, i.e. bearing of $w_{S,k}$ seen from $w_{S,k-1}$
$\psi^{a_i,c}_{w_{S,k+1}}$	⟨0, 360]	deg	Magnetic track of $w_{S,k+1}$, i.e. bearing of $w_{S,k+1}$ seen from $w_{S,k}$
$\psi^{a_i,c}_{\mathrm{A},w_{S,k}}$	⟨0, 360]	deg	Bearing (*tangent*) of $w_{S,k}$ seen from a_i
$\psi^{a_i,c}_{\mathrm{D},w_{S,k}}$	⟨0, 360]	deg	Bearing (*direct*) of $w_{S,k}$ seen from a_i
$\psi^{a_i,c}_{\mathrm{A},w_{I_2,S,k}}$	⟨0, 360]	deg	Bearing (*tangent*) of $w_{I_2,S,k}$ seen from a_i
$\psi^{a_i,c}_{\mathrm{D},w_{I_2,S,k}}$	⟨0, 360]	deg	Bearing (*direct*) of $w_{I_2,S,k}$ seen from a_i
$\psi^{a_i,c}_{\mathrm{A},w_{I_4,S}}$	⟨0, 360]	deg	Bearing (*tangent*) of $w_{I_4,S}$ seen from a_i
$\psi^{a_i,c}_{\mathrm{D},w_{I_4,S}}$	⟨0, 360]	deg	Bearing (*direct*) of $w_{I_4,S}$ seen from a_i
$\psi^{a_i,c}_{\mathrm{A},w_{I_6,S,N}}$	⟨0, 360]	deg	Bearing (*tangent*) of $w_{I_6,S,N}$ seen from a_i
$\psi^{a_i,c}_{\mathrm{D},w_{I_6,S,N}}$	⟨0, 360]	deg	Bearing (*direct*) of $w_{I_6,S,N}$ seen from a_i
$\psi^{a_i,c}_{\mathrm{A},w_{I_8,S}}$	⟨0, 360]	deg	Bearing (*tangent*) of $w_{I_8,S}$ seen from a_i
$\psi^{a_i,c}_{\mathrm{D},w_{I_8,S}}$	⟨0, 360]	deg	Bearing (*direct*) of $w_{I_8,S}$ seen from a_i
$\psi^{a_i,c}_{\mathrm{A},w_{\mathrm{IF}}}$	⟨0, 360]	deg	Bearing (*tangent*) of w_{IF} seen from a_i
$\psi^{a_i,c}_{\mathrm{D},w_{\mathrm{IF}}}$	⟨0, 360]	deg	Bearing (*direct*) of w_{IF} seen from a_i

Table 32 – List of variables that belong to the controller agent c

Parameter	Value	Unit	Description
$Q_{\mathrm{b},I_{15}}$	2	-	Maximum allowed number of aircraft that are flying the *vector outbound merge* operation at which holding operations become required
$Q_{\mathrm{b},I_{16}}$	0	-	Maximum allowed number of aircraft that are flying the *vector outbound merge* operation at which aircraft may be instructed to exit the holding
$t_{\mathrm{B},C_{\mathrm{D}}}^{a_{1,i},c_1}$	varying	s	*Default time buffer* between a_i and $a_{1,i}$ as applied by c_1 in *reduced capacity mode*
$t_{\mathrm{B},C_{\mathrm{I}}}^{a_{1,i},c_1}$	10	s	*Default time buffer* between a_i and $a_{1,i}$ as applied by c_1 in *normal capacity mode*
$t_{\mathrm{B},C_{\mathrm{D}}}^{a_{1,i},c_2}$	varying	s	*Default time buffer* between a_i and $a_{1,i}$ as applied by c_2 in *reduced capacity mode*
$t_{\mathrm{B},C_{\mathrm{I}}}^{a_{1,i},c_2}$	10	s	*Default time buffer* between a_i and $a_{1,i}$ as applied by c_2 in *normal capacity mode*
$t_{\mathrm{B},C_{\mathrm{D}}}^{a_{1,i},c_3}$	10	s	*Default time buffer* between a_i and $a_{1,i}$ as applied by c_3 in *reduced capacity mode*
$t_{\mathrm{B},C_{\mathrm{I}}}^{a_{1,i},c_3}$	10	s	*Default time buffer* between a_i and $a_{1,i}$ as applied by c_3 in *normal capacity mode*
$t_{\mathrm{B},C_{\mathrm{D}}}^{a_{2,i},c_3}$	70	s	*Default time buffer* between a_i and $a_{2,i}$ as applied by c_3 in *reduced capacity mode*
$t_{\mathrm{B},C_{\mathrm{I}}}^{a_{2,i},c_3}$	15	s	*Default time buffer* between a_i and $a_{2,i}$ as applied by c_3 in *normal capacity mode*
$t_{\mathrm{B},C_{\mathrm{D}}}^{a_{3,i},c_4}$	20	s	*Default time buffer* between a_i and $a_{3,i}$ as applied by c_4 in *reduced capacity mode*
$t_{\mathrm{B},C_{\mathrm{I}}}^{a_{3,i},c_4}$	-10	s	*Default time buffer* between a_i and $a_{3,i}$ as applied by c_4 in *normal capacity mode*
$t_{\mathrm{B,HM}}^{a_{1,i},c_1}$	0	s	*Additional time buffer* between a_i and $a_{1,i}$ as applied by c_1
$t_{\mathrm{B,MH}}^{a_{1,i},c_1}$	0	s	*Additional time buffer* between a_i and $a_{1,i}$ as applied by c_1
$t_{\mathrm{B,HM}}^{a_{1,i},c_2}$	0	s	*Additional time buffer* between a_i and $a_{1,i}$ as applied by c_2
$t_{\mathrm{B,MH}}^{a_{1,i},c_2}$	0	s	*Additional time buffer* between a_i and $a_{1,i}$ as applied by c_2
$t_{\mathrm{B,HM}}^{a_{1,i},c_3}$	0	s	*Additional time buffer* between a_i and $a_{1,i}$ as applied by c_3
$t_{\mathrm{B,MH}}^{a_{1,i},c_3}$	0	s	*Additional time buffer* between a_i and $a_{1,i}$ as applied by c_3
$t_{\mathrm{B,HM}}^{a_{2,i},c_3}$	10	s	*Additional time buffer* between a_i and $a_{2,i}$ as applied by c_3
$t_{\mathrm{B,MH}}^{a_{2,i},c_3}$	15	s	*Additional time buffer* between a_i and $a_{2,i}$ as applied by c_3
$t_{\mathrm{B,HM}}^{a_{3,i},c_4}$	0	s	*Additional time buffer* between a_i and $a_{3,i}$ as applied by c_4
$t_{\mathrm{B,MH}}^{a_{3,i},c_4}$	0	s	*Additional time buffer* between a_i and $a_{3,i}$ as applied by c_4
u_β	45	deg	Maximum allowed interception angle at w_{IF}
β_{b}	18	deg	Bound that determines whether $w_{I_6,S,1}$ or $w_{I_6,S,2}$ should be used

Table 33 – List of parameters that belong to the controller agent c

Set	Contents	Description
$\mathbb{A}^c$	$\subseteq \mathbb{A}$	Sorted list of aircraft that controller c is responsible for. $\mathbb{A}^c$ is sorted by $d_{\mathrm{V},w_{\mathrm{THR}}}^{a_i,c}$
$\mathbb{A}_{\mathrm{FA}}$	$\subseteq \mathbb{A}$	Sequence of aircraft that are flying the intermediate- or final approach segment
$\mathbb{A}_{\mathrm{IF}}$	$\subseteq \mathbb{A}$	Sequence of aircraft that will be or have been instructed the *vector inbound IF* instruction, but which have not yet intercepted w_{IF}
$\mathbb{A}_{I_8}$	$\subseteq \mathbb{A}$	Sequence of aircraft that are operating the *vector outbound trombone* instruction in general
$\mathbb{A}_{I_8,S}$	$\subseteq \mathbb{A}_{I_8}$	Sequence of aircraft that are operating the *vector outbound trombone* instruction towards $w_{I_8,S}$
$\mathbb{A}_{I_9,S}$	$\subseteq \mathbb{A}$	Sequence of aircraft that are operating the *vector inbound trombone* towards $w_{S,\mathrm{R}}$
$\mathbb{A}_S$	$\subseteq \mathbb{A}$	Sequence of aircraft that are or have been flying the STAR procedure S
$\mathbb{A}_{S,H}$	$\subseteq \mathbb{A}_S$	Sequence of aircraft that are flying in holding pattern H
$\mathbb{I}^c$	$\subseteq \mathbb{I}$	Collection of identified tasks/instructions by controller c
$\mathbb{M}$	$\{M_{\mathrm{T}}, M_{\mathrm{O}}\}$	Collection of contextual control modes in which c may operate, where M_{T} represents the tactical control mode and where M_{O} represents the opportunistic control mode
$\mathbb{T}_{\mathrm{I}}^c$		Collection of recent time points at which instructions have been provided by c

Table 34 – List of sets that belong to the controller agent c

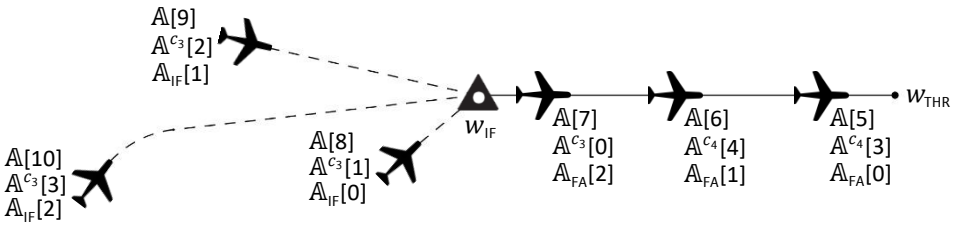


Figure 63 – Example scenario that visualizes the use and purpose of some aircraft collections that are used in the model

C.4.2 Initial actions and values

Each controller will initially operate in the ⟨*tactical*⟩⌊*Control mode*⌋[c] state. See the entry actions of the ⟨*tactical*⟩ state in appendix C.4.5.4 for the initial values of each of the variables that define the performance of the controller in this specific state.

Each controller will initially maintain a throughput capacity that is in accordance with the nominal runway capacity. See the entry actions of the ⟨*recovered capacity update received*⟩⌊*Contact with supervisor about recovered capacity*⌋[c] state in section C.4.5.6 for the initial values of the *time buffer* variables that are used to maintain the initial throughput capacity.

C.4.3 Functions

C.4.3.1 Define vectoring accuracy

Function 20 is used to model the inaccuracies and uncertainties of the controller in providing proper (*inbound*) vector instructions. The function returns a *separation buffer (type II)* that eventually will be added to the 'estimated' distance $\check{d}$.

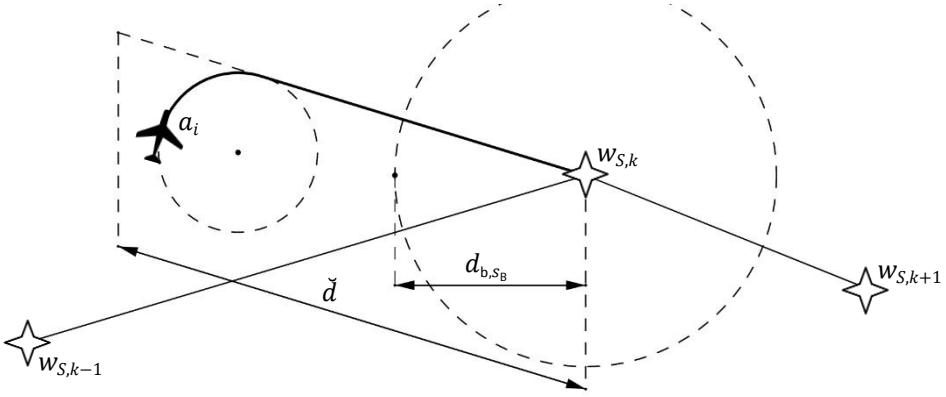


Figure 64 – Two regions with different vector accuracies as defined by the 'threshold' distance

Function $\mathcal{F}_{s_{\text{B}}}(\check{d}, \check{M})$

Argument	State space	Unit	Description
$\check{d}$	$\mathbb{R}_+$	NM	Distance between a_i and the significant point where a_i will be vectored to
$\check{M}$	$\mathbb{M}$	-	Contextual control mode of the controller
Variable	**State space**	**Unit**	**Description**
$\check{u}_{s_{\text{B}}}$	$\mathbb{R}$	NM	Mean value of $\check{s}_{\text{B}}$
$\check{\sigma}_{s_{\text{B}}}$	$\mathbb{R}_+$	NM	Standard deviation of $\check{s}_{\text{B}}$
$\check{s}_{\text{B}}$	$\mathbb{R}$	NM	*Separation buffer (type II)*

Parameter	Value	Unit	Description
d_{b,s_B}	15	NM	Distance (radius) from the significant point where a_i will be vectored to, which marks the boundary between the two regions where different vectoring accuracies apply
$\mu_{s_B,L,O}$	-0.5	NM	Mean value of $\breve{s}_B$ when $\breve{d} > d_{b,s_B}$ and $\breve{M} == M_O$
$\mu_{s_B,L,T}$	-0.4	NM	Mean value of $\breve{s}_B$ when $\breve{d} > d_{b,s_B}$ and $\breve{M} == M_T$
$\mu_{s_B,S,O}$	-0.3	NM	Mean value of $\breve{s}_B$ when $\breve{d} \leq d_{b,s_B}$ and $\breve{M} == M_O$
$\mu_{s_B,S,T}$	-0.2	NM	Mean value of $\breve{s}_B$ when $\breve{d} \leq d_{b,s_B}$ and $\breve{M} == M_T$
$\sigma_{s_B,L,O}$	varying	NM	Standard deviation of $\breve{s}_B$ when $\breve{d} > d_{b,s_B}$ and $\breve{M} == M_O$
$\sigma_{s_B,L,T}$	varying	NM	Standard deviation of $\breve{s}_B$ when $\breve{d} > d_{b,s_B}$ and $\breve{M} == M_T$
$\sigma_{s_B,S,O}$	varying	NM	Standard deviation of $\breve{s}_B$ when $\breve{d} \leq d_{b,s_B}$ and $\breve{M} == M_O$
$\sigma_{s_B,S,T}$	varying	NM	Standard deviation of $\breve{s}_B$ when $\breve{d} \leq d_{b,s_B}$ and $\breve{M} == M_T$

if ($\breve{d} > d_{b,s_B}$) **then**
  if ($\breve{M} == M_T$) **then**
    $\breve{u}_{s_B} = \mu_{s_B,L,T}$
    $\breve{\sigma}_{s_B} = \sigma_{s_B,L,T}$
  else if ($\breve{M} == M_O$) **then**
    $\breve{u}_{s_B} = \mu_{s_B,L,O}$
    $\breve{\sigma}_{s_B} = \sigma_{s_B,L,O}$
  end if
else if ($\breve{d} \leq d_{b,s_B}$) **then**
  if ($\breve{M} == M_T$) **then**
    $\breve{u}_{s_B} = \mu_{s_B,S,T}$
    $\breve{\sigma}_{s_B} = \sigma_{s_B,S,T}$
  else if ($\breve{M} == M_O$) **then**
    $\breve{u}_{s_B} = \mu_{s_B,S,O}$
    $\breve{\sigma}_{s_B} = \sigma_{s_B,S,O}$
  end if
end if
$\breve{s}_B = \mathcal{N}(\breve{u}_{s_B}, \breve{\sigma}_{s_B})$
return: $\breve{s}_B$

Function 20 – Calculation of the separation buffer that is used to model the vectoring (in)accuracy

C.4.3.2 Update the assigned positions of vector points

Function 25 is used to model the so-called "*opening*" vectors. The principle of these modelled "*opening*" vectors is visualized in figure 65. Table 35 contains the parameter values that describe how the position of each vector point will change after a position update.

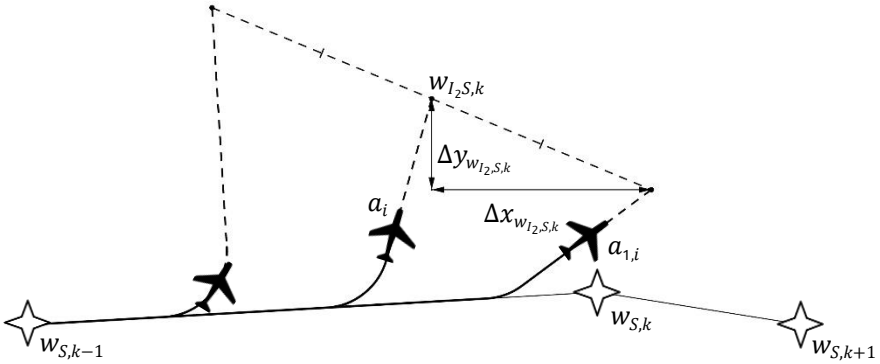


Figure 65 – Visualization of the "opening" divergent vectors due to updated positions of the respective vector waypoints

		$\Delta x_{w_{I_2,S,k}}$		$\Delta y_{w_{I_2,S,k}}$	
		S_1	S_2	S_1	S_2
k	0	N/A	N/A	N/A	N/A
	1	-5	0	0	-5
	2	0	0	-5	-5
	3	0	0	-5	-5
	4	0	5	-5	0
	5	-5	0	0	0
	6	0	0	0	0
	7	0	0	0	0
	8	0	5	0	0
	9	-5	0	0	0
	10	0	0	0	0
	11	0	0	0	0
	12	0	N/A	0	N/A

Table 35 – Parameter values that describe how the positions of each vector point will change after a position update

Function $\mathcal{F}_{w_{I_2}}(S,k)$

Argument	State space	Unit	Description
S	$\mathbb{S}$	-	STAR procedure where the to be updated $w_{I_2,S,k}$ belongs to
k	$[0,..,K_S]$	-	Waypoint number that corresponds to the to be updated $w_{I_2,S,k}$
Parameter	**Value**	**Unit**	**Description**
$\Delta x_{w_{I_2,S,k}}$	Table 35	NM	Degree by which the x-position of $w_{I_2,S,k}$ will change after position update
$\Delta y_{w_{I_2,S,k}}$	Table 35	NM	Degree by which the y-position of $w_{I_2,S,k}$ will change after position update

1: $x_{w_{I_2,S,k}}$ += $\Delta x_{w_{I_2,S,k}}$

2: $y_{w_{I_2,S,k}}$ += $\Delta y_{w_{I_2,S,k}}$

Function 21 – Position update of the vector point $w_{I_2,S,k}$

C.4.4 Events

C.4.4.1 Update situation awareness of controller about a_i

Event $\mathcal{E}^c_{\text{SA}}$ models the updating process of the situation awareness of controller c about a_i ($\forall c \in \mathbb{C} \wedge \forall a_i \in \mathbb{A}^c$). This event collects all essential and required data to identify and instruct the set of defined tasks. The contents of action 2 below provide a structured overview of the multiple subprocesses that are considered in this event. The majority of these subprocesses are meant to define the feasibility and necessity of the set of identified instructions. The memory of the controller agent about the feasibility and/or necessity of each instruction is modelled using a number of Boolean variables. The Boolean variables that are assigned in event $\mathcal{E}^c_{\text{SA}}$ will eventually be evaluated in the condition triggered transitions of the so-called SA-statecharts (appendix C.4.5.1).

- First occurrence time: starts immediately
- Recurrence time period: t^c_{SCN}

Event $\mathcal{E}^c_{\text{SA}}$

1: $t^c_{\text{SCN}} = \mathcal{L}(\mu^c_{t_{\text{SCN}}}, \sigma^c_{t_{\text{SCN}}}, l^c_{t_{\text{SCN}}})$

2: Update SA of controller about the position and orientation of a_i
- Observe the flight data of a_i from the radar screen
- Obtain the required data of $w_{S,k}$ that corresponds to a_i
- Observe the position and orientation of a_i relative to $w_{S,k}$
- Observe the position and orientation of a_i relative to w_{IF}

- Observe the position of a_i relative to $w_{S,H,\text{I}}$
- Observe the position and orientation of a_i relative to the multiple vector points
- Observe the distance between a_i and w_{THR}
- Determine the waypoint number $k^{a_i,c}$ during *vector outbound STAR* operations

3: Update SA of controller about the situation between a_i and $a_{1,i}$
- Find available $a_{1,i}$
- Observe *actual spacing* between a_i and $a_{1,i}$
- Define *separation minima* and *desired spacing* between a_i and $a_{1,i}$
- Check *vector outbound STAR* requirement for a_i
- Decide if a_i is available for a *vector inbound STAR* instruction
- Check *holding entry* requirement for a_i
- Decide if a_i is available for *holding entry*
- Decide if a_i is available for *holding exit*
- Check *holding altitude* requirement for a_i

4: Update SA of controller about the situation between a_i and $a_{2,i}$
- Find available $a_{2,i}$
- Observe *actual spacing* between a_i and $a_{2,i}$
- Define *separation minima* and *desired spacing* between a_i and $a_{2,i}$
- Check *vector outbound IF* requirement for a_i
- Decide if a_i is available for a *vector inbound IF* instruction
- Check *vector outbound merge* requirement for a_i
- Decide if a_i is available for a *vector inbound merge* instruction
- Check *vector outbound trombone* requirement for a_i

5: Update SA of controller about the situation between a_i and $a_{3,i}$
- Find available $a_{3,i}$
- Observe *actual spacing* between a_i and $a_{3,i}$
- Define *separation minima* and *desired spacing* between a_i and $a_{3,i}$
- Decide if a_i is available for *landing clearance*

6: Update SA of controller about the situation between a_i, $a_{4,i}$ and $a_{5,i}$
- Find available $a_{4,i}$ and $a_{5,i}$
- Observe *actual spacing* between a_i and $a_{4,i}$, and between a_i and $a_{5,i}$
- Define *separation minima* and *desired spacing* between a_i and $a_{4,i}$, and between a_i and $a_{5,i}$
- Decide if a_i is available for a *vector inbound trombone* instruction

7: Update SA of controller about handover of a_i
- Decide if a_i is available for *handover to ARR*
- Decide if a_i is available for *handover to TWR*
- Decide if a_i is available for *handover to GND*

Action 2 – Updating process of the situation awareness of controller c about a_i, where $a_i \in \mathbb{A}^c$ ($\forall c \in \mathbb{C}$)

Update SA of controller about the position and orientation of a_i

The first set of actions are (for the most part) meant to identify and specify the position and orientation of a_i relative to the multiple significant points.

Observe the flight data of a_i from the radar screen

1: $x^{a_i,c} = x^{a_i}$
2: $y^{a_i,c} = y^{a_i}$
3: $z^{a_i,c} = z^{a_i}$
4: $\psi^{a_i,c} = \psi^{a_i}$
5: $v_{\text{IAS}}^{a_i,c} = v_{\text{IAS}}^{a_i}$
6: $v_{\text{GS}}^{a_i,c} = \mathcal{F}_{v_{\text{GS}}}(z^{a_i,c}, v_{\text{IAS}}^{a_i,c}, \psi^{a_i,c})$
7: $W^{a_i,c} = W^{a_i}$

8: $S^{a_i,c} = S^{a_i}$

9: $k^{a_i,c} = k^{a_i}$

Obtain the required data of $w_{S,k}$ that corresponds to a_i

10: $x_{w_{S,k}}^{a_i,c}$ = *

11: $y_{w_{S,k}}^{a_i,c}$ = *

12: $z_{w_{S,k}}^{a_i,c}$ = *

13: $v_{\text{IAS},w_{S,k}}^{a_i,c}$ = *

14: $\psi_{w_{S,k}}^{a_i,c}$ = *

15: $\psi_{w_{S,k+1}}^{a_i,c}$ = *

16: $d_{w_{\text{THR}},w_{S,k}}^{a_i,c}$ = *

*obtain data in appendix C.3.2

Observe the position and orientation of a_i relative to $w_{S,k}$

17: $\psi_{\text{A},w_{S,k}}^{a_i,c} = \mathcal{F}_{\psi_\text{A}}(\psi^{a_i,c}, x^{a_i,c}, y^{a_i,c}, x_{w_{S,k}}^{a_i,c}, y_{w_{S,k}}^{a_i,c}, r_\text{T})$

18: $\psi_{\text{D},w_{S,k}}^{a_i,c} = \mathcal{C}_{\text{DEG}}^{\text{RAD}}(\text{atan2}(y_{w_{S,k}}^{a_i,c} - y^{a_i,c}, x_{w_{S,k}}^{a_i,c} - x^{a_i,c}))$

19: $d_{\text{A},w_{S,k}}^{a_i,c} = \mathcal{F}_d(\psi^{a_i,c}, x^{a_i,c}, y^{a_i,c}, x_{w_{S,k}}^{a_i,c}, y_{w_{S,k}}^{a_i,c}, r_\text{T})$

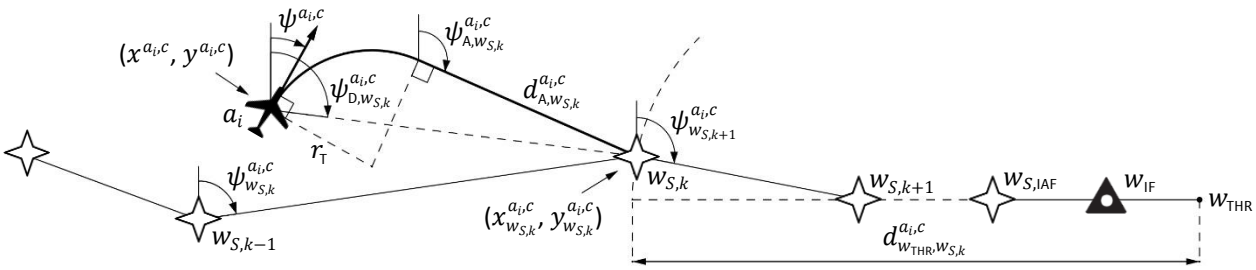


Figure 66 – Visualization of the variables that are used to define the position and orientation of a_i relative to $w_{S,k}$

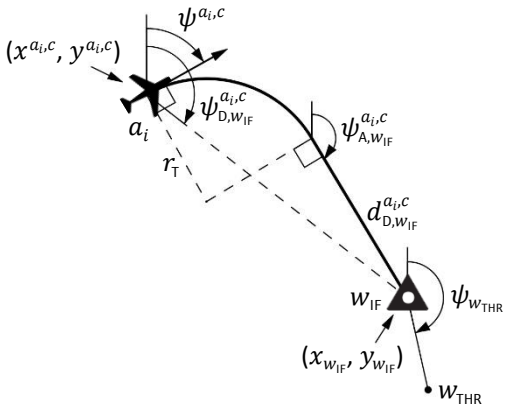
Observe the position and orientation of a_i relative to w_{IF}

20: $\psi_{\text{A},w_{\text{IF}}}^{a_i,c} = \mathcal{F}_{\psi_\text{A}}(\psi^{a_i,c}, x^{a_i,c}, y^{a_i,c}, x_{w_{\text{IF}}}, y_{w_{\text{IF}}}, r_\text{T})$

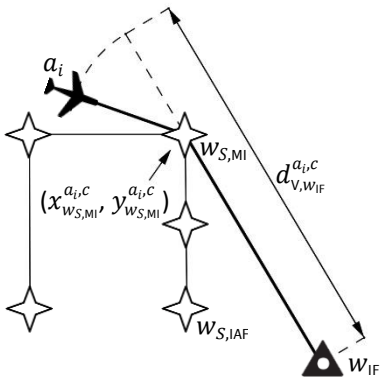
21: $\psi_{\text{D},w_{\text{IF}}}^{a_i,c} = \mathcal{C}_{\text{DEG}}^{\text{RAD}}(\text{atan2}(y_{w_{\text{IF}}} - y^{a_i,c}, x_{w_{\text{IF}}} - x^{a_i,c}))$

22: $d_{\text{D},w_{\text{IF}}}^{a_i,c} = \mathcal{F}_d(\psi^{a_i,c}, x^{a_i,c}, y^{a_i,c}, x_{w_{\text{IF}}}, y_{w_{\text{IF}}}, r_\text{T})$

23: $d_{\text{V},w_{\text{IF}}}^{a_i,c} = \mathcal{F}_d(\psi^{a_i,c}, x^{a_i,c}, y^{a_i,c}, x_{w_{S,\text{MI}}}^{a_i,c}, y_{w_{S,\text{MI}}}^{a_i,c}, r_\text{T}) + \sqrt{((x_{w_{S,\text{MI}}}^{a_i,c} - x_{w_{\text{IF}}})^2 + (y_{w_{S,\text{MI}}}^{a_i,c} - y_{w_{\text{IF}}})^2)}$



(a) a_i directly referenced to w_{IF}



(b) a_i referenced to w_{IF} via $w_{S,MI}$

Figure 67 – Visualization of the variables that are used to define the position and orientation of a_i relative to w_{IF}

Observe the position of a_i relative to $w_{S,H,\text{I}}$

24: $d^{a_i,c}_{\text{D},w_{S,H,\text{I}}} = \sqrt{((x^{a_i,c}_{w_{S,H,\text{I}}} - x^{a_i,c})^2 + (y^{a_i,c}_{w_{S,H,\text{I}}} - y^{a_i,c})^2)}$

Observe the position and orientation of a_i relative to the multiple vector points

25: $x^{a_i,c}_{w_{I_2,S,k}} = *$

26: $y^{a_i,c}_{w_{I_2,S,k}} = *$

27: $\psi^{a_i,c}_{\text{A},w_{I_2,S,k}} = \mathcal{F}_{\psi_\text{A}}(\psi^{a_i,c}, x^{a_i,c}, y^{a_i,c}, x^{a_i,c}_{w_{I_2,S,k}}, y^{a_i,c}_{w_{I_2,S,k}}, r_\text{T})$

28: $\psi^{a_i,c}_{\text{D},w_{I_2,S,k}} = \mathcal{C}^{\text{RAD}}_{\text{DEG}}(\text{atan2}(y^{a_i,c}_{w_{I_2,S,k}} - y^{a_i,c}, x^{a_i,c}_{w_{I_2,S,k}} - x^{a_i,c}))$

29: $\beta^{a_i,c} = \mathcal{F}_\theta(\psi^{a_i,c}_{\text{A},w_{\text{IF}}}, \psi_{w_{\text{THR}}})$

30: **if** $(|\beta^{a_i,c}| \geq \beta_\text{b})$ **then**

31: $\quad N^{a_i,c} = 1$

32: **else if** $(|\beta^{a_i,c}| < \beta_\text{b})$ **then**

33: $\quad N^{a_i,c} = 2$

34: **end if**

35: $x^{a_i,c}_{w_{I_6,S,N}} = *$

36: $y^{a_i,c}_{w_{I_6,S,N}} = *$

37: $\psi^{a_i,c}_{\text{A},w_{I_6,S,N}} = \mathcal{F}_{\psi_\text{A}}(\psi^{a_i,c}, x^{a_i,c}, y^{a_i,c}, x^{a_i,c}_{w_{I_6,S,N}}, y^{a_i,c}_{w_{I_6,S,N}}, r_\text{T})$

38: $\psi^{a_i,c}_{\text{D},w_{I_6,S,N}} = \mathcal{C}^{\text{RAD}}_{\text{DEG}}(\text{atan2}(y^{a_i,c}_{w_{I_6,S,N}} - y^{a_i,c}, x^{a_i,c}_{w_{I_6,S,N}} - x^{a_i,c}))$

39: $\psi^{a_i,c}_{\text{A},w_{I_4,S}} = \mathcal{F}_{\psi_\text{A}}(\psi^{a_i,c}, x^{a_i,c}, y^{a_i,c}, x^{a_i,c}_{w_{I_4,S}}, y^{a_i,c}_{w_{I_4,S}}, r_\text{T})$

40: $\psi^{a_i,c}_{\text{D},w_{I_4,S}} = \mathcal{C}^{\text{RAD}}_{\text{DEG}}(\text{atan2}(y^{a_i,c}_{w_{I_4,S}} - y^{a_i,c}, x^{a_i,c}_{w_{I_4,S}} - x^{a_i,c}))$

41: $\psi^{a_i,c}_{\text{A},w_{I_8,S}} = \mathcal{F}_{\psi_\text{A}}(\psi^{a_i,c}, x^{a_i,c}, y^{a_i,c}, x^{a_i,c}_{w_{I_8,S}}, y^{a_i,c}_{w_{I_8,S}}, r_\text{T})$

42: $\psi^{a_i,c}_{\text{D},w_{I_8,S}} = \mathcal{C}^{\text{RAD}}_{\text{DEG}}(\text{atan2}(y^{a_i,c}_{w_{I_8,S}} - y^{a_i,c}, x^{a_i,c}_{w_{I_8,S}} - x^{a_i,c}))$

*obtain data appendix C.3.2

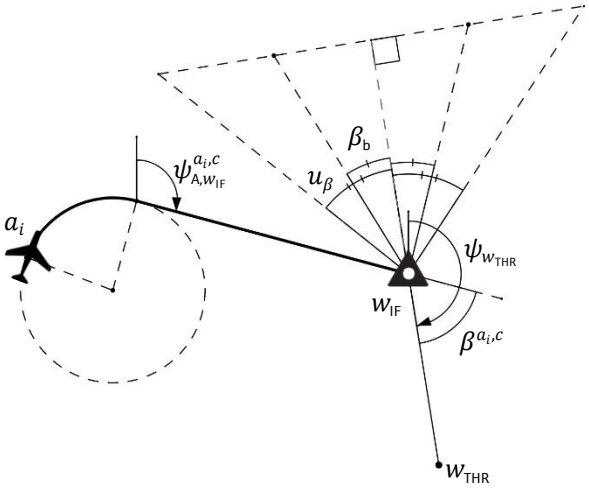


Figure 68 – The angles that are used by the controller agent to define the specific vector operation and vector point

Observe the distance between a_i and w_{THR}

43: **if** $(\langle STAR \rangle \lfloor SA\ operation \rfloor [a_i, c] \ \wedge \neg \langle passed\ final\ waypoint \rangle \lfloor SA\ STAR\ progress \rfloor [a_i, c]) \vee \langle vector\ merge \rangle \lfloor SA\ operation \rfloor [a_i, c] \vee \langle vector\ STAR \rangle \lfloor SA\ operation \rfloor [a_i, c] \vee \langle vector\ outbound\ trombone\ phase\ 2 \rangle \lfloor SA\ operation \rfloor [a_i, c] \vee \langle vector\ inbound\ trombone \rangle \lfloor SA\ operation \rfloor [a_i, c] \vee \langle holding \rangle \lfloor SA\ operation \rfloor [a_i, c])$ **then**

44: $\quad d^{a_i,c}_{\text{V},w_{\text{THR}}} = d^{a_i,c}_{w_{\text{THR}},w_{S,k}} + d^{a_i,c}_{\text{A},w_{S,k}}$

45: **else if** $(\langle vector\ IF \rangle \lfloor SA\ operation \rfloor [a_i, c])$ **then**

46: $d^{a_i,c}_{\mathrm{V},w_{\mathrm{THR}}} = d^{a_i,c}_{\mathrm{D},w_{\mathrm{IF}}} + d_{w_{\mathrm{THR}},w_{\mathrm{IF}}}$
47: **else if** ⟨*intermediate approach*⟩⌊*SA operation*⌋$[a_i, c]$ ∨ ⟨*final approach*⟩⌊*SA operation*⌋$[a_i, c]$) **then**
48: $d^{a_i,c}_{\mathrm{V},w_{\mathrm{THR}}} = \sqrt{((x_{w_{\mathrm{THR}}} - x^{a_i,c})^2 + (y_{w_{\mathrm{THR}}} - y^{a_i,c})^2)}$
49: **else**
50: $d^{a_i,c}_{\mathrm{V},w_{\mathrm{THR}}} = 0$
51: **end if**

Determine the waypoint number $k^{a_i,c}$ during *vector outbound STAR* operations

Models the reasoning of the controller agent in defining the most logical and suitable waypoint ($w_{S,k}$) to vector a_i to. This reasoning is based on the classification of a_i in one of the two possible zones that are bounded by the so-called *update radials* $\psi^{a_i,c}_{\mathrm{U,L}}$ and $\psi^{a_i,c}_{\mathrm{U,R}}$. Both *update radials* are directed towards $w_{S,k}$ of a_i and enclose the radial $\psi^{a_i,c}_{w_{S,k}}$ (circle sector ① in figure 69). The update radials are defined by the bisector(s) of the radials $\psi^{a_i,c}_{w_{S,k}}$ and $\psi^{a_i,c}_{w_{S,k+1}}$. The current waypoint number of a_i ($k^{a_i,c}$) is updated (i.e. incremented) once a_i switches zone, i.e. once a_i is positioned outside the region that is enclosed by the two update radials (circle sectors ② and ③ in figure 69). Note that the directions of the *update radials* approach the logic and reasoning that a controller applies when instructing a *vector inbound STAR*. The reasoning that has been modelled is applicable at all time and to all kind of trajectories and route structures.

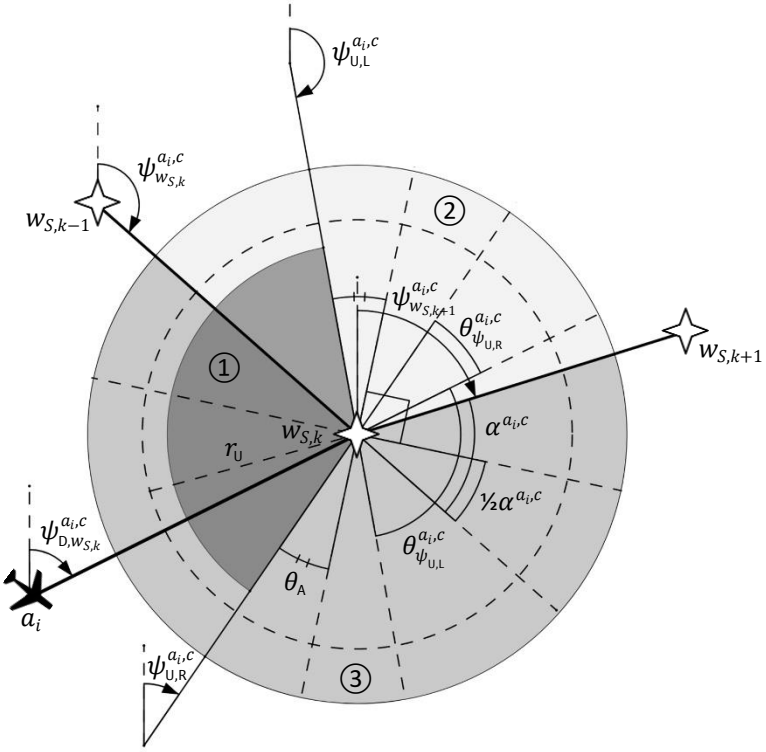


Figure 69 – Visualization of the update radials, i.e. the angles and heading directions that define when the controller will update $k^{a_i,c}$

Variable	State space	Unit	Description
$\alpha^{a_i,c}$	[-180, 180]	deg	Angle between $\psi^{a_i,c}_{w_{S,k}}$ and $\psi^{a_i,c}_{w_{S,k+1}}$
$\psi^{a_i,c}_{\mathrm{U,L}}$	⟨0, 360]	deg	Update radial oriented left of the leg between $w_{S,k-1}$ and $w_{S,k}$
$\psi^{a_i,c}_{\mathrm{U,R}}$	⟨0, 360]	deg	Update radial oriented right of the leg between $w_{S,k-1}$ and $w_{S,k}$
Parameter	**Value**	**Unit**	**Description**
r_{U}	4	NM	Radius that guarantees viable turning movements, > 2 · r_{T}
θ_{A}	20	deg	Angle that defines the shape of the update radials $\psi^{a_i,c}_{\mathrm{U,L}}$ and $\psi^{a_i,c}_{\mathrm{U,R}}$, and used to 'tune' the size of the area that is enclosed by the two update radials (①)

52: $\alpha^{a_i,c} = \mathcal{F}_\theta(\psi^{a_i,c}_{w_{S,k}}, \psi^{a_i,c}_{w_{S,k+1}})$
53: $\psi^{a_i,c}_{\mathrm{U,L}} = \psi^{a_i,c}_{w_{S,k}} + ½\alpha^{a_i,c} + 90 - \theta_{\mathrm{A}}$

54: $\psi_{\text{U,R}}^{a_i,c} = \psi_{w_{S,k}}^{a_i,c} + ½\alpha^{a_i,c} - 90 + \theta_{\text{A}}$
55: **if** ($\psi_{\text{U,L}}^{a_i,c} \leq 0$) **then**
56: $\psi_{\text{U,L}}^{a_i,c}$ += 360
57: **else if** ($\psi_{\text{U,L}}^{a_i,c} > 360$) **then**
58: $\psi_{\text{U,L}}^{a_i,c}$ -= 360
59: **end if**
60: **if** ($\psi_{\text{U,R}}^{a_i,c} \leq 0$) **then**
61: $\psi_{\text{U,R}}^{a_i,c}$ += 360
62: **else if** ($\psi_{\text{U,R}}^{a_i,c} > 360$) **then**
63: $\psi_{\text{U,R}}^{a_i,c}$ -= 360
64: **end if**
65: $\theta_{\psi_{\text{U,L}}}^{a_i,c} = \mathcal{F}_\theta(\psi_{\text{D},w_{S,k}}^{a_i,c}, \psi_{\text{U,L}}^{a_i,c})$
66: $\theta_{\psi_{\text{U,R}}}^{a_i,c} = \mathcal{F}_\theta(\psi_{\text{D},w_{S,k}}^{a_i,c}, \psi_{\text{U,R}}^{a_i,c})$
67: **if** (⟨*vector outbound STAR*⟩⌊*SA operation*⌋$[a_i, c]$ ∧ ⟨*available for waypoint update*⟩⌊*SA waypoint update history*⌋ $[a_i, c]$ ∧ (($\theta_{\psi_{\text{U,L}}}^{a_i,c} < 0 \vee \theta_{\psi_{\text{U,R}}}^{a_i,c} > 0$) ∧ $d_{\text{A},w_{S,k}}^{a_i,c} \geq r_{\text{U}}$) ∨ $d_{\text{A},w_{S,k}}^{a_i,c} < r_{\text{U}}$)) **then**
68: Trigger t_1 of ⌊*SA waypoint update history*⌋$[a_i, c]$
69: **if** ($k^{a_i,c} \neq K^{a_i,c} - 1$) **then**
70: $k^{a_i,c}$++
71: k^{a_i}++
72: **end if**
73: **if** (⟨*nearing final waypoint*⟩⌊*SA STAR progress*⌋$[a_i, c]$) **then**
74: Trigger t_5 of ⌊*SA STAR progress*⌋$[a_i, c]$
75: **end if**
76: **end if**

Update SA of controller about the situation between a_i and $a_{1,i}$

The second category of clustered actions describe the controller's situation awareness of aircraft a_i and $a_{1,i}$. This obtained information is used to decide upon the necessity and feasibility of the *vector outbound STAR*, *vector inbound STAR*, *holding entry*, *holding exit* and *holding altitude* instructions.

Find available $a_{1,i}$

77: $j_{\mathbb{A}_S}^{a_i,c} = (i\{a_i\} \in \mathbb{A}_S)$
78: $j_{\mathbb{A}_{S,H}}^{a_i,c} = (i\{a_i\} \in \mathbb{A}_{S,H})$
79: **if** ($j_{\mathbb{A}_S}^{a_i,c} > 0$) **then**
80: $a_{1,i} = \mathbb{A}_S[j_{\mathbb{A}_S}^{a_i,c} - 1]$
81: obtain $W^{a_{1,i},c}$ and $d_{\text{V},w_{\text{THR}}}^{a_{1,i},c}$
82: **end if**

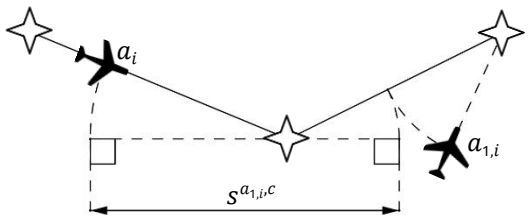


Figure 70 – Visualization of the observed actual spacing between a_i and $a_{1,i}$

Observe *actual spacing* between a_i and $a_{1,i}$

if ($j_{\mathbb{A}_S}^{a_i,c} > 0$) **then**
    $s^{a_{1,i},c} = d_{\mathrm{V},w_{\mathrm{THR}}}^{a_i,c} - d_{\mathrm{V},w_{\mathrm{THR}}}^{a_{1,i},c}$
end if

Define *separation minima* and *desired spacing* between a_i and $a_{1,i}$

Variable	State space	Unit	Description
$\check{s}_{\mathrm{B}}^{a_{1,i},c}$	$\mathbb{R}_+$	NM	*Separation buffer (type I)* between a_i and $a_{1,i}$
$\check{t}_{\mathrm{B}}^{a_{1,i},c}$	$\mathbb{R}_+$	s	*Time buffer* between a_i and $a_{1,i}$

if ($\{a_i\} \in \mathbb{A}^{c_1}$) **then**
    $\check{t}_{\mathrm{B}}^{a_{1,i},c} = t_{\mathrm{B}}^{a_{1,i},c_1}$
    if ($W^{a_{1,i},c} == W_{\mathrm{H}} \wedge W^{a_i,c} == W_{\mathrm{M}}$) **then**
        $\check{t}_{\mathrm{B}}^{a_{1,i},c}$ += $t_{\mathrm{B,HM}}^{a_{1,i},c_1}$
    else if ($W^{a_{1,i},c} == W_{\mathrm{M}} \wedge W^{a_i,c} == W_{\mathrm{H}}$) **then**
        $\check{t}_{\mathrm{B}}^{a_{1,i},c}$ += $t_{\mathrm{B,MH}}^{a_{1,i},c_1}$
    end if
else if ($\{a_i\} \in \mathbb{A}^{c_2}$) **then**
    $\check{t}_{\mathrm{B}}^{a_{1,i},c} = t_{\mathrm{B}}^{a_{1,i},c_2}$
    if ($W^{a_{1,i},c} == W_{\mathrm{H}} \wedge W^{a_i,c} == W_{\mathrm{M}}$) **then**
        $\check{t}_{\mathrm{B}}^{a_{1,i},c}$ += $t_{\mathrm{B,HM}}^{a_{1,i},c_2}$
    else if ($W^{a_{1,i},c} == W_{\mathrm{M}} \wedge W^{a_i,c} == W_{\mathrm{H}}$) **then**
        $\check{t}_{\mathrm{B}}^{a_{1,i},c}$ += $t_{\mathrm{B,MH}}^{a_{1,i},c_2}$
    end if
else if ($\{a_i\} \in \mathbb{A}^{c_3}$) **then**
    $\check{t}_{\mathrm{B}}^{a_{1,i},c} = t_{\mathrm{B}}^{a_{1,i},c_3}$
    if ($W^{a_{1,i},c} == W_{\mathrm{H}} \wedge W^{a_i,c} == W_{\mathrm{M}}$) **then**
        $\check{t}_{\mathrm{B}}^{a_{1,i},c}$ += $t_{\mathrm{B,HM}}^{a_{1,i},c_3}$
    else if ($W^{a_{1,i},c} == W_{\mathrm{M}} \wedge W^{a_i,c} == W_{\mathrm{H}}$) **then**
        $\check{t}_{\mathrm{B}}^{a_{1,i},c}$ += $t_{\mathrm{B,MH}}^{a_{1,i},c_3}$
    end if
end if
$\check{s}_{\mathrm{B}}^{a_{1,i},c} = v_{\mathrm{GS}}^{a_i,c} \cdot \check{t}_{\mathrm{B}}^{a_{1,i},c} / 3600$
if ($j_{\mathbb{A}_S}^{a_i,c} > 0$) **then**
    $s_{\mathrm{S}}^{a_{1,i},c} = \mathcal{F}_{s_{\mathrm{S}}}(W^{a_{1,i},c}, W^{a_i,c})$
end if
$s_{\mathrm{U}}^{a_{1,i},c} = s_{\mathrm{S}}^{a_{1,i},c} + \check{s}_{\mathrm{B}}^{a_{1,i},c}$

Check *vector outbound STAR* requirement for a_i

$b_{\mathrm{R},I_{2,1}}^{a_i,c}$ = false
$b_{\mathrm{R},I_{2,2}}^{a_i,c}$ = false
if ($j_{\mathbb{A}_S}^{a_i,c} > 0 \wedge (\langle STAR \rangle \lfloor SA\ operation \rfloor [a_i, c] \vee \langle vector\ inbound\ STAR \rangle \lfloor SA\ operation \rfloor [a_i, c]) \wedge \neg \langle holding \rangle \lfloor SA\ operation \rfloor [a_i, c] \wedge \neg \langle holding \rangle \lfloor SA\ operation \rfloor [a_{1,i}, c]$) **then**
    if ($s^{a_{1,i},c} < s_{\mathrm{S}}^{a_{1,i},c}$) **then**
        $b_{\mathrm{R},I_{2,1}}^{a_i,c}$ = true
    else if ($s^{a_{1,i},c} < s_{\mathrm{U}}^{a_{1,i},c}$) **then**
        $b_{\mathrm{R},I_{2,2}}^{a_i,c}$ = true
    end if

121: **end if**

Decide if a_i is available for a *vector inbound STAR* instruction

Variable	State space	Unit	Description
$\check{s}_{\mathrm{B},I_3}^{a_{1,i},c}$	$\mathbb{R}$	NM	*Separation buffer (type II)* between a_i and $a_{1,i}$ when instructing I_3

122: $\check{s}_{\mathrm{B},I_3}^{a_{1,i},c} = \mathcal{F}_{s_\mathrm{B}}(d_{\mathrm{A},w_{S,k}}^{a_i,c}, M^c)$
123: $b_{\mathrm{A},I_3}^{a_i,c}$ = false
124: **if** $((j_{\mathbb{A}_S}^{a_i,c} == 0 \wedge \langle$*vector outbound STAR*$\rangle\lfloor$*SA operation*$\rfloor[a_i, c]) \vee I^{a_i,c} == I_3)$ **then**
125: $b_{\mathrm{A},I_3}^{a_i,c}$ = true
126: **else if** $((j_{\mathbb{A}_S}^{a_i,c} > 0 \wedge (s^{a_{1,i},c} > s_\mathrm{U}^{a_{1,i},c} + \check{s}_{\mathrm{B},I_3}^{a_{1,i},c}) \wedge \langle$*vector outbound STAR*$\rangle\lfloor$*SA operation*$\rfloor[a_i, c]) \vee I^{a_i,c} == I_3)$ **then**
127: $b_{\mathrm{A},I_3}^{a_i,c}$ = true
128: **else if** $(\langle$*vector required*$\rangle\lfloor$*SA vector outbound merge requirement*$\rfloor[a_i, c])$ **then**
129: $b_{\mathrm{A},I_3}^{a_i,c}$ = false
130: **end if**

Check *holding entry* requirement for a_i

Variable	State space	Unit	Description
$\check{a}$	$\mathbb{A}$	-	Iterator, i.e. specific aircraft in $\mathbb{A}$, starting at $\mathbb{A}[0]$

131: $b_{\mathrm{R},I_{15}}^{a_i,c}$ = false
132: Q_{a,I_5} = 0
133: **for** $(\forall \check{a} \in \mathbb{A})$ **do**
134: **if** $(\langle$*vector outbound merge*$\rangle\lfloor$*SA operation*$\rfloor[\check{a}, c])$ **then**
135: Q_{a,I_5}++
136: **end if**
137: **end for**
138: **if** $(k^{a_i,c} \leq k_{I_{15},S}^{a_i,c} \wedge (Q_{a,I_5} \geq Q_{\mathrm{b},I_{15}} \vee |\mathbb{A}_{S,H}| > 0) \wedge \neg\langle$*assigned holding entry*$\rangle\lfloor$*SA holding entry*$\rfloor[a_i, c])$ **then**
139: $b_{\mathrm{R},I_{15}}^{a_i,c}$ = true
140: **end if**

Decide if a_i is available for *holding entry*

Parameter	Value	Unit	Description
$d_{l,w_{S,H,\mathrm{l}}}$	1	NM	Minimum allowed value of $d_{\mathrm{D},w_{S,H,\mathrm{l}}}^{a_i,c}$ for a feasible *holding entry* instruction
$d_{u,w_{S,H,\mathrm{l}}}$	5	NM	Maximum allowed value of $d_{\mathrm{D},w_{S,H,\mathrm{l}}}^{a_i,c}$ for a feasible *holding entry* instruction

141: $b_{\mathrm{A},I_{15}}^{a_i,c}$ = false
142: **if** $((k^{a_i,c} == k_{I_{15},S}^{a_i,c} \wedge (\langle$*STAR*$\rangle\lfloor$*SA operation*$\rfloor[a_i, c] \vee \langle$*vector inbound STAR*$\rangle\lfloor$*SA operation*$\rfloor[a_i, c]) \wedge d_{l,w_{S,H,\mathrm{l}}} \leq d_{\mathrm{D},w_{S,H,\mathrm{l}}}^{a_i,c} \leq d_{u,w_{S,H,\mathrm{l}}}) \vee I^{a_i,c} == I_{15})$ **then**
143: $b_{\mathrm{A},I_{15}}^{a_i,c}$ = true
144: **end if**

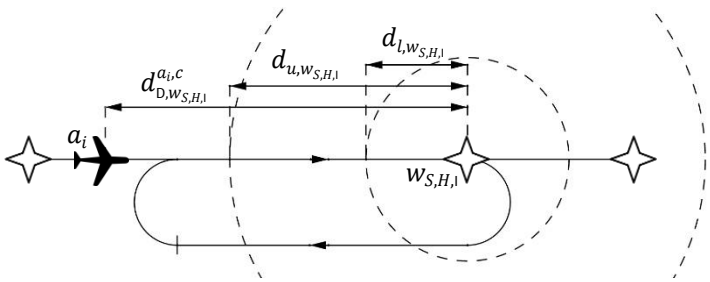


Figure 71 – Distances relative to $w_{S,H,\mathrm{l}}$ that are used to determine if a_i is available for holding entry

Decide if a_i is available for *holding exit*

Variable	State space	Unit	Description
$\breve{a}$	$\mathbb{A}^{c_3}$	-	Iterator, i.e. specific aircraft in $\mathbb{A}^{c_3}$, starting at $\mathbb{A}^{c_3}[0]$
$\breve{b}$	*Boolean*	-	Temporary Boolean
f_{s_U}	$\mathbb{R}_{>0}$	-	Multiplier of $s_\mathrm{U}^{a_{1,i},c}$ describing the rate at which a_i exits the holding
$\breve{k}^{\breve{a},c}$	$\mathbb{N}_0$	-	Current waypoint number of $\breve{a}$
$k_{I_4,S}^{\breve{a},c}$	$\mathbb{N}_0$	-	Waypoint (number) that corresponds to $w_{S,\mathrm{MO}}$
$k_{I_5,S}^{\breve{a},c}$	$\mathbb{N}_0$	-	Waypoint (number) that corresponds to $w_{S,\mathrm{MI}}$
$\breve{s}_{\mathrm{B},I_{16}}^{a_i,c}$	$\mathbb{R}$	NM	*Separation buffer (type II)* between a_i and $a_{1,i}$ when instructing I_{16}
Parameter	**Value**	**Unit**	**Description**
$d_{\mathrm{b},w_{S,H,\mathrm{I}}}$	1	-	Minimum allowed value of $d_{\mathrm{D},w_{S,H,\mathrm{I}}}^{a_i,c}$ for a feasible *holding exit* instruction
$f_{s_\mathrm{U},0}$	1	-	Multiplier of $s_\mathrm{U}^{a_{1,i},c}$ describing the default rate at which a_i exits the holding
$f_{s_\mathrm{U},1}$	3	-	Multiplier of $s_\mathrm{U}^{a_{1,i},c}$ describing the adjusted rate at which a_i exits the holding

$\breve{s}_{\mathrm{B},I_{16}}^{a_i,c} = \mathcal{F}_{s_\mathrm{B}}(d_{\mathrm{D},w_{S,H,\mathrm{I}}}^{a_i,c}, M^c)$
$b_{\mathrm{A},I_{16}}^{a_i,c}$ = false
$f_{s_\mathrm{U}} = f_{s_\mathrm{U},0}$
for ($\forall \breve{a} \in \mathbb{A}^{c_3}$) **do**
    if ($\breve{k}^{\breve{a},c} == k_{I_4,S}^{\breve{a},c} \vee \breve{k}^{\breve{a}} == k_{I_5,S}^{\breve{a},c}$) **then**
    $f_{s_\mathrm{U}} = f_{s_\mathrm{U},1}$
    break
    end if
end for
$\breve{b}$ = false
if ($j_{\mathbb{A}_S}^{a_i,c} == 0$) **then**
    $\breve{b}$ = true
else if ($j_{\mathbb{A}_S}^{a_i,c} > 0 \wedge (s^{a_{1,i},c} > f_{s_\mathrm{U}} \cdot s_\mathrm{U}^{a_{1,i},c} + \breve{s}_{\mathrm{B},I_{16}}^{a_i,c})$)
    $\breve{b}$ = true
end if
if ($(j_{\mathbb{A}_{S,H}}^{a_i,c} == 0 \wedge \breve{b} \wedge d_{\mathrm{D},w_{S,H,\mathrm{I}}}^{a_i,c} > d_{\mathrm{b},w_{S,H,\mathrm{I}}} \wedge Q_{a,I_5} \leq Q_{\mathrm{b},I_{16}} \wedge \langle$*straight flight*$\rangle\lfloor$*Heading control*$\rfloor[$*MCP* $a_i]) \vee I^{a_i,c} == I_{16}$) **then**
    $b_{\mathrm{A},I_{16}}^{a_i,c}$ = true
end if

Check *holding altitude* requirement for a_i

$b_{\mathrm{R},I_{17}}^{a_i,c}$ = false
$z_{\mathrm{R},w_{S,H}}^{a_i,c} = z_{w_{S,H}}^{a_i,c}$
if ($\neg\langle$*holding*$\rangle\lfloor$*SA operation*$\rfloor[a_i, c]$) **then**
    $z_{\mathrm{R},w_{S,H}}^{a_i,c} = z_{w_{S,H}}^{a_i,c}$ + floor($|\mathbb{A}_{S,H}|$ / 2) · 1000
else if ($\langle$*holding*$\rangle\lfloor$*SA operation*$\rfloor[a_i, c]$) **then**
    $z_{\mathrm{R},w_{S,H}}^{a_i,c} = z_{w_{S,H}}^{a_i,c}$ + floor($j_{\mathbb{A}_{S,H}}^{a_i,c}$ / 2) · 1000
    if ($z_{|}^{a_i,c} \neq z_{\mathrm{R},w_{S,H}}^{a_i,c}$) **then**
        $b_{\mathrm{R},I_{17}}^{a_i,c}$ = true
    end if
end if

Update SA of controller about the situation between a_i and $a_{2,i}$

The third category of clustered actions describe the controller's situation awareness of aircraft a_i and $a_{2,i}$. This obtained information is used to decide upon the necessity and feasibility of the *vector outbound IF*, *vector inbound IF*, *vector outbound merge*, *vector inbound merge* and *vector outbound trombone* instructions.

Find available $a_{2,i}$

173: $j_{\mathbb{A}_{IF}}^{a_i,c} = (i\{a_i\} \in \mathbb{A}_{IF})$
174: **if** $(j_{\mathbb{A}_{IF}}^{a_i,c} > 0)$ **then**
175: $a_{2,i} = \mathbb{A}_{IF}[j_{\mathbb{A}_{IF}}^{a_i,c} - 1]$
176: obtain $W^{a_{2,i},c}$, $d_{D,w_{IF}}^{a_{2,i},c}$, $d_{V,w_{IF}}^{a_{2,i},c}$ and $d_{V,w_{THR}}^{a_{2,i},c}$
177: **end if**

Observe *actual spacing* between a_i and $a_{2,i}$

178: **if** $(j_{\mathbb{A}_{IF}}^{a_i,c} > 0)$ **then**
179: $s_D^{a_{2,i},c} = d_{D,w_{IF}}^{a_i,c} - d_{D,w_{IF}}^{a_{2,i},c}$ //figure 72a
180: **if** $(\langle vector\ IF\rangle\lfloor SA\ operation\rfloor[a_{2,i}, c] \vee (\langle STAR\rangle\lfloor SA\ operation\rfloor[a_{2,i}, c] \wedge \langle final\ segment\rangle\lfloor SA\ STAR\ progress\rfloor[a_{2,i}, c]))$ **then**
181: $s_V^{a_{2,i},c} = d_{V,w_{IF}}^{a_i,c} - d_{D,w_{IF}}^{a_{2,i},c}$ //figure 72b
182: **else**
183: $s_V^{a_{2,i},c} = d_{V,w_{IF}}^{a_i,c} - d_{V,w_{IF}}^{a_{2,i},c}$ //figure 72c
184: **end if**
185: **end if**

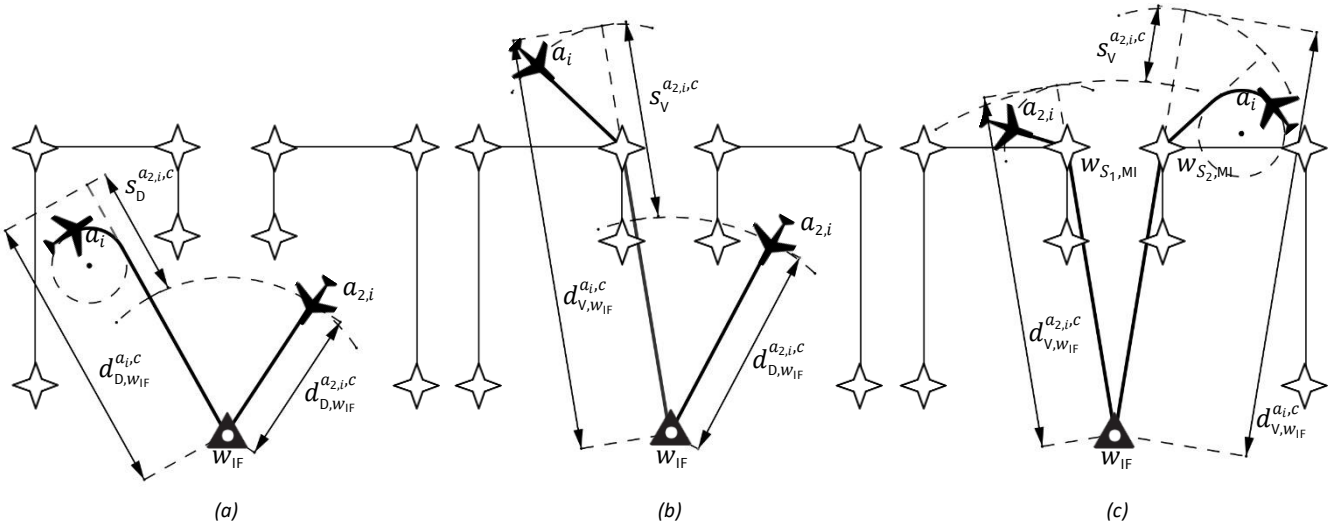


Figure 72 – Three different types of observed actual separation distances between a_i and $a_{2,i}$

Define *separation minima* and *desired spacing* between a_i and $a_{2,i}$

Variable	State space	Unit	Description
$\check{s}_B^{a_{2,i},c}$	$\mathbb{R}_+$	NM	*Separation buffer (type I)* between a_i and $a_{2,i}$
$\check{t}_B^{a_{2,i},c}$	$\mathbb{R}_+$	s	*Time buffer* between a_i and $a_{2,i}$

186: $\check{t}_B^{a_{2,i},c} = t_B^{a_{2,i},c_3}$
187: **if** $(W^{a_{2,i},c} == W_H \wedge W^{a_i,c} == W_M)$ **then**
188: $\check{t}_B^{a_{2,i},c}$ += $t_{B,HM}^{a_{2,i},c_3}$
189: **else if** $(W^{a_{2,i},c} == W_M \wedge W^{a_i,c} == W_H)$ **then**
190: $\check{t}_B^{a_{2,i},c}$ += $t_{B,MH}^{a_{2,i},c_3}$
191: **end if**
192: $\check{s}_B^{a_{2,i},c} = v_{GS}^{a_i,c} \cdot \check{t}_B^{a_{2,i},c}$ / 3600
193: **if** $(j_{\mathbb{A}_{IF}}^{a_i,c} > 0)$ **then**
194: $s_S^{a_{2,i},c} = \mathcal{F}_{s_S}(W^{a_{2,i},c}, W^{a_i,c})$

195: **end if**
196: $s_{\mathrm{U}}^{a_{2,i},c} = s_{\mathrm{S}}^{a_{2,i},c} + \breve{s}_{\mathrm{B}}^{a_{2,i},c}$

Check *vector outbound IF* requirement for a_i

197: $b_{\mathrm{R},I_{6,1}}^{a_i,c}$ = false
198: $b_{\mathrm{R},I_{6,2}}^{a_i,c}$ = false
199: **if** ($j_{\mathbb{A}_{\mathrm{IF}}}^{a_i,c} > 0 \wedge \langle$*vector inbound IF*$\rangle\lfloor$*SA operation*$\rfloor[a_i, c]$) **then**
200: **if** ($s_{\mathrm{D}}^{a_{2,i},c} < s_{\mathrm{S}}^{a_{2,i},c}$) **then**
201: $b_{\mathrm{R},I_{6,1}}^{a_i,c}$ = true
202: **else if** ($s_{\mathrm{D}}^{a_{2,i},c} < s_{\mathrm{U}}^{a_{2,i},c}$) **then**
203: $b_{\mathrm{R},I_{6,2}}^{a_i,c}$ = true
204: **end if**
205: **end if**

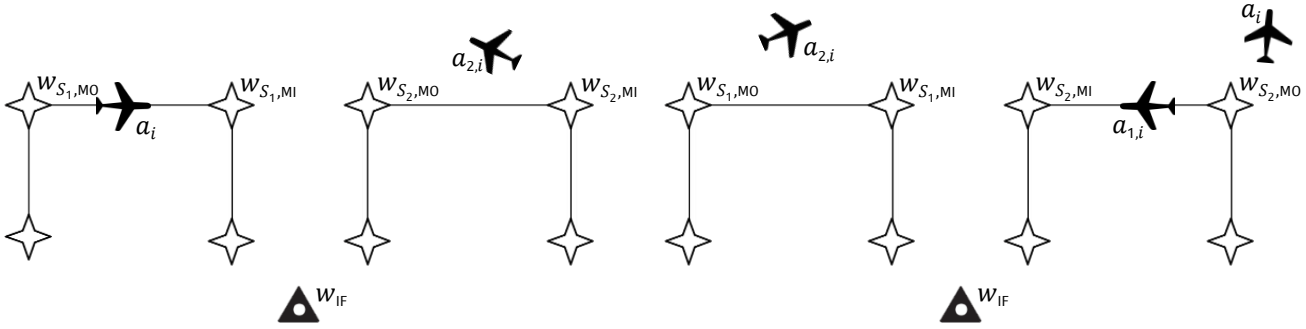
Decide if a_i is available for a *vector inbound IF* instruction

Variable	State space	Unit	Description
$\breve{s}_{\mathrm{B},I_7}^{a_{2,i},c}$	$\mathbb{R}$	NM	*Separation buffer (type II)* between a_i and $a_{2,i}$ when instructing I_7

206: $\breve{s}_{\mathrm{B},I_7}^{a_{2,i},c} = \mathcal{F}_{s_{\mathrm{B}}}(d_{\mathrm{D},w_{\mathrm{IF}}}^{a_i,c}, M^c)$
207: $b_{\mathrm{A},I_7}^{a_i,c}$ = false
208: **if** ($j_{\mathbb{A}_{\mathrm{IF}}}^{a_i,c} == 0$) **then**
209: $b_{\mathrm{A},I_7}^{a_i,c}$ = true
210: **else if** ($j_{\mathbb{A}_{\mathrm{IF}}}^{a_i,c} > 0 \wedge (s_{\mathrm{D}}^{a_{2,i},c} > s_{\mathrm{U}}^{a_{2,i},c} + \breve{s}_{\mathrm{B},I_7}^{a_{2,i},c}) \wedge \langle$*vector inbound IF*$\rangle\lfloor$*SA operation*$\rfloor[a_{2,i}, c]$) **then**
211: $b_{\mathrm{A},I_7}^{a_i,c}$ = true
212: **end if**
213: **if** (($b_{\mathrm{A},I_7}^{a_i,c} \wedge (\langle$*STAR*$\rangle\lfloor$*SA operation*$\rfloor[a_i, c] \vee \langle$*vector outbound IF*$\rangle\lfloor$*SA operation*$\rfloor[a_i, c] \vee (\langle$*vector outbound STAR*$\rangle\lfloor$*SA operation*$\rfloor[a_i, c] \wedge \langle$*final segment*$\rangle\lfloor$*SA STAR progress*$\rfloor[a_i, c])) \wedge \neg\langle$*vector merge*$\rangle\lfloor$*SA operation*$\rfloor[a_i, c] \wedge \beta^{a_i,c} < u_\beta) \vee I^{a_i,c} == I_7$) **then**
214: $b_{\mathrm{A},I_7}^{a_i,c}$ = true
215: **else**
216: $b_{\mathrm{A},I_7}^{a_i,c}$ = false
217: **end if**

Check *vector outbound merge* requirement for a_i

218: $b_{\mathrm{R},I_4}^{a_i,c}$ = false
219: **if** ($\langle$*STAR*$\rangle\lfloor$*SA operation*$\rfloor[a_i, c] \wedge \langle$*base segment*$\rangle\lfloor$*SA STAR progress*$\rfloor[a_i, c] \wedge \langle$*not available for vector*$\rangle\lfloor$*SA vector inbound IF*$\rfloor[a_i, c]$) **then** //figure 73a
220: $b_{\mathrm{R},I_4}^{a_i,c}$ = true
221: **end if**
222: **if** ($\langle$*vector STAR*$\rangle\lfloor$*SA operation*$\rfloor[a_i, c] \wedge \langle$*base segment*$\rangle\lfloor$*SA STAR progress*$\rfloor[a_i, c] \wedge j_{\mathbb{A}_S}^{a_i} > 0 \wedge \langle$*vector merge*$\rangle\lfloor$*SA operation*$\rfloor[a_{1,i}, c]$) **then** //figure 73b
223: $b_{\mathrm{R},I_4}^{a_i,c}$ = true
224: **end if**



(a) a_i flying STAR procedure *(b) a_i flying vector STAR*

Figure 73 – Two typical configurations between a_i and $a_{2,i}$ that require a_i to fly a vector outbound merge operation

ecide if a_i is available for a *vector inbound merge* instruction

Variable	State space	Unit	Description
$\check{s}^{a_i,c}_{\text{B},I_5}$	$\mathbb{R}$	NM	*Separation buffer (type II)* between a_i and $a_{1,i}$, and between a_i and $a_{2,i}$ when instructing I_5

$\check{s}^{a_i,c}_{\text{B},I_5} = \mathcal{F}_{s_\text{B}}(d^{a_i}_{\text{A},w_{S,k}}, M^c)$
$b^{a_i,c}_{\text{A},I_5}$ = false
if ($j^{a_i,c}_{\mathbb{A}_\text{IF}}$ == 0 ∨ ($j^{a_i,c}_{\mathbb{A}_\text{IF}}$ > 0 ∧ $s^{a_{2,i},c}_\text{V} > s^{a_{2,i},c}_\text{U} + \check{s}^{a_i,c}_{\text{B},I_5}$)) **then**
    $b^{a_i,c}_{\text{A},I_5}$ = true
end if
if ($b^{a_i,c}_{\text{A},I_5}$ ∧ ($j^{a_i,c}_{\mathbb{A}_S}$ == 0 ∨ ($j^{a_i,c}_{\mathbb{A}_S}$ > 0 ∧ $s^{a_{1,i},c} > s^{a_{1,i},c}_\text{U} + \check{s}^{a_i,c}_{\text{B},I_5}$))) **then**
    $b^{a_i,c}_{\text{A},I_5}$ = true
else
    $b^{a_i,c}_{\text{A},I_5}$ = false
end if
if (¬⟨*vector outbound merge*⟩⌊*SA operation*⌋[a_i, c]) **then**
    $b^{a_i,c}_{\text{A},I_5}$ = false
end if
if ($I^{a_i,c}$ == I_5) **then**
    $b^{a_i,c}_{\text{A},I_5}$ = true
end if

Check *vector outbound trombone* requirement for a_i

Parameter	Value	Unit	Description
d_{b,w_IF}	6	NM	Distance around w_IF that marks the area where conflicting aircraft (a_i) will be instructed the *vector outbound trombone* operation

$b^{a_i,c}_{\text{R},I_8}$ = false
if ((⟨*STAR*⟩⌊*SA operation*⌋[a_i, c] ∧ (⟨*passed final waypoint*⟩⌊*SA STAR progress*⌋[a_i, c] ∨ ⟨*nearing final waypoint*⟩⌊*SA STAR progress*⌋[a_i, c])) ∨ (⟨*vector outbound STAR*⟩⌊*SA operation*⌋[a_i, c] ∧ ⟨*passed final waypoint*⟩⌊*SA STAR progress*⌋[a_i, c])) ∧ ⟨*not available for vector*⟩⌊*SA vector inbound IF*⌋[a_i, c]) **then**
    $b^{a_i,c}_{\text{R},I_8}$ = true
else if (⟨*vector inbound IF*⟩⌊*SA operation*⌋[a_i, c] ∧ ¬⟨*in no conflict*⟩⌊*SA vector outbound IF requirement*⌋[a_i, c] ∧ $d^{a_i,c}_{\text{D},w_\text{IF}} < d_{\text{b},w_\text{IF}}$) **then**
    $b^{a_i,c}_{\text{R},I_8}$ = true
else if (⟨*vector outbound IF*⟩⌊*SA operation*⌋[a_i, c] ∧ $\beta^{a_i,c} > u_\beta$) **then**
    $b^{a_i,c}_{\text{R},I_8}$ = true
end if

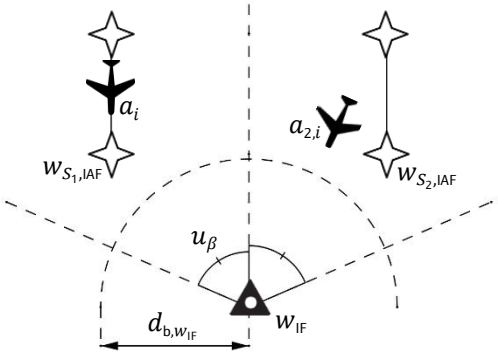


Figure 74 – Example situation in which a_i is likely to receive a vector outbound trombone instruction

Update SA of controller about the situation between a_i and $a_{3,i}$

The fourth category of clustered actions describe the controller's situation awareness of aircraft a_i and $a_{3,i}$. This obtained information is used to decide upon the necessity and feasibility of the *landing clearance* and *go-around* instructions.

Find available $a_{3,i}$

249: $j_{\mathbb{A}_{\text{FA}}}^{a_i,c} = (i\{a_i\} \in \mathbb{A}_{\text{FA}})$
250: **if** $(j_{\mathbb{A}_{\text{FA}}}^{a_i,c} > 0)$ **then**
251: $\quad a_{3,i} = \mathbb{A}_{\text{FA}}[j_{\mathbb{A}_{\text{FA}}}^{a_i,c} - 1]$
252: $\quad$ obtain $W^{a_{3,i},c}$, $d_{\text{V},w_{\text{THR}}}^{a_{3,i},c}$
253: **end if**

Observe *actual spacing* between a_i and $a_{3,i}$

254: **if** $(j_{\mathbb{A}_{\text{FA}}}^{a_i,c} > 0)$ **then**
255: $\quad s^{a_{3,i},c} = d_{\text{V},w_{\text{THR}}}^{a_i,c} - d_{\text{V},w_{\text{THR}}}^{a_{3,i},c}$
256: **end if**

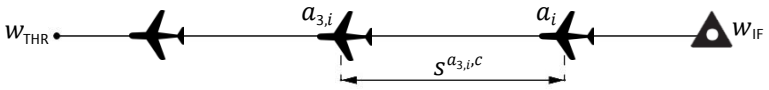


Figure 75 – Visualization of the observed actual spacing between a_i and $a_{3,i}$

Define *separation minima* and *desired spacing* between a_i and $a_{3,i}$

Variable	State space	Unit	Description
$\breve{s}_{\text{B}}^{a_{3,i},c}$	$\mathbb{R}_+$	NM	*Separation buffer (type I)* between a_i and $a_{3,i}$
$\breve{t}_{\text{B}}^{a_{3,i},c}$	$\mathbb{R}_+$	s	*Time buffer* between a_i and $a_{3,i}$

257: $\breve{t}_{\text{B}}^{a_{3,i},c} = t_{\text{B}}^{a_{3,i},c_4}$
258: **if** $(W^{a_{3,i},c} == W_{\text{H}} \wedge W^{a_i,c} == W_{\text{M}})$ **then**
259: $\quad \breve{t}_{\text{B}}^{a_{3,i},c}$ += $t_{\text{B,HM}}^{a_{3,i},c_4}$
260: **else if** $(W^{a_{3,i},c} == W_{\text{M}} \wedge W^{a_i,c} == W_{\text{H}})$ **then**
261: $\quad \breve{t}_{\text{B}}^{a_{3,i},c}$ += $t_{\text{B,MH}}^{a_{3,i},c_4}$
262: **end if**
263: $\breve{s}_{\text{B}}^{a_{3,i},c} = v_{\text{GS}}^{a_i,c} \cdot \breve{t}_{\text{B}}^{a_{3,i},c}$ / 3600
264: **if** $(j_{\mathbb{A}_{\text{FA}}}^{a_i,c} > 0)$ **then**
265: $\quad s_{\text{S}}^{a_{3,i},c} = \mathcal{F}_{s_{\text{S}}}(W^{a_{3,i},c}, W^{a_i,c})$
266: **end if**
267: $s_{\text{U}}^{a_{3,i},c} = s_{\text{S}}^{a_{3,i},c} + \breve{s}_{\text{B}}^{a_{3,i},c}$

Decide if a_i is available for *landing clearance*

$b_{\mathrm{A},I_{13}}^{a_i,c}$ = false
$b_{\mathrm{R},I_{14}}^{a_i,c}$ = false
if ($\langle$*final approach*$\rangle\lfloor$*SA operation*$\rfloor[a_i, c] \wedge j_{\mathbb{A}_{\mathrm{FA}}}^{a_i,c} > 0 \wedge s^{a_{3,i},c} < s_{\mathrm{U}}^{a_{3,i},c}$) **then**
    $b_{\mathrm{R},I_{14}}^{a_i,c}$ = true
else if ($\langle$*final approach*$\rangle\lfloor$*SA operation*$\rfloor[a_i, c] \wedge j_{\mathbb{A}_{\mathrm{FA}}}^{a_i,c} == 0$) **then**
    $b_{\mathrm{A},I_{13}}^{a_i,c}$ = true
end if

Update SA of controller about the situation between a_i, $a_{4,i}$ and $a_{5,i}$

The fifth category of clustered actions describe the controller's situation awareness of aircraft a_i, $a_{4,i}$ and $a_{5,i}$. This obtained information is used to decide upon the feasibility of the *vector inbound trombone* instruction.

Find available $a_{4,i}$ and $a_{5,i}$

Variable	State space	Unit	Description
$\breve{a}_4$	$\mathbb{A}_S$	-	Iterator, i.e. specific aircraft in $\mathbb{A}_S$, starting at the last item of $\mathbb{A}_S$, i.e. at $\mathbb{A}_S[\lvert\mathbb{A}_S\rvert]$
$\breve{a}_5$	$\mathbb{A}_S$	-	Iterator, i.e. specific aircraft in $\mathbb{A}_S$, starting at the first item of $\mathbb{A}_S$, i.e. at $\mathbb{A}_S[0]$
$\breve{k}^{\breve{a}_4}$	$\mathbb{N}_0$	-	Current waypoint number of $\breve{a}_4$
$\breve{k}^{\breve{a}_5}$	$\mathbb{N}_0$	-	Current waypoint number of $\breve{a}_5$

for ($\forall \breve{a}_4 \in \mathbb{A}_S$) **do** //reverse order iteration, starting at the last element of the set
    if ($\lvert\mathbb{A}_S\rvert > 0 \wedge \breve{k}^{\breve{a}_4} > k_{I_9,S}^{a_i,c} \wedge \breve{a}_4 \neq a_i$) **then**
        $a_{4,i} == \breve{a}_4$
        obtain $W^{a_{4,i},c}$ and $d_{\mathrm{V},w_{\mathrm{THR}}}^{a_{4,i},c}$
        break
    end if
end for
if ($\lvert\mathbb{A}_{I_9,S}\rvert > 0 \wedge \mathbb{A}_{I_9,S}[\lvert\mathbb{A}_{I_9,S}\rvert - 1] \neq a_i$) **then** //replace $a_{4,i}$ if there are already aircraft operating the *vector inbound trombone*
    $a_{4,i} == \mathbb{A}_{I_9,S}[\lvert\mathbb{A}_{I_9,S}\rvert - 1]$
    obtain $W^{a_{4,i},c}$ and $d_{\mathrm{V},w_{\mathrm{THR}}}^{a_{4,i},c}$
end if
for ($\forall \breve{a}_5 \in \mathbb{A}_S$) **do** //iteration will start at first element of the set
    if ($\breve{k}^{\breve{a}_5} \leq k_{I_9,S}^{a_i,c} \wedge \breve{a}_5 \neq a_i \wedge \neg\langle$*vector inbound trombone*$\rangle\lfloor$*SA operation*$\rfloor[\breve{a}_5, c]$) **then**
        $a_{5,i} == \breve{a}_5$
        obtain $W^{a_{5,i},c}$, $d_{\mathrm{V},w_{\mathrm{THR}}}^{a_{5,i},c}$ and $v_{\mathrm{GS}}^{a_{5,i},c}$
        break
    end if
end for

Observe *actual spacing* between a_i and $a_{4,i}$, and between a_i and $a_{5,i}$

if ($\exists a_{4,i}$) **then**
    $s^{a_{4,i},c} = d_{\mathrm{V},w_{\mathrm{THR}}}^{a_i,c} - d_{\mathrm{V},w_{\mathrm{THR}}}^{a_{4,i},c}$
end if
if ($\exists a_{5,i}$) **then**
    $s^{a_{5,i},c} = d_{\mathrm{V},w_{\mathrm{THR}}}^{a_{5,i},c} - d_{\mathrm{V},w_{\mathrm{THR}}}^{a_i,c}$
end if

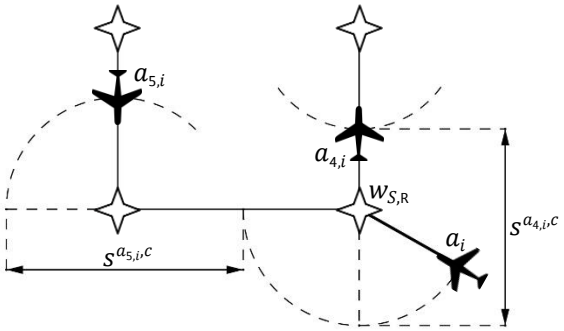


Figure 76 – Visualization of the observed actual spacing between a_i and $a_{4,i}$, and between a_i and $a_{5,i}$

Define *separation minima* and *desired spacing* between a_i and $a_{4,i}$, and between a_i and $a_{5,i}$

Variable	State space	Unit	Description
$\check{s}_{\mathrm{B}}^{a_{4,i},c}$	$\mathbb{R}_+$	NM	*Separation buffer (type I)* between a_i and $a_{4,i}$
$\check{s}_{\mathrm{B}}^{a_{5,i},c}$	$\mathbb{R}_+$	NM	*Separation buffer (type I)* between a_i and $a_{5,i}$
$\check{t}_{\mathrm{B}}^{a_{4,i},c}$	$\mathbb{R}_+$	s	*Time buffer* between a_i and $a_{4,i}$
$\check{t}_{\mathrm{B}}^{a_{5,i},c}$	$\mathbb{R}_+$	s	*Time buffer* between a_i and $a_{5,i}$

$\check{t}_{\mathrm{B}}^{a_{4,i},c} = t_{\mathrm{B}}^{a_{1,i},c_3}$
if $(W^{a_{4,i},c} == W_{\mathrm{H}} \wedge W^{a_i,c} == W_{\mathrm{M}})$ **then**
    $\check{t}_{\mathrm{B}}^{a_{4,i},c} \mathrel{+}= t_{\mathrm{B,HM}}^{a_{1,i},c_3}$
else if $(W^{a_{4,i},c} == W_{\mathrm{M}} \wedge W^{a_i,c} == W_{\mathrm{H}})$ **then**
    $\check{t}_{\mathrm{B}}^{a_{4,i},c} \mathrel{+}= t_{\mathrm{B,MH}}^{a_{1,i},c_3}$
end if
$\check{s}_{\mathrm{B}}^{a_{4,i},c} = v_{\mathrm{GS}}^{a_i,c} \cdot \check{t}_{\mathrm{B}}^{a_{4,i},c} \,/\, 3600$
if $(\exists a_{4,i})$ **then**
    $s_{\mathrm{S}}^{a_{4,i},c} = \mathcal{F}_{s_{\mathrm{S}}}(W^{a_{4,i},c}, W^{a_i,c})$
end if
$s_{\mathrm{U}}^{a_{4,i},c} = s_{\mathrm{S}}^{a_{4,i},c} + \check{s}_{\mathrm{B}}^{a_{4,i},c}$
$\check{t}_{\mathrm{B}}^{a_{5,i},c} = t_{\mathrm{B}}^{a_{1,i},c_3}$
if $(W^{a_{5,i},c} == W_{\mathrm{H}} \wedge W^{a_i,c} == W_{\mathrm{M}})$ **then**
    $\check{t}_{\mathrm{B}}^{a_{5,i},c} \mathrel{+}= t_{\mathrm{B,HM}}^{a_{1,i},c_3}$
else if $(W^{a_{5,i},c} == W_{\mathrm{M}} \wedge W^{a_i,c} == W_{\mathrm{H}})$ **then**
    $\check{t}_{\mathrm{B}}^{a_{5,i},c} \mathrel{+}= t_{\mathrm{B,MH}}^{a_{1,i},c_3}$
end if
$\check{s}_{\mathrm{B}}^{a_{5,i},c} = v_{\mathrm{GS}}^{a_{5,i},c} \cdot \check{t}_{\mathrm{B}}^{a_{5,i},c} \,/\, 3600$
if $(\exists a_{5,i})$ **then**
    $s_{\mathrm{S}}^{a_{5,i},c} = \mathcal{F}_{s_{\mathrm{S}}}(W^{a_{5,i},c}, W^{a_i,c})$
end if
$s_{\mathrm{U}}^{a_{5,i},c} = s_{\mathrm{S}}^{a_{5,i},c} + \check{s}_{\mathrm{B}}^{a_{5,i},c}$

Decide if a_i is available for a *vector inbound trombone* instruction

Variable	State space	Unit	Description
$s_{\mathrm{B},I_9}^{a_i,c}$	$\mathbb{R}$	NM	*Separation buffer (type II)* between a_i and $a_{4,i}$ when instructing I_9

$s_{\mathrm{B},I_9}^{a_i,c} = \mathcal{F}_{s_{\mathrm{B}}}(d_{\mathrm{A},w_{S,k}}^{a_i,c}, M^c)$
$b_{\mathrm{A},I_9}^{a_{4,i},c}$ = false
if $(\nexists a_{4,i})$ **then**
    $b_{\mathrm{A},I_9}^{a_{4,i},c}$ = true
else if $(\exists a_{4,i} \wedge (s^{a_{4,i},c} > s_{\mathrm{U}}^{a_{4,i},c} + s_{\mathrm{B},I_9}^{a_i,c}))$ **then**
    $b_{\mathrm{A},I_9}^{a_{4,i},c}$ = true

327: **end if**
328: $b_{A,I_9}^{a_{5,i},c}$ = false
329: **if** ($\nexists a_{5,i}$) **then**
330: $b_{A,I_9}^{a_{5,i},c}$ = true
331: **else if** ($\exists a_{5,i} \wedge s^{a_{5,i},c} > s_U^{a_{5,i},c}$) **then**
332: $b_{A,I_9}^{a_{5,i},c}$ = true
333: **end if**
334: $b_{A,I_9}^{a_i,c}$ = false
335: **if** (($b_{A,I_9}^{a_{4,i},c} \wedge b_{A,I_9}^{a_{5,i},c} \wedge \langle$*vector outbound trombone phase 2*$\rangle\lfloor$*SA operation*$\rfloor[a_i, c] \wedge |\mathbb{A}_{I_8,S}| > 0 \wedge \mathbb{A}_{I_8,S}[0] == a_i) \vee I^{a_i,c} == I_9$) **then**
336: $b_{A,I_9}^{a_i,c}$ = true
337: **end if**

Update SA of controller about handover of a_i

The last category of clustered actions describe the controller's situation awareness of aircraft a_i about its availability for handover.

Decide if a_i is available for *handover to ARR*

338: $b_{A,I_{10}}^{a_i,c}$ = false
339: **if** ((($\{a_i\} \in \mathbb{A}^{c_1} \vee \{a_i\} \in \mathbb{A}^{c_2}) \wedge k^{a_i,c} > k_{I_{10},S}^{a_i,c} \wedge \langle$*in no conflict*$\rangle\lfloor$*SA vector outbound STAR requirement*$\rfloor[a_i, c] \wedge \langle$*STAR*$\rangle\lfloor$*SA operation*$\rfloor[a_i, c]) \vee I^{a_i,c} == I_{10}$) **then**
340: $b_{A,I_{10}}^{a_i,c}$ = true
341: **end if**

Decide if a_i is available for *handover to TWR*

342: $b_{A,I_{11}}^{a_i,c}$ = false
343: **if** (($\{a_i\} \in \mathbb{A}^{c_3} \wedge \langle$*final approach*$\rangle\lfloor$*SA operation*$\rfloor[a_i, c]) \vee I^{a_i,c} == I_{11}$) **then**
344: $b_{A,I_{11}}^{a_i,c}$ = true
345: **end if**

Decide if a_i is available for *handover to GND*

346: $b_{A,I_{12}}^{a_i,c}$ = false
347: **if** (($\{a_i\} \in \mathbb{A}^{c_4} \wedge (\langle$*constant ground run*$\rangle\lfloor$*SA operation*$\rfloor[a_i, c] \vee \langle$*vacate runway*$\rangle\lfloor$*SA operation*$\rfloor[a_i, c] \vee \langle$*runway vacated*$\rangle\lfloor$*SA operation*$\rfloor[a_i, c])) \vee I^{a_i,c} == I_{12}$) **then**
348: $b_{A,I_{12}}^{a_i,c}$ = true
349: **end if**

C.4.4.2 Update number of recently provided instructions

Event $\mathcal{E}_M^c$ models the situation awareness of controller c ($c \in \mathbb{C}$) about its current and recent workload. This situation awareness is periodically updated by keeping track of the number of provided instructions in the recent past ($\mathbb{T}_I^c$). Figure 77 depicts a general timeline with time points at which instructions have been provided by controller c. These time points are denoted by x-marks. The 'instructions that have been provided in the recent past' can be described as those that were instructed at time points that fall within the time interval t_I. Event $\mathcal{E}_M^c$ is meant to periodically update the current number of time points that lie within this time span t_I. All these found time points are collected in the set $\mathbb{T}_I^c$, which is used to define the controller's workload and with that the control mode wherein the controller is operating.

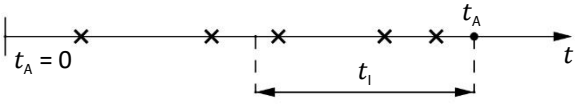


Figure 77 – Time points at which instructions have been provided by c, visualized by a x-mark

- First occurrence time: starts immediately
- Recurrence time period: $t_{R,I}$

Event $\mathcal{E}_M^c$			
Parameter	**Value**	**Unit**	**Description**
t_I	600	s	Time interval that is used to describe the workload of the controller in terms of provided instructions in the near past, i.e. described with respect to t_A
$t_{R,I}$	10	s	Recurrence time period of event $\mathcal{E}_M^c$
1: Update the time points within $\mathbb{T}_I^c$ with the most recent (instruction) time points, i.e. time points that lie within t_I			

Action 3 – Update the situation awareness of c ($\forall c \in \mathbb{C}$) about the number of recently provided instructions, i.e. the contents of $\mathbb{T}_I^c$

C.4.4.3 Manage merging sequence towards w_{IF}

Event $\mathcal{E}_{MS}^{a_i,c}$ manages the merging sequence towards w_{IF}, i.e. the sequence in which aircraft will be instructed the *vector inbound IF* operation. Aircraft will be added to the merging sequence once they pass $w_{S,R}$.

- Condition: $k^{a_i,c} > k_{I_9,S}^{a_i,c}$

Event $\mathcal{E}_{MS}^{a_i,c}$
1: $\{a_i\} \cup \mathbb{A}_{IF}$

Action 4 – Manage sequence in which aircraft will be vectored towards w_{IF}

C.4.5 Statecharts

C.4.5.1 Situation awareness

The situation awareness of controller c about the situation of a_i ($a_i \in \mathbb{A}^c$, $c \in \mathbb{C}$) is modelled using a set of statecharts that each describe a specific situation or situational element in terms of observation, reasoning and memory. Table 36 provides an overview of the SA-statecharts that are considered in the model specification.

Statechart	**Models the situation awareness of c about**	**Appendix**
⌊*SA operation*⌋	the operational mode of a_i during arrival/approach	C.4.5.1.1
⌊*SA STAR progress*⌋	the progress/position of a_i in the STAR procedure	C.4.5.1.2
⌊*SA vector instruction history*⌋	the time period since the last instructed vector to a_i	C.4.5.1.3
⌊*SA waypoint update history*⌋	the time period since the last update of $k^{a_i,c}$	C.4.5.1.4
⌊*SA STAR speed*⌋	the airspeed of a_i when flying the STAR procedure	C.4.5.1.5
⌊*SA STAR altitude*⌋	the altitude of a_i when flying the STAR procedure	C.4.5.1.6
⌊*SA STAR speed instruction requirement*⌋	the necessity of I_1	C.4.5.1.7
⌊*SA STAR altitude instruction requirement*⌋	the necessity of I_1	C.4.5.1.8
⌊*SA vector outbound STAR requirement*⌋	the necessity of $I_{2,1}$ and $I_{2,2}$	C.4.5.1.9
⌊*SA vector inbound STAR*⌋	the feasibility of I_3	C.4.5.1.10
⌊*SA vector outbound merge requirement*⌋	the necessity of I_4	C.4.5.1.11
⌊*SA vector outbound STAR history*⌋	the feasibility of I_4	C.4.5.1.12
⌊*SA vector inbound merge*⌋	the feasibility of I_5	C.4.5.1.13
⌊*SA vector outbound IF requirement*⌋	the necessity of $I_{6,1}$ and $I_{6,2}$	C.4.5.1.14
⌊*SA vector inbound IF*⌋	the feasibility of I_7	C.4.5.1.15
⌊*SA vector outbound trombone requirement*⌋	the necessity of I_8	C.4.5.1.16
⌊*SA vector inbound trombone*⌋	the feasibility of I_9	C.4.5.1.17
⌊*SA handover to ARR*⌋	the feasibility of I_{10}	C.4.5.1.18
⌊*SA handover to TWR*⌋	the feasibility of I_{11}	C.4.5.1.19
⌊*SA handover to GND*⌋	the feasibility of I_{12}	C.4.5.1.20

⌊*SA landing clearance*⌋	the feasibility of I_{13} and necessity of I_{14}	C.4.5.1.21
⌊*SA holding requirement*⌋	the necessity of I_{15}	C.4.5.1.22
⌊*SA holding entry*⌋	the feasibility of I_{15}	C.4.5.1.23
⌊*SA holding exit*⌋	the feasibility of I_{16}	C.4.5.1.24
⌊*SA holding altitude requirement*⌋	the necessity of I_{17}	C.4.5.1.25

Table 36 – List of statecharts that are used to describe the situation awareness of controller c ($c \in \mathbb{C}$) about the situation of a_i ($a_i \in \mathbb{A}^c$)

The majority of the SA-statecharts are related to the situation awareness of the controller about the necessity and feasibility of the modelled instructions. The Boolean variables that are evaluated in the condition triggered transitions of these statecharts are defined in event $\mathcal{E}_{SA}^{c}$ in appendix C.4.4.1. These specific SA-statecharts are each composed of five similar type of states. These specific states can be described with the following general descriptions:

- **⟨not available⟩**: indicates that a_i is not (yet) considered available for instruction I.
- **⟨available⟩**: indicates that a_i is considered available for instruction I.
- **⟨assigned⟩**: indicates that a_i has already been instructed instruction I.
- **⟨not required⟩**: indicates that instruction I is not considered required based on the observed situation of a_i.
- **⟨required⟩**: indicates that instruction I is considered required based on the observed situation of a_i.

C.4.5.1.1 SA operation

The ⌊*SA operation*⌋$[a_i, c]$ statechart (figure 78) models the situation awareness of the controller agent about the operational state of a_i.

⟨STAR⟩

- State entry actions:

1: Trigger transition t_6 of ⌊*SA STAR progress*⌋$[a_i, c]$

⟨vector outbound trombone phase 2⟩

- State entry actions:

1: $k^{a_i,c} = k_{I_9,S}^{a_i,c}$

- State exit actions:

1: $\{a_i\} \setminus \mathbb{A}_{I_8}$
2: $\{a_i\} \setminus \mathbb{A}_{I_8,S}$

⟨vector inbound trombone⟩

- State exit actions:

1: $\{a_i\} \setminus \mathbb{A}_{I_9,S}$

⟨intermediate approach⟩

- State entry actions:

1: $\{a_i\} \setminus \mathbb{A}_{IF}$
2: $\{a_i\} \cup \mathbb{A}_{FA}$
3: **if** $(v_{FA}^{a_i} < v_{IAS}^{a_i})$ **then**
4: $v_{IAS,S}^{a_i} = v_{FA}^{a_i}$
5: **end if**

⟨go-around⟩

- State entry actions:

1: $\{a_i\} \setminus \mathbb{A}_{FA}$

⟨deceleration⟩

Represents the deceleration phase of a_i just after touchdown at w_{THR} by using its brakes, spoilers and thrust reverse.

- State entry actions:

1: $\{a_i\} \setminus \mathbb{A}_{FA}$
2: $v_{IAS,S}^{a_i} = v_{HST}$

3: $\psi^{a_i} = \mathcal{C}^{\text{RAD}}_{\text{DEG}}(\text{atan2}(y_{w_{\text{HST1}}} - y^{a_i}, x_{w_{\text{HST1}}} - x^{a_i}))$

⟨constant ground run⟩

Represents the phase in which a_i taxies with a constant speed over the runway up to the moment when a_i is located next to w_{HST1} or w_{HST2}.

⟨vacate runway⟩

Represents the period in which a_i starts turning its nose wheel in order to vacate the runway up to the moment when it really has vacated the runway.

- State entry actions:

1: $\psi^{a_i}_{\text{A,S}} = \psi_{w_{\text{HST}}}$
2: $\psi^{a_i}_{\text{D,S}} = \psi_{w_{\text{HST}}}$
3: $r_{\text{T}} = r_{\text{HST}} \cdot f^{\text{M}}_{\text{NM}}$
4: Trigger t_1 of ⌊*Heading control*⌋[*MCP* a_i]

⟨runway vacated⟩

Represents the taxiing of a_i after it has vacated the runway. This state is still modelled to provide the TWR controller the ability to hand the aircraft over to GND if this has not been done yet when the aircraft was still on the runway.

- State entry actions:

1: $\{a_i\} \setminus \mathbb{A}$

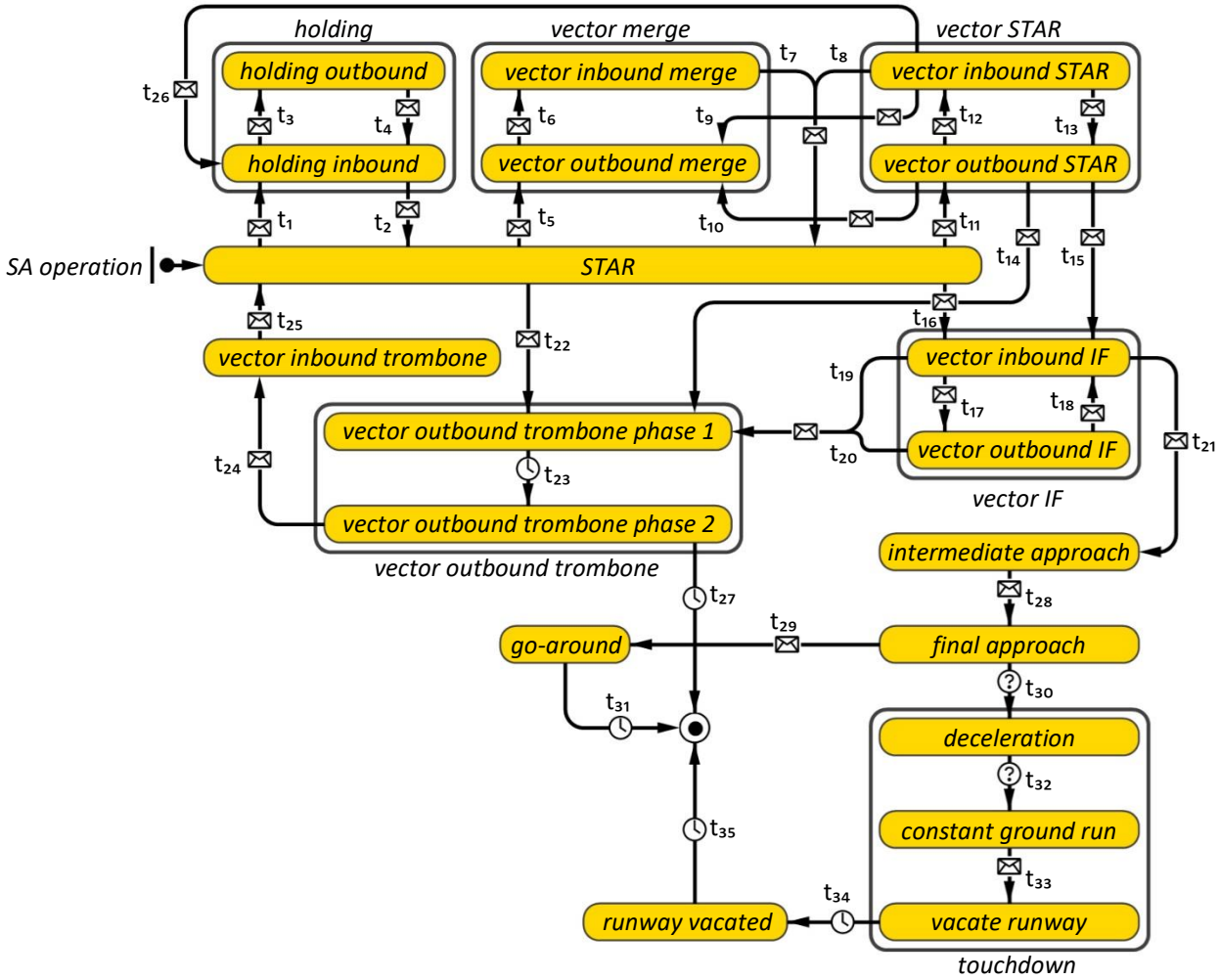


Figure 78 – ⌊SA operation⌋[a_i, c]

⟨final state⟩
Marks the last state of the $\lfloor$*SA operation*$\rfloor[a_i, c]$ statechart. At this stage the aircraft is removed from all aircraft collections and then removed from the model (i.e. simulation)

- State entry actions:

1: $\{a_i\} \setminus \mathbb{A}$

The remaining states do not have any entry- and/or exit actions

Transition t_{23}
The timeout value of $t_{t_{23}}$ is chosen such that a_i is given sufficient time to leave the dense airspace section between w_{IF} and $w_{S,IAF}$ before the controller may consider a_i available again for a *vector inbound trombone* instruction.

Parameter	Value	Unit	Description
$t_{t_{23}}$	60	s	Timeout value of t_{23} in $\lfloor$*SA operation*$\rfloor[a_i, c]$

- Timeout: $t_{t_{23}}$

Transition t_{27}
Transition t_{27} is taken when the controller was not able to evaluate a feasible *vector inbound trombone* operation within the time period $t_{t_{27}}$, i.e. a_i is currently located too far away from $w_{S,R}$ for still being instructed a *vector inbound trombone* operation.

Parameter	Value	Unit	Description
$t_{t_{27}}$	200	s	Timeout value of t_{27} in $\lfloor$*SA operation*$\rfloor[a_i, c]$

- Timeout: $t_{t_{27}}$
- Guard: $I^{a_i} \neq I_9$

Transition t_{30}

- Condition: $z^{a_i} < |\dot{z}^{a_i}| \,/\, 60 \cdot \Delta t$

Transition t_{31}
The ⟨*go-around*⟩ state has no modelled actions associated and has also not a really important meaning in the model. This timeout transition is only modelled to visualize the flown go-around procedures in *AnyLogic*, which are used for validation.

Parameter	Value	Unit	Description
$t_{t_{31}}$	320	s	Timeout value of t_{31} in $\lfloor$*SA operation*$\rfloor[a_i, c]$

- Timeout: $t_{t_{31}}$

Transition t_{32}

- Condition: $v_{IAS}^{a_i} \leq v_{HST}$

Transition t_{34}
Transition t_{34} models the total time that each aircraft requires to leave the runway. This phase starts when the aircraft starts turning its nose gear when still being aligned with the runway centreline and ends when the aircraft has left the runway in total.

Parameter	Value	Unit	Description
$t_{t_{34}}$	10	s	Timeout value of t_{34} in $\lfloor$*SA operation*$\rfloor[a_i, c]$

- Timeout: $t_{t_{34}}$

Transition t_{35}
Transition t_{35} models the time period in which the TWR controller can still hand the aircraft over to GND when this is not yet done when the aircraft was still on the runway.

Parameter	Value	Unit	Description
$t_{t_{35}}$	40	s	Timeout value of t_{35} in $\lfloor$*SA operation*$\rfloor[a_i, c]$

- Timeout: $t_{t_{35}}$

Transition (remaining)
Many of the transitions in the ⌊*SA operation*⌋[a_i, c] statechart are triggered in states and transitions of other statecharts. The table below lists the states and transitions where the specific transitions in the ⌊*SA operation*⌋[a_i, c] statechart are triggered in. Each of these transitions do not contain any actions.

Transition	State or transition where the specific transition in ⌊*SA operation*⌋[a_i, c] is triggered in
t_1	⟨*holding entry instruction*⟩⌊*Contact with flight crew* a_i⌋[c]
t_2	⟨*holding exit instruction*⟩⌊*Contact with flight crew* a_i⌋[c]
t_3	t_6 of ⌊*Interception of fly-by waypoints*⌋[*MCP* a_i]
t_4	t_7 of ⌊*Interception of fly-by waypoints*⌋[*MCP* a_i]
t_5	⟨*vector outbound merge instruction*⟩⌊*Contact with flight crew* a_i⌋[c]
t_6	⟨*vector inbound merge instruction*⟩⌊*Contact with flight crew* a_i⌋[c]
t_7	t_1 and t_2 of ⌊*Interception of fly-by waypoints*⌋[*MCP* a_i]
t_8	t_1, t_2 and t_5 of ⌊*Interception of fly-by waypoints*⌋[*MCP* a_i]
t_9	⟨*vector outbound merge instruction*⟩⌊*Contact with flight crew* a_i⌋[c]
t_{10}	⟨*vector outbound merge instruction*⟩⌊*Contact with flight crew* a_i⌋[c]
t_{11}	⟨*vector outbound STAR instruction*⟩⌊*Contact with flight crew* a_i⌋[c]
t_{12}	⟨*vector inbound STAR instruction*⟩⌊*Contact with flight crew* a_i⌋[c]
t_{13}	⟨*vector outbound STAR instruction*⟩⌊*Contact with flight crew* a_i⌋[c]
t_{14}	⟨*vector outbound trombone instruction*⟩⌊*Contact with flight crew* a_i⌋[c]
t_{15}	⟨*vector inbound IF instruction*⟩⌊*Contact with flight crew* a_i⌋[c]
t_{16}	⟨*vector inbound IF instruction*⟩⌊*Contact with flight crew* a_i⌋[c]
t_{17}	⟨*vector outbound IF instruction*⟩⌊*Contact with flight crew* a_i⌋[c]
t_{18}	⟨*vector inbound IF instruction*⟩⌊*Contact with flight crew* a_i⌋[c]
t_{19}	⟨*vector outbound trombone instruction*⟩⌊*Contact with flight crew* a_i⌋[c]
t_{20}	⟨*vector outbound trombone instruction*⟩⌊*Contact with flight crew* a_i⌋[c]
t_{21}	t_3 and t_4 of ⌊*Interception of fly-by waypoints*⌋[*MCP* a_i]
t_{22}	⟨*vector outbound trombone instruction*⟩⌊*Contact with flight crew* a_i⌋[c]
t_{24}	⟨*vector inbound trombone instruction*⟩⌊*Contact with flight crew* a_i⌋[c]
t_{25}	t_1 and t_2 of ⌊*Interception of fly-by waypoints*⌋[*MCP* a_i]
t_{26}	⟨*holding entry instruction*⟩⌊*Contact with flight crew* a_i⌋[c]
t_{28}	t_1 of ⌊*Glide slope interception*⌋[*MCP* a_i]
t_{29}	⟨*go-around instruction*⟩⌊*Contact with flight crew* a_i⌋[c]
t_{33}	⟨*passed* w_{HST1}⟩ and ⟨*passed* w_{HST2}⟩ of ⌊*SA HST*⌋[*flight crew* a_i]

C.4.5.1.2 SA STAR progress

The ⌊*SA STAR progress*⌋[a_i, c] statechart (figure 79) models the situation awareness of the TNW/TNE/ARR controller agent about the specific position of a_i in the trombone segment of the STAR procedure.

No further clarification needed for the **⟨downwind segment⟩**, **⟨base segment⟩** and **⟨final segment⟩** states

⟨not nearing final waypoint⟩
Indicates that a_i is positioned on a leg within the *final segment* that is not connected to $w_{S,IAF}$.

⟨nearing final waypoint⟩
Indicates that a_i is positioned on a leg within the *final segment* that is connected to $w_{S,IAF}$. This state indicates therefore that a_i is about to complete its flown STAR procedure.

⟨passed final waypoint⟩

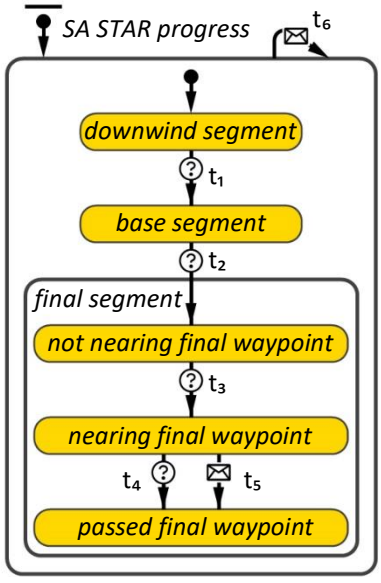


Figure 79 – ⌊SA STAR progress⌋[a_i, c]

Marks the completion of the flown STAR procedure. This state can either be achieved when a_i has passed $w_{S,\text{IAF}}$, or when a_i is no longer able to receive a *vector inbound STAR* instruction towards $w_{S,\text{IAF}}$.

Transition t_1 - t_4

All condition triggered:

Transition	Condition
t_1	$(\langle STAR\rangle\lfloor SA\ operation\rfloor[a_i, c] \wedge k^{a_i,c} > k^{a_i,c}_{I_4,S}) \vee (\langle vector\ STAR\rangle\lfloor SA\ operation\rfloor[a_i, c] \wedge k^{a_i,c} > k^{a_i,c}_{I_4,S} - 1)$
t_2	$k^{a_i,c} > k^{a_i,c}_{I_5,S}$
t_3	$k^{a_i,c} == K_S - 1$
t_4	$\langle STAR\rangle\lfloor SA\ operation\rfloor[a_i, c] \wedge d^{a_i,c}_{\text{A},w_{S,k}} \leq d^{a_i}_{\Delta t}$

Transition t_5

Triggered in the "Determine the waypoint number $k^{a_i,c}$ during *vector outbound STAR* operations" phase in event $\mathcal{E}^c_{\text{SA}}$ (appendix C.4.4.1). In this situation a_i is passing/has passed $w_{S,\text{IAF}}$ while flying a *vector outbound STAR* operation.

Transition t_6

Triggered in the entry actions of $\langle STAR\rangle\lfloor SA\ operation\rfloor[a_i, c]$

C.4.5.1.3 SA vector instruction history

The $\lfloor SA\ vector\ instruction\ history\rfloor[a_i, c]$ statechart (figure 80) models the memory of the TNW/TNE/ARR controller about the last vector operation that has been instructed to a_i. This memory is required to prevent the provision of too many vector instructions to the same aircraft in a relatively short time period. The statechart therefore imposes a time period in which the controller cannot instruct a new vector operation to a_i.

⟨available for vector instruction⟩

Indicates that the controller is free to instruct a_i a vector operation. This means that the last vector instruction for a_i has been instructed a sufficiently long period ago, according to the modelled memory of the controller.

⟨vector recently instructed⟩

Activated just after the controller has instructed a vector operation to a_i. When this state is active the controller temporarily cannot instruct a_i a new vector operation.

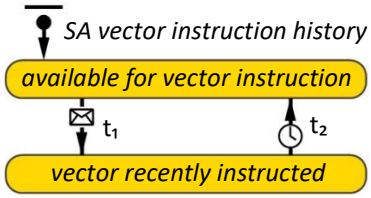


Figure 80 – $\lfloor SA\ vector\ instruction\ history\rfloor[a_i, c]$

Transition t_1

Triggered in ⟨*vector outbound STAR instruction*⟩, ⟨*vector inbound STAR instruction*⟩, ⟨*vector outbound merge instruction*⟩, ⟨*vector inbound merge instruction*⟩, ⟨*vector outbound IF instruction*⟩, ⟨*vector inbound IF instruction*⟩, ⟨*vector inbound trombone instruction*⟩ of $\lfloor Contact\ with\ flight\ crew\ a_i\rfloor[c]$

Transition t_2

Defines the time period in which the controller cannot instruct a new vector operation to a_i. The duration of this timeout period is chosen such that it prevents the controller from providing too frequent vector instructions to the same aircraft. It is assumed that the value of this timeout triggered transition represents a reasonable and sufficient long time period after which the controller may again intervene if necessary.

Parameter	Value	Unit	Description
t_{AV}	30	s	Timeout value of t_2 in $\lfloor SA\ vector\ instruction\ history\rfloor[a_i, c]$

- Timeout: t_{AV}

C.4.5.1.4 SA waypoint update history

The $\lfloor SA\ waypoint\ update\ history\rfloor[a_i, c]$ statechart (figure 81) models the memory of the TNW/TNE/ARR controller about the last time that it has updated its situation awareness with respect to the current $w_{S,k}$ of a_i. The statechart imposes a time period in which the controller cannot update $k^{a_i,c}$. This time period makes sure that aircraft that are operating the *vector outbound STAR* will not be instructed a *vector inbound STAR* that is directed towards an unrealistic waypoint.

⟨available for waypoint update⟩

Indicates that the controller is free to update the waypoint number $k^{a_i,c}$ when a_i is operating the *vector outbound STAR* instruction.

⟨waypoint recently updated⟩
Activated just after the controller has updated the waypoint number $k^{a_i,c}$. When this state is active the controller temporarily cannot update $k^{a_i,c}$.

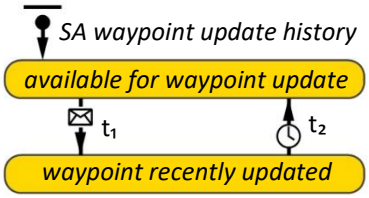


Figure 81 – ⌊SA waypoint update history⌋[a_i, c]

Transition t_1
Triggered in the "Determine the waypoint number $k^{a_i,c}$ during *vector outbound STAR* operations" phase in event $\mathcal{E}^c_{\text{SA}}$ (appendix C.4.4.1).

Transition t_2
Defines the time period in which the controller cannot update $k^{a_i,c}$. The duration of this timeout period (i.e. t_{AW}) is chosen such that it prevents too frequent and therefore unrealistic increments of $k^{a_i,c}$. The transition makes sure that the controller keeps on referencing a_i to the same waypoint for a given period of time before the controller can consider another waypoint to be more suitable for a *vector inbound STAR* instruction.

Parameter	Value	Unit	Description
t_{AW}	40	s	Timeout value of t_2 of ⌊*SA waypoint update history*⌋[a_i, c]

- Timeout: t_{AW}

C.4.5.1.5 SA STAR speed

The ⌊*SA STAR speed*⌋[a_i, c] statechart (figure 82) models the situation awareness of the TNW/TNE/ARR controller about the airspeed of a_i. The statechart compares the observed airspeed of a_i with the airspeed that a_i is desired to fly when considering the current position of a_i along the STAR procedure or within the arrival phase in general. This observation is used to evaluate if the current airspeed of a_i is in proportion with the prescribed speed constraints of the STAR procedure.

⟨correct speed⟩
Indicates that the observed airspeed of a_i is in proportion with the prescribed speed constraints of the STAR.

⟨too fast⟩
Indicates that the observed airspeed of a_i is higher than the prescribed speed constraints of the STAR.

⟨too slow⟩
Indicates that the observed airspeed of a_i is lower than the prescribed speed constraints of the STAR.

Transition t_1 - t_4
All condition triggered:

Transition	Condition
t_1	$v_{\text{IAS}}^{a_i,c} > v_{\text{IAS},w_{S,k}}^{a_i,c} + v_{\text{IAS},\varepsilon}$
t_2	$v_{\text{IAS}}^{a_i,c} \leq v_{\text{IAS},w_{S,k}}^{a_i,c} + v_{\text{IAS},\varepsilon}$
t_3	$v_{\text{IAS}}^{a_i,c} < v_{\text{IAS},w_{S,k}}^{a_i,c} - v_{\text{IAS},\varepsilon}$
t_4	$v_{\text{IAS}}^{a_i,c} \geq v_{\text{IAS},w_{S,k}}^{a_i,c} - v_{\text{IAS},\varepsilon}$

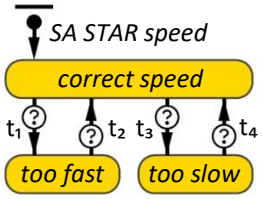


Figure 82 – ⌊SA STAR speed⌋[a_i, c]

C.4.5.1.6 SA STAR altitude

The ⌊*SA STAR altitude*⌋[a_i, c] statechart (figure 83) models the situation awareness of the TNW/TNE/ARR controller about the altitude of a_i. The statechart compares the observed altitude of a_i with the altitude that a_i is desired to fly when considering the current position of a_i along the STAR procedure or within the arrival phase in general. This observation is used to evaluate if the current altitude of a_i is in proportion with the prescribed altitude constraints of the STAR procedure.

⟨correct altitude⟩
Indicates that the observed altitude of a_i is in proportion with the prescribed altitude constraints of the STAR.

⟨too high⟩
Indicates that the observed altitude of a_i is higher than the prescribed altitude constraints of the STAR.

⟨too low⟩
Indicates that the observed altitude of a_i is lower than the prescribed altitude constraints of the STAR.

Transition t_1 - t_4
All condition triggered:

Transition	Condition
t_1	$z^{a_i,c} > z^{a_i,c}_{w_{S,k}} + z_\varepsilon$
t_2	$z^{a_i,c} \leq z^{a_i,c}_{w_{S,k}} + z_\varepsilon$
t_3	$z^{a_i,c} < z^{a_i,c}_{w_{S,k}} - z_\varepsilon$
t_4	$z^{a_i,c} \geq z^{a_i,c}_{w_{S,k}} - z_\varepsilon$

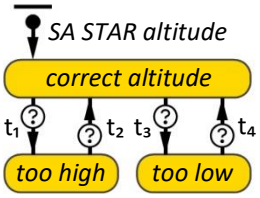


Figure 83 – ⌊SA STAR altitude⌋[a_i, c]

C.4.5.1.7 SA STAR speed instruction requirement

The ⌊*SA STAR speed instruction requirement*⌋[a_i, c] statechart (figure 84) models the situation awareness of the TNW/TNE/ARR controller about the state of a_i concerning the requirement of a *STAR speed and/or altitude* instruction. The statechart compares both the actual observed speed and instructed speed of a_i with the speed that a_i is desired to fly when considering the current position of a_i along the STAR procedure or within the arrival phase in general. By evaluating this comparison the controller can decide if a speed instruction is required.

⟨no instruction required⟩
Indicates that a *SA STAR speed and/or altitude* instruction is not considered to be required based on the observed airspeed data of a_i. It indicates that the current airspeed of a_i is in proportion with the prescribed speed constraints of the STAR procedure, or that a_i is (already) operating the desired and instructed airspeed.

⟨instruction required⟩
Indicates that both the actual- and instructed airspeed of a_i do not match with the airspeed that a_i is desired to fly when considering the current position of a_i along the STAR procedure or during arrival in general.

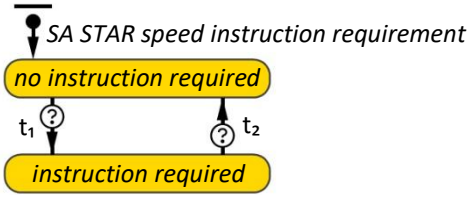


Figure 84 – ⌊SA STAR speed instruction requirement⌋[a_i, c]

Transition t_1
- Condition: (⟨*too fast*⟩⌊*SA STAR speed*⌋[a_i, c] ∧ $v^{a_i,c}_{\text{IAS,I}} > v^{a_i,c}_{\text{IAS},w_{S,k}}$) ∨ (⟨*too slow*⟩⌊*SA STAR speed*⌋[a_i, c] ∧ $v^{a_i,c}_{\text{IAS,I}} < v^{a_i,c}_{\text{IAS},w_{S,k}}$)

Transition t_2
Note that t_2 will also be taken if a_i is currently still approaching the instructed airspeed despite not being yet operating the desired and instructed airspeed.
- Condition: (⟨*too fast*⟩⌊*SA STAR speed*⌋[a_i, c] ∧ $v^{a_i,c}_{\text{IAS,I}} == v^{a_i,c}_{\text{IAS},w_{S,k}}$) ∨ (⟨*too slow*⟩⌊*SA STAR speed*⌋[a_i, c] ∧ $v^{a_i,c}_{\text{IAS,I}} == v^{a_i,c}_{\text{IAS},w_{S,k}}$) ∨ ⟨*correct speed*⟩⌊*SA STAR speed*⌋[a_i, c]

C.4.5.1.8 SA STAR altitude instruction requirement

The ⌊*SA STAR altitude instruction requirement*⌋[a_i, c] statechart (figure 85) models the situation awareness of the TNW/TNE/ARR controller about the state of a_i concerning the requirement of a *STAR speed and/or altitude* instruction. The statechart compares both the actual observed altitude and instructed altitude of a_i with the altitude that a_i is desired to fly when considering the current position of a_i along the STAR procedure or within the arrival phase in general. By evaluating this comparison the controller can decide if an altitude instruction is required.

⟨no instruction required⟩
Indicates that a *SA STAR speed and/or altitude* instruction is not considered to be required based on the observed altitude data of a_i. It indicates that the current altitude of a_i is in proportion with the prescribed altitude constraints of the STAR procedure, or that a_i is (already) operating the desired and instructed altitude.

⟨instruction required⟩
Indicates that both the actual- and instructed altitude of a_i do not match with the altitude that a_i is desired to fly when considering the current position of a_i along the STAR procedure or during arrival in general.

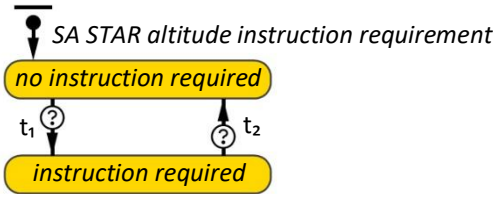


Figure 85 – ⌊SA STAR altitude instruction requirement⌋[a_i, c]

Transition t_1

- Condition: (⟨*too high*⟩⌊*SA STAR altitude*⌋[a_i, c] ∧ $z_{|}^{a_i,c} > z_{w_{S,k}}^{a_i,c}$) ∨ (⟨*too low*⟩⌊*SA STAR altitude*⌋[a_i, c] ∧ $z_{|}^{a_i,c} < z_{w_{S,k}}^{a_i,c}$)

Transition t_2
Note that t_2 will also be taken if a_i is currently still approaching the instructed altitude despite not being yet operating the desired and instructed altitude.

- Condition: (⟨*too high*⟩⌊*SA STAR altitude*⌋[a_i, c] ∧ $z_{|}^{a_i,c} == z_{w_{S,k}}^{a_i,c}$) ∨ (⟨*too low*⟩⌊*SA STAR altitude*⌋[a_i, c] ∧ $z_{|}^{a_i,c} == z_{w_{S,k}}^{a_i,c}$) ∨ ⟨*correct altitude*⟩⌊*SA STAR altitude*⌋[a_i, c]

C.4.5.1.9 SA vector outbound STAR requirement

The ⌊*SA vector outbound STAR requirement*⌋[a_i, c] statechart (figure 86) models the situation awareness of the TNW/TNE/ARR controller about the state of a_i concerning the requirement of a *vector outbound STAR* instruction.

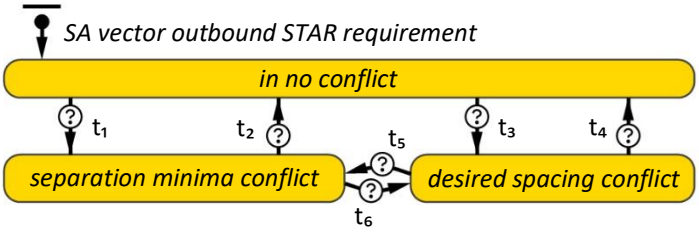


Figure 86 – ⌊SA vector outbound STAR requirement⌋[a_i, c]

Transitions t_1 - t_6
All condition triggered:

Transition	Condition	Transition	Condition
t_1	$b_{R,I_{2,1}}^{a_i,c}$ == true	t_4	$b_{R,I_{2,1}}^{a_i,c}$ == false ∧ $b_{R,I_{2,2}}^{a_i,c}$ == false
t_2	$b_{R,I_{2,1}}^{a_i,c}$ == false ∧ $b_{R,I_{2,2}}^{a_i,c}$ == false	t_5	$b_{R,I_{2,1}}^{a_i,c}$ == true
t_3	$b_{R,I_{2,1}}^{a_i,c}$ == false ∧ $b_{R,I_{2,2}}^{a_i,c}$ == true	t_6	$b_{R,I_{2,1}}^{a_i,c}$ == false ∧ $b_{R,I_{2,2}}^{a_i,c}$ == true

C.4.5.1.10 SA vector inbound STAR

The ⌊*SA vector inbound STAR*⌋[a_i, c] statechart (figure 87) models the situation awareness of the TNW/TNE/ARR controller about the state of a_i concerning the *vector inbound STAR* instruction.

Transitions t₁, t₂, t₄

All condition triggered:

Transition	Condition
t₁	$b^{a_i,c}_{A,I_3}$ == true
t₂	$b^{a_i,c}_{A,I_3}$ == false
t₄	⟨*STAR*⟩⌊*SA operation*⌋[a_i, c] ∨ ⟨*vector outbound STAR*⟩⌊*SA operation*⌋[a_i, c] ∨ ⟨*vector outbound merge*⟩ ⌊*SA operation*⌋[a_i, c]

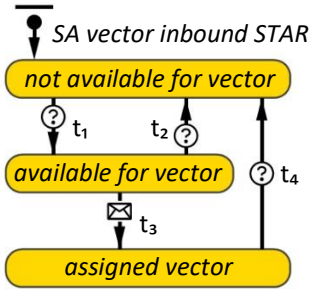


Figure 87 – ⌊SA vector inbound STAR⌋[a_i, c]

Transition t₃

Triggered in ⟨*vector inbound STAR instruction*⟩⌊*Contact with flight crew* a_i⌋[c]

C.4.5.1.11 SA vector outbound merge requirement

The ⌊*SA vector outbound merge requirement*⌋[a_i, c] statechart (figure 88) models the situation awareness of the ARR controller about the state of a_i concerning the requirement of a *vector outbound merge* instruction.

Transition t₁

- Condition: $b^{a_i,c}_{R,I_4}$== true

Transition t₂

- Condition: $b^{a_i,c}_{R,I_4}$ == false

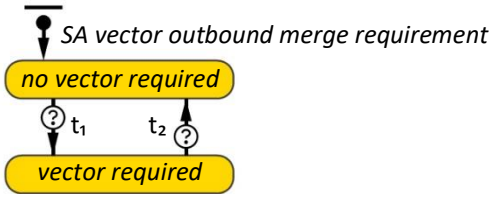


Figure 88 – ⌊SA vector outbound merge requirement⌋[a_i, c]

C.4.5.1.12 SA vector outbound STAR history

The ⌊*SA vector outbound STAR history*⌋[a_i, c] statechart (figure 89) is used in the model to incorporate a time period in which the ARR controller is not able to instruct the *vector outbound merge* to a_i. This statechart has only a function when a_i is flying the *vector outbound STAR* operation near the *base segment* of the STAR procedure. At a given point in time the *vector outbound merge* instruction may become required for a_i (⌊*SA vector outbound merge requirement*⌋[a_i, c]) while a_i is still operating the *vector outbound STAR*. The ⌊*SA vector outbound STAR history*⌋[a_i, c] statechart is meant to prevent the ARR controller from instructing too quickly the *vector outbound merge* operation while the *vector outbound STAR* has been instructed just recently. This modelled delay will result in a relatively longer execution of the *vector outbound STAR* instruction, which in turn will improve the resilient buffer capacities near the *base segment* of the STAR procedure. Only if a_i has been operating the *vector outbound STAR* for a sufficient period of time a_i becomes available again for the *vector outbound merge* instruction. Figure 90 visualizes the relative effects of different values for the timeout triggered transition t₂ within the ⌊*SA vector outbound STAR history*⌋ statechart. The timeout value of this transition defines and shapes the buffer areas around the *base segment* of the STAR procedure.

⟨available for vector outbound merge⟩

Indicates that a_i is available to receive a *vector outbound merge* instruction when considering the duration of the flown *vector outbound STAR* operation.

⟨not available for vector outbound merge⟩

Activated just after the controller has instructed a_i a *vector outbound STAR* operation. When this state is active the controller temporarily cannot instruct a_i a *vector outbound merge* operation.

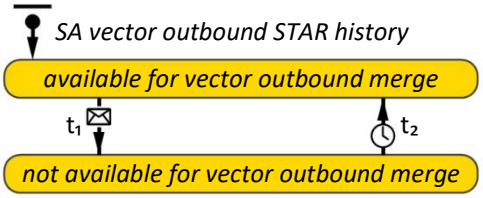


Figure 89 – ⌊SA vector outbound STAR history⌋[a_i, c]

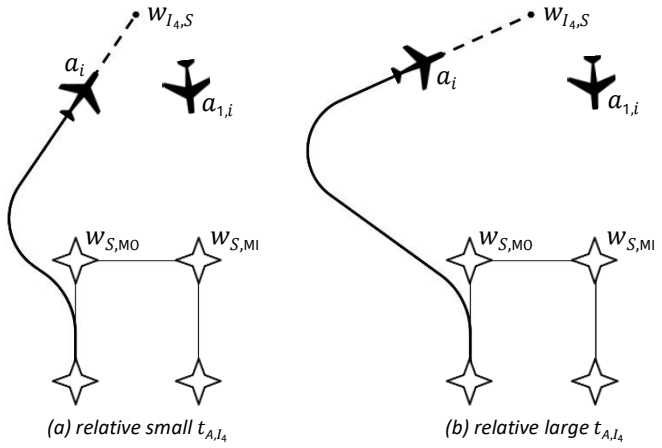


Figure 90 – Relative impact of different values for t_{A,I_4} on the buffer capacities of the STAR procedure around the base segment

Transition t_1

Triggered in ⟨*vector outbound STAR instruction*⟩⌊*Contact with flight crew* a_i⌋$[c]$

Transition t_2

Transition t_2 defines the time period in which the controller temporarily cannot instruct a_i a *vector outbound merge* operation, because a *vector outbound STAR* operation has been instructed just recently. The duration of this timeout period is chosen such that it allows a_i to operate the *vector outbound STAR* for a sufficient period of time. This timeout value thus prevents a too quickly instructed *vector outbound merge* operation while a_i has been instructed the *vector outbound STAR* operation just recently. It is assumed that the value of this timeout triggered transition represents a reasonable and sufficient long time period after which the controller may instruct the *vector outbound merge* operation as a continuation of the *vector outbound STAR* operation. The relative impact of different values for t_{A,I_4} on the resilient buffer capacities of the STAR procedure around the *base segment* are visualized in figure 90.

Parameter	Value	Unit	Description
t_{A,I_4}	90	s	Timeout value of t_2 in ⌊*SA vector outbound STAR history*⌋$[a_i, c]$

- Timeout: t_{A,I_4}

C.4.5.1.13 SA vector inbound merge

The ⌊*SA vector inbound merge*⌋$[a_i, c]$ statechart (figure 91) models the situation awareness of the ARR controller about the state of a_i concerning the *vector inbound merge* instruction.

Transitions t_1, t_2, t_4

All condition triggered:

Transition	Condition
t_1	$b_{A,I_5}^{a_i,c}$ == true
t_2	$b_{A,I_5}^{a_i,c}$ == false
t_4	$k^{a_i,c} > k_{I_5,S}^{a_i,c}$

Transition t_3

Triggered in ⟨*vector inbound merge instruction*⟩⌊*Contact with flight crew* a_i⌋$[c]$

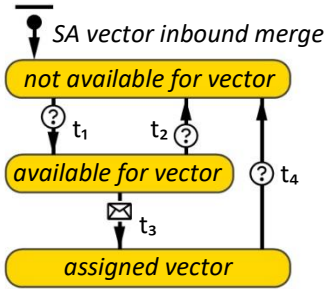


Figure 91 – ⌊SA vector inbound merge⌋$[a_i, c]$

C.4.5.1.14 SA vector outbound IF requirement

The ⌊*SA vector outbound IF requirement*⌋$[a_i, c]$ statechart (figure 92) models the situation awareness of the ARR controller about the state of a_i concerning the requirement of a *vector outbound IF* instruction.

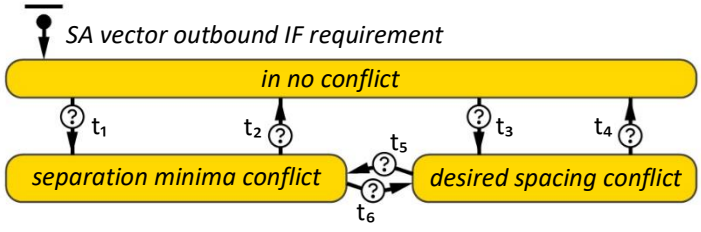


Figure 92 – ⌊SA vector outbound IF requirement⌋$[a_i, c]$

Transitions t_1 - t_6

All condition triggered:

Transition	Condition	Transition	Condition
t_1	$b_{R,I_{6,1}}^{a_i,c}$ == true	t_4	$b_{R,I_{6,1}}^{a_i,c}$ == false ∧ $b_{R,I_{6,2}}^{a_i,c}$ == false
t_2	$b_{R,I_{6,1}}^{a_i,c}$ == false ∧ $b_{R,I_{6,2}}^{a_i,c}$ == false	t_5	$b_{R,I_{6,1}}^{a_i,c}$ == true
t_3	$b_{R,I_{6,1}}^{a_i,c}$ == false ∧ $b_{R,I_{6,2}}^{a_i,c}$ == true	t_6	$b_{R,I_{6,1}}^{a_i,c}$ == false ∧ $b_{R,I_{6,2}}^{a_i,c}$ == true

C.4.5.1.15 SA vector inbound IF

The ⌊*SA vector inbound IF*⌋[a_i, c] statechart (figure 93) models the situation awareness of the ARR controller about the state of a_i concerning the *vector inbound IF* instruction.

Transitions t_1, t_2, t_4, t_5

All condition triggered:

Transition	Condition
t_1	$b_{A,I_7}^{a_i,c}$ == true
t_2	$b_{A,I_7}^{a_i,c}$ == false
t_4	⟨*vector outbound IF*⟩⌊*SA operation*⌋[a_i, c] ∨ ⟨*vector outbound trombone phase 1*⟩⌊*SA operation*⌋[a_i, c]
t_5	⟨*intermediate approach*⟩⌊*SA operation*⌋[a_i, c]

Transition t_3

Triggered in ⟨*vector inbound IF instruction*⟩⌊*Contact with flight crew* a_i⌋[c]

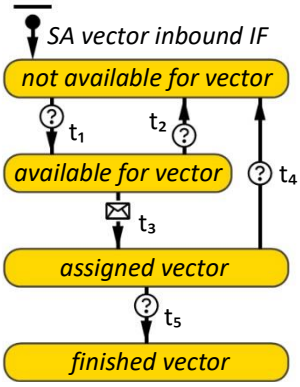


Figure 93 – ⌊SA vector inbound IF⌋[a_i, c]

C.4.5.1.16 SA vector outbound trombone requirement

The ⌊*SA vector outbound trombone requirement*⌋[a_i, c] statechart (figure 94) models the situation awareness of the ARR controller about the state of a_i concerning the requirement of a *vector outbound trombone* instruction.

Transition t_1

- Condition: $b_{R,I_8}^{a_i,c}$ == true

Transition t_2

- Condition: $b_{R,I_8}^{a_i,c}$ == false

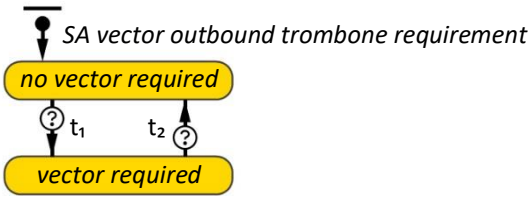


Figure 94 – ⌊SA vector outbound trombone requirement⌋[a_i, c]

C.4.5.1.17 SA vector inbound trombone

The ⌊*SA vector inbound trombone*⌋[a_i, c] statechart (figure 95) models the situation awareness of the ARR controller about the state of a_i concerning the *vector inbound trombone* instruction.

Transitions t_1, t_2, t_4

All condition triggered:

Transition	Condition
t_1	$b_{A,I_9}^{a_i,c}$ == true
t_2	$b_{A,I_9}^{a_i,c}$ == false
t_4	⟨*STAR*⟩⌊*SA operation*⌋[a_i, c]

Transition t_3

Triggered in ⟨*vector inbound trombone instruction*⟩ ⌊*Contact with flight crew* a_i⌋[c]

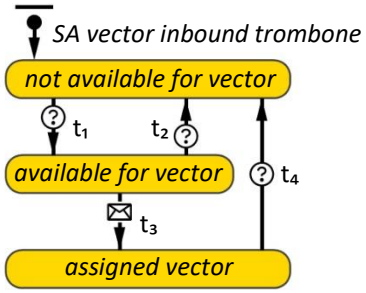


Figure 95 – ⌊SA vector inbound trombone⌋[a_i, c]

C.4.5.1.18 SA handover to ARR

The ⌊*SA handover to ARR*⌋[a_i, c] statechart (figure 96) models the situation awareness of the TNW/TNE controller about the state of a_i concerning the *handover to ARR* instruction.

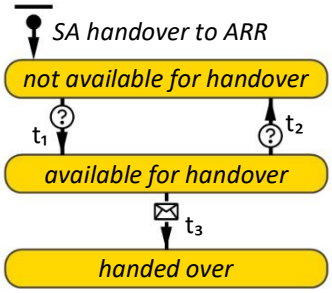


Figure 96 – ⌊SA handover to ARR⌋[a_i, c]

Transition t₁

- Condition: $b^{a_i,c}_{A,I_{10}}$ == true

Transition t₂

- Condition: $b^{a_i,c}_{A,I_{10}}$ == false

Transition t₃

Triggered in ⟨*handover to ARR controller instruction*⟩⌊*Contact with flight crew* a_i⌋[c]

C.4.5.1.19 SA handover to TWR

The ⌊*SA handover to TWR*⌋[a_i, c] statechart (figure 97) models the situation awareness of the ARR controller about the state of a_i concerning the *handover to TWR* instruction.

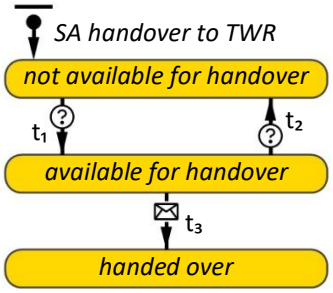


Figure 97 – ⌊SA handover to TWR⌋[a_i, c]

Transition t₁

- Condition: $b^{a_i,c}_{A,I_{11}}$ == true

Transition t₂

- Condition: $b^{a_i,c}_{A,I_{11}}$ == false

Transition t₃

Triggered in ⟨*handover to TWR controller instruction*⟩⌊*Contact with flight crew* a_i⌋[c]

C.4.5.1.20 SA handover to GND

The ⌊*SA handover to GND*⌋[a_i, c] statechart (figure 98) models the situation awareness of the TWR controller about the state of a_i concerning the *handover to GND* instruction.

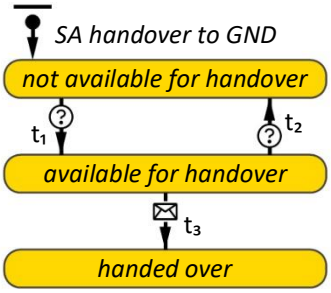


Figure 98 – ⌊SA handover to GND⌋[a_i, c]

Transition t₁

- Condition: $b^{a_i,c}_{A,I_{12}}$ == true

Transition t₂

- Condition: $b^{a_i,c}_{A,I_{12}}$ == false

Transition t₃

Triggered in ⟨*handover to GND controller instruction*⟩⌊*Contact with flight crew* a_i⌋[c]

C.4.5.1.21 SA landing clearance

The ⌊*SA landing clearance*⌋[a_i, c] statechart (figure 99) models the situation awareness of the TWR controller about the state of a_i concerning the *landing clearance* instruction.

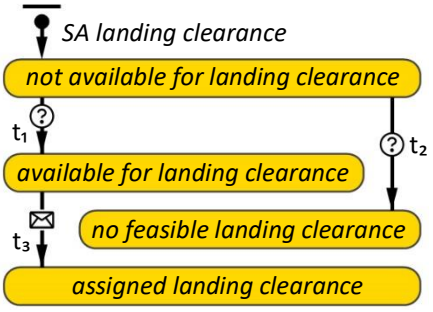


Figure 99 – ⌊SA landing clearance⌋[a_i, c]

Transition t₁

- Condition: $b^{a_i,c}_{A,I_{13}}$ == true

Transition t₂

- Condition: $b^{a_i,c}_{R,I_{14}}$ == true

Transition t₃

Triggered in ⟨*landing clearance instruction*⟩⌊*Contact with flight crew* a_i⌋[c]

C.4.5.1.22 SA holding requirement

The ⌊*SA holding requirement*⌋[a_i, c] statechart (figure 100) models the situation awareness of the TNW/TNE controller about the state of a_i concerning the requirement of a *holding entry* instruction.

Transition t₁

- Condition: $b_{R,I_{15}}^{a_i,c}$ == true

Transition t₂

- Condition: $b_{R,I_{15}}^{a_i,c}$ == false

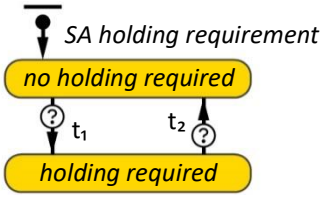


Figure 100 – ⌊SA holding requirement⌋[a_i, c]

C.4.5.1.23 SA holding entry

The ⌊*SA holding entry*⌋[a_i, c] statechart (figure 101) models the situation awareness of the TNW/TNE controller about the state of a_i concerning the *holding entry* instruction.

Transition t₁

- Condition: $b_{A,I_{15}}^{a_i,c}$ == true

Transition t₂

- Condition: $b_{A,I_{15}}^{a_i,c}$ == false

Transition t₃

Triggered in ⟨*holding entry instruction*⟩⌊*Contact with flight crew* a_i⌋[c]

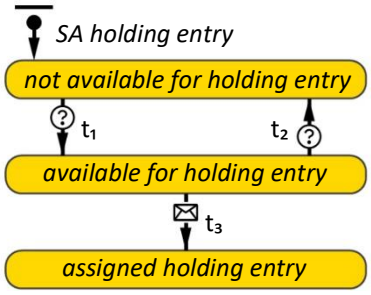


Figure 101 - ⌊SA holding entry⌋[a_i, c]

C.4.5.1.24 SA holding exit

The ⌊*SA holding exit*⌋[a_i, c] statechart (figure 102) models the situation awareness of the TNW/TNE controller about the state of a_i concerning the *holding exit* instruction.

Transition t₁

- Condition: $b_{A,I_{16}}^{a_i,c}$ == true

Transition t₂

- Condition: $b_{A,I_{16}}^{a_i,c}$ == false

Transition t₃

Triggered in ⟨*holding exit instruction*⟩⌊*Contact with flight crew* a_i⌋[c]

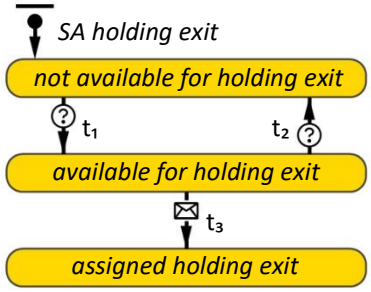


Figure 102 - ⌊SA holding exit⌋[a_i, c]

C.4.5.1.25 SA holding altitude requirement

The ⌊*SA holding altitude requirement*⌋[a_i, c] statechart (figure 103) models the situation awareness of the TNW/TNE controller about the state of a_i concerning the requirement of a *holding altitude* instruction.

Transition t₁

- Condition: $b_{R,I_{17}}^{a_i,c}$ == true

Transition t₂

- Condition: $b_{R,I_{17}}^{a_i,c}$ == false

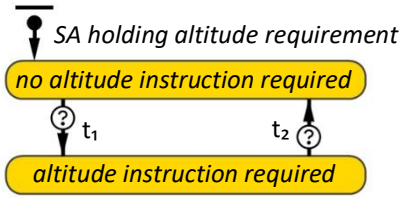


Figure 103 – ⌊SA holding altitude requirement⌋[a_i, c]

C.4.5.2 Workflow

The [*Workflow*][c] statechart (figure 104) models the way in which controller c ($c \in \mathbb{C}$) continuously identifies, schedules and executes observed tasks.

⟨task identification⟩

- State entry actions: ($\forall I \in \mathbb{I}$)

Variable	State space	Unit	Description
$\breve{a}$	$\mathbb{A}^c$	-	Iterator, i.e. specific aircraft in $\mathbb{A}^c$

1: $a_I^c = \emptyset$
2: **for** ($\forall \breve{a} \in \mathbb{A}^c$) **do** //see table 8 for the specific controller(s) c where this identification process belongs to
3: **if** (states of a_i that should be active to identify instruction I, see table 37) **then**
4: $\{I\} \cup \mathbb{I}^c$
5: $a_I^c = \breve{a}$
6: **break**
7: **end if**
8: **end for**

I ==	States of a_i that should be active to identify I
I_1	(⟨*STAR*⟩⌊*SA operation*⌋$[a_i, c]$ ∨ ⟨*vector STAR*⟩⌊*SA operation*⌋$[a_i, c]$) ∧ (⟨*instruction required*⟩⌊*SA STAR speed instruction requirement*⌋$[a_i, c]$ ∨ ⟨*instruction required*⟩⌊*SA STAR altitude instruction requirement*⌋$[a_i, c]$)
$I_{2,1}$	⟨*separation minima conflict*⟩⌊*SA vector outbound STAR requirement*⌋$[a_i, c]$ ∧ ⟨*available for vector instruction*⟩⌊*SA vector instruction history*⌋$[a_i, c]$
$I_{2,2}$	⟨*desired spacing conflict*⟩⌊*SA vector outbound STAR requirement*⌋$[a_i, c]$ ∧ ⟨*available for vector instruction*⟩⌊*SA vector instruction history*⌋$[a_i, c]$
I_3	⟨*available for vector*⟩⌊*SA vector inbound STAR*⌋$[a_i, c]$ ∧ ⟨*available for vector instruction*⟩⌊*SA vector instruction history*⌋$[a_i, c]$
I_4	⟨*vector required*⟩⌊*SA vector outbound merge requirement*⌋$[a_i, c]$ ∧ ⟨*available for vector outbound merge*⟩⌊*SA vector outbound STAR history*⌋$[a_i, c]$ ∧ ⟨*available for vector instruction*⟩⌊*SA vector instruction history*⌋$[a_i, c]$
I_5	⟨*available for vector*⟩⌊*SA vector inbound merge*⌋$[a_i, c]$ ∧ ⟨*available for vector instruction*⟩⌊*SA vector instruction history*⌋$[a_i, c]$
$I_{6,1}$	⟨*separation minima conflict*⟩⌊*SA vector outbound IF requirement*⌋$[a_i, c]$ ∧ ⟨*available for vector instruction*⟩⌊*SA vector instruction history*⌋$[a_i, c]$
$I_{6,2}$	⟨*desired spacing conflict*⟩⌊*SA vector outbound IF requirement*⌋$[a_i, c]$ ∧ ⟨*available for vector instruction*⟩⌊*SA vector instruction history*⌋$[a_i, c]$
I_7	⟨*available for vector*⟩⌊*SA vector inbound IF*⌋$[a_i, c]$ ∧ ⟨*available for vector instruction*⟩⌊*SA vector instruction history*⌋$[a_i, c]$
I_8	⟨*vector required*⟩⌊*SA vector outbound trombone requirement*⌋$[a_i, c]$
I_9	⟨*available for vector*⟩⌊*SA vector inbound trombone*⌋$[a_i, c]$
I_{10}	⟨*available for handover*⟩⌊*SA handover to ARR*⌋$[a_i, c]$
I_{11}	⟨*available for handover*⟩⌊*SA handover to TWR*⌋$[a_i, c]$
I_{12}	⟨*available for handover*⟩⌊*SA handover to GND*⌋$[a_i, c]$
I_{13}	⟨*available for landing clearance*⟩⌊*SA landing clearance*⌋$[a_i, c]$
I_{14}	⟨*no feasible landing clearance*⟩⌊*SA landing clearance*⌋$[a_i, c]$ ∧ ¬⟨*go-around*⟩⌊*SA operation*⌋$[a_i, c]$
I_{15}	⟨*available for holding entry*⟩⌊*SA holding entry*⌋$[a_i, c]$ ∧ ⟨*holding required*⟩⌊*SA holding requirement*⌋$[a_i, c]$
I_{16}	⟨*available for holding exit*⟩⌊*SA holding exit*⌋$[a_i, c]$
I_{17}	⟨*altitude instruction required*⟩⌊*SA holding altitude requirement*⌋$[a_i, c]$

*here $\breve{a}$ is iterated over $\mathbb{A}_{I_8}$ instead of $\mathbb{A}^c$

Table 37 – The set of states corresponding to a_i that should be active in order to let controller c identify instruction I

⟨task scheduling⟩

- State entry actions:

1: **if** ($I_{2,1} \in \mathbb{I}^c$) **then**
2: $I^c = I_{2,1}$
3: **else if** ($I_{2,2} \in \mathbb{I}^c$) **then**
4: $I^c = I_{2,2}$
5: *continue likewise*
6: **else if** ($I_1 \in \mathbb{I}^c$) **then**
7: $I^c = I_1$

8: **end if**

highest priority - $I_{2,1}$ - $I_{2,2}$ - I_8 - $I_{6,1}$ - $I_{6,2}$ - I_{15} - I_{16} - I_4 - I_3 - I_7 - I_5 - I_{14} - I_{13} - I_9 - I_{10} - I_{11} - I_{12} - I_{17} - I_1 - **lowest priority**

Table 38 – Task/instruction priorities used in the task scheduling process

⟨task execution⟩

- State entry actions:

1: $\{t_A\} \cup \mathbb{T}_I^c$

- State exit actions:

1: $I^c = \emptyset$
2: $\mathbb{I}^c = \emptyset$

Transition t_1

- Timeout: t_{IDE}^c
- Action:

1: $t_{IDE}^c = \mathcal{L}(\mu_{t_{IDE}}^c, \sigma_{t_{IDE}}^c, l_{t_{IDE}}^c)$

Transition t_2

Transition t_2 is the default branch transition of ⟨b_1⟩

Transition t_3

- Condition: $\mathbb{I}^c \neq \emptyset$

Transition t_4

- Timeout: t_{SHD}^c
- Action:

1: $t_{SHD}^c = \mathcal{L}(\mu_{t_{SHD}}^c, \sigma_{t_{SHD}}^c, l_{t_{SHD}}^c)$

Transition t_5

- Condition: ⟨*no contact*⟩⌊*Contact with flight crew* a_i⌋[c]

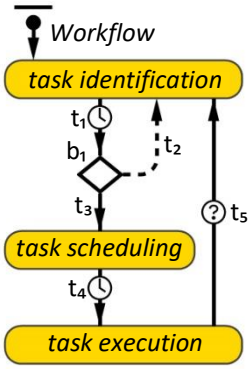


Figure 104 – ⌊Workflow⌋[c]

C.4.5.3 Contact with flight crew a_i

The ⌊*Contact with flight crew* a_i⌋[c] statechart (figure 105) models the communication between c ($c \in \mathbb{C}$) and the flight crew a_i agent ($a_i \in \mathbb{A}^c$) from the controller's perspective.

⟨no contact⟩

Indicates that there is currently no active communication between c and the flight crew a_i.

⟨in contact⟩

Indicates that c has successfully made contact with the flight crew a_i. The controller is within the ⟨*in contact*⟩ state either sending I^c or receiving pilot read-back of I^c.

⟨sending instruction⟩

Represents the time period in which c communicates the scheduled I^c ($I^c \in \mathbb{I}^c$) to the flight crew a_i.

⟨receiving pilot read-back⟩

Represents the time period in which the controller is receiving pilot read-back after having sent I^c.

All the remaining descriptions of the ⌊*Contact with flight crew* a_i⌋[c] states formalise the specific contents of the instructions that may be sent by c, where $c == c^{a_i}$. Note that the variables that are assigned in these entry/exit actions all belong to a_c^c, i.e. $a_i = a_c^c$. The specific aircraft agent where the identified and scheduled task should be instructed to has been assigned in the ⟨*task identification*⟩[*Workflow*][c] state. All the entry actions of the remaining states will initially contain the following two statements when a_c^c has been assigned:

1: Trigger t_1 of ⌊*Contact with* c^{a_i}⌋⌊*flight crew* a_i⌋
2: $I^{a_i} = I^c$

3: $I^{a_i,c} = I^c$

⟨STAR speed and/or altitude instruction⟩

- State entry actions:

1: $a_C^c = a_{I_1}^c$
2: $z_I^{a_i} = z_{w_{S,k}}^{a_i,c}$
3: $z_I^{a_i,c} = z_{w_{S,k}}^{a_i,c}$
4: $v_{IAS,I}^{a_i} = v_{IAS,w_{S,k}}^{a_i,c}$
5: $v_{IAS,I}^{a_i,c} = v_{IAS,w_{S,k}}^{a_i,c}$

⟨vector outbound STAR instruction⟩

- State entry actions:

1: **if** $(I^c == I_{2,1})$ **then**
2: $a_C^c = a_{I_{2,1}}^c$
3: **else if** $(I^c == I_{2,2})$ **then**
4: $a_C^c = a_{I_{2,2}}^c$
5: **end if**
6: $z_I^{a_i} = z_{w_{S,k}}^{a_i,c}$
7: $z_I^{a_i,c} = z_{w_{S,k}}^{a_i,c}$
8: $v_{IAS,I}^{a_i} = v_{IAS,w_{S,k}}^{a_i,c}$
9: $v_{IAS,I}^{a_i,c} = v_{IAS,w_{S,k}}^{a_i,c}$
10: $\psi_{A,I}^{a_i} = \psi_{A,w_{I_2,S,k}}^{a_i,c}$
11: $\psi_{D,I}^{a_i} = \psi_{D,w_{I_2,S,k}}^{a_i,c}$
Trigger t_{11} and t_{13} of $\lfloor$*SA operation*$\rfloor[a_i, c]$,
12: t_1 of $\lfloor$*SA vector instruction history*$\rfloor[a_i, c]$ and
t_1 of $\lfloor$*SA vector outbound STAR history*$\rfloor[a_i, c]$
13: $\mathcal{F}_{w_{I_2}}(S^{a_i,c}, k^{a_i,c})$

⟨vector inbound STAR instruction⟩

- State entry actions:

1: $a_C^c = a_{I_3}^c$
2: $z_I^{a_i} = z_{w_{S,k}}^{a_i,c}$
3: $z_I^{a_i,c} = z_{w_{S,k}}^{a_i,c}$
4: $v_{IAS,I}^{a_i} = v_{IAS,w_{S,k}}^{a_i,c}$
5: $v_{IAS,I}^{a_i,c} = v_{IAS,w_{S,k}}^{a_i,c}$
6: $\psi_{A,I}^{a_i} = \psi_{A,w_{S,k}}^{a_i,c}$
7: $\psi_{D,I}^{a_i} = \psi_{D,w_{S,k}}^{a_i,c}$
8: $x_I^{a_i} = x_{w_{S,k}}^{a_i,c}$
9: $y_I^{a_i} = y_{w_{S,k}}^{a_i,c}$
Trigger t_{12} of $\lfloor$*SA operation*$\rfloor[a_i, c]$,
10: t_3 of $\lfloor$*SA vector inbound STAR*$\rfloor[a_i, c]$ and
t_1 of $\lfloor$*SA vector instruction history*$\rfloor[a_i, c]$

⟨vector outbound merge instruction⟩

- State entry actions:

1: $a_C^c = a_{I_4}^c$
2: $k^{a_i,c} = k_{I_5,S}^{a_i,c}$
3: $k^{a_i} = k_{I_5,S}^{a_i,c}$
4: $z_I^{a_i} = z_{w_{S,k}}^{a_i,c}$
5: $v_{IAS,I}^{a_i} = v_{IAS,w_{S,k}}^{a_i,c}$

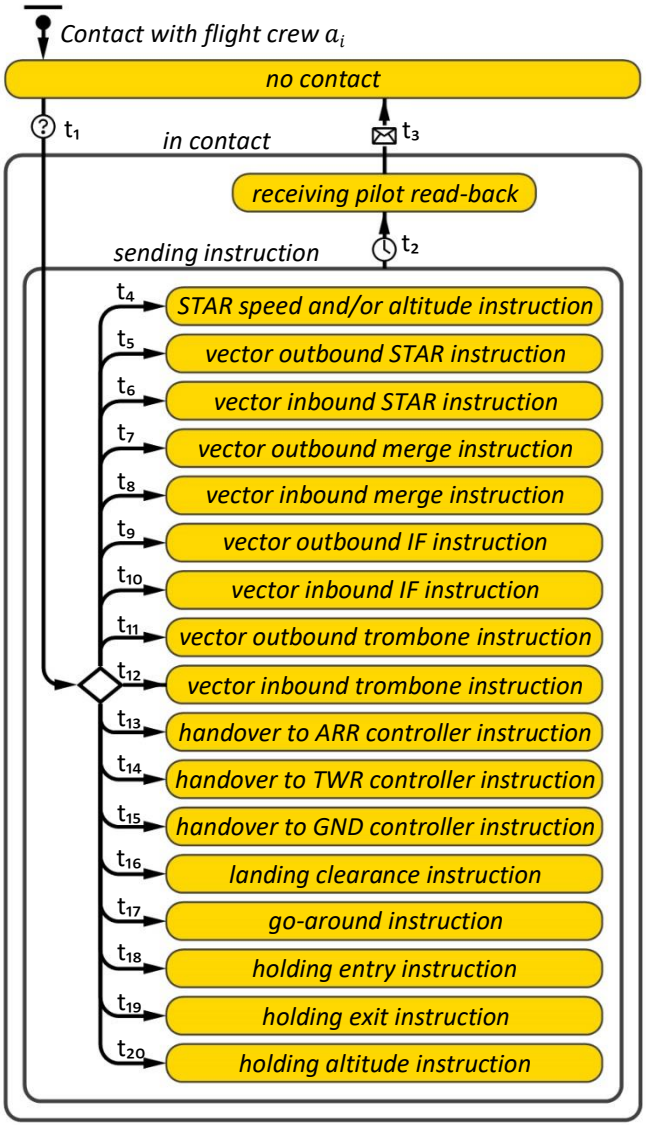


Figure 105 – ⌊Contact with flight crew a_i⌋[c]

6: $\psi_{\mathrm{A,I}}^{a_i} = \psi_{\mathrm{A},w_{I_4,S}}^{a_i,c}$
7: $\psi_{\mathrm{D,I}}^{a_i} = \psi_{\mathrm{D},w_{I_4,S}}^{a_i,c}$
8: Trigger t_5, t_9 and t_{10} of $\lfloor$*SA operation*$\rfloor[a_i, c]$ and t_1 of $\lfloor$*SA vector instruction history*$\rfloor[a_i, c]$

⟨vector inbound merge instruction⟩

- State entry actions:

1: $a_{\mathrm{C}}^{c} = a_{I_5}^{c}$
2: $z_{\mathrm{I}}^{a_i} = z_{w_{S,k}}^{a_i,c}$
3: $v_{\mathrm{IAS,I}}^{a_i} = v_{\mathrm{IAS},w_{S,k}}^{a_i,c}$
4: $\psi_{\mathrm{A,I}}^{a_i} = \psi_{\mathrm{A},w_{S,k}}^{a_i,c}$
5: $\psi_{\mathrm{D,I}}^{a_i} = \psi_{\mathrm{D},w_{S,k}}^{a_i,c}$
6: $x_{\mathrm{I}}^{a_i} = x_{w_{S,k}}^{a_i,c}$
7: $y_{\mathrm{I}}^{a_i} = y_{w_{S,k}}^{a_i,c}$
8: Trigger t_6 of $\lfloor$*SA operation*$\rfloor[a_i, c]$, t_3 of $\lfloor$*SA vector inbound merge*$\rfloor[a_i, c]$ and t_1 of $\lfloor$*SA vector instruction history*$\rfloor[a_i, c]$

⟨vector outbound IF instruction⟩

- State entry actions:

1: **if** $(I^c == I_{6,1})$ **then**
2: $a_{\mathrm{C}}^{c} = a_{I_{6,1}}^{c}$
3: **else if** $(I^c == I_{6,2})$ **then**
4: $a_{\mathrm{C}}^{c} = a_{I_{6,2}}^{c}$
5: **end if**
6: $\psi_{\mathrm{A,I}}^{a_i} = \psi_{\mathrm{A},w_{I_6,S,N}}^{a_i,c}$
7: $\psi_{\mathrm{D,I}}^{a_i} = \psi_{\mathrm{D},w_{I_6,S,N}}^{a_i,c}$
8: Trigger t_{17} of $\lfloor$*SA operation*$\rfloor[a_i, c]$ and t_1 of $\lfloor$*SA vector instruction history*$\rfloor[a_i, c]$

⟨vector inbound IF instruction⟩

- State entry actions:

1: $a_{\mathrm{C}}^{c} = a_{I_7}^{c}$
2: $z_{\mathrm{I}}^{a_i} = z_{w_{\mathrm{IF}}}$
3: $v_{\mathrm{IAS,I}}^{a_i} = v_{\mathrm{IAS},w_{\mathrm{IF}}}$
4: $\psi_{\mathrm{A,I}}^{a_i} = \psi_{\mathrm{A},w_{\mathrm{IF}}}^{a_i,c}$
5: $\psi_{\mathrm{D,I}}^{a_i} = \psi_{\mathrm{D},w_{\mathrm{IF}}}^{a_i,c}$
6: $x_{\mathrm{I}}^{a_i} = x_{w_{\mathrm{IF}}}$
7: $y_{\mathrm{I}}^{a_i} = y_{w_{\mathrm{IF}}}$
8: Trigger t_{15}, t_{16} and t_{18} of $\lfloor$*SA operation*$\rfloor[a_i, c]$, t_3 of $\lfloor$*SA vector inbound IF*$\rfloor[a_i, c]$ and t_1 of $\lfloor$*SA vector instruction history*$\rfloor[a_i, c]$
9: $\{a_i\} \setminus \mathbb{A}_S$

⟨vector outbound trombone instruction⟩

- State entry actions:

1: $a_{\mathrm{C}}^{c} = a_{I_8}^{c}$
2: $z_{\mathrm{I}}^{a_i} = z_{w_{S,k}}^{a_i,c}$
3: $v_{\mathrm{IAS,I}}^{a_i} = v_{\mathrm{IAS},w_{S,k}}^{a_i,c}$
4: $\psi_{\mathrm{A,I}}^{a_i} = \psi_{\mathrm{A},w_{I_8,S}}^{a_i,c}$
5: $\psi_{\mathrm{D,I}}^{a_i} = \psi_{\mathrm{D},w_{I_8,S}}^{a_i,c}$
6: Trigger t_{14}, t_{19}, t_{20} and t_{22} of $\lfloor$*SA operation*$\rfloor[a_i, c]$
7: $\{a_i\} \setminus \mathbb{A}_{\mathrm{IF}}$
8: $\{a_i\} \setminus \mathbb{A}_S$

9: $\{a_i\} \cup \mathbb{A}_{I_8}$
10: $\{a_i\} \cup \mathbb{A}_{I_8,S}$

⟨vector inbound trombone instruction⟩

- State entry actions:

Variable	Initial value	State space	Unit	Description
$\breve{a}$	-	$\mathbb{A}_S$	-	Iterator, i.e. specific aircraft in $\mathbb{A}_S$, starting at $\mathbb{A}_S[0]$
$\breve{b}$	false	*Boolean*	-	Boolean that indicates if $a_{5,i}$ is found
$\breve{\imath}$	0	$\mathbb{N}_0$	-	Index of $a_{5,i}$ in $\mathbb{A}_S$, where $S == S^{a_i}$
$\breve{k}^{\breve{a}}$	-	$\mathbb{N}_0$	-	Current waypoint number of $\breve{a}$

1: $a_C^c = a_{I_9}^c$
2: $z_{\text{I}}^{a_i} = z_{w_{S,k}}^{a_i,c}$
3: $v_{\text{IAS,I}}^{a_i} = v_{\text{IAS},w_{S,k}}^{a_i,c}$
4: $\psi_{\text{A,I}}^{a_i} = \psi_{\text{A},w_{S,k}}^{a_i,c}$
5: $\psi_{\text{D,I}}^{a_i} = \psi_{\text{D},w_{S,k}}^{a_i,c}$
6: $x_{\text{I}}^{a_i} = x_{w_{S,k}}^{a_i,c}$
7: $y_{\text{I}}^{a_i} = y_{w_{S,k}}^{a_i,c}$
8: Trigger t_{24} of $\lfloor$*SA operation*$\rfloor[a_i, c]$, t_3 of $\lfloor$*SA vector inbound trombone*$\rfloor[a_i, c]$ and t_1 of $\lfloor$*SA vector instruction history*$\rfloor[a_i, c]$
9: restart $\mathcal{E}_{\text{MS}}^{a_i,c}$
10: $\{a_i\} \cup \mathbb{A}_{I_9,S}$
11: **for** ($\forall \breve{a} \in \mathbb{A}_S$) **do**
12: **if** ($\breve{k}^{\breve{a}} < k_{I_9,S}$) **then**
13: $\breve{\imath} = (i\{\breve{a}\} \in \mathbb{A}_S)$
14: $\breve{\imath}\{a_i\} \cup \mathbb{A}_S$ //add a_i at index $\breve{\imath}$ in $\mathbb{A}_S$, i.e. add a_i before $\breve{a}$
15: $\breve{b}$ = true
16: **break**
17: **end if**
18: **end for**
19: **if** ($\breve{b}$ == *false*) **then**
20: $\{a_i\} \cup \mathbb{A}_S$ //add a_i to $\mathbb{A}_S$, i.e. add a_i after last element of $\mathbb{A}_S$
21: **end if**

⟨handover to ARR controller instruction⟩

- State entry actions:

1: $a_C^c = a_{I_{10}}^c$
2: $\{a_i\} \setminus \mathbb{A}^c$, where $c == c^{a_i}$ and $c \in \{c_1, c_2\}$
3: Trigger t_3 of $\lfloor$*SA handover to ARR*$\rfloor[a_i, c]$

⟨handover to TWR controller instruction⟩

- State entry actions:

1: $a_C^c = a_{I_{11}}^c$
2: $\{a_i\} \setminus \mathbb{A}^c$, where $c == c^{a_i} == c_3$
3: Trigger t_3 of $\lfloor$*SA handover to TWR*$\rfloor[a_i, c]$

⟨handover to GND controller instruction⟩

- State entry actions:

1: $a_C^c = a_{I_{12}}^c$
2: $\{a_i\} \setminus \mathbb{A}^c$, where $c == c^{a_i} == c_4$
3: Trigger t_3 of $\lfloor$*SA handover to GND*$\rfloor[a_i, c]$

⟨landing clearance instruction⟩

- State entry actions:

1: $a_C^c = a_{I_{13}}^c$
2: Trigger t_3 of $\lfloor$*SA landing clearance*$\rfloor[a_i, c]$

⟨go-around instruction⟩

- State entry actions:

1: $a_C^c = a_{I_{14}}^c$
2: $z_I^{a_i} = z_{I_{14}}$
3: $v_{IAS,I}^{a_i} = v_{IAS,I_{14}}$
4: $\{a_i\} \setminus \mathbb{A}^c$, where $c == c^{a_i} == c_4$
5: $\{a_i\} \setminus \mathbb{A}$
6: Trigger t_{29} of $\lfloor$*SA operation*$\rfloor[a_i, c]$

⟨holding entry instruction⟩

- State entry actions:

1: $a_C^c = a_{I_{15}}^c$
2: $k^{a_i,c} = k_{I_{15},S}^{a_i,c}$
3: $k^{a_i} = k_{I_{15},S}^{a_i,c}$
4: $z_I^{a_i} = z_{R,w_{S,H}}^{a_i,c}$
5: $z_I^{a_i,c} = z_{R,w_{S,H}}^{a_i,c}$
6: $v_{IAS,I}^{a_i} = v_{IAS,w_{S,H}}^{a_i,c}$
7: $\{a_i\} \cup \mathbb{A}_{S,H}$
8: Trigger t_1 and t_{26} $\lfloor$*SA operation*$\rfloor[a_i, c]$ and t_3 of $\lfloor$*SA holding entry*$\rfloor[a_i, c]$

⟨holding exit instruction⟩

- State entry actions:

1: $a_C^c = a_{I_{16}}^c$
2: $\psi_{A,I}^{a_i} = \psi_{w_{S,H,I}}^{a_i}$
3: $\psi_{D,I}^{a_i} = \psi_{D,w_{S,H,I}}^{a_i}$
4: $x_I^{a_i} = x_{w_{S,H,I}}^{a_i}$
5: $y_I^{a_i} = y_{w_{S,H,I}}^{a_i}$
6: $\{a_i\} \setminus \mathbb{A}_{S,H}$
7: Trigger t_2 of $\lfloor$*SA operation*$\rfloor[a_i, c]$ and t_3 of $\lfloor$*SA holding exit*$\rfloor[a_i, c]$

⟨holding altitude instruction⟩

- State entry actions:

1: $a_C^c = a_{I_{17}}^c$
2: $z_I^{a_i} = z_{R,w_{S,H}}^{a_i,c}$
3: $z_I^{a_i,c} = z_{R,w_{S,H}}^{a_i,c}$

Transition t_1

- Condition: ⟨*task execution*⟩$\lfloor$*Workflow*$\rfloor[c]$

Transition t_2

- Timeout: t_{INS}^c
- Action:

1: Trigger t_2 of $\lfloor$*Contact with* $c^{a_i}\rfloor[$*flight crew* $a_i]$
2: $t_{INS}^c = \mathcal{L}(\mu_{t_{INS}}^c, \sigma_{t_{INS}}^c, l_{t_{INS}}^c)$

Transition t_3

Triggered in each simple state in ⟨*execution and read-back*⟩$\lfloor$*Contact with* $c^{a_i}\rfloor[$*flight crew* $a_i]$, i.e. when c receives read-back by the pilots of a_i.

Transitions t_4 - t_{20}

All condition triggered:

Transition	Condition	Transition	Condition	Transition	Condition	Transition	Condition
t_4	$I^c == I_1$	t_9	$I^c == I_{6,1} \vee I^c == I_{6,2}$	t_{14}	$I^c == I_{11}$	t_{19}	$I^c == I_{16}$
t_5	$I^c == I_{2,1} \vee I^c == I_{2,2}$	t_{10}	$I^c == I_7$	t_{15}	$I^c == I_{12}$	t_{20}	$I^c == I_{17}$
t_6	$I^c == I_3$	t_{11}	$I^c == I_8$	t_{16}	$I^c == I_{13}$		
t_7	$I^c == I_4$	t_{12}	$I^c == I_9$	t_{17}	$I^c == I_{14}$		
t_8	$I^c == I_5$	t_{13}	$I^c == I_{10}$	t_{18}	$I^c == I_{15}$		

C.4.5.4 Control mode

The $\lfloor$*Control mode*$\rfloor[c]$ statechart (figure 106) models the different contextual control modes of controller c ($c \in \mathbb{C}$) and the corresponding implications of each specific control mode.

⟨tactical⟩

Parameter	Value	Unit	Description
$l_{t_{IDE,T}}$	2.1	s	Minimum value of t^c_{IDE} *
$l_{t_{INS,T}}$	2.6	s	Minimum value of t^c_{INS} *
$l_{t_{SCN,T}}$	0.6	s	Minimum value of t^c_{SCN} *
$l_{t_{SHD,T}}$	0.4	s	Minimum value of t^c_{SHD} *
$\mu_{t_{IDE,T}}$	1.1	s	Mean value of t^c_{IDE} *
$\mu_{t_{INS,T}}$	2.5	s	Mean value of t^c_{INS} *
$\mu_{t_{SCN,T}}$	1.4	s	Mean value of t^c_{SCN} *
$\mu_{t_{SHD,T}}$	1.0	s	Mean value of t^c_{SHD} *
$\sigma_{t_{IDE,T}}$	0.4	s	Standard deviation of t^c_{IDE} *
$\sigma_{t_{INS,T}}$	0.2	s	Standard deviation of t^c_{INS} *
$\sigma_{t_{SCN,T}}$	0.2	s	Standard deviation of t^c_{SCN} *
$\sigma_{t_{SHD,T}}$	0.3	s	Standard deviation of t^c_{SHD} *

* when c is operating in the tactical control mode

Table 39 – Parameter values that define the durations of the scanning and task identification, -scheduling and execution processes when the controller is operating in the tactical control mode

- State entry actions:

1: $M^c = M_T$

2: $\mu^c_{t_{IDE}} = \mu_{t_{SCN,T}}$ 3: $\sigma^c_{t_{IDE}} = \sigma_{t_{SCN,T}}$ 4: $l^c_{t_{IDE}} = l_{t_{SCN,T}}$ 5: $t^c_{IDE} = \mathcal{L}(\mu^c_{t_{IDE}}, \sigma^c_{t_{IDE}}, l^c_{t_{IDE}})$

6: $\mu^c_{t_{INS}} = \mu_{t_{SCN,T}}$ 7: $\sigma^c_{t_{INS}} = \sigma_{t_{SCN,T}}$ 8: $l^c_{t_{INS}} = l_{t_{SCN,T}}$ 9: $t^c_{INS} = \mathcal{L}(\mu^c_{t_{INS}}, \sigma^c_{t_{INS}}, l^c_{t_{INS}})$

10: $\mu^c_{t_{SCN}} = \mu_{t_{SCN,T}}$ 11: $\sigma^c_{t_{SCN}} = \sigma_{t_{SCN,T}}$ 12: $l^c_{t_{SCN}} = l_{t_{SCN,T}}$ 13: $t^c_{SCN} = \mathcal{L}(\mu^c_{t_{SCN}}, \sigma^c_{t_{SCN}}, l^c_{t_{SCN}})$

14: $\mu^c_{t_{SHD}} = \mu_{t_{SCN,T}}$ 15: $\sigma^c_{t_{SHD}} = \sigma_{t_{SCN,T}}$ 16: $l^c_{t_{SHD}} = l_{t_{SCN,T}}$ 17: $t^c_{SHD} = \mathcal{L}(\mu^c_{t_{SHD}}, \sigma^c_{t_{SHD}}, l^c_{t_{SHD}})$

⟨opportunistic⟩

Parameter	Value	Unit	Description
$l_{t_{IDE,O}}$	1.1	s	Minimum value of t^c_{IDE} *
$l_{t_{INS,O}}$	2.5	s	Minimum value of t^c_{INS} *
$l_{t_{SCN,O}}$	0.4	s	Minimum value of t^c_{SCN} *
$l_{t_{SHD,O}}$	0.3	s	Minimum value of t^c_{SHD} *
$\mu_{t_{IDE,O}}$	1.1	s	Mean value of t^c_{IDE} *
$\mu_{t_{INS,O}}$	1.4	s	Mean value of t^c_{INS} *
$\mu_{t_{SCN,O}}$	1.1	s	Mean value of t^c_{SCN} *
$\mu_{t_{SHD,O}}$	0.6	s	Mean value of t^c_{SHD} *
$\sigma_{t_{IDE,O}}$	0.4	s	Standard deviation of t^c_{IDE} *

$\sigma_{t_{\text{INS,O}}}$	0.3	s	Standard deviation of t^c_{INS} *
$\sigma_{t_{\text{SCN,O}}}$	0.2	s	Standard deviation of t^c_{SCN} *
$\sigma_{t_{\text{SHD,O}}}$	0.6	s	Standard deviation of t^c_{SHD} *

* when c is operating in the opportunistic control mode

Table 40 – Parameter values that define the durations of the scanning and task identification, -scheduling and execution processes when the controller is operating in the opportunistic control mode

- State entry actions:

1: $M^c = M_{\text{O}}$
2: $\mu^c_{t_{\text{IDE}}} = \mu_{t_{\text{SCN,O}}}$
3: $\sigma^c_{t_{\text{IDE}}} = \sigma_{t_{\text{SCN,O}}}$
4: $l^c_{t_{\text{IDE}}} = l_{t_{\text{SCN,O}}}$
5: $t^c_{\text{IDE}} = \mathcal{L}(\mu^c_{t_{\text{IDE}}}, \sigma^c_{t_{\text{IDE}}}, l^c_{t_{\text{IDE}}})$
6: $\mu^c_{t_{\text{INS}}} = \mu_{t_{\text{SCN,O}}}$
7: $\sigma^c_{t_{\text{INS}}} = \sigma_{t_{\text{SCN,O}}}$
8: $l^c_{t_{\text{INS}}} = l_{t_{\text{SCN,O}}}$
9: $t^c_{\text{INS}} = \mathcal{L}(\mu^c_{t_{\text{INS}}}, \sigma^c_{t_{\text{INS}}}, l^c_{t_{\text{INS}}})$
10: $\mu^c_{t_{\text{SCN}}} = \mu_{t_{\text{SCN,O}}}$
11: $\sigma^c_{t_{\text{SCN}}} = \sigma_{t_{\text{SCN,O}}}$
12: $l^c_{t_{\text{SCN}}} = l_{t_{\text{SCN,O}}}$
13: $t^c_{\text{SCN}} = \mathcal{L}(\mu^c_{t_{\text{SCN}}}, \sigma^c_{t_{\text{SCN}}}, l^c_{t_{\text{SCN}}})$
14: $\mu^c_{t_{\text{SHD}}} = \mu_{t_{\text{SCN,O}}}$
15: $\sigma^c_{t_{\text{SHD}}} = \sigma_{t_{\text{SCN,O}}}$
16: $l^c_{t_{\text{SHD}}} = l_{t_{\text{SCN,O}}}$
17: $t^c_{\text{SHD}} = \mathcal{L}(\mu^c_{t_{\text{SHD}}}, \sigma^c_{t_{\text{SHD}}}, l^c_{t_{\text{SHD}}})$

Parameter	Value	Unit	Description
$Q_{\text{b},I,\text{O}}$	20	-	Bound (i.e. number of instructions) that marks the transition from the tactical control mode to the opportunistic control mode when the number of recently provided instructions exceeds $Q_{\text{b},I,\text{O}}$
$Q_{\text{b},I,\text{T}}$	15	-	Bound (i.e. number of instructions) that marks the transition from the opportunistic control mode to the tactical control mode when the number of recently provided instructions drops below $Q_{\text{b},I,\text{T}}$

Transition t_1

- Condition: $|\mathbb{T}^c_{\text{I}}| \geq Q_{\text{b},I,\text{O}}$

Transition t_2

- Condition: $|\mathbb{T}^c_{\text{I}}| \leq Q_{\text{b},I,\text{T}}$

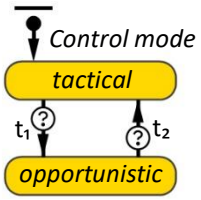


Figure 106 – [Control mode][c]

C.4.5.5 Contact with supervisor about reduced capacity

The [*Contact with supervisor about reduced capacity*][c] statechart (figure 107) models the contact between the supervisor and the controller in which the controller agent is informed about the reduced throughput capacity due to a sudden bad weather disturbance at the airport.

⟨on stand-by for reduced capacity update⟩

- State entry actions: N/A

⟨reduced capacity update received⟩

- State entry actions:

1: **if** (c == c_1) **then**
2: $t_{\text{B}}^{a_{1,i},c_1} = t_{\text{B},C_{\text{D}}}^{a_{1,i},c_1}$
3: **else if** (c == c_2) **then**
4: $t_{\text{B}}^{a_{1,i},c_2} = t_{\text{B},C_{\text{D}}}^{a_{1,i},c_2}$
5: **else if** (c == c_3) **then**
6: $t_{\text{B}}^{a_{1,i},c_3} = t_{\text{B},C_{\text{D}}}^{a_{1,i},c_3}$
7: $t_{\text{B}}^{a_{2,i},c_3} = t_{\text{B},C_{\text{D}}}^{a_{2,i},c_3}$
8: **else if** (c == c_4) **then**
9: $t_{\text{B}}^{a_{3,i},c_4} = t_{\text{B},C_{\text{D}}}^{a_{3,i},c_4}$
10: **end if**

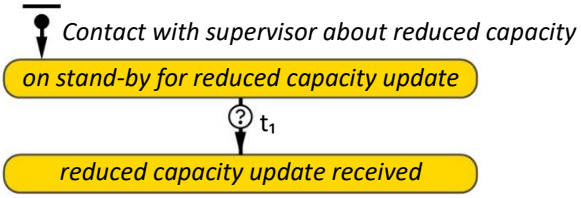


Figure 107 – [Contact with supervisor about reduced capacity][c]

Transition t_1

- Condition: (c == c_1 ∧ ⟨*TNW controller has been informed*⟩⌊*Contact with TNW controller about reduced capacity*⌋[*Supervisor APP*]) ∨ (c == c_2 ∧ ⟨*TNE controller has been informed*⟩⌊*Contact with TNE controller about reduced capacity*⌋[*Supervisor APP*]) ∨ (c == c_3 ∧ ⟨*ARR controller has been informed*⟩⌊*Contact with ARR controller about reduced capacity*⌋[*Supervisor APP*]) ∨ (c == c_4 ∧ ⟨*TWR controller has been informed*⟩⌊*Contact with TWR controller about reduced capacity*⌋[*Supervisor TWR*])

C.4.5.6 Contact with supervisor about recovered capacity

The ⌊*Contact with supervisor about recovered capacity*⌋[c] statechart (figure 108) models the contact between the supervisor and the controller in which the controller agent is informed about the recovered throughput capacity due to normalised weather conditions at the airport.

⟨on stand-by for recovered capacity update⟩

- State entry actions: N/A

⟨recovered capacity update received⟩

- State entry actions:

1: **if** (c == c_1) **then**
2: $t_B^{a_{1,i},c_1} = t_{B,C_l}^{a_{1,i},c_1}$
3: **else if** (c == c_2) **then**
4: $t_B^{a_{1,i},c_2} = t_{B,C_l}^{a_{1,i},c_2}$
5: **else if** (c == c_3) **then**
6: $t_B^{a_{1,i},c_3} = t_{B,C_l}^{a_{1,i},c_3}$
7: $t_B^{a_{2,i},c_3} = t_{B,C_l}^{a_{2,i},c_3}$
8: **else if** (c == c_4) **then**
9: $t_B^{a_{3,i},c_4} = t_{B,C_l}^{a_{3,i},c_4}$
10: **end if**

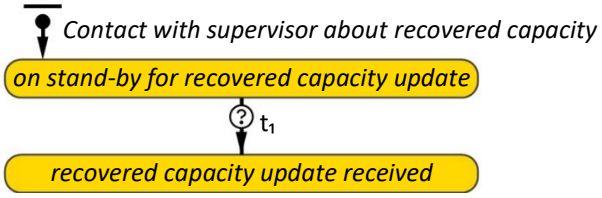


Figure 108 – ⌊Contact with supervisor about recovered capacity⌋[c]

Transition t_1

- Condition: (c == c_1 ∧ ⟨*TNW controller has been informed*⟩⌊*Contact with TNW controller about recovered capacity*⌋[*Supervisor APP*]) ∨ (c == c_2 ∧ ⟨*TNE controller has been informed*⟩⌊*Contact with TNE controller about recovered capacity*⌋[*Supervisor APP*]) ∨ (c == c_3 ∧ ⟨*ARR controller has been informed*⟩⌊*Contact with ARR controller about recovered capacity*⌋[*Supervisor APP*]) ∨ (c == c_4 ∧ ⟨*TWR controller has been informed*⟩⌊*Contact with TWR controller about recovered capacity*⌋[*Supervisor TWR*])

C.5 Feeder controller

C.5.1 Applied strategy in modelling aircraft generations

This section is meant to clarify the use of the fictitious aircraft pairing a_1 and a_2 that is used to model feasible aircraft generations at each entry point. The general setup can be seen visualized in figure 109. The aircraft pairing a_1 and a_2 is used by the feeder agent to define the required time period between two subsequent generations at each entry point. Fictitious aircraft have been considered instead of already generated aircraft since they allow to compare the desired inter-arrival time periods with the required inter-arrival time periods due to the required separation distance between a_1 and a_2. In that way the fictitious aircraft a_1 and a_2 enable aircraft generations at $w_{S,0}$ ($S \in \mathbb{S}$) with 1) a sufficient initial separation between the generated aircraft and 2) with a rate that approaches the desired throughput capacity of the feeder agent.

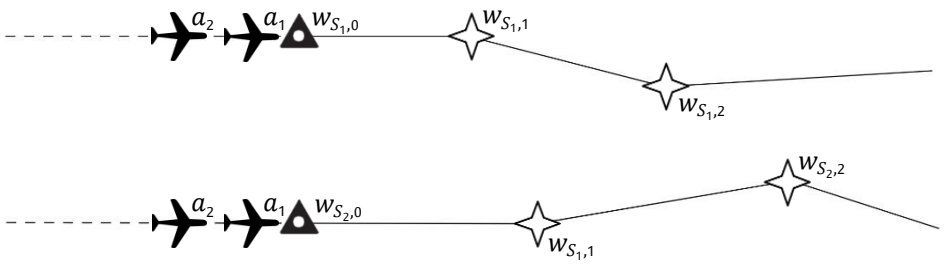


Figure 109 – General aircraft setting with a_1 and a_2 at $w_{S,0}$ ($\forall S \in \mathbb{S}$) as considered by the feeder agent

The initial spacing between a_1 and a_2 is achieved and managed using the same separation practice as the one that is applied by the controller agents. Figure 110 visualizes the variables that are used by the feeder agent to guarantee sufficient (initial) spacing between the (to be) generated aircraft. These specific variables are used in the multiple sections below that relate to the specification of the generation practice of the feeder agent.

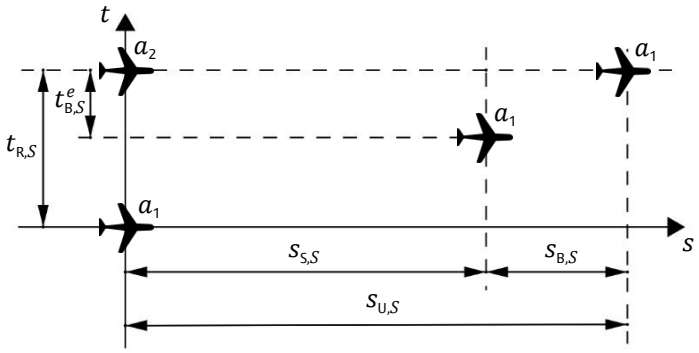


Figure 110 – Logic behind the generation of aircraft while ensuring a viable separation between a_1 and a_2 at $w_{S,0}$

C.5.2 Variables, parameters and sets

Variable	State space	Unit	Description
C	$\mathbb{R}_+$	ac·h^{-1}	Desired (total) throughput capacity
$D_S^{a_1}$	$\mathbb{D}$	-	Aircraft type of a_1
$D_S^{a_2}$	$\mathbb{D}$	-	Aircraft type of a_2
l_{t_G}	$\mathbb{R}_+$	s	Minimum value of t_G
$s_{B,S}$	$\mathbb{R}_+$	NM	*Separation buffer (type I)* between a_1 and a_2
$s_{S,S}$	$\mathbb{R}_{>0}$	NM	*Separation minima* between a_1 and a_2
$s_{U,S}$	$\mathbb{R}_{>0}$	NM	*Desired spacing* between a_1 and a_2
t_B	$\mathbb{R}_+$	s	Default *time buffer* between a_1 and a_2
$t_{B,S}$	$\mathbb{R}_+$	s	*Time buffer* between a_1 and a_2
t_G	$\mathbb{R}_+$	s	Recurrence time period of $\mathcal{E}_{TG}$
$t_{N,S}$	$\mathbb{R}_{>0}$	s	Minimum time point at which a next feasible aircraft generation may occur
$t_{R,S}$	$\mathbb{R}_{>0}$	s	Required time period between the generations of a_1 and a_2 to achieve and guarantee sufficient initial spacing
u_{t_G}	$\mathbb{R}_+$	s	Maximum value of t_G
$v_{GS,S}$	$\mathbb{R}_{>0}$	kt	Initial ground speed of a_1
$W_S^{a_1}$	$\mathbb{W}$	-	WTC of a_1
$W_S^{a_2}$	$\mathbb{W}$	-	WTC of a_2
μ_{t_G}	$\mathbb{R}_+$	s	Mean value of t_G

Table 41 – List of variables that belong to the feeder agent

Parameter	Value	Unit	Description
C_{I}	varying	ac·h^{-1}	Desired (total) throughput capacity in *normal capacity mode*
C_{D}	21	ac·h^{-1}	Desired (total) throughput capacity in *reduced capacity mode*
$t_{\mathrm{b,G}}$	40	s	Bound that determines the band width of t_{G}, i.e. $l_{t_{\mathrm{G}}}$ and $u_{t_{\mathrm{G}}}$
$t_{\mathrm{B},C_{\mathrm{D}}}^{a_{1,2}}$	10	s	*Time buffer* between a_1 and a_2 in *reduced capacity mode*
$t_{\mathrm{B},C_{\mathrm{I}}}^{a_{1,2}}$	60	s	*Time buffer* between a_1 and a_2 in *normal capacity mode*
$t_{\mathrm{B,HM}}^{a_{1,2}}$	40	s	Additional *time buffer* between a_1 and a_2
$t_{\mathrm{B,MH}}^{a_{1,2}}$	60	s	Additional *time buffer* between a_1 and a_2
$\sigma_{t_{\mathrm{G}}}$	10	s	Standard deviation of t_{G}

Table 42 – List of parameters that belong to the feeder agent

Set	Description
$\mathbb{T}_{\mathrm{G},S}$	Set of time points that define when a_i should be generated at $w_{S,0}$ ($S \in \mathbb{S}$) according to event $\mathcal{E}_{\mathrm{TG}}$

Table 43 – List of sets that belong to the feeder agent

C.5.3 Initial actions and values

1: $C = C_{\mathrm{I}}$
2: $t_{\mathrm{B}} = t_{\mathrm{B},C_{\mathrm{I}}}^{a_{1,2}}$
3: $\mu_{t_{\mathrm{G}}} = 3600 \,/\, C$
4: $l_{t_{\mathrm{G}}} = \mu_{t_{\mathrm{G}}} - t_{\mathrm{b,G}}$
5: $u_{t_{\mathrm{G}}} = \mu_{t_{\mathrm{G}}} + t_{\mathrm{b,G}}$
6: $t_{\mathrm{G}} = \mathcal{N}_t(\mu_{t_{\mathrm{G}}}, \sigma_{t_{\mathrm{G}}}, l_{t_{\mathrm{G}}}, u_{t_{\mathrm{G}}})$

Action 5 – Setting up the variables of the feeder agent that require an initial value

1: $W_S^{a_1} = \mathcal{F}_W()$
2: $D_S^{a_1} = \mathcal{F}_D(W_S^{a_1})$
3: $W_S^{a_2} = \mathcal{F}_W()$
4: $D_S^{a_2} = \mathcal{F}_D(W_S^{a_2})$
5: $v_{\mathrm{GS},S} = \mathcal{F}_{v_{\mathrm{GS}}}(z_{w_{S,0}}, v_{\mathrm{IAS},w_{S,0}}, \psi_{w_{S,0}})$
6: $t_{\mathrm{B},S} = t_{\mathrm{B}}$
7: **if** ($W_S^{a_1} == W_{\mathrm{H}} \wedge W_S^{a_2} == W_{\mathrm{M}}$) **then**
8: $\quad t_{\mathrm{B},S}$ += $t_{\mathrm{B,HM}}^{a_{1,2}}$
9: **else if** ($W_S^{a_1} == W_{\mathrm{M}} \wedge W_S^{a_2} == W_{\mathrm{H}}$) **then**
10: $\quad t_{\mathrm{B},S}$ += $t_{\mathrm{B,MH}}^{a_{1,2}}$
11: **end if**
12: $s_{\mathrm{B},S} = t_{\mathrm{B},S} \cdot v_{\mathrm{GS},S} \,/\, 3600$
13: $s_{\mathrm{S},S} = \mathcal{F}_{s_{\mathrm{S}}}(W_S^{a_1}, W_S^{a_2})$
14: $s_{\mathrm{U},S} = s_{\mathrm{S},S} + s_{\mathrm{B},S}$
15: $t_{\mathrm{R},S} = s_{\mathrm{U},S} \,/\, v_{\mathrm{GS},S} \cdot 3600$
16: $t_{\mathrm{N},S} = 0$

Action 6 – Set up initial aircraft pairing a_1 and a_2 at $w_{S,0}$ ($\forall S \in \mathbb{S}$)

C.5.4 Functions

C.5.4.1 Generate random WTC

Function $\mathcal{F}_W()$

Variable	State space	Unit	Description
$\breve{W}$	$\mathbb{W}$	-	WTC, either MEDIUM (W_M) or HEAVY (W_H)
$\breve{X}$	[0, 100]	%	Random variable that determines which $\breve{W}$ will be returned
Parameter	**Value**	**Unit**	**Description**
P_M	80	%	Probability of WTC being of category W_M

1: $\breve{X} = \mathcal{U}(0, 100)$
2: **if** ($\breve{X} < P_M$) **then**
3: $\quad \breve{W} = W_M$
4: **else**
5: $\quad \breve{W} = W_H$
6: **end if**
7: **return:** $\breve{W}$

Function 22 – Generate random WTC

C.5.4.2 Generate random aircraft type

Function $\mathcal{F}_D(\breve{W})$

Argument	State space	Unit	Description
$\breve{W}$	$\mathbb{W}$	-	WTC, either MEDIUM (W_M) or HEAVY (W_H)
Variable	**State space**	**Unit**	**Description**
$\breve{D}$	$\mathbb{D}$	-	Aircraft type, either B738 (D_1) or B744 (D_2)

1: **if** ($\breve{W}$ == W_M) **then**
2: $\quad \breve{D} = D_1$
3: **else if** ($\breve{W}$ == W_H) **then**
4: $\quad \breve{D} = D_2$
5: **end if**
6: **return:** $\breve{D}$

Function 23 – Generate random aircraft type that corresponds to a given WTC

C.5.5 Events

C.5.5.1 Generation of time points

Event $\mathcal{E}_{TG}$ is concerned with the generation of time points at which aircraft are (to be) generated at one of the entry points and models the attempt of the feeder agent to deliver a desired number of aircraft per time period at the entry points to approach the desired throughput capacity. Figure 111 visualizes the probability density function that is used to define the recurrence time period of $\mathcal{E}_{TG}$, i.e. to define the desired inter-arrival times.

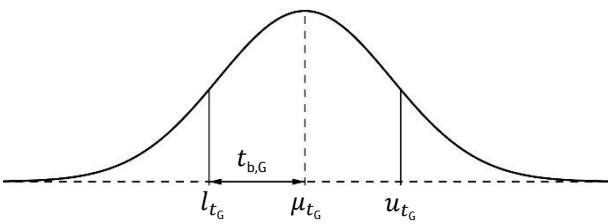


Figure 111 – On scale truncated probability function that is used to define t_G

- First occurrence time: $t_{G,0}$
- Recurrence time period: t_G

Event $\mathcal{E}_{TG}$

Variable	State space	Unit	Description
$\check{X}$	[0, 100]	%	Random variable that determines at which $w_{S,0}$ ($S \in \mathbb{S}$) a_i will be generated
Parameter	**Value**	**Unit**	**Description**
P_{S_1}	35	%	Probability that a_i will be generated at $w_{S_1,0}$
$t_{G,0}$	1	s	First occurrence time of $\mathcal{E}_{TG}$

1: $\check{X} = \mathcal{U}(0, 100)$
2: **if** ($\check{X} < P_{S_1}$) **then**
3: $t_A \cup \mathbb{T}_{G,S_1}$
4: **else**
5: $t_A \cup \mathbb{T}_{G,S_2}$
6: **end if**
7: $t_G = \mathcal{N}_t(\mu_{t_G}, \sigma_{t_G}, l_{t_G}, u_{t_G})$

Action 7 – Creation of aircraft generation time points

C.5.5.2 Generation of aircraft agents

Event $\mathcal{E}_{AG,S}$ is concerned with the actual generation of aircraft agents at $w_{S,0}$ ($S \in \mathbb{S}$). This second event is meant to evaluate the feasibility of the generated time points by $\mathcal{E}_{TG}$ in terms of initial separation. This specific event first defines the minimum time point at which aircraft may be generated by considering the required spacing between a_1 and a_2. Then the event compares the (first) scheduled and desired generation time point with the minimum generation time point. An aircraft agent will only be generated once the actual model time point is larger than the minimum generation time point or larger than the desired generation time point, whichever is greater. By confirming to this condition, traffic is generated conform the specified throughput capacity and without risk of initial conflicts. When the second event considers a generation at one of the entry points not (yet) feasible, then the scheduled time point is set 'on hold' until it satisfies the specified condition. Figure 112 shows the relations between the time points t_A, $\mathbb{T}_{G,S}[0]$ and $t_{N,S}$.

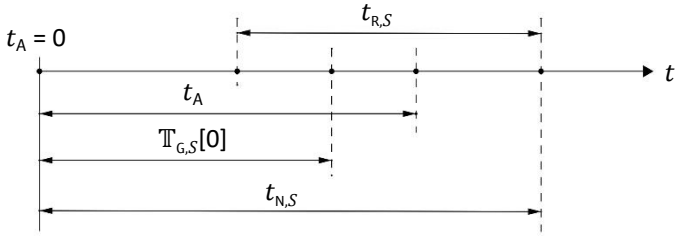


Figure 112 – Relations between the time variables that are used in events $\mathcal{E}_{TG}$ and $\mathcal{E}_{AG,S}$

- Condition: $\mathbb{T}_{G,S} \neq \emptyset \wedge (t_{N,S} < \mathbb{T}_{G,S}[0] < t_A \vee \mathbb{T}_{G,S}[0] < t_{N,S} < t_A)$, where $S \in \mathbb{S}$

Event $\mathcal{E}_{AG,S}$

Variable	State space	Unit	Description
$c_0^{a_i}$	$\{c_1, c_2\} \in \mathbb{C}$	-	Initial controller agent
D^{a_i}	$\mathbb{D}$	-	Aircraft type
S^{a_i}	$\mathbb{S}$	-	STAR procedure
$v_{IAS,0}^{a_i}$	$[v_{IAS,w_{S_1,0}}, v_{IAS,w_{S_2,0}}]$	kt	Initial calibrated airspeed
W^{a_i}	$\mathbb{W}$	-	WTC
$x_0^{a_i}$	$[x_{w_{S_1,0}}, x_{w_{S_2,0}}]$	NM	Initial x-position
$y_0^{a_i}$	$[y_{w_{S_1,0}}, y_{w_{S_2,0}}]$	NM	Initial y-position

$z_0^{a_i}$	$[z_{w_{S_1,0}}, z_{w_{S_2,0}}]$	ft	Initial altitude
$\psi_0^{a_i}$	$[\psi_{w_{S_1,0}}, \psi_{w_{S_2,0}}]$	deg	Initial heading

* all the variables of above correspond to the (to be) generated aircraft a_i

2: $x_0^{a_i} = x_{w_{S,0}}$
3: $y_0^{a_i} = y_{w_{S,0}}$
4: $z_0^{a_i} = z_{w_{S,0}}$
5: $v_{\text{IAS},0}^{a_i} = v_{\text{IAS},w_{S,0}}$
6: $\psi_0^{a_i} = \psi_{w_{S,0}}$
7: $W^{a_i} = W_S^{a_1}$
8: $D^{a_i} = D_S^{a_1}$
9: $S^{a_i} = S$
10: **if** $(S^{a_i} == S_1)$ **then**
11: $\quad c_0^{a_i} = c_1$
12: **else if** $(S^{a_i} == S_2)$ **then**
13: $\quad c_0^{a_i} = c_2$
14: **end if**
15: Create aircraft agent and assign the initial variables of above
16: $\mathbb{T}_{\text{G},S}[0] \setminus \mathbb{T}_{\text{G},S}$
17: $W_S^{a_1} = W_S^{a_2}$
18: $D_S^{a_1} = D_S^{a_2}$
19: $W_S^{a_2} = \mathcal{F}_W()$
20: $D_S^{a_2} = \mathcal{F}_D(W_S^{a_2})$
21: $v_{\text{GS},S} = \mathcal{F}_{v_{\text{GS}}}(z_{w_{S,0}}, v_{\text{IAS},w_{S,0}}, \psi_{w_{S,0}})$
22: $t_{\text{B},S} = t_{\text{B}}$
23: **if** $(W_S^{a_1} == W_{\text{H}} \wedge W_S^{a_2} == W_{\text{M}})$ **then**
24: $\quad t_{\text{B},S}$ += $t_{\text{B,HM}}^{a_{1,2}}$
25: **else if** $(W_S^{a_1} == W_{\text{M}} \wedge W_S^{a_2} == W_{\text{H}})$ **then**
26: $\quad t_{\text{B},S}$ += $t_{\text{B,MH}}^{a_{1,2}}$
27: **end if**
28: $s_{\text{B},S} = t_{\text{B},S} \cdot v_{\text{GS},S}$ / 3600
29: $s_{\text{S},S} = \mathcal{F}_{s_{\text{S}}}(W_S^{a_1}, W_S^{a_2})$
30: $s_{\text{U},S} = s_{\text{S},S} + s_{\text{B},S}$
31: $t_{\text{R},S} = s_{\text{U},S} / v_{\text{GS},S} \cdot 3600$
32: $t_{\text{N},S} = t_{\text{A}} + t_{\text{R},S}$
33: restart $\mathcal{E}_{\text{AG},S}$

Action 8 – Generate aircraft a_i at $w_{S,0}$ ($S \in \mathbb{S}$), and set up new aircraft pair a_1 and a_2

C.5.6 Statecharts

C.5.6.1 Contact with Supervisor APP about reduced capacity

The ⌊*Contact with Supervisor APP about reduced capacity*⌋[*feeder*] statechart (figure 113) models the contact between the Supervisor APP and the feeder controller in which the feeder agent is informed about the reduced throughput capacity due to a sudden bad weather disturbance at the airport.

⟨on stand-by for reduced capacity update⟩

- State entry actions: N/A

⟨reduced capacity update received⟩

- State entry actions:

1: $C = C_{\text{D}}$

2: $t_B = t^{a_{1,2}}_{B,C_D}$
3: $\mu_{t_G} = 3600 / C$
4: $l_{t_G} = \mu_{t_G} - t_{b,G}$
5: $u_{t_G} = \mu_{t_G} + t_{b,G}$
6: $t_G = \mathcal{N}_t(\mu_{t_G}, \sigma_{t_G}, l_{t_G}, u_{t_G})$
7: Clear $\mathbb{T}_{G,S}$

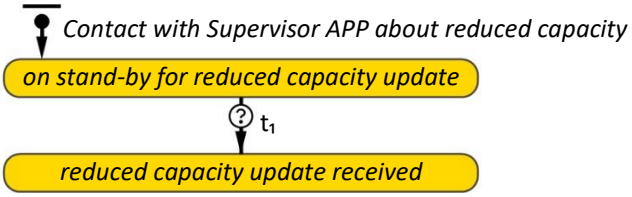


Figure 113 – ⌊Contact with Supervisor APP about reduced capacity⌋[feeder]

Transition t_1

- Condition: ⟨*feeder has been informed*⟩⌊*Contact with feeder about reduced capacity*⌋[*Supervisor APP*]

C.5.6.2 Contact with Supervisor APP about recovered capacity

The ⌊*Contact with Supervisor APP about recovered capacity*⌋[*feeder*] statechart (figure 114) models the contact between the Supervisor APP and the feeder controller in which the feeder agent is informed about the recovered throughput capacity due to normalised weather conditions at the airport.

⟨on stand-by for recovered capacity update⟩

- State entry actions: N/A

⟨recovered capacity update received⟩

- State entry actions:

1: $C = C_I$
2: $t_B = t^{a_{1,2}}_{B,C_I}$
3: $\mu_{t_G} = 3600 / C$
4: $l_{t_G} = \mu_{t_G} - t_{b,G}$
5: $u_{t_G} = \mu_{t_G} + t_{b,G}$
6: $t_G = \mathcal{N}_t(\mu_{t_G}, \sigma_{t_G}, l_{t_G}, u_{t_G})$

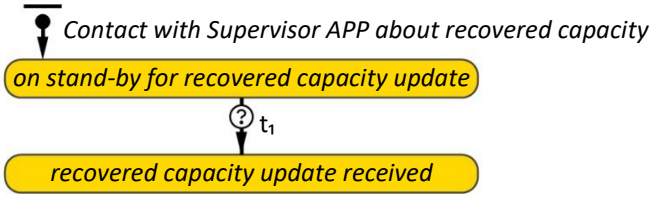


Figure 114 – ⌊Contact with Supervisor APP about recovered capacity⌋[feeder]

Transition t_1

- Condition: ⟨*feeder has been informed*⟩⌊*Contact with feeder about recovered capacity*⌋[*Supervisor APP*]

C.6 MCP

C.6.1 Variables, parameters and sets

Some of the variables in table 44 below are dependent on and defined by the STAR procedure that a_i is/has been operating (i.e. S^{a_i}), by the specific waypoint where a_i is referenced to (i.e. k^{a_i}) or by the aircraft type of a_i (i.e. D^{a_i}). These dependent variables contain the subscripts "S", "k", "D" respectively, where $S = S^{a_i}$, $k = k^{a_i}$ and $D = D^{a_i}$.

Variable	State space	Unit	Description
$d^{a_i}_{\Delta t}$	$\mathbb{R}_+$	NM	Travelled distance per Δt
$d^{a_i}_{A,w_{S,k}}$	$\mathbb{R}_+$	NM	Distance ("*tangent*") between a_i and $w_{S,k}$
$d^{a_i}_{w_{TP},w_{S,k}}$	$\mathbb{R}_+$	NM	Distance from $w_{S,k}$ to start intercepting $w_{S,k}$
$d^{a_i}_{D,w_{S,H,I}}$	$\mathbb{R}_+$	NM	Distance ("*direct*") between a_i and $w_{S,H,I}$
$d^{a_i}_{D,w_{S,H,O}}$	$\mathbb{R}_+$	NM	Distance ("*direct*") between a_i and $w_{S,H,O}$
$d^{a_i}_{A,w_{IF}}$	$\mathbb{R}_+$	NM	Distance ("*tangent*") between a_i and w_{IF}
$d^{a_i}_{w_{TP},w_{IF}}$	$\mathbb{R}_+$	NM	Distance from w_{IF} to start intercepting w_{IF}
$d^{a_i}_{D,w_{THR}}$	$\mathbb{R}_+$	NM	Distance ("*direct*") between a_i and w_{THR}
$d^{a_i}_{D,w_{HST1}}$	$\mathbb{R}_+$	NM	Distance ("*direct*") between a_i and w_{HST1}
$d^{a_i}_{D,w_{HST2}}$	$\mathbb{R}_+$	NM	Distance ("*direct*") between a_i and w_{HST2}
D^{a_i}	$\mathbb{D}$	-	Aircraft type
K^{a_i}	$\mathbb{N}_0$	-	Number of waypoints in S^{a_i}

k^{a_i}	$[0, K^{a_i}]$	-	Current waypoint number of S^{a_i}
$l^{a_i}_{v_{\text{AT},D}}$	$\mathbb{R}_+$	kt	Minimum indicated airspeed at w_{THR}
$l^{a_i}_{v_{\text{FA},D}}$	$\mathbb{R}_+$	kt	Minimum final approach speed
S^{a_i}	$\mathbb{S}$	-	STAR procedure that is/has been operated
$u^{a_i}_{v_{\text{AT},D}}$	$\mathbb{R}_+$	kt	Maximum indicated airspeed at w_{THR}
$u^{a_i}_{v_{\text{FA},D}}$	$\mathbb{R}_+$	kt	Maximum final approach speed
$\dot{v}^{a_i}$	$\mathbb{R}$	kt·s^{-1}	Acceleration/deceleration
$\dot{v}^{a_i}_{\text{C},D}$	$\mathbb{R}_+$	kt·s^{-1}	Acceleration during climb
$\dot{v}^{a_i}_{\text{D},D}$	$\mathbb{R}_-$	kt·s^{-1}	Deceleration during descent
$\dot{v}^{a_i}_{\text{L},D}$	$\mathbb{R}_-$	kt·s^{-1}	Deceleration in level flight
$v^{a_i}_{\text{AT}}$	$\mathbb{R}_+$	kt	Indicated airspeed at w_{THR}
$v^{a_i}_{\text{FA}}$	$\mathbb{R}_+$	kt	Final approach speed
$v^{a_i}_{\text{GS}}$	$\mathbb{R}_+$	kt	Ground speed
$v^{a_i}_{\text{IAS}}$	$\mathbb{R}_+$	kt	Indicated (calibrated) airspeed
$v^{a_i}_{\text{IAS,S}}$	$\mathbb{R}_+$	kt	Indicated (calibrated) airspeed as set by the flight crew a_i agent
$v^{a_i}_{\text{TAS}}$	$\mathbb{R}_+$	kt	True airspeed
W^{a_i}	$\mathbb{W}$	-	WTC
x^{a_i}	$\mathbb{R}$	NM	x-position
$x^{a_i}_{\text{S}}$	$\mathbb{R}$	NM	x-position of the point to which the aircraft is set to turn
$x^{a_i}_{w_{S,k}}$	$\mathbb{R}$	NM	x-position of $w_{S,k}$
$x^{a_i}_{w_{S,H,\text{I}}}$	$\mathbb{R}$	NM	x-position of $w_{S,H,\text{I}}$
$x^{a_i}_{w_{S,H,\text{O}}}$	$\mathbb{R}$	NM	x-position of $w_{S,H,\text{O}}$
y^{a_i}	$\mathbb{R}$	NM	y-position
$y^{a_i}_{\text{S}}$	$\mathbb{R}$	NM	y-position of the point to which the aircraft is set to turn
$y^{a_i}_{w_{S,k}}$	$\mathbb{R}$	NM	y-position of $w_{S,k}$
$y^{a_i}_{w_{S,H,\text{I}}}$	$\mathbb{R}$	NM	y-position of $w_{S,H,\text{I}}$
$y^{a_i}_{w_{S,H,\text{O}}}$	$\mathbb{R}$	NM	y-position of $w_{S,H,\text{O}}$
$\dot{z}^{a_i}$	$\mathbb{R}$	ft·min^{-1}	Vertical speed
z^{a_i}	$\mathbb{R}_+$	ft	Altitude
$z^{a_i}_{\text{S}}$	$\mathbb{R}_+$	ft	Altitude as set by the flight crew a_i agent
γ^{a_i}	[-90, 90]	deg	Flight path angle
$\gamma^{a_i}_{\text{APP}}$	[0, 90]	deg	Approach angle
$\theta^{a_i}_{w_{S,k+1}}$	[-180, 180]	deg	Angle between ψ^{a_i} and $\psi^{a_i}_{w_{S,k+1}}$
$\theta^{a_i}_{w_{\text{IF}}}$	[-180, 180]	deg	Angle between ψ^{a_i} and $\psi_{w_{\text{THR}}}$
ψ^{a_i}	⟨0, 360]	deg	Heading direction with respect to magnetic north
$\psi^{a_i}_{w_{S,k}}$	⟨0, 360]	deg	Magnetic track of $w_{S,k}$, i.e. bearing of $w_{S,k}$ seen from $w_{S,k-1}$
$\psi^{a_i}_{w_{S,k+1}}$	⟨0, 360]	deg	Magnetic track of $w_{S,k+1}$, i.e. bearing of $w_{S,k+1}$ seen from $w_{S,k}$
$\psi^{a_i}_{w_{S,H,\text{I}}}$	⟨0, 360]	deg	Magnetic track of the inbound leg towards $w_{S,H,\text{I}}$
$\psi^{a_i}_{w_{S,H,\text{O}}}$	⟨0, 360]	deg	Magnetic track of the outbound leg towards $w_{S,H,\text{O}}$
$\psi^{a_i}_{A,S}$	⟨0, 360]	deg	Heading direction (*tangent*) as set by the flight crew a_i agent
$\psi^{a_i}_{D,S}$	⟨0, 360]	deg	Heading direction (*direct*) as set by the flight crew a_i agent
$\psi^{a_i}_{A,w_{S,k}}$	⟨0, 360]	deg	Bearing (*tangent*) of $w_{S,k}$ seen from a_i
$\psi^{a_i}_{D,w_{S,k}}$	⟨0, 360]	deg	Bearing (*direct*) of $w_{S,k}$ seen from a_i
$\psi^{a_i}_{D,w_{S,H,I}}$	⟨0, 360]	deg	Bearing (*direct*) of $w_{S,H,\text{I}}$ seen from a_i
$\psi^{a_i}_{D,w_{S,H,O}}$	⟨0, 360]	deg	Bearing (*direct*) of $w_{S,H,\text{O}}$ seen from a_i
$\psi^{a_i}_{A,w_{IF}}$	⟨0, 360]	deg	Bearing (*tangent*) of w_{IF} seen from a_i
$\psi^{a_i}_{D,w_{IF}}$	⟨0, 360]	deg	Bearing (*direct*) of w_{IF} seen from a_i
$\psi^{a_i}_{w_{THR}}$	⟨0, 360]	deg	Bearing (*direct*) of w_{THR} seen from a_i
$\dot{\psi}^{a_i}$	$\mathbb{R}$	deg·s^{-1}	Current and real rate of turn
$\dot{\psi}^{a_i}_{\Delta t}$	$\mathbb{R}$	deg·s^{-1}	Achievable rate of turn per Δt

Table 44 – List of variables that belong to the MCP a_i agent

Parameter	Value	Unit	Description
f_{NM}^{M}	$5.40 \cdot 10^{-4}$	-	[m] to [NM] conversion factor
$l_{v_{\text{AT},D_1}}$	140	kt	Minimum indicated airspeed of D_1 at w_{THR}
$l_{v_{\text{AT},D_2}}$	155	kt	Minimum indicated airspeed of D_2 at w_{THR}
$l_{v_{\text{FA},D_1}}$	160	kt	Minimum final approach speed of D_1
$l_{v_{\text{FA},D_2}}$	180	kt	Minimum final approach speed of D_2
r_{T}	1.75	NM	Turn radius
r_{HST}	275	m	Radius of turnoff curve of both w_{HST1} and w_{HST2}
$u_{v_{\text{AT},D_1}}$	150	kt	Maximum indicated airspeed of D_1 at w_{THR}
$u_{v_{\text{AT},D_2}}$	165	kt	Maximum indicated airspeed of D_2 at w_{THR}
$u_{v_{\text{FA},D_1}}$	180	kt	Maximum final approach speed of D_1
$u_{v_{\text{FA},D_2}}$	200	kt	Maximum final approach speed of D_2
v_{HST}	40	kt	Taxi speed
$\dot{v}_{\text{C},D_1}$	0.5	kt·s^{-1}	Acceleration of D_1 during climb
$\dot{v}_{\text{C},D_2}$	0.5	kt·s^{-1}	Acceleration of D_2 during climb
$\dot{v}_{\text{D},D_1}$	-0.35	kt·s^{-1}	Deceleration of D_1 during descent
$\dot{v}_{\text{D},D_2}$	-0.30	kt·s^{-1}	Deceleration of D_2 during descent
$\dot{v}_{\text{L},D_1}$	-1.0	kt·s^{-1}	Deceleration of D_1 in level flight
$\dot{v}_{\text{L},D_2}$	-0.9	kt·s^{-1}	Deceleration of D_2 in level flight
$\dot{v}_{\text{R},m_0}$	-5	kt·s^{-1}	Deceleration on runway in normal weather conditions
$\dot{v}_{\text{R},m_1}$	-3	kt·s^{-1}	Deceleration on runway in bad weather conditions
$\dot{z}_{\text{C}}$	1500	ft·min^{-1}	Vertical speed during climb
γ_{GS}	3	deg	Glide slope

Table 45 – List of parameters that belong to the MCP a_i agent

C.6.2 Initial actions and values

1: $v_{\text{AT}}^{a_i} = \mathcal{U}(l_{v_{\text{AT},D}}^{a_i}, u_{v_{\text{AT},D}}^{a_i})$
2: $v_{\text{FA}}^{a_i} = \mathcal{U}(l_{v_{\text{FA},D}}^{a_i}, u_{v_{\text{FA},D}}^{a_i})$
3: D^{a_i} *
4: S^{a_i} *
5: W^{a_i} *
6: $k^{a_i} = 1$
7: $x^{a_i} = x_0^{a_i}$ *
8: $y^{a_i} = y_0^{a_i}$ *
9: $z^{a_i} = z_0^{a_i}$ *
10: $v_{\text{IAS}}^{a_i} = v_{\text{IAS},0}^{a_i}$ *
11: $\psi^{a_i} = \psi_0^{a_i}$ *
12: $\dot{v}^{a_i} = 0$
13: $\dot{\psi}^{a_i} = 0$
14: $\dot{z}^{a_i} = 0$
15: $\gamma^{a_i} = 0$
16: $v_{\text{IAS,S}}^{a_i} = v_{\text{IAS}}^{a_i}$
17: $z_{\text{S}}^{a_i} = z^{a_i}$

* defined by feeder agent in event $\mathcal{E}_{\text{AG},S}$ (appendix C.5.5.2)

Action 9 – Initial actions and values of MCP a_i

C.6.3 Functions

C.6.3.1 Return vertical speed for a given altitude layer and aircraft type

	$\dot{z}^z_{D_1}$	$\dot{z}^z_{D_2}$		$\dot{z}^z_{D_1}$	$\dot{z}^z_{D_2}$
FL000	754	803	FL140	1987	2048
FL005	769	822	FL160	2040	2093
FL010	813	880	FL180	2091	2137
FL015	782	1006	FL200	2143	2180
FL020	943	1029	FL220	2193	2223
FL030	1001	1177	FL240	2243	2264
FL040	1020	1206	FL260	2291	2305
FL060	1327	1318	FL280	2338	2344
FL080	1371	1359	FL290	2360	2363
FL100	1882	1958	FL310	3359	2399
FL120	1935	2003			

Table 46 – Vertical speed profiles (BADA) for D_1 and D_2 in [ft·min^{-1}] during descent

Function $\mathcal{F}_{\dot{z}}(\breve{D}, \breve{z})$

Argument	State space	Unit	Description
$\breve{D}$	$\mathbb{D}$	-	Type of aircraft to return the BADA vertical speed value for
$\breve{z}$	$\mathbb{R}_+$	ft	Specific altitude layer to return the BADA vertical speed data for
Variable	**State space**	**Unit**	**Description**
$\breve{\dot{z}}$	$\mathbb{R}_+$	ft·min^{-1}	Assigned BADA vertical speed value
Parameter	**State space**	**Unit**	**Description**
$\dot{z}^z_D$	$\mathbb{R}_{>0}$	ft·min^{-1}	BADA vertical speed value parameters corresponding to the given $\breve{D}$ and $\breve{z}$, see table 46.

if ($\breve{D}$ == D_1) **then**
  if (0 ≤ $\breve{z}$ < 500) **then**
    $\breve{\dot{z}} = \dot{z}_{D_1}^{\text{FL000}} + (\breve{z} - 0) \cdot (\dot{z}_{D_1}^{\text{FL005}} - \dot{z}_{D_1}^{\text{FL000}}) \,/\, (500 - 0)$
  else if (500 ≤ $\breve{z}$ < 1000) **then**
    $\breve{\dot{z}} = \dot{z}_{D_1}^{\text{FL005}} + (\breve{z} - 500) \cdot (\dot{z}_{D_1}^{\text{FL010}} - \dot{z}_{D_1}^{\text{FL005}}) \,/\, (1000 - 500)$
  continue likewise
  else if (31000 ≤ $\breve{z}$) **then**
    $\breve{\dot{z}} = \dot{z}_{D_1}^{\text{FL310}}$
  end if
else if ($\breve{D}$ == D_2) **then**
  if (0 ≤ $\breve{z}$ < 500) **then**
    $\breve{\dot{z}} = \dot{z}_{D_2}^{\text{FL000}} + (\breve{z} - 0) \cdot (\dot{z}_{D_2}^{\text{FL005}} - \dot{z}_{D_2}^{\text{FL000}}) \,/\, (500 - 0)$
  else if (500 ≤ $\breve{z}$ < 1000) **then**
    $\breve{\dot{z}} = \dot{z}_{D_2}^{\text{FL005}} + (\breve{z} - 500) \cdot (\dot{z}_{D_2}^{\text{FL010}} - \dot{z}_{D_2}^{\text{FL005}}) \,/\, (1000 - 500)$
  continue likewise
  else if (31000 ≤ $\breve{z}$) **then**
    $\breve{\dot{z}} = \dot{z}_{D_2}^{\text{FL310}}$
  end if
end if
return: $\breve{\dot{z}}$

Function 24 – Return BADA vertical speed (descent) value for given altitude layer and aircraft type

C.6.3.2 Updating the position and orientation of a_i relative to the multiple significant points

Function 25 models the 'continuous' updating process of the situation awareness of the MCP a_i agent about the position and orientation of a_i relative to all "real" significant points. The function is split up into separate actions that each cover the calculations and associated variables corresponding to a specific type of significant point(s).

Function $\mathcal{F}_{w_P}()$
1: Obtain the required data of $w_{S,k}$ that corresponds to a_i
2: Update the position and orientation of a_i relative to: $w_{S,k}$, $w_{S,H,I}$, $w_{S,H,O}$, w_{IF}, w_{THR}, w_{HST1} and w_{HST2}
3: Update the approach angle of a_i relative to w_{THR}

Function 25 – Updating process of the situation awareness of the MCP a_i agent about the position and orientation of a_i relative to the multiple significant points

Obtain the required data of $w_{S,k}$ that corresponds to a_i
1: $x^{a_i}_{w_{S,k}}$ = *
2: $y^{a_i}_{w_{S,k}}$ = *
3: $\psi^{a_i}_{w_{S,k}}$ = *
4: $\psi^{a_i}_{w_{S,k+1}}$ = *
*obtain data in appendix C.3.2

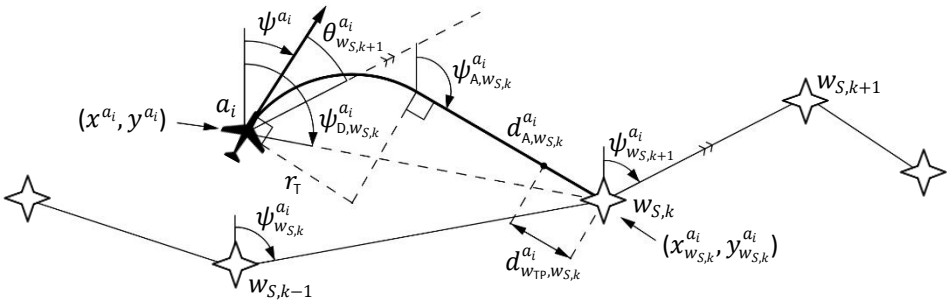


Figure 115 – Visualization of the variables that are used to define the position and orientation of a_i relative to $w_{S,k}$

Update the position and orientation of a_i relative to $w_{S,k}$
1: $\psi^{a_i}_{D,w_{S,k}} = \mathcal{C}^{RAD}_{DEG}(\text{atan2}(y^{a_i}_{w_{S,k}} - y^{a_i}, x^{a_i}_{w_{S,k}} - x^{a_i}))$
2: $\psi^{a_i}_{A,w_{S,k}} = \mathcal{F}_{\psi_A}(\psi^{a_i}, x^{a_i}, y^{a_i}, x^{a_i}_{w_{S,k}}, y^{a_i}_{w_{S,k}}, r_T)$
3: $d^{a_i}_{A,w_{S,k}} = \mathcal{F}_d(\psi^{a_i}, x^{a_i}, y^{a_i}, x^{a_i}_{w_{S,k}}, y^{a_i}_{w_{S,k}}, r_T)$
4: $\theta^{a_i}_{w_{S,k+1}} = \mathcal{F}_\theta(\psi^{a_i}, \psi^{a_i}_{w_{S,k+1}})$
5: $d^{a_i}_{w_{TP},w_{S,k}} = r_T \cdot \tan(½\lvert\theta^{a_i}_{w_{S,k+1}}\rvert)$

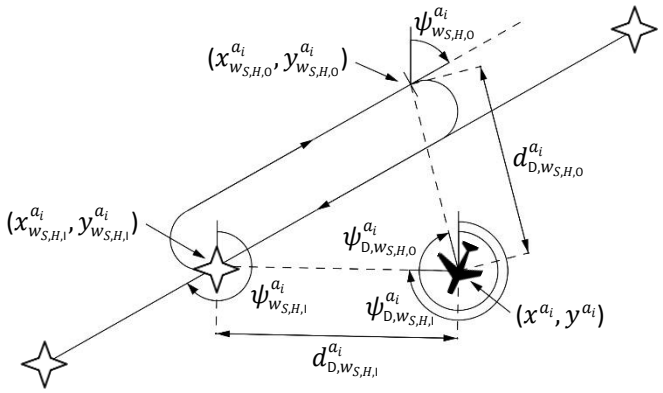


Figure 116 – Visualization of the variables that are used to define the position of a_i relative to $w_{S,H,I}$ and $w_{S,H,O}$

Update the position of a_i relative to $w_{S,H,\text{I}}$ and $w_{S,H,\text{O}}$

1: $d^{a_i}_{\text{D},w_{S,H,\text{I}}} = \sqrt{((x^{a_i}_{w_{S,H,\text{I}}} - x^{a_i})^2 + (y^{a_i}_{w_{S,H,\text{I}}} - y^{a_i})^2)}$
2: $\psi^{a_i}_{\text{D},w_{S,H,\text{I}}} = \mathcal{C}^{\text{RAD}}_{\text{DEG}}(\text{atan2}(y^{a_i}_{w_{S,H,\text{I}}} - y^{a_i}, x^{a_i}_{w_{S,H,\text{I}}} - x^{a_i}))$
3: $d^{a_i}_{\text{D},w_{S,H,\text{O}}} = \sqrt{((x^{a_i}_{w_{S,H,\text{O}}} - x^{a_i})^2 + (y^{a_i}_{w_{S,H,\text{O}}} - y^{a_i})^2)}$
4: $\psi^{a_i}_{\text{D},w_{S,H,\text{O}}} = \mathcal{C}^{\text{RAD}}_{\text{DEG}}(\text{atan2}(y^{a_i}_{w_{S,H,\text{O}}} - y^{a_i}, x^{a_i}_{w_{S,H,\text{O}}} - x^{a_i}))$

Update the position and orientation of a_i relative to w_{IF}

1: $\psi^{a_i}_{\text{D},w_{\text{IF}}} = \mathcal{C}^{\text{RAD}}_{\text{DEG}}(\text{atan2}(y_{w_{\text{IF}}} - y^{a_i}, x_{w_{\text{IF}}} - x^{a_i}))$
2: $\psi^{a_i}_{\text{A},w_{\text{IF}}} = \mathcal{F}_{\psi_{\text{A}}}(\psi^{a_i}, x^{a_i}, y^{a_i}, x_{w_{\text{IF}}}, y_{w_{\text{IF}}}, r_{\text{T}})$
3: $d^{a_i}_{\text{A},w_{\text{IF}}} = \mathcal{F}_{d}(\psi^{a_i}, x^{a_i}, y^{a_i}, x_{w_{\text{IF}}}, y_{w_{\text{IF}}}, r_{\text{T}})$
4: $\theta^{a_i}_{w_{\text{IF}}} = \mathcal{F}_{\theta}(\psi^{a_i}, \psi_{w_{\text{THR}}})$
5: $d^{a_i}_{w_{\text{TP}},w_{\text{IF}}} = r_{\text{T}} \cdot \tan(\tfrac{1}{2}|\theta^{a_i}_{w_{\text{IF}}}|)$

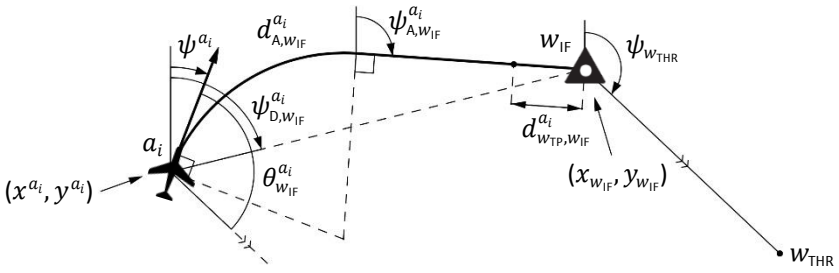


Figure 117 – Visualization of the variables that are used to define the position and orientation of a_i relative to w_{IF}

Update the position and orientation of a_i relative to w_{THR}, w_{HST1} and w_{HST2}

1: $\psi^{a_i}_{\text{D},w_{\text{THR}}} = \mathcal{C}^{\text{RAD}}_{\text{DEG}}(\text{atan2}(y_{w_{\text{THR}}} - y^{a_i}, x_{w_{\text{THR}}} - x^{a_i}))$
2: $d^{a_i}_{\text{D},w_{\text{THR}}} = \sqrt{((x_{w_{\text{THR}}} - x^{a_i})^2 + (y_{w_{\text{THR}}} - y^{a_i})^2)}$
3: $d^{a_i}_{\text{D},w_{\text{HST1}}} = \sqrt{((x_{w_{\text{HST1}}} - x^{a_i})^2 + (y_{w_{\text{HST1}}} - y^{a_i})^2)}$
4: $d^{a_i}_{\text{D},w_{\text{HST2}}} = \sqrt{((x_{w_{\text{HST2}}} - x^{a_i})^2 + (y_{w_{\text{HST2}}} - y^{a_i})^2)}$

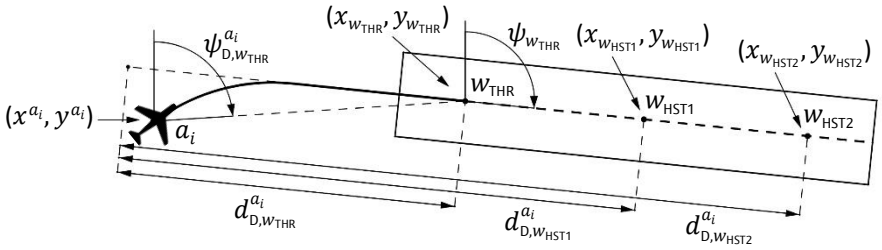


Figure 118 – Visualization of the variables that are used to define the position and orientation of a_i relative to w_{THR}, w_{HST1} and w_{HST2}

Update the approach angle of a_i relative to w_{THR}

Parameter	Value	Unit	Description
$f^{\text{NM}}_{\text{FT}}$	6076.11545	-	[NM] to [ft] conversion factor

1: $\gamma^{a_i}_{\text{APP}} = \tan(z^{a_i} / (d^{a_i}_{\text{D},w_{\text{THR}}} \cdot f^{\text{NM}}_{\text{FT}})) \cdot 180 / \pi$

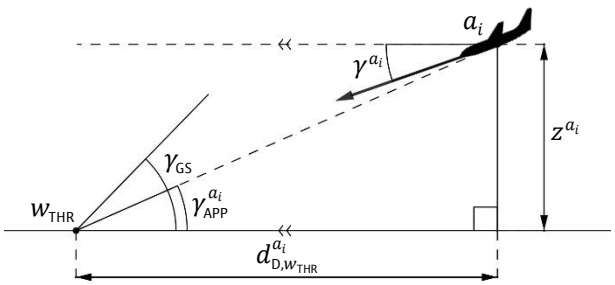


Figure 119 – Visualization of the variables that are used to define the vertical orientation of a_i relative to w_{THR}

C.6.3.3 Updating the waypoint number k^{a_i}

Multiple modelling elements within the MCP a_i agent (i.e. transitions, states, functions) require the agent to update its current waypoint number k^{a_i}. This is generally the case when the aircraft has passed the turn point w_{TP} of waypoint $w_{S,k}$. Function 26 models this simple updating process.

Function $\mathcal{F}_k()$

1: **if** $(k^{a_i} \neq K^{a_i} - 1)$ **then** *//i.e. if a_i is not yet approaching $w_{S,IAF}$*
2: k^{a_i}++
3: $k^{a_i,c}$++
4: **end if**

Function 26 – Update the waypoint number of the aircraft

C.6.4 Statecharts

The functionalities of the MCP a_i agent are modelled with the following five statecharts:

- ⌊*Speed control*⌋[*MCP* a_i] (appendix C.6.4.1)
- ⌊*Altitude control*⌋[*MCP* a_i] (appendix C.6.4.2)
- ⌊*Heading control*⌋[*MCP* a_i] (appendix C.6.4.3)
- ⌊*Interception of fly-by waypoints*⌋[*MCP* a_i] (appendix C.6.4.4)
- ⌊*Glide slope interception*⌋[*MCP* a_i] (appendix C.6.4.5)

All five statecharts monitor a certain condition or (operational) state of a_i. The resulting actions of these five statecharts will eventually shape the three-dimensional trajectory of a_i in the simulation environment.

C.6.4.1 Speed control

The ⌊*Speed control*⌋[*MCP* a_i] statechart (figure 120) models the automated speed control functionality of the aircraft's autopilot system, as controlled by the MCP a_i agent.

⟨constant speed⟩
Indicates that a_i is flying with a constant indicated airspeed.

⟨decelerating⟩
Indicates that a_i is decelerating.

⟨accelerating⟩
Indicates that a_i is accelerating.

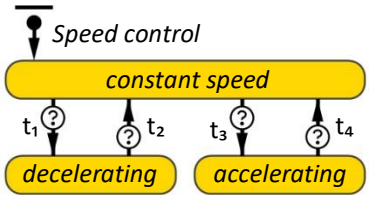


Figure 120 – ⌊Speed control⌋[MCP a_i]

Transition t_1 - t_4
All condition triggered:

Transition	Condition
t_1	$v_{IAS}^{a_i} \geq v_{IAS,S}^{a_i} + v_{IAS,\varepsilon}$
t_2	$v_{IAS}^{a_i} \leq v_{IAS,S}^{a_i}$
t_3	$v_{IAS}^{a_i} \leq v_{IAS,S}^{a_i} - v_{IAS,\varepsilon}$
t_4	$v_{IAS}^{a_i} \geq v_{IAS,S}^{a_i}$

C.6.4.2 Altitude control

The ⌊*Altitude control*⌋[*MCP* a_i] statechart (figure 121) models the automated altitude control functionality of the aircraft's autopilot system, as controlled by the MCP a_i agent.

⟨level flight⟩
Indicates that a_i is flying at a constant altitude.

⟨descending⟩
Indicates that a_i is descending.

⟨climbing⟩
Indicates that a_i is climbing.

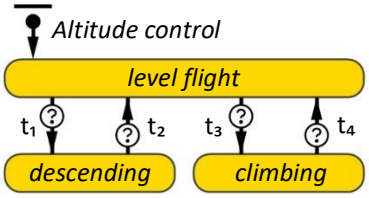


Figure 121 – ⌊Altitude control⌋[MCP a_i]

Transition t_1 - t_4
All condition triggered:

Transition	Condition
t_1	$z^{a_i} \geq z_S^{a_i} + z_\varepsilon$
t_2	$z^{a_i} \leq z_S^{a_i}$
t_3	$z^{a_i} \leq z_S^{a_i} - z_\varepsilon$
t_4	$z^{a_i} \geq z_S^{a_i}$

C.6.4.3 Heading control

The ⌊*Heading control*⌋[*MCP* a_i] statechart (figure 122) models the automated heading control functionality of the aircraft's autopilot system, as controlled by the MCP a_i agent.

⟨straight flight⟩
Indicates that a_i is flying a straight trajectory.

⟨turning flight⟩
Indicates that a_i is no longer flying a straight trajectory.

⟨turning right⟩
Indicates that a_i is performing a turning movement towards the right.

⟨turning left⟩
Indicates that a_i is performing a turning movement towards the left.

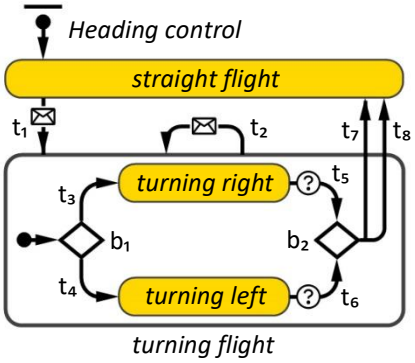
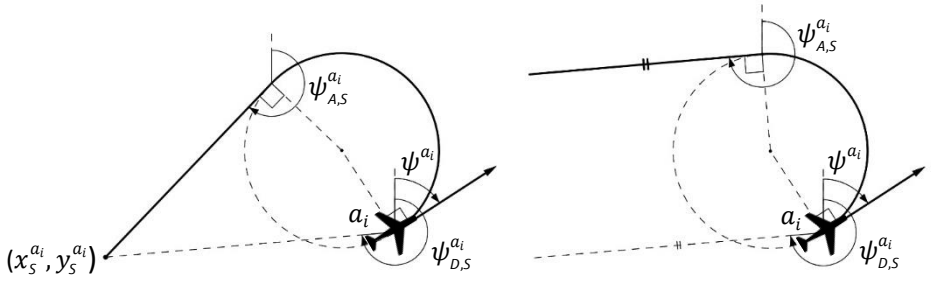


Figure 122 – ⌊Heading control⌋[MCP a_i]

Transition t_1
Triggered in t_1, t_3, t_6 and t_7 of ⌊*Interception of fly-by waypoints*⌋[*MCP* a_i], and in states ⟨*vector outbound STAR instruction*⟩, ⟨*vector inbound STAR instruction*⟩, ⟨*vector outbound merge instruction*⟩, ⟨*vector inbound merge instruction*⟩, ⟨*vector outbound IF instruction*⟩, ⟨*vector inbound IF instruction*⟩, ⟨*vector outbound trombone instruction*⟩, ⟨*vector inbound trombone instruction*⟩, ⟨*holding exit instruction*⟩ of ⌊*Contact with* c^{a_i}⌋[*flight crew* a_i], and in ⟨*vacate runway*⟩⌊*SA operation*⌋[a_i, c]

Transition t_2
This transition is applicable in situations where the MCP a_i agent receives a command to turn to a point or heading while a_i is already turning towards a specific point or heading. Transition t_2 therefore leaves the composite state and re-enters it in order to re-evaluate the conditions of t_3 and t_4, i.e. to decide if it should proceed turning in the same direction or if it should change the direction of the turning movement.
Triggered in states ⟨*vector outbound STAR instruction*⟩, ⟨*vector inbound STAR instruction*⟩, ⟨*vector outbound merge instruction*⟩, ⟨*vector inbound merge instruction*⟩, ⟨*vector outbound IF instruction*⟩, ⟨*vector inbound IF instruction*⟩, ⟨*vector outbound trombone instruction*⟩, ⟨*vector inbound trombone instruction*⟩, ⟨*holding exit instruction*⟩ of ⌊*Contact with* c^{a_i}⌋[*flight crew* a_i]



(a) Turning movement towards a point *(b) Turning movement towards a radial*

Figure 123 – Two different types of turns with the same logic about what direction to turn to and when to stop turning

Transition t_3 - t_6

All condition triggered:

Transition	Condition
t_3	$\mathcal{F}_\theta(\psi^{a_i}, \psi^{a_i}_{D,S}) \geq 0$
t_4	$\mathcal{F}_\theta(\psi^{a_i}, \psi^{a_i}_{D,S}) < 0$
t_5	$\mathcal{F}_\theta(\psi^{a_i}, \psi^{a_i}_{A,S}) \leq 0$
t_6	$\mathcal{F}_\theta(\psi^{a_i}, \psi^{a_i}_{A,S}) \geq 0$

Transition t_7

Related to turns that should end at a specific radial.

- Condition: ⟨*vector outbound STAR*⟩⌊*SA operation*⌋$[a_i, c]$ ∨ ⟨*vector outbound trombone*⟩⌊*SA operation*⌋$[a_i, c]$ ∨ ⟨*vector outbound IF*⟩⌊*SA operation*⌋$[a_i, c]$ ∨ ⟨*vector outbound merge*⟩⌊*SA operation*⌋$[a_i, c]$ ∨ ⟨*vacate runway*⟩⌊*SA operation*⌋$[a_i, c]$
- Action:

1: $\psi^{a_i} = \psi^{a_i}_{A,S}$

Transition t_8

Related to turns that should end at a specific point in space

- Condition: ⟨*STAR*⟩⌊*SA operation*⌋$[a_i, c]$ ∨ ⟨*vector inbound STAR*⟩⌊*SA operation*⌋$[a_i, c]$ ∨ ⟨*vector inbound merge*⟩⌊*SA operation*⌋$[a_i, c]$ ∨ ⟨*vector inbound IF*⟩⌊*SA operation*⌋$[a_i, c]$ ∨ ⟨*vector inbound trombone*⟩⌊*SA operation*⌋$[a_i, c]$ ∨ ⟨*holding*⟩⌊*SA operation*⌋$[a_i, c]$ ∨ ⟨*intermediate approach*⟩⌊*SA operation*⌋$[a_i, c]$
- Action:

1: $\psi^{a_i} = C^{RAD}_{DEG}(\text{atan2}(y^{a_i}_S - y^{a_i}, x^{a_i}_S - x^{a_i}))$

C.6.4.4 Interception of fly-by waypoints

The ⌊*Interception of fly-by waypoints*⌋$[$*MCP* $a_i]$ statechart (figure 124) models the interception of fly-by waypoints by the MCP a_i agent using its turn anticipation capacities.

⟨check for waypoint interception⟩

Represents the state of the MCP a_i agent in which it is continuously monitoring the position of a_i relative to $w_{S,k}$ and w_{IF} to define when it should intercept the respective fly-by waypoint.

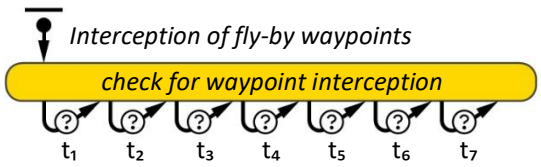


Figure 124 – ⌊Interception of fly-by waypoints⌋[MCP a_i]

The ⌊*Interception of fly-by waypoints*⌋$[$*MCP* $a_i]$ statechart contains seven condition triggered transitions, where each transition is related to a specific operational state and position of a_i. The fly-by waypoint will be intercepted by the MCP a_i when one of the transitions is taken. The conditions in these transitions are based on the anticipation logic that has been modelled. The variables that are used to model this turn anticipation are listed in table 47 and are shown in figure 125. The MCP a_i agent will initiate a turning movement to intercept the fly-by waypoint once $\breve{d}^{a_i}_{A,w} < \breve{d}^{a_i}_{w_{TP},w}$

Variable	State space	Unit	Description
$\breve{d}^{a_i}_{A,w}$	$\mathbb{R}_+$	NM	Distance (“*tangent*”) between a_i and w
$\breve{d}^{a_i}_{w_{TP},w}$	$\mathbb{R}_+$	NM	Distance from w to start intercepting w
$\breve{\theta}^{a_i}_{w_{TP}}$	$[-\pi, \pi]$	rad	Angle between $\breve{\psi}^{a_i}$ and $\breve{\psi}_G$
$\breve{\psi}^{a_i}$	⟨0, 360]	deg	Heading direction with respect to magnetic north
$\breve{\psi}_G$	⟨0, 360]	deg	Magnetic track of the next segment of the route or procedure

Table 47 - Variables that are used to calculate the distance from the fly-by waypoint at which the aircraft should initiate a turning movement to intercept the fly-by waypoint

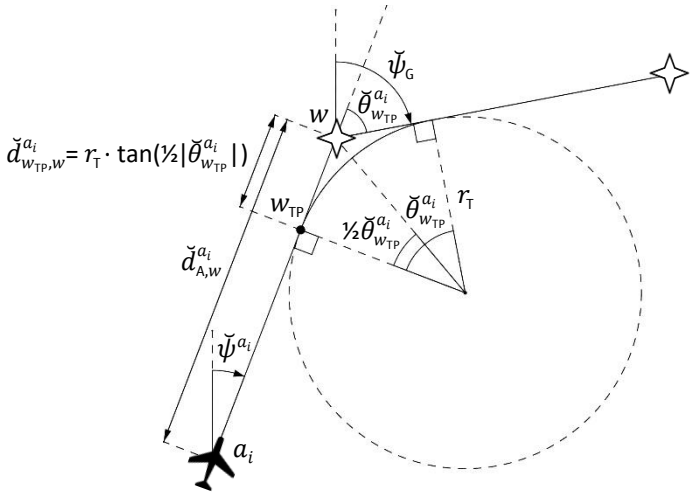


Figure 125 – Variables that are used to calculate the distance from the fly-by waypoint at which the aircraft should initiate a turning movement to intercept the fly-by waypoint

Transition t_1

Represents the situation where a_i is approaching $w_{S,k}$, where $|\theta^{a_i}_{w_{S,k+1}}| > 0$

- Condition: $\langle$*straight flight*$\rangle\lfloor$*Heading control*$\rfloor[$*MCP* $a_i] \wedge (\langle$*STAR*$\rangle\lfloor$*SA operation*$\rfloor[a_i, c] \vee \langle$*vector inbound trombone*$\rangle\lfloor$*SA operation*$\rfloor[a_i, c] \vee \langle$*vector inbound STAR*$\rangle\lfloor$*SA operation*$\rfloor[a_i, c] \vee \langle$*vector inbound merge*$\rangle\lfloor$*SA operation*$\rfloor[a_i, c]) \wedge (d^{a_i}_{A,w_{S,k}} \leq d^{a_i}_{w_{TP},w_{S,k}}) \wedge (k^{a_i} \neq K^{a_i} - 1)$
- Action:

1: $\mathcal{F}_k()$
2: $\mathcal{F}_{w_P}()$
3: $\psi^{a_i}_{A,S} = \psi^{a_i}_{w_{S,k}}$
4: $\psi^{a_i}_{D,S} = \psi^{a_i}_{D,w_{S,k}}$
5: $x^{a_i}_S = x^{a_i}_{w_{S,k}}$
6: $y^{a_i}_S = y^{a_i}_{w_{S,k}}$
7: Trigger t_1 of $\lfloor$*Heading control*$\rfloor[$*MCP* $a_i]$ and t_7, t_8 and t_{25} of $\lfloor$*SA operation*$\rfloor[a_i, c]$

Transition t_2

Represents the situation where a_i is approaching $w_{S,k}$, where $|\theta^{a_i}_{w_{S,k+1}}| \approx 0$

- Condition: $\langle$*straight flight*$\rangle\lfloor$*Heading control*$\rfloor[$*MCP* $a_i] \wedge (\langle$*STAR*$\rangle\lfloor$*SA operation*$\rfloor[a_i, c] \vee \langle$*vector inbound trombone*$\rangle\lfloor$*SA operation*$\rfloor[a_i, c] \vee \langle$*vector inbound STAR*$\rangle\lfloor$*SA operation*$\rfloor[a_i, c] \vee \langle$*vector inbound merge*$\rangle\lfloor$*SA operation*$\rfloor[a_i, c]) \wedge (|\theta^{a_i}_{w_{S,k+1}}| < \dot{\psi}^{a_i}_{\Delta t}) \wedge (d^{a_i}_{A,w_{S,k}} \leq d^{a_i}_{\Delta t}) \wedge (k^{a_i} \neq K^{a_i} - 1)$
- Action:

1: $\mathcal{F}_k()$
2: $\mathcal{F}_{w_P}()$
3: Trigger t_7, t_8 and t_{25} of $\lfloor$*SA operation*$\rfloor[a_i, c]$
4: $\psi^{a_i} = \mathcal{C}^{RAD}_{DEG}(\text{atan2}(y^{a_i}_{w_{S,k}} - y^{a_i}, x^{a_i}_{w_{S,k}} - x^{a_i}))$

Transition t_3

Represents the situation where a_i is approaching w_{IF}, where $|\theta^{a_i}_{w_{IF}}| > 0$

- Condition: $\langle$*straight flight*$\rangle\lfloor$*Heading control*$\rfloor[$*MCP* $a_i] \wedge \langle$*vector inbound IF*$\rangle\lfloor$*SA operation*$\rfloor[a_i, c] \wedge (d^{a_i}_{A,w_{IF}} \leq d^{a_i}_{w_{TP},w_{IF}})$
- Action:

1: $\psi^{a_i}_{A,S} = \psi_{w_{THR}}$
2: $\psi^{a_i}_{D,S} = \psi^{a_i}_{D,w_{THR}}$
3: $x^{a_i}_S = x_{w_{THR}}$
4: $y^{a_i}_S = y_{w_{THR}}$
5: Trigger t_1 of $\lfloor$*Heading control*$\rfloor[$*MCP* $a_i]$ and t_{21} of $\lfloor$*SA operation*$\rfloor[a_i, c]$

Transition t_4

Represents the situation where a_i is approaching w_{IF}, where $|\theta_{w_{\text{IF}}}^{a_i}| \approx 0$

- Condition: ⟨*straight flight*⟩⌊*Heading control*⌋[*MCP* a_i] ∧ ⟨*vector inbound IF*⟩⌊*SA operation*⌋$[a_i, c]$ ∧ $(|\theta_{w_{\text{IF}}}^{a_i}| < \dot{\psi}_{\Delta t}^{a_i}) \wedge (d_{\text{A},w_{\text{IF}}}^{a_i} \leq d_{\Delta t}^{a_i})$
- Action:

1: Trigger t_{21} of ⌊*SA operation*⌋$[a_i, c]$
2: $\psi^{a_i} = C_{\text{DEG}}^{\text{RAD}}(\text{atan2}(y_{w_{\text{IF}}} - y^{a_i}, x_{w_{\text{IF}}} - x^{a_i}))$

Transition t_5

Represents the situation where a_i is approaching $w_{S,\text{IAF}}$

- Condition: ⟨*straight flight*⟩⌊*Heading control*⌋[*MCP* a_i] ∧ ⟨*vector inbound STAR*⟩⌊*SA operation*⌋$[a_i, c]$ ∧ $(d_{\text{A},w_{S,k}}^{a_i} \leq d_{w_{\text{TP}},w_{S,k}}^{a_i}) \wedge (k^{a_i} == K^{a_i} - 1)$
- Action:

1: Trigger t_8 of ⌊*SA operation*⌋$[a_i, c]$

Transition t_6

Represents the situation where a_i is approaching $w_{S,H,\text{I}}$

- Condition: ⟨*straight flight*⟩⌊*Heading control*⌋[*MCP* a_i] ∧ ⟨*holding inbound*⟩⌊*SA operation*⌋$[a_i, c]$ ∧ $(d_{\text{D},w_{S,H,\text{I}}}^{a_i} \leq d_{\Delta t}^{a_i})$
- Action:

1: $\psi_{\text{A,S}}^{a_i} = \psi_{w_{S,H,\text{O}}}^{a_i}$
2: $\psi_{\text{D,S}}^{a_i} = \psi_{\text{D},w_{S,H,\text{O}}}^{a_i}$
3: $x_{\text{S}}^{a_i} = x_{w_{S,H,\text{O}}}^{a_i}$
4: $y_{\text{S}}^{a_i} = y_{w_{S,H,\text{O}}}^{a_i}$
5: Trigger t_1 of ⌊*Heading control*⌋[*MCP* a_i] and t_3 of ⌊*SA operation*⌋$[a_i, c]$

Transition t_7

Represents the situation where a_i is approaching $w_{S,H,\text{O}}$

- Condition: ⟨*straight flight*⟩⌊*Heading control*⌋[*MCP* a_i] ∧ ⟨*holding outbound*⟩⌊*SA operation*⌋$[a_i, c]$ ∧ $(d_{\text{D},w_{S,H,\text{O}}}^{a_i} \leq d_{\Delta t}^{a_i})$
- Action:

1: $\psi_{\text{A,S}}^{a_i} = \psi_{w_{S,H,\text{I}}}^{a_i}$
2: $\psi_{\text{D,S}}^{a_i} = \psi_{\text{D},w_{S,H,\text{I}}}^{a_i}$
3: $x_{\text{S}}^{a_i} = x_{w_{S,H,\text{I}}}^{a_i}$
4: $y_{\text{S}}^{a_i} = y_{w_{S,H,\text{I}}}^{a_i}$
5: Trigger t_1 of ⌊*Heading control*⌋[*MCP* a_i] and t_4 of ⌊*SA operation*⌋$[a_i, c]$

C.6.4.5 Glide slope interception

The ⌊*Glide slope interception*⌋[*MCP* a_i] statechart (figure 126) models the situation awareness of the MCP a_i agent about a_i having intercepted the glideslope of runway 16L at Rome Fiumicino. This situation awareness is therefore also used to mark the transition of a_i from the intermediate approach to the final approach. The MCP a_i agent will initiate a descent when it detects that a_i is intercepting the glideslope.

⟨glide slide not intercepted⟩

Initial state that has only a functional meaning when a_i is flying the intermediate approach, generally in a level flight condition. The state indicates that a_i is approaching the FAP, after which it will start descending.

⟨glide slide intercepted⟩

Indicates that a_i is currently intercepting, or has already intercepted the glide slope. This means that a_i has passed the FAP and that it is currently

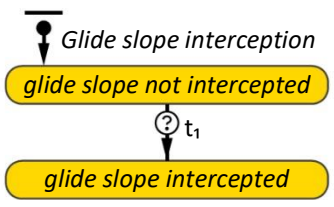


Figure 126 – ⌊Glide slope interception⌋[MCP a_i]

on final approach in a descending state.

- State entry actions:

1: $v_{IAS,S} = v_{AT}$
2: $z_S = 0$

Transition t_1

- Condition: $\langle intermediate\ approach \rangle \lfloor SA\ operation \rfloor [a_i, c] \wedge \langle straight\ flight \rangle \lfloor Heading\ control \rfloor [MCP\ a_i] \wedge \gamma_{APP}^{a_i} \geq \gamma_{GS}$
- Action:

1: Trigger t_{28} of $\lfloor SA\ operation \rfloor [a_i, c]$

C.6.5 Events

C.6.5.1 Aircraft movement and update of flight performance data

Event $\mathcal{E}_{FP}^{a_i}$ models the movement of aircraft a_i in the simulation environment by 'continuously' updating the flight performance variables of aircraft a_i using a set of discretized continuous differential equations. The use of discretized equations allows to optimize the performance of the MonteCarlo simulations. The time step of 0.1 s (Δt = 0.1) is considered to provide an acceptable balance between the performance of the model and the relative amount by which the flight performance variables change.

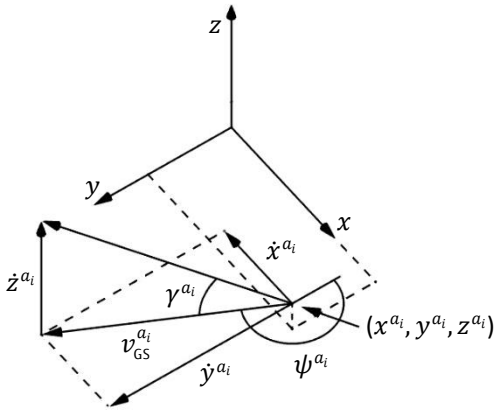


Figure 127 – Specification of the position and the direction of movement of a_i in the simulation environment

- First occurrence time: starts immediately upon generation of a_i
- Recurrence time period: Δt

Event $\mathcal{E}_{FP}^{a_i}$

Parameter	Value	Unit	Description
Δt	0.1	s	Recurrence time period of $\mathcal{E}_{FP}^{a_i}$
f_{NMS}^{FPM}	$2.74298 \cdot 10^{-6}$	-	[ft·min^{-1}] to [NM·s^{-1}] conversion factor
f_{FPM}^{KTS}	101.268592	-	[kt·s^{-1}] to [ft·min^{-1}] conversion factor
f_{NMS}^{KTS}	$2.77778 \cdot 10^{-4}$	-	[kt·s^{-1}] to [NM·s^{-1}] conversion factor

1: $\dot{v}^{a_i} = 0$
2: **if** $(\langle decelerating \rangle \lfloor Speed\ control \rfloor [MCP\ a_i] \wedge \langle level\ flight \rangle \lfloor Altitude\ control \rfloor [MCP\ a_i])$ **then**
3: $\quad \dot{v}^{a_i} = \dot{v}_{L,D}^{a_i}$
4: **else if** $(\langle decelerating \rangle \lfloor Speed\ control \rfloor [MCP\ a_i] \wedge \langle descending \rangle \lfloor Altitude\ control \rfloor [MCP\ a_i] \wedge \neg \langle final\ approach \rangle \lfloor SA\ operation \rfloor [a_i, c])$ **then**
5: $\quad \dot{v}^{a_i} = \dot{v}_{D,D}^{a_i}$
6: **else if** $(\langle accelerating \rangle \lfloor Speed\ control \rfloor [MCP\ a_i] \wedge \langle climbing \rangle \lfloor Altitude\ control \rfloor [MCP\ a_i])$ **then**
7: $\quad \dot{v}^{a_i} = \dot{v}_{C,D}^{a_i}$
8: **else if** $(\langle decelerating \rangle \lfloor Speed\ control \rfloor [MCP\ a_i] \wedge \langle descending \rangle \lfloor Altitude\ control \rfloor [MCP\ a_i] \wedge \langle final\ approach \rangle \lfloor SA\ operation \rfloor [a_i, c])$ **then**

9: $\dot{v}^{a_i} = ((v_{\text{IAS}}^{a_i} - v_{\text{AT}}^{a_i}) / (z^{a_i} - 1000) \cdot \dot{z}^{a_i} / 60) \cdot 1.1$
10: **end if**
11: **if** ($\langle$*deceleration*$\rangle\lfloor$*SA operation*$\rfloor[a_i, c]$) $\wedge$ $\langle$*normal weather condition*$\rangle\lfloor$*Weather condition*$\rfloor$[*Meteo Office*]) **then**
12: $\dot{v}^{a_i} = \dot{v}_{\text{R},m_0}$
13: **else if** ($\langle$*deceleration*$\rangle\lfloor$*SA operation*$\rfloor[a_i, c]$) $\wedge$ $\langle$*bad weather condition*$\rangle\lfloor$*Weather condition*$\rfloor$[*Meteo Office*]) **then**
14: $\dot{v}^{a_i} = \dot{v}_{\text{R},m_1}$
15: **end if**
16: $v_{\text{TAS}}^{a_i} = \mathcal{C}_{\text{TAS}}^{\text{IAS}}(v_{\text{IAS}}^{a_i}, z^{a_i})$
17: $v_{\text{TAS}}^{a_i}$ += $\dot{v}^{a_i} \cdot \Delta t$
18: $v_{\text{IAS}}^{a_i} = \mathcal{C}_{\text{IAS}}^{\text{TAS}}(v_{\text{TAS}}^{a_i}, z^{a_i})$
19: $v_{\text{GS}}^{a_i} = \mathcal{F}_{v_{\text{GS}}}(z^{a_i}, v_{\text{IAS}}^{a_i}, \psi^{a_i})$
20: $\dot{z}^{a_i} = 0$
21: **if** ($\langle$*descending*$\rangle\lfloor$*Altitude control*$\rfloor[$*MCP* $a_i]$) **then**
22: $\dot{z}^{a_i} = -\mathcal{F}_{\dot{z}}(D^{a_i}, z^{a_i})$
23: **else if** ($\langle$*climbing*$\rangle\lfloor$*Altitude control*$\rfloor[$*MCP* $a_i]$) **then**
24: $\dot{z}^{a_i} = \dot{z}_{\text{C}}$
25: **end if**
26: $\gamma^{a_i} = \text{atan}((\dot{z}^{a_i} \cdot f_{\text{NMS}}^{\text{FPM}}) / (v_{\text{GS}}^{a_i} \cdot f_{\text{NMS}}^{\text{KTS}})) \cdot 180 / \pi$
27: **if** ($\langle$*final approach*$\rangle\lfloor$*SA operation*$\rfloor[a_i, c]$) **then**
28: $\gamma^{a_i} = -\gamma_{\text{APP}}^{a_i}$
29: $\dot{z}^{a_i} = v_{\text{GS}}^{a_i} \cdot \tan(\gamma^{a_i} \cdot \pi / 180) \cdot f_{\text{FPM}}^{\text{KTS}}$
30: **end if**
31: $\dot{\psi}^{a_i} = 0$
32: **if** ($\langle$*turning flight*$\rangle\lfloor$*Heading control*$\rfloor[$*MCP* $a_i]$) **then**
33: $\dot{\psi}^{a_i} = (v_{\text{GS}}^{a_i} / r_{\text{T}}) \cdot (180 / \pi) / 3600$
34: **if** ($\langle$*turning right*$\rangle\lfloor$*Heading control*$\rfloor[$*MCP* $a_i]$) **then**
35: ψ^{a_i} += $\dot{\psi}^{a_i} \cdot \Delta t$
36: **else if** ($\langle$*turning left*$\rangle\lfloor$*Heading control*$\rfloor[$*MCP* $a_i]$) **then**
37: ψ^{a_i} -= $\dot{\psi}^{a_i} \cdot \Delta t$
38: **end if**
39: **if** ($\psi^{a_i} > 360$) **then**
40: ψ^{a_i} -= 360
41: **else if** ($\psi^{a_i} \leq 0$) **then**
42: ψ^{a_i} += 360
43: **end if**
44: **end if**
45: $\dot{\psi}_{\Delta t}^{a_i} = |(v_{\text{GS}}^{a_i} / r_{\text{T}}) \cdot (180 / \pi) / 3600 \cdot \Delta t|$
46: $d_{\Delta t}^{a_i} = v_{\text{GS}}^{a_i} \cdot \Delta t / 3600$
47: x^{a_i} += $v_{\text{GS}}^{a_i} \cdot f_{\text{NMS}}^{\text{KTS}} \cdot \Delta t \cdot \cos(\mathcal{C}_{\text{RAD}}^{\text{DEG}}(\psi^{a_i}))$
48: y^{a_i} += $v_{\text{GS}}^{a_i} \cdot f_{\text{NMS}}^{\text{KTS}} \cdot \Delta t \cdot \sin(\mathcal{C}_{\text{RAD}}^{\text{DEG}}(\psi^{a_i}))$
49: z^{a_i} += $\dot{z}^{a_i} \cdot \Delta t / 60$
50: $\mathcal{F}_{w_{\text{P}}}()$

Action 10 – Updating the performance variables of a_i that are used to model the three-dimensional movement of a_i in the simulation environment

C.7 Flight crew

C.7.1 Variables, parameters and sets

Variable	State space	Unit	Description

c^{a_i}	$\mathbb{C}$	-	Active executive controller agent that is responsible for a_i
I^{a_i}	$\mathbb{I}$	-	Instruction that is currently instructed by c^{a_i}
$t_{\text{EXE}}^{a_i}$	$\mathbb{R}_+$	s	Time period required to execute instruction and send read-back to c^{a_i}
$v_{\text{IAS,I}}^{a_i}$	$\mathbb{R}_+$	kt	Instructed indicated airspeed
$x_{\text{I}}^{a_i}$	$\mathbb{R}$	NM	x-position of the point to which the aircraft is instructed to turn
$y_{\text{I}}^{a_i}$	$\mathbb{R}$	NM	y-position of the point to which the aircraft is instructed to turn
$z_{\text{I}}^{a_i}$	$\mathbb{R}_+$	ft	Instructed altitude
$\psi_{\text{A,I}}^{a_i}$	⟨0, 360]	deg	Instructed heading direction ("*tangent*")
$\psi_{\text{D,I}}^{a_i}$	⟨0, 360]	deg	Instructed heading direction ("*direct*")

Table 48 – List of variables that belong to the flight crew a_i agent

Parameter	Value	Unit	Description
$l_{t_{\text{EXE}}}$	3.6	s	Minimum value of $t_{\text{EXE}}^{a_i}$
$\mu_{t_{\text{EXE}}}$	1.5	s	Mean value of $t_{\text{EXE}}^{a_i}$
$\sigma_{t_{\text{EXE}}}$	0.3	s	Standard deviation of $t_{\text{EXE}}^{a_i}$

Table 49 – List of parameters that belong to the flight crew a_i agent

C.7.2 Initial actions and values

1: $v_{\text{IAS,I}}^{a_i} = v_{\text{IAS,0}}^{a_i}$ *
2: $z_{\text{I}}^{a_i} = z_0^{a_i}$ *
3: $c^{a_i} = c_0^{a_i}$ *
4: $\{a_i\} \cup \mathbb{A}$
5: $\{a_i\} \cup \mathbb{A}^c$, where $c = c^{a_i}$
6: $\{a_i\} \cup \mathbb{A}_S$, where $S = S^{a_i}$

* defined in appendix C.5.5.2

Action 11 – Initial actions and values of flight crew a_i agent

C.7.3 Statecharts

C.7.3.1 SA HST

The ⌊*SA HST*⌋[*flight crew* a_i] statechart (figure 128) models the situation awareness of the flight crew a_i agent about the position of a_i on the runway relative to w_{HST1} and w_{HST2}.

⟨not passed w_{HST1} and w_{HST2}⟩
Initial state that has only a useful function after a_i has landed. At touchdown a_i is located somewhere near w_{THR}. From this point a_i starts decelerating in a direction towards w_{HST1} and w_{HST2}.

⟨passed w_{HST1}⟩
Indicates that a_i has already passed, or is currently passing w_{HST1} but not yet w_{HST2}.

- State entry actions:

1: $\psi^{a_i} = \mathcal{C}_{\text{DEG}}^{\text{RAD}}(\text{atan2}(y_{w_{\text{HST2}}} - y^{a_i}, x_{w_{\text{HST2}}} - x^{a_i}))$
2: Trigger t_{33} of ⌊*SA operation*⌋[a_i, c]

⟨passed w_{HST2}⟩
Indicates that a_i has already passed, or is currently passing w_{HST2}.

- State entry actions:

1: Trigger t_{33} of ⌊*SA operation*⌋[a_i, c]

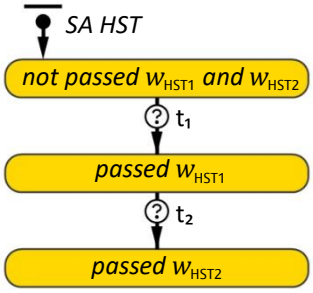


Figure 128 – ⌊SA HST⌋[flight crew a_i]

Transition t_1

- Condition: ⟨*touchdown*⟩⌊*SA operation*⌋[a_i, c] ∧ $d^{a_i}_{\mathrm{D},w_{\mathrm{HST1}}} < d^{a_i}_{\Delta t}$ / 3600

Transition t_2

- Condition: ⟨*touchdown*⟩⌊*SA operation*⌋[a_i, c] ∧ $d_{\mathrm{D},w_{\mathrm{HST2}}} < d^{a_i}_{\Delta t}$ / 3600

C.7.3.2 Contact with c^{a_i}

The ⌊*Contact with* c^{a_i}⌋[*flight crew* a_i] statechart (figure 129) models the communication between the flight crew a_i agent and c^{a_i} from the pilots' perspective.

⟨no contact⟩
Indicates that there is currently no active communication between the flight crew a_i and c^{a_i}.

⟨in contact⟩
Indicates that c^{a_i} has successfully made contact with the flight crew a_i. The pilots are within the ⟨*in contact*⟩ state either receiving I^{a_i} or executing and reading back I^{a_i}.

⟨receiving instruction⟩
Represents the time period in which the flight crew a_i is receiving the incoming I^{a_i} from c^{a_i}.

⟨execution and read-back⟩
Represents the time period in which the flight crew a_i executes and reads-back the received I^{a_i} from c^{a_i}.

All the remaining descriptions of the ⌊*Contact with* c^{a_i}⌋[*flight crew* a_i] states formalise the specific actions of the flight crew a_i agent upon reception of instruction I^{a_i} from c^{a_i}. Most of these actions are related to setting one of the functionalities of the MCP a_i agent. All the remaining states contain the following state exit action (except for ⟨*handover to ARR controller instruction*⟩, ⟨*handover to TWR controller instruction*⟩, ⟨*handover to GND controller instruction*⟩):

1: Trigger t_3 of ⌊*Contact with flight crew* a_i⌋[c^{a_i}]

⟨STAR speed and/or altitude instruction⟩

- State entry actions:

1: $z^{a_i}_{\mathrm{S}} = z^{a_i}_{\mathrm{I}}$
2: $v^{a_i}_{\mathrm{IAS,S}} = v^{a_i}_{\mathrm{IAS,I}}$

⟨vector outbound STAR instruction⟩

- State entry actions:

1: $z^{a_i}_{\mathrm{S}} = z^{a_i}_{\mathrm{I}}$
2: $v^{a_i}_{\mathrm{IAS,S}} = v^{a_i}_{\mathrm{IAS,I}}$
3: $\psi^{a_i}_{\mathrm{D,S}} = \psi^{a_i}_{\mathrm{D,I}}$
4: $\psi^{a_i}_{\mathrm{A,S}} = \psi^{a_i}_{\mathrm{A,I}}$
5: Trigger t_1 and t_2 of ⌊*Heading control*⌋[*MCP* a_i]

⟨vector inbound STAR instruction⟩

- State entry actions:

1: $z^{a_i}_{\mathrm{S}} = z^{a_i}_{\mathrm{I}}$
2: $v^{a_i}_{\mathrm{IAS,S}} = v^{a_i}_{\mathrm{IAS,I}}$
3: $\psi^{a_i}_{\mathrm{D,S}} = \psi^{a_i}_{\mathrm{D},w_{S,k}}$ ($\approx\psi^{a_i}_{\mathrm{D,I}}$)
4: $\psi^{a_i}_{\mathrm{A,S}} = \psi^{a_i}_{\mathrm{A},w_{S,k}}$ ($\approx\psi^{a_i}_{\mathrm{A,I}}$)
5: $x^{a_i}_{\mathrm{S}} = x^{a_i}_{w_{S,k}}$ ($=x^{a_i}_{\mathrm{I}}$)
6: $y^{a_i}_{\mathrm{S}} = x^{a_i}_{w_{S,k}}$ ($=y^{a_i}_{\mathrm{I}}$)
7: Trigger t_1 and t_2 of ⌊*Heading control*⌋[*MCP* a_i]

⟨vector outbound merge instruction⟩
Same contents as ⟨*vector outbound STAR instruction*⟩

⟨vector inbound merge instruction⟩
Same contents as ⟨*vector inbound STAR instruction*⟩

⟨vector outbound IF instruction⟩

- State entry actions:

1: $\psi_{D,S}^{a_i} = \psi_{D,I}^{a_i}$
2: $\psi_{A,S}^{a_i} = \psi_{A,I}^{a_i}$
3: Trigger t_1 and t_2 of ⌊*Heading control*⌋[*MCP* a_i]

⟨vector inbound IF instruction⟩

- State entry actions:

1: $z_S^{a_i} = z_I^{a_i}$
2: $v_{IAS,S}^{a_i} = v_{IAS,I}^{a_i}$
3: $\psi_{D,S}^{a_i} = \psi_{D,w_{IF}}^{a_i} (\approx \psi_{D,I}^{a_i})$
4: $\psi_{A,S}^{a_i} = \psi_{A,w_{IF}}^{a_i} (\approx \psi_{A,I}^{a_i})$
5: $x_S^{a_i} = x_I^{a_i}$
6: $y_S^{a_i} = y_I^{a_i}$
7: Trigger t_1 and t_2 of ⌊*Heading control*⌋[*MCP* a_i]

⟨vector outbound trombone instruction⟩
Same contents as ⟨*vector outbound STAR instruction*⟩

⟨vector inbound trombone instruction⟩
Same contents as ⟨*vector inbound STAR instruction*⟩

⟨handover to ARR controller instruction⟩

- State exit actions:

1: Trigger t_3 of ⌊*Contact with flight crew* a_i⌋[c^{a_i}]
2: $c^{a_i} = c_3$
3: $\{a_i\} \cup \mathbb{A}^c$, where $c == c^{a_i}$

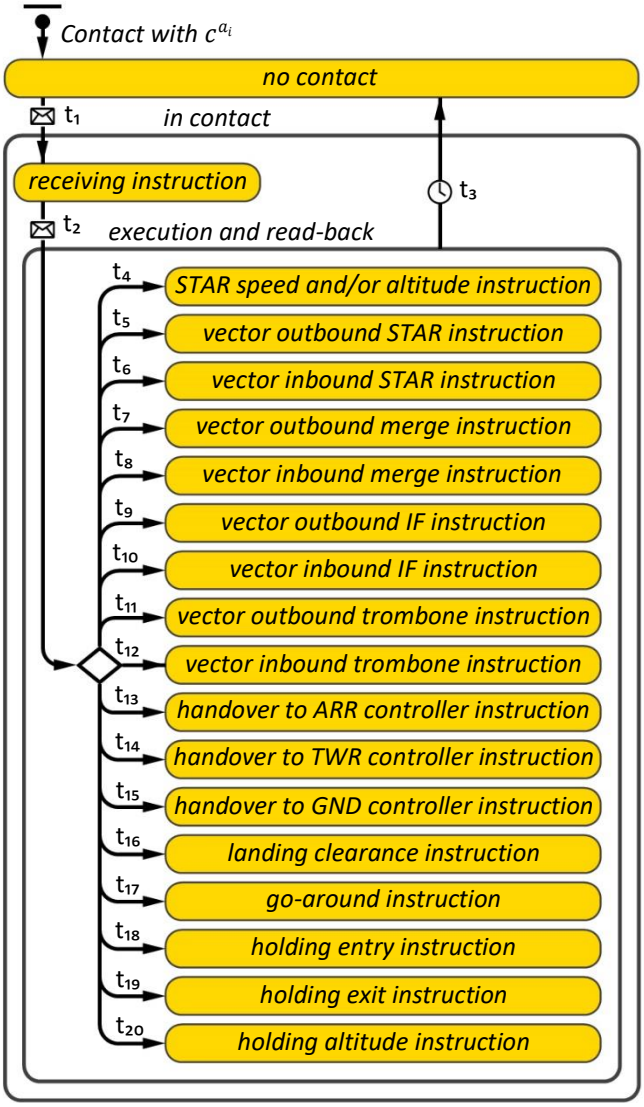


Figure 129 – ⌊Contact with c^{a_i}⌋[flight crew a_i]

⟨handover to TWR controller instruction⟩

- State exit actions:

1: Trigger t_3 of ⌊*Contact with flight crew* a_i⌋[c^{a_i}]
2: $c^{a_i} = c_4$
3: $\{a_i\} \cup \mathbb{A}^c$, where $c == c^{a_i}$

⟨handover to GND controller instruction⟩

- State exit actions:

1: Trigger t_3 of ⌊*Contact with flight crew* a_i⌋[c^{a_i}]
2: $c^{a_i} = c_5$

⟨landing clearance instruction⟩
No specific actions

⟨go-around instruction⟩

- State entry actions:

1: $z_S^{a_i} = z_I^{a_i}$
2: $v_{IAS,S}^{a_i} = v_{IAS,I}^{a_i}$

⟨holding entry instruction⟩

- State entry actions:

1: $z_S^{a_i} = z_I^{a_i}$
2: $v_{IAS,S}^{a_i} = v_{IAS,I}^{a_i}$

⟨holding exit instruction⟩

- State entry actions:

1: $\psi_{D,S}^{a_i} = \psi_{D,w_{S,H,I}}^{a_i}$ (=$\psi_{D,I}^{a_i}$)
2: $\psi_{A,S}^{a_i} = \psi_{w_{S,H,I}}^{a_i}$ (=$\psi_{A,I}^{a_i}$)
3: $x_S^{a_i} = x_{w_{S,H,I}}^{a_i}$ (=$x_I^{a_i}$)
4: $y_S^{a_i} = y_{w_{S,H,I}}^{a_i}$ (=$y_I^{a_i}$)
5: Trigger t_1 and t_2 of ⌊*Heading control*⌋[*MCP* a_i]

⟨holding altitude instruction⟩

- State entry actions:

1: $z_S^{a_i} = z_I^{a_i}$

Transition t_1

Triggered in each simple state in ⟨*sending instruction*⟩⌊*Contact with flight crew* a_i⌋[c], i.e. once c^{a_i} has made contact with a_i

Transition t_2

Triggered in t_2 of ⌊*Contact with flight crew* a_i⌋[c], i.e. once c^{a_i} has finished sending the instruction to a_i

Transition t_3

- Timeout: $t_{EXE}^{a_i}$
- Action:

1: $t_{EXE}^{a_i} = \mathcal{L}(\mu_{t_{EXE}}, \sigma_{t_{EXE}}, l_{t_{EXE}})$

Transitions t_4 - t_{20}

All condition triggered:

Transition	Condition	Transition	Condition	Transition	Condition	Transition	Condition
t_4	$I^{a_i} == I_1$	t_9	$I^{a_i} == I_{6,1} \vee I^{a_i} == I_{6,2}$	t_{14}	$I^{a_i} == I_{11}$	t_{19}	$I^{a_i} == I_{16}$
t_5	$I^{a_i} == I_{2,1} \vee I^{a_i} == I_{2,2}$	t_{10}	$I^{a_i} == I_7$	t_{15}	$I^{a_i} == I_{12}$	t_{20}	$I^{a_i} == I_{17}$
t_6	$I^{a_i} == I_3$	t_{11}	$I^{a_i} == I_8$	t_{16}	$I^{a_i} == I_{13}$		
t_7	$I^{a_i} == I_4$	t_{12}	$I^{a_i} == I_9$	t_{17}	$I^{a_i} == I_{14}$		
t_8	$I^{a_i} == I_5$	t_{13}	$I^{a_i} == I_{10}$	t_{18}	$I^{a_i} == I_{15}$		

C.8 Meteo Office

C.8.1 Statecharts

C.8.1.1 Weather condition

The ⌊*Weather condition*⌋[*Meteo Office*] statechart (figure 130) models the current weather conditions at the airport.

Variable	State space	Unit	Description
b_{m_1}	*Boolean*	-	Boolean that indicates if the bad weather disturbance has already occurred, where the initial value equals '*false*'

⟨normal weather condition⟩

- State entry actions: N/A

⟨bad weather condition⟩

- State entry actions:

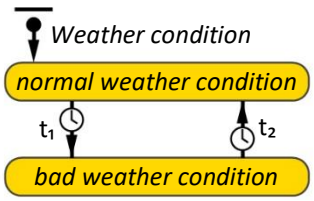


Figure 130 – ⌊Weather condition⌋[Meteo Office]

1: b_{m_1} = true

Transition t_1

Parameter	Value	Unit	Description
t_{m_1}	3600	s	Time point at which the weather conditions will quickly deteriorate, measured relative to t_{SM}

- Timeout: $t_{SM} + t_{m_1}$
- Guard: b_{m_1} == false

Transition t_2

Parameter	Value	Unit	Description
t_{m_0}	7200	s	Duration of the bad weather disturbance, i.e. the time period after which the weather conditions will return to its initial normal state again

- Timeout: t_{m_0}

C.8.1.2 Contact with Supervisor TWR about bad weather condition

The ⌊*Contact with Supervisor TWR about bad weather condition*⌋[*Meteo Office*] statechart (figure 131) models the contact between the Meteo Office and the Supervisor TWR in which the Supervisor TWR is informed about the observed bad weather condition.

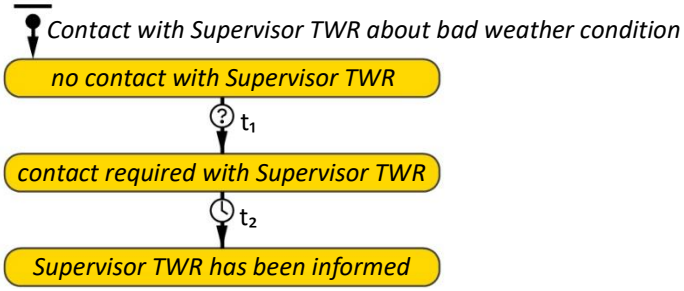


Figure 131 – ⌊Contact with Supervisor TWR about bad weather condition⌋[Meteo Office]

Transition t_1

- Condition: ⟨*bad weather condition*⟩⌊*Weather condition*⌋[*Meteo Office*]

Transition t_2

- Timeout: t_{INF}
- Action:

1: $t_{INF} = \mathcal{L}(\mu_{t_{INF}}, \sigma_{t_{INF}}, l_{t_{INF}})$

C.8.1.3 Contact with Supervisor TWR about normal weather condition

The ⌊*Contact with Supervisor TWR about normal weather condition*⌋[*Meteo Office*] statechart (figure 132) models the contact between the Meteo Office and the Supervisor TWR in which the Supervisor TWR is informed about the observed normalized weather condition.

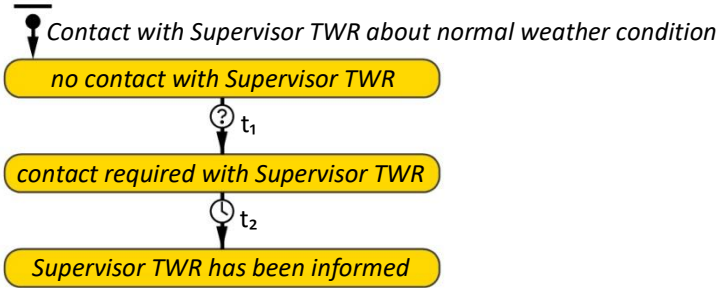


Figure 132 – ⌊Contact with Supervisor TWR about normal weather condition⌋[Meteo Office]

Transition t_1

- Condition: ⟨*normal weather condition*⟩⌊*Weather condition*⌋[*Meteo Office*] ∧
  ⟨*Supervisor TWR has been informed*⟩⌊*Contact with Supervisor TWR about bad weather condition*⌋[*Meteo Office*]

Transition t_2

- Timeout: t_{INF}
- Action:

1: $t_{INF} = \mathcal{L}(\mu_{t_{INF}}, \sigma_{t_{INF}}, l_{t_{INF}})$

C.9 Supervisor TWR

C.9.1 Statecharts

C.9.1.1 Contact with Meteo Office about bad weather condition

The ⌊*Contact with Meteo Office about bad weather condition*⌋[*Supervisor TWR*] statechart (figure 133) models the contact between the Supervisor TWR and the Meteo Office in which the Supervisor TWR agent is informed about deteriorated weather conditions at the airport.

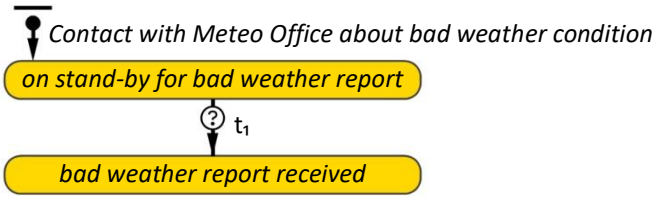


Figure 133 – ⌊Contact with Meteo Office about bad weather condition⌋[Supervisor TWR]

Transition t_1

- Condition: ⟨*Supervisor TWR has been informed*⟩⌊*Contact with Supervisor TWR about bad weather condition*⌋
  [*Meteo Office*]

C.9.1.2 Contact with Meteo Office about normal weather condition

The ⌊*Contact with Meteo Office about normal weather condition*⌋[*Supervisor TWR*] statechart (figure 134) models the contact between the Supervisor TWR and the Meteo Office in which the Supervisor TWR agent is informed about normalized weather conditions at the airport.

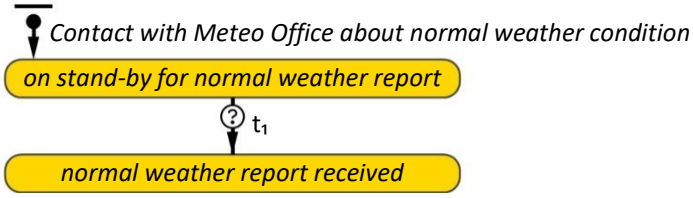


Figure 134 – ⌊Contact with Meteo Office about normal weather condition⌋[Supervisor TWR]

Transition t_1

- Condition: ⟨*Supervisor TWR has been informed*⟩⌊*Contact with Supervisor TWR about normal weather condition*⌋
  [*Meteo Office*]

C.9.1.3 Contact with TWR controller about reduced capacity

The ⌊*Contact with TWR controller about reduced capacity*⌋[*Supervisor TWR*] statechart (figure 135) models the contact between the Supervisor TWR and the TWR controller in which the TWR controller is informed about the reduced runway capacity.

Transition t_1

- Condition: ⟨*bad weather report received*⟩⌊*Contact with Meteo Office about bad weather condition*⌋[*Supervisor TWR*]

Transition t_2

- Timeout: t_{INF}
- Action:

1: $t_{INF} = \mathcal{L}(\mu_{t_{INF}}, \sigma_{t_{INF}}, l_{t_{INF}})$

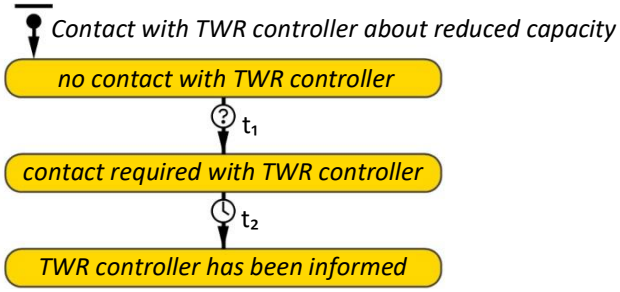


Figure 135 – ⌊Contact with TWR controller about reduced capacity⌋[Supervisor TWR]

C.9.1.4 Contact with TWR controller about recovered capacity

The ⌊*Contact with TWR controller about recovered capacity*⌋[*Supervisor TWR*] statechart (figure 136) models the contact between the Supervisor TWR and the TWR controller in which the TWR controller is informed about the recovered runway capacity.

Transition t_1

- Condition: ⟨*normal weather report received*⟩⌊*Contact with Meteo Office about normal weather condition*⌋[*Supervisor TWR*]

Transition t_2

- Timeout: t_{INF}
- Action:

1: $t_{INF} = \mathcal{L}(\mu_{t_{INF}}, \sigma_{t_{INF}}, l_{t_{INF}})$

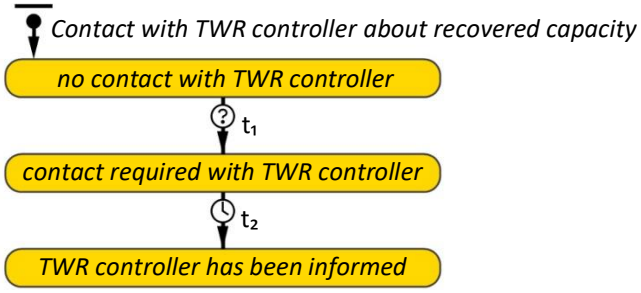


Figure 136 – ⌊Contact with TWR controller about recovered capacity⌋[Supervisor TWR]

C.9.1.5 Contact with Supervisor APP about reduced capacity

The ⌊*Contact with Supervisor APP about reduced capacity*⌋[*Supervisor TWR*] statechart (figure 137) models the contact between the Supervisor TWR and the Supervisor APP in which the Supervisor APP is informed about the reduced runway capacity.

Transition t_1

- Condition: ⟨*TWR controller has been informed*⟩⌊*Contact with TWR controller about reduced capacity*⌋[*Supervisor TWR*]

Transition t_2

- Timeout: t_{INF}
- Action:

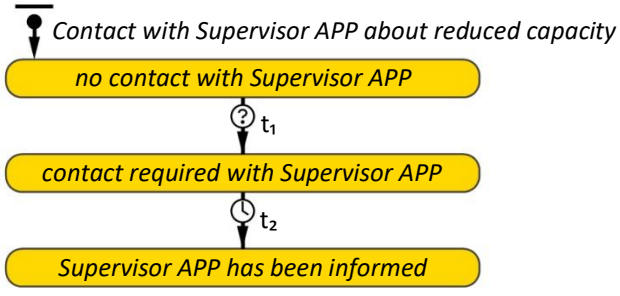


Figure 137 – ⌊Contact with Supervisor APP about reduced capacity⌋[Supervisor TWR]

1: $t_{INF} = \mathcal{L}(\mu_{t_{INF}}, \sigma_{t_{INF}}, l_{t_{INF}})$

C.9.1.6 Contact with Supervisor APP about recovered capacity

The ⌊*Contact with Supervisor APP about recovered capacity*⌋[*Supervisor TWR*] statechart (figure 138) models the contact between the Supervisor TWR and the Supervisor APP in which the Supervisor APP is informed about the recovered runway capacity.

Transition t_1

- Condition: ⟨*TWR controller has been informed*⟩⌊*Contact with TWR controller about recovered capacity*⌋ [*Supervisor TWR*]

Transition t_2

- Timeout: t_{INF}
- Action:

1: $t_{INF} = \mathcal{L}(\mu_{t_{INF}}, \sigma_{t_{INF}}, l_{t_{INF}})$

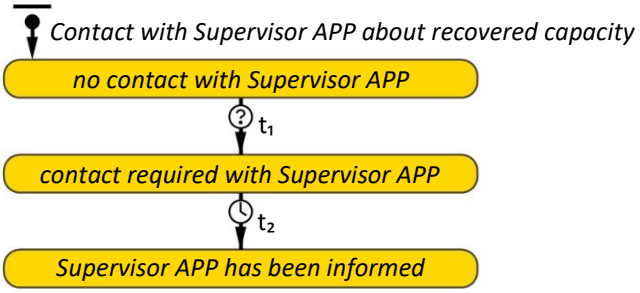


Figure 138 – ⌊Contact with Supervisor APP about recovered capacity⌋[Supervisor TWR]

C.10 Supervisor APP

C.10.1 Variables and sets

Variable	State space	Unit	Description
Q_{c,C_D}	$\mathbb{N}_0$	-	Number of controllers that have been informed about the reduced capacity, initially zero
Q_{c,C_I}	$\mathbb{N}_0$	-	Number of controllers that have been informed about the recovered capacity, initially zero

Table 50 – List of variables that belong to the Supervisor APP

Set	Description
$\mathbb{C}_{C_D}$	Collection of controllers where the index of the elements represents the sequence in which the respective controller is or will be informed about the reduced capacity, i.e. the first element ($\mathbb{C}_{C_D}[0]$) is informed first
$\mathbb{C}_{C_I}$	Collection of controllers where the index of the elements represents the sequence in which the respective controller is or will be informed about the recovered capacity, i.e. the first element ($\mathbb{C}_{C_I}[0]$) is informed first

Table 51 – List of sets that belong to the Supervisor APP

C.10.2 Statecharts

C.10.2.1 Contact with Supervisor TWR about reduced capacity

The ⌊*Contact with Supervisor TWR about reduced capacity*⌋[*Supervisor APP*] statechart (figure 139) models the contact between the Supervisor APP and the Supervisor TWR in which the Supervisor APP is informed about the reduced runway capacity.

⟨on stand-by for reduced capacity update⟩

- State entry actions: N/A

⟨reduced capacity update received⟩

- State entry actions:

1: $\{c_3\} \cup \mathbb{C}_{C_D}$
2: **if** $(|\mathbb{A}^{c_1}| > |\mathbb{A}^{c_2}|)$ **then**
3: $\{c_1\} \cup \mathbb{C}_{C_D}$
4: $\{c_2\} \cup \mathbb{C}_{C_D}$
5: **else**
6: $\{c_2\} \cup \mathbb{C}_{C_D}$
7: $\{c_1\} \cup \mathbb{C}_{C_D}$
8: **end if**
9: $\{c_0\} \cup \mathbb{C}_{C_D}$

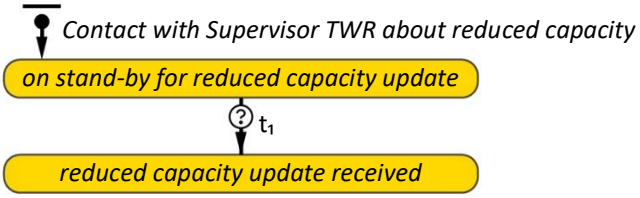


Figure 139 – [Contact with Supervisor TWR about reduced capacity][Supervisor APP]

Transition t_1

- Condition: ⟨*Supervisor APP has been informed*⟩[*Contact with Supervisor APP about reduced capacity*] [*Supervisor TWR*]

C.10.2.2 Contact with Supervisor TWR about recovered capacity

The [*Contact with Supervisor TWR about recovered capacity*][*Supervisor APP*] statechart (figure 140) models the contact between the Supervisor APP and the Supervisor TWR in which the Supervisor APP is informed about the recovered runway capacity.

⟨on stand-by for recovered capacity update⟩

- State entry actions: N/A

⟨recovered capacity update received⟩

- State entry actions:

1: $\{c_3\} \cup \mathbb{C}_{C_I}$
2: **if** $(|\mathbb{A}^{c_1}| > |\mathbb{A}^{c_2}|)$ **then**
3: $\{c_1\} \cup \mathbb{C}_{C_I}$
4: $\{c_2\} \cup \mathbb{C}_{C_I}$
5: **else**
6: $\{c_2\} \cup \mathbb{C}_{C_I}$
7: $\{c_1\} \cup \mathbb{C}_{C_I}$
8: **end if**
9: $\{c_0\} \cup \mathbb{C}_{C_I}$

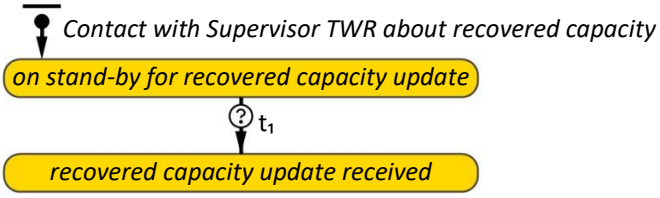


Figure 140 – [Contact with Supervisor TWR about recovered capacity][Supervisor APP]

Transition t_1

- Condition: ⟨*Supervisor APP has been informed*⟩[*Contact with Supervisor APP about recovered capacity*] [*Supervisor TWR*]

C.10.2.3 Contact with TNW controller about reduced capacity

The [*Contact with TNW controller about reduced capacity*][*Supervisor APP*] statechart (figure 141) models the contact between the Supervisor APP and the TNW controller in which the TNW controller is informed about a reduced throughput capacity.

⟨no contact with TNW controller⟩

- State entry actions: N/A

⟨contact required with TNW controller⟩

- State entry actions: N/A

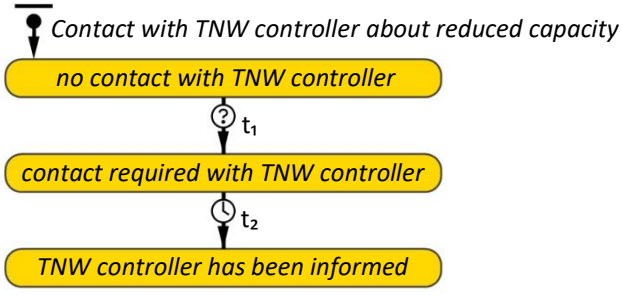


Figure 141 – [Contact with TNW controller about reduced capacity][Supervisor APP]

⟨TNW controller has been informed⟩

- State entry actions:

1: Q_{c,C_D}++

Transition t_1

- Condition: ⟨*reduced capacity update received*⟩⌊*Contact with Supervisor TWR about reduced capacity*⌋ [*Supervisor APP*] ∧ $\mathbb{C}_{C_D}[Q_{c,C_D}]$ == c_1

Transition t_2

- Timeout: t_{INF}
- Action:

1: $t_{INF} = \mathcal{L}(\mu_{t_{INF}}, \sigma_{t_{INF}}, l_{t_{INF}})$

C.10.2.4 Contact with TNW controller about recovered capacity

The ⌊*Contact with TNW controller about recovered capacity*⌋[*Supervisor APP*] statechart (figure 141) models the contact between the Supervisor APP and the TNW controller in which the TNW controller is informed about a recovered throughput capacity.

⟨no contact with TNW controller⟩

- State entry actions: N/A

⟨contact required with TNW controller⟩

- State entry actions: N/A

⟨TNW controller has been informed⟩

- State entry actions:

1: Q_{c,C_I}++

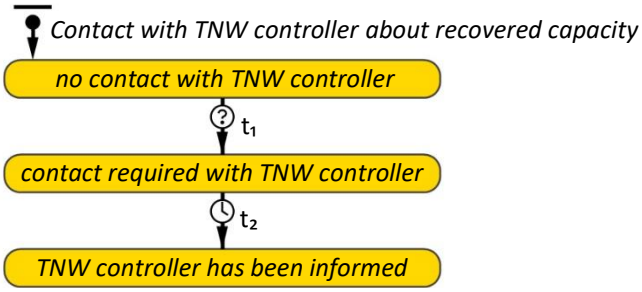


Figure 142 – ⌊Contact with TNW controller about recovered capacity⌋[Supervisor APP]

Transition t_1

- Condition: ⟨*recovered capacity update received*⟩⌊*Contact with Supervisor TWR about recovered capacity*⌋ [*Supervisor APP*] ∧ $\mathbb{C}_{C_I}[Q_{c,C_I}]$ == c_1

Transition t_2

- Timeout: t_{INF}
- Action:

1: $t_{INF} = \mathcal{L}(\mu_{t_{INF}}, \sigma_{t_{INF}}, l_{t_{INF}})$

C.10.2.5 Contact with TNE controller about reduced capacity

The [*Contact with TNE controller about reduced capacity*][*Supervisor APP*] statechart has the same structure, purpose and functioning as the [*Contact with TNW controller about reduced capacity*][*Supervisor APP*] statechart. Note that the last variable of transition t_1 (i.e. c_1) should be changed accordingly.

C.10.2.6 Contact with TNE controller about recovered capacity

The [*Contact with TNE controller about recovered capacity*][*Supervisor APP*] statechart has the same structure, purpose and functioning as the [*Contact with TNW controller about recovered capacity*][*Supervisor APP*] statechart. Note that the last variable of transition t_1 (i.e. c_1) should be changed accordingly.

C.10.2.7 Contact with ARR controller about reduced capacity

The [*Contact with ARR controller about reduced capacity*][*Supervisor APP*] statechart has the same structure, purpose and functioning as the [*Contact with TNW controller about reduced capacity*][*Supervisor APP*] statechart. Note that the last variable of transition t_1 (i.e. c_1) should be changed accordingly.

C.10.2.8 Contact with ARR controller about recovered capacity

The [*Contact with ARR controller about recovered capacity*][*Supervisor APP*] statechart has the same structure, purpose and functioning as the [*Contact with TNW controller about recovered capacity*][*Supervisor APP*] statechart. Note that the last variable of transition t_1 (i.e. c_1) should be changed accordingly.

C.10.2.9 Contact with feeder controller about reduced capacity

The [*Contact with feeder controller about reduced capacity*][*Supervisor APP*] statechart has the same structure, purpose and functioning as the [*Contact with TNW controller about reduced capacity*][*Supervisor APP*] statechart. Note that the last variable of transition t_1 (i.e. c_1) should be changed accordingly.

C.10.2.10 Contact with feeder controller about recovered capacity

The [*Contact with feeder controller about recovered capacity*][*Supervisor APP*] statechart has the same structure, purpose and functioning as the [*Contact with TNW controller about recovered capacity*][*Supervisor APP*] statechart. Note that the last variable of transition t_1 (i.e. c_1) should be changed accordingly.

Dedicated to innovation in aerospace

Netherlands Aerospace Centre

NLR is a leading international research centre for aerospace. Bolstered by its multidisciplinary expertise and unrivalled research facilities, NLR provides innovative and integral solutions for the complex challenges in the aerospace sector.

NLR's activities span the full spectrum of Research Development Test & Evaluation (RDT & E). Given NLR's specialist knowledge and facilities, companies turn to NLR for validation, verification, qualification, simulation and evaluation. NLR thereby bridges the gap between research and practical applications, while working for both government and industry at home and abroad.

NLR stands for practical and innovative solutions, technical expertise and a long-term design vision. This allows NLR's cutting edge technology to find its way into successful aerospace programs of OEMs, including Airbus, Embraer and Pilatus. NLR contributes to (military) programs, such as ESA's IXV re-entry vehicle, the F-35, the Apache helicopter, and European programs, including SESAR and Clean Sky 2.

Founded in 1919, and employing some 600 people, NLR achieved a turnover of 76 million euros in 2017, of which 81% derived from contract research, and the remaining from government funds.

For more information visit: **www.nlr.org**

For more information visit: **www.nlr.org**

Postal address
PO Box 90592
1006 BM Amsterdam, The Netherlands
e) info@nlr.nl **i)** www.nlr.org

NLR Amsterdam
Anthony Fokkerweg 2
1059 CM Amsterdam, The Netherlands
p) +31 88 511 3113

NLR Marknesse
Voorsterweg 31
8316 PR Marknesse, The Netherlands
p) +31 88 511 4444