TOWARD A GRACEFUL DEGRADATION OF AIR TRAFFIC MANAGEMENT SYSTEMS

A Thesis
Presented to
The Academic Faculty

by

Maxime Gariel

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
The Daniel Guggenheim School of Aerospace Engineering

Georgia Institute of Technology
August 2010

TOWARD A GRACEFUL DEGRADATION OF AIR TRAFFIC MANAGEMENT SYSTEMS

Approved by:

Professor Eric Feron
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA, USA
The Daniel Guggenheim School of
Aerospace Engineering
Georgia Institute of Technology

Professor Karen Feigh
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA, USA
The Daniel Guggenheim School of
Aerospace Engineering
Georgia Institute of Technology

Professor John-Paul Clarke
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA, USA
The Daniel Guggenheim School of
Aerospace Engineering
Georgia Institute of Technology

Professor Vitali Volovoi
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA, USA
The Daniel Guggenheim School of
Aerospace Engineering
Georgia Institute of Technology

Professor Daniel Delahaye
Département de Mathématiques et
d'Informatique
Ecole Nationale de l'Aviation Civile
Toulouse, France
The Daniel Guggenheim School of
Aerospace Engineering
Georgia Institute of Technology

Date Approved: June 5, 2010

ACKNOWLEDGEMENTS

I would like to thank Eric Feron for taking me into the adventure of a Ph.D.. I really enjoyed our formal and mostly informal meetings and interactions. Thank you Eric for our friendship, for trusting me and believing in me. You made this work possible.

I would like to thank Daniel Delahaye for his advices, for hosting me in France at ENAC, and for our friendship. I would like to thank John-Paul Clarke, Karen Feigh, and Vitali Voilovoi for being on my thesis committee, for their advices, and for reviewing this document. I would like to thank Ashok Srivastava for his precious advices and for giving me the great opportunity to spend a summer working at NASA Ames.

I would like to thank Thales ATM Inc, and NASA (Grant NNX08AY52A), for sponsoring my thesis.

I sincerely thank my parents for everything they have done for me, especially for their unconditionnal support and understanding. I want to thank all my friends in Atlanta who made the four years I spent in Atlanta a real pleasure. Starring, by alphabetical order, Adan, Alej I, Alex P, Allen, Bernie, Claus, Erwan, Girish, Greg, GT skydive members, Laci, Lau, Lou, Mauro, Mike, Pasha, Rene, Samsung, and Yoko. I'll stop the list here, but the credits could be much longer. I apologize to my friends I didn't mention above, but be sure you made my time here a damn good time. Special thanks to Alex V, for bearing with me and being such a great person.

Thanks for reading those acknoledgments, now, please enjoy the remainder of the thesis.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

GLOSSARY

ACAS Airborn Collision Avoidance System, p. 151.

ARTCC Air Route Traffic Control Center, p. 5.

ASAS Airborne Separation Assurance System, p. 152.

ATC Air Traffic Control, p. 1.

ATCs Air Traffic Controllers, p. 2.

ATM Air Traffic Management, p. 1.

ATMS Air Traffic Management System, p. 1.

C-ATM Co-operative Air Traffic Management, p. 151.

CNS Communication Navigation and Surveillance, p. 1.

CONOPS CONcepts of OPerationS, p. 152.

CTAS Center TRACON Automation System, p. 2.

DA Descent Advisor, p. 2.

FAA Federal Aviation Administration, p. 6.

FAST Final Approach Spacing Tool, p. 2.

FSM Flight Schedule Monitor, p. 3.

GDP Ground Delay Program, p. 2.

GLPK Gnu Linear Programming Kit.

IFR Instrument Flight Rules, p. 7.

ILS Instrument Landing System, p. 30.

IMC Instrument Meteorological Conditions, p. 70.

JPDO Joint Planning and Development Office, p. 1.

LCS Longest Common Subsequence, p. 131.

McTMA Multi-Center Traffic Management Advisor, p. 2.

NAS National Airspace System, p. 8.

NCT Northern California TRACON, p. 41.

NextGen Next Generation Air Traffic System, p. 1.

OAK Oakland International Airport, p. 41.

POD Proper Orthogonal Decomposition, p. 131.

PSR Primary Surveillance Radar, p. 68.

RNP Required Navigation Performance, p. 36.

RQ Research Question, p. 10.

SCT Southern California TRACON, p. 4.

SESAR Single European Sky ATM Research, p. 1.

SFO San Francisco International Airport, p. 13.

SID Standard Instrument Departure, p. 41.

SJC San Jose International Airport, p. 41.

SSR Secondary Surveillance Radar, p. 27.

STAR Standard Terminal Arrival Route, p. 41.

TBS Time Based Separation, p. 152.

TCAS Traffic Collision Avoidance System, p. 6.

TFM Traffic Flow Management, p. 1.

TFMs Traffic Flow Managers, p. 2.

TIS Traffic Information System, p. 36.

TMA Terminal Radar Approach Control, p. 2.

TRACON Terminal Radar Approach Control, p. 2.

VFR Visual Flight Rules, p. 7.

CHAPTER I

INTRODUCTION

1.1 Air Traffic Management

Air Traffic Management (ATM) is the service ensuring the safe, efficient and expeditious movement of aircraft in the airspace. The objectives of ATM are to ensure safe separation between aircraft to prevent collisions, to organize and expedite the flows of traffic, and to provide awareness and support to pilots about other aircraft and potential threats. ATM comprises two main components: Air Traffic Control (ATC) and Traffic Flow Management (TFM). ATC is the tactical safety separation service, that prevents collision between aircraft, and between aircraft, terrain and obstructions. ATC time window ranges from 0 minute to approximatively 15 minutes. TFM allocates traffic flows to scarce capacity resources (e.g. it meters arrival at capacity constrained airports). TFM time window ranges from 15 minutes to days, weeks or even months, for planning purposes.

Air Traffic Management relies on several layers of technology supporting three essential functions: Communication, Navigation and Surveillance (CNS). Communication is the capability to exchange information between aircraft, pilots, and air traffic controllers. Navigation is the capability of aircraft to follow a given trajectory within prescribed accuracy limits. Surveillance is the capability of air traffic controllers to monitor the positions of the aircraft relative to each other and relative to obstacles or restricted airspaces.

In Europe, in the United States and around the world, current Air Traffic Management Systems (ATMS) are aging, and large projects are being developed to handle the fast-growing traffic demand. The resulting concepts of operations were introduced by the Joint Planning and Development Office (JPDO) [69] and by the Eurocontrol SESAR Consortium [32]. The JPDO proposes the NEXT GENeration (NextGen) [70] air transportation systems for US operations, and the SESAR Consortium proposes the Single European Sky ATM Research (SESAR) [123, 124, 125, 126] project for European operations. And indeed, the

air traffic management system is left with no choice but to leverage the concurrent advent of digital communication technology, satellite-based navigation and overall improvements of available instrumentation to handle the demand and reduce system inefficiencies. However, the obligation for the ATMS to maintain very high reliability and safety levels implies that such new system technologies can be implemented only if they lead to a system with equal or better safety characteristics than currently available. While system safety includes the ability for the system to operate well under nominal conditions, it is also concerned with off-nominal system behaviors, whereby operations are expected to still remain accident-free for all known failure modes of the system.

1.2 Automation in Air Traffic Management Systems

Since the advent of the radar, automation has helped air traffic controllers (ATCs) and traffic flow managers (TFMs) to obtain aircraft position, to display information about the flights on the controller's screen, to manage flight plans, to deconflict aircraft, to strategically plan operations, etc. The objective has always been to improve the safety and make the work of the controllers more efficient. Tools such as the Center TRACON Automation System (CTAS) [29, 95] address this purpose and provide a large array of automation tools for planning and controlling arriving and departing traffic. TRACON stands for terminal radar approach control. CTAS includes various tools and control techniques to reduce traffic congestion. The Traffic Management Advisor (TMA) [97, 98] and Multi-Center Traffic Management Advisor (McTMA) [96, 18] are used to control arriving aircraft that enter the Center from an adjacent Center or depart from feeder airports within the Center. Those systems have already proven their capabilities [63, 34]. At the arrival level, the Descent Advisor (DA) [49] and the Final Approach Spacing Tool (FAST) [19, 20] enable more efficient operations.

Finally, ground operations and efficient runway operations planning can also improve landing capacities [3]. On the departure side, Ground Delay Programs (GDP) [6] are used to decrease the rate of incoming flights into a destination airport by delaying takeoffs, when it is projected that arrival demand will exceed arrival capacity. The Fligh Schedule Monitor

(FSM) [5] provides users with a real time estimate of the number of expected flights arriving or departing to the 800 largerst airports in the US. The aim of those different layers of traffic control is to maximize the use of available resources by ensuring smooth aircraft flows and by avoiding terminal area congestion. Before their implementation, the most tacticals of those systems requiring high automation, those involving safety, need to be extensively tested and proven safe [22, 50, 83, 138].

One of the downsides of automation can be its poor implementation. Clumsy automation has resulted in many incidents [122] some of them fatal. In 1991, writing about air transport operations, Billings [9] formulated the following question:

> Given the level of automation now available in transport aircraft, what should be the role of the pilots? Present [1991, editor's note] automation makes it easy and even boring for the pilot to operate the aircraft under nominal conditions, but more difficult to operate in contingencies. Yet pilots have demonstrated throughout the history of aviation that they have little difficulty in operating under normal circumstances. Should aircraft automation perhaps be designed instead to provide somewhat less assistance under nominal conditions and considerably more assistance under more difficult circumstances?

Automation in air traffic control is likely to follow the same pattern, even though aircraft automation started earlier. More than 18 years after Billings stated these questions, they are still current and real issues for air traffic control. The following states Billings' view of human-centered automation:

> The pilot could call on automation modules to assist in problem diagnosis, in evaluation of available alternatives, and in execution of alternative plans. The automation would serve as the pilot's assistant, providing and calculating data, **watching for the unexpected**, and keeping track of resources and rate of expenditure.

This statement is also valid for air traffic controllers. Current systems tend to provide controllers with decision help, conflict detection and conflict resolution advisories. Watching

for unexpected events is not only useful to help pilot and controllers, but is also a prerequisite to ensure a safe degradation of automation in the event of a failure.

1.3 Failures in ATMS

The range of failures that can affect the ATMS is very broad, because of the complexity of the system. The following presents a list of failures that have occurred in current ATMS.

- **Ground infrastructures:** Buildings and facilities are regularly affected by unexpected exogenous factors such as fires: the Southern California TRACON (SCT) had to be evacuated in 2003 because of wildfires [111] threatening the facility. Tens of aircraft were left in the sky without controllers to monitor and direct them.

- **ATM computational infrastructure:** Computers and network systems form the backbone of the air transportation system information infrastructure. Computers are not exempt from failures, whether the failures involve hardware (motherboard and wiring) or software (incomplete functional requirements or erroneous software implementation). Even though ATM providers make sure that the infrastructure is redundant and robust to its own failures, it still happens that the entire system may crash, for example due to a third-party application running on the same computers. In 2004, a computer glitch in the radar system disabled the surveillance of flights above 24,000ft [51] in the United Kingdom. Flight data had to be entered manually, resulting in a decrease in capacity and in an increase in spacing distances: On September 24, 2007, a major breakdown [4] affected Southern California's air traffic control system due to a "design anomaly" in the way Microsoft Windows servers were integrated into the system. The radio system shutdown, which lasted more than three hours, left 800 planes in the air without contact to air traffic control, and at least five near misses were reported. Air traffic controllers had to rely on their personal mobile phones to send warnings to controllers at other facilities, and watched several losses of separation without being able to alert pilots. This computer issue was known and had not been fixed after a first failure that occurred in 2004 [133]. In October 2009 during the test of a new system, as air traffic controllers watched the skies above

Salt Lake City International Airport, a Texas-bound flight departing from Salt Lake City International Airport morphed into a flight from Nebraska that just landed [23]; Continental Airlines Flight 440 had just left the Salt Lake City International Airport airspace and an air traffic controller transferred the flight to the computer of another air traffic controller. But the computer misidentified it as SkyWest Flight 4881, which had just landed in Salt Lake City from Omaha, Neb. The new system was shut down, the backup system fired up and then the backup system was replaced with the 20-year-old system air traffic controllers are currently using. To buy time to shut down and boot the computer systems, air traffic control required that pilots place their aircraft farther apart. Another potential threat to the computational infrastructure is cyber attacks against the ATMS system [137], where a hacker could take control over the computer system.

- **Communications:** Reliable communications are necessary to transmit information from the radar to the control facilities, between ground facilities and aircraft, and among ground facilities. If a control center loses its communication capacities, tens or hundreds of aircraft are left deaf and blind, the air traffic controllers being the eyes of the pilots. For instance, on September 25, 2007, a truck ran into a phone line pole, shutting down all the communications in and out of Memphis Air Route Traffic Control Center (ARTCC) . The pole was located two miles away from the control center. Controllers had to coordinate with other ARTCCs using their cellphones [27, 57, 91, 99]. The breakdown lasted for about 4 hours. At the time of failure, there were about 200 aircraft in the center's airspace.

- **Surveillance:** Computer or radar failures can cause surveillance failures. For instance, on July 9, 2008, Dublin's radar system repeatedly failed to display the call signs that normally identify each incoming aircraft [100]. After the second breakdown, controllers lost confidence in the system and aircraft had to be grounded or diverted to other airports. The outage affected more than 200 flights. In May 2007, the Southern California TRACON (SCT) was affected by an outage that let the controllers be

mapless for an hour [134].

- **Operations:** In the event of a degradation of current operations, there exist backup procedures described by ICAO [65]. This document explains the procedures that air traffic controllers and pilot should follow, a failure should happen. Despite those procedures, degradations of the operations and failures to follow the procedures still occur. For instance, issues in closely-spaced parallel runway operations have been investigated in several papers [114, 115]. An analysis of current en route air traffic control usage during special situations can be found in an analysis [38] by the Federal Aviation Administration (FAA). Operation failures can lead to accidents such as the 2002 collision [143] between a Russian passenger jet and a cargo plane over Germany. Contradictory orders provided by the Traffic Collision Avoidance System (TCAS) and the controller led to the collision. Operations might also be disrupted by exogenous factors such as lightnings strikes [136]. In most cases, aircraft just fly to the nearest airport [94, 136], disrupting local operations, but sometimes the strike leads to disastrous consequences [101].

- **Navigation:** If the frequencies carrying the GPS signals are jammed [107] or spoofed, GPS navigation becomes impossible. Aircraft highly rely on automated navigation using autopilots and flight management guidance computers (FMGC). With the increase of automation in modern aircraft, a loss of navigation not only impacts the situational awarness, but also the flying capabilities since autopilot and autothrottle become unavailable [1]. Navigation issues can also result from human errors: the irresponsiveness of flight 188 [2] is an interesting example of communication and navigation failures due to a human error, when the pilots did not contact ATCs for over an hour and a half and overshot their destination point by 90 NM.

- **Vehicles:** An intruder can enter the airspace and consequently jeopardize the safety of surrounding traffic. For instance, a private pilot can unintentionally get too close to a restricted airspace such as the vicinity of major airports; A well known example is the “Charles river VFR route” over the Charles River in boston, where aircraft often

intrude in Boston International Airport's airspace. If an aircraft flying Visual Flight Rules (VFR) gets in the landing path of an aircraft flying Instrument Flight Rules (IFR), this generates an abnormal situation that must be solved by the controller. Vehicles can also be affected by exogenous factors: Flight 1549 [145] had to alight on the Hudson river because of a bird strike.

- **Airport closure:** An airport or part of it can be closed, e.g. because of weather conditions or other reasons [100]. The traffic needs to be reorganized and rerouted towards other airports. For instance, Bangor International Airport is often a diversion destination when freezing rain, snow or fog close Boston, New York or other major Northeast metropolitan centers. On April 23, 2009, the control tower at Atlanta Hartsfield Jackson International Airport was hit by a lightning strikes and severe storms knocked out power to the area and the airport lights [15]. The tower had to be evacuated, leaving the airport inoperative, no aircraft being able to take off or land.

In most of the examples above, disastrous consequences were avoided thanks to the extraordinary ability of humans to accommodate unexpected situations.

Mostly due to exogenous factors, failures occur and will still occur in future ATMS, despite all efforts made towards preventing them. A key point in ensuring safe air traffic operation is to ensure a safe degradation of ATMS in the event of failures or unexpected exogenous events.

1.4 Graceful Degradation of ATMS

Within this thesis, the process by which the current or future system keeps operating safely despite degradation of the sustaining ATMS infrastructure will be called *Graceful Degradation*. When all the subsystems of the ATMS work properly, the ATMS is said to be working in nominal mode. When one or several subsystems fail, the ATMS works in degraded mode: A graceful degradation occurs when the transition from a nominal mode of operation to a degraded mode of operation is smooth and with no catastrophic event.

The term *Graceful Degradation* finds its origins in complex computer systems [146]. In 1967, the use of on-line spare modules and self-reconfigurable systems was already present in

the literature [146]. Even though the current computer systems are much more complex, the methods to ensure a graceful degradation have not changed much since then; Redundancy, replication and diversity are the methods usually used to create fault tolerant systems that ensure a graceful degradation [56, 88]. Such methods are used for hardware onboard aircraft or in traffic control centers. Due to the size and the complexity of the ATMS, those methods cannot always be used. For instance, an aircraft cannot be duplicated and the cost of duplicating every control center would be prohibitive. However, the concept can be immediately extended to overall systems such as the National Airspace System (NAS), which include both physical assets (airplanes) and complex information infrastructures.

The failure modes identidied in section 1.3 and how to recover from them have been identified on an *ad hoc* basis, whereby accidents have triggered extensive studies and redesigns of the air traffic management operations. Extensive experience by air traffic controllers about incidents has progressively led them to always address "what if" questions during routine operations, leading them to safe operations. This system comes complete with degraded operation and recovery procedures, as described by the ICAO [65]. One example of such fault-tolerant, or fail-safe procedure concerns departure operations: A safe, collision-free path is completely specified to the aircraft prior to take-off, in such a way that the aircraft may follow this path safely even in the case of complete communication failure during take-off [65].

The examination of concepts of operation such as NextGen in the US [70] and SESAR [125, 126] in Europe reveals that system safety and graceful degradation are considered open research issues for most future operations. Some of these research issues are gathered in Appendix B, Table 6 for NextGen and Table 7 for SESAR.

In this thesis, the problem of graceful degradation is addressed from two viewpoints: an analysis perspective and a design perspective.

1. **Analysis: Graceful degradation sensitivity.** *Consider a given air traffic situation (consisting of a number of aircraft with given positions, velocities and headings) along with a Communication, Navigation and Surveillance system infrastructure, what is the "sensitivity" of this situation relative to a sudden degradation in the ATMS?*

2. **Design: Graceful degradation-compatible guidance.** *Consider a traffic configurations, that is, a set of aircraft with given origins and destinations, find a sequence of avoidance maneuvers such that the safety of the airspace is never compromised.*

The problem of a graceful degradation for ATMS by means of actions on traffic faces two major challenges. The first challenge lies in the complexity of the system: ATMSs comprise thousands of pieces of equipment and people working together, and the safety of passengers and crews is at stake. The other challenge lies in the intrinsic nature of degradations: Degradations are **easy to identify**, meaning that it is usually easy to determine if a piece of equipment is functional or not. Then, a degradation is **difficult to describe**, meaning that analyzing the impact of the degradation on the overall system is not an easy task. Finally, since the consequences of the failure are usually difficult to evaluate, it is even **harder to handle** the degradation and take appropriate mitigation actions.

Sheridan [128] analyzed the issues associated with the human-automation interactions in the next generation air transportation systems: *"Because of the greater interconnectedness of aircraft and subsystems, equipment failures and misapplied procedures can cause perturbations that cascade throughout the whole system."* In the event of a failure of the automation, degradation modes should be available for the human controller to safely handle the system. A thorough knowledge and modeling of the degradation modes of the ATMS is necessary to ensure its safety.

Figure 1 presents a notional picture of the problem of graceful degradation for ATMS. Nominal operations are represented by the green ellipse. The sustainability of the nominal operations is ensured by a fully functional ATMS, depicted by the big blue shaded box at the bottom of the figure. If a failure happens in the ATMS, it will propagate throughout the system, which could potentially lead to off-nominal operations, possibly jeopardizing the safety of the airspace (big red ellipse in the figure). Graceful degradation can be ensured if a way to steer traffic from nominal to safe degraded conditions operations (dark green ellipse) is always available. The degradation is called graceful if the operations never enter the "red zone" of off-nominal operations. Enabling a graceful degradation is also achieved by monitoring the degradability of the operations (blue box at the top of the figure).

The identification of a degradation can be achieved at the technology level, using integrity sensors that provides the user with a health evaluation of the technology. It can also be achieved at the operational level, by monitoring aircraft operations for unexpected events.

1.5 Airspace Monitoring

Airspace monitoring consists of observing and analyzing the behavior of aircraft in the airspace or a subset of the airspace. Depending on the phase of the flight and on the objectives, different parameters can be monitored. This thesis focuses on the graceful degradation of ATMS, therefore proposes monitoring tools that surveil parameters critical to a graceful degradation. As presented in section 1.3, failures can affect technologies and infrastructures, but also aircraft operations. Airspace monitoring can evaluate the capability of a given traffic configuration to gracefully degrade, but also analyze the conformance of current flight path to predetermined typical operations.

1.6 Objectives of the thesis

The objective of this thesis is to introduce the concept of *graceful degradation* for air traffic management systems and to develop a better understanding of how to design, simulate, measure, and monitor the true level of safety of an airspace. The objectives of the thesis will be fulfilled by answering the following research questions (RQ):

- **Research Question 0:** What are the failures that can affect the ATMS?
- **Research Question 1:** What is the chain of events that results from a failure in the ATMS?
- **Research Question 2:** What is the impact of different classes of failures on operations?
- **Research Question 3:** What is the sensitivity of a traffic configuration to potential degradations?

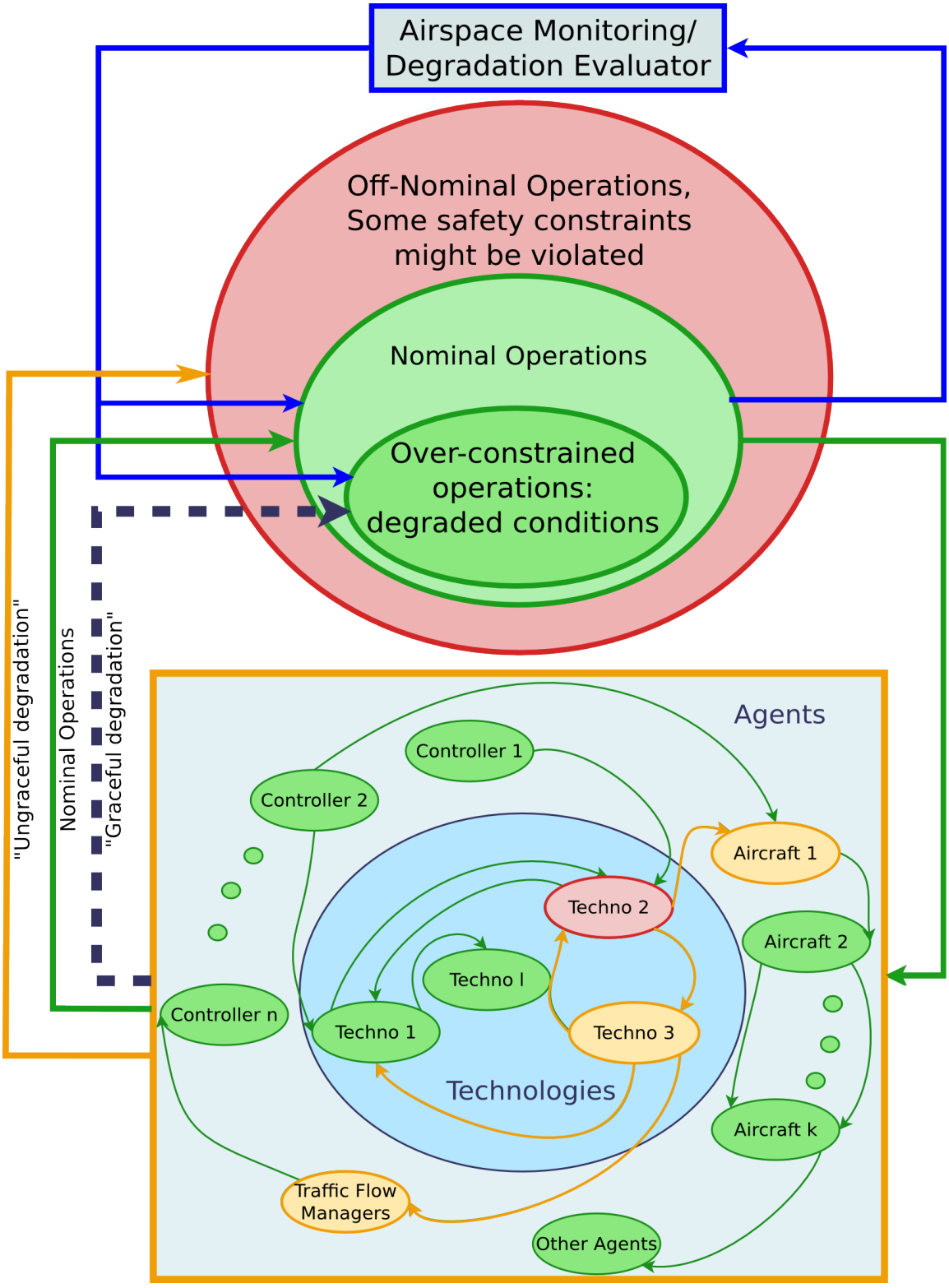


Figure 1: Scope of the thesis

- **Research Question 4:** How can the smooth transition from nominal mode of operations to degraded mode of operations be ensured?
- **Research Question 5:** Is it possible to monitor the airspace for a potential operational degradation?

1.7 Contributions

The main contribution of this thesis is to apply the notion of graceful degradation to air traffic management systems. This non exhaustive study of a graceful degradation of ATMS was achieved by the following contributions:

- Elaboration of an ontology for ATMS that enables the tracking of cascading failures and the measurement of their operational impacts.
- Creation and validation of an input-output model for a TRACON and determination its optimal capacity.
- Developement of an algorithm to probe traffic configurations in the event of a degradation of CNS systems.
- Development of an algorithm to mitigate CNS degradations by spreading out aircraft.
- Creation of monitoring tools for en-route and terminal areas traffic using a novel trajectory clustering algorithm.

1.8 Thesis outline

This thesis is organized in 6 chapters. This first chapter has provided the introduction of the concept of graceful degradation applied to ATMS, and the motivation behind this work. The following chapters address the various research questions presented in section 1.6.

The motivation of this thesis has been addressed by answering RQ 0 in this introduction (section 1.3) by listing various classes of degradation that can affect the ATMS.

Chapter 2 addresses RQ 1 and RQ 2 by tackling the problem of describing degradations through the analysis of failure propagation in ATMS. Measuring the impact of single or multiple failures on the entire system is a key element to ensure an appropriate mitigation.

Thus, an ontology of the ATMS is created and used to analyze the impact and measure the operational consequences of a failure of a subsystem of the ATMS, by tracking the propagation of technology failures and their impact on operations. This ontology enables to measure operational impacts of technology or human failures in a systematic way.

Chapters 3 and 4 address RQ 3 and RQ 4 by tackling the problem of handling certain categories of degradation and proposing solutions for a graceful degradation. In both chapters, the operational impact of particular failures or degradations is modeled and the response of the system to the degradation is analyzed. Then, the design question is addressed by proposing solutions to ensure the smooth transition from nominal mode of operation to degraded mode of operation. The modeled degradation in Chapter 3 is a landing capacity reduction at San Francisco International Airport (SFO) . Radar data are analyzed and a queueing model for the TRACON is created and validated. This model is then used to determine the optimal number of aircraft that maximizes the runway throughput, as well as reduces airspace congestion. In Chapter 4, the modeled degradation is a failure in the CNS system. A conservative model is presented and used in two different algorithms for aircraft conflict avoidance in the presence of uncertainties. An planar version that provides optimal avoidance maneuvers is used to probe traffic configurations and analyze their "degradation" capabilities. A 3-dimensional version of the algorithm is also proposed. This algorithm ensures graceful degradation by spreading out the traffic and leaving a conflict free configuration for 15 minutes.

Chapter 5 addresses RQ 4 and RQ 5 by tackling the problem of monitoring current operations for a potential degradation and have a solution ready to ensure a smooth degradation. For en-route traffic, the modeled degradation is a failure in the CNS system as presented in chapter 4. Complexity maps based on degradation of CNS capabilities are presented. For terminal areas traffic, a knowledge base for typical operations is generated, and a tool that identifies in real time the conformance of current flight to trajectories experimentally identified as typical is presented. The knowledge base is built using new trajectory clustering methods presented in Appendix A.

Finally, chapter 6 draws the conclusions of the thesis.

CHAPTER II

FROM TECHNOLOGY FAILURE TO OPERATIONAL DEGRADATION: TRACKING FAILURES IN THE AIR TRAFFIC MANAGEMENT SYSTEM

2.1 Introduction

This chapter addresses RQ 1 and RQ 2, that is, "What is the chain of events that results from a failure in the ATMS?", and "What is the impact of a failure on operations?", respectively.

The ATMS is a complex system that involves thousands of pieces of equipment, vehicles, facilities and people working together. In NextGen and SESAR, aircraft are expected to have a broader range of capabilities than today and to support varying levels of total system performance via onboard capabilities and associated crew training. To enable future capacity, novel technologies, vehicle types and, operational concepts are needed and they will ultimately bring forth new types (or modes) of failures and disruptions. If unattended, these disruptions could result in severe setbacks for the NextGen and SESAR agendas and for the health of the air transportation system as a whole. The increasing complexity of the ATMS makes tracking the propagation and the impact of failures more difficult and critical.

The objective of this chapter is to present an ontology that captures how failures or perturbations cascade throughout the system and that measures their impact in terms of loss of capabilities. If the impact of a failure is known, it becomes easier to ensure the graceful degradation of operations when a failure occurs. The introduction of new technologies and new aircraft is not possible unless they have been certified with a very low failure tolerance, resulting in very few critical onboard failures. Nevertheless, failures still occur as listed in section 1.3.

The ontology of the ATMS presented in this chapter is based on a multidimensional

decomposition, aiming at analyzing the impact of failures. A large share of the work in ATM is devoted to improving the performances of the current system and assessing new concepts of operations. Being at the center of air traffic operations, ATCs have been often modeled ([16, 80, 102]). Human in the loop simulations are used to validate new concepts [113]. Such experiments have also been developed to identify human errors in air traffic control [130]. At a higher level, Pinon et al. modeled the air transportation system as a supply chain [108] to measure its performance and constraints. Pinon et al. also presented a morphological decomposition of the air traffic operations, to evaluate and select airport technologies [109]. The system was decomposed from traffic phase, to possible improvements, to operational concepts, to functions, to technologies and finally, to sub-technologies. A matrix of alternatives is created from this morphological analysis [120]. This decomposition is oriented towards finding appropriate technologies to optimize operations. However, this decomposition is unidirectional and therefore, it is not possible to keep track of failures and to measure the losses of performance. Di Benedetto et al. modeled the ATMS as a stochastic hybrid system [24] to detect faults and mainly non-deterministics human errors due to lack of "situational awareness" [28]. Those hybrid systems take into account the dynamics of the agents and their potential errors. The ontology presented in this chapter does not include the system's dynamics but rather focuses on the deterministic health of the overall system.

Sussman has developed processes to study "Complex, Large-Scale, Interconnected, Open, Sociotechnical" (CLIOS) systems [132] and applied them to systems such as the transportation system of Mexico City. CLIOS systems are very large systems involving a large number of agents and technologies, where social behaviors enter into account. CLIOS systems are characterized by their internal and behavioral complexities [25]. The internal complexity is the number of components in the system and their interractions. The behavioral complexity [25] is the type of behavior that emerges from the system due to the components interactions. The ATMS possesses both internal and behavioral complexities of a CLIOS system. The emergent behavior is difficult to predict but unlike CLIOS systems, the relationships between the subsystems of the AMTS are known and can be modeled, despite

their complexity. The use of CLIOS system analysis methodologies focuses on identifying policies or management intervention to improve the systems. If the ATMS was to be studied in the framework of a CLIOS system, this ontology would be an key enabler towards understanding its complexity.

In a common report [39], the FAA and Eurocontrol presented the safety techniques used in ATM. Techniques used to model the impact of failures on operations include, but are not limited to, bow-tie analysis, even trees and fault trees [52, 81]. In all those frameworks, a failure or an error is first identified, and then the possible causes (i.e the branches of the tree) need to be generated by an expert. This ontology is an automated way to generate fault trees.

The remainder of this chapter is organized as follows: section 2.2 introduces the model, its objectives, how the ATMS is decomposed, the links between the blocks, and how failures are introduced and tracked. Section 2.3 presents a validation of the ontology on two scenarios. Finally, before the concluding remarks, section 2.4 presents two case studies to illustrate the uses of the ontology.

2.2 Presentation of the ontology

2.2.1 Objective of the ontology

The objective of this ontology is to provide a better understanding of the propagation of failures in the ATM system, and to measure their impact in terms of loss of capabilities. The ontology shows the propagation of failures, from a facility, a controller or a technology, all the way to operational capabilities. The model enables the identification of alternate or backuptechnologies and the analysis of a loss of automation can be handled by a human controller to ensure the safe transition from a nominal and automated mode of operation to a degraded and manual mode of operation.

2.2.2 System

The modeled system is the air traffic management system. It comprises all the infrastructures, technologies, communication media, people, etc. The system also includes aircraft

and pilots. In the current version, only existing infrastructures and technologies are included, but the model is flexible enough, so that new technologies and automation systems can be added to the ontology while they are being tested.

2.2.3 Ontology description

The ontology combines a physical decomposition of the major components of the ATMS, and a functional decomposition of the air traffic operations into tasks and then functions. This ontology combines elements from the decompositions presented by Pinon et al. on the one hand [109], and Kim et al. on the other hand [71]. Pinon et al. decomposed air traffic operations to identify enabling technologies. Kim et al. proposed a task decomposition for function allocation.

This ontology starts from a physical decomposition of the system into facilities and aircraft, then decomposes them into technologies and human operators. Human operators and technologies execute functions that are enabled by other functions and communication media.Then, those functions are used to execute tasks. Finally, tasks are combined together to enable operations. The functional decomposition is linked to the physical decomposition through the functions.

Figure 2 presents a diagram of the decomposition of the elements of the model. The terms used in this ontology are defined as follows:

- **Facilities/Aircraft:** This category groups physical pieces of equipment and/or people located at the same place. A *Facility* refers to a building or place that provides a particular service or is used for a particular purpose. For instance, the TRACON facility refers to the physical building in which air traffic controllers work to direct aircraft in the corresponding TRACON airspace. A facility can also refer to a simple building, e.g. the building and mount for a radar or an Automatic Dependent Surveillance - Broadcast (ADS-B) ground station.

  An *Aircraft* refers to a vehicle that can fly and enter the controlled airspace, such as an airplane, a helicopter or an unmanned aerial vehicle.

- **Technologies:** A *Technology* refers to a physical piece of equipment such as a

transponder, a radar, etc. A technology is located in a facility or an aircraft and executes one or several functions.

- **Human Operators:** A *Human Operator* refers to a human being qualified to execute the tasks required by his position/job. Human operators include pilots, air traffic controllers, traffic flow managers, etc. Human operators are located in facilities or in aircraft and execute one or several functions.

- **Communication Media:** A *Communication Medium* refers to the transmission channel or tool used to deliver information, such as radio waves in a given range of frequencies, phone lines, etc.

- **Functions:** A *Function* refers to "a capability without a goal", of a technology or a human being. Transmitting information or displaying information on a screen are examples of functions.

- **Tasks:** A *Task* refers to a tangible activity with a goal. A task is made possible through by combining functions together. Monitoring aircraft position is an example of task.

- **Operations:** An *Operation* refers to a tangible activity with a goal resulting from the combination of several tasks. For instance, sequencing and merging is an operation that requires air traffic controllers to direct aircraft and pilots to follow ATC instructions and fly the aircraft.

This decomposition enables the introduction of failures at different levels (Section 2.2.5). The propagation of a failure can be tracked in the ontology using its influence structure, defined thereafter.

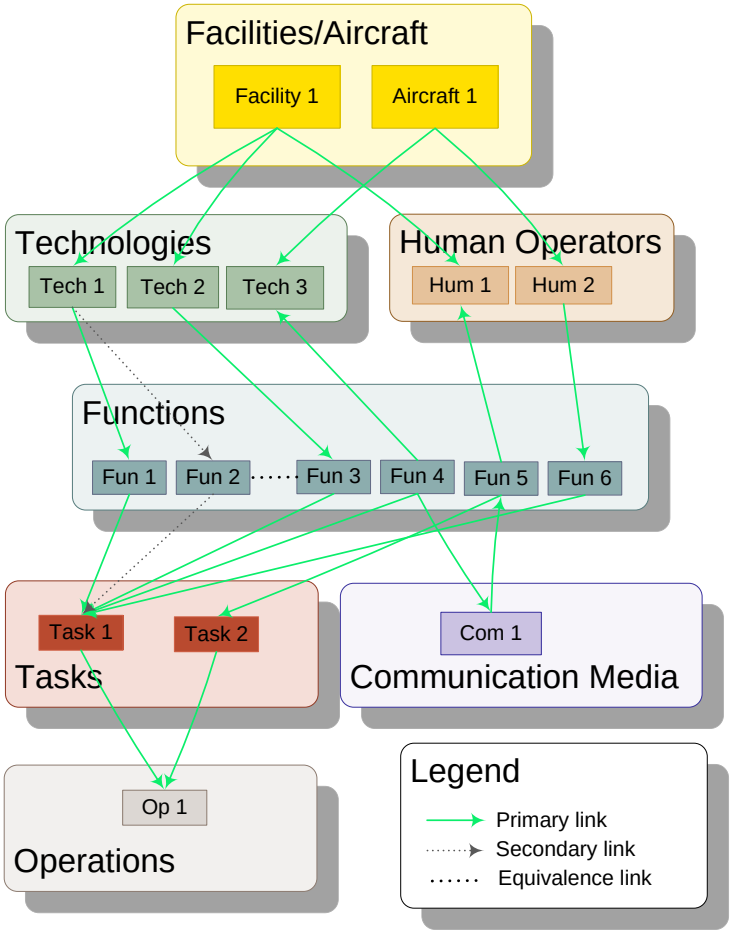


Figure 2: Model's decomposition

Table 1: Influence structure and dimensionnality

Origin		Destination	Dimension	Meaning
Facility	$\rightarrow$	Technology	Hosts	The technology is physically located inside the facility.
Facility	$\rightarrow$	Human operator	Hosts	The human operator is physically located inside the facility.
Technology	$\rightarrow$	Function	Executes	The technology executes this function. The information available to the technology is used to perform the function that will generate new information.
Function	$\rightarrow$	Technology	Provides information	The output of this function is used by the function. The information generated by the function is used by the technology.
Human operator	$\rightarrow$	Function	Executes	The human operator executes this function, generating new information.
Function	$\rightarrow$	Human operator	Provides information	The human operator uses the output of this function. The information is received by the operator.
Function	$\rightarrow$	Communication media	Emits on	The output of the function is transmitted over the communication medium. The communication medium must be available for the information to be successfully transmitted.
Communication media	$\rightarrow$	Functions	Transmits	The communication medium transmits information that can be captured by the receiving function.
Function	$\cdots$	Function	Is equivalent	The two functions are equivalent, in terms of role. They might have a different level of performance.
Function	$\rightarrow$	Tasks	Enables	The function enables the task. A task might require several functions to be achieved.
Tasks	$\rightarrow$	Operation	Enables	The accomplishment of the task is required for the operation to be conducted.

2.2.4 Influence structure and dimension

The influence structure of the ontology is the set of relationships and links existing between the different components of the ontology. The signification of the links between the elements is presented in Table 1. The dimension corresponds to the type of relationship existing between the linked blocks. The term *origin* refers to the block at the tail of the arrow, and *destination* refers to the block located at the head of the arrow. The relationship "Hosts" means that the destination block is located inside the origin block. The relationship "Executes" means that the origin block executes the destination block. The relationship "Emits" means that the origin block emits information using the destination block. The relationship "Transmits" means that the origin block transfers the information to the destination block. The relationship "Enables" means that the origin block makes the achievement of the destination block feasible. The relationship "Equivalence" does not carry any dependence information. It is used to determine redundancy in the technologies.

The ontology has three types of links: primary, secondary and equivalence. The test cases of section 2.4 provide diagrams for visual representation of those links with their associated color code.

- **Primary links:** Primary links correspond to nominal interactions between the different components. They are represented by colored arrows: a green and plain arrow indicates a link working nominally. A dashed orange link indicates that some of the information nominally carried by the link is missing. A dotted red arrow indicates the the link is no functional.

- **Secondary links:** Secondary links correspond to redundancies, not used in nominal modes. They are also represented by colored arrows: when the link is inactive, it is represented by a dashed gray line and when it is active, it takes the colors of a primary link. For instance, the primary radar can be used as a backup for the secondary radar, but it does not provide the same level of performance. The functions enabled by the primary radar are contained in the ontology, but the links are listed as secondary, since they are not used during nominal operations.

- **Equivalence links:** Equivalence links connect blocks with similar characteristics. They are represented by black dotted lines. Two technologies are equivalent if and only if they are identical. If they are not identical, they can perform identical functions which will have the equivalence relationship. Functions do not need to be exactly identical to be equivalent. Two equivalent functions can have different levels of performance. Equivalence links allow the ontology to find redundant systems to perform failed functions.

2.2.5 Failures and degradations modes

The ontology enables the introduction of failures at all the levels of the decomposition. Failures can affect single or multiple blocks but cannot be introduced on links, since links have no physical meaning. The color of the links presented in the previous section refers to the type and the availability of the information they carry. Failures can affect:

- **Facilities/Aircraft:** Failures affecting facilities and aircraft are potentially the most difficult to handle, since they host many people and technologies. Such failures can be total or partial. When a facility failure is total, it is propagated to technologies and human operators located in this facility. When the failure is partial, only some technologies or human operators will be set as "failed". A total failure can be visualized as a master switch for all the technologies and people in the facility. A partial failure can be seen as a switch for a particular room.

- **Technologies:** A technology can fail because the facility in which it is located fails, or because the technology itself fails. The same way facilities fail, technology failures can be partial or total. If the failure is total, all the functions enabled by the technology will be set to inoperative. If the failure is partial, only a set of functions will be set to inoperative.

- **Human Operators:** Failures affecting human operators are modeled the same way as failures affecting technologies.

- **Communication Media:** When a communication medium fails, the information it

carries cannot reach its destination. Therefore, the destination function of the links exiting the medium will be disabled, because the information could not be transmitted.

- **Functions:** A failure cannot be introduced at the function level. If a technology or an operator cannot execute a function, it is modeled as a partial failure of the technology or operator. A function can fail if its input link(s) carry failures.

- **Tasks:** Tasks can fail by the propagation of functions failures. Failures at a task level can also be introduced to model human errors. Task failures propagate to the operation level.

- **Operations:** Operations can fail because of the propagation of tasks failures.

When a failure is introduced in the ontology, the failure is propagated throughout the system by the links and its impact can be measured by an incapacity of executing tasks and operations. Since links also carry partial failures, the ontology also allows to measure decreases in performance.

2.3 Validation

This section presents a implementation of the ontology using MATLAB and Simulink. To show the capabilities of the ontology, a limited ATMS was simulated. Matlab and Simulink were chosen for the rapidity of prototyping. Simulink offers a great simulation environment with interaction and communication between elements. The ontology was implemented in Simulink, and a control panel module to introduce the failures and track their impact was created using MATLAB. In this implementation of the ontology, there is no notion of time and propagation time for the failure. This could be achieved by introducting delays along the links of the Simulink model.

2.3.1 Simulink Model

In this implementation, yellow blocks represent facilities and aircraft. When going inside those blocks, the technologies and human operators hosted by those facilities appear. The

outputs of the technologies and operators are the functions accomplished. This Matlab-Simulink implementation is deterministic. Future implementations may consider introducing propagation time in order to provide a prioritization of the actions to take.

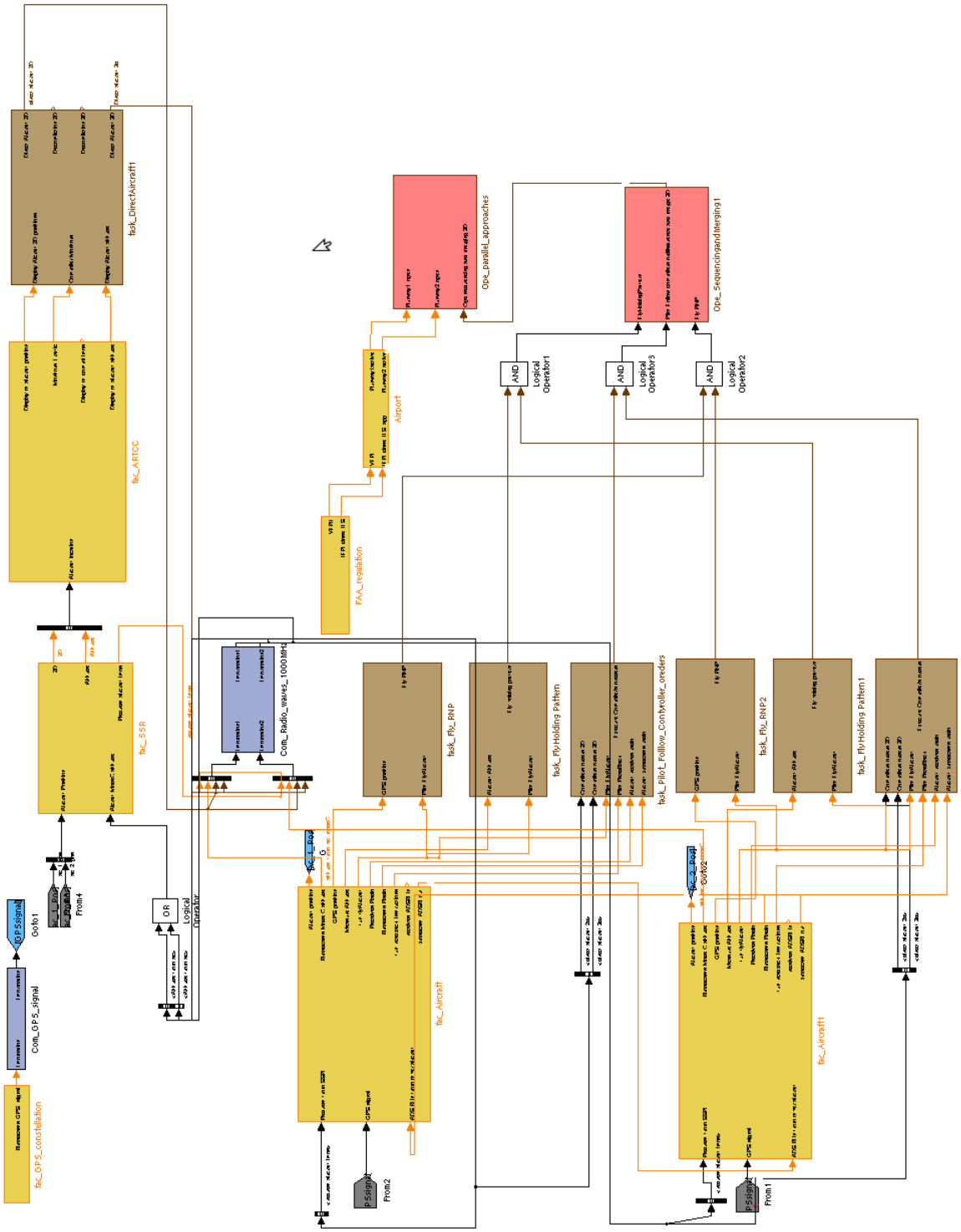


Figure 3: Simulink model for the ATMS Ontology

2.3.2 Matlab Interface

An automated MATLAB interface was developed together with the Simulink model. This interface automatically reads the Simulink model and look for all the facilities, technologies, etc. Figures 4 and 6 present the control panel with the facilities, technologies, operators and their associated functions. The introduction of failures is enabled by check-boxes. By unchecking one or several boxes, the corresponding components are set as inoperative. Figures 5 and 7 present the control panel that shows the status of communication medias, tasks and operations. In both control panels, a green background indicates a nominal state. A red background indicates that the function,task or operation cannot be executed.

To validate the ontology, two test cases have been run, using two types of events coming from the news. The first event consists of a loss of a radar, and the second event is degradation due to poor weather.

2.3.3 Event 1: Loss of radar

2.3.3.1 Event description

Radars are key elements of the ATMS. In the event of a failure of one of them, operations are dramatically affected. Radar failure is a relatively frequent event around the world: Southern California TRACON radar [134] in May 2007, Dublin airport radar [55, 100] in July 2008, a radar in Manila, Philippines, [105] in 2009, United Kingdom airspace radar [7] in 2010, and a radar in Delhi, India [135] in 2010. The list is non exhaustive but shows the frequency of such outages. From the cited articles, it was determined that the consequences of a radar failure are that

- Inbound and overflying aircraft were diverted
- Departing aircraft were grounded
- System was working at reduced capacity
- Controllers could not obtain aircraft identification tags on their screens.
- Controllers could not obtain aircraft altitudes

- The airport was shut down
- The backup system used was less performant as it did not handle flight plan information
- No map of traffic was available to air traffic controllers
- Since no map was available, no vector could be given to aircraft.

2.3.3.2 Ontology results

The ontology was run with a failure introduced at the level of the facility for the secondary radar. The radar appears as still active because the failure is introduced at the level of the facility. Note that the functions of the radar (as a technology) are disabled. Figures 4 and 5 present the MATLAB interface when the secondary surveillance radar (SSR) is inoperative. The cascade of failures affects principally the controller that cannot monitoring traffic since its control station does not display aircraft's positions. Aircraft's mode-C transponder will not broadcast the aircraft's altitude since there were no such request, the secondary surveillance radar being inoperative. The impact on tasks and operations is major since all activities requiring air traffic control feedback cannot be executed anymore. The aircraft are left on their own. The mitigation of such a scenario is developed in Chapter 4. The ontology predicts the same impact on operations as the ones observed in the section 2.3.3.1. The controllers cannot direct the aircraft nor deconflict them. The operations of sequencing, merging and approach are unavailable, therefore no aircraft can safely fly in the considered airspace. The controllers cannot identify the aircraft nor monitor traffic, because the control station do not display aircraft position or altitude.

2.3.4 Event 2: Poor weather over SFO

2.3.4.1 Event description

Operations at San Francisco International airport (SFO) are highly constrained by the weather. The FAA requires a minimum 4,300 ft runway separation for simultaneous parallel instrument landings [35]. SFO runways centerlines are only 750 ft apart (Figure 10).

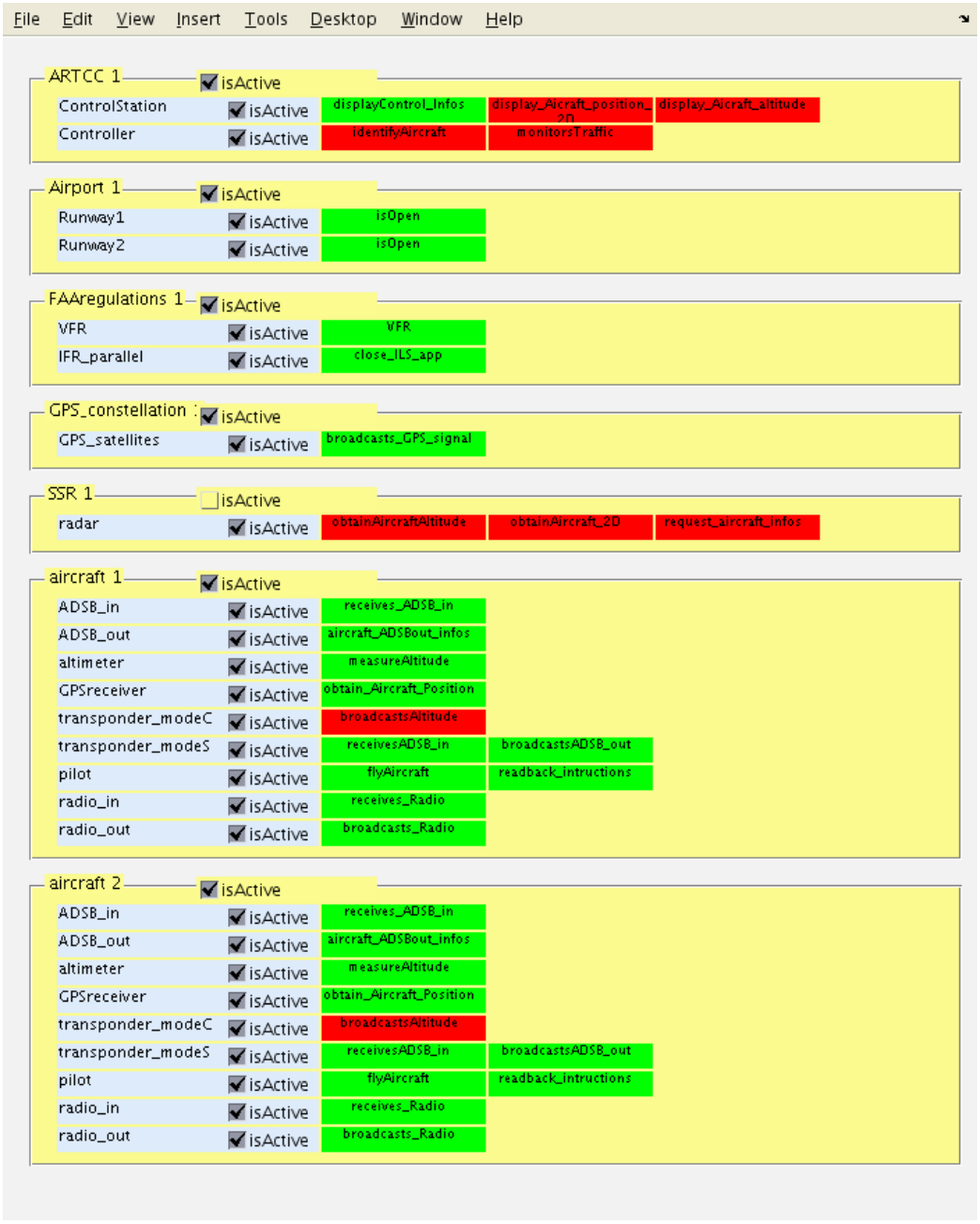


Figure 4: MATLAB interface - Technologies control panel for the ontology in the event of a failure of the Secondary Radar

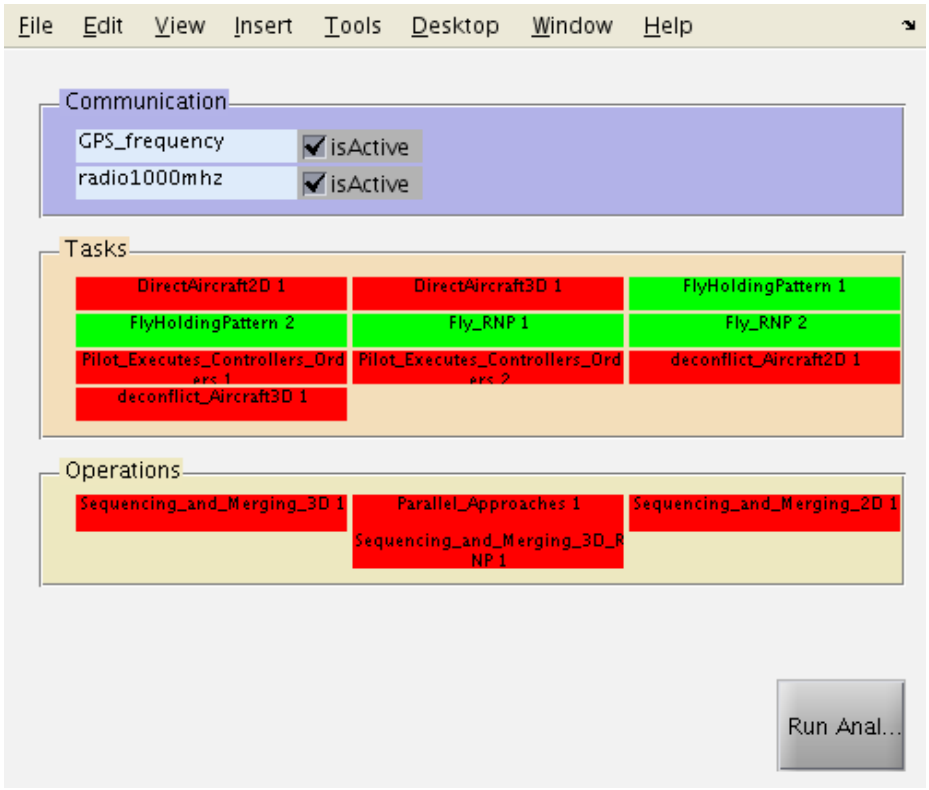


Figure 5: MATLAB interface - Operations control panel for the ontology in the event of a failure of the Secondary Radar

Because of the substandard runway separation, during poor weather conditions, simultaneous landings are not permitted and the landing capacity is reduced from 60 landings per hour on two runways to 30 landings per hour on one runway. During parallel landing operations, one aircraft can use the Instrument Landing System (ILS) and the following aircraft lands using VFR. When the conditions are poor, VFR landings are not possible.

2.3.4.2 Ontology results

The ontology was run with a failure introduced at the level of the FAA regulations, that do not permit VFR landings. Figures 6 and 7 the MATLAB interface for this scenario. The degradation propagates throughout the system and results in the first runway of the airport becoming inoperative. Therefore, the operation consisting of parallel approaches is unavailable. Chapter 3 presents an analysis of such a scenario at San Francisco International Airport, when the fog causes the airport to run operations with only one runway active for landings. The ontology predicts the same consequences as the ones observed in the real system.

The two test cases presented validate the ontology and show its possible uses. The next section present other test cases of failures founds in the literature.

2.4 Application of the ontology

This section presents two case studies to illustrate the uses of the ontology. The first case illustrates the propagation of a failure of the barometric altimetry in one aircraft. The second case shows how the ontology can be used to find alternative technologies in the event of a GPS jamming.

2.4.1 Failure of the barometric altimetry in an aircraft

In this example, a failure is introduced in the barometric altimetry of an aircraft [142]. It is assumed that the technologies providing this function, that is the barometric altimeters, are inoperative. Figure 8 presents a simplified version of the ontology. The blocs are depicted in red if they are the origin of the failure, or if this element has failed totally. A block in orange is partially affected by the failure. It is visually easy to follow all the elements

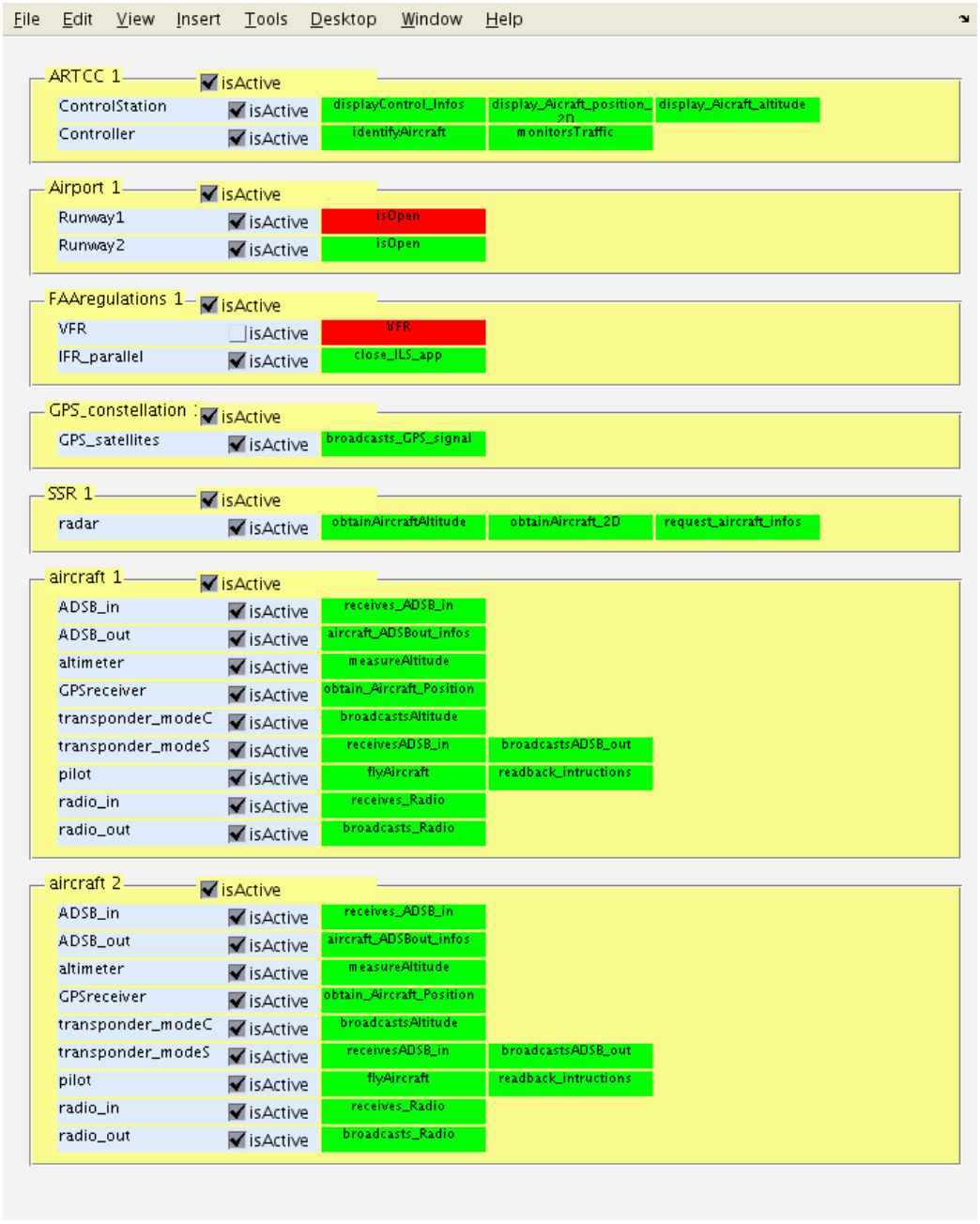


Figure 6: MATLAB interface - Technologies control panel for the ontology in the event of poor weather

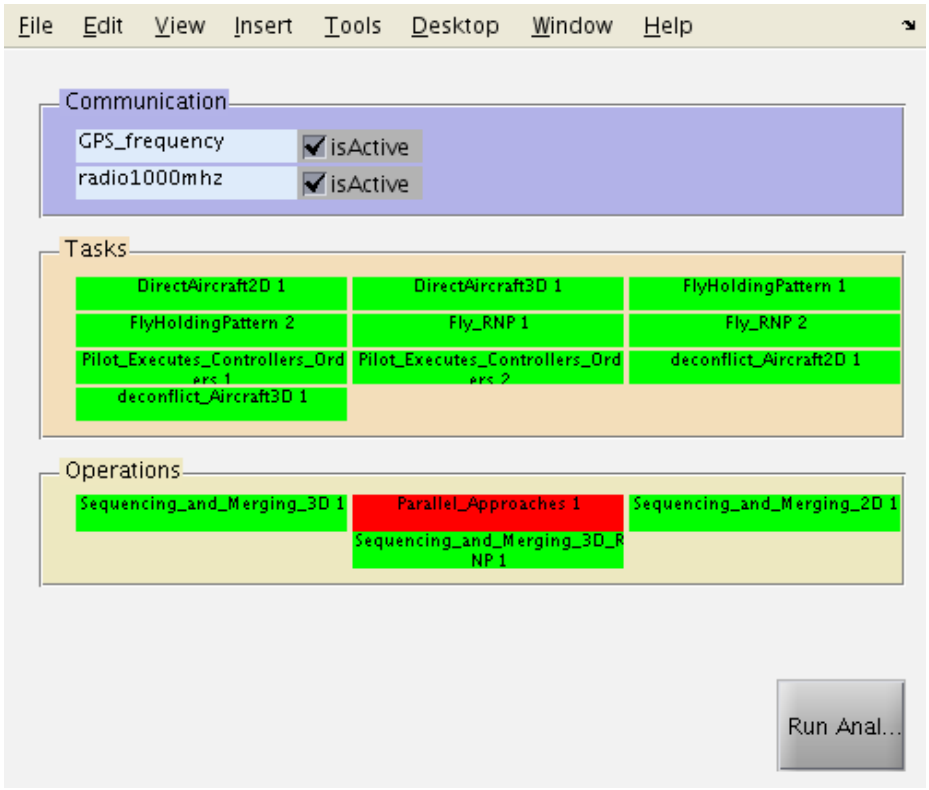


Figure 7: MATLAB interface - Operations control panel for the ontology in the event of poor weather

that completely failed and those which suffer of a loss of capabilities. Table 2 summarizes how the failure propagates along the ontology. Only some blocks were selected to illustrate the example. When the barometric altimeter fails, its main function, which is to measure the pressure altitude is not available anymore. Then, the onboard Mode-S transponder which broadcasts the altitude cannot obtain the altitude information and this information cannot be broadcast over the 1090 MHz radio frequency. The radio waves do not carry this information and therefore cannot be received by the ground-based Mode-S transponder, which cannot display the information onto the displays of air traffic controllers. Since the air traffic controller does not know the altitude of the aircraft, the task "surveillance" cannot be fully achieved. Moreover, since the altitude is not available to the pilot or to the autopilot, it is not possible for the aircraft to fly a holding pattern, which requires to maintain a constant altitude. Finally, since the two tasks "surveillance" and "fly holding pattern" cannot be achieved, the operation consisting of "sequencing and merging" cannot be realized.

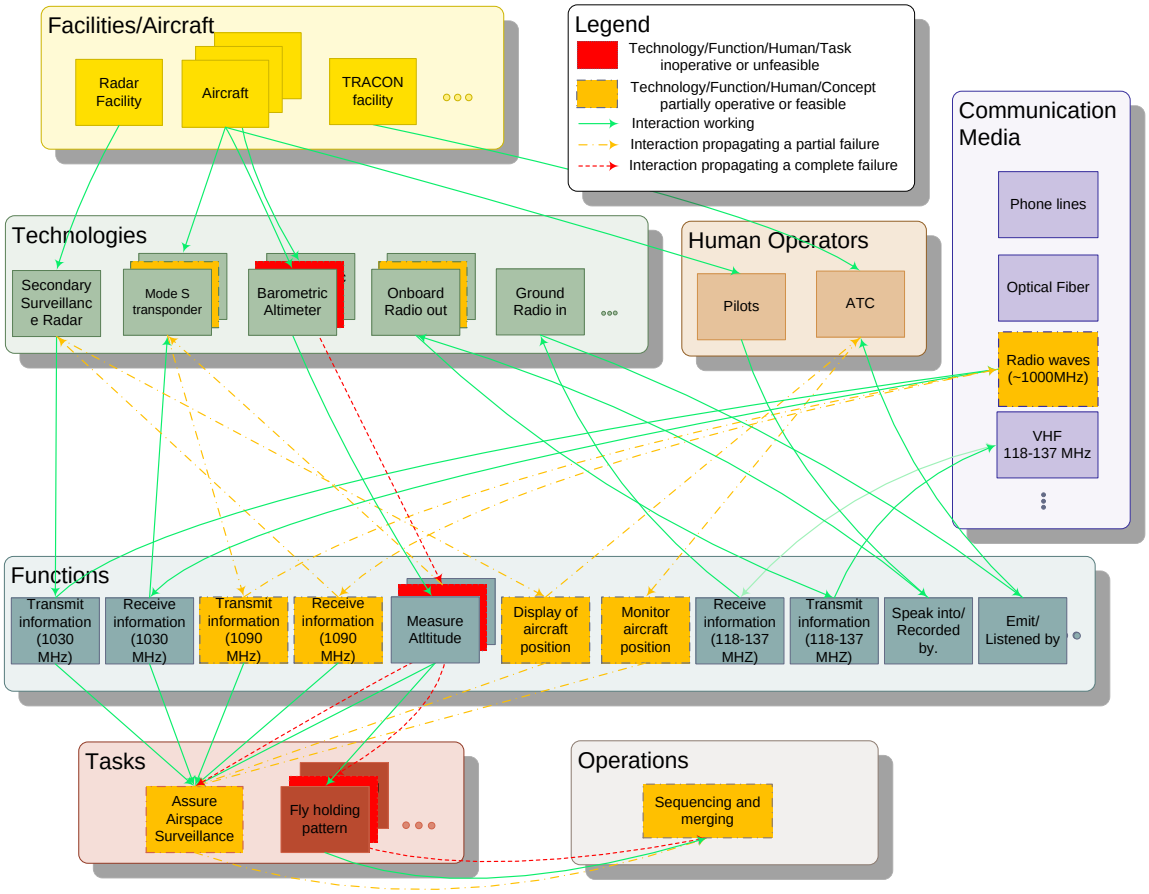


Figure 8: Simplified Air Traffic System Model with a failure of the barometric altimeter in one aircraft

Table 2: Example 1: Propagation of a failure due to inoperative barometric altimeter

Origin (Type)		Destination (Type)	Explanation	Failure
Barometric altimeter (Tec)	→	Measure Altitude (Fun)	The altimeter cannot measure the altitude.	Total
Measure Altitude (Fun)	→	Onboard Mode-S Transponder (Tec)	The Mode-S transponder cannot get the altitude information.	Total
Onboard Mode-S Transponder (Tec)	→	Transmit Information (1090MHz) (Fun)	The Mode-S transponder cannot transmit the altitude information.	Partial
Transmit Information (1090MHz) (Fun)	→	Radio Waves (1000Mhz) (Com)	There is no altitude information to transmit.	Partial
Radio Waves (1000Mhz) (Com)	→	Receive Information (1090MHz) (Fun)	There is no altitude information to receive.	Partial
Receive Information (1090MHz) (Fun)	→	Ground Mode-S Transponder (Tec)	The transponder cannot receive the altitude information.	Partial
Ground Mode-S Transponder (Tec)	→	Display of aircraft position (Fun)	The position of the aircraft cannot be accurately displayed since the Mode-S transponder did not receive altitude information.	Partial
Display of aircraft position (Fun)	→	Surveillance (Task)	The surveillance task cannot be executed properly as the altitude of an aircraft is missing.	Partial
Measure Altitude (Fun)	→	Fly holding Pattern (Task)	It is not possible to fly a holding pattern since it requires to maintain the altitude, which is not available.	Total
Fly holding Pattern (Task)	→	Sequencing and Merging (Ope)	Sequencing and merging might require an aircraft to fly a holding pattern. The Operation cannot be achieved.	Partial
Surveillance (Task)	→	Sequencing and Merging (Ope)	Sequencing and merging requires that the surveillance task is achieved properly.	Partial

2.4.2 Jamming of the GPS Signal

Future operations will highly rely on accurate positioning using GPS. Super Density Operations [72] will require aircraft to precisely follow predetermined trajectories consisting of a sequence of way-points coordinates. GPS is necessary to ensure Required Navigation Performance (RNP) operations. The automatic dependant system - broadcast (ADS-B) literally depends on the aircraft being able to determine its position, in order to broadcast it to ground stations and to other aircraft, using "ADSB in" and "ADS-B out" capabilities. The advent of ADS-B might allow self separation for en-route traffic and eventually in congested terminal areas. Such operations would be severely jeopardized in the event of a failure or a jamming of the GPS system. Figure 9 depicts the impact of a GPS jam on ADS-B operations. This representation is slightly simplified and is organized by entity for a more compact view. This example presents only one direction of communication, that is only aircraft 2 trying to determine the position of aircraft 1 using ADS-B. This example shows how the traffic information service (TIS) could be used as a backup to ADS-B in operations in terminal areas. The TIS is a first-generation traffic system that enables the display of surrounding traffic in the cockpit, using information uploaded from 107 Mode S terminal radars. The link between "ADS-B out" of aircraft 1 and "ADS-B in" of aircraft 2 is the primary link for aircraft 2 to obtain surrounding traffic's information. If this link fails, the secondary link is activated and the TIS is used as a backup system.

2.5 Summary

This chapter answers RQ 1 and RQ 2, that is, "What is the chain of events that results from a failure in the ATMS?", and "What is the impact of a failure on operations?" by presenting a ontology for the air traffic system based on a physical and a functional decomposition of the system. The ontology enables the tracking of failures throughout the ATMS and mesure their impact on operations. The losss of capabilities is instantaneously evaluted.

The ontology links two decompositions: a physical decomposition and a functional decomposition. The physical decomposition begins with facilities and aircraft, and decomposes them into technologies and human operators. The physical decomposition also includes

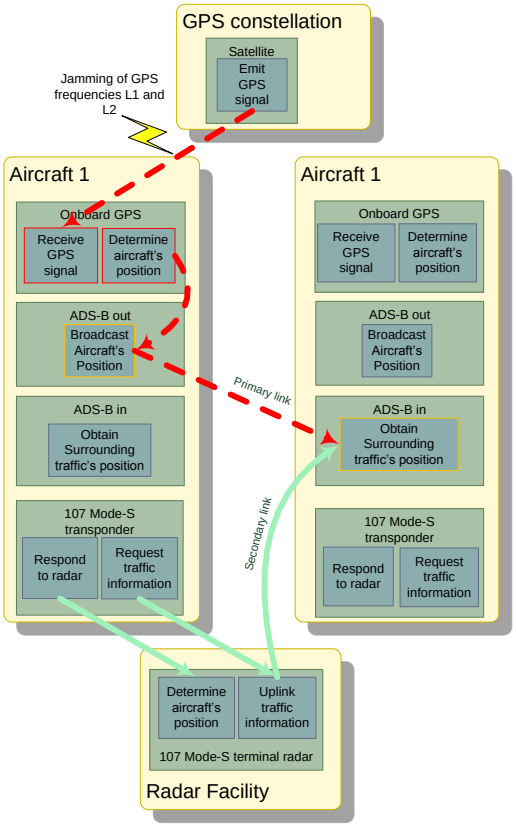


Figure 9: Impact of GPS jamming on ADS-B operations

communication media. The functional decomposition begins with operations and splits them into tasks and then into functions. The functions are the meeting point between the functional and the physical decompositions, since functions are executed by technologies or human operators. Those functions can communicate via communication media. The different components of the ontology are linked together by different relationships that enable the propagation of failures. Failures of systems or subsystems can be introduced at different levels and their impact can be tracked all the way to the decrease of performances in operations. The ontology is validated on two examples using a MATLAB-Simulink implementation. The results of the propagation of the failure through the implemented ontology are compared with the consequences of similar failures in the real ATMS.

As new technologies are introduced to leverage new concepts of operation, this ontology enables the study of their modes of failure and to find alternative solutions to ensure the graceful degradation of the ATMS. This ontology answers the question "What is the chain of events that results from a failure in the ATMS?" by instantaneously providing a clear understanding of the propagation of the failure, the people, technologies, functions and tasks that are affected. Ultimately, it answers the question "What is the impact the impact of a failure on operations?" by bringing forward the loss of operational capabilities resulting from a failure. This ontology can be used as a health monitoring tool for the entire ATMS. For instance, in each control center, ARTCC or TRACON, the tool could be running and be connected to sensors for each technology, facility, etc. In the event of the failure of one of them, the manager of the control center could instantaneously evaluate the severity of the situation and take appropriate mitigation actions.

CHAPTER III

OPERATIONAL DEGRADATION CASE STUDY 1: DEGRADATION OF AIRPORT LANDING CAPACITY

3.1 Introduction

This chapter addresses RQ 3 and RQ 4, that is "What is the sensitivity of traffic configurations to potential degradations?" and "How can the smooth transition from nominal mode of operations to degraded mode of operation be ensured?", respectively. In this chapter, the modeled degradation is a loss of landing capacity at San Francisco International Airport (SFO), as presented in section 2.3.4. This loss of capacity is the result of a runway closure, most often due to weather in this case, but it can also result from an incident on the airport surface. The sensitivity of the traffic configuration to a reduction in landing capacity is measured as the aircraft congestion building up inside the TRACON. Terminal Radar Approach Control (TRACON) areas constitute a critical airspace buffer around major airports, which receive aircraft flows from adjacent centers and deliver highly structured traffic flows to the arrival runways. TRACON facilities control the airspace surrounding the largest 166 US airports [40]. The TRACON airspace is usually a cylinder of radius 30 to 50 NM and height 10,000 to 20,000 ft [144]. Inbound flows of traffic from adjacent centers to the TRACON are regulated using the Traffic Management Advisor (TMA) [98, 79], which protects TRACONs from unacceptable congestion, thereby alleviating the TRACON air traffic controller's workload. The TMA aslo time meters the arrivals of aircraft in the TRACON in order to provide smooth flow of aircraft to the TRACON controllers. Nevertheless, unexpected events such as runway closures can disrupt the operations, leading to congested TRACONs. TRACONs constitute one of the most vulnerable parts of the air traffic control system as air traffic keeps growing. Reducing TRACONs traffic load to the necessary minimum is a key safety matter. TMA provides an efficient solution to regulate inflow of traffic to the TRACON. The results presented in this chapter complement the

TMA by providing bounds for the inflow of traffic.

In this chapter, a queueing model for the TRACON is developed. There already exist many queueing models for ATM applications. Several works focus on ground operations, where the queueing process is more visible: all aircraft are present at the same time on the ramps and taxiways, just landed or ready for departure. Idris proposed a queueing model to predict taxi times [62]. Ground operations were also modeled by Pujet [116], especially at Boston Logan Internation Airport. At the terminal airspace level, Vandevenne and Lippert [140] proposed two algorithms to sequence aircraft and optimize delays in the terminal area. The results showed that at airports with multiple runways, optmizing the sequence of aircraft inboud for each runway could considerably improve the airport throughput. Chen and Zhao presented a queueing model [14] to estimate delays at landing. Chen and Zhao modeled the interarrival time between aircraft using a constant value plus a random variable. The aim of this interarrival time probability distribution is to model the spacing designed by controllers. The model presented in this chapter present similarities with the one introduced by Pujet [116].

The first step towards answering RQ 3 is to model the system. Theoretically, runway capacities can be calculated using analytical formulas and nominal separation standards. In this chapter, those capacities are determined and analyzed through available TRACON data. Then, a model for the TRACON that captures airport performance as a function of the demand is presented. The model a simple queueing model where aircraft claim a time slot when they enter the TRACON. The model is calibrated using data, i.e the distribution of arrival times as well as travel times in the TRACON are computed from available data. The airport performance is the effective landing capacity, which depends on the number of runways in use. The proposed model is validated with San Francisco TRACON radar tracks records.

Once the model is available, it becomes easy to measure the sensitivity of traffic configurations to degraded capacity, by varying the queueing policy. Then, a solution to ensure a smooth transition from nominal to degraded mode of operation is proposed by determining the optimal number of aircraft present in the TRACON that maximizes the airport

throughput while minimizing congestion.

This chapter is organized as follows: Section 3.2 presents the available data, and introduces some definitions. In section 3.3, radar tracks are analyzed to determine the airport and TRACON characteristics. Section 3.4 presents the model developed. Finally, before the summary of the chapter, section 3.5 evaluates different queueing policies and determines the optimal number of aircraft that should be simultaneously in the TRACON to ensure the best degradation possible in terms of congestion, in the event of a loss of landing capacity.

3.2 Data presentation and definitions

3.2.1 Data source

The dataset used in this thesis was provided by the Aicraft Noise Abatement Office of San Francisco International Airport (SFO). The dataset contains the track records of all the flights in the Northern California TRACON (NCT) from January to September 2006. This dataset is furthermore used in Chapter 5 and in Appendix A, for airspace monitoring and trajectory clustering purposes.

3.2.2 Data selection

The presence of several airports – Oakland International Airport (OAK), San Francisco International Airport (SFO), San Jose International Airport (SJC) and many small airports – in a small perimeter such as San Francisco Bay implies dedicated routes for landings and takeoffs for each airport in the TRACON. Figure 11 presents the major jet arrival and departure routes in the San Francisco Bay area. Those routes are based on Standard Instrument Departure (SID) and Standard Terminal Arrival Route (STAR) procedures. Those routes decrease the available options to reroute aircraft. For practical reasons such as fuel capacity and for financial reasons, it is required to minimize the number of waiting aircraft and their waiting time. SFO (Figure 10) has four runways arranged in two sets of parallel runways. The weather and the terrain configuration (presence of residential areas located west, north and south of the airport) are such that the "West" configuration is the most used in order to reduce the noise over residential areas. In that configuration, aircraft take off on runways 01L/R, departing over the bay and land on runways 28L/R,

maximizing the distance flown over the water in order to reduce the environmental impact of noise. This study focuses on this configuration. The runway configuration of SFO is such that departures and landings are on separate runways. During typical operations, departures on runways 01L/R are sequenced to immediately follow arriving aircraft on runways 28L/R. Basically, during parallel approaches, as soon as the aircraft landing on runways 28L/R have cleared the intersection with runways 01L/R, the departing aircraft on runways 01L/R are cleared for takeoff. Therefore, takeoffs are sequenced in such a manner that they do not influence landings.

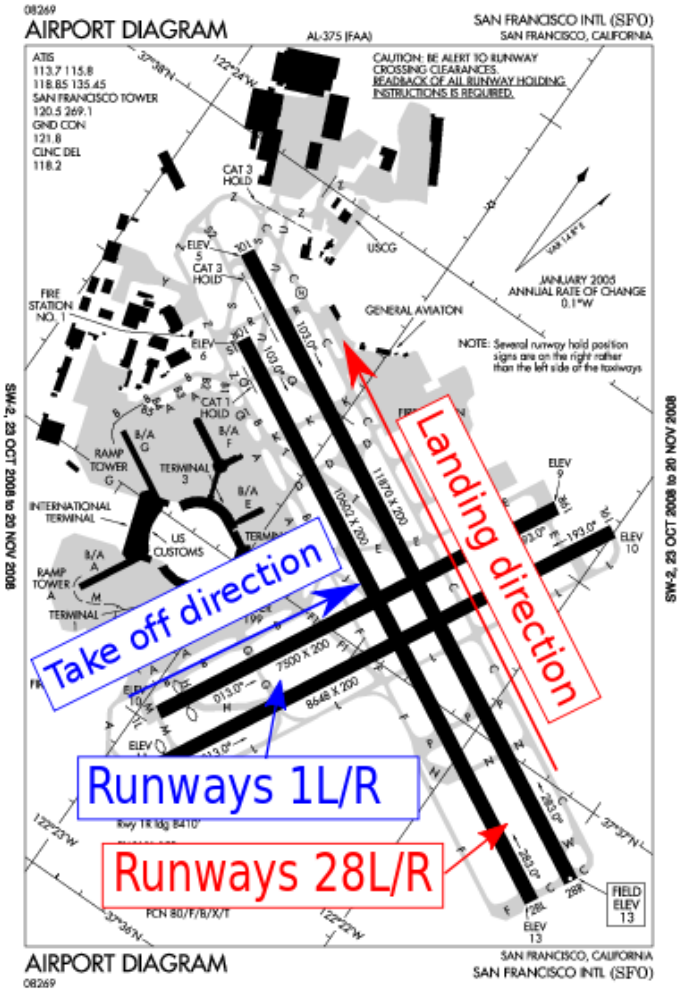


Figure 10: San Francisco International Airport diagram

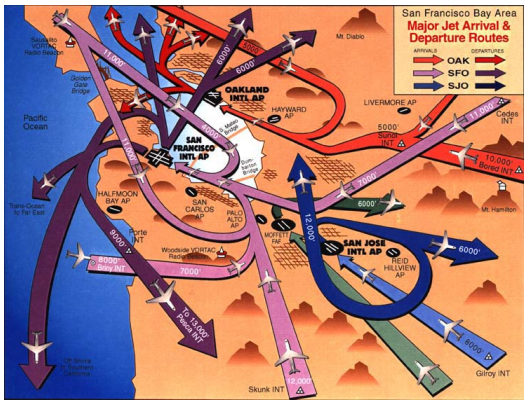


Figure 11: NCT standard traffic patterns, west configuration, image courtesy of Federal Aviation Administration

3.2.3 Definitions

This study focuses on the arrival process at SFO. Hence, in this chapter, the term *system* will refer to the TRACON area which is a 50 NM radius cylinder centered on Oakland Airport and of height approximatively 20,000 ft. The *inputs* to the system are the aircraft entering the TRACON, intending to land at SFO on 28L/R runways (Figure 10), and the *outputs* are the aircraft actually landing on runways 28L/R at SFO. Runways 28R and 28L are only 750 ft apart which does not allow simultaneous parallel instrument landings (FAA rules require a minimum 4,300 ft runway separation for simulateneous independant parallel landings) [35]. Hence, if the weather is not good enough for VFR landings, only one runway can be used.

Different parameters can be used to define the behavior of the system. The evolution of the outputs (landing traffic) can be analyzed as a function of the inputs (traffic entering the TRACON) and parameters such as the number of runways in use. This behavior is also characterized by the shape of aircraft trajectories as a function of the same parameters. Each aircraft has a different track leading to the runway. The analysis is made on flattened trajectories (2D trajectories on the x-y plan). Track are defined as *straight* or *rerouted*. A *straight* track corresponds to the trajectory of an aircraft that could go directly from its entering point in the TRACON to the runway threshold just following the way points. A *rerouted* track corresponds to the trajectory of an aircraft that can not land directly

and has to change its trajectory. It does not follow the nominal track and must follow a curved trajectory or even holding patterns. Appendix A presents two trajectory clustering algorithms to automatically classify trajectories. Figure 12 depicts several trajectories where rerouting was identified.

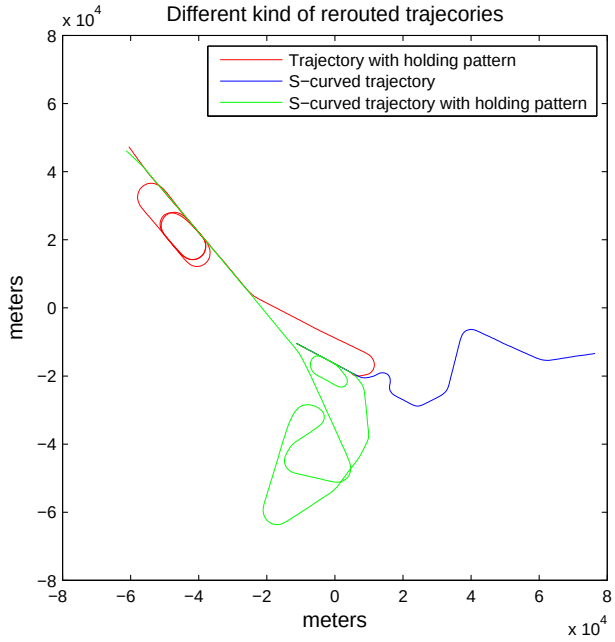
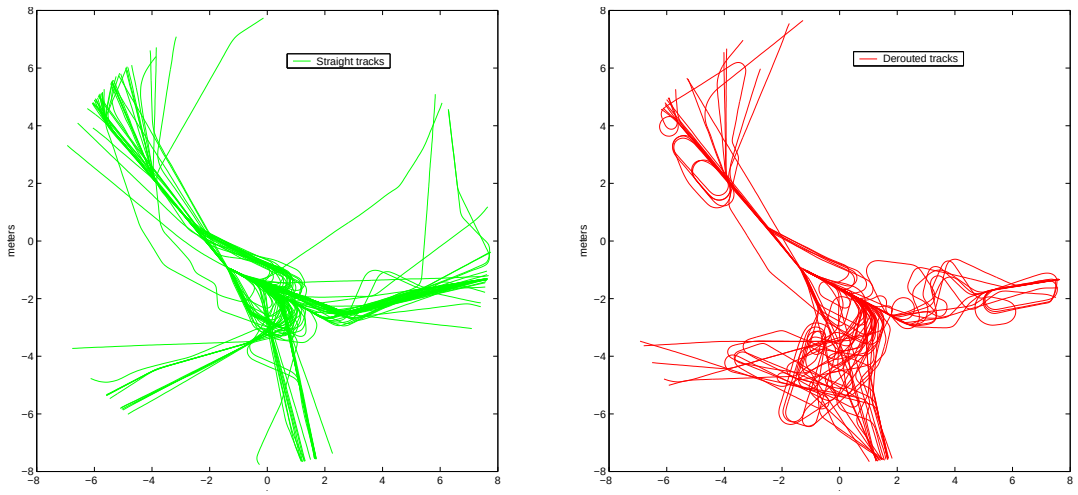


Figure 12: Different kinds of rerouted trajectories

3.3 Radar track analysis

To evaluate the behavior of the system, each track is analyzed in order to determine whether it is a direct track or a rerouted one. To identify rerouted tracks, an algorithm for trajectory clustering was developed. The algorithm is presented in appendix A. The rerouted trajectories were then identified as outliers. Figure 13 presents the results of this algorithm on a given day of arrivals at SFO (from 6 AM to 4 PM).

Figure 13(a) presents the tracks identified as direct tracks. Except some minor vectoring, direct tracks head directly to the runway, following what is identified as typical operations in appendix A. Figure 13(b) presents the tracks identified as rerouted. All those tracks



(a) *Direct* tracks (b) *Rerouted* tracks

Figure 13: SFO tracks analysis

contain a change of direction not corresponding to the route the aircraft should follow for a nominal approach. The rerouting is done either by smooth changes of trajectory, e.g *S-turns* corresponding to vectoring, or by important changes of trajectory, such as a holding pattern (Figure 12). In the next section, the evolution of SFO arrival process is investigated in two different ways. The first method is a time-based analysis and the second method is an aircraft-based analysis.

3.3.1 Time-based method

In this method, the time is divided in periods of fixed length during which the following elements are counted:

- The number of aircraft present in the system which corresponds to the number of aircraft that are, at some point of this period, in the TRACON area (Figure 15(a)).
- The number of aircraft entering the system during this period.
- The number of aircraft landing during this period.
- The number of aircraft rerouted during that period.

It can be noted that the same aircraft can be counted as present and rerouted over several time periods but enters or lands only once. It is said that only one runway is in use if

during this time period, aircraft landed more than 75% of the time on the same runway. Otherwise, two runways are in use

Figure 14 presents the results of the time-based analysis. Time is on the x axis. The positive y axis is a number of flights while the negative y axis represents the fraction of the total number of aircraft in the system that is rerouted, scaled from 1 to 10. Each bar corresponds to a 15 min time period. The light blue bars represent the total number of aircraft, the dark blue bars represent the number of aircraft entering the system, the red bars represent the number of rerouted aircraft. Yellow indicates that only one runway is in use while green stands for two runways.

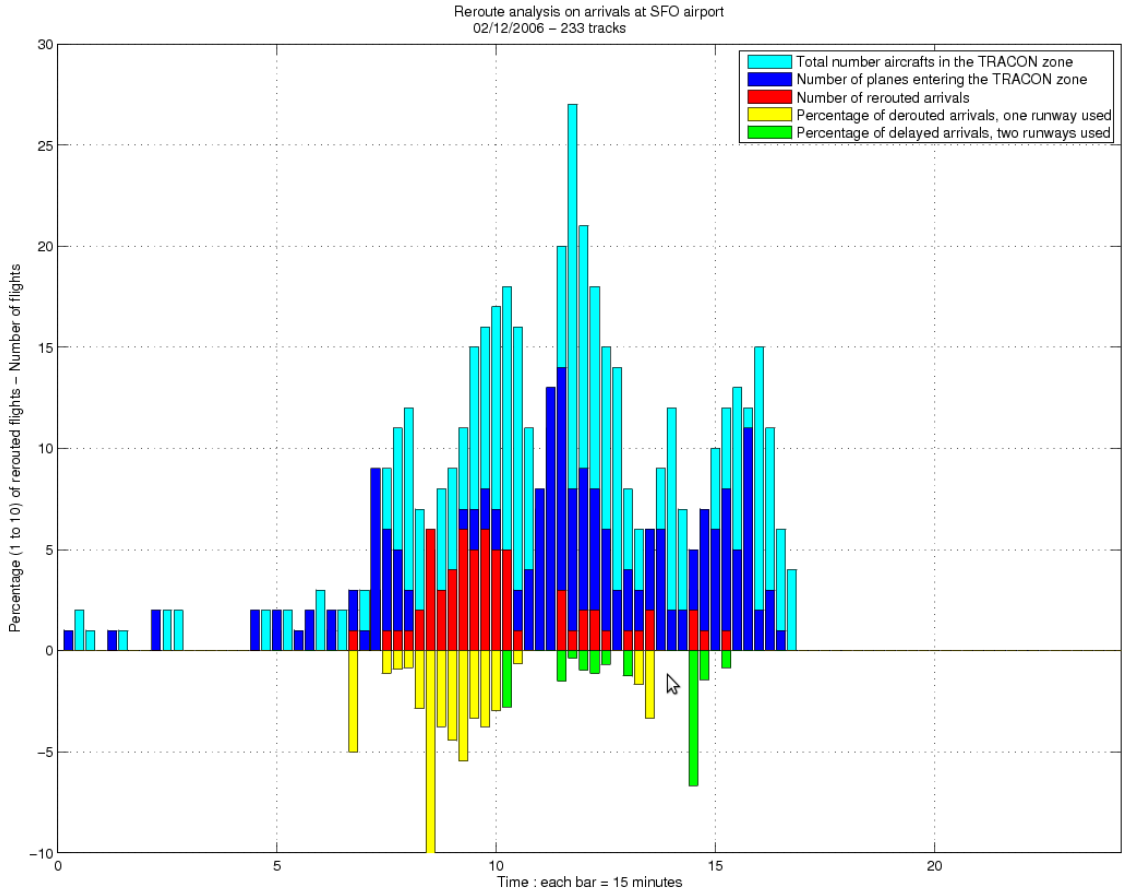


Figure 14: Arrivals analysis: February 12, 2006, local time

As shown in Figure 14, in case of one runway in use, the number of rerouted aircraft increases with the number of aircraft in the system. As soon as the second runway opens,

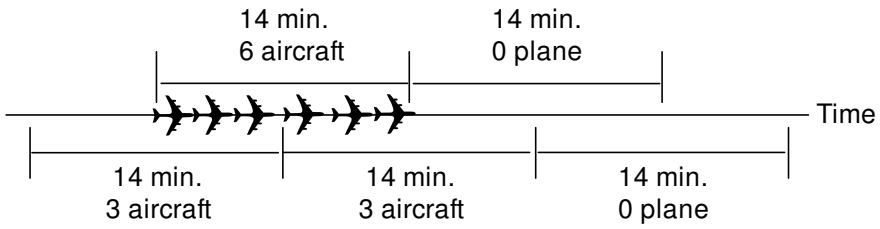
the number of rerouted aircraft drops. In the two-runway configuration, even a peak of incoming aircraft or a large number of aircraft present in the TRACON does not increase the number of rerouted aircraft significantly. This method is more developed and used to show the benefits of the McTDMA, in the paper of Idris and Evans [63]. The results they present intend to show the benefits of McTMA and hence, take in account a much larger traffic: they care for the demand to the airport and not only the aircraft currently in the TRACON or soon entering it. In the case presented in Figure 14, there is no arrival after 5 PM because the airport configuration changed.

3.3.2 Aircraft-based method

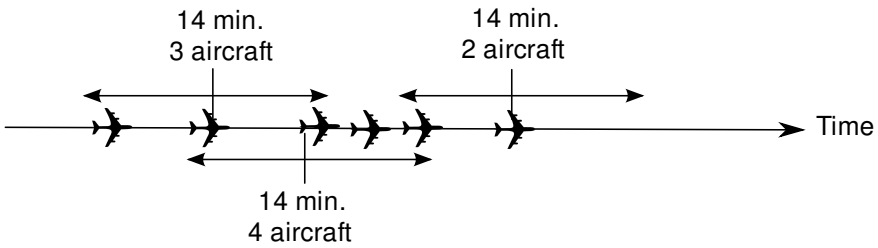
The method presented in this section is based on an "aircraft-centric" point of view. The flow of entering aircraft, the flow of landing aircraft, the number of aircraft present in the system, and the number of aircraft rerouted are looked at. Those terms need to be well defined. To calculate flows, a time period of length T is required. The term "for each aircraft" means that the count has been done for all the aircraft contained in the data set used. The total number of selected aircraft is around 95,000. Figure 15 presents a comparison of counts for time-based methodology and the aircraft-based methodology.

- For each aircraft, the number of aircraft present in the system corresponds to the number of aircraft that are, at some point, physically present in the TRACON at the same time, as showin in Figure 15(b).
- For each aircraft, the flow of entering aircraft is equal to the number of aircraft entering the TRACON during a period of length T centered on the time at which this aircraft entered in the TRACON, divided by the period length T.
- For each aircraft, the flow of landing aircraft is equal to the number of aircraft landing at SFO during a period of length T centered on the landing time of this aircraft at SFO, divided by the period length T.
- For each aircraft, the number of rerouted aircraft that is equal to the number of aircraft present in the system that have a *rerouted* trajectory.

- For each aircraft, the number of runways used corresponds to an average over a period of length T centered on the aircraft landing time. When one runway is used more than 75% of the time during this period, it is said than only one runway is in use, otherwise, two runways are in use.



(a) Time-based method to count the number of aircraft present in the TRACON



(b) Aicraft-based method to count the number of aircraft present in the TRACON

Figure 15: Comparison of time-based method and aircraft-based method

Figure 16 shows the average flow of entering aircraft in the TRACON (green diamonds), the average flow of landing aircraft (red crosses) and its standard deviation (vertical red lines), and the average number of rerouted aircraft (blue stars), as functions of the number of aircraft in the TRACON and of the number of runways in use. The blue line represents the data frequency (i.e the number of occurence of this configuration). The length of the period chosen is $T = 15$ min, to match the average time spent by aircraft in the system which is 14 min 30 s. The following presents an analysis of the TRACON characteristics as a function of the number of runways in use:

One runway in use: Up to 13 aircraft present in the TRACON, the flows of entering and landing aircraft are very close and proportional to the number of aircraft in the system. The average flow of entering aircraft is larger than the average flow of landing

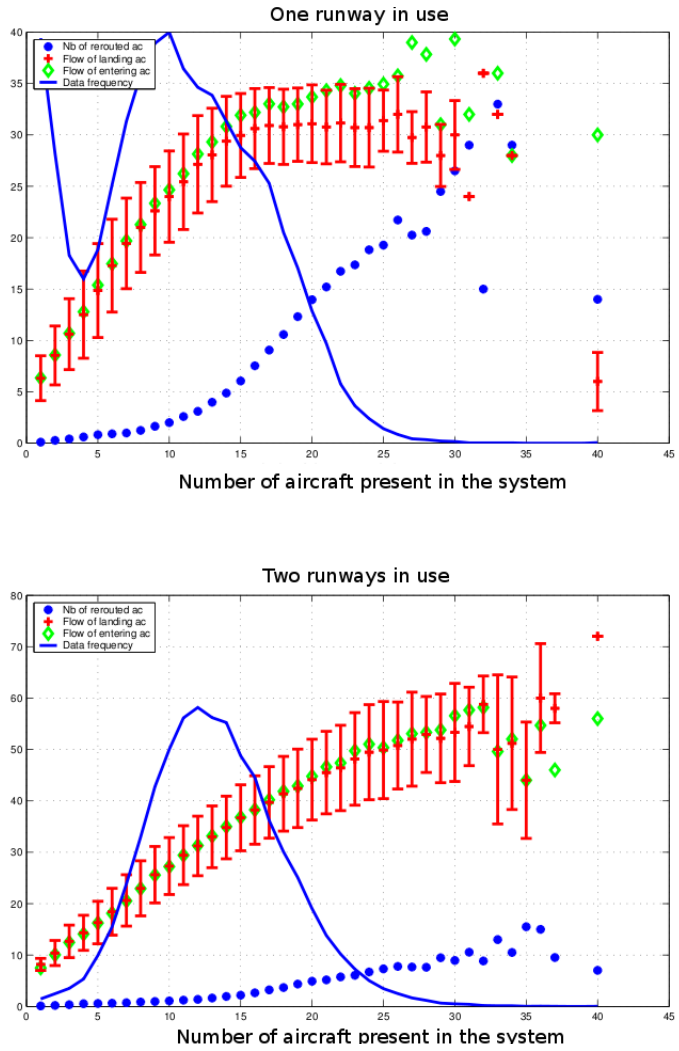


Figure 16: Aircraft-based analysis of landing and rerouting

aircraft. Past 14 aircraft present in the TRACON, the flow of landing aircraft remains at 31 aircraft per hour whatever the number of entering aircraft and the number of aircraft in the TRACON during the 15 min period. Up to 12 aircraft present in the TRACON, the number of rerouted aircraft increases slowly. Past 13 aircraft present in the TRACON, this number increases quickly and reaches 20 rerouted aircraft for 25 aircraft present in the TRACON. Past 25 aircraft in the system, the number of rerouted aircraft stabilizes around 20. The cases over 25 aircraft present in the TRACON are isolated cases (they happened once or twice in 9 month). The standard deviation of the number of landing aircraft starts at 2.2 aircraft/hour and is almost all the time around 4.5 aircraft/hour. In the modeling of the departure process by

Pujet [116], the standard deviation for throughput is over 7 aircraft/hour.

Two runways in use: Up to 29 aircraft in the system, the flow of entering aircraft and the flow of landing aircraft are almost the same . The number of landing aircraft increases almost proportionally with the number of aircraft in the TRACON. When there are more than 29 aircraft in the TRACON, the data frequency is very low (isolated cases) so the results are not very relevant. The number of rerouted aircraft increases slowly and reaches at the maximum 10 rerouted aircraft when 30 aircraft are present in the system.

3.3.3 Airport characteristics

From figures 14 and 16 some characteristics of SFO can be extracted. First of all, the analysis indicates that a single runway capacity a slightly exceeds 30 aircraft per hour. This can be explained by the fact that the second runway could have been used a few time during the period, and by the fact that the capacity of 30 aircraft can be over-achieved by spacing the aircraft slightly closer than the regulation allows. In addition, the spacing time between aircraft depends on their size. Therefore, a sequence of small aircraft will land in a shorter time than a sequence of alterning large and small aircraft. The time required for a small aircraft to land after a large aircraft is longer than the time required to land after a small aircraft. Observe that when there are more than 15 aircraft in the TRACON, the throughput saturates. When this limit is reached, the number of rerouted aircraft increases very fast. The runway occupation is fully optimized and hence, aircraft have to be rerouted to arrive at the exact time they have been assigned by the controllers. When the number of aircraft entering the system is high, most of them have to be rerouted.

In the two-runway configuration, whatever the number of aircraft in the system, the maximum airport arrival capacity is hardly reached. Hence, the number of rerouted aircraft is not high. This configuration can handle a peak of arrivals or a large number of aircraft in the TRACON without having to reroute hardly any aircraft.

3.3.4 TRACON characteristics

From the analysis of figures 14 and 16, it is clear that once more than 15 aircraft are in the system, the maximum landing capacity is reached. Allowing more aircraft in the TRACON only leads to more aircraft rerouting. This increases the TRACON controller workload, increases the density of aircraft and the airspace complexity and decreases the available rerouting options. This notion of airspace complexity for terminal areas is further developed in Chapter 5, section 5.3. While a minimum number of aircraft in the TRACON is required to reach the maximum average landing capacity, exceeding that number, decreases the TRACON safety and does not improve airport performance.

3.3.5 Pros and cons of both methods

3.3.5.1 Time-based method

Pros This method is easy to understand and to interpret. It is a common way to analyze data. As shown in figure 14 the evolution of the number of rerouted aircraft as a function of time is very clear.

Cons The time period is given and the start of the first period is chosen arbitrarily: it is the first minute of the first day of records. Since it is arbitrary, some boundary effects can appear and change the calculated values slightly. Figure 15(a) shows the impact of this chosen time period.

3.3.5.2 Aircraft-based method

Pros The aircraft counted in the system correspond to the aircraft that interact which each other, that is all the aircraft flying simultaneously in the TRACON. This has a real meaning when talking about density or complexity. This method grants a very easy comparison of the results with a discrete event model. The analysis is exactly the same for all the aircraft.

Cons For the number of aircraft in the system, only the aircraft flying in the TRACON are taken into account: it is assumed that aircraft that have already landed and aircraft

outside the TRACON do not have any influence on airborne aircraft.

3.4 Model description, calibration and validation

This section presents the developed model of the TRACON. One of the objectives of this model is to capture the phenomena observed in TRACON. This queueing model presents similarities with earlier push-back and departure process model [116].

3.4.1 Description of the model

This model is a discrete time input-output model implemented using MATLAB.

System: The modeled system is the TRACON together with two parallels runways. In nominal configuration (i.e. the two runways are open and simultaneous landing are possible), the maximum landing capacity is 60 aircraft per hour, 30 for each runway.

Inputs: The input of the system is the sequence of entering aircraft in the TRACON.

Outputs: The output of the system is the sequence of landing aircraft on the two runways.

Time: The time is divided into 30 s periods.

Travel time: When an aircraft enters the system, a *nominal* travel time is randomly drawn with a probability distribution describing the configuration of the TRACON (i.e the different possible routes leading aircraft from TRACON entry to the runway threshold). This nominal time corresponds to the time the aircraft would spend in the TRACON if the runway were not congested (Section 3.4.2).

Runway closure: To degrade the system, it is possible to close one or both runways. If a runway is closed no aircraft can land on it.

Runway availability: Once an aircraft lands, the runway is unavailable for 2 minutes, that is 4 time periods.

Runway attribution: When an aircraft enters the system and is given a nominal travel time, drawn from a random distribution (Section 3.4.2), it requests the runway for the 2 minutes period corresponding to the end of its travel time. If the runway is

available, it can land and the runway becomes unavailable for this period. If no runway is available for this period, the aircraft must enter a queue, modeling the actual vectoring or rerouting.

Queuing: If the runway is not available, the aircraft waits until one runway is available. As soon as one runway is available, the available slot is attributed to the aircraft.

3.4.2 Parameters for the model

This model is generic and can be used for any TRACON, but requires calibration for every specific TRACON. The required parameters are the capacity of the runway(s) and the average nominal time spent in the TRACON before landing. For the average nominal time spend in the TRACON, the probability distribution used to draw nominal travel times is presented in Figure 17. The gray shaded bars represent the travel time distribution for unimpeded trajectories extracted from the first 8,000 flights of the data set (out of 81,000) and the red line represents the probability distribution used for the model.

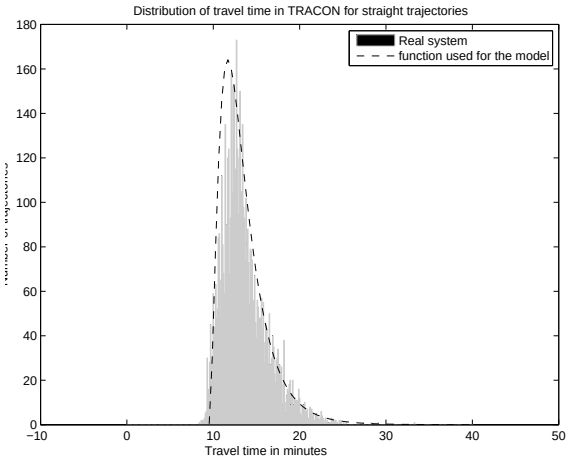


Figure 17: Distribution of travel time in the TRACON

3.4.3 Model validation

To validate the results of this model, the dataset used is the one presented in section 3.2, excluding the 8,000 trajectories used for calibration. The sequence of entering aircraft was extracted and used as input to the model. The number of open runways has been extrapolated from the landing sequences in the data.

Figure 18 presents a simulation of the modeled system over two days. The top graph presents the entering sequence, the second graph presents the runway occupancy: the dots indicates the slots when the runway was occupied. The first runway, which is always open, is indicated in red with value 1 and the second runway is indicated in blue with the value -1. The green line presents the period where the second runway is open. The third graph presents the evolution of the system. The blue line represents the number of aircraft in the system and the red line represents the number of rerouted aircraft. The x axis is time, in 30 seconds time intervals. One day contains 2,880 time intervals of 30 seconds.

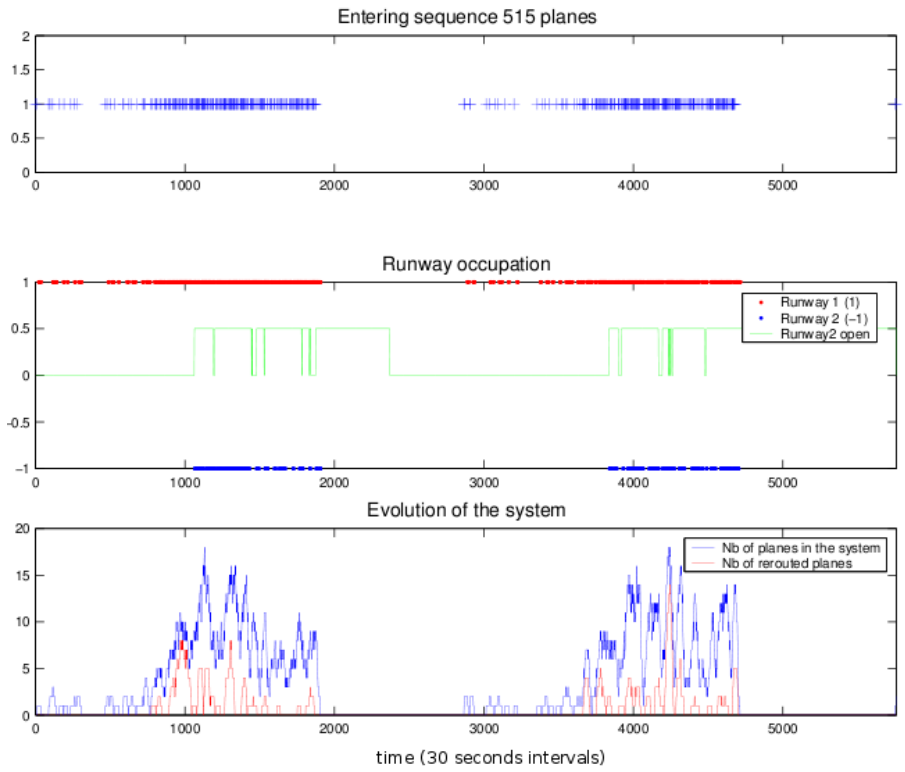


Figure 18: Evolution of the modeled system simulation over 2 days.

For Figures 19 and 20, 250 days were simulated, corresponding to 67,218 aircraft, 22,428 landings when one runway is open and 44,790 landings when two-runways are open.

Figure 19 presents exactly the same analysis as in section 3.3.2 for the simulated model. The figure plots the average flow of entering aircraft in the TRACON (green diamonds), the average flow of landing aircraft (red crosses) and its standard deviation (vertical red lines), and the average number of rerouted aircraft (blue stars), as functions of the number of aircraft in the TRACON and of the number of runways in use. These terms are definded in section 3.3.2. The continuous line represents the data frequency. For the model, the period chosen is 16 minutes to capture the real flow of this discrete model.

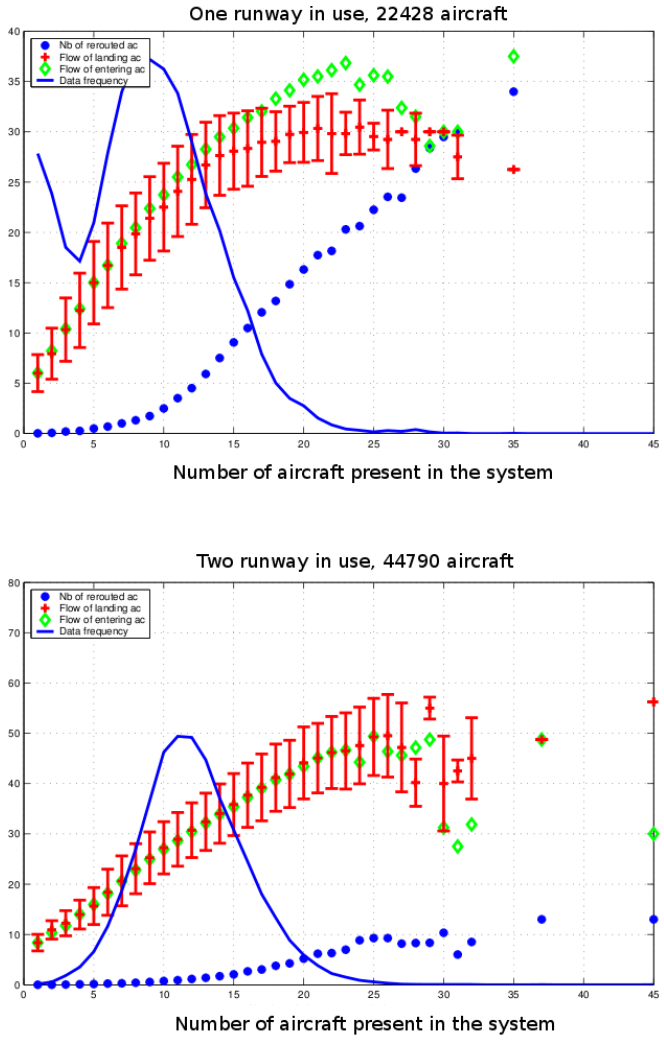
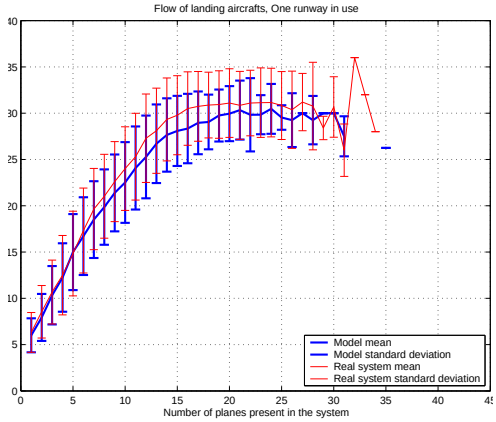
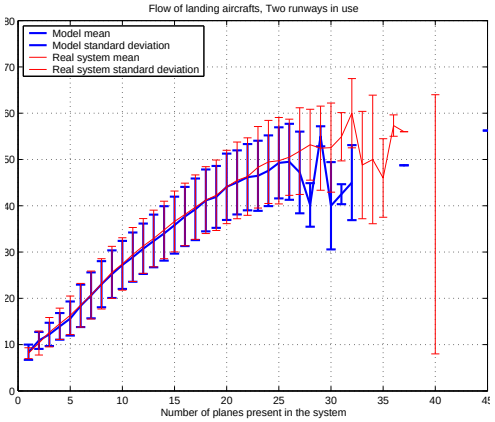


Figure 19: Aircraft based analysis of landing and rerouting on the model

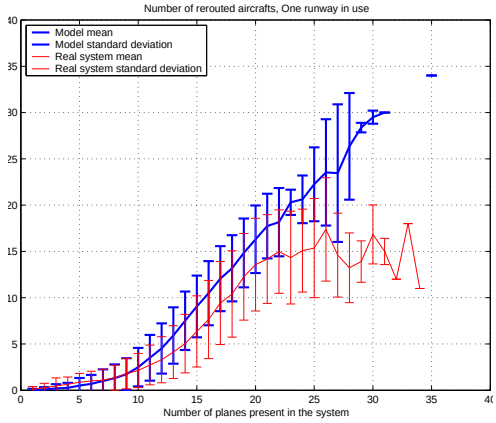
Figure 20 presents the comparison between the model and the actual system. Figures 20(a) and 20(b) present the comparison of the landing flow of the model with the real system. The blue line represents the average flow of landing aircraft as a function of the number of aircraft in the system for the model and the red line presents the same information for the real system. The vertical lines represent the standard deviation. Figures 20(c) and 20(d) present the comparison of the landing flow for the model with the real system. The blue line represents the average flow of landing aircraft as a function of the number of aircraft in the system for the model and the red line presents the same information for the real system. The vertical lines represent the standard deviation.



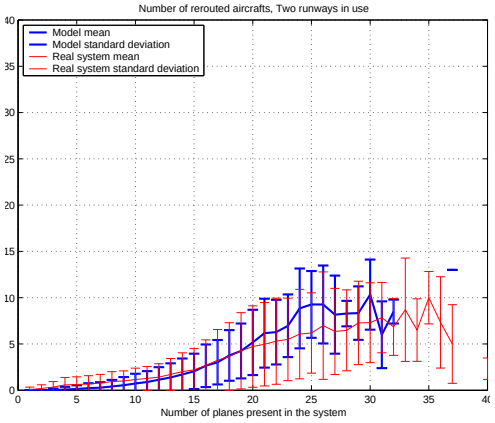
(a) Flow of landing aircraft, 1 runway



(b) Flow of landing aircraft, 2 runways



(c) Number of rerouted aircraft, 1 runway



(d) Number of rerouted aircraft, 2 runways

Figure 20: Comparison of landing and rerouting between the model and the real system

As shown in figures 20(a) and 20(b), the behavior of the model is very close to the real system, as far as landing aircraft flow is concerned. When one runway is in use, the model predicts a flow slightly lower than the real system. When the number of aircraft in the system is high, the average throughput is 30 landings/hour, which corresponds to the regulation. Notice that the standard deviation of the capacities obtained by the model and measured in the real system are always very close. When two runways are in use, the model predicts exactly the same behavior as that of the real system up to 25 aircraft in the system. Past 25 aircraft, there are some discrepancies. However this situation is infrequent and represent only $> 0.9\%$ of total flights.

In the one-runway configuration, figure 20(c) shows that the predicted number of rerouted aircraft is pretty accurate up to 10 aircraft in the system. Between 10 and 22 aircraft in the system, prediction are slightly higher than the real system. Over 22 aircraft in the system, the prediction keeps increasing at the same rate while in the real system, it stabilizes. On the two-runway configuration, figure 20(d) shows that the prediction are accurate up to 20 aircraft in the system. Past 20 aircraft, it is slightly higher.

3.4.4 Analysis of the model results

The maximum number of aircraft present in the actual system exceeds 34 while it never goes higher than 31 in the model. An explanation could be that there always is one runway open in the model, whereas in the real system both runways can be closed simultaneously. The model does not take in account weather constraints. It suffices that both runways be closed during a few minutes and then both reopen to drastically increase the number of rerouted aircraft, present in the system. In the model, one runway is always open to remove the traffic congestion. In one-runway configuration, the model predicts the number number of rerouted aircraft to increase with the number of aircraft in the system. In the real system, it stops increasing. In the real system, only trajectory shapes changes were accounted for: speed changes, which are also used by controller to regulate traffic were not taken into account. This difference can also be explained by the fact that in the model, when there is one runway in use, the second runway can absolutely not be used, while in

the real system it can account for up to 25% of the landings. In the actual system, even when the weather does not permit to use parallel landings, runways might be alternated to increase capacity by slightly reducing time intervals between landings.

3.4.5 Possible model evolutions

- Once an aircraft lands, the time when the runway is unavailable is always exactly two minutes. The aircraft type (light, medium, heavy) is not taken into account in this model. In a future model, the unavailability time could be dependent on the aircraft's type. The size of the aircraft would be randomly drawn with a probability law based on the real distribution of aircraft.

- Queuing and landing methods are very deterministic. When a plane is queuing, as soon as the runway is available, the plane lands. It does not take in account the dynamic of the aircraft and any event that could impact the precision on the landing time. A small uncertainty could be added on the moment a plane lands [14].

- The model does not include different routes in the TRACON. The time spent in the TRACON is randomly drawn with a certain probability law. This could be more representative if the entering sequence were to include several entry points corresponding to different routes with different travel time probability distributions for each route. Some routes could be longer or more frequent.

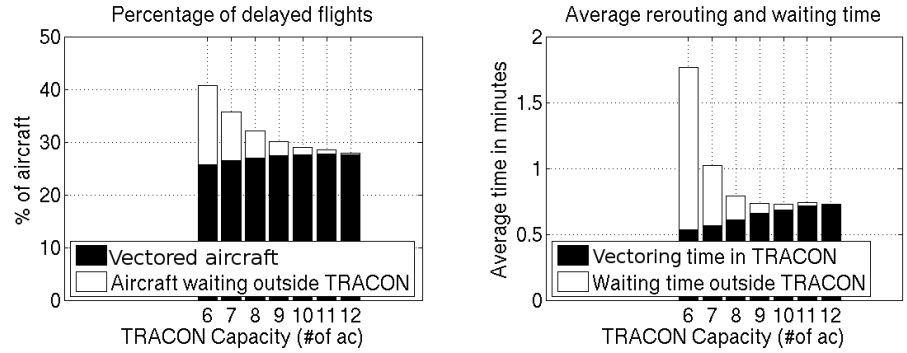
3.5 TRACON operations sensitivity to landing capacity degradation

A study about the impact of the landing demand on landing capacity[63] for Newark Airport shows that when demand is higher than the capacity of the runway, it results in a drop of runway capacity. Plausible factors can be the application of traffic flow restrictions and holding. Other factors may include a high workload of air traffic controllers and airspace complexity constraints.

To measure the sensitivity of various traffic configurations on congestion and delays, arrival simulations have been made using San Francisco demand sequence. The traffic conditions studied are generated by modifying the TRACON queueing policy: The TRACON

has been given a limited capacity: no aircraft can enter it if this limit is reached. As soon as an aircraft lands, one aircraft waiting outside can enter the TRACON. *Waiting* time refers to the time the aircraft were asked to wait outside the TRACON and *rerouting* time corresponds to the time the aircraft were asked to wait for the runway to be available before landing. The queue outside the TRACON is a First-In First-Out queue. The simulations cover 243 days and 66,067 aircraft.

Figure 21 presents the results of simulation for TRACON limit capacity varying from 6 to 12 aircraft. Black bars stand for rerouting values and white bars for waiting waiting. Figure 21(a) presents the average delay and figure presents the percentage of delayed aircraft 21(b).



(a) Mean rerouting and waiting time (b) Percentage of delayed flights

Figure 21: Simulated delays function of the number of allowed aircraft in the TRACON

Setting a low TRACON capacity has two effects. First, the number of aircraft waiting outside the TRACON is very high and second, the runway capacity will not be reached, which results in an inefficient configuration. Then, when the number of aicraft allowed in the TRACON increases, the mean rerouting time decreases and stabilizes while the percentage of delayed flight slightly decreases. When the allowed number of aircraft in the TRACON varies from 9 to 12, the average rerouting time is almost the same, but, as the number aircraft in the TRACON is smaller, the complexity of the system is smaller. The average rerouting time does not change much, because the runway's maximum capacity is reached and aircraft have to queue anyway. As shown in section 3.3.2, past a certain number of aircraft in the TRACON, the number of rerouted aircraft increases very quickly. If the

number of simultaneous rerouted aircraft is high, the controller workload increases. An analysis of the complexity related to rerouting and holding patterns is presented in Chapter 5. By reducing the controller workload, the controller could focus on fewer aircraft and better optimize the trajectories, which would result in an increase of the runway capacity. Moreover, the higher the number of present aircraft in a given area, the less the flexibility for an efficient rerouting process is. TRACONs are very congested areas and limiting the number of aircraft present in those areas would increase the safety without generating extra delays. This work is a complement to the TMA, which regulates the flows of aircraft in the TRACON. This section provides bounds on the number of aircraft TMA should allow in the TRACON.

3.6 Summary

This chapter focuses on the operational impact of a degradation of the landing capacities at SFO [44]. For that purpose, an input-output queueing model for the NCT is presented, calibrated and validated using available radar data. The model accurately predicts the landing capacity at SFO when the airport has either one or two runways in use. Then, the model shows the influence of the number of aircraft simultaneously present in the TRACON on the amount of rerouting that takes place within the TRACON.

The impact of a landing capacity reduction on traffic configurations was studied by simulating the arrival process with different bounds on the TRACON capacity. It was shown that allowing a large number of aircraft to enter the TRACON does not lead to increased airport performance and may create the conditions for reduced airspace safety margins. Past 9 aircraft, an increase in the number of aircraft simultaneously present in the NCT does not have a positive effect on the delays encountered when one runway in use. Limiting the number of aircraft in the TRACON does not increase average aircraft delay but reduces the controller workload and the complexity of the traffic.

By analyzing various queueing policies, this chapter answers RQ 3, that is "What is the sensitivity of traffic configurations to a potential degradation?", in the case of a reduction in landing capacity. It was shown that a reduction in landing capacity impacts the tracks

of aircraft, because of vectoring and holding patterns. This rerouting has an impact on airspace safety as it becomes more crowded, leaving less margin for errors. Then, RQ 4, that is "How can the smooth transition from nominal mode of operations to degraded mode of operation be ensured?" is answered by determining the optimal TRACON capacity.

CHAPTER IV

OPERATIONAL DEGRADATION CASE STUDY 2: DEGRADATION OF CNS SYSTEMS

4.1 Introduction

This chapter addresses RQ 3 and RQ 4, that is, "What is the sensitivity of a traffic configuration to potential degradations?" and "How can the smooth transition from nominal mode of operations to degraded mode of operations be ensured?", respectively. This chapter provides an answer to those questions for a particular category of degradation. The potential degradation of interest in this chapter encompasses degradations affecting the communication, navigation and surveillance systems (CNS). After modeling the operational impact of such failures, different traffic configurations are probed. The model proposes a smooth transition from nominal to degraded mode of operations by increasing the required spacing distance bewteen aircraft. Finally, solutions to ensure the graceful degradation will be proposed using an algorithm of avoidance under uncertainties in 3 dimensions.

Conflict resolution in the presence of uncertainties is the object of several papers. Kuchar and Yang present several models to account for uncertainties in their review of conflict detection and resolution modeling methods [74, 75]. They reviewed over 68 conflict detection and resolution algorithms. Figure 22 presents the three fundamental trajectory extrapolation methods used in conflict detection and resolution, and identified by Kuchar and Yang. In the nominal case (Figure 22(a)) the current states are projected into the future along a single trajectory, without direct consideration of uncertainties. In most cases, a nominal trajectory model may be quite accurate but does not account for the possibility that an aircraft may not behave as expected. It also does not account for position reporting errors. The other extreme of dynamic modeling is the "worst case" projection (Figure 22(b)). From an initial position and heading, the model assumes that the aircraft could perform any of a range of maneuvers. If any of these maneuvers could cause a conflict, then a conflict

is predicted. In the probabilistic method (Figure 22(c)), uncertainties are modeled to describe potential variation in the future trajectory of the aircraft. This model is used in the URET [11] and CTAS [95], for example. A position error is added to a nominal trajectory, from which the conflict probability can be derived [11, 67]. Another approach is to develop a complete set of possible future trajectories, each weighted by a probability of occurring (e.g. using probability density functions). The trajectories are then propagated into the future to determine the probability of conflict [61, 84, 112].

(a) Nominal (b) Worst case (c) Probabilistic

Figure 22: Model of uncertainties

Erzberger et al. [30] also proposed a method for conflict detection and resolution in the presence of uncertainties. The uncertainty on the trajectory is modeled using growing ellipsoids centered on the aircraft's position. The initial circles of avoidance are projected forward in time and become ellipsoids. The major axis of the ellipsoid is aligned with the velocity vector and the minor axis is perpendicular to it. This accounts mainly for an error on the speed of the aircraft. Figure 23 presents the propagated ellipsoids for two aircraft. To detect conflicts, a coordinate transformation is used resulting in an ellipsoid and an error circle. The probability of conflict of the aircraft is computed using the volume of the intersection of the ellipsoid and the error circle.

An approach to conflict resolution in the presence of uncertainties using dynamic programming is proposed by Blin et al. [10]. This formulation tries to find the "optimal" time to maneuver and the best maneuver possible to avoid conflict. This algorithm finds the "optimal" maneuvers for an aircraft with a limited knowledge of other aircraft trajectories. The uncertainty not only depends on the aircraft's position, but on their intent and possible

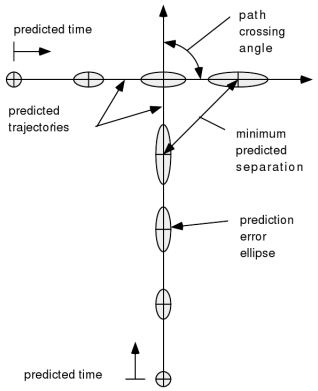


Figure 23: Uncertainty modeling by Erzberger et al. [30]

change of flight plan.

Literature on conflict resolution in the presence of uncertainties is pretty rich, but none addresses the issue of a degradation in the CNS system. A degradation in the navigation system might result in a slight deviation from the intended trajectory. A degradation of the surveillance system would result in a decrease of accuracy in aircraft positions reporting.

The focus of this chapter is on the operational impact of degradation of communication, navigation and surveillance systems. As seen in Chapter 2, it is difficult to accurately measure the impact of a failure in one of those systems on operations. A model of the operational impact of such a degradation is proposed. This model is conservative as it encompasses many types of degradations affecting CNS systems. This model is then used in section 4.3 to compare the sensitivity of two types of operations for arrivals: Free Flight versus Miles In Trail. The probe used to compare those traffic configurations is a quasi-optimal algorithm of avoidance in the presence of uncertainties, presented in section 4.4. Then, a 3 dimensional version of the algorithm of avoidance in the presence of uncertainties is introduced in section 4.5. The objective of this algorithm in 3D is to provide controllers with a simple set of avoidance maneuvers in the event of a loss of performance in the CNS system. The goal is not to optimize all avoidance maneuvers, but rather to focus on airspace safety and to be able to handle a large number of aircraft simultaneously. A set of maneuvers that solve all the potential conflicts in the case of growing radii of avoidance is first determined and then, a single maneuver for each aircraft is chosen by formulating and

solving a mixed integer program. The cost of each maneuver depends on the imminence of the conflict.

4.2 Modeling the degradation of CNS systems

4.2.1 Degradation of CNS systems

The model presented in this chapter focuses on the impact of a degradation in the CNS systems on aircraft separation requirements. Indeed, the primary mission of the air traffic control system is to ensure safe aircraft separation under all regular and degraded circumstances. Conventional surveillance systems include radar-based technology, such as primary and secondary radars, and ground-based beacons. New, higher-resolution surveillance systems are enabled by satellite-based positioning systems. Such technologies may enable reduced horizontal separation minima and therefore higher airborne aircraft densities. In the ontology presented in Chapter 2, a validation case involving the failure of a radar was studied and the impact on operations was highlighted.

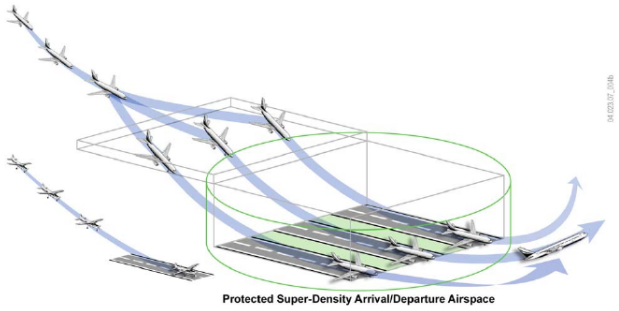


Figure 24: Super-Density operations [70]

The principle that drives this analysis can be sketched as follows: Considering a set of aircraft operating under a "high performance" surveillance system, can safety be maintained despite a failure of the surveillance system?

Assuming that failures of the surveillance system consist of a partial loss of vehicle coverage, maneuvers that will allow aircraft to remain provably separated will be computed. Indeed, partial or complete loss of aircraft position coverage results in growing uncertainty about aircraft positions, in such a way that aircraft initially close to each other may not be distinguishable from each other shortly after the failure, unless they maneuver to augment

their physical separation. Thus, underpinning the analysis of the ability for a particular airborne configuration to gracefully degrade, it is necessary to design procedures enabling traffic to maintain provable separation under degraded conditions. The ability for such configurations to gracefully degrade is measured using a conflict resolution strategy under degraded conditions. In this chapter, such procedure will consist of a novel aircraft conflict management algorithm.

Considering a planar traffic environment for simplicity, nominal and degraded operations are defined as follows:

1. **Nominal operations:** *Nominal operations consist of all allowable aircraft operations when the CNS systems are working properly. The nominal minimum aircraft separation distance (expressed in nautical miles) will be denoted* $2r_0$.

2. **Degraded operations:** *Degraded operations consist of all allowable aircraft operations when a failure has affected the CNS systems, and the reported position and intent of aircraft are more uncertain or exceed a given threshold of uncertainties. The degraded minimum aircraft separation distance (expressed in nautical miles) will be denoted* $2r_f$*, with* $r_f > r_0$.

Radar precision is one of the main reasons for deciding on specific aircraft separation standards. The uncertainty in the position seen by the controllers leads to a separation requirement that can be interpreted as a circle of avoidance around each aircraft. The circle of avoidance corresponds to the area around the aircraft where no other aircraft and its circle of avoidance is allowed. Its radius is generally 2.5 NM for en-route and 1.5 NM for approach. The separation is guaranteed if the circles around each aircraft don't intersect, resulting in 5 NM or 3NM separation minima. This distance ensures safety if the position of the aircraft is relatively well known and regularly updated. Accurate positioning systems such as ADS-B will probably enable a reduction in allowable spacing distance [110]. If a failure happens, the system works in degraded mode, resulting in an increase in uncertainties on the aircraft position observed by the controller. As aircraft positions are known less accurately, the resulting radius of avoidance must be increased. The growth of the avoidance circle is

limited by backup positioning systems (Primary Surveillance Radars, radio...) that enable controllers to get reports on aircraft positions. Such systems can be identified using the ontology presented in Chapter 2. For instance, in the case of a radar breakdown, separation distances must be increased to procedural separation standards [65]. The position of the aircraft will be reported by the pilot to the controller by radio with a low update rate. Between updates, the position of the aircraft is not known and its position uncertainty increases with time.

4.2.2 Algorithmic aspects

From these considerations, it is not difficult to create aircraft configurations which are conflict-free, yet will rapidly generate conflicts in case of positioning system degradation. The following conflict detection and resolution algorithm will be used as a probe to evaluate the configurations' sensitivity to degradation. This section presents an overview of the algorithm and mathematical details are explained in section 4.4.

In the remainder of this chapter, the surveillance system failure occurs at time $t = 0$ and the time of failure is known in real time. The resolution maneuver is an indicator for the severity of the situation. Prior the failure time, nominal aircraft position accuracy will translate into a circle of avoidance with radius r_0.

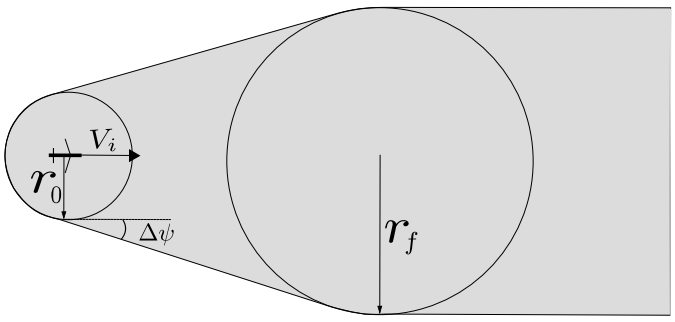


Figure 25: Track of a growing circle of avoidance

Past the failure time ($t > 0$), the model for position uncertainties is time-varying, whereby the radius of avoidance grows from r_0 to $r_f > r_0$ over a given period of time. Figure 25 presents the track of a growing circle of avoidance. For instance, r_0 can be the radius of avoidance provided by an ADS-B positioning system, while r_f can be the one

provided by a Primary Surveillance Radar (PSR) . In the event of an ADS-B failure, the transition from r_0 and r_f must be eventless, in the sense that the transition should not jeopardize the safety of the overall traffic.

For each aircraft i, a constant growth rate $\dot{r}_i$ is assumed, such that $r(t) = r_0 + \dot{r}_i t$. Such a model approximately captures the growing, but bounded uncertainty on aircraft position once the navigation system has failed. Such uncertainty might reflect the effect of uncertainties on the aircraft heading (denoted $\Delta\psi$) and on the aircraft velocity (denoted ΔV_i). Figure 26 shows the uncertainty on the trajectory. A simple way to connect these uncertainties to the growing avoidance radius is to write, for example, the following first order conservative approximation for $\dot{r}_i$:

$$\dot{r}_i \approx \sqrt{V_i^2 \sin^2 \Delta\psi + \Delta V_i^2}.$$

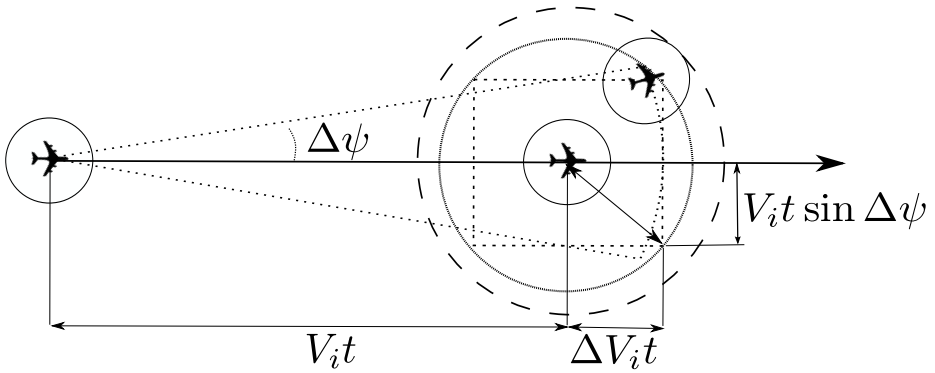
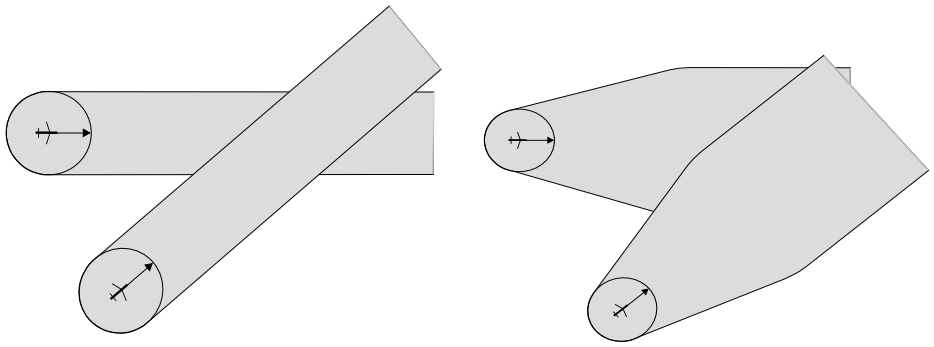


Figure 26: Uncertainty on the trajectory

Figure 27 shows the difference between a classical conflict avoidance problem and the problem under considerations in this paper. In the classical avoidance problem, the radius of avoidance is constant. In this problem, the growing radius of avoidance makes the formulation and resolution more complicated since it is time dependent.

When several aircraft are present within a given airspace sector, the conflict resolution problem under degrading position uncertainty becomes much more complex. An approach using mathematical programming based on a formulation originally presented by Pallotino et al. [103] is introduced in section 4.4. The novelty of the presented algorithm is the presence of growing radii of avoidance. It is not only a conflict detection and resolution,



(a) Nominal conflict avoidance problem (b) Conflict avoidance problem with growing uncertainties

Figure 27: Comparison of conflict avoidance problems

but it also manages an increases in spacing distances. The problem is formulated as a Mixed Integer Programming (MIP) and is implemented using the AMPL/CPLEX linear programming tool set [42, 64].

4.3 Free Flight versus Miles-In-Trail degradation analysis

The parts of airspace where the highest aircraft densities occur are the terminal areas in the vicinity of airports. Therefore, these constitute an ideal setting for evaluating the ability for traffic to undergo graceful degradation. This choice is also motivated by current vistas on future operations in terminal areas, which have been named "Super-Density operations" (NextGen) and "High Complexity Terminal operations" (SESAR). Both consist of increasing the airspace capacity around busy airports. A solution proposed in SESAR [125] is the multiple merge points arrival operation shown in Figure 28. One way to interpret this solution is to assimilate the new mode of operation to a "Free-Flight" scenario, whereby the route structure in the terminal area is relaxed and the only constraint on incoming aircraft is for them to meet a specific arrival time at the merge point. This scenario contrasts sharply with current airport arrival practices, where high-density arrival flows of aircraft are organized tens or even hundred of miles prior to landing, by lining up aircraft along arrival routes and spacing them appropriately. Such operations are often denoted Miles-In-Trail (MIT) operations.

For that purpose, a scenario considered involves 8 aircraft merging to a common point of

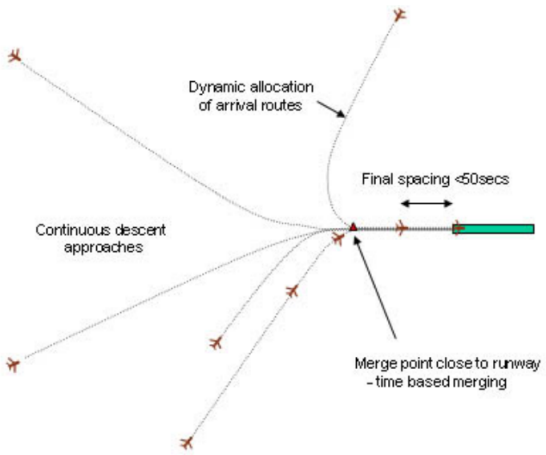


Figure 28: Arrival Routes, Multiple Merge Points [125]

coordinates $(0, 0)$. All aircraft are time separated: Each aircraft is given an arrival time so as to meet a precise and regular arrival rate at the merge point. Assuming all aircraft fly at the same speed of 200 kt, the aircraft must therefore be initially located on regularly spaced circles centered at the merge point, as shown in Figure 29. The inter-aircraft spacings are designed to emulate future Instrument Meteorological Conditions (IMC) arrival operations on closely spaced parallel runways, such as San Francisco Airport: As observed in chapter 3, the average interarrival time for each runway is slightly less than 2 minutes. This translates into a 3NM average separation between aircraft when they fly at 200 kt and the two runways are in use. This separation also turns out to be a very conservative estimate of separation requirements for satellite-based navigation systems [110], leading us to an initial circle of avoidance of radius $r_0 = 1.5$NM as proposed by ICAO for ADS-B[66]. The final radius of avoidance was chosen to be $r_f = 2.5$ NM to reflect the surveillance degradation that would occur, should a GPS-based surveillance fail and backup radar-based technology be used. The rate of growth reflects a heading uncertainty $\Delta\psi = 5°$, leading to $\dot{r} = 0.29$ NM/min. The transition time between r_0 and r_f is $T = 3.44$ min.

To evaluate the impact of changing arrival operations from Mile-In-Trail towards Free Flight, aircraft will fly on an "arrival cone" whose vertex is at the merge point. When the cone's angular width is zero or takes small values, it corresponds to highly structured Miles-In-Trail operations. When the cone's angular width is large, it corresponds to less

structured, Free-Flight-like operations.

To understand the impact of the cone angular width and the aircraft initial separations on traffic degradation, should a surveillance system failure occur, the angular width was varied from 10° to 55°. The initial aircraft separation, that is, the distance between two consecutive circles, was chosen to be 3.05 and 3.5 NM. A total of 1,650 cases with different arrival angles have been simulated. The following procedure has been used to generate the cases: Since the distance of each aircraft to the merging point is known (fixed separation distance), the aircraft initial heading was picked at random, using a uniform probability distribution in the allowed interval ($\pm 5°, \pm 10°, \ldots$). Figure 29 presents on the same diagram a MIT configuration and a FF configuration. The circles represent the avoidance circles (dashed/black = MIT, continuous/red = FF). They have initial radius 1.5 NM and are centered on the aircraft. The blue line shows the aircraft's heading. Aircraft are represented by a small circle for the MIT configuration and asterisks for FF configuration. The allowed arrival cone for the FF configuration is the shaded cone shown in Figure 29.

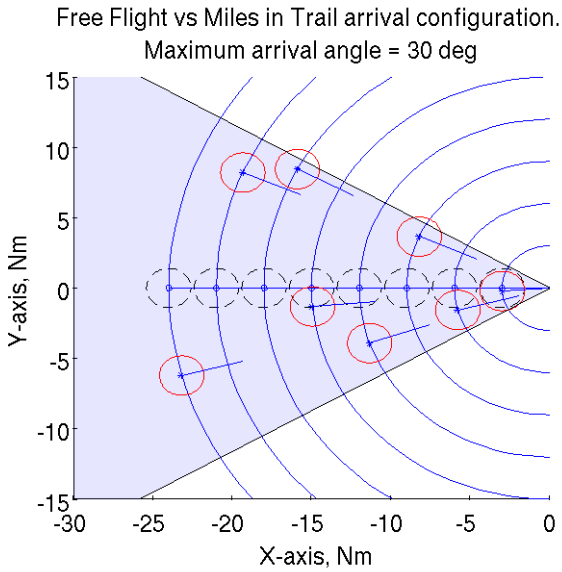


Figure 29: Free Flight and Miles-In-Trail arrival configurations

These traffic configurations have been probed by the conflict resolution algorithm under uncertainties developed in section 4.4. Let S be the set of all generated cases. The severity of the traffic management degradation on the traffic situation $s \in S$ was evaluated by

measuring the average deviation m_s required for each aircraft, denoting ψ_{is0} the initial heading of aircraft i and ψ_{is} its heading after resolution in this situation. The average deviation for a traffic configuration is

$$m_s = \frac{\sum_{i=1}^{n} |\psi_{is} - \psi_{is0}|}{n}, \tag{1}$$

where n is the number of aircraft. All the cases were then sorted by absolute value of the maximum aircraft arrival angle and grouped in parameter increments of 2.5°. Let S_k denote the following subsets of S:

$$S_5 = \left\{ s \in S \text{ such that } \max_i \{\psi_{is0}\} \in [0, 5), i = 1 \dots n \right\}, \tag{2}$$

$$S_k = \left\{ s \in S \text{ such that } \max_i \{\psi_{is0}\} \in [k - 2.5, k), i = 1 \dots n \right\}, k = 7.5, 10, \dots, 30. \tag{3}$$

Figure 30 presents the results of the analysis for the initial distances of 3.05 NM in blue and for an initial distance of 3.5 NM in red. The blue and red line correspond to the average deviation required and the vertical bars represent the standard deviation. The best and worst cases constitute the envelope of the blue and red shaded areas. The initial separation distance can be interpreted as a time separation of the aircraft at the runway threshold. The ratio between distance due to the time separation and the diameter of the circle of avoidance is an important parameter for this study and will be denoted $\frac{D_0}{r_0}$

Although the figure suffers from sampling irregularities, the following general trend may be observed:

- At small arrival angles, that is in Miles-In-Trail like configuration, there is a very high sensitivity to $\frac{D_0}{r_0}$. When D_0 and r_0 are very close, the amplitude of the required maneuvers can vary from 10.5° up to 27°. There is a very high variability. When D_0 is larger than r_0, there is less variability and the required deviation is between 8° and 11° with a low variability.

- When the arrival cone angle increases, that is in Free-Flight like configuration, the impact of the ratio $\frac{D_0}{r_0}$ is less noticeable.

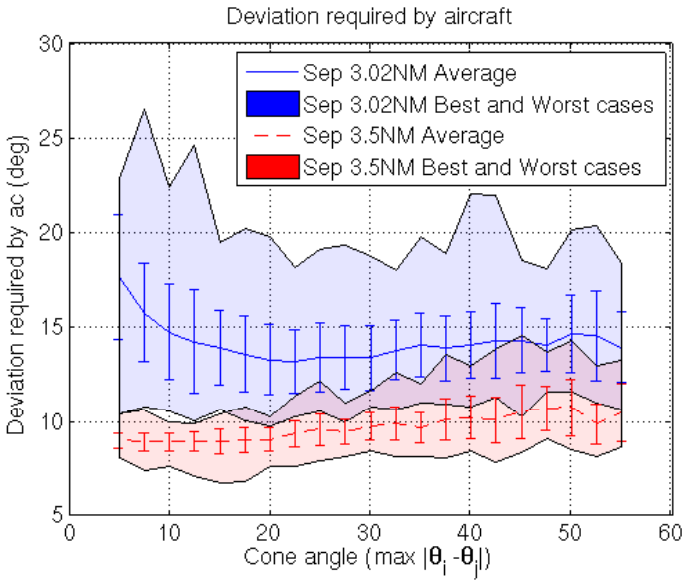


Figure 30: Evolution of the deviation required for initial separation distance of 3.05 NM and 3.5 NM

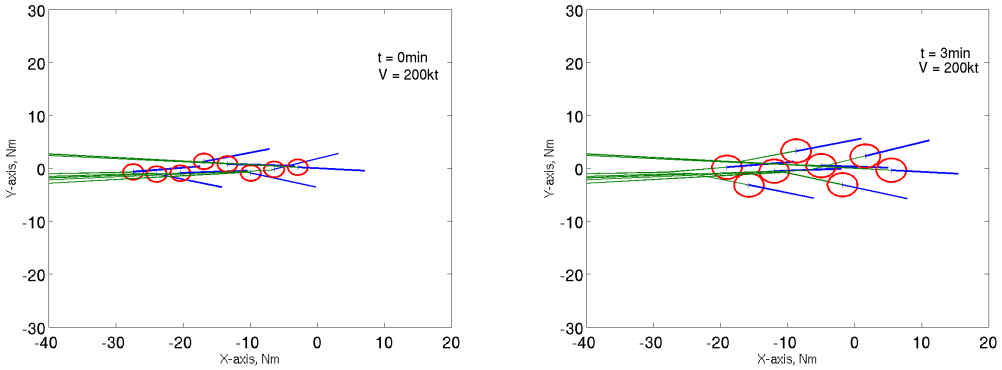
- For the case $r_0 = 3.05$ NM, the variability of the required deviation is quite constant and the average initially decreases as the angle of arrival increases, and then remains constant.

- For the case $r_0 = 3.5$ NM, the variability of the required deviation slightly increases as the arrival cone angle increases.

To conclude, Free Flight or Miles-In-Trail do not appear to be significantly different from the stand point of surveillance degradation. The important parameter in this study is the ratio $\frac{D_0}{r_0}$. Having a buffer between the circles of avoidance allows for smaller avoidance maneuvers. However, these conclusions were reached using computer-based conflict management, which may differ from human-based conflict management.

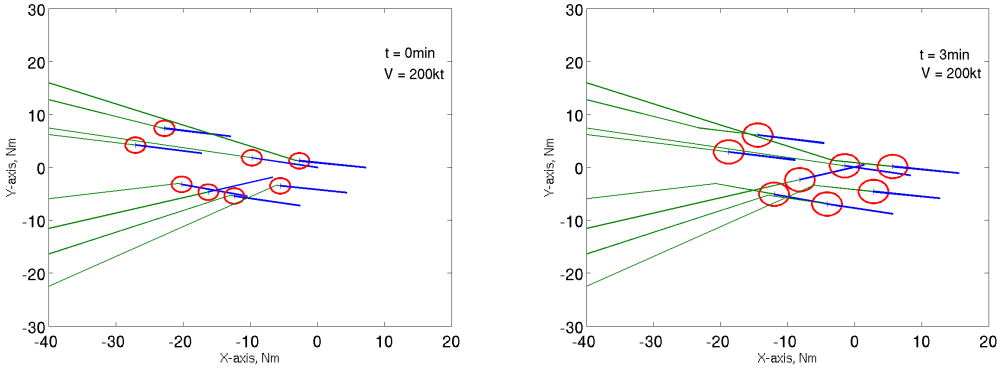
Figures 31 and 32 show the avoidance maneuvers for the worst cases of S_5 and S_{30} with 3.5NM initial separation. Figure 31(a) and 32(a) present the configuration at $t = 0$. Aircraft are at the position where the avoidance maneuver is calculated. The aircraft trajectories for $t < 0$ are represented and the line pointing out of the aircraft represent the new aircraft's

heading.



(a) $t = 0$ min (b) $t = 3$ min

Figure 31: Avoidance maneuvers for the worst case of S_5



(a) $t = 0$ min (b) $t = 3$ min

Figure 32: Avoidance maneuvers for the worst case of S_{30}

4.4 A near-optimal algorithm of avoidance under uncertainties

This section presents the algorithm used to solve the problem of conflict resolution under growing position uncertainties when several aircraft are present. A typical Mixed Integer Program (MIP) looks like:

$$\begin{aligned} \min_{x,z} \quad & f_1^T x + f_2^T z \\ \text{subject to} \quad & A_1 x + A_2 z \leq b \end{aligned}$$

where $f_1 \in \mathbb{R}^m, f_2 \in \mathbb{R}^n$, $x \in \mathbb{R}^m, z \in \{0,1\}^n$. $A_1 \in \mathbb{R}^{l \times m}, A_2 \in \mathbb{R}^{l \times n}$, $b \in \mathbb{R}^l$. m is the number of real variables, n is the number of binary variables and l is the number of

constraints. The next section focuses on developing a MIP model for the conflict resolution problem of interest in this paper.

Consider a set of n aircraft in a planar space. Each aircraft $i, i = 1, \cdots, n$ is defined by its position (x_i, y_i), its heading ψ_i and its speed V_i.

The relative velocity $\boldsymbol{V}_{ij}$ and speed V_{ij} of aircraft i with respect to aircraft j, and the distance D_{ij} between aircraft are given by:

$$\begin{aligned}
\boldsymbol{V}_{ij} &= [(V_{x_i} - V_{x_j}), (V_{y_i} - V_{y_j})]^T && (4)\\
&= [V_i \cos\psi_i - V_j \cos\psi_j, V_i \sin\psi_i - V_j \sin\psi_j]^T, && (5)\\
V_{ij} &= \sqrt{(V_i \cos\psi_i - V_j \cos\psi_j)^2 + (V_i \sin\psi_i - V_j \sin\psi_j)^2}, && (6)\\
D_{ij} &= \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}. && (7)
\end{aligned}$$

See also Figure 33. Let define some useful parameters for the avoidance problem. Let ψ_{ij} be the angle between the relative velocity $\boldsymbol{V}_{ij}$ and the x-axis, ω_{ij} be the angle between the connector of the aircraft and the x-axis. Finally, let γ_{ij} be the angle between the connector of the aircraft and a line starting from aircraft i and tangent to a circle of radius $2r$ and centered at aircraft j. The angles ψ_{ij}, ω_{ij}, and γ_{ij} are given by

$$\begin{aligned}
\psi_{ij} &= \arctan\frac{V_{y_{ij}}}{V_{x_{ij}}}\\
&= \arctan\frac{V_i \sin\psi_i - V_j \sin\psi_j}{V_i \cos\psi_i - V_j \cos\psi_j}, && (8)\\
\omega_{ij} &= \arctan\frac{y_j - y_i}{x_j - x_i}, && (9)\\
\gamma_{ij} &= \arcsin\frac{2r}{D_{ij}}. && (10)
\end{aligned}$$

4.4.1 Problem structure

The conflict resolution problem arising in this chapter is solved using a single heading change. The originality of this problem lies with the fact that the allowable miss distance between the two aircraft is time-dependent. Namely, at time $t = 0$, the minimum miss distance is $2r_0$. For $0 \le t \le T$, the minimum miss distance grows from $2r_0$ to $2r_f$, and T is given by

$$T = \frac{r_f - r_0}{\dot{r}},$$

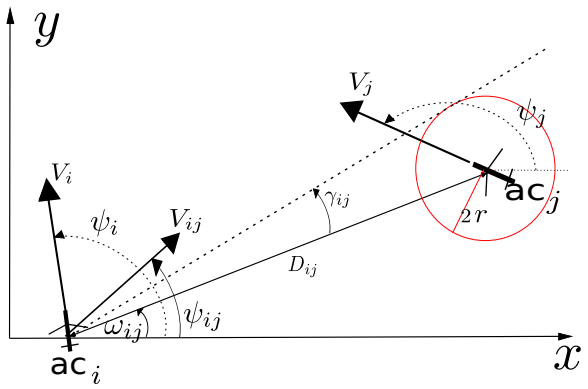


Figure 33: Problem configuration

where $\dot{r}$ is the rate of growth of the circle of avoidance. For $t \geq T$, the miss distance is constant and equal to $2r_f$. Figure 34 illustrates the conflict avoidance constraint in a relative frame of reference. For no conflict to occur, the circle $\mathcal{C}$ of radius $2r_0$ and centered on aircraft j must not intersect the area enclosed by the contour $\mathcal{C}_2$. This contour $\mathcal{C}_2$ can be seen to be the union of the half line $P1$, the circular segment $P3$ and the line segment $P2$ and their symmetric images across the line passing though aircraft i and parallel to the velocity $\boldsymbol{V}_{ij}$.

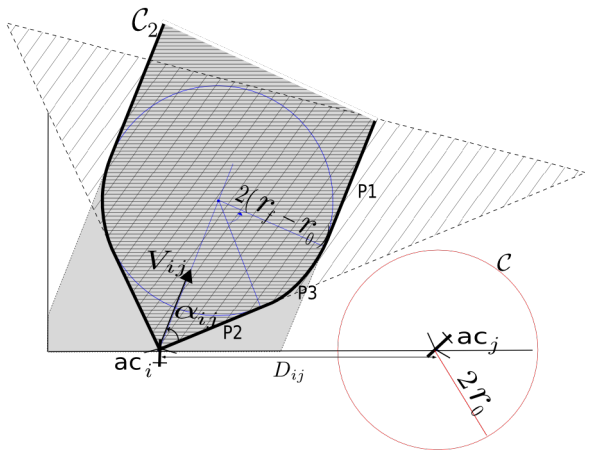


Figure 34: Avoidance constraints

For the purpose of linearization, the contour $\mathcal{C}_2$ is approximated by means of line segments, not to be intersected by the circle $\mathcal{C}$. The first, obvious linear approximation is to ask that the circle $\mathcal{C}$ not intersect either the gray stripe in figure 34, or the hatched cone

whose vertex is aircraft i. Those conditions can be geometrically expressed as follow:

- Asking that the circle $\mathcal{C}$ not intersect the gray stripe is a classical conflict avoidance problem that was developed and solved by Pallottino et al. in [103]. This problem deals with time $t \geq T$, when enough time has elapsed for the two radii of avoidance to be equal to r_f. The avoidance problem presented in figure 34 consists of finding a change of heading for aircraft i and j such that the line parallel to $\boldsymbol{V}_{ij}$ and tangent to the circle of radius $2(r_f - r_0)$ centered on the relative position of aircraft i to aircraft j at time T, does not intersect the circle of radius $2r_0$ centered on aircraft j. This problem is equivalent to the line directed along $\boldsymbol{V}_{ij}$ and passing through aircraft i not intersect a circle of radius $2r_f$ and centered on aircraft j. The avoidance constraints are then:

$$\begin{cases} \psi_{ij} - \omega_{ij} > \tilde{\gamma}_{ij} \\ \text{or} \\ \psi_{ij} - \omega_{ij} < -\tilde{\gamma}_{ij}, \end{cases} \tag{11}$$

  where

$$\tilde{\gamma}_{ij} = \arcsin \frac{2r_f}{D_{ij}}.$$

- Asking that the circle $\mathcal{C}$ not intersect the hatched cone can be detailed as follows, referring back to figure 34. The angular width $2\alpha_{ij}$ of the cone depends on the relative velocity of the aircraft:

$$\sin \alpha_{ij} = \frac{\dot{r}_i + \dot{r}_j}{V_{ij}}.$$

  If $\dot{r}_i + \dot{r}_j > V_{ij}$, the sum of the radii of the avoidance circles increases faster than the aircraft go away from each other. Whatever the relative velocity, the circles are bound to intersect each other. Hence, the cone of avoidance is the entire plan: as $\arcsin \alpha_{ij}$ is not defined, then $\alpha_{ij} = \pi$. If $\dot{r}_i + \dot{r}_j = V_{ij}$, the rate of increase of the circle of avoidance is the same as the relative speed. Hence, the avoidance cone is a half plane perpendicular to the relative velocity and $\alpha_{ij} = \frac{\pi}{2}$. The distance between the circles of avoidance will remain the same. To minimize the number of special cases, aircraft

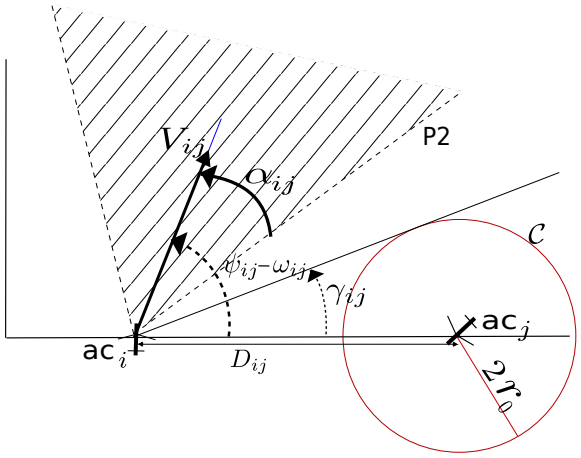


Figure 35: Configuration in the relative frame of reference: cone avoidance

are first ordered so that $-\frac{\pi}{2} \leq \omega_{ij} \leq \frac{\pi}{2}$. Then, the condition of avoidance between two aircraft is given by a condition on angles,

$$\text{for} \quad -\pi \geq \psi_{ij} - \alpha_{ij} \text{ and } \psi_{ij} + \alpha_{ij} \leq \pi: \quad \begin{cases} \psi_{ij} - \omega_{ij} - \alpha_{ij} > \hat{\gamma}_{ij} \\ \text{or} \\ \psi_{ij} - \omega_{ij} + \alpha_{ij} < -\hat{\gamma}_{ij}, \end{cases} \tag{12}$$

$$\text{for} \quad \psi_{ij} - \alpha_{ij} < -\pi: \quad \begin{cases} \psi_{ij} - \omega_{ij} - \alpha_{ij} + 2\pi > \hat{\gamma}_{ij} \\ \text{and} \\ \psi_{ij} - \omega_{ij} + \alpha_{ij} < -\hat{\gamma}_{ij}, \end{cases} \tag{13}$$

$$\text{and for} \quad \pi < \psi_{ij} + \alpha_{ij}: \quad \begin{cases} \psi_{ij} - \omega_{ij} - \alpha_{ij} > \hat{\gamma}_{ij} \\ \text{and} \\ \psi_{ij} - \omega_{ij} + \alpha_{ij} - 2\pi < -\hat{\gamma}_{ij}, \end{cases} \tag{14}$$

where

$$\hat{\gamma}_{ij} = \arcsin \frac{2r_0}{D_{ij}}.$$

Figure 35 presents those avoidance constraints. To avoid any singularity due to angles around $\pm\pi$, it is ensured that $-\frac{\pi}{2} \leq \omega_{ij} \leq \frac{\pi}{2}$. To do so, aircraft are ordered in function of their position (x_i, y_i) so that $x_1 \leq x_2 \leq \ldots \leq x_n$ and if $x_i = x_{i+1}$, $y_i < y_{i+1}$.

The description of the avoidance constraints could be left at that point. However, the developed constraints are somewhat conservative. For example, as shown in figure 36, the

circle $\mathcal{C}$ may intersect the cone and the gray stripe without intersecting $\mathcal{C}_2$. To improve the solution, a new constraint is introduced. Ideally, it would be the tangent (ℓ_1) to $\mathcal{C}_2$ at point A and ask that it does not intersect the circle $\mathcal{C}$. In the relative frame of reference, ℓ_1 has a slope $\psi_{ij} - \frac{\alpha_{ij}}{2}$. Requiring ℓ_1 not intersect $\mathcal{C}_2$ is equivalent to requiring ℓ_3 not intersect the circle centered on aircraft j and of radius $(2r_0 + |AD|)$. To formulate the constraint, the distance $|AD|$ is used. This distance is $2(r_f - r_0) - V_{ij}T \sin \frac{\alpha_{ij}}{2}$. This distance is a non linear function of ψ_i and ψ_j as V_{ij} and α_{ij} are non linear with respect to ψ_i and ψ_j. This function depends on too many parameters to be linearized easily. Nevertheless, this constraint can be approximated by using ℓ_2: the line of slope $\psi_{ij} - \frac{\alpha_{ij}}{2}$ passing through a point between A and B. For that, a majorant to $|DA|$ needs to be found. The following geometrical development gives this majorant: D is the middle of EB and $A \in [DB]$.

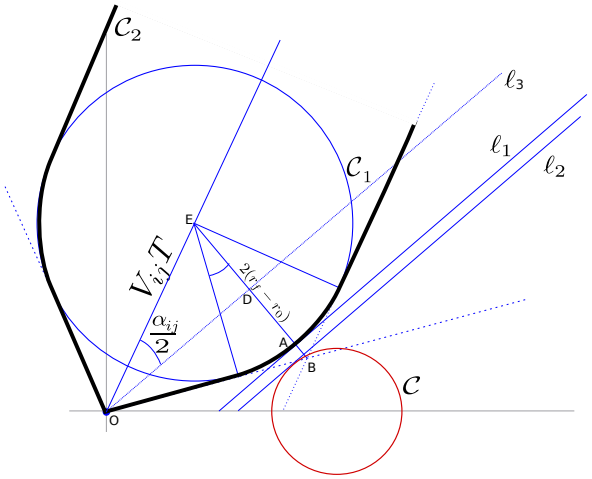


Figure 36: Geometry of the curved-part constraint (P3)

$$ED = \frac{EB}{2} \tag{15}$$
$$> \frac{EA}{2} \tag{16}$$
$$= r_f - r_0. \tag{17}$$

Hence, it brings

$$DA = EA - ED \tag{18}$$
$$< r_f - r_0. \tag{19}$$

This leads to the constraint that the line of slope $\frac{\alpha_{ij}}{2}$ not intersect the circle of radius $2r_0 + \frac{r_f - r_0}{2} = r_0 + r_f$: it is captured by the constraints

$$\text{for} \qquad -\pi \geq \psi_{ij} - \tfrac{\alpha_{ij}}{2} \text{ and } \psi_{ij} + \tfrac{\alpha_{ij}}{2} \leq \pi\text{:}$$
$$\begin{cases} \psi_{ij} - \omega_{ij} - \frac{\alpha_{ij}}{2} > \gamma^*_{ij} \\ \text{or} \\ \psi_{ij} - \omega_{ij} + \frac{\alpha_{ij}}{2} < -\gamma^*_{ij}, \end{cases} \tag{20}$$

$$\text{for} \qquad \psi_{ij} - \tfrac{\alpha_{ij}}{2} < -\pi\text{:}$$
$$\begin{cases} \psi_{ij} - \omega_{ij} - \frac{\alpha_{ij}}{2} + 2\pi > \gamma^*_{ij} \\ \text{and} \\ \psi_{ij} - \omega_{ij} + \frac{\alpha_{ij}}{2} < -\gamma^*_{ij}, \end{cases} \tag{21}$$

$$\text{and for} \qquad \pi < \psi_{ij} + \tfrac{\alpha_{ij}}{2}\text{:}$$
$$\begin{cases} \psi_{ij} - \omega_{ij} - \frac{\alpha_{ij}}{2} > \gamma^*_{ij} \\ \text{and} \\ \psi_{ij} - \omega_{ij} + \frac{\alpha_{ij}}{2} - 2\pi < -\gamma^*_{ij}, \end{cases} \tag{22}$$

where γ^*_{ij} is given by

$$\gamma^*_{ij} = \arcsin \frac{r_0 + r_f}{D_{ij}}.$$

4.4.2 Optimization of the conflict resolution

The previous section presented constraints that should be satisfied by the aircraft headings to avoid a conflict. These constraints may be incorporated in an optimal conflict resolution scheme. Denoting ψ_{i0} the initial heading of the aircraft i and ψ_i its heading after resolution, the problem is to compute:

$$\min J(\psi_1, \psi_2, \ldots \psi_n) = \min \sum_{i=1\ldots n} |\psi_i - \psi_{i0}|, \tag{23}$$

subject to

(11)

$$\text{or 12 if} \quad \begin{cases} -\pi \geq \psi_{ij} - \alpha_{ij} \\ \text{and} \\ \psi_{ij} + \alpha_{ij} \leq \pi \end{cases}$$

$$\text{or (13) if} \quad \psi_{ij} - \alpha_{ij} < -\pi$$

$$\text{or (14) if} \quad \pi < \psi_{ij} + \alpha_{ij}$$

$$\text{or (20) if} \quad \begin{cases} -\pi \geq \psi_{ij} - \frac{\alpha_{ij}}{2} \\ \text{and} \\ \psi_{ij} + \frac{\alpha_{ij}}{2} \leq \pi \end{cases}$$

$$\text{or (21) if} \quad \psi_{ij} - \frac{\alpha_{ij}}{2} < -\pi$$

$$\text{or (22) if} \quad \pi < \psi_{ij} + \frac{\alpha_{ij}}{2}$$

$$i = 1 \ldots n-1, \quad j = i+1 \ldots n,$$

where n is the number of aircraft.

4.4.3 Formulation when speeds are identical

For the sake of computational simplicity, it is assumed all aircraft share the same speed V. This assumption simplifies the formulae for ψ_{ij} and α_{ij} and allows to obtain piecewise linear formulation for psi_{ij} and a linearization for α_{ij}.

4.4.3.1 Expression of the cone angular width $2\alpha_{ij}$ in the relative frame of reference

This section presents the formulas to determine the cone angular width $2\alpha_{ij}$ when aircraft share an identical speed. Rewriting $\dot{r}_i = \dot{r}, i = 1 \ldots n$ leads to

$$\begin{aligned} \alpha_{ij} &= \arcsin \frac{2\dot{r}}{\sqrt{V^2(\cos\psi_i - \cos\psi_j)^2 + V^2(\sin\psi_i - \sin\psi_j)^2}} \\ &= \arcsin \frac{\dot{r}}{V|\sin(\frac{\psi_i - \psi_j}{2})|} \\ &= \arcsin \frac{\tan\Delta\psi}{|\sin(\frac{\psi_i - \psi_j}{2})|}. \end{aligned} \tag{24}$$

The half cone angular width α_{ij} is given by equation (24) and it is a non linear function of $\psi_i - \psi_j$. This function is symmetric about $\psi_i - \psi_j = 0$ and consists of two quasi convex components, separated at $\psi_i - \psi_j = 0$, as shown in Figure 37. The epigraph of each of these quasi convex components can be approximated by the intersection of linear constraints defined by their slopes a_k and intercepts b_k. Using the big-M method allows us to account for the presence of two disconnected components, and this linearization leads to:

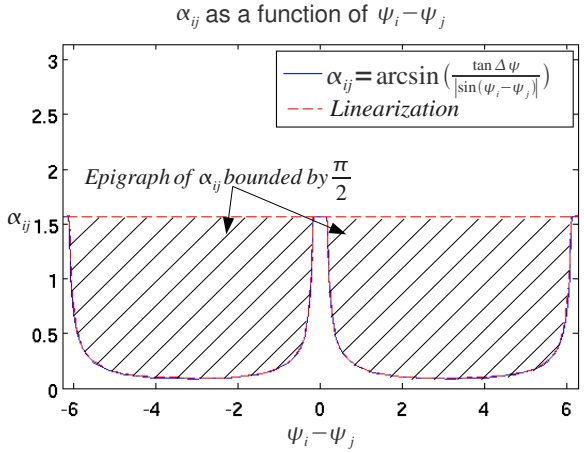


Figure 37: α_{ij} function of $\psi_i - \psi_j$ and the linearization used

$$\begin{cases} a_1(\psi_i - \psi_j) + b_1 \leq \alpha_{ij} \\ a_2(\psi_i - \psi_j) + b_2 \leq \alpha_{ij} \\ \quad \vdots \\ a_l(\psi_i - \psi_j) + b_l \leq \alpha_{ij} \end{cases} \tag{25}$$

or (26)

$$\begin{cases} -a_1(\psi_i - \psi_j) + b_1 \leq \alpha_{ij} \\ -a_2(\psi_i - \psi_j) + b_2 \leq \alpha_{ij} \\ \quad \vdots \\ -a_l(\psi_i - \psi_j) + b_l \leq \alpha_{ij}. \end{cases} \tag{27}$$

4.4.4 Mixed Integer Linear Programming formulation

The global algorithm consists on minimizing expression (23) subject to all the constraints previously developed. The constraints were written in AMPL format and solved using

CPLEX. The *or* conditions can be transformed to *and* conditions, by using the big-M formulation [43]. An example of such formulation can be found in [103]. Another option is to use $if - then - else$ conditions that are now handled by AMPL. MATLAB was used to generate the 1,650 cases and was also used for the post-processing. The computing time to solve an 8 aircraft configuration ranged from less than a second to a minute on a 4-processor, Pentium class computer.

4.4.4.1 Expression of the angle of the relative velocity: ψ_{ij}

Using the assumption of identical speed, the angle between the relative velocity and the x-axis, the expression of ψ_{ij} given by equation 8 can be simplified :

$$\begin{aligned} \psi_{ij} &= \arctan \frac{\sin\psi_i - \sin\psi_j}{\cos\psi_i - \cos\psi_j} \\ &= \arctan \frac{\sin(\frac{\psi_i-\psi_j}{2})\cos(\frac{\psi_i+\psi_j}{2})}{-\sin(\frac{\psi_i-\psi_j}{2})\sin(\frac{\psi_i+\psi_j}{2})}. \end{aligned} \tag{28}$$

ψ_{ij} is a function of $\psi_i + \psi_j$ and $\psi_i - \psi_j$. It can be shown that ψ_{ij} is a piecewise affine function of $\psi_i + \psi_j$ and $\psi_i - \psi_j$ of the form $\psi_{ij} = m_{ij}(\psi_i + \psi_j) + p_{ij}$. The value of m_{ij} if always $\frac{1}{2}$ and the values taken by p_{ij} are summarized in table 3.

Table 3: Intercept coefficient of ψ_{ij}

Case 1	$\psi_i - \psi_j < 0$ and $-2\pi \leq \psi_i + \psi_j < -\pi$	$p_{ij} = \frac{3\pi}{2}$
Case 2	$\psi_i - \psi_j < 0$ and $-\pi \leq \psi_i + \psi_j < 2\pi$	$p_{ij} = -\frac{\pi}{2}$
Case 3	$\psi_i - \psi_j \geq 0$ and $-2\pi \leq \psi_i + \psi_j < \pi$	$p_{ij} = \frac{\pi}{2}$
Case 4	$\psi_i - \psi_j \geq 0$ and $\pi \leq \psi_i + \psi_j < 2\pi$	$p_{ij} = -\frac{3\pi}{2}$

The value of p_{ij} can be computed using boolean variables. Let $bCaseDiffPos_{ij}$, $bCaseSumInfmPi_{ij}$, $bCaseSumSupPi_{ij}$, $bCase1_{ij}$ and $bCase4_{ij}$ be boolean variables such that:

$$bCaseDiffPos_{ij} = 1 \iff \psi_i - \psi_j \geq 0 \tag{29}$$

$$bCaseSumInfmPi_{ij} = 1 \iff -2\pi \leq \psi_i + \psi_j < -\pi \tag{30}$$

$$bCaseSumSupPi_{ij} = 1 \iff \pi \leq \psi_i + \psi_j < 2\pi \tag{31}$$

$$bCase1_{ij} = 1 \iff \begin{cases} bCaseDiffPos_{ij} = 0 \\ \text{and} \\ bCaseSumInfmPi_{ij} = 1 \end{cases} \tag{32}$$

$$bCase4_{ij} = 1 \iff \begin{cases} bCaseDiffPos_{ij} = 1 \\ \text{and} \\ bCaseSumSupPi_{ij} = 1 \end{cases} \tag{33}$$

$$\tag{34}$$

Those boolean variables yield the following expression for p_{ij} and hence ψ_{ij}:

$$p_{ij}{=}(bCaseSumPos_{ij} - \tfrac{1}{2})\pi + 2\pi(bCase1_{ij} - bCase4_{ij}), \tag{35}$$

$$\psi_{ij} = \frac{\psi_i + \psi_j}{2} + (bCaseDiffPos_{ij} - \tfrac{1}{2})\pi + 2\pi(bCase1_{ij} - bCase4_{ij}), \tag{36}$$

which is linear in ψ_i and ψ_j and in the boolean variables.

4.4.4.2 Constraints formulation for ψ_{ij}

The following formulation uses the big-M method. M is a large number such that if multiplied by a boolean set to 1, the constraint is always satisfied, whatever the value of the other variables. This enables us to select between constraints and use *or* relationship between constraints. This is a standard procedure described in [43].

Determination of the sign of $\psi_i - \psi_j$:

$$bCaseDiffPos_{ij} = 1 \iff \begin{cases} \psi_i - \psi_j - MbCaseDiffPos_{ij} < 0, \\ -\psi_i + \psi_j - M(1 - bCaseDiffPos_{ij}) < 0. \end{cases} \tag{37}$$

Determination of the boolean $bCaseSumInfmPi_{ij}$ if $\psi_i + \psi_j < -\pi$:

$$bCaseSumInfmPi_{ij} = 1 \iff \begin{cases} \psi_i + \psi_j + \pi - M(1 - bCaseSumInfmPi_{ij}) < 0, \\ -\psi_i - \psi_j - \pi - MbCaseSumInfmPi_{ij} < 0. \end{cases} \tag{38}$$

Determination if $\psi_i + \psi_j \geq \pi$:

$$bCaseSumInfmPi_{ij} = 1 \Longleftrightarrow \begin{cases} -\psi_i - \psi_j + \pi - M(1 - bCaseSumSupPi_{ij}) \leq 0, \\ \psi_i + \psi_j - \pi - MbCaseSumSupPi_{ij} \leq 0. \end{cases} \tag{39}$$

Determination of the boolean $bCase1_{ij}$:

$$bCase1_{ij} = 1 \Longleftrightarrow \begin{cases} -1.5 + bCaseDiffPos_{ij} - bCaseSumInfmPi_{ij} + 2bCase1_{ij} \leq 0, \\ -0.5 - 2bCaseDiffPos_{ij} + bCaseSumInfmPi_{ij} - bCase1_{ij} \leq 0. \end{cases} \tag{40}$$

Determination of the boolean $bCase4_{ij}$:

$$bCase4_{ij} = 1 \Longleftrightarrow \begin{cases} 1.5 - bCaseDiffPos_{ij} - bCaseSumSupPi_{ij} - 2(1 - bCase4_{ij}) \leq 0, \\ -1.5 + bCaseDiffPos_{ij} + bCaseSumSupPi_{ij} - bCase4_{ij} \leq 0. \end{cases} \tag{41}$$

Determination of the boolean $bA1_{ij}$:

$$bCase4_{ij} = 1 \Longleftrightarrow \begin{cases} \frac{\psi_i + \psi_j}{2} + \pi(\frac{1}{2} + bCaseDiffPos_{ij}) - 2\pi \ldots \\ (bCase1_{ij} + bCase4_{ij} + \alpha_{ij}) - M(1 - bA1_{ij}) \leq 0, \\ -\frac{\psi_i + \psi_j}{2} - \pi(\frac{1}{2} + bCaseDiffPos_{ij}) + \ldots \\ 2\pi(bCase1_{ij} + bCase4_{ij} + \alpha_{ij}) - MbA1_{ij} \leq 0 \end{cases} \tag{42}$$

Determination of the boolean $bA2_{ij}$:

$$bCase4_{ij} = 1 \Longleftrightarrow \begin{cases} -\frac{\psi_i + \psi_j}{2} - \pi(bCaseDiffPos_{ij} - \frac{3}{2}) + 2\pi \ldots \\ (bCase1_{ij} + bCase4_{ij} + \alpha_{ij}) - M(1 - bA2_{ij}) \leq 0, \\ \frac{\psi_i + \psi_j}{2} + \pi(bCaseDiffPos_{ij} - \frac{3}{2}) - \ldots \\ 2\pi(bCase1_{ij} + bCase4_{ij} + \alpha_{ij}) - MbA2_{ij} \leq 0 \end{cases} \tag{43}$$

Determination of the boolean $bB1_{ij}$:

$$bCase4_{ij} = 1 \Longleftrightarrow \begin{cases} \frac{\psi_i + \psi_j}{2} + \pi(\frac{1}{2} + bCaseDiffPos_{ij}) - 2\pi \ldots \\ (bCase1_{ij} + bCase4_{ij} + \frac{\alpha_{ij}}{2}) - M(1 - bB1_{ij}) \leq 0, \\ -\frac{\psi_i + \psi_j}{2} - \pi(\frac{1}{2} + bCaseDiffPos_{ij}) + \ldots \\ 2\pi(bCase1_{ij} + bCase4_{ij} + \frac{\alpha_{ij}}{2}) - MbB1_{ij} \leq 0 \end{cases} \tag{44}$$

Determination of the boolean $bB2_{ij}$:

$$bCase4_{ij} = 1 \Longleftrightarrow \begin{cases} -\frac{\psi_i+\psi_j}{2} - \pi(bCaseDiffPos_{ij} - \frac{3}{2}) + 2\pi \ldots \\ (bCase1_{ij} + bCase4_{ij} + \frac{\alpha_{ij}}{2}) - M(1 - bB2_{ij}) \leq 0, \\ \frac{\psi_i+\psi_j}{2} + \pi(bCaseDiffPos_{ij} - \frac{3}{2}) - \ldots \\ 2\pi(bCase1_{ij} + bCase4_{ij} + \frac{\alpha_{ij}}{2}) - MbB2_{ij} \leq 0 \end{cases} \tag{45}$$

4.4.4.3 Cone angular width $2\alpha_{ij}$ in the relative frame of reference

Using the big-M method, the expression of α_{ij} in equation 4.4.3.1 can be rewritten as:

$$\begin{aligned} a_1(\psi_i - \psi_j) + b_1 - MbCaseAlphaPos_{ij} &\leq \alpha_{ij} \\ a_2(\psi_i - \psi_j) + b_2 - MbCaseAlphaPos_{ij} &\leq \alpha_{ij} \\ &\vdots \\ a_l(\psi_i - \psi_j) + b_l - MbCaseAlphaPos_{ij} &\leq \alpha_{ij} \\ -a_1(\psi_i - \psi_j) + b_1 - M(1 - bCaseAlphaPos_{ij}) &\leq \alpha_{ij} \\ -a_2(\psi_i - \psi_j) + b_2 - M(1 - bCaseAlphaPos_{ij}) &\leq \alpha_{ij} \\ &\vdots \\ -a_l(\psi_i - \psi_j) + b_l - M(1 - bCaseAlphaPos_{ij}) &\leq \alpha_{ij} \end{aligned} \tag{46}$$

with $bCaseAlphaPos_{ij}$ a binary variable used to transform the *or* between the constraints in an *and* relationship.

The avoidance constraints are now given by:

$$\begin{cases} -\frac{\psi_i+\psi_j}{2} - (\frac{1}{2} - bCasePlus_{ij})\pi + \omega_{ij} + \alpha_{ij} \ldots \\ \quad +\hat{\gamma}_{ij} - MbCaseIneqPos_{ij} < 0 \\ \text{and} \\ \frac{\psi_i+\psi_j}{2} + (\frac{1}{2} - bCasePlus_{ij})\pi - \omega_{ij} + \alpha_{ij} \ldots \\ \quad +\hat{\gamma}_{ij} - M(1 - bCaseIneqPos_{ij}) < 0 \end{cases} \tag{47}$$

with $bCaseIneqPos_{ij}$ a binary variable used to transform the *or* between the constraints in an *and* relationship.

4.4.4.4 Mixed constraints for conflict avoidance

The conflict avoidance constraints given by equation 24 can be handled in a linear programming framework by using the big-M method, leading to the following mixed constraints:

$$
\begin{cases}
-\frac{\psi_i+\psi_j}{2} + (\frac{1}{2} - bCasePlus_{ij})\pi + \omega_{ij} + \tilde{\gamma}_{ij} \dots \\
+2\pi(bCase4_{ij} - bCase1_{ij}) - M(bCaseIneqPos_{ij} + bK1_{ij}) & < 0, \\
\frac{\psi_i+\psi_j}{2} - (\frac{1}{2} - bCasePlus_{ij})\pi - \omega_{ij} + \tilde{\gamma}_{ij} \dots \\
-2\pi(bCase4_{ij} - bCase1_{ij}) - M(1 - bCaseIneqPos_{ij} + bK1_{ij}) & < 0, \\
\\
-\frac{\psi_i+\psi_j}{2} + (\frac{1}{2} - bCasePlus_{ij})\pi + \omega_{ij} + \hat{\gamma}_{ij} \dots \\
+\alpha_{ij} + 2\pi(bCase4_{ij} - bCase1_{ij}) - M(bCaseIneqPos_{ij} + 1 - bA0_{ij} + bK2_{ij}) & < 0, \\
\frac{\psi_i+\psi_j}{2} - (\frac{1}{2} - bCasePlus_{ij})\pi - \omega_{ij} + \hat{\gamma}_{ij} \dots \\
+\alpha_{ij} - 2\pi(bCase4_{ij} - bCase1_{ij}) - M(2 - bCaseIneqPos_{ij} - bA0_{ij} + bK2_{ij}) & < 0, \\
\\
-\frac{\psi_i+\psi_j}{2} + (\frac{1}{2} - bCasePlus_{ij})\pi + \omega_{ij} + \hat{\gamma}_{ij} \dots \\
+\alpha_{ij} + 2\pi(bCase4_{ij} - bCase1_{ij} - bA1_{ij}) - M(bA0_{ij} + bK2_{ij}) & < 0, \\
\frac{\psi_i+\psi_j}{2} - (\frac{1}{2} - bCasePlus_{ij})\pi - \omega_{ij} + \hat{\gamma}_{ij} \dots \\
+\alpha_{ij} - 2\pi(bCase4_{ij} - bCase1_{ij} + bA2_{ij}) - M(bA0_{ij} + bK2_{ij}) & < 0, \\
\\
-\frac{\psi_i+\psi_j}{2} + (\frac{1}{2} - bCasePlus_{ij})\pi + \omega_{ij} + \gamma^*_{ij} \dots \\
+\frac{\alpha_{ij}}{2} + 2\pi(bCase4_{ij} - bCase1_{ij}) - M(bCaseIneqPos_{ij} + 1 - bB0_{ij} + bK3_{ij}) & < 0, \\
\frac{\psi_i+\psi_j}{2} - (\frac{1}{2} - bCasePlus_{ij})\pi - \omega_{ij} + \gamma^*_{ij} \dots \\
+\frac{\alpha_{ij}}{2} - 2\pi(bCase4_{ij} - bCase1_{ij}) - M(2 - bCaseIneqPos_{ij} - bB0_{ij} + bK3_{ij}) & < 0, \\
\\
-\frac{\psi_i+\psi_j}{2} + (\frac{1}{2} - bCasePlus_{ij})\pi + \omega_{ij} + \gamma^*_{ij} \dots \\
+\frac{\alpha_{ij}}{2} + 2\pi(bCase4_{ij} - bCase1_{ij} - bA1_{ij}) - M(bB0_{ij} + bK3_{ij}) & < 0, \\
\frac{\psi_i+\psi_j}{2} - (\frac{1}{2} - bCasePlus_{ij})\pi - \omega_{ij} + \gamma^*_{ij} \dots \\
+\frac{\alpha_{ij}}{2} - 2\pi(bCase4_{ij} - bCase1_{ij} + bA2_{ij}) - M(bB0_{ij} + bK3_{ij}) & < 0, \\
\\
bK1_{ij} + bK2_{ij} + bK3_{ij} = 2, \\
bA0_{ij} + bA1_{ij} + bA2_{ij} = 1, \\
bB0_{ij} + bB1_{ij} + bB2_{ij} = 1,
\end{cases}
\tag{48}
$$

with $bA0_{ij}, bB0_{ij}, bK1_{ij}, bK2_{ij}$ and $bK3_{ij}$ binary variables to choose among avoidance mode.

The global algorithm consists on minimizing expression 23 subject to all the constraints previously developed. The constraints were written in AMPL format and solved using CPLEX.

4.4.4.5 Minimization of the objective

The objective of this problem is to minimize the total deviation as stated in equation 23. Introducing a new variable $AbsVal_i$ for each aircraft, minimizing the absolute value of the deviation considering that $-\pi < \psi_i \leq \pi$, can be reformulated as follow

$$\min J(\psi_1, \psi_2, \dots \psi_n) = \min \sum_{i=1\dots n} AbsVal_i, \tag{49}$$

subject to:

$$\begin{cases} -\psi_i + \psi_{0i} + \pi - M(1 - bSupPi_i) < 0 \\ \psi_i - \psi_{0i} - \pi - MbSupPi_i \leq 0 \\ \psi_i - \psi_{0i} + \pi - M(1 - bInfmPi_i) < 0 \\ -\psi_i + \psi_{0i} - \pi - MbInfmPi_i \leq 0 \\ 2\pi bInfmPi_i + \psi_i - \psi_{0i} - MbSupPi_i \leq AbsVal_i \\ 2\pi bSupPi_i - \psi_i + \psi_{0i} - MbInfmPi_i \leq AbsVal_i \end{cases} \tag{50}$$

Where $bSupPi_i$ and $bInfmPi_i$ are boolean variables used for modulo purposes when the deviation $\psi_i - \psi_{0i}$ is greater or smaller than $\pm\pi$.

4.5 Algorithm of avoidance under uncertainties in 3 dimensions

This section presents a new formulation for the algorithm of avoidance in the presence of uncertainties, in order to extend it to 3D. The algorithm minimizes the number of maneuvers required to maintain the safety of the airspace under degraded conditions. Uncertainties are still modeled as an increase in the required separation distance between aircraft, as presented in section 4.2. A single maneuver for each aircraft is chosen to maintain safe separation. Maneuvers include heading changes, speed changes and flight level changes. Maneuvers are simple to execute and guarantee a conflict-free configuration after execution. Their feasibility is constrained by weather avoidance, sector boundaries and aircraft performance. A Mixed Integer Program is used to determine the set of maneuvers to be executed.

4.5.1 Algorithm overview

Consider a system of n aircraft in a 3-dimensional Cartesian airspace. The flat earth approximation is valid for the distances considered in this section. The set of all aircraft in the airspace is denoted $\mathcal{S}$. The horizontal position of aircraft i is (x_i, y_i) and its vertical position is given by its flight level FL_i. Its speed, heading and vertical speed are V_i, ψ_i, and V_{zi} respectively. The algorithm is intended for en route traffic.

An overview of the algorithm is first presented, before giving more details:

1. Organize aircraft by flight level.
2. Determine, for each aircraft i, the set of feasible maneuvers $\mathcal{M}_i^{feas}$ (performance, weather, sector boundary, flight level change constraints).
3. Determine, for each pair of aircraft (i, j), the subset $\mathcal{K}_{ij} \subseteq \mathcal{M}_i^{feas} \times \mathcal{M}_j^{feas}$ of combinations of feasible maneuvers K_{ij}^{kl} that do not lead to a conflict.
4. Cluster aircraft to reduce computing time.
5. Find the set of maneuvers (one per aircraft) that minimizes the overall cost and ensures the safety of the airspace.

4.5.1.1 Organization of aircraft by flight level

During the en-route phase of the flight, aircraft maintain constant pressure altitude, corresponding to flight levels. Pilots need to obtain a clearance to increase altitude when the

weight of the aircraft permits. Then, they can climb to the next allowed flight level. During en-route phases, flight level changes do not occur very often. It is then reasonable to split the overall airspace into layers corresponding to flight levels. Aircraft can transfer from one layer to another but the movement of each aircraft between flight level change phases is considered two-dimensional. When an aircraft is climbing or descending, that is $V_{zi} \neq 0$, the aircraft is accounted for at all the flights levels between the current flight level and the destination flight level.

4.5.1.2 Determination of feasible maneuvers

To provide controllers with a simple set of avoidance maneuvers, the set $\mathcal{M}$ of potential maneuvers for all the aircraft includes the following: No change, speed changes ($\pm\Delta V$), heading changes ($\pm\Delta\psi$) and flight level changes ($\pm\Delta FL$). Only one maneuver per aircraft is allowed, i.e. an aircraft cannot change heading and change speed concurrently. Let m_i^k be a binary variable that indicates that aircraft i executes maneuver k, $k \in \mathcal{M}$. The following discrete dynamics are used for the aircraft:

$$V_i(t+1) = V_i(t) + m_i^1 \Delta V - m_i^2 \Delta V, \tag{51}$$

$$\psi_i(t+1) = \psi_i(t) + m_i^3 \Delta\psi - m_i^4 \Delta\psi, \tag{52}$$

$$FL_i(t+1) = FL_i(t) + m_i^5 \Delta FL - m_i^6 \Delta FL + 1.V_{zi}, \tag{53}$$

where $1.V_{zi}$ is the altitude change of aircraft i between t and $t+1$. Maneuvers are assumed to be immediately executed.

Theoretically, all the aircraft are able to execute all maneuvers in $\mathcal{M}$. In practice, all maneuvers might not be feasible for each aircraft due to weather, airspace boundary, flight level change constraints and aircraft performance restrictions. Let $\mathcal{M}_i^{feas}$ be the set of feasible maneuvers for aircraft i. $\mathcal{M}_i^{feas}$ is a subset of $\mathcal{M}$ and is initially populated with all the elements of $\mathcal{M}$. The following presents the constraints used to determine maneuvers' feasibility. If a maneuver is not feasible for aircraft i, it is removed from $\mathcal{M}_i^{feas}$.

Heading change The constraints accounted for to determine the feasibility of a heading change are weather avoidance and airspace boundary:

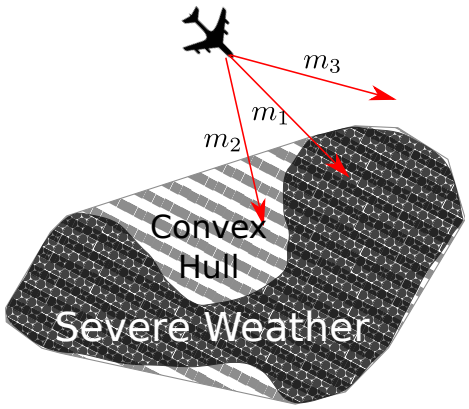


Figure 38: Weather avoidance constraints

- **Weather Avoidance** Let t_w be the time constant for weather avoidance. A maneuver is not feasible if, after execution, the new trajectory encounters a weather cell within a time interval $[0 \quad t_w]$. Severe weather cells are modeled by polyhedral cells in which aircraft are not allowed to fly. Let $\mathcal{W}$ be the set of all severe weather polyhedra $w_l, l = 1 \ldots n_w$. Weather position and size estimates can be obtained or simulated [117].

  To detect whether an aircraft encounters severe weather for $t \leq t_w$, the position of the aircraft is extrapolated for different times smaller than t_w using the above dynamics. If the extrapolated position lies within a weather polyhedron $w_l \in \mathcal{W}$, then the maneuver is not feasible and it is removed from $\mathcal{M}_i^{feas}$.

  Weather polyhedra are 3-dimensional and do not necessarily cover all flight levels. They are extrapolated to their convex hull. The example depicted on Figure 38 shows the reason for dealing with the convex hull instead of the polyhedron itself. The arrows represent the resulting velocity vectors after execution of maneuvers m_1, m_2 or m_3. The velocity vectors are scaled so that the arrow heads correspond to the extrapolated position at time t_w. Maneuver m_1 corresponds to the nominal trajectory and is not feasible since it leads the aircraft into the severe weather cell. If m_2 were executed, the aircraft would not encounter weather for $t < t_w$ but would then have no choice but to turn back. This is the reason for extrapolating the weather cell to its convex hull. Maneuver m_3 is therefore the only feasible maneuver.

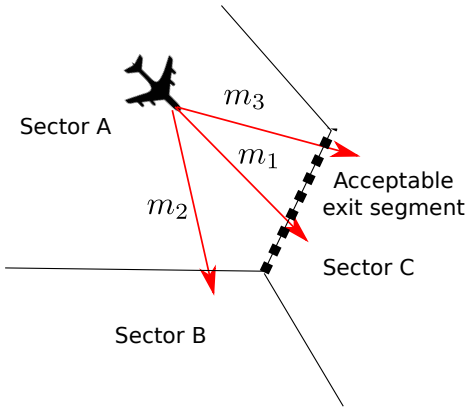


Figure 39: Sector boundary constraints

- **Airspace Boundary Constraints** When proposing a heading change, one needs to be aware of sector changes. When an aircraft changes sector, the surveillance of the aircraft is handed out to another controller. If there is a surveillance degradation, this change might be difficult and should be avoided. Traffic flow management is another reason to avoid sector changes when trying to solve conflicts. If the neighboring sector is at maximum capacity, a heading change leading an aircraft to this sector is not permitted. A heading change leading the aircraft to exit the sector out of the acceptable exit segment is not allowed. In the situation depicted by Figure 39, m_2 is not permissible since it leads the aircraft to exit the sector out of the allowed segment, in a time less than a given duration designated by t_s. Therefore, any maneuver m_i^k leading to a premature exit from the sector is removed from $\mathcal{M}_i^{feas}$.

Speed change Depending on the aircraft's cruise speed and altitude, accelerating or deccelerating could bring the aircraft out of its flight envelope [36]. Flying out of the flight envelope can have dramatic consequences. Therefore, any maneuver m_i^k taking the aircraft out of its flight envelope will be removed from $\mathcal{M}_i^{feas}$.

Flight level change Vertical avoidance maneuvers are very efficient: The first effect of a pitch change is a quick change in vertical speed and therefore altitude [104], but further throttle adjustments have to be done to maintain the climb rate, altitude and speed. In addition, the amplitude of required altitude changes for conflict avoidance are relatively

small (1000 ft) compared to required lateral changes for conflict avoidance (a few NM). Vertical changes are the maneuvers used by TCAS [139]. This algorithm aims at getting a sustainable situation after the resolution maneuver. This means it is not only interested in solving conflicts, but also in ensuring a conflict-free situation for a given period of time. Therefore, vertical avoidance maneuvers are extended to flight level changes. A flight level change maneuver m_i^k is removed from $\mathcal{M}_i^{feas}$ if:

- The aircraft is at its service ceiling, that is its weight does not allow it to climb higher [36].
- Changing flight level might immediately affect traffic on the crossed flight level and on the targeted flight level.
- The aircraft is descending or it is about to initiate its descent.

4.5.1.3 Degradation modeling and conflict detection

The previous session presented how the the set of all possible maneuvers $\mathcal{M}_i^{feas}$ for all $i \in \mathcal{S}$ is pruned of all the infeasible maneuvers. The model for degradation of performances and uncertainties presented in section 4.2 is first summarized before determining if a combination of maneuvers lead to a conflict. The initial radius of avoidance is denoted by r_0 and the final one by r_f. The rate of growth of the radius $\dot{r}$ is assumed constant, i.e. $t_g = \frac{r_f - r_i}{\dot{r}}$, where t_g is the time of growth of the radius of avoidance. It represents the time allowed to increase the separation distance between aircraft and is a variable that can be set in the model. It is assumed that the radius of avoidance has a constant value for $t > t_g$. In this model, the degradation does not affect aircraft vertical performance, that is, traffic does not need to be additionally vertically spread out.

Let t_c be the time horizon for conflict detection. Allowing only maneuvers that do not lead to a conflict in the next t_c minutes ensures that the airspace will remain conflict-free for the next t_c minutes. To determine if a maneuver leads to a conflict, the existence of a point of intersection of the circles of avoidance is sought. If it exists, the time to conflict is calculated. Let X_r and V_r define the relative position and velocity of aircraft j with respect

to aircraft i, that is

$$X_r = \begin{bmatrix} x_j - x_i \\ \\ y_j - y_i \end{bmatrix}, \quad V_r = \begin{bmatrix} v_{x_j} - v_{x_i} \\ \\ v_{y_j} - v_{y_i} \end{bmatrix}. \tag{54}$$

Let present first the ondition for conflict existence with final radius of avoidance r_f. The two circles of avoidance will intersect when the distance between the two aircraft is equal to $2r_f$:

$$\|X_r + V_r t\| = 2r_f. \tag{55}$$

The discriminant of this polynomial in t is

$$\Delta = (2X_r \cdot V_r)^2 - 4V_r^2(X_r^2 - 4r_f^2). \tag{56}$$

If $\Delta \leq 0$, the circles will never intersect or be tangent and there is no conflict. The corresponding maneuvers do not lead to a conflict and they are a feasible combination of maneuvers. If $\Delta > 0$, there are two solutions $t_{1,2}$ corresponding to the very first time the circles intersect and to the end of the intersection period. The solutions are

$$t_{1,2} = \frac{-2X_r \cdot V_r \pm \sqrt{(2X_r \cdot V_r)^2 - 4V_r^2(X_r^2 - 4r_f^2)}}{2V_r^2}. \tag{57}$$

This leads to the following potential cases:

- $t_1 \leq t_2 \leq 0$: The circles intersected in the past, no conflict.
- $t_1 \leq 0 \leq t_2$: The circles of radius r_f are already intersecting. The time of intersection for growing the circles needs to be computed. If $\|X_r\| > 2r_f$, a sufficient condition for no conflict is $X_r \cdot V_r < 0$.
- $0 \leq t_1 \leq t_g$: The circles of avoidance will intersect before the radii of avoidance reach r_f. The time of intersection of the growing circles needs to be computed.
- $0 < t_g \leq t_1 < t_c$: The circles will intersect with radius of avoidance r_f, in a time shorter than t_c. This combination of maneuvers is not permitted, since it leads to a conflict.

- $0 < t_c \leq t_1$: The circles will intersect in a later time than the time horizon fixed for conflict detection. This maneuver is feasible.

Note that this requires that $\|V_r\| \neq 0$. If $\|V_r\| = 0$, aircraft have the same heading and the same speed. A conflict can only occur during the period of growing radii if the distance between the aircraft is smaller than $2r_f$.

If $t_1 \geq t_g$, the conflict will happen after the radius of the circles reaches r_f. Otherwise, the same reasoning applies, except that the value of t_1 is obtained by solving the following equation:

$$\|X_r + V_r t\| = 2(r_i + \dot{r}t), \tag{58}$$

which has solutions (if any)

$$t_{1,2} = \frac{-(X_r \cdot V_r - 8r_i\dot{r})}{(V_r^2 - 4\dot{r}^2)} \pm \frac{\sqrt{(X_r \cdot V_r - 4r_i\dot{r})^2 - 2(V_r^2 - 4\dot{r}^2)(X^2 - 4r_i^2)}}{(V_r^2 - 4\dot{r}^2)}. \tag{59}$$

If no conflict is detected, let K_{ij}^{kl} be a binary variable indicating that aircraft i executes maneuver $m_i^k \in \mathcal{M}_i^{feas}$ and aircraft j executes maneuver $m_j^l \in \mathcal{M}_j^{feas}$. Define $\mathcal{K}_{ij} = \left\{m_i^k \in \mathcal{M}_i^{feas}, m_j^l \in \mathcal{M}_j^{feas}, i \text{ and } j \text{ not in conflict}\right\}$. If $K_{ij}^{kl} = 1, \forall k \in \mathcal{M}_i^{feas}, l \in \mathcal{M}_j^{feas}$, then aircraft i and j are not in conflict, no matter what combination of maneuvers is chosen. Aircraft i and j do not interact. Now that feasible maneuvers and conflict-free combinations of maneuvers have been identified, the following step is a clustering method based on possible interactions used to reduce the computing load.

4.5.1.4 Aircraft clustering

To optimize the computational performance, aircraft are split into clusters $\mathcal{C}$. Clustering enables parallel computation. Aircraft are clustered by potential interaction, i.e aircraft i and j belong to the same cluster if one of the combination of feasible maneuvers of $\mathcal{K}_{ij}$ is executed and leads to a conflict. Algorithm 1 is used to cluster aircraft based on their possible interaction.

```
Input: Set of n aircraft and no cluster
Output: Set of clusters
foreach aircraft i = 2 to n do
    foreach aircraft j = 1 to i − 1 do
        if aircraft i and j can be in conflict then
            if aircraft j belongs to a cluster C_k then
                if aircraft i belongs to a cluster C_l then
                    merge clusters C_k and C_l;
                else
                    add aircraft i to cluster C_k
                end
            else
                if aircraft i belongs to a cluster C_l then
                    add aircraft j to cluster C_l
                else
                    create new cluster with aircraft i and j
                end
            end
        end
    end
end
.
```

Algorithm 1: Clustering algorithm based on possible interaction

4.5.2 MIP formulation

The previous steps provide us with a set of aircraft clusters that can potentially interact, and a set of feasible maneuvers for each aircraft. The near-optimal set of maneuvers that spreads out the traffic and avoids conflicts is selected using a mixed integer program solved with CPLEX [64].

4.5.2.1 Maneuver uniqueness for each aircraft

Only one maneuver per aircraft is permitted, resulting in the constraint

$$\sum_{m_i^k \in \mathcal{M}_i^{feas}} m_i^k = 1, \tag{60}$$

for each aircraft $i \in \mathcal{S}$.

4.5.2.2 Avoidance constraint

For any combination of aircraft $(i, j) \in \mathcal{C}_p$, $K_{ij}^{kl} = 1$ implies that aircraft i executes maneuver $m_i^k \in \mathcal{M}_i^{feas}$ and aircraft j executes maneuver $m_j^l \in \mathcal{M}_j^{feas}$, that is

$$K_{ij}^{kl} \implies m_i^k + m_j^l = 2. \tag{61}$$

Combining this implication with a uniqueness constraint on the choice of possible maneuvers for each combination of aircraft

$$\sum_{K_{ij}^{kl} \in \mathcal{K}_{ij}} K_{ij}^{kl} = 1, \tag{62}$$

ensures that the requirement of only one conflict-free maneuver for each aircraft is met.

4.5.2.3 Maneuver cost

The cost of each maneuver is determined by the speed of its execution and the imminence of the conflict, since there is a risk of collision when the maneuver is not implemented on time. For instance, it takes less time to climb several hundred feet than accelerate 20 kts at constant altitude in cruise. The most efficient maneuver for conflict avoidance is a vertical change. A heading change is the second fastest maneuver to implement and a speed change is the slowest. Therefore, the cost $c_{ij}^{m_i^k}$ for aircraft i of maneuver m_i^k to resolve the conflict with aircraft j will depend on the imminence of a conflict if no maneuver is executed, and if so, on the time to conflict t_1, expressed in minutes.

$$\begin{aligned}
&\text{If no conflict :} && c^{\Delta\psi} < c^{\Delta FL} < c^{\Delta V},\\
&\text{if } t_1 < 2 : && c^{\Delta FL} < c^{\Delta\psi} < c^{\Delta V},\\
&\text{if } 2 \leq t_1 < 8 : && c^{\Delta\psi} < c^{\Delta FL} < c^{\Delta V},\\
&\text{if } 8 \leq t_1 : && c^{\Delta V} < c^{\Delta\psi} < c^{\Delta FL},
\end{aligned}$$

where $c^{\Delta FL}$ is the cost of a flight level change, $c^{\Delta\psi}$ the cost of a heading change and $c^{\Delta V}$ is the cost of a speed change. The final cost for aircraft i to execute maneuver l is given by

$$c_i^{m_i^k} = \sum_{j \in C_p, j \neq i} c_{ij}^{m_i^k}, \tag{63}$$

where C_p is the cluster of aircraft i.

4.5.2.4 Objective function

The objective is to minimize the cost J of the selected maneuvers. The algorithm thus minimizes the cost J_p, for each cluster, subject to the avoidance and maneuver uniqueness constraints

$$\min_{\mathcal{C}_p, \mathcal{M}_i^{feas}} J_p \quad = \quad \min \sum_{i \in \mathcal{C}_p} \sum_{m_i^k \in \mathcal{M}_i^{feas}} c_i^{m_i^k} m_i^k. \tag{64}$$

subject to the avoidance and maneuver uniqueness constraints.

4.5.2.5 Existence of a solution

Given an initial limited set of potential maneuvers $\mathcal{M}_i$ for each aircraft i, there might not exist a solution for all the aircraft. This can be the case if an aircraft that, in addition to being in conflict with another or several other aircraft, is highly constrained by weather, sector boundaries, and performance. To resolve this issue, the set of potential maneuvers for aircraft i and for the aircraft in conflict is extended, i.e a wider range of heading changes is added. Before choosing the best set of maneuvers, it is checked that there exist several feasible maneuvers for each aircraft. If not, new maneuvers are added to $\mathcal{M}_i$ and the algorithm is restarted. Even if there exist several feasible maneuvers for each aircraft, there might not exist a set of maneuvers that satisfy all the constraints, which is immediately detected by the MIP solver. The same solution applies for this case, i.e increasing the set of potential maneuvers. This process is iterated until a solution exists.

4.5.3 Simulation results

A 3-dimensional airspace with 7 flight levels (350 to 410) was simulated. The airspace is a square whose sied is 400 NM. Aircraft fly on distinct flight levels but can change levels to avoid conflicts. Aircraft fly westbound on even-numbered flight levels and eastbound on odd-numbered flight levels. The values of the simulation parameters are presented in Table 4.

Figure 40 presents a screenshot of the airspace during the simulation. The polyhedron in the middle represents a weather phenomenon that should be avoided. The weather does not

Table 4: Parameters used for simulation

r_0	2.5 NM.
r_f	5 NM.
$\dot{r}$	50 kts.
t_c	For same flight level: 15 min.
t_c	When changing flight level: 5 min for flight level crossed.
t_c	When changing flight level: 15 min for destination flight level.
t_g	3 min.
t_w	20 min.
t_s	10 min.
$\Delta\psi$	$\pm 20^\circ$ and $\pm 70^\circ$.
ΔV	-20 kts.
ΔFL	± 2 flight levels.

cover all the flight levels in this simulation so aircraft can fly over it. There are two aircraft on the weather cell, but the 3D view indicates that the cyan aircraft is above the weather cell while the blue aircraft is below the weather cell. Figure 41 displays the number of aircraft in the airspace and the number of maneuvers executed at each time step. At $t = 150$ min, the arrival rate of aircraft in the airspace was increased. Resolution maneuvers are computed every three minutes and maneuvers are instantaneously executed. The growing radius of avoidance accounts for a potential delay in the maneuver execution time. As the number of aircraft in the airspace increases, the number and frequency of maneuvers required also increases.

MATLAB was used to generate the feasible solutions and cluster the aircraft in the simulations. The MIP was solved with CPLEX. The overall computation time for one iteration is approximately one minute, using a standard desktop computer. CPLEX was running on parallel machines to deal with all the clusters. There were up to 25 clusters. Each cluster is solved in a few seconds.

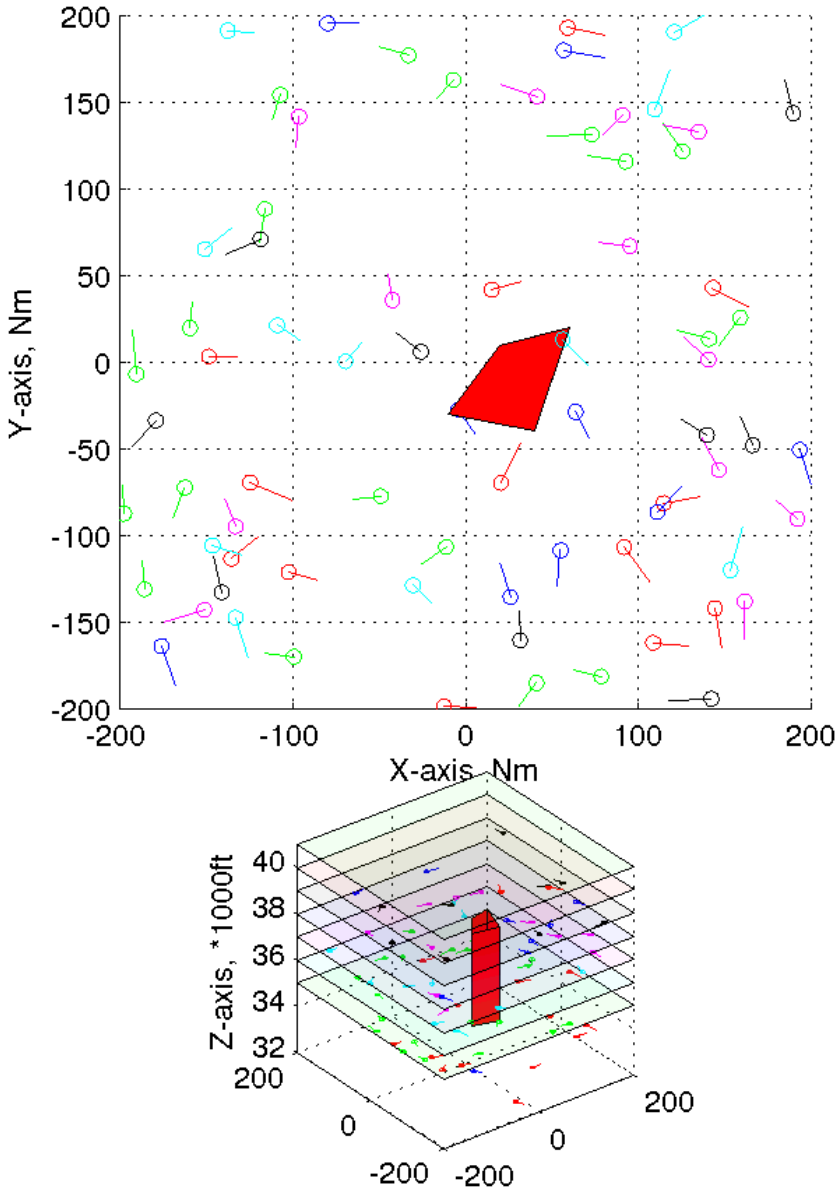


Figure 40: 2D and 3D view of the airspace during simulation

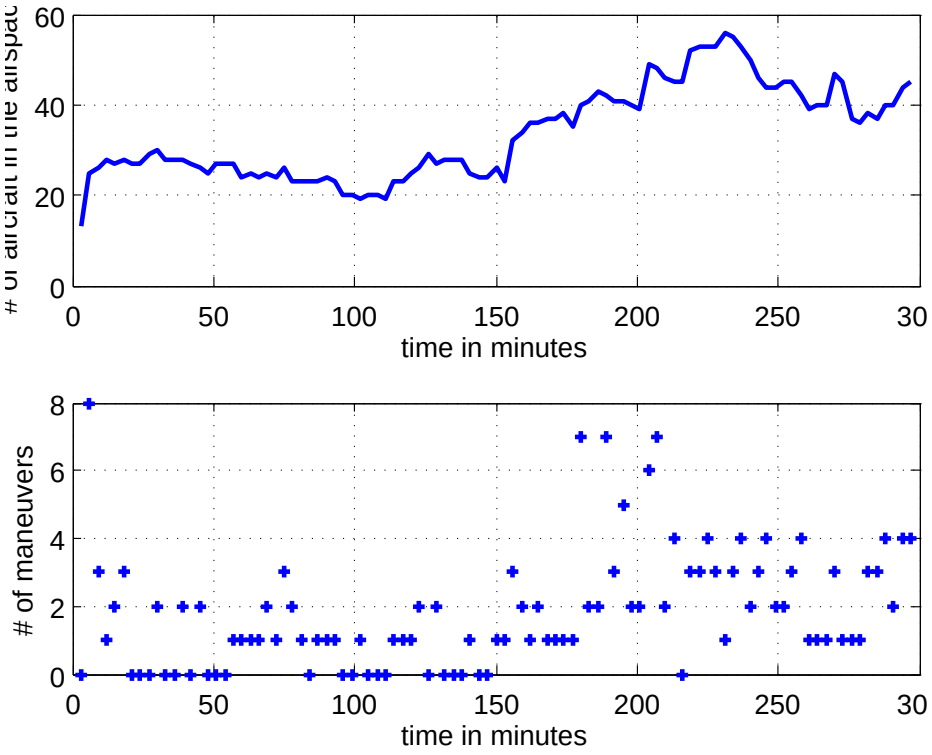


Figure 41: Number of aircraft and maneuvers.

4.6 Summary

This chapter addresses RQ 3 and RQ 4 for a class of operational degradations resulting from a failure in the CNS systems [45]. The principle of graceful degradation of air traffic operations against Communication, Navigation or Surveillance failures is outlined using a model that increases the required separation distance between aircraft.

Then, this model is used as a probe to compare Miles-In-Trail and Free-Flight approach scenarios and answer RQ 3, that is "What is the sensitivity of traffic configuration to potential degradation?".. It was shown that Free-Flight-like approaches do not degrade significantly worse than Miles-In-Trail scenarios when facing failures of the CNS system, when the initial distance between aircraft is slightly larger than the minimum separation distance. The metric used is purely computer based and does not account for the difficultythat a human operator might encounter with one or the other scenario. Probing traffic configurations for potential degradation in CNS systems is an important step in assessing the concepts of operations of NextGen and SESAR.

In support of this study, a new conflict resolution tool that applies to the transient conditions encountered during failures of the CNS system was developed. This algorithm provides near-optimal heading change avoidance maneuvers to increase separation distance and ensure a conflict-free configuration for a given time horizon. The avoidance maneuvers consist of heading changes.

Then, a more realistic algorithm of avoidance in the presence of uncertainties is introduced [46]. This algorithm provides simple avoidance maneuvers in 3 dimensions. Starting with a finite number of possible maneuvers for each aircraft, the set of maneuvers that ensures a conflict-free airspace is determined. Such a tool could be used as a backup system for future operations concepts. This centralized algorithm can handle hundreds of aircraft using clustering and parallel processing. This algorithm answered RQ 4, that is, "How can the smooth transition from nominal mode of operations to degraded mode of operations be ensured?" by proposing avoidance maneuvers to ensure a smooth transition from nominal mode of operation to degraded mode of operation. It was showed that such avoidance maneuvers can be computed in real time.

CHAPTER V

MONITORING OPERATIONS FOR POTENTIAL DEGRADATION: AIRSPACE MONITORING

5.1 Introduction

This chapter addresses RQ 5, that is "It is possible to monitor the airspace for potential degradations?". En-route and terminal traffic are studied in two different sections because of their different nature. During en-route operations, aircraft mostly fly straight legs, with only a few turns and a few flight level changes. During the terminal phase of the flight, aircraft trajectories are subject to much more vertical motion and possibly to vectoring.They may even be asked to fly a holding pattern. The movement of aircraft in terminal areas is more complex than it is during en-route phases.

This chapter introduces complexity measures for en-route and terminal area traffic. Airspace complexity has been and is still studied from different standpoints, but rarely with airspace degradation in mind. Delahaye and Puechmorel present maps using aircraft configuration based on non-linear dynamical systems [21]. Their measure of Lyapunov Exponents captures the structure and organization of a traffic situation, by identifying the organization of trajectories in a traffic pattern. The comparison of the relative complexity of different regions of the airspace is enabled using such maps. Lee proposed a set a complexity maps based on input-output approaches [78]. The response of the airspace to a disturbance is analyzed by measuring the amplitude of traffic deviation that would need to be undergone by traffic for all instances of an intruder aircraft. Both Delahaye and Lee's maps are traffic sensitivity to disturbance measures. Another broadly used approach to evaluate airspace complexity is to measure controller workload. In ATC, traffic complexity and controller's workload are often related [93]. Histon et al. [58, 59, 60] showed that structure and cognitive complexity are also directly related: controllers are more subject to errors when merging traffic with different merging points rather than with a unique one. Sridhar and al. [131]

propose a complexity measure based on dynamic density. Dynamic density is a combination of variables such as traffic density, number of aircraft with speed changes or number of aircraft with heading changes. The weights are determined using linear and non linear regressions from a subjective survey data. Complexity measures can be used for different purposes. Bilimoria and Jastrzebski proposed an aircraft clustering procedures based on airspace complexity [8]. Using complexity, separation and stability measures, they can produce highly complex cluster patterns that are well separated and stable. This complexity based work can improve operations if the overall ATM systems works in nominal mode. Yousefi and Donohue [147] propose complexity maps and airspace sectorization based on a measure of controller's workload. Dividing the airspace in hexagonal cells, they calculate the ATC workload for each of them over a day of simulation. Then, the cells are clustered to construct sectors of limited complexity.

Until this work, the question of airspace degradation had not been addressed in the literature. It is interesting to note that Delahaye and Puechmorel analyze the complexity using open-loop measures, while Lee uses closed-loop measures. From a system's control standpoint, Lee's "controllers" are conflict resolution algorithms. The resulting complexity map depends on the "controller". As shown in previous studies [90, 89], a traffic situation can be very difficult to manage by some control strategies and more easily by some others. The algorithm used and the complexity map generated are highly dependent. In the case of Workload based complexity maps, the measure is closed-loop. The controller is not automatic but the Air Traffic Controller. The resulting maps are very subjective as they are dependent on humans. Many factors affect workload. The same traffic situation presented to the same controller in different environmental conditions can result in different workloads.

Airspace monitoring has also been the topic of several research papers. Krozel [73] proposed an intent based monitoring where the aircraft is tracked relative to a filed flight plan, using NavAids and way-points. The monitoring tasks requires knowledge of the airspace structure, of the trajectory way-points and of the intent of the aircraft. It is a powerful tool when the flights behave according to their flight plan, but when dealing with arrivals, the sequence of way-points might change, some might be skipped or added to ensure an optimal

separation of aircraft at the runway threshold and vectoring is often used. This monitoring method cannot be used. Reynolds et al. [119] introduced a framework for the development of an automated conformance monitoring system. The system described in [119] has two main inputs: the conformance basis, containing target states and trajectory information, and the observation of a surveillance system. Those inputs feed models for pilots' intents, aircraft intents, and aircraft control systems and dynamics. Those models provide an expected state vector that is compared with the observed state vector for conformance analysis. This structure is further used in [118] to monitor the conformance of a trajectory to a flight plan. For instance, it detects if an aircraft does not turn, turns too early or too late at a way-point. The monitoring is based on intent, and knowledge of the the exact expected trajectory is required. An off-line trajectory analysis and taxonomy for arrival trajectories is proposed by Eckstein [26]. The objective of Eckstein is to analyze the performance of area navigation (RNAV) operations for NextGen concepts of operations off-line. The method in [26] uses GPS coordinates of actual way-points to identify and classify segments of trajectories. This approach can be used only if aircraft follow RNAV operations, which is not the case with the data used.

The remainder of this chapter is organized as follows: Section 5.2 focuses on en-route traffic and presents degradation maps that display information related to the degration complexity of the current traffic configuration. Section 5.3 focuses on terminal areas and analyzes the conformance of current flights to predetermined typical operations in real time. Finally, section 5.5 summarizes the results of this chapter.

5.2 En-route monitoring for CNS degradation

5.2.1 Degradation evaluator

A first step towards monitoring the en-route airspace for possible degradation of CNS systems is presented. When monitoring a system for conflicts, one considers the traffic and nominal behavior, together with nominal Communication, Navigation and Surveillance (CNS) infrastructure to predict the occurrence of future conflicts among aircraft. When monitoring system safety against possible degradations, one is concerned not only with

nominal, but also off-nominal behaviors triggered by these degradations. It is the response of the airspace to these *classes* of perturbations that allows us evaluate whether the airspace configuration under consideration is safe or not. "System response" includes not only the required control activity (required aircraft deviation from nominal flight path), but also in terms of the workload imposed on the air traffic controller (in terms, for example, of the number of conflicts to solve, and the number of utterances required for the controller to transmit the corresponding advisories to send to all aircraft). Figure 42 presents the en-

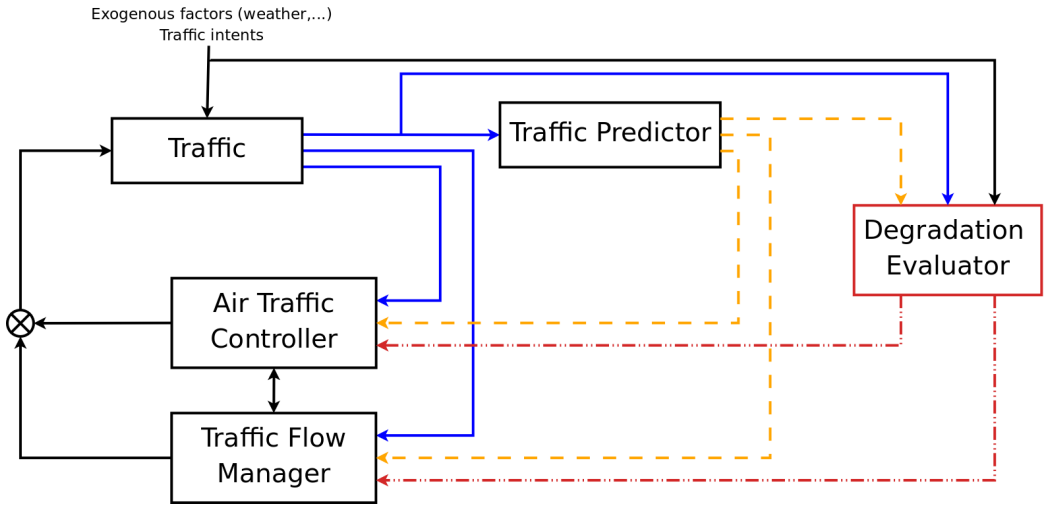


Figure 42: Degradation evaluator in the ATM environment

visionned air traffic health monitoring architecture. The main components of this airspace health monitoring system are a traffic predictor and a degradation evaluator. The traffic predictor plays an important role in monitoring the airspace, but is not the object of this thesis. It should propagate the traffic state forward in time. This state propagation effort would be supported by the current airspace state, together with intent information (flight plans or related quantities such as waypoints), current and future traffic flows out of airports as well as exogenous factors such as weather.

The degradation evaluator is a tool whose inputs are the current traffic situation, the predictions from the traffic predictor and information about exogenous factors, such as weather. Its aim is to provide controllers and traffic flow manager with information regarding the gracefulness of the predicted situation's degradation. Degradation prediction can

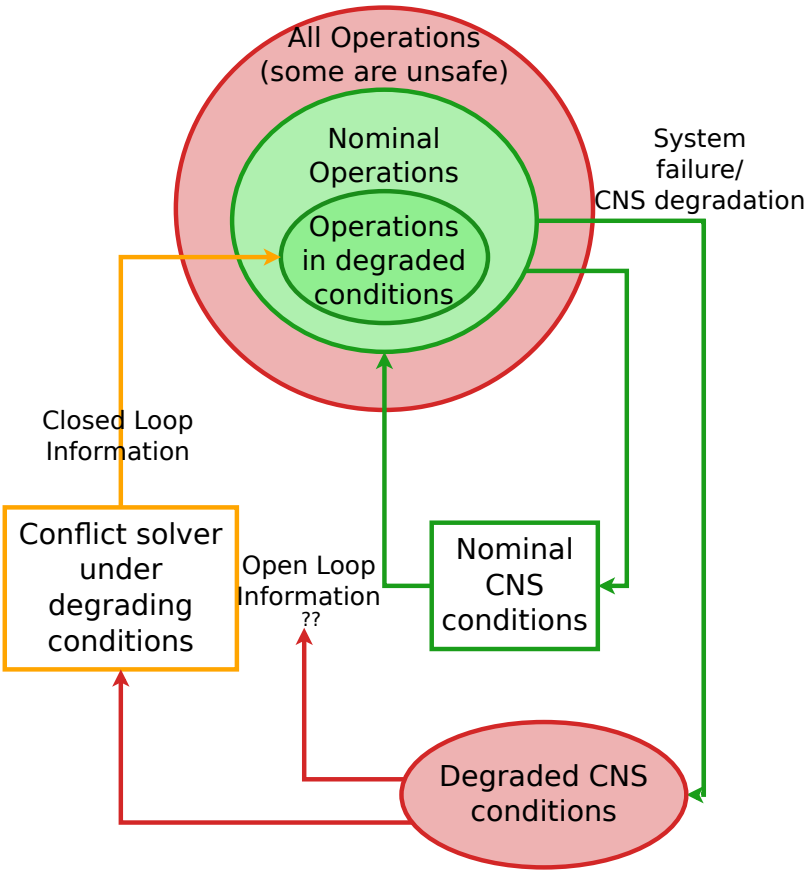


Figure 43: Open- and closed-loop information during a degradation of CNS conditions

be obtained using the results from Chapter 4, and for terminal areas, the optimal number of aircraft was determined in Chapter 3. Feeding back a degradation analysis to air traffic management would appropriately steer traffic away from sensitive situations.

To measure the ability of a traffic situation to "gracefully" degrade, this chapter introduces a new complexity measure based on degradation performances of a known traffic pattern. This complexity measure can be summarized in a map that are called "Dynamic degradation map". The novelty of this degradation maps is the mix of open-loop and closed-loop information as well as the focus on degradation perspective. Degradation maps are introduced to enable an airspace monitoring function. They are "Dynamic Degradation Maps" for the following reasons: They are *dynamic* as they evaluate a traffic configuration at a given time and can be frequently updated. They focus on the *degradation* of CNS performance. The smooth transition from nominal to degraded operations is handled by a

slightly modified version of the algorithm previously presented in section 4.4. Since the focus is on degradation, the initial configuration is conflict free, so that nominal conflict resolution activity does not interfere with the measure. The degraded operations considered here are more conservative than nominal operations since separation distances are increased. In case of a system failure or a CNS degradation, the traffic needs to be steered from nominal operations to degraded operations. Due to the degradation and the separation distance increase, new potential conflicts might appear. Degradation modelling and potential conflict detection are presented in section 4.2. A custom-built solver for degraded conditions conflict solver can be used to steer the system to safe degraded operations. Figure 43 summarizes these concepts.

This work deals with complexity from a system's control standpoint. The considered system is a traffic situation which states are aircraft's heading and position. The system is subject to constraints, such as separation distance. To enforce these separation distances, a control activity is required. Therefore, two important types of information appear, Open- and closed-loop information:

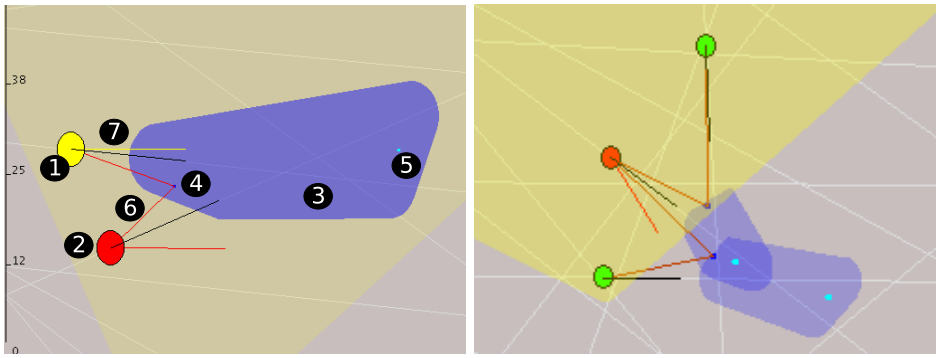
- **Open-loop information:** It is information obtained from the system without using a controller. In this case, it includes potential conflicts, their location, and involved aircraft.

- **Closed-loop information:** It is information provided by the control activity. It includes heading or speed changes, number of maneuvers. While open-loop information provides information about what would happen, closed-loop information provides a measure of the severity of the situation and the difficulty to handle it.

Open-loop information can be gathered from the degraded situation: new potential conflicts location, time, distance to aircraft and involved aircraft. The conflict solver provides closed-loop information to measure the severity of the situation. These informations are heading changes required or number of maneuvers.

5.2.2 Characterization of the degradation complexity

The degradation complexity comes from conflicts araising due to the increase of separation distances. To characterize this complexity, different methods and symbologes - shapes, opacity and colors - are used to represent the different kinds of available information. Figure 44(a) presents a small section of the display of a potential conflict on a dynamic degradation map. The term "potential conflict" is used since the aircraft are conflict free in nominal conditions, i.e the two circles of avoidance of radius r_0 will never intersect. Figure 44 presents two cases of aircraft in conflict. Figure 44(a) and 44(b) presents two



(a) Simple potential conflict

(b) Aircraft involved in two potential conflicts

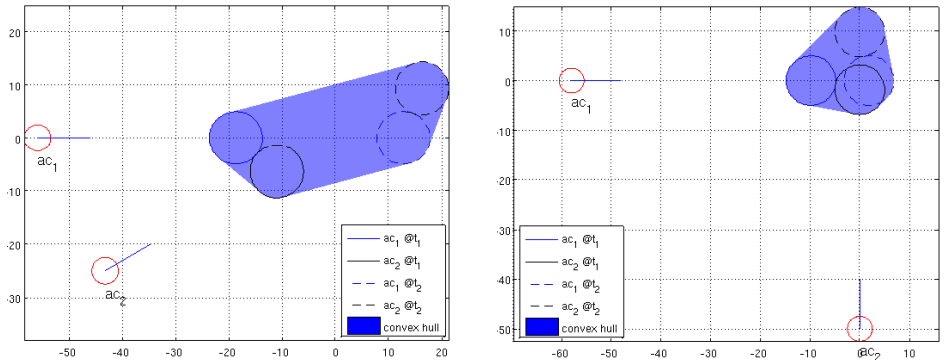
Figure 44: Display of potential conflicts for a dynamic degradation map

and three aircraft in potential conflict, respectively. Aircraft are located at the center of the colored circles. The black line is the aircraft velocity vector. The tip points at the position projected three minutes in the future. The circles have a radius $r_0 = 2.5$ NM, which ensures a 5NM separation distance. Numeration on Figure 44(a) corresponds to the following information:

Open-loop information:

- **Degradation Modeling** The model used is a degradation in CNS system as presented in Chapter 4, section 4.2.
- **Shape of the potential conflict area (3):** The darkened blue area corresponds the potential conflict area, defined by the convex hull of the circles of avoidance at the beginning of the potential conflict and at its end. Let define by beginning of the

potential conflict the time t_1 when the growing circles of avoidance would intersect for the first time if no action is taken. Let define by end of potential conflict the time t_2 when the growing circles of avoidance would stop interecting if no action is taken. Long shape indicates a small relative velocity V_{ij}. $V_{ij} = 0$ will lead to a shape that is not closed. Figure 45 presents the potential conflicts areas in the case of a low and a high relative velocity.



(a) Potential conflict with low relative velocity

(b) Potential conflict with high relative velocity

Figure 45: Potential conflict areas

- **Opacity of of the potential conflict area (3):** It is inversely proportional to the time to conflict. A very transparent blue area means that the potential conflict is far in the future. An opaque blue indicates that the potential conflict will occur in a near future.

- **Conflict location (4) and (5):** The conflict location is represented by a dark blue dot (4) and the end of conflict point by a cyan dot (5). The length between these points gives an indication on the conflict duration. If they are at the same location, it means that the circles of avoidance are tangent.

- **Color of the line joining the aircraft to the conflict location (6):** The color of the line joining the aircraft to the conflict location changes over time, varying from green for a long-term conflict to red for a short-term conflict. It helps identifying aircraft involved in a potential conflict and the time remaining to address the threat.

Table 5: Heading change used to calculate P_{ij}^k

	P_{ij}^1	P_{ij}^2	P_{ij}^3	P_{ij}^4	P_{ij}^5	P_{ij}^6	P_{ij}^7	P_{ij}^8	P_{ij}^9
$\Delta\psi_i$	$-30°$	$-30°$	$-30°$	$0°$	$0°$	$0°$	$30°$	$30°$	$30°$
$\Delta\psi_j$	$-30°$	$0°$	$30°$	$-30°$	$0°$	$30°$	$-30°$	$0°$	$30°$

- **Clusters:** Clustering is a commonly used tool in airspace complexity. Aircraft are clustered, based on possible avoidance maneuvers for all aircraft. A cluster defines a group of aircraft that can possibly be in conflict or interfere within a given time horizon, considering limited maneuvers for each aircraft. The time horizon is set to 15 minutes and the range of allowed maneuvers varies from $-30°$ to $30°$. Assume that aircraft i and j can change heading independently, and by no more than $30°$ in amplitude: If such a maneuver can lead to a potential conflict considering final radii of avoidance r_f for each aircraft, then they belong to the same cluster. Clusters can be merged using the rule that an aircraft cannot belong to two different clusters. Therefore, two aircraft that might not have any direct interaction can be in the same cluster. Algorithm 1 in section 4.5.1.4 is used to create the clusters of aircraft. Figure 46 presents the method used to determine if two aircraft belong to the same cluster. There are two aircraft in the relative axis, i.e aircraft j does not move in these axis. Let denote by P_{ij}^k the possible relative positions of aircraft i with respect to aircraft j after 15 minutes. P_{ij}^k are calculated for all the combinations of heading changes $\Delta\psi_i$ and $\Delta\psi_j$, as presented in table 5. The yellow shaded area is the convex hull of the circles of radius r_f centered on the P_{ij}^k and on the initial position of aircraft i. Aircraft i and j belong to the same cluster if the intersection of the yellow shaded area and a circle of radius r_f centered on the initial position of aircraft j is not empty.

  On the dynamic degradation maps, the yellow shaded areas represent the clusters, or more exactly the convex hull of circles of avoidance of the cluster's aircraft with final radii of avoidance r_f. Note that some aircraft might appear in the yellow area and not belong to the cluster.

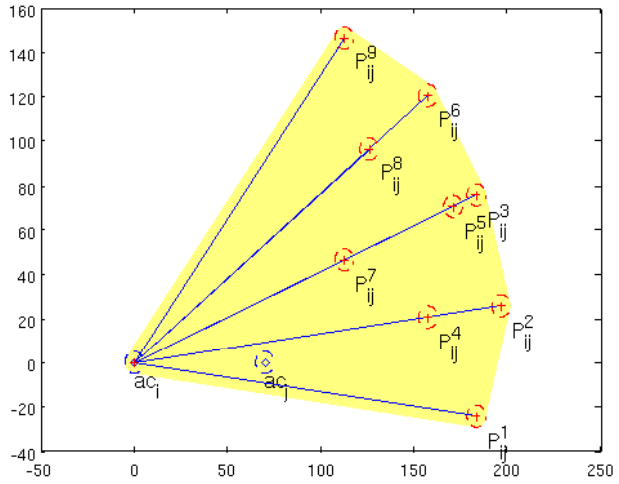


Figure 46: Possible relative position of two aircraft after a maneuver $< 30°$

Clusters are important to evaluate the difficulty to handle the traffic configuration: they determine the groups of aircraft that might interfere with each other, hence the constraints to be taken into account for separation purposes. A small cluster is easier to handle than a larger one. Considering n aircraft, the bigger the clusters, the higher the difficulty to handle the traffic configuration since more aircraft have to be taken into account at the same time. It is more difficult to solve an avoidance problem with many constraints than with no or few. On the maps, aircraft not belonging to a cluster lower the complexity as they don't require surveillance or control activity for the next 15 minutes. This approach to building clusters is different from Bilimoria and Jastrzebski's [8]. Aircraft are clustered using geometrical methods and the complexity is then measured, while Bilimoria and Jastrzebski first measure complexity and cluster aircraft from it.

Clusters also have an impact on the algorithm the algorithm is run separately for each cluster.

Closed-loop information:

- **Proposed solution (7):** The dynamic map proposes a solution made of heading

change instructions. The proposed solutions is provided by the conflict resolution algorithm under uncertainties presented in section 4.4. The algoritm was slightly modified to fit the needs of dynamic degradation maps: First, each proposed heading change $\Delta\psi_i$ is discretized such that $\Delta\psi_i = \pm 0°, 5°, 10° \ldots 30°$. Second, the clusters are computed. The presence of the clusters allows the algorithm to remove all the avoidance constraints between aircraft not belonging to the same cluster. Both changes have a beneficial effect on the computational time. It can be reduced by a factor up to 100.

- **Colors of aircraft (1) & (2):** The aircraft color is coded to reflect the amplitude of the maneuver required to ensure safe separation under degraded conditions. The colors correspond to the following heading change: green = 0°, yellow = ±5°, orange = ±10°, red = ±15°, purple = ±20°, black ≥ ±25°. In the example of Figure 44(a), aircraft (1) should turn left 5° and aircraft (2) turn right 15°.

After all the open- and closed-loop measures have be presented, a map of an ARTCC can be drawn.

5.2.3 Example of a dynamic degradation map for a center

Figure 47 presents the screenshot of a dynamic map for Cleveland Air Route Traffic Control Center (ZOB). The following explains the numbered items on Figure 47.

1. **Sector (1):** The map is presented for one flight level for eastBound traffic. The configuration contains 27 aircraft. The speed of all aircraft is 400kts.

2. **Clusters (2):** 5 clusters have been identified by the clustering method. There are also several isolated aircraft that do not belong to any cluster, because they cannot interact with any other aircraft. It is low complexity aircraft.

3. **Conflicts (3):** Three aircraft are in potential conflict. The proposed solutions consists on deviating two aircraft by 10° to the right for the first one and 15° to the right for the second one. The difference in opacity of the blue is slight indicating that potential conflict will happen almost at the same time.

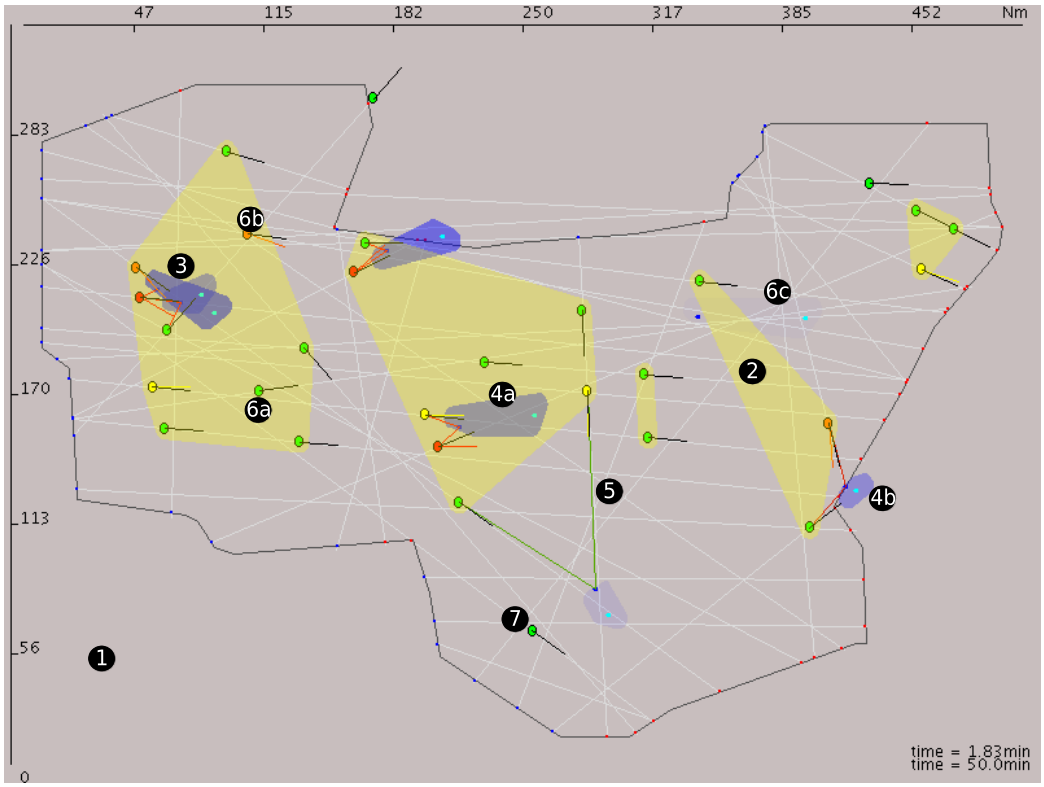


Figure 47: Example of dynamic degradation map

4. **Conflicts (4a) and (4b) :** The shape of the blue area indicates a low relative velocity in for conflict (4a) and a high relative velocity for conflict (4b).

5. **Conflict (6):** Aircraft (6a) and (6b) are in potential conflict. A solution to solve this potential conflict is for aircraft (6b) to turn right by 10°. The conflict area is located at (6c) and is barely visible as the conflict would occur in a quite long time. There is no line joining the aircraft to the conflict indicating that the conflict would occur in more than 20 minutes. Aircraft (6a) and (6b) are in the same cluster not because there is a potential conflict in more than 20 minutes, but because a heading change of 30° could lead them to a conflict with each other or with another aircraft of the cluster in less than 15 minutes.

6. **Aircraft (7)** does not belong to any cluster. If a degradation happens, there is no need for particular surveillance.

5.3 Airspace monitoring for terminal areas

In the previous sections, airspace monitoring for en-route traffic was enabled by dynamic degradation maps that propose avoidance maneuver to increase required separation distances between aircraft and provide controllers with a conflict-free configuration. Such maps cannot be applied to terminal areas traffic because of the nature of the trajectories. Aircraft do not fly straight path and the intent is to land or depart from airports. This section proposes an airspace monitoring technique for terminal areas traffic. It monitors the conformance of current flight path to typical operations for the considered terminal area. The degradation considered in this section is a deficiency of aircraft and pilots to follow typical operations. A monitoring tool that automatically detects when an aircraft is not conforming to typical operations is presented. Typical operations are determined using trajectory clustering methods presented in Appendix A. The objective of the monitoring task is to detect when an aircraft deviates from typical path in real time. In appendix A, typical operations are identified using trajectory clustering. That is, all similar trajectories are grouped together and "abnormal" trajectories are identified as outliers. For each cluster, the center of mass of all the trajectories of this cluster can be computed and is called centroid. The centroid can be seen as a typical operation with some variation around it to account for all the other trajectories of the cluster.

Chapter 3 focuses on minimizing the congestion of the TRACON and determines an optimal capacity. In this chapter, the focus is on identifying aircraft with a "abnormal" behaviors, which might result from a congested airspace. In Chapter 3, the TRACON is seen as a black box and the sequence of incoming aircraft is adjusted to optimize runway throughput and reduce congestion. This chapter "opens" the black box and looks at the trajectories of the aircraft inside the TRACON. The inputs to this analysis are the trajectories of the aircraft inside the TRACON.

5.3.1 Motivation for using data-driven methods

As presented in the introduction of this chapter, flight conformance monitoring tools have already been studied and developed. This chapter presents novel tools because the existings

tools cannot be applied in terminal areas. Figure 48 displays an satellite view of the San Francisco Bay Area. The blue circle represents the outer boundary of the TRACON, given by the area covered by the radar. The white lines are the centroids identified in Appendix A.5.4. The yellow dots are way-points or reporting points. The locations of those points come from Standard Terminal Arrival Routes (STAR) and track logs [41]. The centroids of the clusters pass over only a limited number of way-points. This shows that using published way-points and reporting points cannot be efficiently used to monitor traffic in the TRACON. Likewise, the intent based methods cannot be used in the terminal area, because aircraft and pilots follow the orders of air traffic controllers and not waypoints. This arrival procedure has not been identified by the clustering algorithm because of the relatively small number of aircraft using this route, and of the variability of the flight path following this procedure. During a visit at the Northern California TRACON, the controllers and managers confirmed that aircraft are not directed by waypoints or reporting points, but mainly by vectoring maneuvers given by the controllers. Therefore, it was decided to creaat

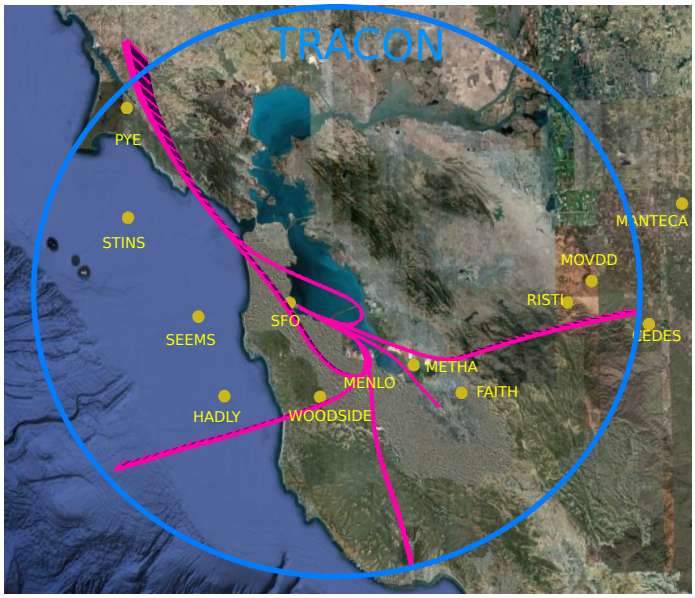


Figure 48: Centroids of the clusters and reporting points/way-points for SFO arrivals. Those centroids differ from the ones in Figure 61 because the algorithm was run with different parameters with a smaller sensitivity.

a real-time trajectory analysis tool built upon the knowledge gathered from the clustering analysis. The tool is called AirTrajectoryMiner (ATM) since it enables the monitoring of

operations in the TRACON. Current aircraft trajectories are compared against nominal trajectories, that is, the trajectories of the cluster of interest. If they differ too much, the current trajectory is tagged as abnormal, or outlier. The only intent used is the final destination airport for each aircraft. The tool automatically detects if the aircraft is flying one of the possible approaches, including most commonly used vectoring maneuvers.

5.3.2 Data formatting

It is not possible to directly compare the current trajectories with nominal trajectories, since current trajectories are incomplete. During real-time system operations, only past data is known. Therefore, the nominal dataset is fragmented. Re-sampled trajectories that had 50 points (see Appendix A) are split in 10 fragments of 5 points. The average travel time in the TRACON for aircraft landing at SFO is about 14 minutes. Therefore, 5 data points correspond to about $14 \times 60/10 = 84$ seconds. A memory of 80 seconds is used for current tracks. The radar hits of the last 80 seconds of flight are re-sampled to 5 points that can now be compared against the database of nominal tracks. This comparison is done using the Inductive Monitoring System.

5.3.3 Anomaly detection: Inductive Monitoring System

To detect abnormal trajectories, the Inductive Monitoring System (IMS) [68] is used. IMS is a good alternative to model based health monitoring systems. It provides a high fidelity detection tool, and there is no need to manually build a model. IMS runs in two steps: a learning phase followed by an anomaly detection phase. IMS learns the nominal behaviors using a training dataset provided by the user. IMS builds clusters using k-means clustering and density-based clustering [68]. During the anomaly detection phase, the input data is compared with the knowledge base built from the training data. The anomaly score can be interpreted as the distance to the nearest cluster. The input data belongs to a cluster if all the parameters values are within the range specified by the cluster limits.

The training dataset includes all the trajectories identified as nominal and fragmented into 10 segments of 5 points. The total number of segments was 276,040.

5.3.4 AirTrajectoryMiner: monitoring tool

AirTrajectoryMiner is a real time TRACON monitoring tool. The inputs to the tool are the set of all the trajectories identified as nominal, work resulting from Appendix A.5.4, and the radar tracks of the flights of interest. AirTrajectoryMiner delivers two types of outputs. On the one hand, it delivers an indication of conformance of current flight to nominal procedures, and on the other hand, it delivers a measure of the complexity in the TRACON that can be incorporated in Traffic Management Advisor (TMA) software.

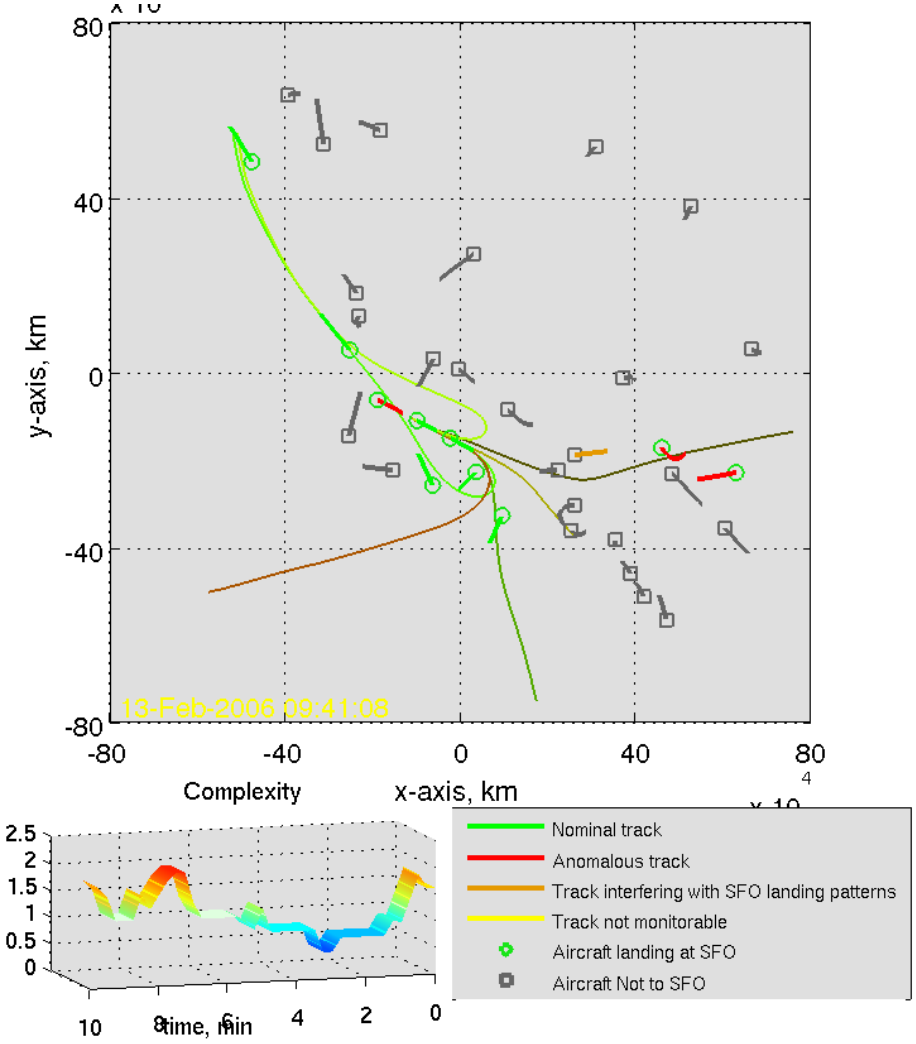


Figure 49: AirTrajectoryMiner display. Top frame: conformance to typical operations. Bottom frame: time history of complexity in the TRACON.

Figure 49 displays the monitoring environment. The top frame is a 2D view of the

airspace. The airspace correspongs to a cylinder of radius 80km, going from the ground up to 6,000 m and centered at OAK. The representation is in 2D but the track analysis considers the 3 dimensions of the tracks. The following provides some information about the display and associated aircraft count.

- **Green circle**: aircraft intended to land at SFO (associated count: n_{SFO}).
- **Grey square**: aircraft not intended to land at SFO (associated count: $n_{\overline{SFO}}$).
- **Green segment**: trajectory of an aircraft intended to land at SFO and following the procedures (associated count: $n_{OK,SFO}$).
- **Red segment**: trajectory of an aircraft intended to land at SFO and whose trajectory is identified as an outlier: it does not to follow the procedures (associated count: $n_{\overline{OK},SFO}$).
- **Grey segment**: trajectory of an aircraft not intended to land at SFO and whose trajectory does not interfere with traffic landing at SFO $n_{OK,\overline{SFO}}$).
- **Orange segment**: trajectory of an aircraft not intended to land at SFO and whose trajectory may interfere with traffic intended to land at SFO. IMS identified the trajectory as nominal for landing at SFO, but the flight is not inbound to SFO (associated count: $n_{\overline{OK},\overline{SFO}}$).
- **Colored lines**: Centroids of the clusters of trajectories identified as nominal. The centroids differ from the ones on Figure 61, because the clustering algorithm was re-run with different parameters allowing more variability in the clusters and therefore creating fewer clusters.

Aircraft intent information, i.e the destination airport, comes from the data. The length of the line following the aircraft corresponds to the part of the trajectory being analyzed, that is the last 80 seconds of the trajectory. The length of the line is therefore proportional to the velocity of the aircraft. This display is intended for an air traffic controller managing the arrivals at SFO. A similar display would be used for managing other arrivals or departures

for San Jose International Airport or Oakland International Airport. The only change would be the training data for IMS and the centroids displayed. Aircraft with a gray segment can be ignored, since they are not landing at SFO and are not interfering with landing traffic at SFO. Aircraft with a green segment are following the typical operations to land at SFO. Aircraft in orange require special attention since they are not intended to land at SFO but present characteristics that identify them as "in the pattern to land at SFO". They conform with some of the SFO landing trajectories. Aircraft in red also need special attention since they are supposed to land at SFO but currently not on typical tracks. The controller needs to make sure they are not generating conflicts or interfering with other traffic.

5.3.5 AirTrajectoryMiner: measure of complexity

Based on the compliance of current flights to procedures, this section defines a measure of complexity for the TRACON, which could provide an automatic feedback of the health of the TRACON to the traffic flow manager who regulates the flow of aircraft arriving in the TRACON. This complexity measure is directly linked to the analysis performed in Chapter 3. High complexity aircraft have trajectories that were named "rerouted" in Chapter 3. The optimization of section 3.5 determines a TRACON capacity to limit the number of such trajectories. According to [59, 92], controllers build a mental model of nominal operations. The complexity of a traffic configuration perceived by the controllers increases when an aircraft flight paths do not follow this mental model. When operations are running as expected, the controller is more efficient and can deal with more aircraft. Thus, increasing the number of aircraft not following nominal procedure reduces the maximum number of aircraft a controller can deal with simultaneously, reducing the capacity of the airspace. During a visit of the Northern California TRACON, the parameters that were identified as important to evaluate complexity are vectoring, aircraft on holding patterns, aircraft mix, i.e turboprops and jets at the same time, and aircraft types (e.g heavy, medium, light). The proposed monitoring tool identifies aircraft that are subject to large vectors, those flying on holding patterns and those executing a go-around as outliers. Turboprops, which increase controller's workload, are also likely to be identified as outliers since they fly

on dedicated routes that were not identified as typical by the presented clustering algorithm. In future work, those flights can be treated separately using the metadata to create clusters of turboprops only.

The proposed complexity measure is based on Shannon's theory of communication [127]. Let S_{SFO} be the sample space of all the aircraft inbound for SFO, in the TRACON at a given time. The random variable X defined on S_{SFO} assigns OK to the aircraft identified as nominal and $\overline{OK_i}, i = 1 \dots n_{\overline{OK},SFO}$ to the aircraft identified as outliers. For the aircraft identified as outliers, it is assumed that each outlier aircraft is unique and independent from other aircraft. At each instant, each outlier is considered different from the other outliers, that is there are $n_{\overline{OK},SFO}$ types of outliers. The resulting probability mass function $f_X(x)$ associated with the random variable X at a given instant is:

$$f_X(x) \begin{cases} \frac{n_{OK,SFO}}{n_{SFO}}, & x \in \{OK\} \\ \frac{1}{n_{SFO}}, & x \in \left\{\overline{OK_i}, i = 1 \dots n_{\overline{OK},SFO}\right\} \end{cases} \tag{65}$$

This means that the instantaneous probability of an aircraft inbound for SFO to be identified as nominal is

$$p(OK|SFO) = \frac{n_{OK,SFO}}{n_{SFO}}, \tag{66}$$

and the probability of an aircraft inbound for SFO to be a specific outlier is

$$p(\overline{OK_i}|SFO) = \frac{1}{n_{SFO}}, \quad i = 1 \dots n_{\overline{OK},SFO}. \tag{67}$$

The entropy H_{SFO} of the aircraft inbound to SFO is therefore

$$\begin{aligned} H_{SFO} =& - p(OK|SFO) \log p(OK|SFO) - \sum_{i=1}^{n_{\overline{OK_i},SFO}} p(\overline{OK_i}|SFO) \log p(\overline{OK_i}|SFO) \\ =& - \frac{n_{OK,SFO}}{n_{SFO}} \log \frac{n_{OK,SFO}}{n_{SFO}} - \frac{n_{\overline{OK},SFO}}{n_{SFO}} \log \frac{1}{n_{SFO}}. \end{aligned} \tag{68}$$

The same reasoning is used for aircraft not inbound to SFO

$$H_{\overline{SFO}} = - \frac{n_{OK,\overline{SFO}}}{n_{\overline{SFO}}} \log \frac{n_{OK,\overline{SFO}}}{n_{\overline{SFO}}} - \frac{n_{\overline{OK},\overline{SFO}}}{n_{\overline{SFO}}} \log \frac{1}{n_{\overline{SFO}}}. \tag{69}$$

The proposed measure of complexity C is the sum of the entropy of aircraft inbound to SFO and the entropy of aircraft not inbound to SFO

$$C = H_{SFO} + H_{\overline{SFO}}. \tag{70}$$

This complexity measure is an indication of the disorder with regard to nominal operations. If no aircraft is identified as an outlier, the complexity is 0. The complexity increases with the number of outliers detected, but also with the number of aircraft. The bottom left plot of Figure 49 shows this measure of the complexity over the last 10 minutes. The plot is refreshed every 15 seconds. When the traffic flow manager sees that the complexity increases, ATM provides information about the operations in the TRACON. If the complexity gets high, the controller in charge of the TRACON is likely to have a high workload. Providing the traffic flow manager with this complexity measure can help him manage the flow of arriving aircraft. A low complexity suggests that more aircraft can be allowed in the TRACON. Increasing complexity suggests that the TRACON controllers are subject to an heavy workload and that the aircraft arrival rate should be reduced.

5.4 Envisioned incorporation of the monitoring tools in the ATMS

Figure 50 shows how the Degradation Evaluator presented in section 5.2 and AirTrajectoryMiner presented in section 5.3 could be incorporated into the air traffic management environment.

From an ATM perspective, aircraft can be split in three categories: aircraft flying in terminal areas, aircraft in en-route phase, and aircraft not flying yet, but scheduled to, and represented as flight plans. All those aircraft are located and surveilled by surveillance systems such as secondary radars, ADS-B, etc. Those surveillance systems feed ATCs and TFMs softwares and displays. Using this information, ATCs close the loop by directing aircraft. TFMs act on flight plans to delay or reroute aircraft. TFMs are helped by the TMA [97]. The degradation evaluator introduced in section 5.2.1 can be used by controllers in ARTCCs as shown in Figure 50. The inputs to the degradation evaluator come from the surveillance systems, which feed the algorithm of avoidance under uncertainties. Those avoidance maneuvers are incorporated in the degradation evaluator to generate degradation maps (section 5.2.2). On the other hand, AirTrajectoryMiner can be useful for both TFMs

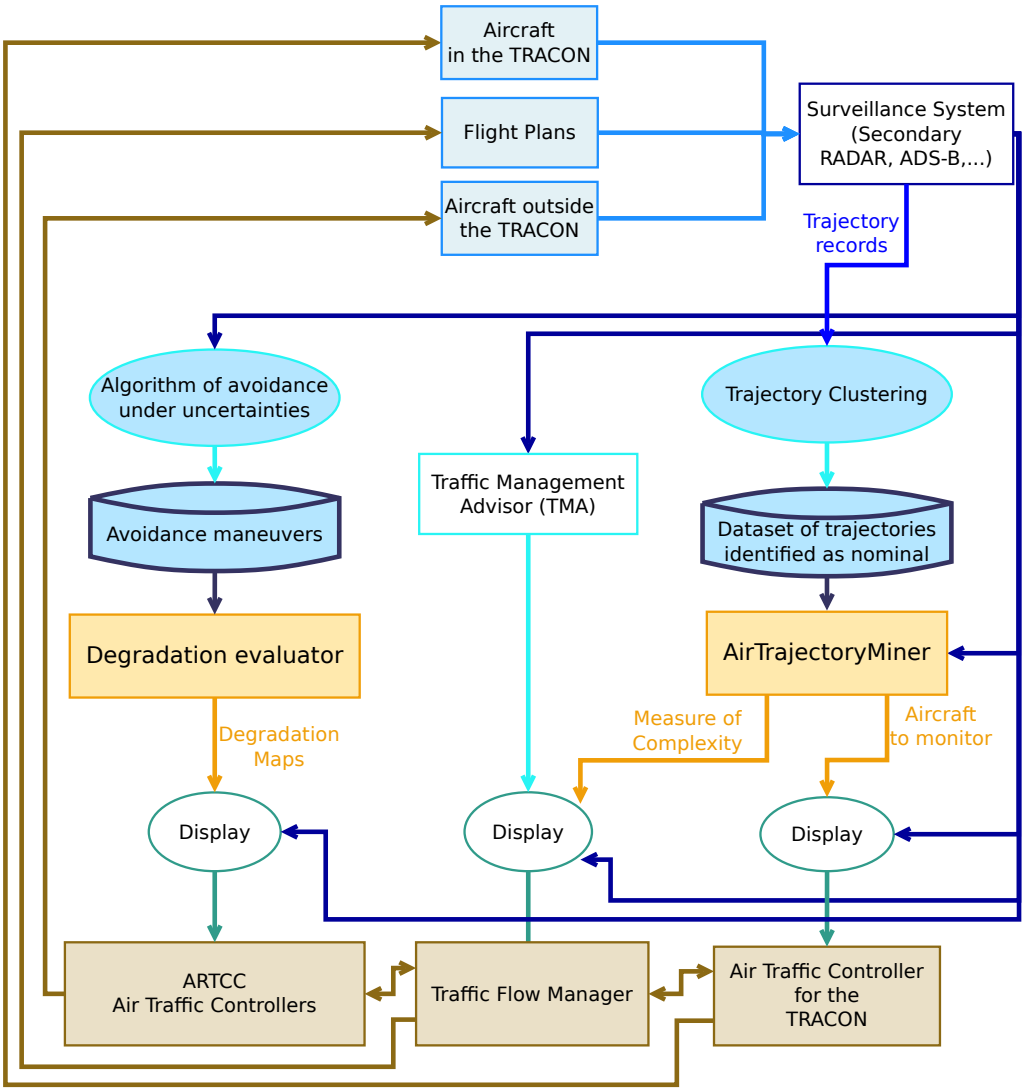


Figure 50: Schematic view of the air traffic control system - Integration of AirTrajectoryMiner and of the degradation maps

and controllers in TRACONs. Historical data from the surveillance systems can be used for learning typical operations and built a set of nominal trajectories. Then, in real time, flight tracks information from the same surveillance systems can be compared to the set of nominal trajectories to determine is the aircraft conforms to it. If not, this information is provided to the TRACON ATCs, and increases the value of the complexity that is fed to the TFMs. AirTrajectoryMiner can be used as an automatic independent monitoring tool.

5.5 Summary

This chapter adresses RQ 5, that is "Is it possible to monitor operations for potential degradations?". The question was answered by introducing two tools for airspace monitoring.

The first tool entitled degradation evaluator measures the ability of current traffic configuration to degrade in the event of a degradation of the CNS systems [47]. The mitigation action is an increase in the separation distance between aircraft. The degradation complexity associated with a traffic configuration is displayed using degradation maps that could be used by air traffic controllers in the event of a degradation. The maps provide a set of simple avoidance maneuvers that should be executed by the aircraft in order to maintain a conflict free situation. For instance, such maps could be available in each ARTCC for the other ARTCC in the event of the complete failure of one of them. Assume that ARTCC A completely fails. ARTCC B could use the map to direct aircraft previously controlled by ARTCC A. By spreading out traffic, controllers from ARTCC B would have time to manage aircraft from ARTCC A and ARTCC A would have some time to recover.

The second tool presented in this chapter focuses on monitoring the conformance of current flight tracks to pre-established typical operations [48]. The degradation monitored is a deficiency of aircraft to follow typical operations. Typical operations are learned from historical data and by using the trajectory clustering algorithm of appendix A. The monitoring tool AirTrajectoryMiner determines in real time if aircraft are conforming to typical operations. In addition, it provides a complexity measure that reflects the disorder in the terminal area. This information can be used by traffic flow managers to reduce the flow of incoming aircraft when the complexity is high. AirTrajectoryMiner could also be used by

air traffic controllers to help them focusing on aircraft with a higher complexity. This tool can be used by TRACON air traffic controllers and enables a feedback from the TRACON to the center.

CHAPTER VI

CONCLUSION

6.1 Thesis summary

Current Air Traffic Management Systems (ATMS) are aging and the introduction and use of new technologies is a prerequisite to achieving the level of traffic expected for the decades to come. Since the successful implementation of the concepts of operations described by NextGen and SESAR rely on highly automated systems, the new technologies need to coexist with the existing systems during technology modernization. In the event of a partial or total failure or a degradation of the automation, the safety of the airspace should not be jeopardized. This thesis addresses the problem of the degradation of such systems, i.e answering the questions: "What if a failure occurs in the ATMS? Is it possible to ensure the safety of the airspace?" To answer these broad questions, this thesis proposed 6 research questions (RQ 0-5). Each chapter addresses one or several of thoses questions.

The research questions address the main topic of this thesis, "Graceful Degradation of ATMS". The process of a safe degradation of performances in the event of a failure in the ATMS is defined by *graceful degradation*. A *graceful degradation* occurs when the transition from nominal mode of operations to degraded mode of operations is smooth and with no catastrophic event. It is now possible to proceed with the conclusions of this thesis.

The fundamental research question, RQ 0 for this work is addressed in the introduction by providing a list of failures that can affect the ATMS. Since the ATMS is a broadly distributed and heterogeneous system, it was decided that an exhaustive list was not the best solution. To be accurate, this list should be updated constantly to be current. The proposed list focuses on the main categories of failures and degradation that can affect the ATMS. To deal with the failures in a systematic way, an ontology of the ATMS is presented in Chapter 2. This ontology provides answers to RQ 1 and 2, by enabling the propagation and tracking of failures throughout the ATMS. As the failure propagates along

the link structure of the model, functionalities of pieces of equipments or people are disabled. Therefore, the execution of some tasks and eventually operations is not feasible anymore. The ontology propagates the failure and the resulting loss of capabilities is instantaneously known. The ontology provides the chain of events that result from a failure and its impact on operations.

Chapter 3 provides answers to RQ 3 and RQ 4 for a particular operational degradation, which is a reduction of landing capacity at SFO. The loss of capacity can be due to the late fog dissipation. An input-output queuing model for the NCT TRACON was developed, calibrated and validated using historical data. The model closely reproduces the behavior of SFO landing capacity when facing runway closures. The sensitivity of different queuing policies was analyzed using the model, in order to determine the optimal TRACON capacity that minimizes delays and congestions on one the one hand, and ensures a maximum runway throughput on the other hand.

Chapter 4 provides answers to RQ 3 and RQ 4 for another particular type of operational degradation: The degradation of Communication, Navigation and Surveillance systems. A conservative model for CNS degradation is developed. It accounts for a wide range of degradations, from a minor navigation degradation, to a surveillance failure. A degradation in navigation capabilities would cause aircraft to follow less accurately the intended trajectory, resulting in a reduction of the predictability from the controller's view point. A degradation in surveillance would result in the controller knowing with less accuracy the actual position of the aircraft. In both cases, the uncertainty on the aircraft position increases. Therefore, the model proposes the increase of separation distance between aircraft. This thesis proposes two algorithmic solutions to solve this new problem of avoidance under uncertainties. First, a near-optimal algorithm of avoidance in the presence of uncertainties is developed and is solved using Mixed Integer Programming. The outputs of the algorithm are the minimum heading changes required to provide a conflict free solution when the distance between aircraft must be increased. This algorithm is used as a probe to analyze the sensitivity of two types of operations to CNS degradation: Free-Flight like arrivals and Miles-In-Trail like arrivals. The conclusion is that Free Flight does not appear to be significantly more

problematic from the standpoint of CNS degradation. The comparison measure is the total heading change required to always ensure a safe separation between aircraft when radii of avoidance increase. This degradation measure does not reflect the issues related with the difficulties a human controller might have handling the less organized traffic arising from Free-Flight operations. The second implementation presented is an algorithm of avoidance in the presence of uncertainties, in 3 dimensions. The algorithm differs from the previous one as it determines a set of simple avoidance maneuvers for each aircraft, from a predetermined pool of possible solutions. The resulting traffic configuration is conflict free for a given period of time that allows controllers to recover from the failure. The avoidance maneuvers account for sector boundaries, aircraft performances and weather. The problem is also solved using Mixed Integer Programming. This algorithm provides solutions to ensure the smooth transition from nominal mode of operations to degraded mode of operations, by means of avoidance maneuvers.

Chapter 5 focuses on the monitoring of operations and answers RQ 5. For en-route traffic, the degradation of interest is a failure in the CNS systems. A monitoring environment using degradation maps is presented. Degradation maps present a simple and efficient way to visualize the complexity of a traffic configuration to gracefully degrade. Using a set of colors and shapes, the maps display open- and closed-loop information. Open-loop information include clusters of aircraft that should be dealt with together and potential conflicts resulting of the increase in separation distances. Closed-loop information include quantized heading changes. For terminal areas, the degradation of interest is a deficiency of aircraft to follow typical operations. A real-time conformance monitoring tool is developed. The tool analyzes the conformance of current flight paths to pre-identified typical operations. Due to the nature of traffic in terminal areas, monitoring conformance using published waypoints and reporting points is not possible. Typical operations were learned using trajectory clustering methods run on month of traffic data in the NCT.

Appendix A presents two trajectory clustering methods. The first method is based on the identification and grouping of turning points in the trajectories into waypoints. Then, trajectories going through the same waypoints are clustered using the longest common

subsequence of waypoints. The second method is based on a principal components analysis of the resampled and augmented trajectories. First, the trajectories are resampled to have the same number of data points. Then, some dimensions are added by computing extra parameters like the heading, from the initial data. After running a principal components analysis on this set of trajectories, the projections of the trajectories onto the first 5 principal components are clusters using DBSCAN, a density based clustering algorithm. The results of this methods are visually good. There exists no objective metric to validate those results. If such a metric existed, it would be used to optimize the clustering results.

6.2 Future work

To make a full use of the ontology proposed in Chapter 2, a shared, a web-based implementation might be necessary. Since not everyone is an expert on the entire system, each expert could contribute their local knowledge to the ontology. Then, it would be easy to add a new block that represent a new technology, task or concept of operation. By introducing failures at different levels, the robustness of the new ConOps can be evaluated. A collaboration with experts from different fields in the ATMS is necessary to ensure a useful model.

The safe degradation process in terminal areas is still a research problem. An important question is "Is it possible to design a system such that, in the event of a TRACON ground control failure, all aircraft can land safely and in a timely manner using self-separation". For en-route, such a problem is called Free-Flight and is the topic of several research programs. It is also under practice and study above the Mediterranean region [33]. The problem is more complex for the terminal area. The idea is to have aircraft with no outside visibility fly VFR operations. With the advances of ADS-B, pilot's awareness of surrounding aircraft increases. Algorithm to prove the feasibility and robustness of such operations are required.

APPENDIX A

TRAJECTORY CLUSTERING

A.1 Introduction

This appendix presents two methods to cluster trajectories and identify flights that followeidentical air routes. The first method is based on the identification of way-points in the trajectories, and the second method is based on a principal components analysis of re-sampled trajectories. Operations in the terminal area are managed by ATCs and are not part of the flight plans. It was therefore decided not to use any flight plan knowledge or aircraft intent other than the destination airport. The methods developed in this appendix are neither location nor data specific and can easily be adapted to other data sets since unsupervised methods are used, and the data is not labeled. This appendix considers radar tracks in the NCT. However, the underlying principles may also be used for other applications, such a flow reconstruction for en route airspace [85, 121]. The use of positioning devices such as GPS and the collection of data has increased over the past 15 years, leading to an increasing number of tracking applications. An objective of tracking is to discover common patterns on the one hand, and detect outliers on the other hand.

Piciareli et al. [106] presented an on-line trajectory clustering method for real time video surveillance. Moving objects, such as pedestrians, are identified in video frames and their trajectories are compared against existing cluster representatives, that is, an average of all the trajectories in the cluster. The match between a trajectory and a cluster is determined using the mean of the normalized distances of every trajectory point to the nearest point of the cluster representative. If a match is found, the cluster representative is updated. If not, a new cluster is created. In this approach, the cluster representatives evolve with time. This clustering method was used by Dahlbom and Niklasson for coastal surveillance but failed to provide satisfactory results when dealing with real data sets [17] such as ship trajectories.

Lee et al. [77] presented a partition-and-group framework for trajectory clustering. Trajectories are partitioned in sub-trajectories. Sub-trajectories are represented by line segments and grouped using a distance function. The distance function incorporates three components that measure the perpendicular distance, the parallel distance and the angular distance between the line segments. The clustering algorithm is density based, i.e clusters are created where the density of points is highest. The formulation is powerful but the results are presented on very noisy data where it is difficult to visually cluster the trajectories. There exists no well-defined measure to assess the results of the clustering method. Based on the same distance measure, Lee et al. [76] present a trajectory outliers detection procedure. The results are presented on the same noisy datasets and are therefore difficult to evaluate visually.

Vlachos et al. used similarity functions based on the longest common subsequence (LCS) to discover similar multidimensional trajectories [141]. Their LCS based clustering method appears to be more efficient than Euclidean distance based measures and dynamic time warping distance functions, especially in the presence of noise.

Eckstein proposed an automated flight track taxonomy [26] . The trajectories are first resampled, then clustered using k-means on a reduced order model. The model reduction is the truncation of a proper orthogonal decomposition (POD) , also called principal components analysis. The trajectories are clustered using only the first two modes of the decomposition, as they capture 95% of the fluctuations of the dataset used.

The first section presents the data set used for the study. The second section presents the waypoint based methods and the third section presents the principal components based method.

A.2 Available data

The available data [1] consists of records of flight tracks over the San Francisco bay area, for the first 3 months of 2006. The records cover NCT, that is, a cylinder of radius 80km and height 6,000m centered on Oakland International Airport (OAK). The NCT contains

[1]The complete dataset is available for download `https://dashlink.arc.nasa.gov/data/flight-tracks-northern-california-tracon/`

3 main airports − Oakland, San Francisco (SFO) and San Jose International airports − as well as many smaller airports. The NCT is the fourth busiest terminal area in the US [37] with an average of 133,000 flight instrument operations per month in 2006. The data, made of the position and speed of aircraft, is organized by flight and also contains meta-data for each flight that include: type of operation (departure/arrival), origin and destination airports, aircraft type (business, jet, helicopter, other, etc), date and time of beginning of record, duration of the record, etc.

Using the available meta data, visual flight rules (VFR) traffic is discarded, since it is more unpredictable and does not follow the same rules as instrument flight rules (IFR) traffic. The meta data is used to sort trajectories by airport and operation type, i.e. take off or landing. After a visual analysis of the flight patterns for the different airports, it was decided to focus the study on the landings at SFO. It is the busiest airport in the NCT and the arrival tracks present the most interesting patterns by their numbers and variety. The most frequent configuration is the "West" configuration, where aircraft land on runways 28L/R and take off from runways 01L/R. A diagram of SFO is presented in Figure 10, and Figure 11 depicts the NCT traffic patterns typically used in the west configuration.

In this work, the axes are set by the radar, located at $(0, 0, 0)$. The x and y axes define the horizontal plane and z the vertical direction, positive going upward. To each recorded flight, corresponds an aircraft i and a trajectory $T_i, i = 1 \dots r$, where r is the total number of trajectories of interest in the dataset. Each trajectory T_i is a $n_i \times 4$ matrix, and the line T_i^l of T_i is the l^{th} radar echo, given by $T_i^l = (x_i^l, y_i^l, z_i^l, t_i^l)$, where (x_i^l, y_i^l, z_i^l) is the 3 dimensional coordinates of aircraft i at time t_i^l. The trajectories have different numbers of points n_i, varying from 10 to about 550 points, depending on the duration of the trajectory. Trajectories with a few data-points usually correspond to short flights from SJC or OAK to SFO. The interval between points is between 4 and 5 seconds and is given by the rotational speed of the radar (most likely 4.8 sec). The time stamp t_i^l is rounded to the nearest second.

A.3 Overview of k-means and DBSCAN Clustering algorithms

This section introduces the two clustering algorithms used in this appendix: k-means [87] and DBSCAN [31].

Overview of k-means [87] This paragraph presents a brief overview of the k-means algorithm. For more details, the reader is refereed to [54]. Given a set $S = (\mathbf{tp}_1, \dots \mathbf{tp}_{|S|})$ of $|S|$ observations (turning points in our case), where each observation is a d-dimensional real vector, then k-means clustering aims at partitioning the $|S|$ observations into k sets, or clusters, $(k < |S|), C = \{C_1, C_2, \dots, C_k\}$ so as to minimize the within-cluster sum of squares:

$$\arg\min_{\mathbf{C}} \sum_{i=1}^{k} \sum_{\mathbf{tp}_j \in C_i} \left\| \mathbf{tp}_j - \mathbf{m}_i \right\|^2 \tag{71}$$

where $\mathbf{m}_i$ is the mean of C_i. The mean $\mathbf{m}_i$ of a cluster is called centroid and is the center of mass of all the elements in the cluster. The number k of clusters is the only input required from the user.

Starting with an initial set of k centers $\mathbf{m}_1^{(1)}, \dots, \mathbf{m}_k^{(1)}$, which may be specified randomly or by some heuristic, the algorithm proceeds by alternating between two steps, also known as Lloyd Algorithm [82] :

Assignment step: Assign each observation to the cluster with the closest mean, that is partition the observations according to the Voronoi diagram generated by the centroids of the clusters. Figure 51 presents the results of k-means clustering and the corresponding Voronoi diagram.

$$C_i^{(t)} = \left\{ \mathbf{tp}_j : \left\| \mathbf{tp}_j - \mathbf{m}_i^{(t)} \right\| \leq \left\| \mathbf{tp}_j - \mathbf{m}_{i^*}^{(t)} \right\|, \text{ for all } i^* = 1, \dots, k \right\} \tag{72}$$

Update step: Calculate the new means $\mathbf{m}_i^{(t+1)}$ to be the centroid of the observations in the cluster

$$\mathbf{m}_i^{(t+1)} = \frac{1}{|C_i^{(t)}|} \sum_{\mathbf{tp}_j \in C_i^{(t)}} \mathbf{tp}_j. \tag{73}$$

The algorithm is deemed to have converged when the assignments no longer change. Since it is a heuristic algorithm, there is no guarantee that it will converge to the global optimum, and the result may depend on the initial clusters. Since the algorithm is usually

very fast, it is common to run it multiple times with different starting conditions and keep the run that resulted in the minimum value for Eq. (71).

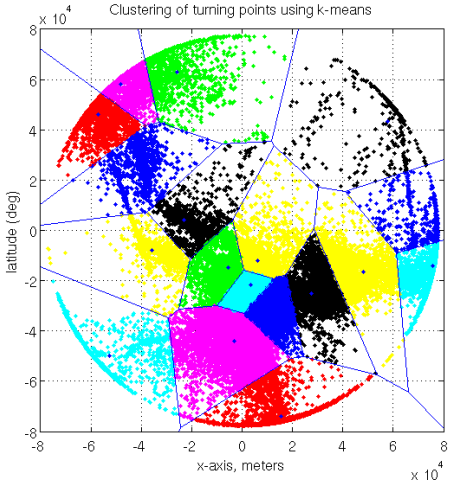


Figure 51: Clusters of (turnings) points using k-means and corresponding Voronoi diagram

Overview of DBSCAN This paragraph presents a brief overview of the DBSCAN algorithm. For more details, the reader is referred to [53]. DBSCAN [31] stands for Density-Based Spatial Clustering of Applications with Noise. DBSCAN clusters points that are close together (in an ϵ neighborhood), and surrounded by sufficiently many points. DBSCAN requires two parameters: a real number, ϵ, and the minimum number of points, $MinPts$, required to form a cluster. The ϵ-neighborhood of a point p consists of all the points q such that $dist(p,q) \leq \epsilon$. If the ϵ-neighborhood of a point p contains more than $MinPts$, a new cluster is started, with p as a core object. DBSCAN then iteratively collects directly density-reachable objects from these core objects. An object q is said to be directly density-reachable from an object p if q is in the ϵ-neighborhood of p and p is a core object.

If a core object q of a cluster C_i is added a cluster C_j, C_i and C_j are merged. When no point can be added to any cluster, the process terminates.

A.4 Way-point based trajectory clustering

This section presents a novel algorithm for aircraft trajectory clustering. This algorithm takes advantage of aircraft trajectory properties: aircraft usually fly straight, with a limited

number of turns. This method arises from the current instrument flight rules procedures. When approaching an airport, aircraft usually follow published procedures made of a sequence of way-points. A way-point is characterized by its GPS coordinates and, sometimes, an altitude indication. The planar localization of a way-point is very accurate but its vertical component often looks like "at or above —ft". Vertical clearances are delivered by ATC and trajectories' vertical profiles are then at the discretion of the pilots. Therefore, this method focuses on the 2D coordinates of the way-points in the (x, y) plane. This method is an efficient way to determine the compliance of flown trajectories with published procedures. Nevertheless, published procedures cannot be used because of the limited number of way-points or reporting points located in the TRACON. In Section 5.3, we further show this by comparing the results of the trajectory clustering with the published way-points.

The objective is to identify and group the turning points into "way-points". A turning point is a point in the trajectory where the aircraft changes heading. Then trajectories are represented by a sequence of way-points. Finally, trajectories are clustered by determining the Longest Common Subsequence (LCS) . The algorithm proceeds using the following steps and it is summarized in Figure 52:

1. Identify the location of the turning points of each trajectory.

2. Cluster the set of all the turning points of all the trajectories. This clustering task is done using k-means [86, 54] or DBSCAN [31] (Density-Based Spatial Clustering of Applications with Noise). Section A.3 gives an overview of those algorithms. This clustering provides a finite number of way-points where it has been determined that aircraft usually turn.

3. Represent each trajectory by its sequence of way-points.

4. Cluster the sequences of way-points using the SequenceMiner algorithm [12, 13]. SequenceMiner provides us with a representative trajectory for each cluster.

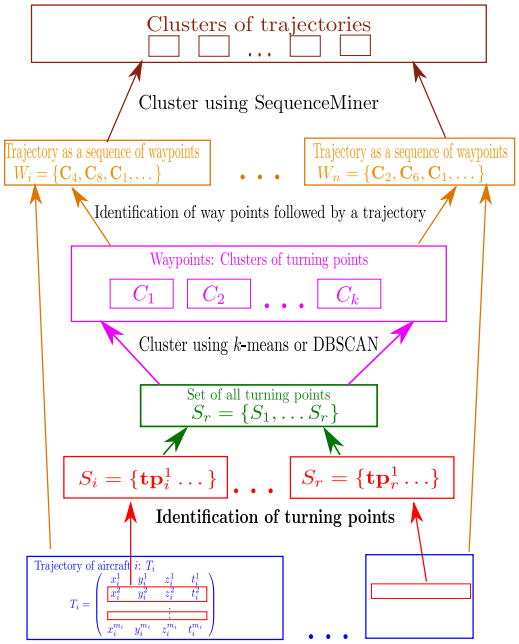


Figure 52: Way-point clustering method

A.4.1 Turning points identification

The first step is to extract the location of the turning points of each trajectory. To simplify the notations, the aircraft index i is omitted in the following equations. The heading Ψ^l of an aircraft at time t^l can be estimated by $\psi^l = \arctan \frac{y^{l+1} - y^{l-1}}{x^{l+1} - x^{l-1}}$, at each point of the trajectory, l, $l = 2 \dots n-1$, where n is the total number of points. Since the trajectory is a bit noisy, a low pass filter is applied:

$$\tilde{\psi}^1 = \psi^1 \tag{74}$$

$$\tilde{\psi}^l = \alpha \psi^l + (1-\alpha)\tilde{\psi}^{l-1}, \quad l = 2 \dots n-1, \tag{75}$$

where α is a constant for the filter. On this data, setting $\alpha = 0.4$ provided good noise filtering results and not too much delay. A turning point **tp** is identified when the heading difference between two consecutive values of the heading exceed a threshold: $|\tilde{\Psi}^l - \tilde{\Psi}^{l-1}| > \Psi_c$. The threshold was chosen relatively small in order to capture small heading changes but not small enough not to capture meaningless heading changes variations: $\Psi_c = 0.025\text{rad} = 1.43°$. This value was set experimentally. The results are not very sensitive to a small change in Ψ_c. The number of turning points is trimmed to avoid long sequences when aircraft are executing

large turns: if two consecutive data-points are determined to be turning points, then only the first one is kept; if three, only the middle one, etc.

The trajectory of aircraft i is now represented as a sequence of turning points S_i :

$$S_i = \left\{ \mathbf{tp}_i^1 \dots \mathbf{tp}_i^s \right\},$$

where $\mathbf{tp}_i^s$ is the 3D coordinate of the s^{th} turning point of trajectory i. The first point of the trajectory is labeled as a turning point. Figure 53 presents a sample of 11 trajectories and the points identified as turning points.

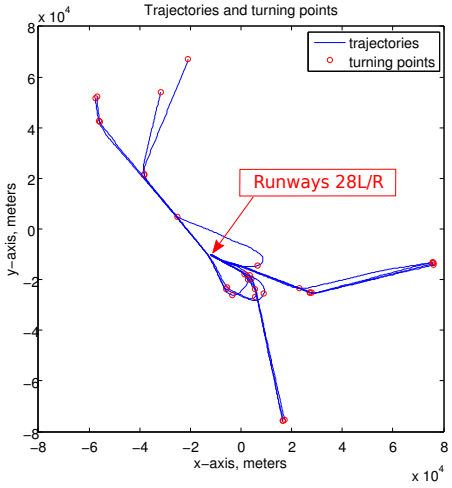


Figure 53: Trajectories and identified turning points for sample trajectories

Denote by S the set of all the turning points for all the trajectories: $S = \{S_1 \dots S_n\}$.

The second step is to cluster the set S of turning points.

A.4.2 Turning points clustering; creation of way-points

To determine the way-points, the turning points are clustered: a way-point is defined as the planar (x, y) coordinates of a cluster of turning points. The idea is to create a way-point where it has been determined that many aircraft turned. Depending on the number and on the density of available turning points, two different algorithms are used. When the spatial distribution of turning points is sparse, k-means is used, and when the distribution of turning points is dense, DBSCAN is used.

A.4.2.1 Case when the data is sparse

When the number of turning points is small, a density based clustering algorithm would provide poor results, identifying most of the points as outliers. Therefore, a distance-based algorithm is used so all the turning points available are used. A way-point is created for each cluster produced by k-means. Using cylindrical coordinates, the coordinates of the center of a way-point are given by (r_m, θ_m). The center is the center of mass of all the points in the cluster. The coordinates of the corners of the way-points are given by $\{(r_m + 2std_r, \theta_m + 2std_\theta), \quad (r_m - 2std_r, \theta_m + 2std_\theta), \quad (r_m - 2std_r, \theta_m - 2std_\theta), (r_m + 2std_r, \theta_m - 2std_\theta)\}$, where std_r and std_θ are the standard deviation of the radial coordinates and angular coordinates of the points in the cluster, respectively. Figure 54 presents the outcome of clustering the way-points for one day of trajectories. Each cluster is represented using a different color/shape combination. The way-points are represented by pairs of nested polygons on the figure. The inside polygon corresponds to $(r_m \pm std_r, \theta_m \pm std_\theta)$ and the outside one to $(r_m \pm 2std_r, \theta_m \pm 2std_\theta)$. The number is the label of the cluster.

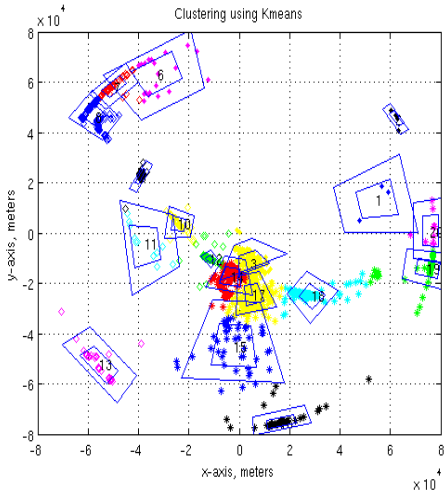


Figure 54: Result of the clustering of the turning points for one day using k-means

A.4.2.2 Case when the data is dense

When the the number of turning points is large, a large share of the airspace is covered with turning points. A distance based algorithm such as k-means provides meaningless clusters for our application. Figure 51 shows the clusters provided by k-means and the

corresponding Voronoi diagram for the turnings point of almost 3 months of data (30,000 trajectories).

To overcome this issue, the turning points were clustered using DBSCAN. DBSCAN is particularly efficient at cluster data in the presence of noise. Way-points are created using the convex hull of the clusters resulting from DBSCAN. Figure 55 shows the result of the clustering of the turning points using DBSCAN. The blue polygons represent the way-points. All the points identified as outliers, i.e not associated with any way-point, are not depicted. The parameters used were $\epsilon = 350$m and $minPts = 10$. The main issue with DBSCAN is its execution time since its complexity is in $O(n \log n)$. Here, the number of turning points to cluster is $n = 118,179$ for 30,000 trajectories.

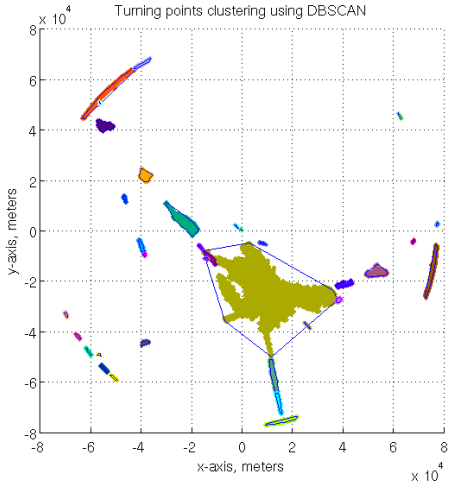


Figure 55: Result of the clustering of the turning points for the entire dataset using DBSCAN. Outliers are not displayed

A.4.3 Converting a trajectory into a sequence of way-points

The way-points have been discovered using the turning points of the trajectories. Nevertheless, some trajectories might go over way-points without actually turning. To identify the sequence of way-points followed by a trajectory, the following procedure is used for each trajectory: start with an empty sequence of way-points, and given the set of all way-points, run the trajectory along its original direction. If one of the points is located over a way-point, the way-point is added to the sequence. Each trajectory is now represented

as an ordered sequence of way-points, where the number of way-points is finite. The next step is to cluster the trajectories determining the longest common subsequence (LCS) of way-points.

A.4.4 Longest common subsequence determination

The sequences of way-points are clustered using the longest common subsequence. The LCS problem is to find the longest subsequence common to all sequences in a set of sequences. SequenceMiner [12, 13] is an algorithm that identifies the LCS and generates clusters of sequences. With this method it is possible to cluster sequences that do not contain the same number of elements. When sequences only have a small number of points, say fewer than 3, this clustering method does not work well. Therefore, only the sequences containing more than 4 way-points are kept. The total number of way-points being small, it is preferable to focus on the way-points at the beginning of the trajectory: since most aircraft do a final turn to get aligned with the runway, this turning point does not bring much information about the trajectory. Therefore, if the last turning point is in the large brown cluster (Figure 55), it is removed from the sequence.

Figure 56 presents the results of the clustering process using k-means and LCS on a low number of trajectories. The dataset used is the tracks of all the aircraft landing at San Francisco (SFO) airport on February 10, 2006. Only the trajectories of that day were used to determine the way-points. Each cluster is represented by a color. The algorithm identifies the main flows but a few trajectories seem not to belong to the expected cluster. The quality of the results is subjective and can only be visually assessed. Figure 57 presents the results for an initial set of 30,000 trajectories, using DBSCAN and LCS. Here, the denomination "Nominal" qualifies the trajectories containing more than 4 way-points. The colors correspond to the clusters. The colors differ on Figures 56 and 57 because the indexing of clusters is random and depends on the order of the data in the dataset.

Overall, this method presents good clustering results. One of the main drawbacks of this method is that it only keeps the trajectories going over the way-points. For instance, consider two parallel trajectories: one going over the way-points and the other one slightly

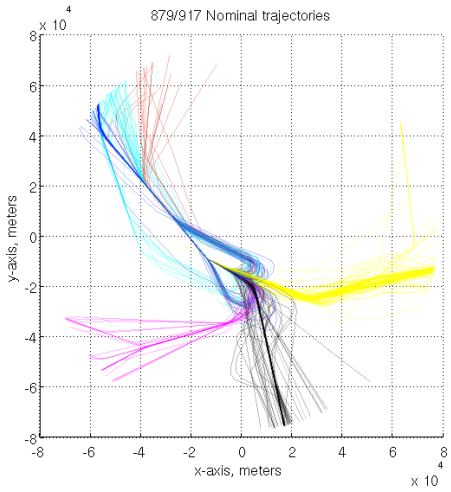


Figure 56: Results of trajectory clustering for the landings of one day at SFO

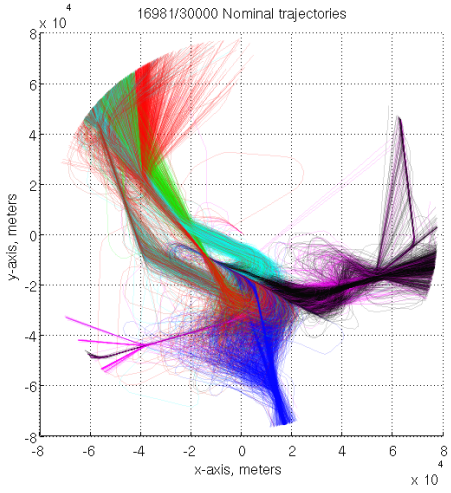


Figure 57: Results of trajectory clustering for 30,000 trajectories

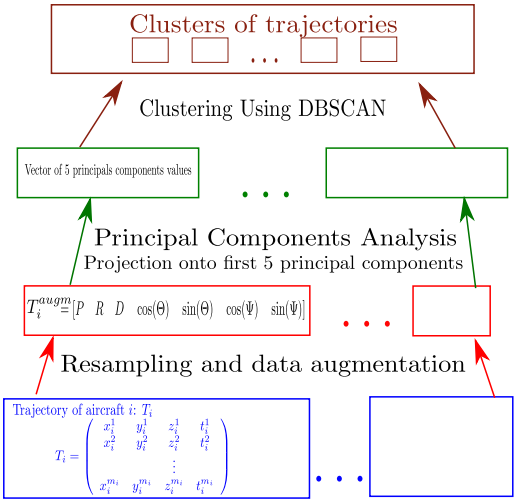


Figure 58: Trajectory clustering method based on Principal Components Analysis

off. The latter will be considered as an outlier even though it is very similar to the first trajectory. In addition, trajectories containing large rerouting periods will belong to the clusters as long as they pass over way-points.

A.5 Trajectory-based clustering via principal components analysis

This method proceeds with the following steps, which are summarized in a diagram in figure 58:

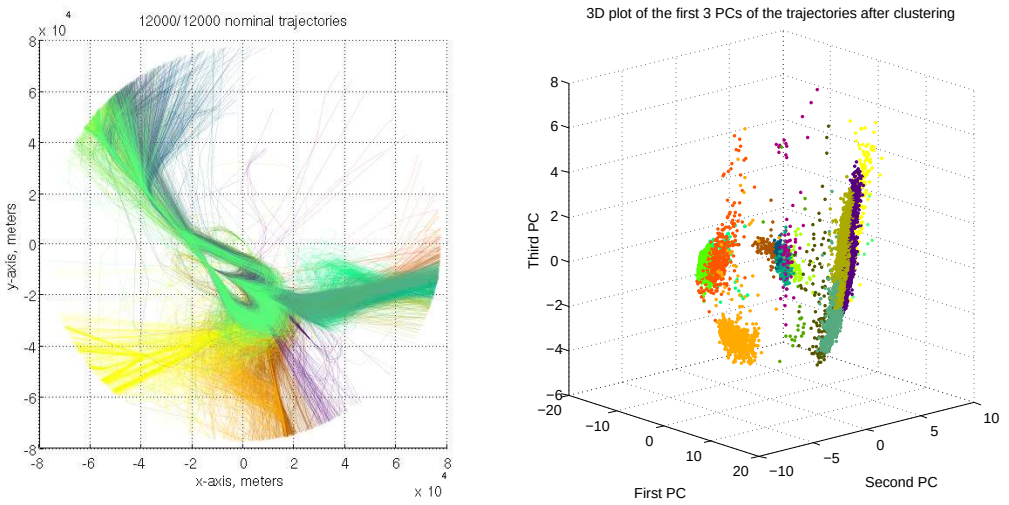
1. Re-sample the trajectories, to obtain time series of equal length for each aircraft.

2. Augment the dimensionality of the data. Normalize and concatenate all the data into a single vector for each flight.

3. Run a principal components analysis (PCA) and keep the first 5 principal components (PCs).

4. Cluster using a density-based clustering algorithm.

This appendix proposes improvements to the approach used by Eckstein [26] to realize a trajectory taxonomy. In [26], trajectories are first re-sampled, then the principal components are extracted and finally, the clustering is realized using k-means on the projections onto the first two principal components. Figure 59(a) presents the resulting clusters on the



(a) Clusters of trajectories

(b) Clusters of the first principal components

Figure 59: Clustering results using the method presented in [26]

principal components and on the trajectories, using the methods introduced in [26]. The clustering technique proposed in [26] does not provide results precise enough for our data set and there is no identification of outliers. Figure 59(b) presents a 3D view of the projection onto the first three PCs (section A.5.3) so it can be compared with our method. Eckstein used only the first two PCs for clustering. The first improvement is to augment the dimensionality of the data. Then, the PCs are computed and the projections of the augmented trajectories onto the first five PCs are clustered using a density based clustering algorithm. This algorithm presents the advantage of identifying outliers. Another advantage is that the number of clusters is not set a priori.

A.5.1 Trajectory resampling

The dataset is well organized and clean. Trajectories with fewer than 50 points are removed from the dataset: to be able to use a clustering algorithm such as DBSCAN, each trajectory must be represented as a vector. All vectors must have the same number of elements n, so their distance can be computed. Since all trajectories do not have the same number of points, re-sampling is necessary. Trajectories are resampled so that the total number of points for each trajectory is 50. For the sole purpose of clustering, fewer than 50 points would have been enough. Nevertheless, to improve the accuracy of the airspace monitoring

function presented in section 5.3, 50 points were used. The re-sampled trajectory T_i' is given by $T_i^{samp} = \left\{T_i^l, l = \left\{\text{round}(\frac{k\ n_i}{50}), k = 1 \dots 50\right\}\right\}$. During this operation, the notion of speed that was given by the distance between the radar echoes is lost. For example, consider the trajectories of two aircraft with the exact same flight path, but one going twice as fast as the other. After re-sampling, the trajectories will have the exact same points and it will be impossible to determine that there was a speed difference.

A.5.2 Dimensionality augmentation

To improve the results of the clustering, the dimensionality of the data was increased. Some of the added dimensions present symmetry with respect to a point or a line and some do not.

- Cartesian position of the aircraft in the re-sampled trajectory:
  $P = [x_i^1 \dots x_i^{50} y_i^1 \dots y_i^{50} z_i^1 \dots z_i^{50}]$. P is a row vector with 150 components. This vector is unique to each trajectory.

- Distance from the center of the TRACON
  $R = \left\{r_i^l = \sqrt{(x_i^l)^2 + (y_i^l)^2 + (z_i^l)^2}, l = 1 \dots 50\right\}$. Provides information about the rate of convergence of the aircraft toward the center of the TRACON, which is located close to the airport. This distance presents a symmetry with respect to the center of the TRACON, i.e two trajectories that are symmetric with respect to the center of the TRACON will be represented with the same vector R.

- Distance from the top left corner:
  $D = \left\{d_i^l = \sqrt{(x_i^l - x_{ref})^2 + (y_i^l - y_{ref})^2 + (z_i^l)^2}, l = 1 \dots 50\right\}$, where (x_{ref}, y_{ref}) are the coordinates of the top left corner of a square containing the TRACON. The top left corner has coordinates $(x_{ref}, y_{ref}) = (-80, 80)km$. This distance presents a symmetry with respect of the diagonal joining the top left corner $(-80, 80)$ and the bottom right corner $(80, -80)$, i.e two trajectories that are symmetric with respect to this diagonal will be represented with the same vector D.

- Angular position in cylindrical coordinates: $\Theta = \left\{\theta_i^l = \arctan(\frac{y_i^l}{x_i^l}), l = 1 \dots 50\right\}$. With

only one dimension, the angular position provides information about the overall location of the trajectory in the TRACON, i.e in which quadrant of the circle the trajectory lies. This information does not present any symmetry.

- Heading of the aircraft $\Psi = \left\{\psi_i^l, l = 1 \dots 50\right\}$. The computation of the heading was done using the filter of equation 74 and then re-sampled to 50 points. A constant value or a slow rate of change indicate a straight trajectory, while a high variability indicates a curved trajectory. This vector is unique to each trajectory.

The sine and cosine values of the angular position and heading are used instead of their actual value to avoid the discontinuity at 2π. Each augmented trajectory is now represented by a vector of dimension 450 given by:
$T_i^{augm} = [P \quad R \quad D \quad \cos(\Theta) \quad \sin(\Theta) \quad \cos(\Psi) \quad \sin(\Psi)]$. The initial vector had dimension 150. The values of each parameter are normalized between 0 and 1 in order to balance their importance during the clustering process. It was decided to add meaningful data such as heading or rate of convergence toward the center of the TRACON. For instance, two aircraft on parallel trajectories will fly the same heading, even if the trajectories are slightly apart from each other. Distance to the center was chosen to identify trajectories that have particular patterns such as vectoring and holding pattern: the distance to the center will present some irregularities as the aircraft flies back and forth. Such irregularities will be highlighted by dimensions such as the heading that will change 180° while the position will only change slightly.

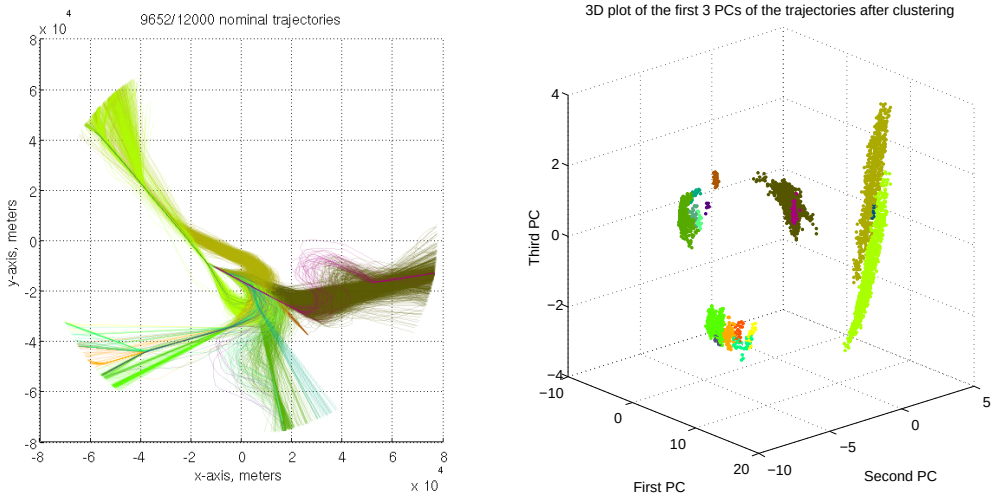
A.5.3 Principal components analysis

A principal components analysis [129] is run on matrix that contains all the re-sampled trajectories. Each trajectory is then projected onto the first p principal components and is now represented by a vector of p values. The choice in the value of p is a trade-off between computational speed when p is small and accuracy when p gets larger. There is no need to get a value of p too large since the first principal components contain most of the information. Different values of p were tried and $p = 5$ gave a satisfactory level of accuracy for this type of data. The added dimensions increase the range of the projection of the

trajectories onto the principal components. This makes the clustering task easier as the clusters are "further apart" in the principal components space.

A.5.4 Clustering

The projections of the trajectories onto the first 5 PCs are clustered using DBSCAN. A density based clustering algorithm like DBSCAN is preferred to a distance based algorithm because of the shape of the clusters can be arbitrary. The other advantage of DBSCAN is the identification of outliers. Figure 60(b) presents the resulting clusters. The axes correspond to the values of the first 3 principal components. Clusters are clearly differentiated, even if they are not easy to distinguish on the plot due to the perspective effect. The resulting clusters of trajectories are visually very clean (Figure 60(a)). Figure 61 presents



(a) Clusters of trajectories

(b) Clusters of the first 5 principal components

Figure 60: Clustering results using re-sampling, data augmentation, PCA decomposition, and DBSCAN on the first 5 principal components.

the centroids, that is the center of mass of the trajectories of each cluster. Those centroids can be seen as "typical operations". Some clusters are minor variations from each other, such as the flights coming from the bottom left corner. This comes from the settings used for DBSCAN. On Figure 60(b), one can clearly identify clusters of points. The algorithm was run with a high sensitivity (ϵ small and $minPts$ large). The parameter ϵ reflects the similarity between trajectories (the smaller the more similar), and $minPts$ is the number of

"similar" trajectories needed to create a new cluster (Section A.4.2.2). A small ϵ generates "narrow" cluster while a larger ϵ will generate clusters with more variability in trajectories. In the application presented in section 5.3, the algorithm is run with a lower sensitivity and provides fewer clusters, with larger variability. The resulting centroids of this run can be seen on Figure 49.

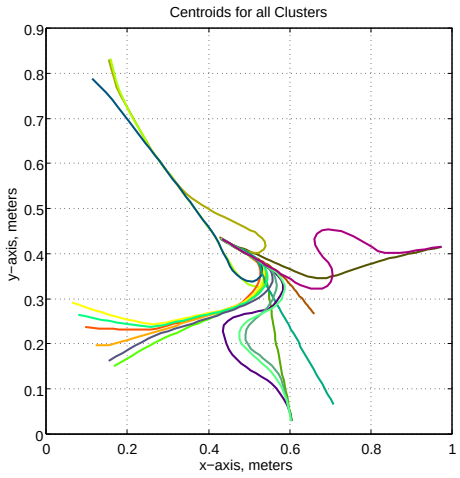


Figure 61: Clusters centroids (average trajectory)

A.5.5 Analysis of outliers

Figure 62 shows the outliers detected by the clustering algorithm. Outliers represent 19.5% of all trajectories. A visual inspection shows that the main reasons for being detected as an outlier is the presence of holding patterns, large vectoring maneuvers or direct routes.

Figure 63 presents the number and frequency of outlier trajectories as a function of the type of aircraft. Commercial jets represent the largest share in numbers, but the frequency is much smaller. Among the trajectories of regional and business aircraft, 4% and 5% are identified as outliers, respectively. A possible explanation is the size, the speed and the maneuverability of the aircraft. To ensure a safe separation at the runway threshold, air traffic controllers "vector" aircraft, that is give a sequence of headings to follow. The vectors given to business and regional aircraft might be different and sharper than the vectors given to larger size jets.

Figure 64 presents the frequency of outliers for each day of study. Each bar represents

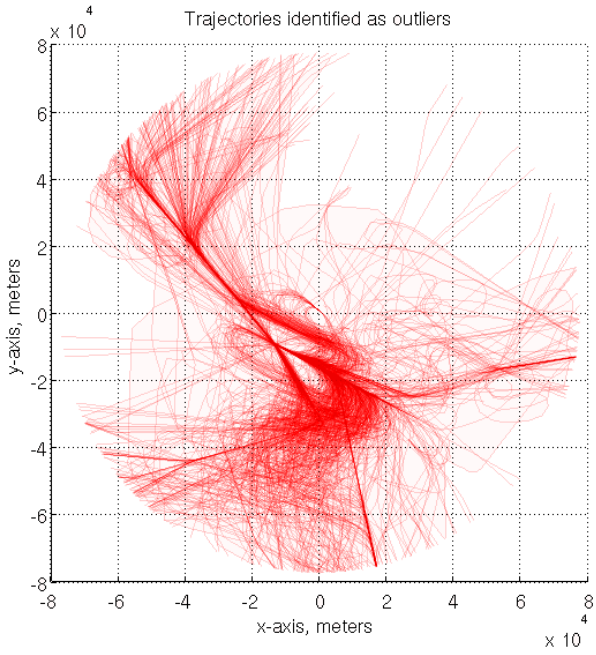


Figure 62: Trajectories identified as outliers

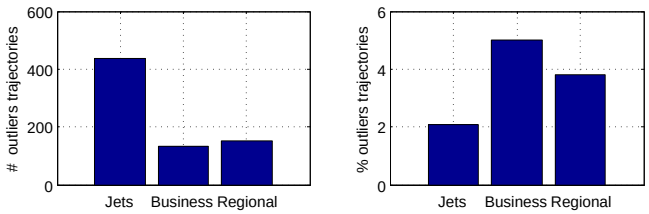


Figure 63: Distribution of outliers by aircraft category

one day. The minimum percentage of outliers is less than 2% and goes up to 16%. The most likely explanation for the outliers is the weather. San Francisco airport usually operates with two close parallel runways. The runways are not independent, that is, they cannot be operated simultaneously when the weather does not permit visual approaches. When a runway is closed, the landing capacity is reduced from 60 to 30 aircraft per hour. Schedules and operations usually take the weather into account , but unexpected late fog dissipation or other type of convective weather might disrupt the operations and force controllers to vector aircraft and put them on holding patterns.

Figure 65 presents the frequency of outliers as a function of the time of the day. The local time is reported on the abscissa axis, starting at midnight. This diagram is an average

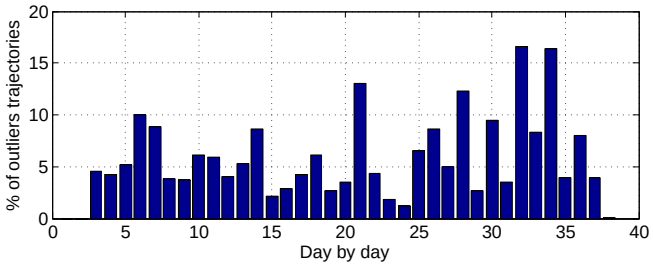


Figure 64: Histogram of outliers, day by day

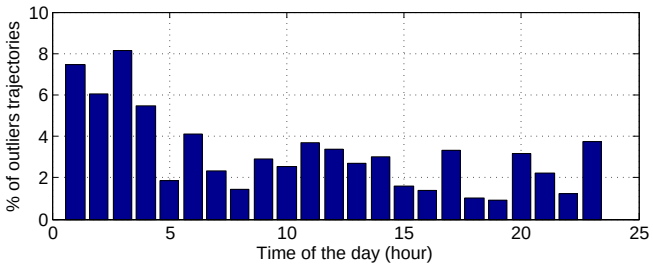


Figure 65: Histogram of outliers, hour by hour, local time

over the entire period of interest. The frequency of outliers is higher during the period 12 a.m. - 4 a.m., then decreases in the early morning, to an increase again with a peak at 11 a.m.. Another peak is visible at 5 p.m.. The outliers identified at night are mostly due to direct routing that is allowed by the very low traffic density at night. During the morning, traffic density increases and requires more rerouting for efficient sequencing and merging. Another possible explanation is the late dissipation of the fog.

APPENDIX B

SELECTED SESAR AND NEXTGEN RESEARCH ISSUES

Table 6: NextGen safety oriented research topics [70]

Ref	Line Reference	Issue
R-6	In addition, backup functions are distributed throughout the system, and there are layers of protection to allow for graceful degradation of services in the event of automation failures.	**Develop guidance for what flexibility is allowed** in the implementation of **airborne separation management** algorithms to ensure operationally consistent results, **understanding whether variations in algorithms can result in major impacts on overall operations**.
R-9	C-ATM is the means by which flight operator objectives are balanced with overall NAS performance objectives and accomplishes many of the objectives for CM, FCM, and TM.	Super Density operations will result in many aircraft in close proximity. Consequently **an aircraft deviating from its assigned trajectory is much more likely to cause an immediate conflict with another aircraft, and safe avoidance maneuvers may be limited or unavailable. How can super density operations be conducted safely, especially in the presence of severe weather?**
R-10	C-ATM is the means by which flight operator objectives are balanced with overall NAS performance objectives and accomplishes many of the objectives for CM, FCM, and TM.	**Develop requirements for a collision avoidance system that is compatible with NextGen tactical separation?** Unless mandated otherwise, some aircraft will likely be equipped with legacy TCAS/ACAS systems which may generate unwanted alerts during normal operations. How should this be accounted for?
R-28	In all, these new kinds of flight operations **dramatically improve en route productivity and capacity** and are essential to achieving NextGen.	**If the automation fails, what is the backup plan in terms of people/procedures/automation?**
R-44	As illustrated in Figure 24, super-density corridors handle arriving and departing traffic, while much nearby airspace remains available to other traffic.	**We do not prove today's ATC system is safe, but rely on historical data. NextGen will be required to both be safe and to demonstrate it is safe. How will safety be designed into all aspects of NextGen and then proven?**
P-1	In addition, backup functions are distributed throughout the system, and there are layers of protection to allow for graceful degradation of services in the event of automation failures.	**Develop policies concerning liability for delegated separation and self-separation operations.**

Table 7: SESAR safety oriented research topics [125]

No	Research topic
15	Study of the following automation topics: Automated separation tools and safety, impact of automation on capacity and impact of loss of situation awareness and tools to manage exceptions associated with loss of situation awareness.
16	Evaluation of ground based deconfliction automation support tools with particular focus on how to ensure feasible solutions with a minimum of constraints on the users trajectory.
41	Evaluation of terminal route structure design involving alternative arrival techniques with multiple or single merging points.
42	Evaluation of Time Based Separation (TBS) on merging points focusing on accuracy requirements and benefits.
49	Study on dynamic risk modeling and management techniques for on-line measurement of safety risk. Study on the assessment of the overall safety of the CONOPS. For now, it is not obvious that the concepts ideas all together are safe in principal (as statemorningbulletin2010d e.g. in Episode 3 objectives).
51	Model complex scenarios of new trajectory based arrival/departure techniques plus existing SID/STAR and also with the SID/STAR from nearby airports plus transit traffic
54	Study controller acceptability of ASAS Spacing versus ASAS Separation during the organization of streams of traffic
62	Development, evaluation and agreement on separation minima for each separation method included in the concept
64	The new separation modes described at least Dynamic Route Allocation, 4D Contracts and ASAS-Self Separation in mixed mode environment shall be assessed with regard to maturity and potential performance: New separation modes shall be assessed with regard to maturity and potential performance: The robustness and stability of the various methods in the face of unexpected events (even of small magnitude) is to be investigated.
80	Elaboration of high density separation concepts and associated airspace issues in terms of detail procedures which should be then validated with a focus on feasibility.

REFERENCES

[1] 37000FEET.COM, NASA ASRS , "Aircraft equipment problem – both fmgc fail crossing restr not met.." `http://www.37000feet.com/report/310451`, last visited May 18, 2010.

[2] AIRLINE BIZ BLOG, "How the FAA scolded the pilots," October 2009. `http://aviationblog.dallasnews.com/archives/2009/10/how-the-faa-scolded-the-pilots.html`, last visited March 8, 2010.

[3] ANAGNOSTAKIS, *A Multi-Objective, Decomposition-Based Design Methodology and its Application to Runway Operations Planning*. PhD thesis, Massachusetts Institute of Technology, September 2004.

[4] ARAIN, S. and SHAIKH, A., "Software failure air traffic control system," October 2009. `http://www.slideshare.net/shakeeiil/software-failure-air-traffic-control-system`, last visited March 8, 2010.

[5] AVIATION, M., "Flight schedule monitor." `http://www.metronaviation.com/solutions/traffic-flow-management/systems/fsm.html`, last visited May 18, 2010.

[6] BALL, M. and LULLI, G., "Ground delay programs: Optimizing over the included flight set based on distance," *Air Traffic Control Quarterly*, vol. 12, no. 1, pp. 1–25, 2004.

[7] BBC, "Radar "glitch" restricts flights in parts of the uk," March 15 2010. `http://news.bbc.co.uk/2/hi/uk_news/8569322.stm`, last visited May 18, 2010.

[8] BILIMORIA, K. and JASTRZEBSKI, M., "Aircraft clustering based on airspace complexity," in *7th AIAA Aviation Technology, Integration and Operations Conference (ATIO)*, 18 - 20 September 2007.

[9] BILLINGS, C., "Toward a Human-Centered Aircraft Automation Philosophy," *The International Journal of Aviation Psychology*, vol. 1, no. 4, pp. 261–270, 1991.

[10] BLIN, K., BONNANS, F., HOFFMAN, E., and ZEGHAL, K., "Conflict resolution in presence of uncertainty: A case study of decision making with dynamic programming," in *AIAA Guidance, Navigation, and Control Conference and Exhibit, Montreal, August*, 2001.

[11] BRUDNICKI, D. and MCFARLAND, A., "User Request Evaluation Tool (URET) Conflict Probe Performance and Benefits Assessment," *MITRE CAASD, MP*, vol. 2147483647, 1997.

[12] BUDALAKOTI, S., SRIVASTAVA, A., AKELLA, R., and TURKOV, E., "Anomaly detection in large sets of high-dimensional symbol sequences," tech. rep., NASA Ames Research Center, 2006.

[13] Budalakoti, S., Srivastava, A., and Otey, M., "Anomaly Detection and Diagnosis Algorithms for Discrete Symbol Sequences with Applications to Airline Safety," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 39, no. 1, pp. 101–113, 2009.

[14] Chent, H. and Zhao, Y., "A new queueing model for aircraft landing process," in *AIAA Guidance, Navigation, and Control Conference and Exhibit, New Orleans, LA*, 1997.

[15] CNN, "Atlanta airport reopens after lightning strike," April 2009. `http://edition.cnn.com/2009/US/04/23/ga.airport.storms/index.html`, last visited March 8, 2010.

[16] Daams, J., Blom, H., and Nijhuis, H., "Modelling human reliability in air traffic management," *FRONTIERS SCIENCE SERIES*, vol. 2, pp. 1193–1200, 2000.

[17] Dahlbom, A. and Niklasson, L., "Trajectory clustering for coastal surveillance," in *Proceedings of the 10th International Conference on Information Fusion*, 2007.

[18] Davis, T., "Multi-Center Traffic Management Advisor," 2002.

[19] Davis, T., Erzberger, H., and Bergeron, H., "Design of a final approach spacing tool for TRACON air traffic control," 1989.

[20] Davis, T., Krzeczowski, K., and Bergh, C., "The final approach spacing tool," in *Proceedings of the 13th IFAC Symposium on Automatic Control in Aerospace*, pp. 70–76, 1994.

[21] Delahaye, D. and Puechmorel, S., "Air traffic complexity map based on non linear dynamical systems," in *INO Workshop*, 2005.

[22] Denery, D. and Erzberger, H., "The center-TRACON automation system: Simulation and field testing," *NASA Technical Memorandum*, vol. 110366, 1995.

[23] DesertNews.com, "S.l. testing air-traffic update," October 2009. `http://www.deseretnews.com/article/705335185/SL-testing-air-traffic-update.html`, last visited March 8, 2010.

[24] Di Benedetto, M., Di Gennaro, S., and D'Innocenzo, A., "Error detection within a specific time horizon and application to air traffic management," in *44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC'05*, pp. 7472–7477, 2005.

[25] Dodder, R., Sussman, J., and McConnell, J., "The Concept of the "CLIOS Process": Integrating the Study of Physical and Policy Systems Using Mexico City as an Example," in *Massachusetts Institute of Technology Engineering Systems Symposium, Cambridge, MA*, vol. 31, Citeseer, 2004.

[26] Eckstein, A., "Automated flight track taxonomy for measuring benefits from performance based navigation," in *Integrated Communications, Navigation and Surveillance Conference*, 2009.

[27] Encyclopedia.com, "Air traffic control failure is examined," October 2007. http://www.encyclopedia.com/doc/1Y1-111174919.html, last visited March 8, 2010.

[28] Endsley, M. and Garland, D., *Situation awareness: analysis and measurement*. CRC Press, 2000.

[29] Erzberger, H., Davis, T., and Green, S., "Design of center-TRACON automation system," *In AGARD, Machine Intelligence in Air Traffic Management 12 p(SEE N 94-29558 08-04)*, vol. 1993, 1993.

[30] Erzberger, H., Paielli, R., Isaacson, D., and Eshow, M., "Conflict detection and resolution in the presence of prediction error," in *1st USA/Europe Air Traffic Management Research & Developpment Seminar*, 1997.

[31] Ester, M., Kriegel, H., Sander, J., and Xu, X., "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. 2nd Int. Conf. on Knowledge Discovery and Data Mining, Portland, OR, AAAI Press*, pp. 226–231, 1996.

[32] Eurocontrol: European Organisation for the Safety of Air Navigation, "SESAR consortium." http://www.eurocontrol.int/sesar/public/subsite_homepage/homepage.html, last visited March 8, 2010.

[33] Eurocontrol, iFly, "Airborne self-separation at innovation workshop." http://www.eurocontrol.int/eec/public/standard_page/ETN_2010_1_IFLY.html, last visited May 18, 2010.

[34] Farley, Landry, Nickelson, Levin, Rowe, and Welch, "Multi-Center Traffic Management Advisor: Operational Test Results," in *AIAA 5th Aviation, Technology, Integration, and Operations Conference (ATIO)*, 2005.

[35] Federal Aviation Administration, *Aeronautical Information Manual*, 2009.

[36] Federal Aviation Administration, *Federal Aviation Regulation, Part 25*, 2009.

[37] Federal Aviation Administration, "Administrator's fact book," tech. rep., December 2006.

[38] Federal Aviation Administration, "An analysis of en route air traffic control system usage during special situations," tech. rep., May 2006.

[39] Federal Aviation Administration - Eurocontrol, "Safety techniques and toolbox," tech. rep., Version 2.0, October 3, 2007.

[40] Feearal Aviation Administration, "TRACON facilities." http://www.faa.gov/about/office_org/headquarters_offices/ato/tracon/, last visited March 8, 2010.

[41] FlightAware, "Track log." http://flightaware.com, last visited March 8, 2010.

[42] Fourer, R., Gay, D., and Kernighan, B., *AMPL: A Modeling Language for Mathematical Programming*. The Scientific Press Series, 1993.

[43] F.S. Hillier and G.J. Lieberman, *Introduction to Operations Research*. seventh ed., 2001.

[44] Gariel, M., Clarke, J.-P., and Feron, E., "A dynamic I/O model for TRACON traffic management," in *AIAA Guidance, Navigation, and Control Conference and Exhibit, Hilton Head, SC*, 2007.

[45] Gariel, M. and Feron, E., "Graceful Degradation of Air Traffic Operations: Airspace Sensitivity to Degraded Surveillance Systems," *Proceedings of the IEEE*, vol. 96, no. 12, pp. 2028–2039, 2008.

[46] Gariel, M. and Feron, E., "3D Conflict Avoidance under Uncertainties," in *28th IEEE/AIAA Digital Avionics Systems Conference*, pp. 4.E.3–1 – 4.E.3–8, October 23-28, 2009, Orlando, Florida.

[47] Gariel, M., Feron, E., and Clarke, J., "Air Traffic Management complexity maps induced by degradation of Communication, Navigation and Surveillance.," in *AIAA Guidance, Navigation and Control Conference and Exhibit*, August 18 - 21, 2008, Honolulu, Hawaii.

[48] Gariel, M., Srivastava, A. N., and Feron, E., "Trajectory clustering and an application to airspace monitoring," *Submitted to IEEE Transactions on Intelligent Transportation*, 2010.

[49] Green, S., Vivona, R., and Sanford, B., "Descent Advisor preliminary field test," 1995.

[50] Green, S. and Vivona, R., "Field evaluation of Descent Advisor trajectory prediction accuracy," in *Proceedings of the AIAA Guidance Navigation and Control Conference*, Citeseer, 1996.

[51] Guardian.co.uk, "Flight delays continue after air traffic control failure," September 2008. `http://www.guardian.co.uk/uk/2008/sep/26/transport.theairlineindustry`, last visited March 8, 2010.

[52] Haasl, D., Roberts, N., Vesely, W., and Goldberg, F., "Fault tree handbook," tech. rep., NUREG-0492, Nuclear Regulatory Commission, Washington, DC (USA). Office of Nuclear Regulatory Research, 1981.

[53] Han, J. and Kamber, M., *Data mining: concepts and techniques*. Morgan Kaufmann, 2006.

[54] Hastie, T., Tibshirani, R., and Friedman, J., *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, 2nd ed., 2009.

[55] Herald.ie, "Failure of radar causes chaos at dublin airport," July 10 2008. `http://www.herald.ie/national-news/failure-of-radar-causes-chaos-at-dublin-airport-1430658.html`, last visited May 19, 2010.

[56] Herlihy, M. and Wing, J., "Specifying graceful degradation in distributed systems," in *PODC '87: Proceedings of the sixth annual ACM Symposium on Principles of distributed computing*, (New York, NY, USA), pp. 167–177, ACM Press, 1987.

[57] Highbeam.com, "ATC Zero: even with backup systems and master plans, a catastrophic failure may still have critical air traffic control being done by cell phone.(SYSTEM NOTES)," December 2007. http://www.highbeam.com/doc/1G1-203027937.html, last visited March 8, 2010.

[58] Histon, J., Hansman, R., Aigoin, G., Delahaye, D., and Puechmorel, S., "Introducing structural considerations into complexity metrics," *Air Traffic Control Quarterly*, 2002.

[59] Histon, J., Hansman, R., Gottlieb, D., Kleinwaks, H., Yenson, S., Delahaye, D., and Puechmorel, S., "Structural considerations and cognitive complexity in air traffic control," *21st IEEE/AIAA Digital Avionics Systems Conference*, 2002.

[60] Histon, J. M. and Hansman, R. J., "The impact of structure on cognitive complexity in air traffic control," tech. rep., MIT International Center for Air Transportation, June 2002.

[61] Hu, J., Lygeros, J., Prandini, M., and Sastry, S., "Aircraft conflict prediction and resolution using Brownian Motion," in *IEEE Conference on Decision and Control (CDC)*, vol. 3, pp. 2438–2443, Citeseer, 1999.

[62] Idris, H., Clarke, J.-P., Bhuva, R., and Kang, L., "Queuing model for taxi-out time estimation," *Air Traffic Control Quarterly*, 2001.

[63] Idris, H. and Evans, A., "Benefits assessment of Multi-Center Traffic Management Advisor for Philadelphia and New York," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, 2003.

[64] ILOG, *CPLEX User's guide*, 1999.

[65] International Civil Aviation Organization (ICAO), *PANS - ATM doc 4444*, 14th edition, 2001.

[66] International Civil Aviation Organization (ICAO), "ADS-B separation standards under development in the ICAO separation and airspace safety panel (SASP)," tech. rep., 2003.

[67] Isaacson, D., Erzberger, H., Center, N., and Field, M., "Design of a conflict detection algorithm for the Center/TRACON automation system," in *16th IEEE/AIAA Digital Avionics Systems Conference*, vol. 2, 1997.

[68] Iverson, D. and Stop, M., "Inductive system health monitoring," in *Proceedings of The 2004 International Conference on Artificial Intelligence (IC-AI04)*, 2004.

[69] Joint Planning and Development Office, "NextGen." http://www.jpdo.gov/, last visited March 8, 2010.

[70] Joint Planning and Development Office, "Concept of Operations for the Next Generation Air Transportation System," tech. rep., JPDO, June 2007.

[71] Kim, S., Feigh, K., Lee, S., and Johnson, E. N., "A Task Decomposition Method for Function Allocation," in *AIAA Infotech@Aerospace Conference*, April 6 - 9, 2009, Seattle, Washington.

[72] Kopardekar, P., Bilimoria, K., and Sridhar, B., "Airspace configuration concepts for the next generation air transportation system," *Air Traffic Control Quarterly*, vol. 16(4), pp. 313–336, 2008.

[73] Krozel, J., "Intelligent tracking of aircraft in the National Airspace System," in *AIAA Guidance, Navigation, and Control Conference and Exhibit, Monterey, CA*, 2002.

[74] Kuchar, J. K. and Yang, L. C., "A review of conflict detection and resolution modeling methods," *IEEE Transactions on Intelligent Transportation Systems*, vol. 1(4), pp. 179–189., 2000.

[75] Kuchar, J. and Yang, L., "Survey of conflict detection and resolution modeling methods," in *AIAA Guidance, Navigation, and Control Conference, New Orleans, LA*, pp. 1388–1397, 1997.

[76] Lee, J., Han, J., and Li, X., "Trajectory outlier detection: A partition-and-detect framework," in *Proc. 24th Intl Conf. on Data Engineering*, pp. 140–149, 2008.

[77] Lee, J., Han, J., and Whang, K., "Trajectory clustering: A partition-and-group framework," in *Proceedings of the 2007 ACM SIGMOD international conference on Management of data*, pp. 593–604, ACM New York, NY, USA, 2007.

[78] Lee, K., Feron, E., and Pritchett, A., "Describing airspace complexity: Airspace response to disturbances," *Journal of Guidance, Control, and Dynamics*, vol. 32, no. 1, 2009.

[79] Lee, K., Quinn, C., Hoang, T., and Sanford, B., "Human Factors Report: TMA Operational Evaluations 1996 & 1998," *Moffett Field, CA, NASA Ames Research Center*, 2000.

[80] Lee, S. M., Kim, S., Feigh, K., and Volovoi, V., "Structural framework for performance-based assessment of atm systems," in *9th AIAA Aviation Technology, Integration, and Operations Conference (ATIO)*, September 21 - 23, 2009, Hilton Head, South Carolina.

[81] Leveson, N., Cha, S., and Shimeall, T., "Safety verification of ADA programs using software fault trees," *IEEE software*, vol. 8, no. 4, pp. 48–59, 1991.

[82] Lloyd, S., "Least squares quantization in PCM," *IEEE Transactions on Information Theory*, vol. 28, no. 2, pp. 129–137, 1982.

[83] Lygeros, J., Pappas, G., and Sastry, S., "An approach to the verification of the Center-TRACON automation system," *Lecture notes in computer science*, vol. 1386/1998, pp. 289–304, 1998.

[84] Lygeros, J. and Prandini, M., "Aircraft and weather models for probabilistic collision avoidance in air traffic control," in *IEEE Conference on Decision and Control*, vol. 3, pp. 2427–2432, Citeseer, 2002.

[85] M. Gariel, A. V. and Feron, E., "Identification of Air-Routes, Flows and Flow Corridors using ETMS Data for Air Traffic Management," in *AIAA 10th Aviation,*

Technology, Integration, and Operations Conference(ATIO), Fort Worth, TX, October 2010.

[86] MacKay, D., *Information theory, inference, and learning algorithms*. Cambridge Univ Pr, 2003.

[87] MacQueen, J. and others, "Some methods for classification and analysis of multivariate observations," 1966.

[88] Mahaney, S. and Schneider, F., "Inexact agreement: Accuracy, precision, and graceful degradation," in *Proceedings of the fourth annual ACM symposium on Principles of distributed computing*, pp. 237–249, ACM New York, NY, USA, 1985.

[89] Mao, Z., Dugail, D., Feron, E., and Bilimoria, K., "Stability of intersecting aircraft flows using heading-change maneuvers for conflict avoidance," in *IEEE Transactions On Intelligent Transportation Systems*, December 2005.

[90] Mao, Z., Feron, E., and Bilimoria, K., "Stability and performance of intersecting aircraft flows under decentralized conflict avoidance rules," in *IEEE Transactions On Intelligent Transportation Systems*, June 2001.

[91] Memphis Business Journal, "Telecom glitch stops departures at memphis international," September 2007. `http://memphis.bizjournals.com/memphis/stories/2007/09/24/daily12.html`, last visited March 8, 2010.

[92] Mogford, R., "Mental models and situation awareness in air traffic control," *The International Journal of Aviation Psychology*, vol. 7, no. 4, pp. 331–341, 1997.

[93] Mogford, R., Guttman, J., Morrow, S., and Kopardekar, P., "The complexity construct in air traffic control: A review and synthesis of the literature (DOT/FAA/-CT TN95/22)," *Atlantic City, NJ: FAA*, 1995.

[94] myfoxphoenix.com, "Us airways plane leaving phx struck by lightning," January 27 2010. `http://www.myfoxphoenix.com/dpp/news/us-airways-plane-struck-by-lightning-1-27-2010`, last visited February 17, 2010.

[95] NASA, "Center TRACON Automation System." `http://www.ctas.arc.nasa.gov/project_description/index.html#overview`, last visited March 8, 2010.

[96] NASA, "Center TRACON Automation System, Multi-Center Traffic Management Advisor." `http://www.ctas.arc.nasa.gov/project_description/mctma.html`, last visited March 8, 2010.

[97] NASA, "Center TRACON Automation System, Traffic Management Advisor." `http://www.ctas.arc.nasa.gov/project_description/tma.html`, last visited March 8, 2010.

[98] Nedell, W., Erzberger, H., and Neuman, F., "The Traffic Management Advisor," in *1990 American Control Conference, 9 th, San Diego, CA*, pp. 514–520, 1990.

[99] News, Y., "Air traffic control failure is examined," October 2007. http://www.newsmgr.com/nm2/uploads/101107_ap_air_traffic_control_failure_is_examined.pdf, last visited March 8, 2010.

[100] newsletter.co.uk, "Dublin airport radar failure grounds travellers," July 10 2008. http://www.newsletter.co.uk/news/Radar-failure-grounds-travellers.4273613.jp, last visited February 17, 2010.

[101] news.scotsman.com, "Doomed beirut flight 'blown to pieces by lightning'," January 26 2010. http://news.scotsman.com/world/Doomed-Beirut-flight-39blown-to.6012510.jp, last visited February 17, 2010.

[102] Niessen, C., Eyferth, K., and Bierwagen, T., "Modelling cognitive processes of experienced air traffic controllers," *Ergonomics*, vol. 42, no. 11, pp. 1507–1520, 1999.

[103] Pallottino, L., Feron, E., and Bicchi, A., "Conflict resolution problems for air traffic management systems solved with mixed integer programming," *IEEE Transactions On Intelligent Transportation Systems*, vol. 3, no. 1, pp. 3–11, 2002.

[104] Phillips, W. E., *Mechanics of Flight*. Wiley, 2nd ed., 2009.

[105] Philstar.com, "Flights delayed by radar failure at naia," September 13 2009. http://www.philstar.com/article.aspx?articleid=505072&publicationsubcategoryid=200, last visited May 18, 2010.

[106] Piciarelli, C., Foresti, G., and Snidara, L., "Trajectory clustering and its applications for video surveillance," in *IEEE Conference on Advanced Video and Signal Based Surveillance, 2005. AVSS 2005*, pp. 40–45, 2005.

[107] Pinker, A. and Smith, C., "Vulnerability of the GPS Signal to Jamming," *GPS Solutions*, vol. 3, no. 2, pp. 19–27, 1999.

[108] Pinon, O. J., Fry, K., and Clarke, J.-P., "The Air Transportation System as a Supply Chain," in *AIAA Guidance, Navigation, and Control Conference and Exhibits, Chicago, Illinois*, August 10 - 13 2009.

[109] Pinon, O. J., Garcia, E., and Mavris, D., "A methodological approach for airport technology evaluation and selection," in *26th International Congres of the Aeronautical Sciences*, September 14 - 19, 2008, Anchorage, Alaska.

[110] Powell, J., Jennings, C., and Holforty, W., "Use of ADS-B and perspective displays to enhance airport capacity," in *24th IEEE/AIAA Digital Avionics Systems Conference*, November 2005.

[111] pprune.org, "Southern California TRACON (SCT) Evacuation," October 2003. http://www.pprune.org/atc-issues/106897-southern-california-tracon-sct-evac.html, last visited March 8, 2010.

[112] Prandini, M. and Watkins, O., "Probabilistic aircraft conflict detection," *HYBRIDGE D*, vol. 2, 2005.

[113] Prevot, T., Palmer, E., Smith, N., and Callantine, T., "A multi-fidelity simulation environment for human-in-the-loop studies of distributed air ground traffic management," in *American Institute of Aeronautics and Astronautics Modeling and Simulation Conference and Exhibit*, August 5 - 8, 2002, Monterey, CA.

[114] Pritchett, A., Carpenter, B., Asari, K., Kuchar, J., and Hansman, R., "Issues in airborne systems for closely-spaced parallel runway operations," in *14th IEEE/AIAA Digital Avionics Systems Conference*, 1995.

[115] Pritchett, A. and Hansman, R., "Pilot non-conformance to alerting system commands during closely spaced parallel approaches," in *16th IEEE/AIAA Digital Avionics Systems Conference*, 1997.

[116] Pujet, N., Delcaire, B., and Feron, E., "Input-output modeling and control of the departure process of congested airports," in *AIAA Guidance, Navigation, and Control Conference and Exhibit, Portland, OR*, pp. 1835–1852, 1999.

[117] Ren, L., Chang, D., Solak, S., Clarke, J.-P., Barnes, E., and Johnson, E., "Simulating Convective Weather for Air Traffic Management Modeling," in *Proceedings of Winter Simulation Conference*, 2007, December 9-12, Washington, D.C.

[118] Reynolds, T. and Hansman, R., "Conformance monitoring approaches in current and future air traffic control environments," in *21st IEEE/AIAA Digital Avionics Systems Conference*, 2002.

[119] Reynolds, T., Histon, J., Davison, H., and Hansman, R., "Structure, Intent & Conformance Monitoring in ATC," *ATM Workshop*, 22-26 September 2002, Capri, Italy.

[120] Ritchey, T., "General Morphological Analysis," *A general method for non-quantified modeling, 16th EURO*, 1998.

[121] Salaun, E., Gariel, M., Vela, A., Feron, E., and Clarke, J.-P., "Statistical Proximity Maps based on Data-Driven Flow Modeling," in *AIAA infotechaerospace, Atlanta, GA*, 2010.

[122] Sarter, N., Woods, D., and Billings, C., "Automation surprises," *Handbook of Human Factors and Ergonomics*, vol. 2, pp. 1926–1943, 1997.

[123] SESAR Consortium, "SESAR Master Plan, deliverable 5," tech. rep., April 2008.

[124] SESAR Consortium, "SESAR Work Programme for 2008-2013, Deliverable 6," tech. rep., April 2008.

[125] SESAR Consortium, "SESAR concept of operation," tech. rep., December 2006.

[126] SESAR Consortium, "SESAR definition phase, deliverable 3 : The ATM target concept," tech. rep., September 2007.

[127] Shannon, C., "A mathematical theory of communication," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 5, no. 1, pp. 3–55, 2001.

[128] Sheridan, T., "Next Generation Air Transportation System: Human-Automation Interaction and Organizational Risks," in *Proceedings of the 2nd Symposium on Resilience Engineering*, November 8-10, 2006, Juan-les-Pins, France.

[129] Shlens, J., "A tutorial on principal component analysis," *Copy retrieved [10-27-2009] from: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.115.3503&rep=rep1&type=pdf*, 2009.

[130] Shorrock, S. and Kirwan, B., "Development and application of a human error identification tool for air traffic control," *Applied Ergonomics*, vol. 33, no. 4, pp. 319–336, 2002.

[131] Sridhar, B., Sheth, K., and Grabbe, S., "Airspace complexity and its application in air traffic management," in *Proceedings of the 2nd USA/EUROPE Air Traffic Management Research & Developpment Seminar SEMINAR*, December 1998.

[132] Sussman, J., *Perspectives on intelligent transportation systems (ITS)*. Springer, 2005.

[133] techworld.com, "Microsoft server crash nearly causes 800-plane pile-up," September 2004. http://news.techworld.com/operating-systems/2275/microsoft-server-crash-nearly-causes-800-plane-pile-up/, last visited March 8, 2010.

[134] The Sweep / North County Times, "Southern California TRACON Goes Mapless," May 2007. http://atcmuseum.wordpress.com/2007/05/25/southern-california-tracon-spends-a-mapless-hour/, last visited March 8, 2010.

[135] The Times of India, "Attempt to revive screen caused igi radar failure," January 16 2010. http://timesofindia.indiatimes.com/city/delhi/Attempt-to-revive-screen-caused-IGI-radar-failure/articleshow/5450187.cms, last visited May 18, 2010.

[136] themorningbulletin.com.au, "Rocky-bound plane hit by lightning," February 8 2010. http://www.themorningbulletin.com.au/story/2010/02/08/weve-just-been-hit-by-lightning, last visited February 17, 2010.

[137] TMCnet.com, "Report Warns Air Traffic Control System Vulnerable to Cyber Terrorism," May 2009. http://www.tmcnet.com/usubmit/2009/05/27/4199192.htm, last visited March 18, 2010.

[138] Tobias, L., Volckers, U., and Erzberger, H., "Controller evaluations of the descent advisor automation aid," in *AIAA Guidance, Navigation and Control Conference, Boston, MA*, pp. 1609–1618, 1989.

[139] US Department of Transportation and Federal Aviation Administration, "Introduction to TCAS II, version 7," 2000.

[140] Vandevenne, H. and Lippert, M., "Evaluation of runway-assignment and aircraft-sequencing algorithms in terminal area automation," *The Lincoln Laboratory Journal*, vol. 7, no. 2, pp. 215–238, 1994.

[141] Vlachos, M., Kollios, G., and Gunopulos, D., "Discovering similar multidimensional trajectories," in *Proceedings of the International Conference on Data Engineering*, pp. 673–684, IEEE Computer Society Press; 1998, 2002.

[142] Volovoi, V., Fraccone, G., Heddrick, M., Kelley, R., and Colon, A., "Agent-based simulation of off-nominal conditions during a spiral descent (nextgen vehicle nra)," in *9th AIAA Aviation Technology, Integration, and Operations Conference (ATIO)*, September 21 - 23, 2009, Hilton Head, South Carolina.

[143] Wikipedia, "2002 berlingen mid-air collision." `http://en.wikipedia.org/wiki/2002_œberlingen_mid-air_collision`, last visited March 8, 2010.

[144] Wikipedia, "Terminal Control Center." `http://en.wikipedia.org/wiki/Terminal_Control_Center`, last visited March 8, 2010.

[145] Wikipedia, "US Airways Flight 1549," January 15 2009. `http://en.wikipedia.org/wiki/US_Airways_Flight_1549`, last visited March 8, 2010.

[146] Wyle, H. and Burnett, G., "Some relationships between failure detection probability and computer system reliability," in *Proceedings of the November 14-16, 1967, fall joint computer conference*, pp. 745–756, ACM, 1967.

[147] Yousefi, A. and Donohue, G., "Temporal and spatial distribution of airspace complexity for air traffic controller workload-based sectorization," in *AIAA 4th Aviation Technology, Integration and Operations (ATIO) Forum*, 20 - 22 September 2004.