UNDERSTANDING CONFLICT-RESOLUTION TASKLOAD: IMPLEMENTING ADVISORY CONFLICT-DETECTION AND RESOLUTION ALGORITHMS IN AN AIRSPACE

A Thesis
Presented to
The Academic Faculty

by

Adan Ernesto Vela

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Mechanical Engineering

Georgia Institute of Technology
December 2011

UNDERSTANDING CONFLICT-RESOLUTION TASKLOAD: IMPLEMENTING ADVISORY CONFLICT-DETECTION AND RESOLUTION ALGORITHMS IN AN AIRSPACE

Approved by:

Prof. William Singhose,
Committee Chair
School of Mechanical Engineering
Georgia Institute of Technology

Prof. Karen Feigh
School of Aerospace Engineering
Georgia Institute of Technology

Prof. John-Paul Clarke
School of Aerospace Engineering
Georgia Institute of Technology

Prof. Wayne Book
School of Mechanical Engineering
Georgia Institute of Technology

Prof. Eric Feron
School of Aerospace Engineering
Georgia Institute of Technology

Prof. Kok-Meng Lee
School of Mechanical Engineering
Georgia Institute of Technology

Date Approved: December 2011

To my loving and supporting family.

ACKNOWLEDGEMENTS

It is impossible to find the perfect words to thank my advisors, family, colleagues and friends for supporting my dream of obtaining a Ph.D. in engineering. They have influenced me with their academic and life wisdom to make me a better engineer and person. In particular, I want to thank my advisors Dr. William Singhose, Dr. John-Paul Clarke and Dr. Eric Feron. In Dr. William Singhose, I found more than a mechanical engineering advisor. He introduced me to Georgia Tech as an undergraduate, and later welcomed me into his lab. He has always guided my Ph.D. studies with challenging and intriguing questions. Through Dr. John-Paul Clark and Dr. Eric Feron, I learned to appreciate the practical and the theoretical sides of air transportation research. Each has taught me to push the boundaries of research and to never be apprehensive about exploration. I would not have acquired the knowledge, nor the skills, to complete my Ph.D. studies without the wisdom and guidance of my advisors.

I also want to thank Dr. Karen Feigh. She taught me the importance of human-centric air traffic control. I must also thank Dr. Daniel Delahaye for inviting me to study air traffic control at ENAC. He always encouraged a fun and challenging discourse on air traffic control. For supporting me from my first days as a Ph.D. student until now, I would like to thank Dr. Senay Solak. He continues to inspire me with his work in Operations Research. He will remain a role model and a source of inspiration. I also thank Dr. Senay Solak for his unwavering encouragement.

In my mechanical engineering lab, I want to give a special thanks and acknowledgement to Dr. Khalid Sorensen and Dr. Joshua Vaughan, who gave me advice at every turn of my Mechanical Engineering Ph.D. process and beyond. For teaching me about the realities of aircraft and air traffic control, I want to acknowledge Jim Brooks and Gaurav Nagel.

Next, I want to acknowledge all my friends and co-workers who have shared and supported my struggles. A special mention goes to Dr. Kuemjin Lee, Dr. Maxime Gariel, and

Dr. Erwan Saluan, who have been comrades in research.

To conclude, I want to thank my family. They are the greatest support system in my life, starting with my parents who devoted their lives to inspire me and many young people to study math, science and engineering. I also thanks my brother and sister who have given me their love and support throughout my whole life.

SUMMARY

From 2010 to 2030, the number of instrument flight rules aircraft operations handled by Federal Aviation Administration en route traffic centers is predicted to increase from approximately 39 million flights to 64 million flights. The projected growth in air transportation demand is likely to result in traffic levels that exceed the abilities of the unaided air traffic controller in managing, separating, and providing services to aircraft. Consequently, the Federal Aviation Administration, and other air navigation service providers around the world, are making several efforts to improve the capacity and throughput of existing airspaces. Ultimately, the stated goal of the Federal Aviation Administration is to triple the available capacity of the National Airspace System by 2025.

In an effort to satisfy air traffic demand through the increase of airspace capacity, air navigation service providers are considering the inclusion of advisory conflict-detection and resolution systems. In a human-in-the-loop framework, advisory conflict-detection and resolution decision-support tools identify potential conflicts and propose resolution commands for the air traffic controller to verify and issue to aircraft. A number of researchers and air navigation service providers hypothesize that the inclusion of combined conflict-detection and resolution tools into air traffic control systems will reduce or transform controller workload and enable the required increases in airspace capacity.

In an effort to understand the potential workload implications of introducing advisory conflict-detection and resolution tools, this thesis provides a detailed study of the conflict event process and the implementation of conflict-detection and resolution algorithms. Specifically, the research presented here examines a metric of controller taskload: how many resolution commands an air traffic controller issues under the guidance of a conflict-detection and resolution decision-support tool. The goal of the research is to understand how the formulation, capabilities, and implementation of conflict-detection and resolution tools affect the controller taskload (system demands) associated with the conflict-resolution process,

and implicitly the controller workload (physical and psychological demands). Furthermore this thesis seeks to establish best practices for the design of future conflict-detection and resolution systems.

To generalize conclusions on the conflict-resolution taskload and best design practices of conflict-detection and resolution systems, this thesis focuses on abstracting and parameterizing the behaviors and capabilities of the advisory tools. Ideally, this abstraction of advisory decision-support tools serves as an alternative to exhaustively designing tools, implementing them in high-fidelity simulations, and analyzing their conflict-resolution taskload. Such an approach of simulating specific conflict-detection and resolution systems limits the type of conclusions that can be drawn concerning the design of more generic algorithms.

In the process of understanding conflict-detection and resolution systems, evidence in the thesis reveals that the most effective approach to reducing conflict-resolution taskload is to improve conflict-detection systems. Furthermore, studies in the this thesis indicate that there is significant flexibility in the design of conflict-resolution algorithms.

TABLE OF CONTENTS

DEDICATION . . . . . **iii**

ACKNOWLEDGEMENTS . . . . . **iv**

SUMMARY . . . . . **vi**

LIST OF TABLES . . . . . **xi**

LIST OF FIGURES . . . . . **xii**

GLOSSARY . . . . . **xvi**

I INTRODUCTION **1**

1.1 Thesis Objectives 12

1.2 Thesis Outline 12

II AIRSPACE MODEL **15**

2.1 Assumptions 16

2.2 Data Set 17

2.3 Sampling Procedure 20

2.4 Separation Detection 25

2.5 Application and Verification of the Sampling Procedure 27

2.5.1 Spatial Distribution of Aircraft 27

2.5.2 Maximum Altitude Distribution of Aircraft 29

2.5.3 Distribution of Aircraft Models 30

2.6 Scaling for Equitable Comparison 31

2.7 Review 36

III SECTOR CONFLICT ANALYSIS **38**

3.1 Assumptions 42

3.2 Uncontrolled Conflicts within a Sector 43

3.3 Controller Communication and Control Time 59

3.4 Review and Analysis 65

IV GRAPH REPRESENTATIONS OF POTENTIAL CONFLICTS . . . **68**

4.1 Assumptions 71

4.2 Representing aircraft and potential conflicts through graphs 72
4.3 Review of Conflict Graphs 77

V ABSTRACTION OF CONFLICT-DETECTION AND RESOLUTION TOOLS 79

5.1 Assumptions 82
5.2 Receding-Horizon Control 83
5.3 Parameterization of Conflict-Detection Systems 85
5.4 Parameterization of Conflict-Resolution Systems 88
5.4.1 Implementations and Capabilities: H_D, H_R, and δt 89
5.4.2 Conflict-Resolution Policies 94
5.5 Solution Times 101
5.6 Special Cases 104
5.7 Clearance-Based Control versus Trajectory-Based Operations 105
5.8 Review 107

VI SIMULATION MODELING FOR ABSTRACTED TOOLS 110

6.1 Creation of Complete Conflict-Graphs 111
6.2 Simulation Implementation 111
6.3 Parameter Settings and Traffic Scenarios 114

VII SIMULATION ANALYSIS 116

7.1 Aggregate Behaviors 117
7.2 Initial Parameter Sensitivity Analysis 122
7.3 Policy Comparison 125
7.4 Relative value between D_{sep}^{min} and H_R 132
7.5 Benefit of Improved Conflict-Resolution 139
7.6 Managing Uncertainty and Solve-Times 142
7.7 Review 147

VIII CONCLUSION 150

8.1 Summary 150
8.2 Contributions 152
8.3 Future Work 153

APPENDIX A — DECISION POLICIES 157

REFERENCES 163

LIST OF TABLES

1 Best-fit coefficients for the uncontrolled conflict models. 50

2 Coefficients and R^2 values for least-squares lines of best-fit considering the ratio of conflict complexity. 57

3 Average communication time for each event. 62

4 Worst-case model errors between simulation averages and predicted averages over all configurations, traffic intensities, and sectors. 118

5 Policy Comparisons: ZMP12 127

6 Policy Comparisons: ZMP16 127

7 Policy Comparisons: ZMP42 127

8 Configurations with $D_{sep}^{min} \leq 5NM$ for which there is a statistical difference in the conflict-resolution taskload distribution over 2.5 minutes for ZMP42 . 131

9 Knowledge-Based Policy Rules 160

LIST OF FIGURES

1 Map of the 20 Air Route Traffic Control Centers covering the continental United States. 2

2 Center map for Minneapolis Center (ZMP) with high-alitude sector boundaries. 3

3 Under current operations controllers use radar and vocal communication systems to track and issue commands to aircraft. 5

4 Conflict-detection and resolution decision-support tools provide possible resolution solutions to air traffic controllers. 7

5 Representation of factors affecting controller workload.[66] 8

6 Chapter dependency 14

7 High-altitude traffic pattern for June 14^{th}, 2007 in Minneapolis Center. . . 18

8 High-altitude boundaries for Areas in ZMP. 18

9 High-altitude sectors boundaries in Area 6 of ZMP. 19

10 Diurnal aircraft counts within Minneapolis Center for the dates May, 21, 2007 through June 17th, 2007. 20

11 Spatial clustering of origins described by latitude and longitude. 21

12 Distribution for origin-destination pairs for flights within Minneapolis Center. 22

13 Clustering relationships between repeated origin-destination pairs and aircraft fights. 22

14 Example conflict between two aircraft. 26

15 Spatial distribution of aircraft for June 14^{th}, 2007 and a generated traffic scenario. 28

16 Distribution of maximum aircraft altitudes for June 14^{th}, 2007 and a generated traffic scenario. 30

17 Distribution of aircraft models for June 14^{th}, 2007 and a generated traffic scenario. 31

18 Illustrative example of the subsampling procedure. 33

19 Extrapolated conflict totals based on sub-sampled traffic scenarios. 34

20 Coefficient of determination, R^2 for models based on sub-sampled traffic scenarios used to predict conflict totals as a function of traffic intensity in each sector. 36

21 Example of conflict graph $\mathcal{G}$ with three conflict-clusters. 40

22 Potential conflicts can be characterized by the number of aircraft involved and overall complexity of generating a resolution. 41

23 Conflict totals for sectors in Minneapolis Center over a 24hr time period when $D_{min}^{sep} = 9$ NM. 44

24 Traffic Density Maps. 44

25 ZMP12 : Probability distribution of flight phases. 46

26 ZMP42 : Probability distribution of flight phases. 46

27 ZMP16 : Probability distribution of flight phases. 46

28 Location of uncontrolled conflicts according to conflict configuration. 47

29 Discrete empirical probability distribution of uncontrolled conflict crossing angles (5^o increments). 48

30 Cumulative number of uncontrolled conflicts for sectors ZMP12, ZMP17, and ZMP42, when $D_{min}^{sep} = 9$ NM. 49

31 Cumulative number of aircraft arrivals for sectors ZMP12, ZMP17, and ZMP42. 49

32 Simulated number of uncontrolled conflicts and best-fit lines according to traffic intensity and separation distance for ZMP42. 51

33 Average rate of uncontrolled conflicts when $D_{sep}^{min} = 9$ NM. 52

34 Cumulative distribution of the number of uncontrolled conflicts for the busiest 5 minute period of the day when $D_{sep}^{min} = 9$ NM for ZMP12. 53

35 Cumulative distribution of the number of uncontrolled conflicts for the busiest 5 minute period of the day when $D_{sep}^{min} = 9$ NM for ZMP16. 53

36 Distribution of the number of uncontrolled conflicts for the busiest 5 minute period of the day when $D_{sep}^{min} = 9$ NM for ZMP42. 54

37 Distribution of the number of aircraft invovled in each conflict cluster as traffic intensity increases for $D_{sep}^{min} = 9$ NM for ZMP42. 55

38 Distribution of the number of edges in each conflict cluster as traffic intensity increases for $D_{sep}^{min} = 9$ NM for ZMP42. 55

39 Fraction of pairwise uncontrolled conflicts. 56

40 Average growth of conflict clusters as intensity increases. 57

41 Number of expected conflicts versus number of expected aircraft in each cluster. 58

42 Two conflict clusters of differing conflict complexity. 58

43 Representation of the event process within an airspace, and method for calculating the minimum estimated controller communication time. 61

44 Average utilization time with upper and lower quartiles for ZMP42. 63

45 Average utilization time with upper and lower quartiles for $T_w = 5$ minutes. 63

46 Probability of over-utilization for different time-windows, T_w. 65

47 Air traffic controller deferring to an advisory conflict-detection and resolution system. . . . 70
48 Example graph representation of aircraft and potential conflict relationships. 73
49 Conflict-resolution problem solved in a receding-horizon control framework. 84
50 Safety region around an aircraft (not to scale) . . . 85
51 Parameters determining labeling of potential conflicts. . . . 86
52 Representation of potential conflicts that are considered in the decision-making process. . . . 90
53 Larger values of the guaranteed conflict-free time, H_R, help to prevent secondary conflicts. . . . 92
54 With solve-times of δt, trajectory predictions are updated for each implementation of the conflict-resolution tool. . . . 94
55 Sector map of ZMP42 with jetroutes . . . 98
56 Example application of the minimum vertex cover problem. . . . 100
57 In the planar case, there exist two topologically distinct options for each resolution command. . . . 102
58 Finding solution to the motion planning problem becomes increasingly difficult as time horizons increase. . . . 104
59 Two example resolution solutions for a potential conflict . . . 106
60 Difference between avoidance maneuvers and closed-loop resolution commands. 107
61 Traffic density map of ZMP center, highlighting sectors of interest. . . . 115
62 Qualitative behavior of conflict-resolution taskload with regards to the traffic intensity and aircraft spacing. . . . 118
63 Qualitative behavior of conflict-resolution taskload with regards to the uncertainty and solve-time. . . . 119
64 Qualitative behavior of conflict-resolution taskload with regards to the decision-horizon time. . . . 120
65 Qualitative behavior of conflict-resolution taskload with regards to conflict-free resolution time. . . . 121
66 FCFS parameter sensitivity analysis: Range of percent reduction in conflict-resolution taskload. . . . 124
67 MCRT parameter sensitivity analysis: Range of percent reduction in conflict-resolution taskload. . . . 124
68 Representative timeline of conflict-resolution advisories. . . . 126

69 Percent of configurations for which the MCRT policy is statistical different from a rule-based heuristic policy (Reject H_o). 128

70 Maximum difference in the expected conflict-resolution taskload between the MCRT policy and the random policy over a range of configurations. 129

71 Probability distributions of the conflict-resolution taskload for the busiest 2.5 minute time period in each traffic scenario. 130

72 Expected number of advisory resolution commands for the FCFS policy at 1X, 2X, and 3X traffic intensities. 134

73 Required value of H_R to match a reduction in aircraft spacing (ZMP12). . . 136

74 Required value of H_R to match a reduction in aircraft spacing (ZMP42). . . 136

75 Required value of H_R to be equivalent to a reduction in the separation requirements of $\Delta D_{sep}^{min} = 0.5$ NM (ZMP42). 137

76 Percent decrease in the conflict-resolution taskload by reducing aircraft spacing when compared to $D_{sep}^{initial} = 9$ NM. (At the 1X, 2X, and 3X traffic intensities) 138

77 Conflict-resolution taskload for ZMP42 (FCFS, $\delta t = 0$, $D_{sep}^{r} = 0$). 139

78 Percent reduction in conflict-resolution taskload compared to $H_R = 6$ minutes (ZMP42). 140

79 Relative reductions in conflict-resolution taskload for ZMP12 and ZMP16. . 141

80 Distribution of aircraft transit times [minutes] through sector (1 minute increments). 142

81 Distribution of inter-arrival times [minutes] between the first and last potential conflict of an aircraft (1 minute increments). 142

82 Conflict-resolution taskload as a function of the solve-time and uncertainty parameters. 143

83 Required value of H_R to match faster solve-times. 146

84 Representative summary of analysis. 147

85 Taking full advantage of conflict-resolution systems may require a system-wide airspace redesign. 156

86 Different phases of flight considered in the knowledge-based policy. 160

GLOSSARY

Acknowledgement When an aircraft enters an airspace, the pilot must announce his presence to the air traffic controller managing it. The air traffic controller responds by acknowledging the pilot.

Air Navigation Service Provider (ANSP) Organization responsible for managing aircraft in an airspace on behalf of a nation or a number of nations.

Air Route Traffic Control Center (ARTCC) A large volume of airspace, defined by vertical and lateral boundaries. Commonly referred to as centers. Centers are divided into a number of sectors. In the United States, a center covers several states.

Air Traffic Controller The acting agent of an air navigation service provider. The air traffic controller is responsible for managing the traffic and ensuring required spatial separation between all aircraft in an airspace.

Airspace A generic volume of space defined by vertical and lateral boundaries.

Airspace Capacity The nominal number of aircraft an air traffic controller is able to manage under reasonable workload.

Automatic Dependent Surveillance-Broadcast A dedicated data channel and protocol for communicating between aircraft and ground stations. A major function of ADS-B is to act as a primary surveillance source for reporting GPS position information.

Clearance-Based Control The current operational protocol for managing aircraft. Aircraft trajectories are managed in piecewise increments through a sequence of request and clearances.

Complexity A generic term relating the difficulty of a particular traffic scenario.

Conflict When any two aircraft break minimum separation requirements. For the en route environment aircraft must maintain 5NM lateral, and 1,000ft vertical separation from each other at all times.

Conflict-Resolution Taskload The number of resolution commands used in ensuring separation between aircraft in a given time period.

Controller Workload The amount of effort, both physical and psychological, expended in response to system demands (task load) and also in accordance with the operator's internal standard of performance.

Direct Routing When aircraft fly between two points in space along the shortest path.

EUROCONTROL The air navigation service provider for much of Europe.

Federal Aviation Adminstration (FAA) The air navigation service provider for the United States of America.

Hand-off When an aircraft transitions from one airspace to another, the air traffic controller in the originating airspace provides the aircraft with the communication frequency of the destination airspace. The pilot of the aircraft is then expected to communicate with the controller in the destination airspace in order to be acknowledged.

Instrument Flight Rules (IFR) A set of rules and regulations aircraft and controllers must operate under when guided by instrumentation (e.g. GPS and Radar). For en route aircraft this is typically for aircraft above 20,000ft.

National Airspace System Refers to a volume of airspace, and the people, procedures, equipment, and facilities required to manage it.

NextGen The conceptual next generation air traffic system for the Federal Aviation Administration.

Potential Conflict A potential conflict exists between two aircraft when the best available information predicts that their aircraft trajectories will violate separation requirements.

Sector A singular unit of airspace under the control of one or two air traffic controllers.

Separation requirement The minimum required distance between any two aircraft. If separation requirements are violated, then a conflict is deemed to have occured.

SESAR Single European Sky ATM Research (SESAR) is the conceptual next generation air traffic system for EUROCONTROl.

Taskload The number of tasks or frequency of task occurrence associated with a specific job description.

Trajectory-Based Operations (TBO) A future operational protocol and concept for managing aircraft. Replacing clearance-based operations, trajectory-based operations supports longer-term path planning of aircraft. Instead of providing clearance for individual aircraft actions (heading, altitude, and speed changes), the air traffic controller approves complete trajectories.

CHAPTER I

INTRODUCTION

This thesis is concerned with providing support for the design of decision-support tools to aid air traffic controllers. The projected growth in air transportation demand is likely to result in traffic levels that exceed the abilities of the unaided air traffic controller in managing, separating, and providing services to aircraft. From 2010 to 2030, the number of instrument flight rules (IFR) aircraft operations handled by Federal Aviation Administration (FAA) en route traffic centers is predicted to increase from approximately 39 million flights to 64 million flights [35]. Consequently, the FAA, and other air navigation service providers (ANSP) around the world, are making several efforts to improve the capacity and throughput of existing airspaces by means of airspace redesign, trajectory based operations, new traffic flow management tools, and automated data communication and navigation systems [49, 86]. Ultimately, the stated goal of the Joint Planning and Development Office is to triple the available capacity of the National Airspace System by 2025 [95].

National Airspace System. In order to address how air traffic controllers can be best aided, it is important to understand the composition of the National Airspace System (NAS). The National Airspace System, as operated by the FAA, enables the safe and efficient transport of aircraft. It consists of a heterogeneous set of interacting units that includes people, airspaces, technologies, practices, procedures, and policies. Major functional elements of the National Airspace System include the 22 Air Route Traffic Control Centers (ATRCC), whose responsibilities include managing en route aircraft. A map of the 20 centers spanning the continental United States is shown in Figure 1. Each center is stratified into multiple layers (commonly, low-altitude, high-altitude, and super-high-altitude), which are further subdivided into en route sectors. Figure 2 provides an example center map with a subset of the high-altitude sector boundaries for Minneapolis Center (ZMP).

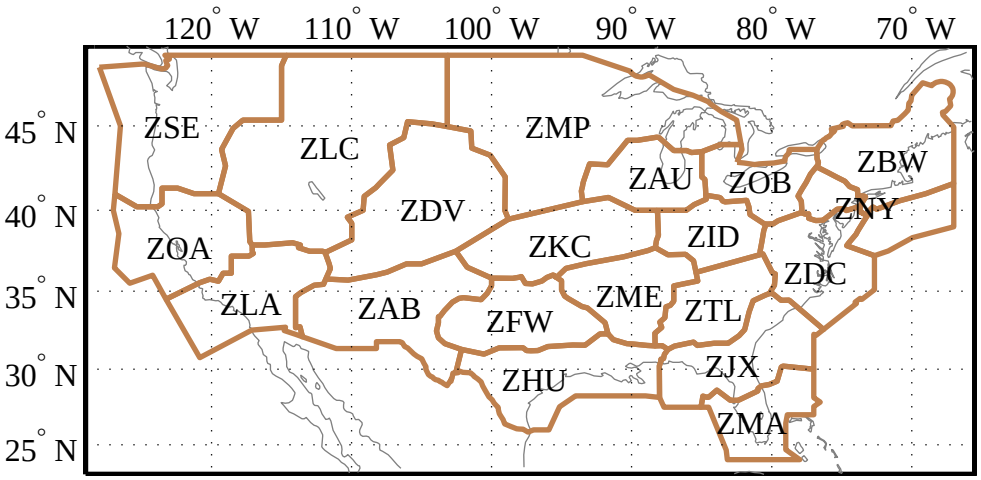


Figure 1: Map of the 20 Air Route Traffic Control Centers covering the continental United States.

Depending on traffic conditions, one or two air traffic controllers are responsible for managing aircraft traffic in a sector. The primary objective of the air traffic controller working an en route sector is to ensure the proper separation of aircraft at all times. A variety of approaches and mechanisms exist to improve air traffic management.

Airspace Redesign. In an effort to limit the workload of controllers in the face of increasing traffic demand, airspaces have been redesigned with smaller and more specialized sectorizations for air traffic controllers to manage. Such sectorization allows air traffic controllers to monitor a smaller set of aircraft, while decreasing variance in traffic patterns. Examples of such airspace redesign include sectors ZMP15 and ZMP16 in Minneapolis center, both which are dominated by aircraft arrivals into Minneapolis - St. Paul International Airport and are significantly smaller than other sectors in the center, c.f. Figure 2. There exist limits for which the strategy of sectorization will remain fruitful in limiting controller workload. The introduction of smaller sectors inherently reduces the amount of traffic that any single sector can absorb, and places restrictions on the types of conflict-resolution maneuvers available to controllers. Additionally, smaller sectors require additional staffing and greater coordination between adjacent sectors. The advantage of smaller sectors in terms of managing controller workload and traffic complexity is reaching its limits. Thus,

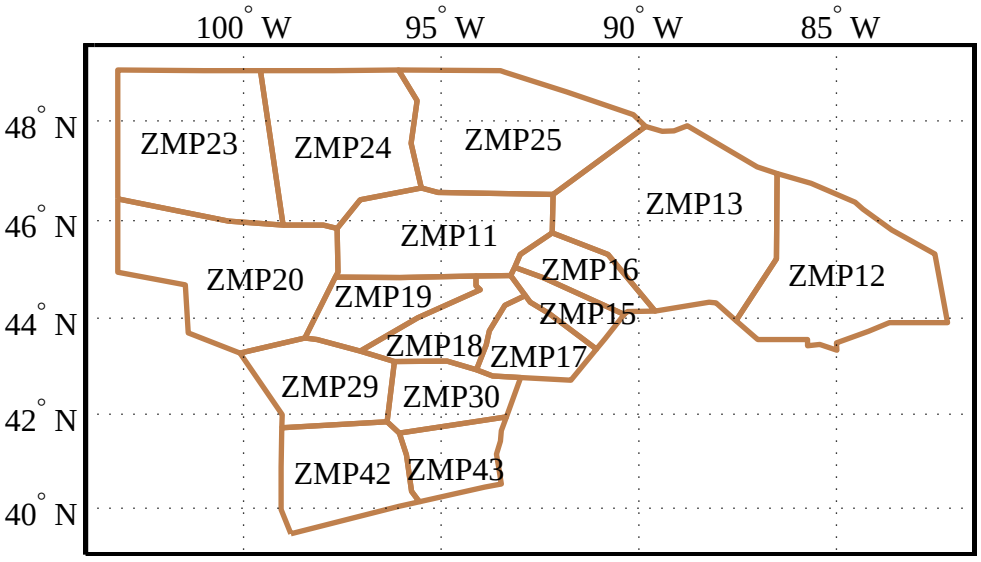


Figure 2: Center map for Minneapolis Center (ZMP) with high-alitude sector boundaries.

while airspace redesign, along with the introduction of tighter navigation standards, will result in further capacity gains, these changes alone are not sufficient to accommodate the anticipated growth in air traffic demand.

Automated Conflict-Resolution. There has also been a significant investment into the study and development of aircraft conflict and conflict-resolution systems in order to increase capacity. Implicitly, conflict-detection algorithms are designed to identify conflicts and enable the conflict-resolution programs to generate conflict-free aircraft trajectory solutions. A number of researchers and ANSPs hypothesize that conflict-detection and resolution tools will reduce or transform controller workload by decreasing the amount of time and mental effort controllers spend detecting and resolving potential conflicts [8, 23, 89, 98]. Indeed, the conflict-detection and resolution process can be time consuming. Under current operations, for each potential conflict, air traffic controllers are required to discern if intervention is required; generate resolution commands for one or more aircraft; verbally issue the commands to the aircraft; and monitor and ensure the implementation of the commands. For potential conflicts between two aircraft, one study estimates the average total time required

from controllers, excluding monitoring implementation, is 27.6 seconds [20]. Despite the appearance of potential reductions in controller workload by displacing some or all separation tasks to a computer system, there are still concerns that conflict-detection and resolution algorithms do not aid and could hinder air traffic control performance. Furthermore, practical implementation has not been achieved. From a human-factors perspective, researchers have begun to address the topic of 'allocation of function' in air traffic control: that is, what functions and levels of control should be designated to humans or computers. Early results have exposed a number key benefits and drawbacks of various levels of automation.

Early examples of conflict-resolution algorithms include [28, 69, 70], with a more comprehensive survey of proposed models presented in [56]. More recent research and development in conflict-resolution algorithms seeks to design automated algorithms that provide provably safe solutions while attempting to satisfy aircraft dynamic limitations [16, 25]. The majority of these previous developments address a completely automated conflict-detection and resolution system. Such an automated system is consistent with the proposal, advocated by some air traffic operators and researchers, that the tactical role of radar air traffic controllers should eventually transition into one of supervisory control of aircraft; an example discussion is provided in [33]. In a supervisory control framework, it is the role of automated tactical conflict-detection and resolution algorithms to ensure separation.

There are significant limitations associated with automated conflict-resolution algorithms. All automated approaches require advanced digital communication and navigation sub-systems to directly communicate with – and control – aircraft. The development and implementation of these systems are hindered by the slow uptake of the advanced avionics required to fully support them. Furthermore, the safety guarantees of these conflict-resolution algorithms have yet to be established, particularly in accordance with aircraft dynamics, as all algorithms make use of simplified aircraft models. Such systems are in contrast to current operations, which are depicted in Figure 3. As it stands, air traffic controllers use radar systems and vocal communication to manage and issue commands to aircraft. There are only limited forms of automation, none of which correspond to conflict-resolution. In these systems, air traffic controllers are responsible for generating resolution commands on

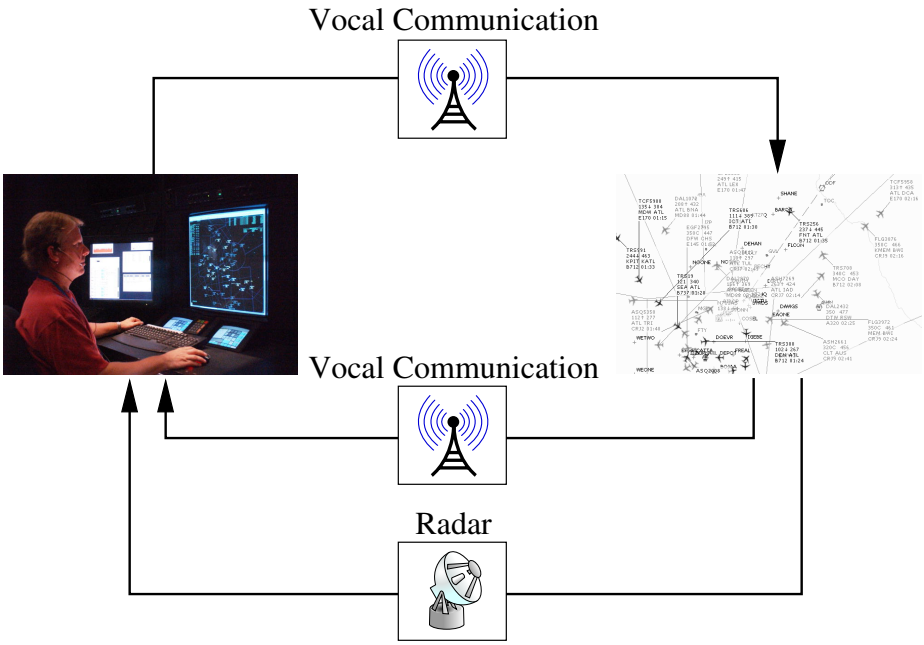


Figure 3: Under current operations controllers use radar and vocal communication systems to track and issue commands to aircraft.

their own.

While there are many proponents for automated conflict-detection and resolution algorithms to automate air traffic systems and separate aircraft, human-factors research has identified potential hazards. When acting in a supervisory role, under greater levels of automation, air traffic controllers lose situational awareness and are unable to identify all future conflicts [64]. Thus, if automated conflict-detection and resolution systems fail to provide a guaranteed safe solution or the resolution command is not implemented properly, then it is unlikely an air traffic controller, acting in a supervisory role, will be able to identify the failure. Furthermore, in the event when air traffic controllers are completely removed from the system, the lack of fall-back options in the case of failure or nonconforming aircraft is a prohibitive factor. The option of supervisory control is deemed unsuitable by many researchers because such a framework ultimately leads to skill atrophy [71, 51]. In cases of system failure, air traffic controllers lack the proper training and preparedness to respond appropriately. There are also major concerns about the adverse impact on motivation of air traffic controllers and the impact on safety.

Mixed Automation for Conflict-Resolution. Other efforts have been directed at mixed-levels of automation in the conflict-resolution process that are more consistent with human-factors issues. ERASMUS, a EUROCONTROL funded project to study methods for including automation concepts into air traffic control, introduced the concept of 'subliminal control.' In this approach, significant portions of traffic are deconflicted or further spaced with minor automated speed control commands ($\pm 6\%$); speed-change commands of this magnitude are not perceptible to pilots or controllers. The use of minor speed changes to space aircraft reduces both the total number of aircraft that controllers actively monitor for potential conflicts and the number of resolution commands that controllers issue. There exist numerous publications related to the ERASMUS project and subliminal control [22, 29, 40]. Early human-in-the-loop simulation studies suggest that controller workload is reduced.

Despite the potential benefits of subliminal-control, there are concerns about the practicality and ability to implement the systems in the real world: any subliminal-control system would require advanced data communication and navigation systems to control aircraft directly. Studies have also indicated that mixed-modes of operation that include dynamic allocation of separation tasks leads to ambiguity about who is in control and who is responsible for recovery in the case of system failure: pilots or controllers. There is also concern that the process may lead to loss of job satisfaction for air traffic controllers as a greater portion of functions is allocated to automation systems [9].

Improving User Displays. Another approach that has received considerable support from human-factors researchers is the improved design of cognitive and display tools to provide information in a manner that aids air traffic controllers in parsing information and coming to quick and appropriate decisions, an example of which is provided in [57]. Overall, this framework requires little adjustment to the fundamental work practice and methods employed by controllers. However, it is unknown if the improvement of user displays will remain suitable at high traffic volumes when taskload demands grows large.

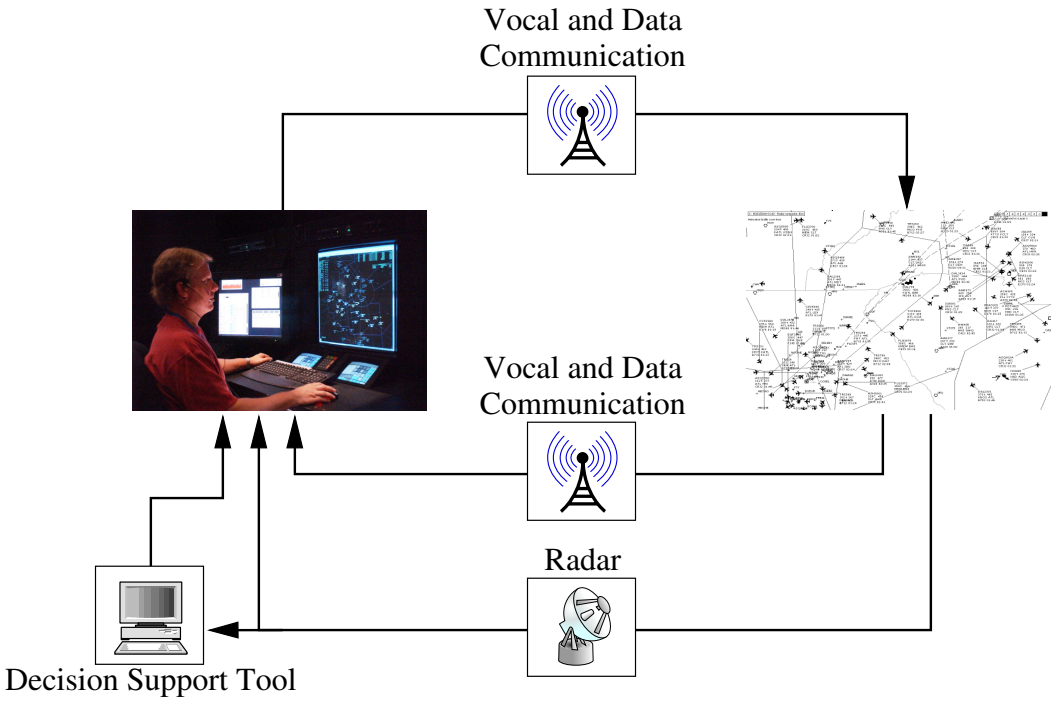


Figure 4: Conflict-detection and resolution decision-support tools provide possible resolution solutions to air traffic controllers.

Advisory Decision-Support Tools. The research presented here focuses on another option touted by human-factors researchers, which is the inclusion of advisory conflict-detection and resolution decision-support tools to aid air traffic controllers, without replacing them. Advisory decision-support tools that provide conflict-resolution options to air traffic controllers are an alternative to fully- or semi-automated systems that require advanced data communication and navigation sub-systems. A schematic diagram of the control structure for human-in-the-loop control with conflict-detection and resolution decision-support tools is illustrated in Figure 4. In a human-in-the-loop framework, conflict-detection and resolution decision-support tools identify conflicts and propose resolution commands for the air traffic controller to verify and issue to aircraft. The solution trajectories can then be uploaded through a data-link into the aircraft flight management system for approval by pilots.

The major benefits of decision-support tools include an indifference to high traffic volumes and neutral effects on situational awareness. Much like automated or highly-computerized solutions, conflict-detection and resolution tools can be designed to handle

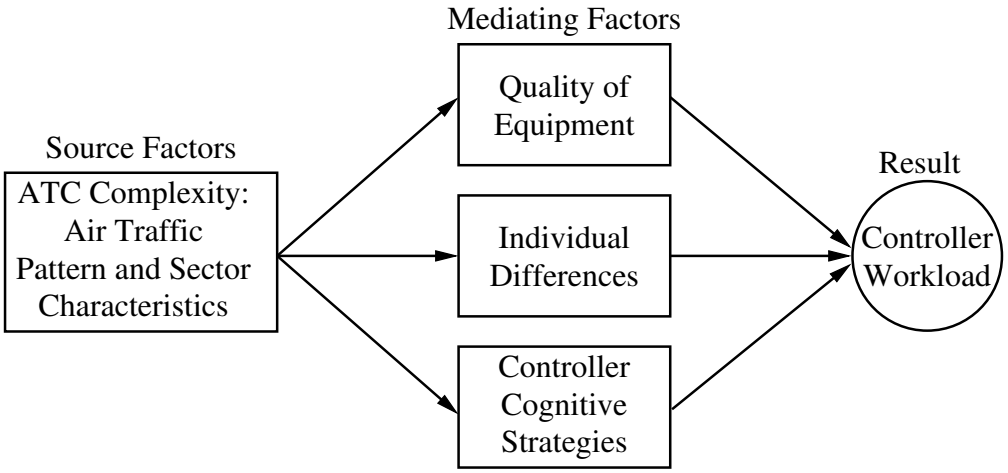


Figure 5: Representation of factors affecting controller workload.[66]

large traffic loads, and provide solutions in real-time. Furthermore, because the air traffic controller is still part of the decision process and has the option to accept or reject proposed solutions, there exists a safety fall-back. However, concerns still remain about the skill atrophy of controllers and the proper design the decision-support tools. For example, conflict-detection and resolution systems should be designed appropriately so that human controllers neither overly depend on the tools nor lack faith in the tool and disregard it [75, 91].

The inclusion of human-in-the-loop decision-support tools in human-based air traffic control operations requires a fundamentally different approach to the design and implementation of conflict-detection and resolution algorithms. The algorithms must explicitly acknowledge the role of the controllers and accommodate their abilities. Central to determining the potential success of decision-support tools for separation assurance, research is required to demonstrate real and meaningful benefits. There is a need to establish that the inclusion of conflict-resolution decision-support tools will reduce, or at the very least not worsen, controller workload, while improving service. The research presented here seeks to understand how the formulation, capabilities, and implementation of conflict-detection and resolution tools affect the controller. Specifically, the metric of interest is the taskload associated with the conflict-resolution process, which is implicitly tied to controller workload.

Controller Workload. Controller workload associated with air traffic control corresponds to the stress, or "...the amount of effort, both physical and psychological, expended in response to system demands (task load) and also in accordance with the operators internal standard of performance" [94].' The factors leading to controller workload are numerous. In the review by Mogford et al [66], the authors identify four factors that influence controller workload: air traffic control complexity, quality of equipment, individual differences, and controller cognitive strategies. Air traffic control complexity refers to a set of descriptions of the prevailing air traffic patterns and sector characteristics that aid or hinder in the management of aircraft. The next three factors (equipment, individual differences, and strategies) are referred to as mediating factors. Quality of equipment, for example, considers the accuracy of radar systems or the usability of user displays and interfaces for computer-support tools. Individual differences account for anxiety levels, personality, age, and experience, which have been demonstrated to correlate with controller workload. And lastly, controller cognitive strategies incorporate how controllers adjust their strategies in managing traffic according to pressures from increasing traffic demands. For example, at low traffic volumes controllers consider a wide variety of information and options in managing traffic, while at high traffic volumes, controllers limit the amount of information they consider and adjust management strategies based on economy, including strategies for resolving potential aircraft conflicts [92].

Complexity. Early technical practitioners attempted to characterize controller workload through a set of intrinsic properties associated with source factors, without directly dealing with the mediating factors. They established the term air traffic control complexity (sometimes simply referred to as complexity) to indicate the relative ease or difficulty of managing traffic within an airspace. The dynamic density of an airspace is a validated and commonly used measure of complexity [17, 55, 58, 93]. Initially, the goal of dynamic density was to match the perceived workload of air traffic controllers by considering a large set of measurable factors (number of aircraft in the airspace, distribution of aircraft speeds, spatial distribution of aircraft, etc.). Through human-in-the-loop experiments, regression models

were established to estimate controller workload. Kopardekar et al. [55] present a review of regression parameters used to compute the dynamic density. While only accounting for half the variance in controller workload, dynamic density established that the number of potential conflicts and aircraft maneuvers, in addition to the local density of aircraft and other measures, are strong factors in predicting controller workload.

More recent efforts seek to establish intrinsic measures of complexity using different features than those accounted for in dynamic density. Prandini et al. [78] propose probabilistic air traffic complexity measures based on aircraft occupancy within the airspace. A similar probability model based on traffic flows is proposed in [84]. The authors demonstrate how parameterized generative models can be used to create airspace maps to highlight regions of high traffic or regions where aircraft interactions and conflicts are more likely. Delahaye and Puechmorel [24] propose a complexity measure based on non-linear dynamical systems. Their measure of Lyapunov Exponents captures the structure in a situation by identifying the organization of trajectories in traffic patterns.

A common theme to the previous works is that they all implicitly consider aircraft conflicts and interactions. Dynamic density acknowledges aircraft interactions by including the spatial density of aircraft and number of potential conflicts in its measure. Furthermore, approximations of the conflict-resolution process is partially accounted for by including aircraft altitude, heading, and speed changes. These maneuver actions are often in response to potential conflicts or a desire by controllers to further space aircraft to ensure clear separation. The second set of complexity measures discussed are based on either a measure of interaction (i.e. probability of conflict) or measure of organization of the traffic flow. Regardless, all the previous methods implicitly consider potential conflicts, and the resulting conflict-resolution process.

There should be no surprise that aircraft interactions and conflicts are sources of complexity and controller workload. The primary objective of the air traffic controller is to ensure the proper separation of aircraft, thereby avoiding collisions, at all times. Thus, conflict resolution takes precedence over any other management tasks.

Research Needs. In an effort to understand the potential workload implications of introducing conflict-detection and resolution tools into air traffic control systems, a detailed study of the conflict event process and the implementation of the separation algorithms are necessary. Specifically, the research presented here addresses the conflict taskload requirements to resolve traffic. That is, the studies contained in this thesis examine how many resolution commands air traffic controllers issue in order to separate traffic under the guidance of a conflict-detection and resolution decision-support tool. Such an approach is valid given the importance of the conflict-resolution process to influencing controller workload, as implicitly validated by dynamic density and other methods. Addressing taskload requirements is something previous researchers have considered in limited forms, even in regards to the development of conflict-resolution algorithms.

Studying and modeling the conflict-detection and resolution event process enables an assessment of the number of potential conflicts air traffic controllers can expect to encounter, the structure of the potential conflicts (i.e. how many aircraft are involved), and the manner in which the potential conflicts occur in time. Furthermore, analysis of potential conflicts facilitates a discussion of the best practices for designing and implementing conflict-detection and resolution systems in an effort to regulate the number of resolution commands used to space aircraft.

The method by which conflict-detection and resolution algorithms identify and resolve conflicts is critical to the function of a decision-support tool. Much like human air traffic controllers assess potential conflicts and determine conflict-resolution solutions, any human-in-the-loop decision-support tool will be required to do the same. The process by which these tasks occur is not straightforward. The ability to identify potential conflicts depends on the accuracy of trajectory prediction and aircraft position reporting systems. Meanwhile, the guarantee that any resolution command ensures conflict-free travel for aircraft depends on the accuracy of the conflict-detection algorithms and the strength of the conflict-resolution algorithms. Previous development in conflict-resolution algorithms has largely overlooked the accuracy and availability of information in regards to trajectory prediction and conflict detection. Ultimately, any conflict-detection and resolution algorithm must consider if,

when, and what resolution commands should be issued. These considerations are a function of both the information available and the policy under which the conflict-resolution algorithm operates.

1.1 Thesis Objectives

In the face of increasing air traffic, there is a growing need to design systems to support, not replace, air traffic controllers in managing and separating aircraft. If human-in-the-loop decision-support tools for conflict-detection and resolution are to be integrated into the repertoire of air traffic controllers, then it is vital that the tools operate in a manner consistent with human work practice. Thus, at a minimum, designers of conflict-detection and resolution algorithms must account for taskload implications inherent in any design. The objective of this thesis is to explore and accomplish the following:

- Generate a method to model traffic in order to assess the conflict event-process within an airspace.
- Abstract the behaviors and characteristics of conflict-detection and resolution systems. According to the abstractions, analyze how their implementation and capabilities affect controller taskload.
- When possible, establish best-practices for designing conflict-detection and resolution algorithms to manage the number of resolution commands used to prevent air traffic conflicts.

By researching the topics above, the major contribution of the thesis is to derive performance reference models that serve as a comparison tool, and ideally, to establish best practices for the design of real-world conflict-detection and resolution systems. Lastly, understanding how the design of conflict-detection and resolution systems affects controller taskload provides a basis to improve models for predicting controller workload.

1.2 Thesis Outline

The thesis is divided into two major sections. The first major section, begins with Chapter 2 introducing a procedure for modeling uncontrolled air traffic within an airspace. The

resulting traffic model serves as a basis for the remainder of the thesis, except where stated. The traffic model enables the analysis of the conflict event process presented in Chapter 3, where the temporal and spatial distributions of conflict events are explored. Multiple airspaces are studied to demonstrate that the design and structure of an airspace results in different levels of air traffic control complexity. The analysis in Chapter 3 motivates the need for advisory conflict-detection and resolution decision-support tools.

Following the initial analysis, the second major section, beginning with Chapter 4, proposes a graph-based methodology for modeling the conflict-detection and resolution process. Next, in Chapter 5, a framework for abstracting conflict-detection and resolution is described. The abstraction considers the implementations and design features for conflict-detection and resolution algorithms. Using the uncontrolled traffic models developed in Chapter 2 and the abstraction of conflict-detection and resolution systems in Chapter 5, a process for executing traffic simulation is described in Chapter 6. From the simulations of the graph-based conflict-detection and resolution algorithms, taskload requirements for air traffic controllers using these tools are analyzed and best practices are proposed in Chapter 7. Finally, a summary and conclusions on the design of conflict-detection and conflict-resolution algorithms are discussed in Chapter 7.

To aid in navigation of the thesis, a representation of the chapter dependencies is shown in Figure 6.

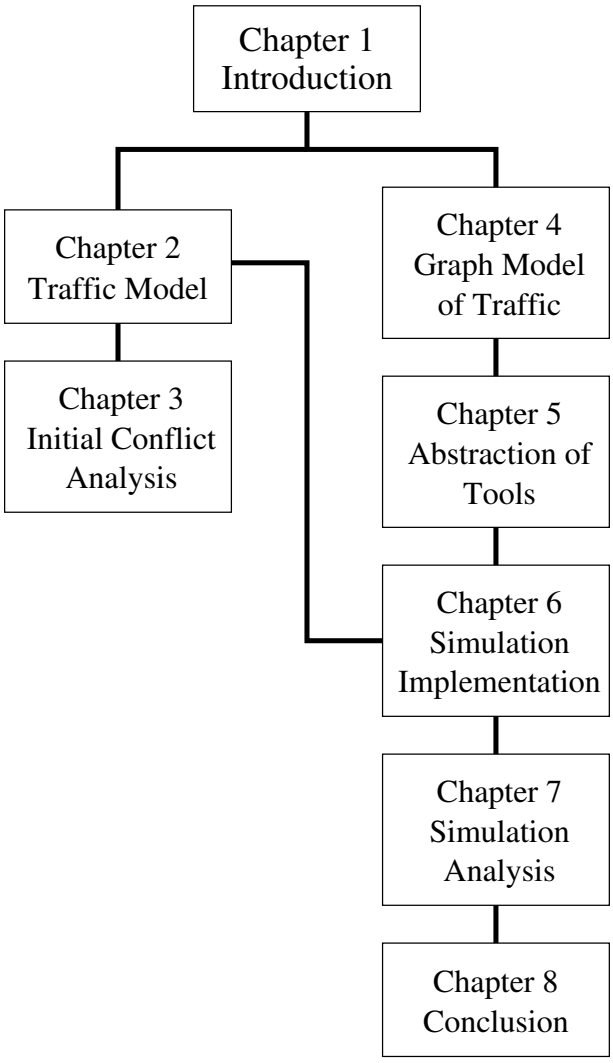


Figure 6: Chapter dependency

CHAPTER II

AIRSPACE MODEL

In this chapter, a methodology for generating an uncontrolled open-loop air traffic model is presented. The resulting uncontrolled traffic model approximates air traffic within an airspace as if aircraft flew irrespective of each other and without controller input. The uncontrolled air traffic model enables characterization of the conflict-event process within a sector and is later used to assess the performance of conflict-detection and resolution algorithms. Specifically, the uncontrolled air traffic model attempts to replicate potential conflicts that would be commonly resolved in real-world operations - in both location and aircraft configuration and structure. The traffic model is based on a re-sampling of historical aircraft radar data.

Using resampled historical radar data, scenarios of mock 4D uncontrolled flight trajectories are created that include variations over a range of variables. The goal is to replicate variance that is commonly found in air traffic, even for aircraft flying along the same path. In practice, variations in aircraft trajectories commonly appear in aircraft climb rates and speeds. These differences are due to airline and pilot preferences, as well as present atmospheric conditions, aircraft and engine capabilities, flight management systems (FMS), and aircraft weight.

The methodology presented here to model uncontrolled air traffic is an alternative to collecting historical flight plan information and simulating aircraft traffic along the flight plans with stochastic properties, similar to that in [3, 10]. It is also an alternative to using the software described in [10], which was inaccessible due to security restrictions.

The methodology presented here to model uncontrolled air traffic is an alternative to collecting historical flight plan information and simulating aircraft traffic along the flight plans with stochastic properties, similar to that in [3, 10]. By re-sampling aircraft trajectories, proprietary aircraft and engine manufacturer data, as well as sophisticated simulation

software are not required. The methodology described here is an alternative to using the software described in [10], which was inaccessible due to security restrictions. Properties of the methodology for generating high-intensity traffic are verified to be consistent with the software in [10].

The traffic model created in this chapter is used throughout the remainder of the thesis, except where noted. First, in Section 2.1 a list of assumptions required for the development and the use of the model is given. In Section 2.2, a description of the data set used to construct the uncontrolled open-loop model is provided. Next, Section 2.3 details the procedure for creating sample traffic days of uncontrolled air traffic. Section 2.4 defines how potential aircraft conflicts are identified. Finally, verification of key properties is provided in Section 2.5.

2.1 Assumptions

The open-loop traffic model generated in this thesis attempts to represent uncontrolled air traffic within an airspace, as if aircraft flew irrespective of each other, and without controller input. The traffic model is based on a re-sampling procedure of historical flight data. The traffic model is based on the following assumptions:

- A traffic pattern can be parameterized by the origin and destination of each flight. Re-sampling flights to maintain the same origin-destination pairs results in traffic cases that mimic the original spatial traffic pattern. Replicating the spatial traffic pattern preserves the overall experience of the air traffic controller.

- Scaling traffic by reducing the time frame over which it occurs is equivalent to increasing the total amount of traffic (i.e. simulating 24 hours of traffic over 8 hours is similar to three times the traffic in a 24 hour time period). By extension, a 20 minute time period with triple the traffic demand can be scaled over a longer time period to compute taskload demands over a complete 24 hour day.

- Origin-destination demand pattern, aircraft models, sector boundaries, and routes remain similar as traffic intensity increases. Or implicitly, in 20 years when traffic

volumes have increased, the traffic, aircraft, and sector properties remain similar. From a practical perspective, this is a tenuous assumption, as almost surely, fleet mixes and sector boundaries will change in time. Furthermore, there is limited understanding of the impact of unmanned aerial vehicles on air traffic management.

- The sampling procedure models uncontrolled air traffic in a manner that allows for valid assessment of the conflict-event process.

- The validity of the uncontrolled traffic model in an airspace is preserved despite possible controller intervention of aircraft in adjacent airspaces.

2.2 Data Set

The traffic model of uncontrolled flight trajectories is based on 28 days of Performance Data Analysis and Reporting System (PDARS) data. The PDARS data set includes aircraft radar position information (latitude, longitude, altitude) nominally sampled every 12 seconds, covering the dates May, 21, 2007 through June 17th, 2007, over Minneapolis Center (ZMP). Each day of flight data is compiled separately. The complete data set contains 107,671 high-altitude flights that pass at or above 20,000 ft at some point in their trajectory; this includes both general aviation and scheduled flights. A single day of traffic includes data for flights departing their origin on the stated day. Thus, a flight leaving from California late-night may appear inside ZMP within the 25th hour of the day.

A presence map of the traffic over the the center is presented in Figure 7 for June 14^{th}, 2007. Darker regions of the map indicate areas of high traffic; conversely, low traffic regions are lighter in color. Over the 28 days, variations in the traffic pattern exist due to weather and wind conditions, as well as origin-destination demand. However, qualitatively, the daily traffic pattern remains similar over the complete data set.

Minneapolis center is divided into 6 major areas according to function, traffic type, and volume. The area boundaries are illustrated in Figure 8. Each area is further subdivided into sectors; each sector spans a given area and altitude bound. Typically, the busier areas, by aircraft count, are located in the southern regions of the center, while the least busy areas are in the north. Accordingly, the sector divisions within each area are

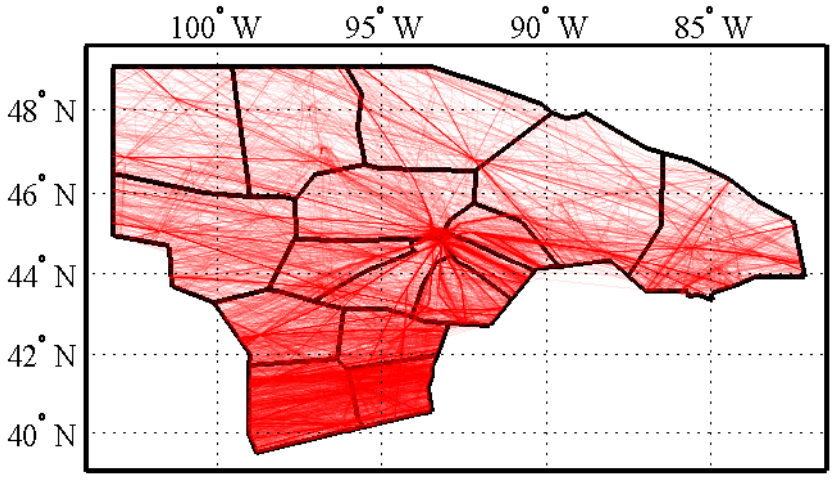


Figure 7: High-altitude traffic pattern for June 14^{th}, 2007 in Minneapolis Center.

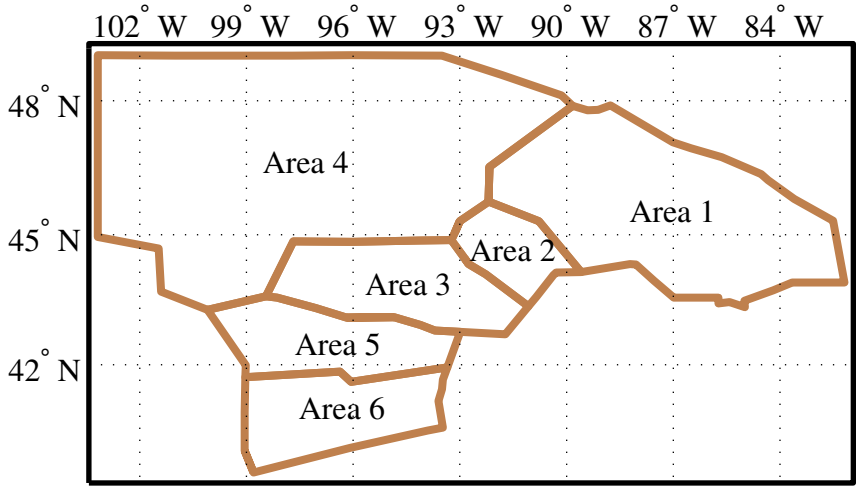


Figure 8: High-altitude boundaries for Areas in ZMP.

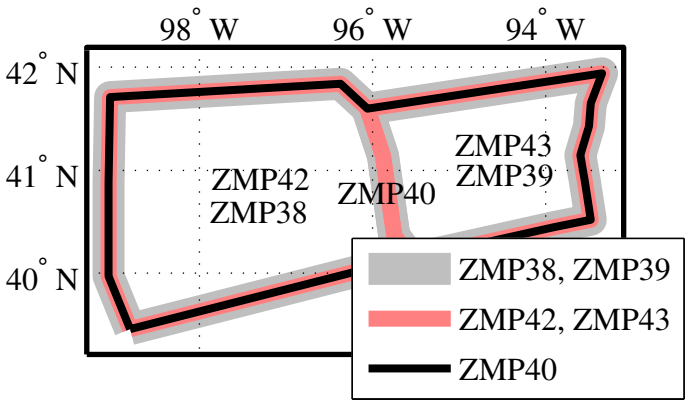


Figure 9: High-altitude sectors boundaries in Area 6 of ZMP.

different. For some sectors in Area 1 and Area 4, where traffic volumes tend to be light, an air traffic controller manages the airspace from the ground up, excluding any special airspaces (restricted, military, airport, etc.). In contrast, Area 6, contains highly stratified sectors; the high-altitude sector boundaries are illustrated in Figure 9. Two low altitude sectors cover the airspace from ground-level to just under FL240. High altitude airspace is divided by Sectors 38 and 39 controlling aircraft from FL240 to FL340, and Sectors 42 and 43 controlling aircraft from FL340 to FL390. Finally, the air traffic controller for Sector 40 is responsible for managing all aircraft at FL400 and above in Area 6. The vertical division of Area 6 is an example of the sectorization of en route centers that is reaching its useful limits.

Traffic within the center fluctuates over the course of a day and from day to day. The 25^{th}, 50^{th}, and 75^{th} percentiles for the daily aircraft traffic counts in the center are given in Figure 10. While keeping a relatively consistent shape, the demand does vary between days. Additionally, demand is seasonally varying, especially in the northern areas [36]. Accordingly, while the air traffic model presented in this chapter seeks to aid in modeling the conflict-detection and resolution process, it must be understood that the resulting model cannot exactly represent all days, as variation between days and within days exists.

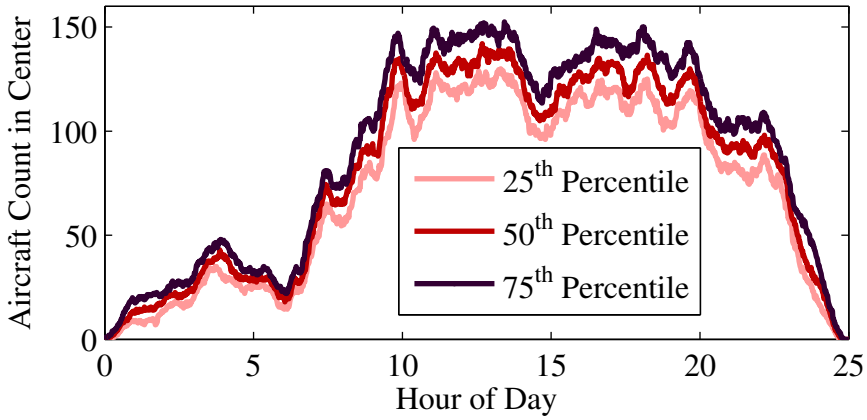


Figure 10: Diurnal aircraft counts within Minneapolis Center for the dates May, 21, 2007 through June 17th, 2007.

2.3 Sampling Procedure

To generate potential conflicts based on realistic air traffic patterns, flight data from a seed day is used to create 100 new scenarios of uncontrolled flight trajectories. A scenario of uncontrolled flight data approximates the traffic pattern of the seed day. The uncontrolled flight trajectories represent flight paths that aircraft would fly irrespective of other aircraft, and as if air traffic controllers provided no input. In generating the 100 traffic scenarios, flights from all days, excluding the seed day, are sampled according to the origin-destination pairs of the seed day to populate the flight trajectories. By sampling according to origin-destination pairs found in the seed day, the overall traffic pattern within the center is preserved. Preservation of other key metrics is demonstrated in Section 2.5. Based on the 100 traffic scenarios, the timespan over which the flights occur is reduced to increase the traffic intensity in the center.

The first step in the resampling process is to index all flights according to origin-destination pairs, and to identify the entry and exit times of the aircraft. In the PDARS data, the origin and destination information of the flight plan is described according to fixes (e.g. NASAL), airports (e.g. KATL), and latitude-longitude data (e.g. 4510N/10354W). Origin and destination pairs that are described by fixes and airports are indexed directly. For approximately 4% of the flights (4329 trajectories), the origin data is described by latitude and longitude. In these cases, the origins are manually clustered by location and

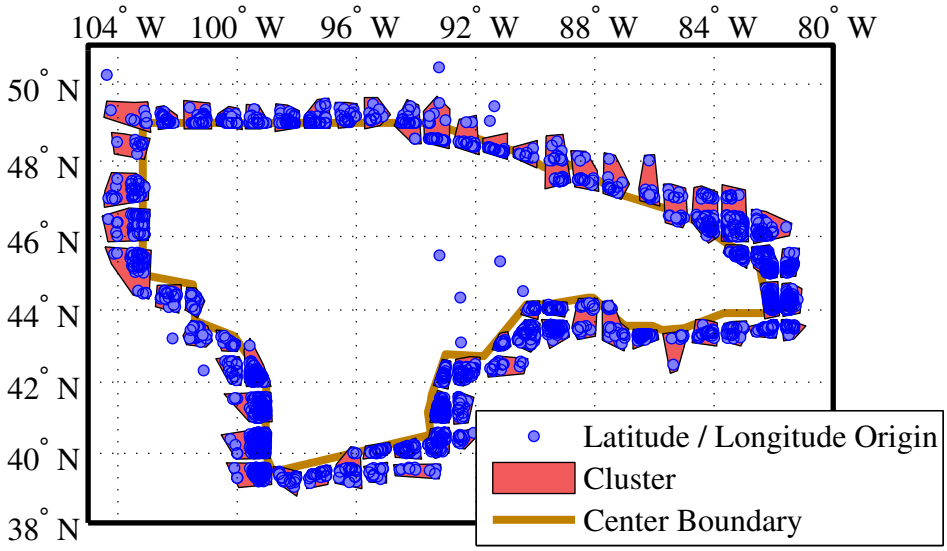


Figure 11: Spatial clustering of origins described by latitude and longitude.

indexed sequentially. Any latitude-longitude origins that are not clustered form an 'outlier' cluster. Figure 11 illustrates the complete set of origins described by latitude and longitude, and the manual cluster partitions.

Following clustering and indexing, a total of 11,014 origin-destination pairs are identified. Ranking the origin-destination pairs according to the number of occurrences, the frequency distribution, as well as the probability and cumulative distributions, are depicted in Figure 12. As an example, the most common origin-destination pair, ROBBY to KMSP, occurs 2442 times. The fix ROBBY is located outside and to the south of ZMP, near the border with Chicago Center; KMSP corresponds to Minneapolis St. Paul International Airport. According to the relationships illustrated in Figure 13, 224 origin-destination pairs repeat at least 100 times over 28 days and account for $\sim$ 52% of the traffic. Origin-destination pairs that occur less than 10 times are declared to be rare. Rare origin-destination pairs account for 15% of all flights, yet constitute 9639 of the 11014 pairs. The classification of rare origin-destination pairs is later used by the sampling procedure in creating new uncontrolled flight trajectories.

For a seed day with M flights passing through the center, the origin-destination pairs are given by the set of ordered pairs, $P_{od}^{s} = \{(O_1^s, D_1^s), \ldots, (O_M^s, D_M^s)\}$, and the arrival

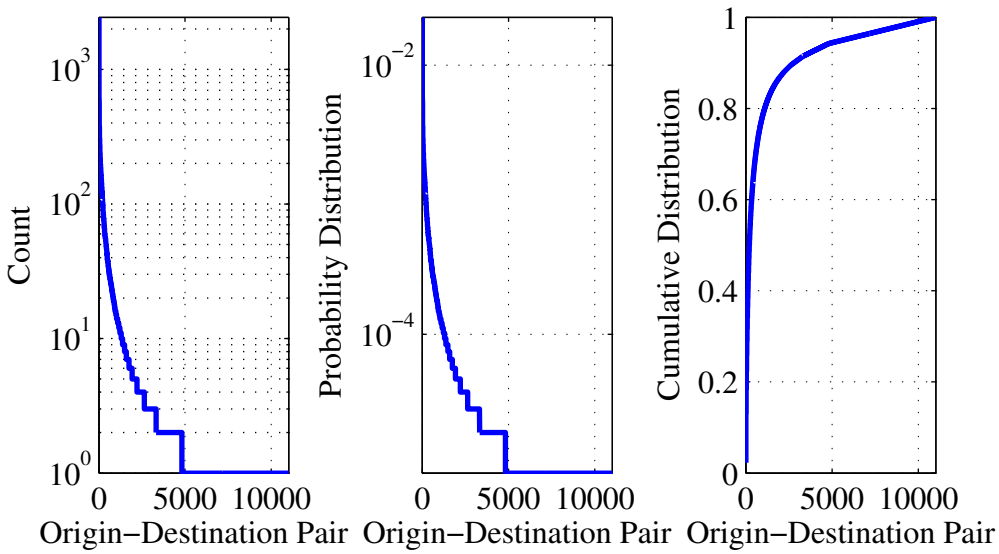


Figure 12: Distribution for origin-destination pairs for flights within Minneapolis Center.

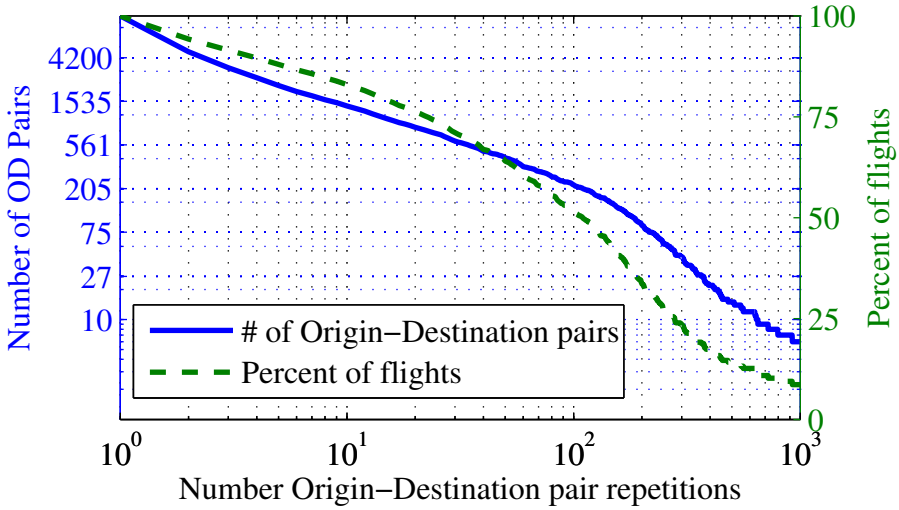


Figure 13: Clustering relationships between repeated origin-destination pairs and aircraft fights.

times into the center $T^s = \{T_1^s, \ldots, T_M^s\}$ for all flights. Here, the superscript 's' indicates that the variable relates to the seed day. For an arbitrary i^{th} day, the origin-destination pairs and arrival times P_{od}^i and T^i are defined in a similar manner. Furthermore, all flights are indexed for each day, such that $F_j^i = \{(x_1, y_1, z_1, t_1), \ldots, (x_k, y_k, z_k, t_k)\}$ corresponds to the space-time trajectory points for the j^{th} flight of the i^{th} day. The spatial values are given in longitude (degrees), latitude (degrees), and altitude (x100ft); time is measured in seconds from midnight (Coordinated Universal Time) on 1/1/1970, i.e. unix time. For each new traffic scenario, the re-sampling procedure assigns origin-destination pairs and arrival times to $P^{rs} = \{(O_1^{rs}, D_1^{rs}), \ldots, (O_M^{rs}, D_M^{rs})\}$ and $T^{rs} = \{T_1^{rs}, \ldots, T_M^{rs}\}$, along with flight trajectories to F^{rs}, thus, generating a resampled traffic model. The procedure for populating the resampled traffic scenarios is now detailed.

When resampling the seed day with uncontrolled flight trajectories to create each traffic scenario, the first step is to set $P^{rs} = P^s$, thereby preserving the traffic demand pattern within the center. Next, arrival times for each of the resampled trajectories are assigned according to the original arrival times of the seed aircraft, and the desired traffic intensity. For the resampled model, the arrival time for each aircraft includes a normal random term, $\sigma\mathcal{N}$, and is inversely scaled by I to increase the traffic intensity. That is,

$$T_j^{rs} = (T_j^s + \sigma\mathcal{N})/I, \tag{1}$$

By adding the random normal variable $\sigma\mathcal{N}$ to the original arrival times, T_j^s, arrival times are slightly perturbed. When $I = 1$, the scaling value implies that the traffic intensity of the resampled traffic case matches the seed day. For $I = 3$, the traffic case is three times the traffic intensity. Dividing by the intensity term I compresses the time frame over which traffic is simulated. Later in the thesis, analysis focuses on the number of potential conflicts in a traffic scenario. A multiplicative factor is required to scale conflict counts for the compressed traffic cases (i.e. $I > 1$). Scaling allows study of the compressed traffic cases over a standard 24 hour day. Otherwise, unequal day lengths prevent meaningful comparison between any two different traffic intensities. For the traffic scenarios generated in this thesis, $\sigma = 5$ minutes.

For each flight trajectory F_j^{rs} in F^{rs}, a new flight trajectory is sampled from all other flights with the same origin-destination pair given by (O_j^{rs}, D_j^{rs}). If an arbitrary origin-destination pair, (O_j^{rs}, D_j^{rs}), is classified as rare, then the origin-destination pair and the new trajectory is resampled from all origin-destination pairs that are also rare and that occur within 15 minutes of the original arrival time of the seed aircraft, T_j^s. By constraining rare flights to be sampled within a 30 minute window of the original arrival time, the approximate behavior of traffic around the same time of day is preserved. The time values for the space-time trajectory are reset, starting with the resampled arrival time T_j^{rs}. Thus, if the sampled flight originally contained trajectory points for times $\{t_1, \ldots, t_k\}$, then the new times for the resampled flight trajectories are $\{T_j^{rs}, t_2 - t_1 + T_j^{rs}, \ldots, t_k - t_1 + T_j^{rs}\}$. Through this sampling procedure, all flight trajectories are decorrelated and do not contain controller actions relative to one another. Each traffic scenario can then contain potential conflicts.

The last steps of the sampling procedure include parsing aircraft trajectories according to the sectors they pass through. First, each flight trajectory is interpolated at 1 second increments. Portions of a trajectory that pass through a given sector are isolated. The isolated trajectory is then projected into a local coordination frame using the standard gnomonic projection about the centroid of the sector. The projection generates a local coordination frame in units of NM for the X-Y axis. Also calculated are the arrival and exit times of the aircraft for the sector. The process is repeated for each sector and for all flights in the resampled traffic scenario. The final result is a set of trajectories that corresponds to a traffic scenario for a specified sector. Traffic scenarios for individual sectors are of particular interest because each airspace corresponds to a control volume managed by a single air traffic controller.

The complete resampling procedure provides a method for generating uncontrolled flight trajectories through each sector. Initialized with a large set of aircraft trajectories and origin destination data - selecting a seed day, traffic intensity and sector - the output of the resampling procedure is 100 traffic scenarios with aircraft trajectories passing through the specified sector.

2.4 Separation Detection

For the purposes of determining controller-action taskloads associated with the conflict-resolution process, relative aircraft distances are checked to determine if aircraft pairs come in close proximity. According to the relative proximity of aircraft pairs, action by the controller may be required. In this section, a mathematical description of the time and distance at which aircraft come in close proximity is provided. The proximity distances and times are calculated for all traffic scenarios and traffic intensities. Later chapters of this thesis make use the distances and times as estimated measurements for detecting potential conflicts.

Given a sector and traffic scenario, aircraft separation distances are calculated. Denote the trajectory of an arbitrary aircraft A_i in the local frame by the parametric functions $(x_i(t), y_i(t), z_i(t))$, representing the position and altitude of the aircraft in time. For two aircraft, A_i and A_j, the separation distance $D_{i,j}(t)$, at any point in time, is

$$D_{i,j}(t) = \begin{cases} \sqrt{(x_i(t) - x_j(t))^2 + (y_i(t) - y_j(t))^2} & \text{if } |z_i(t) - z_j(t)| < 1000 \\ \infty & \text{else.} \end{cases} \tag{2}$$

According to standard separation requirements for the en route environment, a conflict between two aircraft A_i and A_j is said to occur if the aircraft come within 5NM laterally, and 1000 ft vertically of each other, i.e. when $D_{i,j}(t) < 5\text{NM}$ for some value of t. Thus, if a conflict-detection probe indicates the potential for a conflict in the near-term (e.g. 5 minutes), then the air traffic controller should issue resolution commands to at least one aircraft to ensure a conflict does not occur. Otherwise, without resolution, conflict is likely.

Often times however, a more conservatively safe approach is taken in determining if a resolution maneuver is required. How a conflict-detection tool classifies a potential conflict between two aircraft depends on the certainty of trajectory prediction tools and their uncontrolled minimum miss-distance. The uncontrolled minimum-miss distance, $D_{i,j}^{miss}$, is defined as

$$D_{i,j}^{miss} = \min_t \ D_{i,j}(t) \tag{3}$$

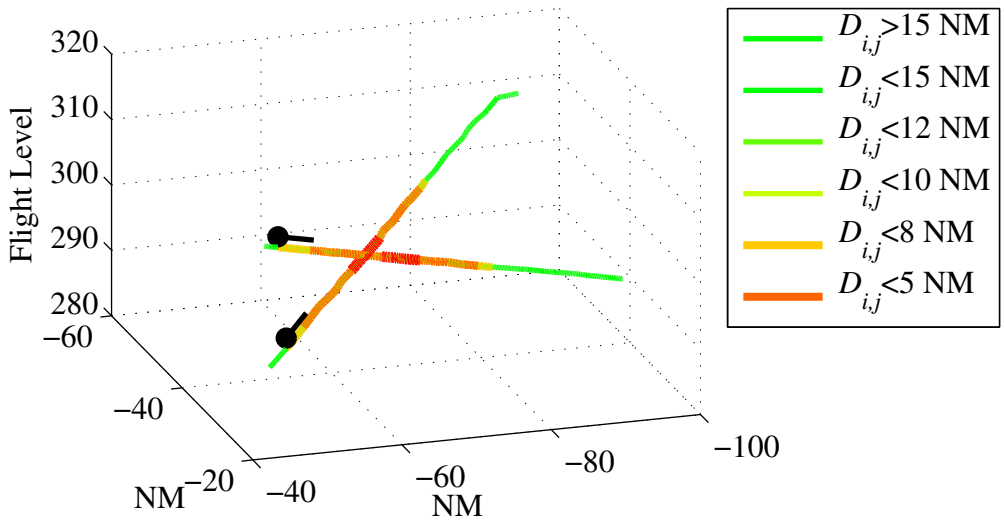


Figure 14: Example conflict between two aircraft.

prior to any action by the air traffic controller.

An example of an uncontrolled conflict is provided in Figure 14. The figure illustrates two aircraft breaking the 5NM mile separation requirement. In this case, one aircraft is ascending, while the other aircraft is in level flight. In the figure, as the two aircraft close in lateral and vertical distance, the trajectories lines grow thicker and become darker and more red in color. After passing each other, the thinner green segments of the trajectories indicate that the aircraft are a safe distance apart.

For the purposes of this thesis, the time of the uncontrolled minimum miss-distance is of interest. For the aircraft pair A_i and A_j, the critical time is denoted by $t^c_{i,j}$, and given by

$$t^c_{i,j} = \underset{t}{\operatorname{argmin}}\ D_{i,j}(t). \tag{4}$$

The critical times values are later used in Chapter 6 to indicate the predicted time of potential conflict between two aircraft when simulating conflict-detection and resolution algorithms.

Based on the mathematical descriptions above, the minimum miss-distances, $D^{miss}_{i,j}$, and the critical times, $t^c_{i,j}$, are calculated for all pairs of aircraft. For aircraft pairs with minimum miss-distances less than 15NM, qualitative and quantitative information pertaining to the

flights is saved (e.g. flight-phase, conflict-angles, groundspeed). The information is later used by conflict-detection and resolution models to determine to which aircraft maneuvers should be issued commands to resolve potential conflicts.

2.5 Application and Verification of the Sampling Procedure

The resampling procedure is applied using June 14, 2007 as the seed day. Over the 28 days of traffic, June 14^{th} corresponds to the day with the largest observed traffic load. Thus, the day provides a nominal lower bound on the maximum capacity of the center. In line with the goal of the Joint Office of Planning and Development, which envisions a tripling of the National Airspace System (NAS) capacity [49], increasing the traffic intensity for June 14^{th}, according to the resample procedure, provides a reasonable test-bed. For the remainder of the thesis the traffic scenarios of uncontrolled aircraft trajectories generated from the seed day are used for study. In this section, the center-wide traffic scenarios are tested to support the assumption that the resampling procedure yields traffic scenarios similar to the seed day. In particular, the spatial, aircraft model, and maximum altitude distributions are compared against the seed day using statistical methods.

Because the statistical tests are only concerned with the spatial, aircraft model, and altitude distributions, they only need to be applied to the traffic scenarios for a single traffic intensity. Each of the metrics is independent of the time scaling found in Equation 1, and as such, any statistical results for one traffic intensity holds for all traffic intensities.

2.5.1 Spatial Distribution of Aircraft

To verify that the 2D spatial distribution of the resampled uncontrolled trajectories match that of the seed day, the distributions are compared using a two-sample Kolmogorov-Smirnov test (KS test). The two-sample KS test is used to verify if two empirical distributions are sampled from the same distribution. The null hypothesis and alternative hypothesis are as follows:

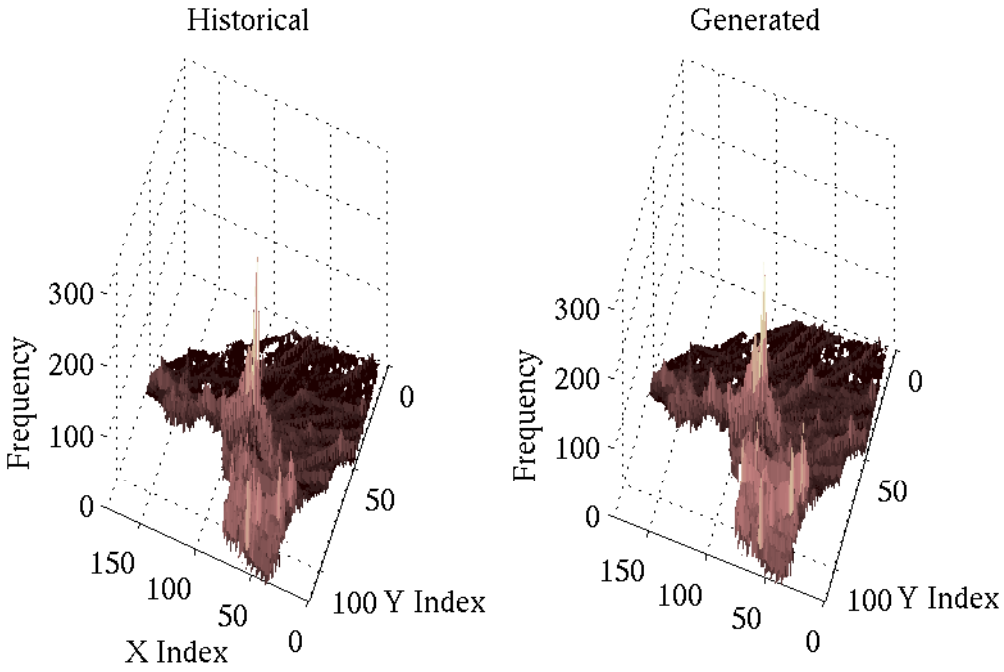


Figure 15: Spatial distribution of aircraft for June 14^{th}, 2007 and a generated traffic scenario.

H_o: Samples come from the same distribution.

$H_a =$: Samples come from the different distributions.

Rejection of the null hypothesis is quite meaningful, as it implies that the resampled traffic scenarios are not representative of the seed day. Instead of the more stringent $\alpha = .01$ value, setting $\alpha = .05$ takes a more conservative approach in determining if traffic scenarios are similar.

The two-sample KS test requires single-dimensional empirical distributions. As such, spatial aircraft counts over the center must be reshaped to be consistent with testing. The single-dimension empirical distribution for the seed day is generated by first overlaying a set of 5NM by 5NM bins over the airspace. For each aircraft that passes over an arbitrary bin b_i, the single-dimension sampled distribution includes the value i. For example, if 64 aircraft pass through the piece of airspace corresponding to bin, b_{13}, then the value 13 is included in the sampled empirical distribution 64 times.

For the historical June 14^{th} seed day, the empirical spatial distribution of aircraft within

the center is represented by the frequency map in Figure 15. Each bin index i is calculated according to the X and Y indicies generated by the 5NM by 5NM grid. As an example, the figure also includes the spatial distribution of aircraft for one of the traffic scenarios generated using the resampling procedure. Applying the KS test (with $\alpha = .05$) between the seed day and the 100 traffic scenarios, the null hypothesis is never rejected. Thus, it cannot be stated with certainty that the sampled spatial distributions come from different populations.

The KS test supports the assumption that the spatial traffic patterns of the resampled traffic scenarios are representative of the original seed day (see Section 2.1). However, despite accepting the null hypothesis for each case, any spatial correlation in the empirical distributions cannot be accounted for in the verification procedure due to reordering of the bins. Additionally, any correlations between aircraft flight trajectories cannot be verified by the KS test.

2.5.2 Maximum Altitude Distribution of Aircraft

The previous verification test checked that the spatial distribution of aircraft trajectories in the X-Y plane for the seed day and generated traffic scenarios are similar. To complete verification of the spatial distribution of aircraft, another study is required that considers the vertical direction. Division of the test into the planar and vertical directions is required because a complete comparison of the 3D spatial distributions is not meaningful; the sparseness of the trajectories ensures that the possibility of rejecting the null hypothesis is extremely unlikely. To consider the vertical distribution of aircraft, the maximum attained altitudes of the seed day and generated traffic scenarios are compared.

For each flight, the maximum altitude attained by each aircraft is extracted. Let $A^s_{max} = \{A^s_1, \ldots, A^s_M\}$ be the set of maximum altitudes for each aircraft in the seed day, where the maximum altitude of the j^{th} flight is given by $A^s_j = \max(z_i \mid (x_i, y_i, z_i, t_i) \in F^s_j)$. Similarly, let A^{rs}_{max} be defined for each of the resampled traffic scenarios. The distribution A^s_{max} for the seed day and A^{rs}_{max} for one of the generated traffic scenarios are illustrated in Figure 16. As shown in the figure, the distributions of maximum altitudes between the two

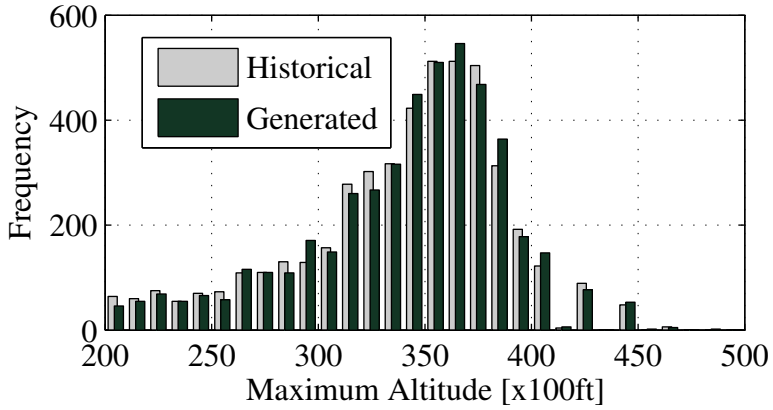


Figure 16: Distribution of maximum aircraft altitudes for June 14^{th}, 2007 and a generated traffic scenario.

data-sets are similar. When applying the KS test to verify or refute if the samples come from the same distribution, the null hypothesis is accepted. More so, the null hypothesis is accepted for the 100 generated scenarios.

2.5.3 Distribution of Aircraft Models

Because the resampling procedure is based on origin-destination pairs, there is no guarantee that the aircraft model types in the generated traffic scenarios are representative of the seed day. Thus, the two sample KS test is applied to verify or refute that the distribution of aircraft models are similar. Maintaining similar distributions of aircraft models is important because it is the variation in aircraft performance (often dependent on aircraft model) that results in differences in aircraft dynamics (especially climb rates). Furthermore, the decision-making process for conflict-resolution (for both automated systems and human air traffic controllers) must consider the dynamic capabilities of aircraft in implementing any resolution commands.

The distribution of the top 20 aircraft models found in the seed day are shown in Figure 17. The figure also includes the frequency of occurrence for the same 20 aircraft in one of the generated traffic scenarios. When applying the two-sample KS-test to the complete distribution of aircraft models, the null hypothesis is accepted; the empirical distribution of aircraft model types for the seed day and the generated traffic scenarios are similar. Matching aircraft model distributions is an example of how the resampling

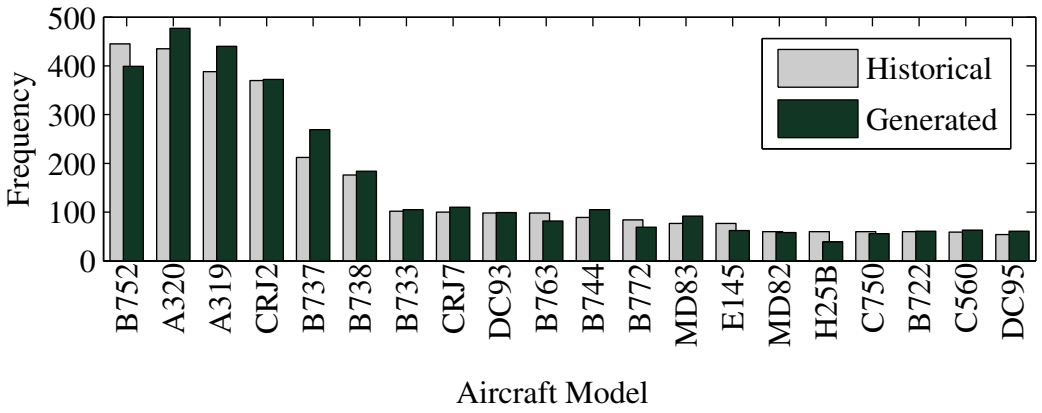


Figure 17: Distribution of aircraft models for June 14^{th}, 2007 and a generated traffic scenario.

procedure maintains specific properties despite the fact that the procedure only considers origin-destination pairs in creating new traffic scenarios.

2.6 Scaling for Equitable Comparison

As the traffic intensity increases, i.e. for $I > 1$, the length of each simulation day decreases inversely proportional to the intensity factor. A result of the compressed time scale is that aircraft are more closely spaced, and thus, the likelihood for potential conflicts increases. Problematic however, is that when considering the number of potential conflicts in each traffic scenario across traffic intensities, unequal lengths of day prevent fair comparison directly; the number of conflicts in an 8 hour day (3X intensity) is not equivalent to the number of conflicts in a 12 hour day (2X intensity) or 24 hours day (1X intensity). In this section, a scaling factor applied to the compressed days is proposed, thereby allowing equal comparison of conflict counts across all traffic intensities. Validity of a scaling factor is tested through a process that extrapolates conflict totals from subsampled traffic data. Additionally, in the process of verifying the scaling factor, the expected number of uncontrolled conflicts is demonstrated to increase quadratically with traffic intensity.

The primary metric of interest in this section is the mean total number of uncontrolled conflicts that occurs in a 24 hour day for each traffic intensity level within a sector. The labeling of an uncontrolled conflict follows the definition provided in Section 2.4; a conflict occurs when the minimum-miss distance between two aircraft, $D_{i,j}^{miss}(t)$, is less than 5NM

at some point in time. Uncontrolled conflict totals at the sector level, not the center level, are of particular interest; controllers are responsible for managing and separating aircraft in a single sector. While the mean conflict totals for compressed traffic intensities $I > 1$ can be extracted for the native shortened simulation day, a scaling factor is required to estimate the total number of conflicts over a 24 hour day. Accordingly, the following hypothesis is made:

Hypothesis 1. *Denoting $\bar{N}_C^{I,C}$ to be the average number of uncontrolled conflicts for the traffic intensity $I > 1$ for the compressed day, and $E[N_C^{I,24}]$ to be the expected number of conflicts for the traffic intensity $I > 1$ over a 24 hour period, then*

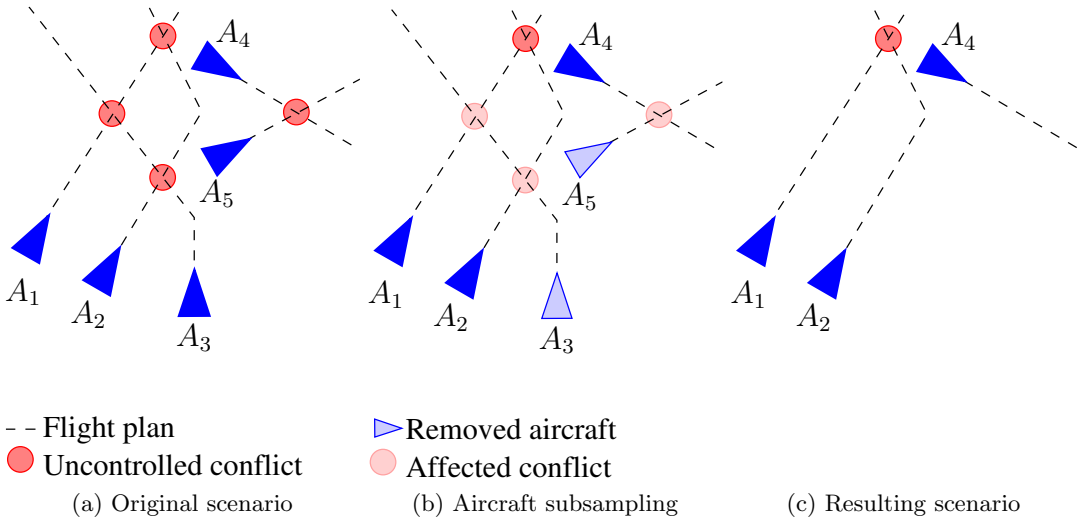
$$E[N_C^{I,24}] = \bar{N}_C^{I,C} \times I. \tag{5}$$

Further, it is hypothesized that $E[N_C^{I,24}]$ can be modeled by the equation

$$E[N_C^{I,24}] = k_1 I^2 + k_2 I. \tag{6}$$

To support the hypothesis above, a subsampling procedure is used to create traffic scenarios with traffic intensities $I < 1$. The procedure for generating subsampled traffic scenarios maintains a 24 hour day, thereby allowing equal comparison of conflict-counts between other subsampled intensities. Next, the average conflict totals of the subsampled traffic are extrapolated for higher intensity values (i.e. $I > 1$) to demonstrate that the expected number of conflicts in a 24 hour day for the compressed traffic scenarios matches the hypothesized values of $E[N_C^{I,24}]$.

The process for generating the subsampled traffic is detailed in Procedure 1, and depicted in Figure 18. In the procedure, the variables P^{rs}, F^{rs}, T^{rs} correspond to the origin-destination pairs, trajectories, and aircraft arrival times of a resampled traffic scenario with traffic intensity $I = 1$. These variables were previously generated using the process outlined in Section 2.3. For the subsampled traffic intensity $I_s < 1$, a new traffic scenario is created by randomly removing aircraft and trajectories from the original 1X resampled traffic scenario. Figure 18 is an example depiction of how the number of uncontrolled conflicts is reduced as aircraft are removed from a traffic scenario. The aircraft that remain are used to



(a) Original scenario (b) Aircraft subsampling (c) Resulting scenario

Figure 18: Illustrative example of the subsampling procedure.

populate the variables P^{rs,I_s}, F^{rs,I_s}, T^{rs,I_s}, which corresponding to the origin-destination pairs, trajectories, and aircraft arrival times of the subsampled traffic scenario.

Procedure 1 Subsampling Procedure

$N = card(P^{rs})$

$c_1 = 0, c_2 = 0$

while $c_1 <= N$ **do**

$c_1 = c_1 + 1$

if $rand < I_s$ **then**

$c_2 = c_2 + 1$

$P_{c_2}^{rs,I_s} = P_{c_1}^{rs}$

$F_{c_2}^{rs,I_s} = F_{c_1}^{rs}$

$T_{c_2}^{rs,I_s} = T_{c_1}^{rs}$

end if

end while

Application of the subsampling procedure to the 1X traffic scenarios results in subsampled traffic scenarios that occur over a 24 hour time period and contain a fraction of the

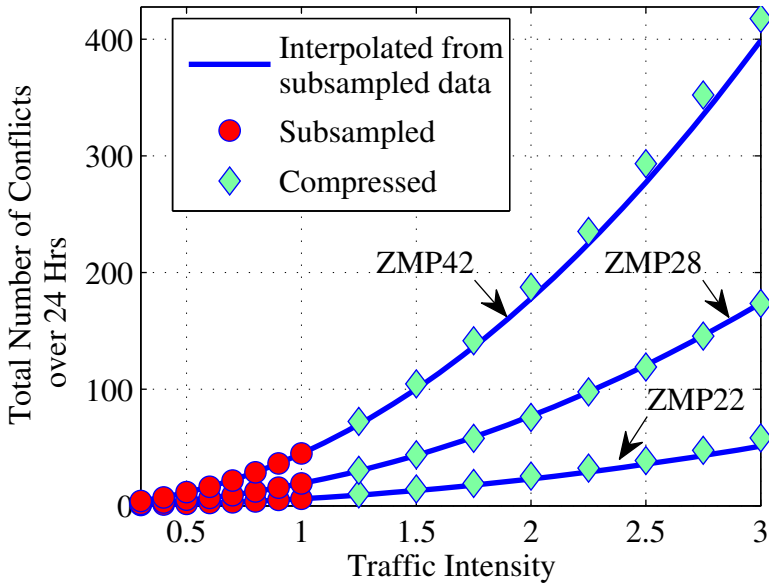


Figure 19: Extrapolated conflict totals based on sub-sampled traffic scenarios.

original flights. For a subsampled traffic intensity value of $I_s = .5$, the expected number of aircraft in each subsampled traffic scenario is half that in the original resampled 1X traffic scenario. Again, because the uncontrolled conflict totals in each case corresponds to a 24 hour time period, no scaling is required for direct comparison of uncontrolled conflict totals between intensities.

For traffic intensities $I = [.3, .4, .5, \ldots, 1]$, the average total number of uncontrolled conflicts occurring over the 24 hour simulation is illustrated in Figure 19 for sectors ZMP22, ZMP28, and ZMP42. The resampling process for creating the 1X traffic scenarios, and the subsampling procedure for generating lower traffic intensity scenarios, yield traffic scenarios occurring over 24 hour time periods.

A least-squares fit is applied to the 1X and subsampled average uncontrolled conflict totals for the 24 hour day, assuming a quadratic model of the form

$$E[N_C^{I,24}] = k_1 I^2 + k_2 I + k_3, \tag{7}$$

where I corresponds to the traffic intensity. At $I = 0$, the traffic intensity implies that there are no aircraft in the sector and hence no potential conflicts. Accordingly, the y-intercept value for the quadratic model is set to $k_3 = 0$. The remaining coefficient terms, k_1 and k_2,

are calculated by the least-squares solution

$$K = (A^T A)^{-1} A^T b \tag{8}$$

where,

$$A = \begin{bmatrix} .3^2 & .3 \\ .4^2 & .4 \\ \vdots & \vdots \\ 1 & 1 \end{bmatrix}, \quad b = \begin{bmatrix} E[N_C^{3,24}] \\ E[N_C^{4,24}] \\ \vdots \\ E[N_C^{1,24}] \end{bmatrix}, \quad K = \begin{bmatrix} k_1 & k_2 \end{bmatrix}.$$

The interpolated quadratic lines of best-fit for sectors ZMP22, ZMP28, and ZMP42 are shown in Figure 19 and extrapolated up to the 3X traffic intensity. The extrapolated best-fit lines only consider the 1X and subsampled traffic intensities they do not take into account the conflict counts for the compressed traffic scenarios.

The extrapolated quadratic line of best-fit provides a basis to verify that the 24 hour uncontrolled conflict totals can be estimated by scaling the compressed conflict totals (i.e. $E[N_C^{I,24}] = \bar{N}_C^{I,C} \times I$ for traffic intensities with $I > 1$). Figure 19 includes the estimated 24 hour uncontrolled conflict totals, $E[N_C^{I,24}]$, according to the hypothesized scaling for the sectors ZMP22, ZMP28, and ZMP42.

To assess the validity of the extrapolated expected results with the hypothesize of uncontrolled conflict totals, the coefficient of determination, R^2, of the model in Equation 5 is calculated. In the general case, with observed values $\hat{\theta}$, and modeled values θ, the R^2 value is given by

$$R^2 = 1 - \frac{\sum_i (\hat{\theta}_i - \theta_i)^2}{\sum_i (\theta_i - mean(\hat{\theta}))^2}. \tag{9}$$

An R^2 value close to 1 indicates a good fit. The R^2 values for the average 24 hour scaled uncontrolled conflict totals in each of the 22 sectors are provided in Figure 20 . In the calculations, the observed values, $\hat{\theta}$, are given by scaling the uncontrolled conflict totals $\bar{N}_C^{I,24}$ to $E[N_C^{I,24}]$ for the compressed traffic intensities according to Hypothesis 1. The predicted number of uncontrolled conflict totals, θ, is provided by the extrapolated quadratic line of best-fit from Equation 7 and Equation 8 using the sub-sampled traffic scenarios. For all sectors, the R^2 values approach 1. Therefore, the hypothesis that scaling the uncontrolled

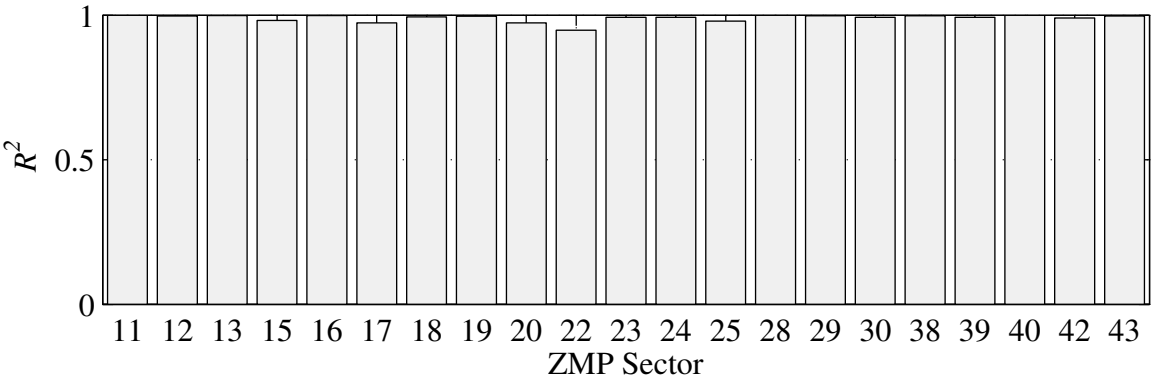


Figure 20: Coefficient of determination, R^2 for models based on sub-sampled traffic scenarios used to predict conflict totals as a function of traffic intensity in each sector.

conflict totals according to Equation 5 is believed to be valid. Even for the worst-case fit, corresponding to ZMP22, the R^2 value is .9478. Later studies in this thesis extend application of Hypothesis 1 to smaller time increments.

The quadratic model in Hypothesis 1 follows the relationship between traffic intensity and uncontrolled conflicts also described in the study found in [48]. Furthermore, the quadratic model is consistent with the analytical results in [85]. By maintaining similar conflict models, aspects of the compressed traffic model for generating potential conflicts at high-intensity traffic levels are verified.

2.7 Review

This chapter presents a modeling procedure to generate uncontrolled traffic scenarios based on a seed day. The goal of the modeling procedure is to create uncontrolled traffic scenarios that contains the same variations present in current air traffic operations, yet that match the original traffic pattern of the seed day without being overly specific. Hypothesis testing demonstrated that it is likely both these goals were achieved (i.e. the null hypothesis is never rejected) through the resampling of aircraft trajectories according to origin-destination pairs. Ideally, by matching the spatial distribution of aircraft trajectories, the ultimate goal of creating traffic scenarios that manifest in potential conflicts similar to real-world operations is achieved. Extending upon this idea, it is also desired that the generated potential conflicts match the behaviors and properties at current traffic levels,

and that they also model potential conflicts at higher traffic intensities than have yet to be observed in real-world operations. Key properties include: the number of aircraft involved, the location of the potential conflicts, the flight phase of each aircraft, etc. If these and other key properties are maintained, then it can be assumed that the overall experience of the air traffic controller is preserved according to the air traffic model. While many of these key properties cannot be verified, the overall modeling procedure suggests a satisfactory model.

While other models ([3, 10]) exist for generating uncontrolled aircraft trajectories, the model presented here is a first in creating traffic scenarios without flight plans, while attempting to model possible variations and uncertainties through a data-driven model. It should be noted, however, that there is limited scope in the relevance of the proposed procedure for generating traffic scenarios. Thus, the procedure may only be applicable to the study at hand involving the characterization of the conflict-detection and resolution process.

CHAPTER III

SECTOR CONFLICT ANALYSIS

To gain a better understanding of the traffic and the conflict event-process within a sector, an initial study of the uncontrolled flight plans generated in Chapter 2 is provided. The analysis in this chapter motivates the improvement of conflict-detection tools and the introduction of conflict-resolution systems in air traffic control. The analysis focuses on the number of potential conflicts and the rate at which potential conflicts occur for uncontrolled air traffic over a single day. Additionally, a taskload utilization analysis is performed based on the minimum required communication time between controllers and pilots. Ultimately, the analysis framework and the results motivate the inclusion of unified conflict-detection and resolution decision-support tools to aid air traffic controllers in managing and separating aircraft within an airspace.

In current en route operations, a conflict occurs between two aircraft when they break minimum separation requirements (5 NM laterally and 1000 ft vertically). When an air traffic controller identifies a pair of aircraft that has the potential to break minimum separation requirements, the event is labeled a potential conflict. For a potential conflict, the air traffic controller should take action to prevent its occurrence. Thus, the number of potential conflicts arising in an airspace is one measure of controller's effort . While there are a number of other tasks that controllers must attend to in current day operations - including, but not limited to, acknowledgements, hand-off, and clearances - the time spent resolving potential conflicts can become taxing. One study estimates that for each potential conflict, the controller spends 27.6 seconds identifying it, generating a resolution command, and communicating the command to the appropriate aircraft [20]. More so, this time does not include verification that the resolution command is properly implemented by the pilots and aircraft. As traffic levels increase, air traffic controllers will likely dedicate greater amounts of time and mental effort to handling potential conflicts. This chapter demonstrates that

if current practices are continued, then the effort required to separate aircraft will reach a point when air traffic controllers are no longer able to maintain situation awareness and ensure safety within the airspace.

To estimate the number of potential conflicts an air traffic controller resolves, the uncontrolled flight paths generated in the previous chapter are processed to identify uncontrolled potential conflicts. An uncontrolled potential conflict is said to occur if any two aircraft come within D_{sep}^{min} NM and 1000 ft of each other; unless stated otherwise, $D_{sep}^{min} = 9$ NM (Note, D_{sep}^{min} is not prescribed by the 5 NM en route regulation). The separation distance criteria of 9 NM, set by D_{sep}^{min}, and used in the labeling of potential conflicts, is an approximate value related to controller action and concern [6, 7, 40, 101]. This is not to say a dangerous situation will arise if aircraft come within 9 NM. Rather, the separation distance reflects the fact that controllers behave conservatively in how they space aircraft. The extra buffer distance beyond the 5 NM requirement is a result of cautious actions taken by air traffic controllers to ensure clear separation between aircraft. Thus, if aircraft come within D_{sep}^{min} of each other, the model in this chapter assumes the air traffic controller issues a resolution command. Accordingly, the parameter D_{sep}^{min} is henceforth referred to as the aircraft spacing distance.

According to the uncontrolled potential conflicts identified by the criteria above, an undirected graph model is formed to represent the conflict relationships between all aircraft. For a scenario representing a single day of traffic, the corresponding constructed conflict graph is denoted by $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Aircraft are represented by nodes in the vertex set $\mathcal{V} = \{n_1, \ldots, n_M\}$, where node n_i corresponds to aircraft A_i. Potential conflicts are indicated by an undirected edge in the edge set $\mathcal{E}$. That is, for a potential conflict between aircraft A_i and A_j, there is an edge $(n_i, n_j) \in \mathcal{E}$. An example conflict graph is provided in Figure 21. For this chapter, the graph is assumed to be static and complete, implying that accurate knowledge of the uncontrolled flight plans exists up to the $D_{sep}^{min} = 9$ NM aircraft spacing distance. It is then assumed that all uncontrolled potential conflicts coming within 9 NM are deemed to require a resolution command. By taking this approach, the graph represents all aircraft passing through the airspace and all potential conflicts an air traffic controller

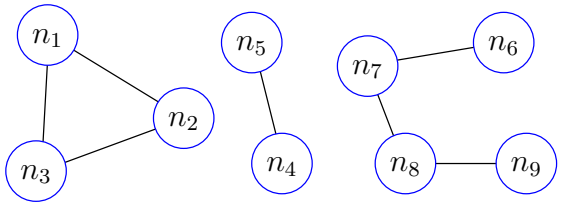


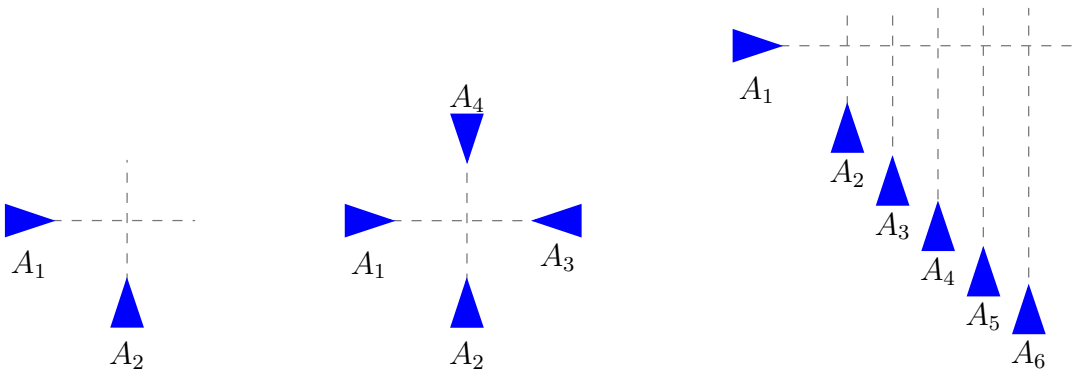
Figure 21: Example of conflict graph $\mathcal{G}$ with three conflict-clusters.

would attempt to resolve. A more formal model on the creation and definition of the conflict graphs is presented in Chapter 4 and further extended to the dynamic case.

In relation to the provided description of conflict graphs, the number of uncontrolled potential conflicts in a day, for an arbitrary airspace, is given by the total number of edges in the edge set $\mathcal{E}$.

Definition 1. *For a conflict graph, $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, representing traffic for a predefined time period and airspace according to the modeling description in Chapter 2, the number of uncontrolled potential conflicts is defined as $card(\mathcal{E})$.*

The analysis in this chapter also considers the complexity of potential conflicts. Here, the complexity of a cluster of potential conflicts is associated with the number of potential conflicts and the number of aircraft involved. Consider the examples of potential conflicts provided in Figure 22. For each case, let aircraft be traveling on the same flight-level at equal speeds. For the potential conflict between two aircraft illustrated in Figure 22a, the conflict complexity is said to be low (e.g. similar aircraft speeds and no emergency situations related to the aircraft). There is only one potential conflict and two aircraft. Hence, the conflict-resolution problem is straight forward to solve barring any other difficult circumstances outside the two aircraft (e.g. weather, airspace restrictions, or other aircraft). In fact, the safety of a number of pairwise conflict-resolution algorithms has been formally verified [26, 37, 61]. Figure 22b shows a set of 4 aircraft converging to a single point at the same time. The conflict complexity for this configuration is considered to be higher; a larger number of aircraft is involved, and all aircraft have potential to be in conflict with one another. However, the number of aircraft does not necessarily imply a high level of conflict complexity in solving the conflict-resolution problem. The example provided in



(a) Simple potential conflict with two aircraft.

(b) A high complexity set of potential conflicts with many aircraft.

(c) A low complexity set of potential conflicts with many aircraft.

Figure 22: Potential conflicts can be characterized by the number of aircraft involved and overall complexity of generating a resolution.

Figure 22c depicts a single aircraft that has potential conflicts with 5 other aircraft. While the number of aircraft in this example is greater than that in Figure 22b, with the same number of potential conflicts, the conflict complexity is considered less. Resolution of all potential conflicts is possible with a single aircraft maneuver. Provided sufficient time and free airspace exist, a single altitude change to aircraft A_1 is a possible resolution.

In general, the notion of conflict complexity is somewhat vague; while there is some relationship between the number of aircraft involved as well as the number of potential conflicts, other factors such as geometry and variation in aircraft properties (e.g. groundspeed or dynamic capabilities) are also important. Furthermore, conflict complexity is not only intrinsic to the circumstances of the potential conflicts but also depends on the conflict-resolution process. Poor decision making that creates secondary conflicts or overly complex dynamic models can make the conflict-resolution problem difficult to solve, and possibly intractable.

In this chapter, the assessment of conflict complexity is limited to a description of the number of uncontrolled conflicts; the rate at which the uncontrolled conflicts occur; and the average number of aircraft and potential conflicts involved in multi-aircraft conflicts. Based on the air traffic model described in Chapter 2, analysis indicates that as the traffic volume increases, the number of potential conflicts in an airspace and the average number

of potential conflicts per aircraft increase. Growing complexity supports the claim that air traffic controllers require additional support through the improvement of conflict-detection tools and the introduction of conflict-resolution decision-support tools.

Prior to the analysis, additional vocabulary is required to describe conflict complexity of an air traffic scenario. Making use of the definition provided in [41], the term *conflict clusters* is introduced.

Definition 2. *A conflict cluster is a sub-graph formed by the transitive closure of aircraft pairs in potential conflict. For example, if aircraft A_i and A_j have a potential conflict, and A_j and A_k have a potential conflict, then A_i, A_j, and A_k, form part of a conflict cluster, and define an associated sub-graph. [41]*

For an example of conflict clusters, consider the arbitrary graph, $\mathcal{G}$, illustrated in Figure 21 representing a simple traffic scenario. The conflict graph contains 3 conflict-clusters. Two conflict clusters, i.e. sub-graphs, represent aircraft that all have potential conflicts with each other. In the third sub-graph, although there is no potential conflict between aircraft A_6 and A_9, represented by nodes n_6 and n_9, they still form part of the same conflict cluster.

3.1 Assumptions

This chapter analyzes the number of uncontrolled conflicts occurring in an airspace. It also proposes a simple communication taskload model. The communication model is used to determine the likelihood that events in an airspace require more attention than the air traffic controller can provide within a given timeframe. Both these studies and their associated models utilize the following assumptions:

- Potential conflicts affect the air traffic controller uniformly. Further, the amount of effort required by the air traffic controller in spacing aircraft is uniform, regardless of the conflict configuration. For example, the difficulty of resolving a potential conflict between two cruising aircraft is assumed to be the same as managing a potential conflict between an ascending aircraft and a cruising aircraft.

- The number of potential conflicts is unaffected by the conflict-resolution process.
- Controller decisions to maneuver an aircraft are based on a deterministic model that considers lateral separations between aircraft at the same flight-level. The value of parameter D_{sep}^{min} is used to label potential conflicts as requiring resolution.
- Communication times between controllers and pilots for specific events are static, regardless of situational factors within the airspace. Furthermore, events (e.g. hand-off, potential conflicts) are handled by the controller in the order they occur.
- It is unsafe for air traffic controllers to sustain 10 minutes of continuous communication and control of aircraft without respite.

The second assumption is removed in future chapters in the thesis. This allows for comparison of advisory conflict-detection and resolution tools in reducing the effort related to the conflict-resoluiton process.

3.2 Uncontrolled Conflicts within a Sector

In current operations, an air traffic controller's priority is to maintain a safe airspace. Thus, if the number of potential conflicts an air traffic controller is presented with regularly exceeds his or her ability to resolve them in the required amount of time, then systematic change is required. Otherwise, if air navigation service providers such as the Federal Aviation Administration do not address the issue of overworked air traffic controllers, then unsafe operating environments will become increasingly prevalent.

Following the traffic model presented in Chapter 2, Hypothesis 1, and the labeling of uncontrolled conflicts utilizing the spacing parameter $D_{min}^{sep} = 9$ NM, analysis indicates that as traffic intensity increases, the number of uncontrolled conflicts increases as well. Figure 23 illustrates the expected number of uncontrolled conflicts for the 100 traffic scenarios representing June 14^{th} for different traffic intensities. Considering all sectors in Minneapolis Center (ZMP), ZMP42 contains the greatest average number of uncontrolled conflicts over the 24 hour time period at each traffic intensity. The sector is located in the southern region of the center and contains a portion of the airspace between FL350 and FL380. Because

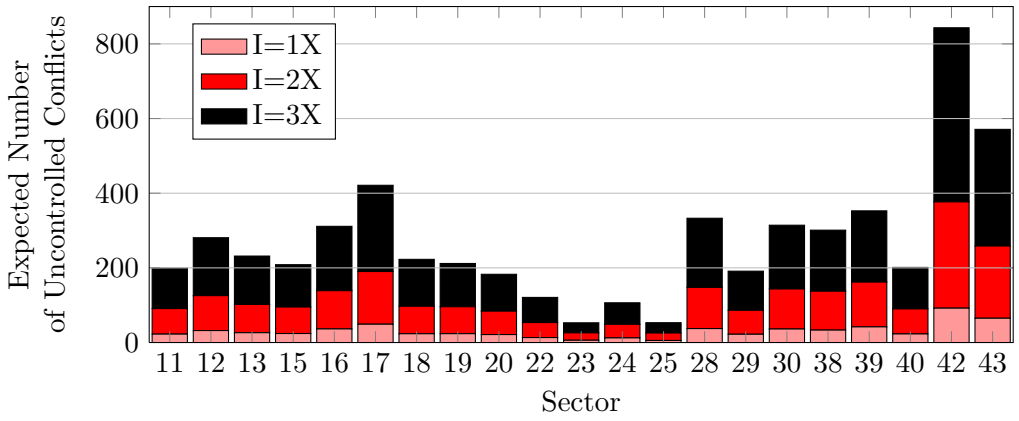
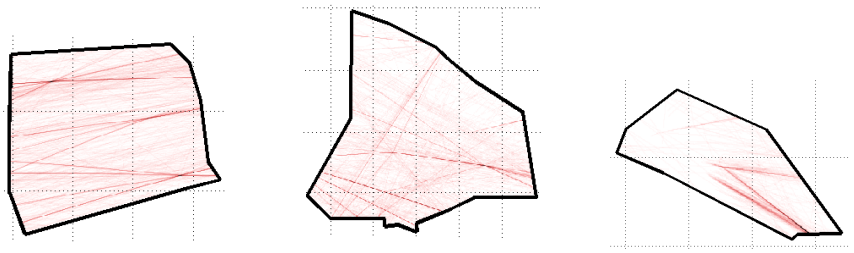


Figure 23: Conflict totals for sectors in Minneapolis Center over a 24hr time period when $D_{min}^{sep} = 9$ NM.



(a) ZMP42 (b) ZMP12 (c) ZMP16

Figure 24: Traffic Density Maps.

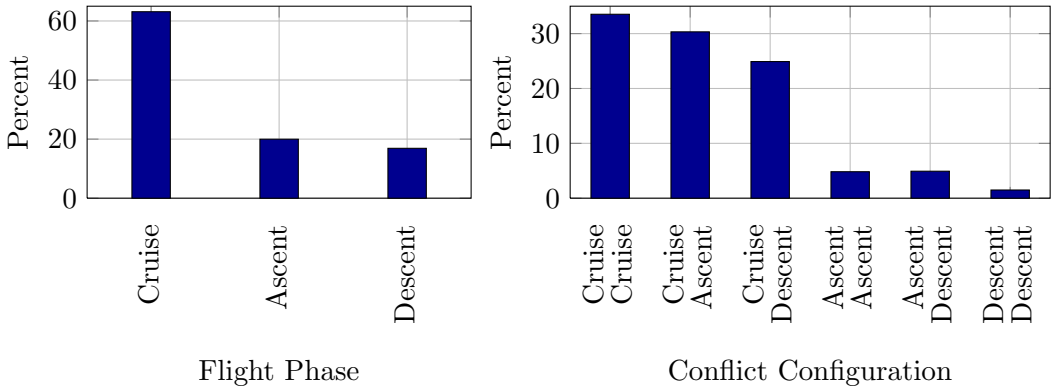
of the sector's spatial location and air traffic demand within the National Airspace, the majority of traffic within ZMP42 is high altitude cruising aircraft traveling eastbound or westbound. Figure 24a, a density plot of the traffic within the sector, illustrates the eastbound and westbound nature of the traffic flow, as well as the 2D spatial traffic distribution over the airspace. Two other airspaces of interest are sectors ZMP12 and ZMP16. These two sectors contain air traffic patterns representative of distinct classes of airspace. In the following analysis, the key differences between the sectors are highlighted.

The sector ZMP12 is similar to ZMP42, in that the majority of traffic is in its cruising phase. However, there are numerous distinctions between ZMP12 and ZMP42. First, air traffic within ZMP12 is more diverse in directionality, and according to the traffic density map in Figure 24b, air traffic is relatively sparse outside of primary traffic flows that cover an area 67% larger than ZMP42. Additionally, traffic in ZMP12 covers a larger range of flight

levels from FL240 and up and, as such, has a greater fraction of ascending and descending traffic ($\sim$ 37%) in comparison to ZMP42 ($\sim$ 7%). The difference in the proportions of ascending, descending, and cruising traffic is visible by flight phases distributions shown in Figure 25a and Figure 26a. While 92.5% of all traffic in ZMP42 in its cruising phase, the fraction of cruising aircraft in ZM12 is 63.1% of the traffic.

Unlike ZMP12 and ZMP42, sector ZMP16 is dominated by descending traffic ($\sim$ 73%). The sector is located near Minneapolis - St. Paul International Airport; as such, much of the traffic in ZMP16 corresponds to arrivals coming from the east that are headed towards the airport. Because of the dominant traffic pattern within the sector, as illustrated in Figure 24c, the majority of uncontrolled conflicts occur at the merge point near the center of the sector. Uncontrolled conflicts also occur along each of the arrival routes as aircraft are lining up to enter the airport terminal area. Accordingly, the qualitative characteristics of the uncontrolled conflicts in ZMP16 are different from those in ZMP12 and ZMP42. As shown in Figure 25, Figure 27, and Figure 26, the distribution of ascending, descending and cruising aircraft is different between the sectors, and likewise, so is the classification of uncontrolled conflicts. In sector ZMP12, most aircraft are found in their cruising phase, while a sizable remainder of flights are either ascending or descending at some point in the sector. Accordingly, most of the uncontrolled conflicts involve at least one cruising aircraft and smaller fractions involve two aircraft that contain a combination of ascending and descending flights. In contrast, for ZMP42, which contains an even greater percentage of cruising aircraft, approximately 60% of all uncontrolled conflicts involve two cruising aircraft. Furthermore, as shown in Figure 26b, less than 1% of the uncontrolled conflicts involve two aircraft that are ascending or descending, or a combination of the two.

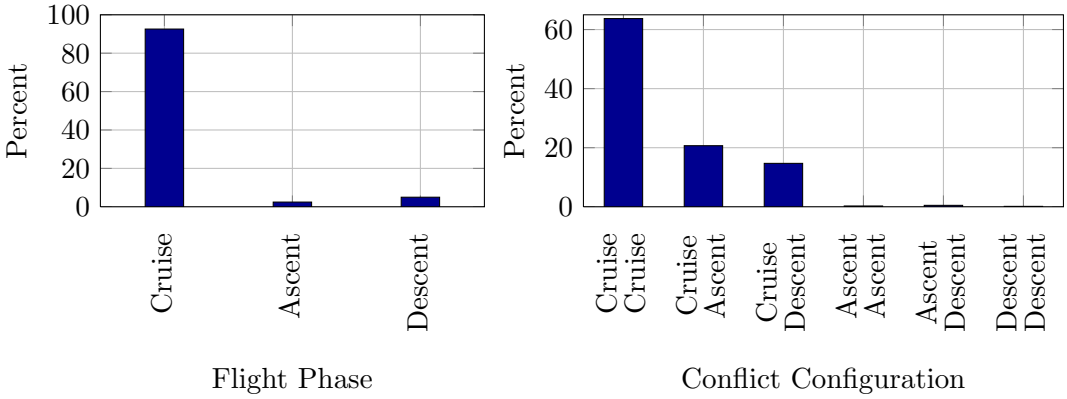
The location of uncontrolled conflicts is also a reflection of the dominant traffic patterns within the airspaces. As mentioned previously, and illustrated in Figure 28a, the majority of uncontrolled conflicts in ZMP16 occurs at the merge point in the direction of a major airport and along each of the descent routes. The traffic pattern in ZMP42 engenders uncontrolled conflicts along the dominant eastbound and westbound traffic routes. In contrast, as shown in Figure 28b, with a larger fraction of crossing traffic, a significant portion of



(a) Aircraft flight phase distribution

(b) Conflict configuration distribution

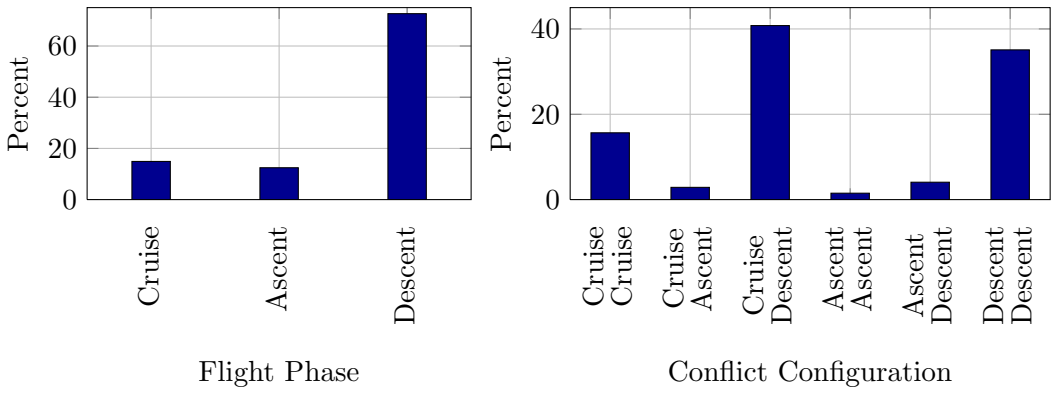
Figure 25: ZMP12 : Probability distribution of flight phases.



(a) Aircraft flight phase distribution

(b) Conflict configuration distribution

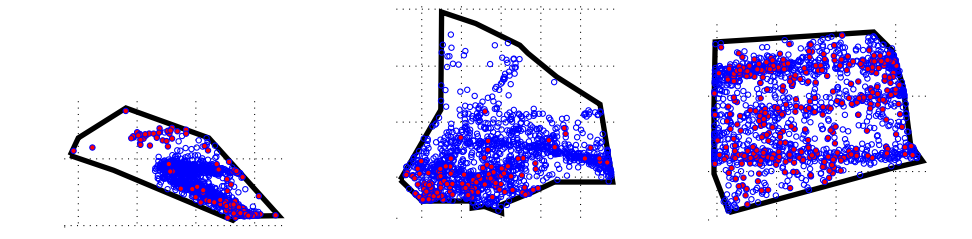
Figure 26: ZMP42 : Probability distribution of flight phases.



(a) Aircraft flight phase distribution

(b) Conflict configuration distribution

Figure 27: ZMP16 : Probability distribution of flight phases.

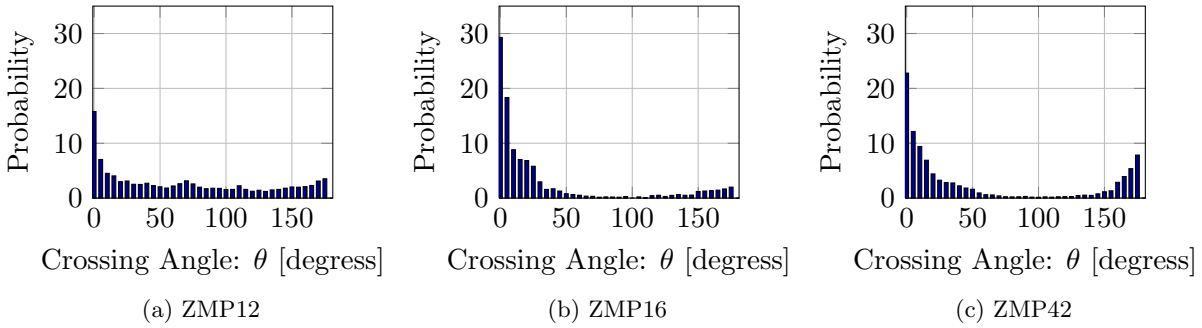


(a) ZMP16 (b) ZMP12 (c) ZMP42

Figure 28: Location of uncontrolled conflicts according to conflict configuration.

uncontrolled conflicts occurs at route crossings in ZMP12. This leaves large areas of the airspace void of potential conflicts. The difference in routing and conflict manifestation between ZMP12 and ZMP42 is also expressed through the distribution of aircraft crossing angles for the uncontrolled conflicts. As shown in Figure 29a and Figure 29c, the uncontrolled conflict crossing angles for ZMP12 are more evenly dispersed than for ZMP42. A larger fraction of uncontrolled conflicts occurs at high crossing angles (i.e., $45^o - 135^o$) for ZMP12. Using communication entropy, $H' = -\sum p_i \log(p_i)$, as a measure of diversity for the conflict angles, the three sectors in ascending order of entropy are ZMP16 ($H' = 2.49$), ZMP42 ($H' = 2.76$), and ZMP12 ($H' = 3.36$); higher values of entropy indicate greater diversity. Again, the distribution of the crossing angles in ZMP12 is a result of the mixture of east-west and north-south traffic, while ZMP42 is dominated by east-west traffic. The function that ZMP16 serves in providing a rough initial spacing of traffic landing at Minneapolis-St. Paul Airport is apparent in the the crossing angles as well. The largest bulk of conflict crossing angles is between $0^o - 30^o$, with a large peak around 0^o, which is a consequence of the merging and spacing operations along dominant traffic flows within the sector.

The variety in conflict configurations and flight phases demonstrates that for an advisory conflict-resolution tool to be useful, it must be able to handle a diversity of problems. Simple conflict-resolution algorithms that only consider level flight aircraft are inappropriate for virtually all sectors of airspace. Likewise, conflict-detection systems must be capable of



(a) ZMP12 (b) ZMP16 (c) ZMP42

Figure 29: Discrete empirical probability distribution of uncontrolled conflict crossing angles (5^o increments).

predicting potential conflicts over a wide range of traffic scenarios.

The total number of conflicts is not completely representative of the conflict-event process within each sector. In fact, the conflict-event process is non-homogeneous and varies according to traffic demand over the day. As shown in Figure 30, during the early morning and late night, the change in the cumulative number of uncontrolled conflicts is small in comparison to the time period between 10AM and 8PM. An initial assessment of the relationship between aircraft arrivals and conflict events is possible by comparing their cumulative counts. As shown in Figure 31, the hours between 10AM and 8PM account for the greatest traffic throughput in the sectors, which thereby drives the conflict-event process during the same time period. When analyzing the effort required by air traffic controllers to separate aircraft, these peak hours are most relevant; outside these times, air traffic controllers are unlikely to exceed their workload limits. This result is further supported by the communication analysis in Section 3.3.

Considering the 10 hour time window between 10AM and 8PM for each sector, it is asserted that the number of uncontrolled conflicts is heavily dictated by the traffic intensity and aircraft spacing distances. Applying and extending the assertion from Hypothesis 1 in Chapter 2 to the 10 hour time window, the following hypothesis is made:

Hypothesis 2. *The expected number of uncontrolled conflicts in a sector with traffic intensity I, spacing distance D_{sep}^{min}, and time window w, denoted by $E[N_C^{I,w}]$ can be estimated*

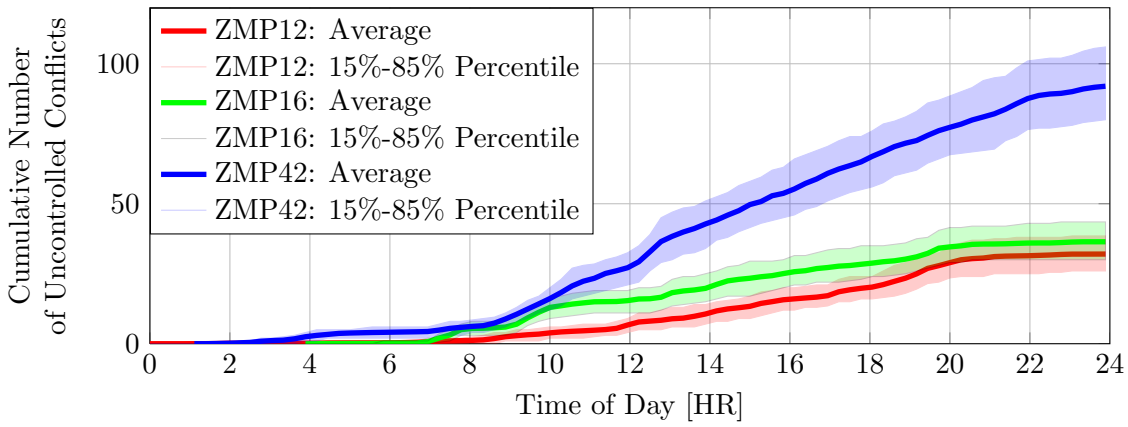


Figure 30: Cumulative number of uncontrolled conflicts for sectors ZMP12, ZMP17, and ZMP42, when $D_{min}^{sep} = 9$ NM.

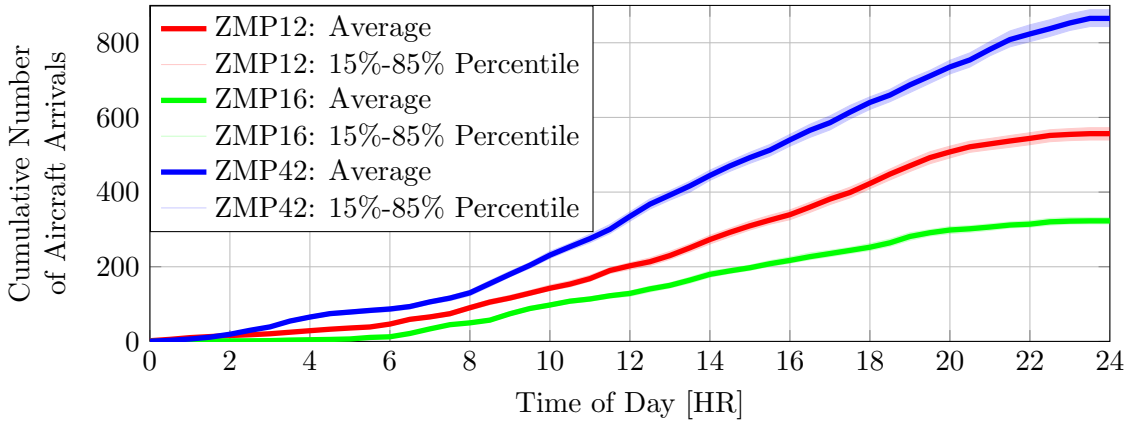


Figure 31: Cumulative number of aircraft arrivals for sectors ZMP12, ZMP17, and ZMP42.

Table 1: Best-fit coefficients for the uncontrolled conflict models.

Sector	c_1	c_2	c_3	c_4	R^2 (Complete)	R^2 (Simple)
ZMP12	0.0755	1.6688	0.4186	0.0062	.9997	.9925
ZMP16	0.1258	1.3224	0.2564	-0.0313	.9998	.9857
ZMP42	0.1722	5.8083	-0.8882	0.0403	.9999	.9953

by a function of the form

$$E[N_C^{I,w}] = c_1 I^2 (D_{sep}^{min})^2 + c_2 I^2 D_{sep}^{min} + c_3 I D_{sep}^{min} + c_4 I (D_{sep}^{min})^2. \tag{10}$$

The form of Equation 10 is a conjecture supported through testing. Further, through qualitatively analysis of real-world operations, there should be an understanding that if there is no traffic (i.e. $I = 0$) or there are no separation requirements (i.e. $D_{sep}^{min} = 0$ NM) then, inherently, there cannot be any conflicts. Thus, the variables I and D_{sep}^{min} should be found in each term so that if either is value 0, then $E[N_C^{I,w}] = 0$.

For the 10 hour time window considered, the coefficients (c_1, c_2, c_3, and c_4), calculated according to the least-squares best-fit of Equation 10, for sectors ZMP12, ZMP16, and ZMP42 are provided in Table 1. Also included in Table 1 are the R^2 for each line of best-fit when the complete model with all coefficients is used.

When considering the relative influence each coefficient plays, the term $I^2 D_{sep}^{min}$ is most important in determining the expected number of uncontrolled conflicts; a best-fit with only the term $I^2 D_{sep}^{min}$ results in an R^2 value of about .99 for each sector. The R^2 values for the simple model only considering the term $I^2 D_{sep}^{min}$ (s.t. $c_1 = 0$, $c_3 = 0$,$c_4 = 0$) is included in Table 1. The remaining terms, $I^2 (D_{sep}^{min})^2$, $I D_{sep}^{min}$ and $I (D_{sep}^{min})^2$, improve the model marginally. The major implication of such a result is that the number of uncontrolled conflicts grows quadratically as traffic intensity increases. Furthermore, the number of uncontrolled conflicts is linearly proportional to the aircraft spacing distance. Example lines of best-fit for ZMP42 are shown in Figure 32 across different values of D_{sep}^{min} for the simple model.

The relationship between traffic intensity and uncontrolled conflicts is supported by a similar study in [48]. Barring increased regulation of transport by government agencies, the traffic intensity is effectively an uncontrollable variable. Conversely, aircraft spacing

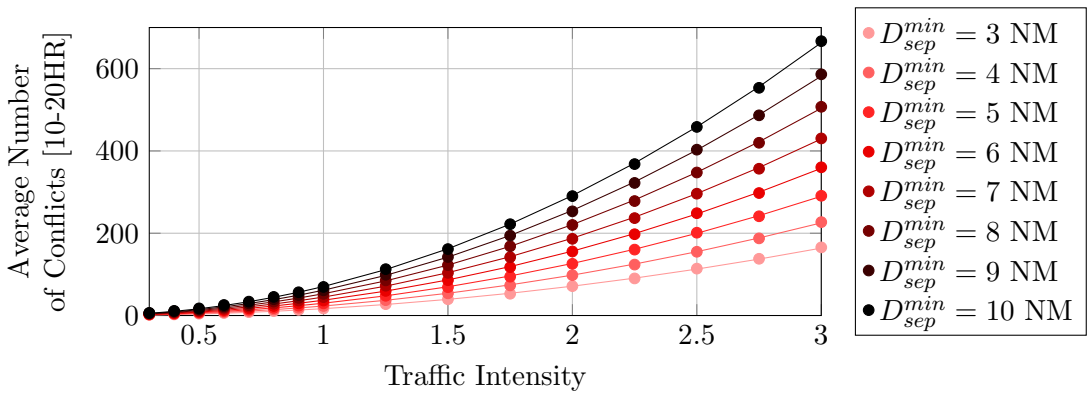
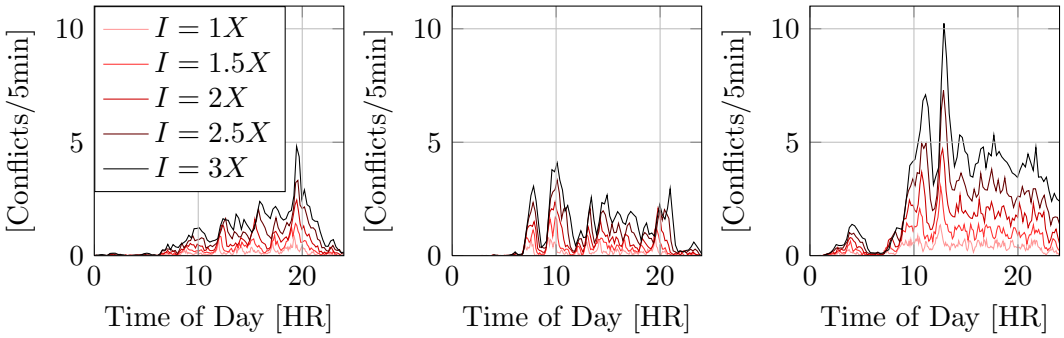


Figure 32: Simulated number of uncontrolled conflicts and best-fit lines according to traffic intensity and separation distance for ZMP42.

distances can be reduced through improvements in technology that promote changes in policy.

Key technological improvements to support the reduction of spacing distances include increased accuracy in trajectory prediction and aircraft position sensing. Reducing aircraft spacing provides the opportunity to reduce the number of labeled potential conflicts requiring resolution. Otherwise, if air traffic controllers continue to space aircraft at 9 NM, and perform their tasks according to current methods, then a tripling in traffic intensity will results in approximately an 850% increase in the expected uncontrolled conflict rate for ZMP42 (700% and 800% increase for ZMP12 and ZMP16, respectively).

To maintain a similar number of uncontrolled conflicts at the 3X traffic intensity when compared to current uncontrolled conflict counts, air traffic controllers in ZMP42 would be required to space aircraft at 1.2 NM (1.5 NM for both sectors ZMP12 and ZMP16). The possibility of spacing aircraft at these distances is hindered by a number of factors. First, conflict-resolution tools, human air traffic controllers included, are required to issue resolution commands minutes in advance of a potential conflict to ensure smooth and physically realizable trajectories for maneuvering aircraft. Otherwise, if resolution commands are issued seconds before a potential conflict, the range of solutions that are dynamically acceptable shrink significantly. Because of the required buffer time between issuing a resolution command and when the potential conflict may occur, the uncertainty in predicting aircraft trajectories must decrease significantly to allow for 1.2 NM separation. For current



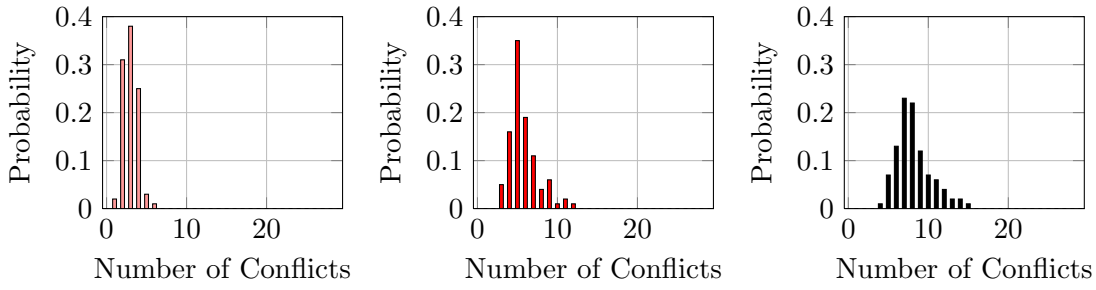
(a) ZMP12 (b) ZMP16 (c) ZMP42

Figure 33: Average rate of uncontrolled conflicts when $D_{sep}^{min} = 9$ NM.

systems, estimates indicate that uncertainty in the along track position of aircraft grows at a rate between .20-.25 NM/minute [47, 74, 103] at the 1 standard deviation level. Thus, over 5 minutes, the 1.2 NM separation distance is almost eclipsed by the uncertainty in predicting the aircraft positions. The difficulty of predicting aircraft trajectories is exacerbated when aircraft are turning or changing altitude.

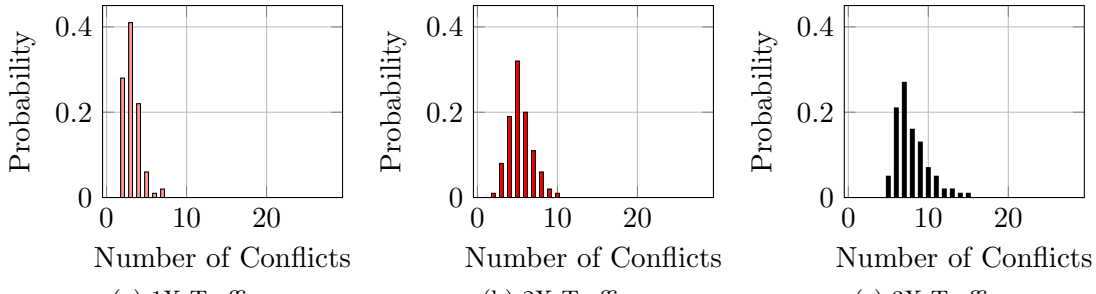
However, even if it were technically possible to predict aircraft trajectories minutes in advance, spacing aircraft at 1.2 NM is potentially unsafe due to the wake vortexes shedding off aircraft. These wake vortexes can interfere with the stability of other nearby aircraft [99]. As it stands, aircraft at airports are required to maintain spacing separations for arrival and departure procedures to prevent unsafe operating conditions due to wake turbulence [44]. Also preventing smaller aircraft spacings is the ever-present safety margin needed in case of aircraft or system failures. For example if aircraft position sensing systems go down (i.e. ADS-B, or radar) or aircraft lose controllability through the loss of a rudder, ailerons, or engine, then small separation distances hinder the ability of controllers and nearby pilots to respond in time to prevent a potentially catastrophic situation in the case of converging trajectories.

Assuming that the number of uncontrolled conflicts constitutes a significant portion of the effort required by air traffic controllers in managing an airspace, then the rate that uncontrolled conflicts occur is of potential concern as traffic intensity increases. Shown in Figure 33, for sectors ZMP12 and ZMP16, the rate of uncontrolled conflicts remains below



(a) 1X Traffic (b) 2X Traffic (c) 3X Traffic

Figure 34: Cumulative distribution of the number of uncontrolled conflicts for the busiest 5 minute period of the day when $D_{sep}^{min} = 9$ NM for ZMP12.

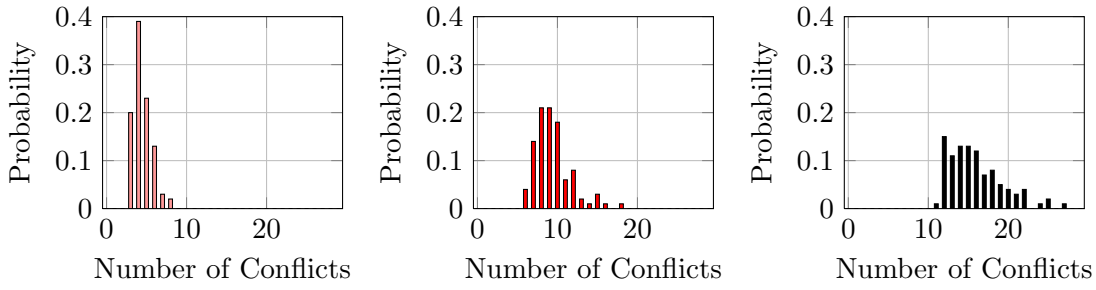


(a) 1X Traffic (b) 2X Traffic (c) 3X Traffic

Figure 35: Cumulative distribution of the number of uncontrolled conflicts for the busiest 5 minute period of the day when $D_{sep}^{min} = 9$ NM for ZMP16.

5 per 5 minute time period. For ZMP42, the average uncontrolled conflict rate reaches an average maximum of 10 conflicts per 5 minutes at the 3X traffic intensity. Assuming that air traffic controllers require 30 seconds to resolve each potential conflict, and little follow-up or conformance checking is required, then the average conflict rates are still within required limits to provide sufficient time for air traffic controllers to space aircraft. Further, when one considers that air traffic controllers often employ quick and simple resolution commands during heavy traffic loads [52], thereby reducing the time required to resolve each potential conflict, then even at 3X traffic intensity, the average conflict rates may appear to be manageable by air traffic controllers, when excluding other tasks (aircraft acknowledgements, hand-off, clearances, etc.).

Perhaps more telling to future concerns is the distribution of the number of uncontrolled conflicts within the busiest 5 minute time period over each of the 100 traffic scenarios. This distribution provides information about the conflict-resolution taskload during the busiest



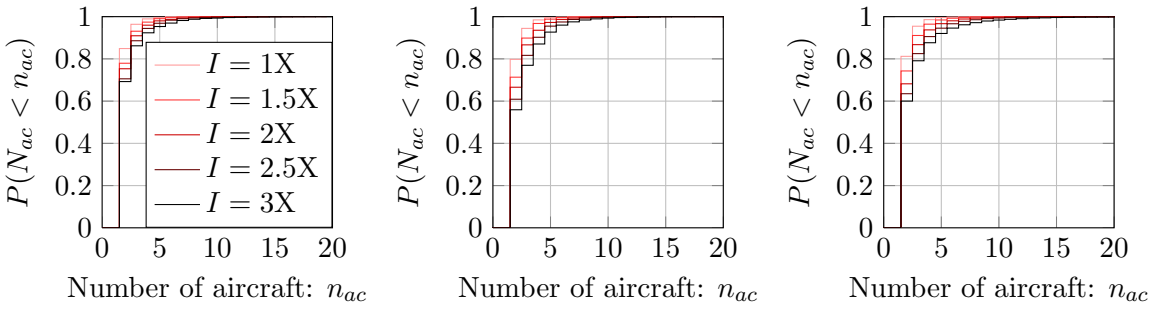
(a) 1X Traffic (b) 2X Traffic (c) 3X Traffic

Figure 36: Distribution of the number of uncontrolled conflicts for the busiest 5 minute period of the day when $D_{sep}^{min} = 9$ NM for ZMP42.

time period of a given day. For sectors ZMP12 and ZMP16, as the traffic intensities increases the mean and variance of the number of uncontrolled conflicts in the busiest 5 minutes increase as well. These results are illustrated in Figure 34 and Figure 35 and are indicated by the rightward shifting and increased width of the probability distribution functions.

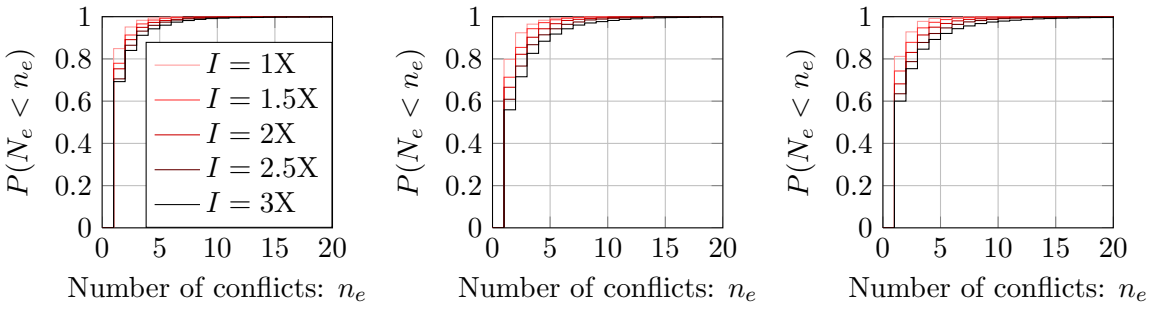
For both sectors, at the 3X traffic intensity, there is a measurable probability that 10 or more uncontrolled conflicts will occur during the 5 minute period. Because of the air traffic patterns within the sectors that bring about merging and spacing operations, the large fraction of ascending and descending traffic, and the high angles of intersection, the complexity and controller workload of resolving such conflicts becomes increasingly difficult for the unaided air traffic controller [58]. Assuming an ability to resolve each conflict in 30 seconds, then the case of encountering 10 conflicts in 5 minutes leaves no time for the air traffic controller to complete any other tasks - even those that maintain situational awareness. Taken together, not only is the number of potential conflicts large, but so is the difficulty of resolving each one.

Figure 36 shows distribution of the number of uncontrolled conflicts in ZMP42 for the 1X, 2X, and 3X traffic intensities. The number of uncontrolled conflicts in ZMP42 at 3X traffic intensity is quite troubling. For approximately 20% of the traffic scenarios, during the busiest 5 minute time period, 20 or more uncontrolled conflicts occur. If a scenario like this manifested in real-world operations, then the air traffic controller would be required to issue a resolution command, on average, every 15 seconds for a sustained period of time. Such



(a) ZMP12 (b) ZMP16 (c) ZMP42

Figure 37: Distribution of the number of aircraft invovled in each conflict cluster as traffic intensity increases for $D_{sep}^{min} = 9$ NM for ZMP42.



(a) ZMP12 (b) ZMP16 (c) ZMP42

Figure 38: Distribution of the number of edges in each conflict cluster as traffic intensity increases for $D_{sep}^{min} = 9$ NM for ZMP42.

a sustained taskload would probably be considered inconceivable by air traffic controllers. Even at the 2X traffic intensity, the number of uncontrolled conflicts in ZMP42 is quite large. In fact, the distribution of uncontrolled conflicts is similar to ZMP12 and ZMP16 at the 3X traffic intensity. This result indicates that if a uniform growth in air traffic demand occurs, then ZMP42 will be impacted much earlier than other airspaces.

As traffic intensity increases, not only will the number and rate of potential conflicts grow, but so will the complexity. Analysis of the traffic scenarios for each sector indicates that with growing traffic intensity, the number of aircraft and number of uncontrolled conflicts associated with each conflict cluster increases. The sampled cumulative distributions for both measures are provided in Figure 37 and Figure 38. The sampled distributions for the sectors show that both the mean and variance for both measures of complexity

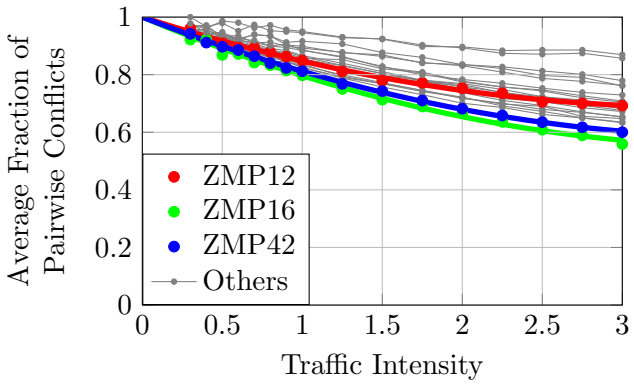
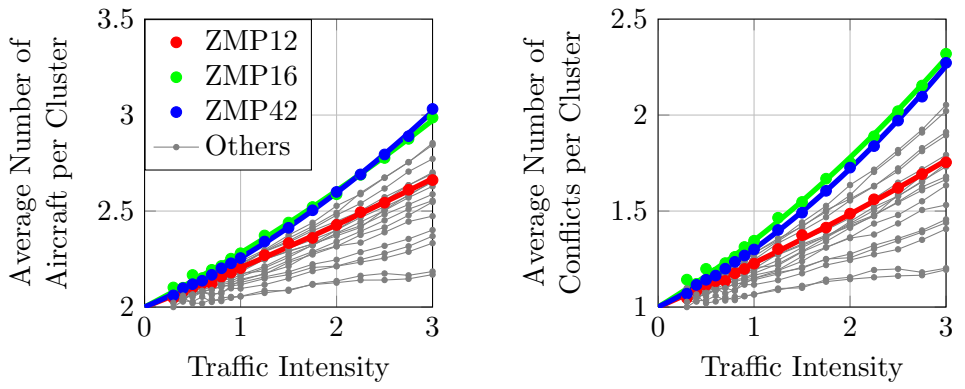


Figure 39: Fraction of pairwise uncontrolled conflicts.

(number of aircraft and number of conflicts) increase with traffic intensity (indicated by the movement of the cumulative probability distributions towards larger values). From the sampled distributions, more concise information can be extracted, such as the fraction of conflict clusters that only include two aircraft (i.e. strictly pairwise conflicts). As shown in Figure 39, as traffic intensity grows, the fraction of pairwise conflicts decreases. Such a result implies that with multiplying traffic intensity, the ability of humans and conflict-resolution tools to remain applicable while still using pairwise decision-making strategies becomes increasingly limited. Additionally, with larger conflict clusters, the secondary effects of a single resolution maneuver become increasingly important. Conflict-resolution should consider more than just the two aircraft that are part of the immediately pending potential conflict. Other aircraft that form part of the same conflict cluster can be affected.

Despite the growth in conflict complexity with regards to the number of aircraft and number of conflicts involved, there are indications that conflict complexity growth is only weakly "quadratic," especially when compared to the growth rate of uncontrolled conflicts. That is, while the total number of uncontrolled conflicts in a sector grows quadratically, the conflict complexity for each cluster grows at a much slower rate. Consider Figure 40a and Figure 40b, which depict the average number of aircraft and average number of uncontrolled conflicts per cluster. As traffic intensity increases, the growth in the number of aircraft and uncontrolled conflicts is approximately linear. More interesting, is that when plotting the number of aircraft per cluster versus the number of conflicts, their relationship is linear, as



(a) Average number of aircraft per conflict cluster, with quadric line of best fit.

(b) Average number of conflicts per conflict cluster, with quadric line of best fit.

Figure 40: Average growth of conflict clusters as intensity increases.

Table 2: Coefficients and R^2 values for least-squares lines of best-fit considering the ratio of conflict complexity.

Sector	Slope	R^2
ZMP12	1.135	.9997
ZMP16	1.300	.9977
ZMP42	1.218	.9995

shown in Figure 41. Table 2 contains the slope of the least-squares linear fit, along with the coefficient of determination value, R^2, associated with each linear fit. The coefficient of determination values indicates a strong fit with the linear model for the three sectors.

The relative rate of growth between the two complexity measures, i.e. the slopes provided in Table 2, is of particular interest because it provides a normalized measure related to the amount of effort required to deconflict clusters as the cluster sizes grow. Consider the two conflict clusters illustrated in Figure 42. In order to completely resolve all potential conflicts, the first conflict cluster requires at least 2 maneuvers. Meanwhile, the second conflict cluster can be resolved with a single resolution command issued to the aircraft A_5. The first conflict cluster corresponds to a special type of graph referred to as a complete graph; complete graphs have edges connecting each node. For a complete graph with n nodes, the total number of edges is $n(n-1)/2$ [72], and accordingly, requires $n(n-1)/2-1$ resolution commands to prevent all potential conflicts. For the conflict-resolution problem requiring

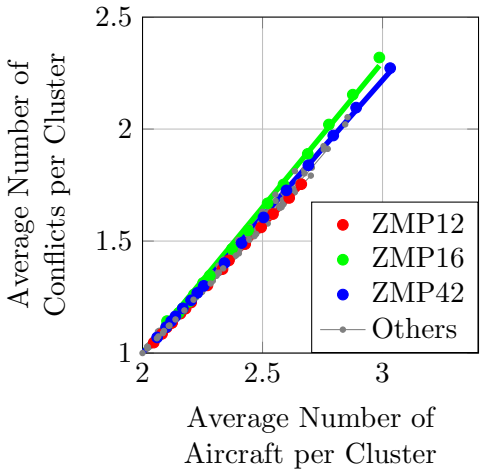


Figure 41: Number of expected conflicts versus number of expected aircraft in each cluster.

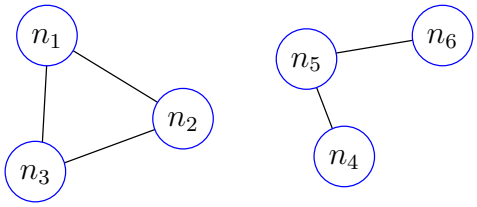


Figure 42: Two conflict clusters of differing conflict complexity.

that all potential conflicts are resolved (i.e. edges removed from the graphs), conflict clusters associated with complete graphs require equal or more resolution commands than any other type of conflict cluster with the same number of nodes. Therefore, because the lines of best fit in Figure 41 grow linearly, and not according to n^2, there is indication that the conflict clusters are not exclusively complete graphs (outside of pairwise conflicts). In the case where a non-complete conflict graph is present, a conflict-resolution program might be able to appropriately select which aircraft to maneuver in order to reduce to the number of commands used to space aircraft. For example, maneuvering aircraft A_5, instead of both A_4 and A_6, resolves all potential conflicts. The selection of which aircraft to maneuver to reduce the conflict-resolution taskload is considered beginning in Chapter 5.

The linear fit between between the number of aircraft and the number of conflicts dictates that on average, every additional aircraft in a conflict cluster is associated with just over one potential conflict; a larger slope signifies faster complexity growth. For the three sectors considered, the conflict complexity ratio of ZMP16 is the greatest, followed by

ZMP42, and ZMP12.

3.3 Controller Communication and Control Time

Despite the prior section's emphasis on the conflict counts as a measure of difficulty, a more appropriate estimate of controller overload is required. A better metric is needed to determine when controllers encounter scenarios that are beyond their capabilities. The process described here in detecting unmanageable traffic levels is based on a taskload and communications model that is associated with the control process between air traffic controllers and aircraft.

As it stands, the capacity to properly manage and separate air traffic directly depends on the controller workload [11]. Unfortunately, controller workload is difficult to measure quantitatively and depends on each individual controller's capability and perception. In current operations, instead of relying on controller workload as a measure for sector capacities, the Federal Aviation Administration has relied on a simple proxy: the number of aircraft present in a sector. The limit on this value is established by the Monitor Alert Parameter (MAP). If aircraft counts are within the MAP value, then it is assumed that traffic conditions are within the controller's abilities. However, MAP values do not accurately represent sector capacity - and often times lead to congestion, or conversely, under-utilization of the airspace because they does not accurately address performance limits.

Radio communication time has been considered an objective metric to evaluate controller workload while managing traffic. A series of experiments have concluded that realistic radio activities can be used to provide objective measures of workload [14, 21, 87]. Additionally, other studies have demonstrated the high correlation between communication duration and controller workload, thereby effectively validating communication time as another workload measure [63, 77]. By its very nature, communication time can be related to bandwidth limitations within the human-in-the-loop control system. Because communication is used in both the management and control of aircraft, if events within the airspace require greater amounts of communication than time permits, then some fundamental limit

has been reached. Accordingly, this section, and the proposed communication model, studies how often communication requirements exceed reasonable capabilities of the air traffic controller.

In current operations, each aircraft passing through a sector is communicated with at least twice by the managing air traffic controller: once to acknowledge the aircraft as it enters the sector and again when the aircraft is handed-off to the next sector. Another prevalent communication type typically occurs when an airspace is congested and there is the potential for conflict. Provided sufficient concern by air traffic controllers exists that a pair of aircraft might conflict, then a resolution command is issued. In the case of a potential conflict, air traffic controllers must determine safe routes for all aircraft and communicate them to each pilot. For this process to occur in a safe manner, there must also be sufficient time for the controller to gain situational awareness and to monitor conformance of the resolution commands. Finally (as part of the current system of clearance based control, in which requests are made by pilots, and verified for safety by the controller), any request for changes in heading, speed or altitude requires communication. The most common of these pilot requests is for altitude changes, either to ascend or descend in flight-level.

Using the 100 traffic scenarios created in Chapter 2, an event process for each scenario is generated. The set of events considered as part of each scenario's event process includes entering and exiting aircraft, as well as the detection and resolution of potential conflicts. Further, entering aircraft are classified according to whether or not they require clearances for altitude changes. Each of the events gives rise to a response and communication sequence between the controller and the pilots. By summing up the communication time, a measure of the required control effort is generated. However, in this case, because some communication events are not accounted for (e.g. read-backs, and weather and congestion advisories), the estimated communication times represent a lower bound, when assuming static communication times.

The event process for a traffic scenario is represented by the function $f_{event}(t)$, which contains a sequence of Dirac delta functions. An example depiction of $f_{event}(t)$ is illustrated in Figure 43. The magnitude of each Dirac delta function corresponds to the amount of

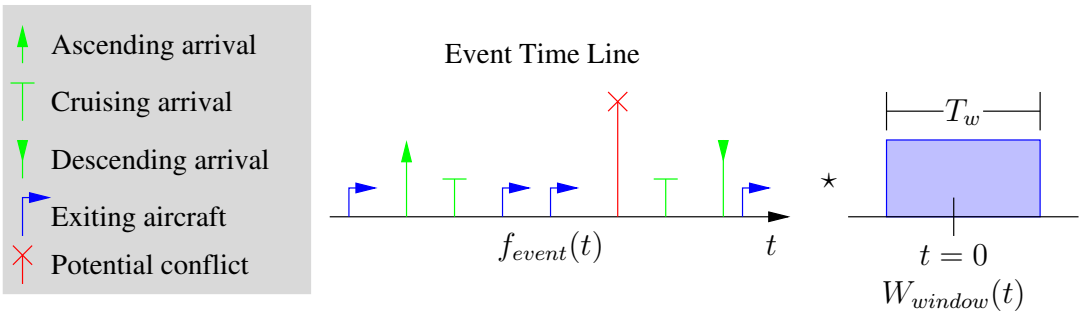


Figure 43: Representation of the event process within an airspace, and method for calculating the minimum estimated controller communication time.

time and effort required to communicate commands and information between the controller and the pilots for that event. For example, when an aircraft enters an airspace and requests to ascend in altitude, the communication time from the initial acknowledgement to the granted clearance takes approximately 11 seconds - that is, if the air traffic controller does not detect any immediately pending conflicts as a result of the maneuver. Table 3 contains the communication times associated with each event considered in this communications model. Except for potential conflicts, the communication times are averages based on a sample audio recording of air traffic controllers within the Atlanta Center on November 3, 2010 over a 1 hour time period. The communication time for potential conflicts is selected to be 20 seconds. While the time is less than the previously cited 27.6 seconds [20], it is representative of more efficient actions taken by the air traffic controller. In this way, the resulting model for channel utilization more closely approximates a lower bound on real-world practices. Denoting the sequence of times for arriving level aircraft by $\{t_n^{arr,l}\}$; arriving descending aircraft by $\{t_n^{arr,d}\}$; arriving ascending aircraft by $\{t_n^{arr,a}\}$; exiting aircraft by $\{t_n^{exit}\}$; and potential conflicts by $\{t_n^{con}\}$; the complete event process function for a traffic scenario is

$$\begin{aligned} f_{event}(t) =& W^{arr,l} \sum_i (\delta(t - t_i^{arr,l})) + W^{arr,d} \sum_i (\delta(t - t_i^{arr,d})) \\ &+ W^{arr,a} \sum_i (\delta(t - t_i^{arr,a})) + W^{exit} \sum_i (\delta(t - t_i^{exit})) \\ &+ W^{con} \sum_i (\delta(t - t_i^{con})), \end{aligned} \tag{11}$$

where the weightings $W^{arr,l}$, $W^{arr,d}$, $W^{arr,a}$, W^{exit}, and W^{con} correspond to the times in

Table 3: Average communication time for each event.

Event	Average Time [seconds]	Weighting Variable
Arrival at level-flight	6	$W^{arr,l}$
Arrival requiring clearance	11	$W^{arr,d}$ and $W^{arr,a}$
Departure	6	W^{exit}
Potential conflict	15	W^{con}

Table 3.

To calculate the communication time over all time intervals for an entire traffic scenario, the event process function $f_{event}(t)$ is convolved with a square-function centered at 0 with the appropriate time-width. That is, for a time interval of T_w minutes, the square-function is

$$h^{sq}(t) = \begin{cases} 1, & \text{if } t < T_w \\ \\ 0 & \text{else.} \end{cases} \tag{12}$$

The resulting communication time, $C_{T_w}(t)$, within $T_w/2$ minutes of time t is

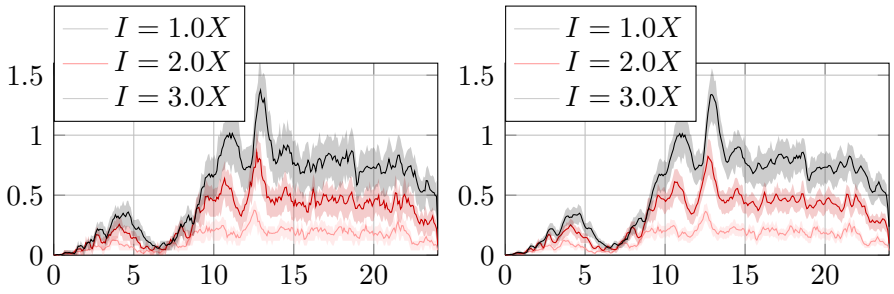
$$C_{T_w}(t) = h^{sq}(t) \star f_{event}(t). \tag{13}$$

An illustration depicting the computation of communication times within a desired time interval is shown in Figure 43.

The percent utilization of the communication channel for the time window T_w, denoted $U_{T_w}(t)$, is calculated by the equation

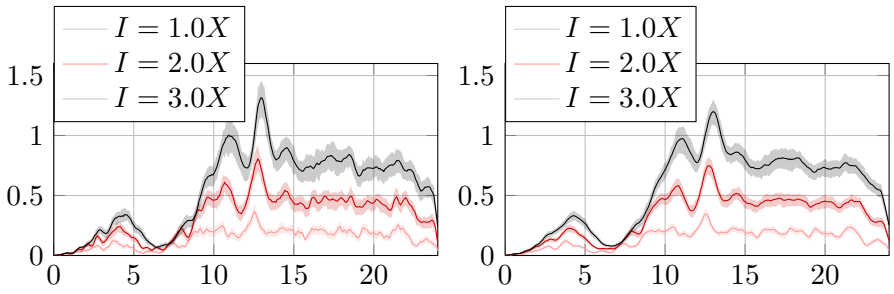
$$U_{T_w}(t) = C_{T_w}(t)/T_w. \tag{14}$$

If at time t, $U_{T_w}(t) = 1$, then events within the airspace require that the air traffic controller and pilots communicate for the complete T_w minutes. Note, the case when $U_{T_w}(t) > 1$ does not necessarily imply infeasibility in regards to the communication and control limits. In practice, some communication events can be processed earlier (e.g hand-offs) or later (e.g. acknowledgements, and clearances) than normally would occur without loss of safety or situational awareness. Thus, time-shifting of events can still allow for feasibility over shorter time windows. However, while $U_{T_w=2.5}(t) > 1$ and $U_{T_w=5.0}(t) > 1$ might still be feasible, $U_{T_w=10.0}(t) > 1$ is surely not.



(a) $T_w = 2.50$ minutes

(b) $T_w = 5.00$ minutes



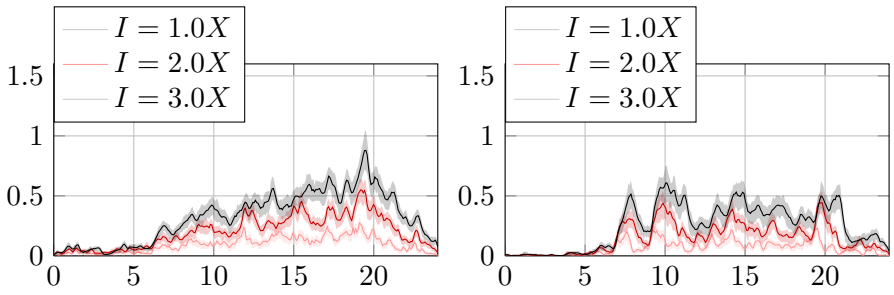
(c) $T_w = 10.00$ minutes

(d) $T_w = 20.00$ minutes

Figure 44: Average utilization time with upper and lower quartiles for ZMP42.

The expected utilization times, $E[U_{T_w}(t)]$, for ZMP42 for different values of T_w and traffic intensities are shown in Figure 44. Similar to conflict counts within the sector, the expected utilization time increases significantly during peak traffic hours. Furthermore, as traffic intensity increases, the average utilization time multiples. The utilization times for ZMP12 and ZMP16 with $T_w = 5$ minutes are illustrated in Figure 45.

While it is tempting to use the utilization times $U_{T_w}(t)$ to estimate controller workload or to establish when traffic conditions exceed reasonable workload limits, such an approach



(a) ZMP12

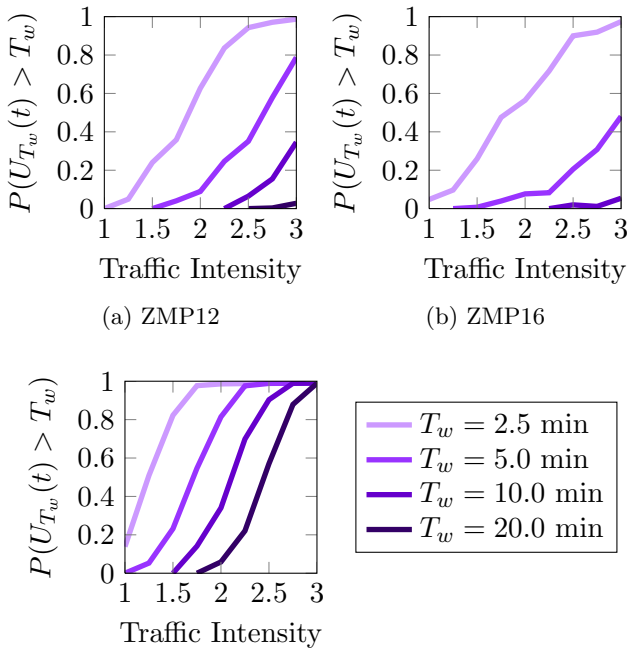
(b) ZMP16

Figure 45: Average utilization time with upper and lower quartiles for $T_w = 5$ minutes.

is not taken. Because the manifestation and dictation of workload limits in each of the sectors and for each controller are different, it is not possible to establish a consistent bound. As noted in prior results in this chapter, the type of potential conflicts commonly encountered varies significantly between the sectors. Because the effort required to resolve different types of conflicts is not uniform, a simplistic model that linearly scales and sums the communication times does not accurately capture controller workload. Furthermore, in human-factors research there is a recognition that "workload is not something imposed upon a passive [air traffic controller] but, rather, is something the [air traffic controller] actively manages" [59]. In virtually all aspects of their role, air traffic controllers adjust their strategies in response to system changes, future occurrences, and perceived workload.

Instead of using $U_{T_w}(t)$ as a measure of controller workload, the utilization time is checked to see if it ever exceeds the value of 1. In this manner a more robust and conservative approach is taken to understanding the limits of air traffic controllers. And again, there should be an understanding that if events within an airspace require more time than is available, then no matter what strategies the controller takes in regulating his or her workload, proper management of the airspace is not possible.

Following the analysis framework above, the probability that utilization rates exceed the capability of the air traffic controller at some point during the day, i.e. $P(\exists t | U_{T_w}(t) > T_w)$, is considered. Excessive utilization is a suitable measure to gauge when traffic conditions are no longer manageable – likely well beyond human controller workload limits. Figure 46 shows the cumulative probability distributions of excessive utilization for different values of T_w. According to the results in Figure 46c, for the 5 minute time window, at 1.7X traffic levels in ZMP42, the minimum required communication and control effort exceeds the time allowed approximately 50% of the time. For the sectors ZMP12 and ZMP16, the corresponding traffic levels when utilization rates at $T_w = 5$ minutes are broken 50% of the time occurs at the 2.6X and 3X traffic intensities, respectively. The results suggest that these traffic intensities, while potentially manageable, are risky. This conclusion is supported by the human-in-the-loop simulations in [54] that indicated controllers struggle to maintain safe operating conditions at the 2X traffic level, and fail at the 3X traffic level.



(a) ZMP12 (b) ZMP16

(c) ZMP42

Figure 46: Probability of over-utilization for different time-windows, T_w.

Working under the assumption that air traffic controllers are unable to manage sustained efforts, perhaps equally important is identifying traffic levels when $P(\exists t | U_{T_w=10}(t) > 1)$. That is, air traffic controllers are presented with scenarios that require more than 10 continuous minutes of communication and control time. For the sectors ZMP12, ZMP16, and ZMP42, breaking medium-term communication and control limits occurs at the 2.25X, 2.25X, and 1.5X traffic intensities. While more restrictive than the 50% limit set above, these margins are more reflective of the longer-term management of airspaces. Even at these traffic levels, there exists the probability that within 5 minute time windows, communication and control limits are exceeded.

3.4 Review and Analysis

This chapter explores the uncontrolled conflict event process in sectors as traffic intensity increases. Assuming that air traffic control continues with the same policies, practices, and technologies in current operations, the forecast models suggest that increased traffic intensities will ultimately lead to traffic scenarios that are beyond the capabilities of the

unaided air traffic controller. Of particular concern is the quadratic growth rate of uncontrolled conflicts in relation to traffic intensity. For a doubling in traffic, the conflict count model proposed in this chapter predicts that uncontrolled conflicts will quadruple. And for sectors that are already heavily sectorized, more radical changes beyond airspace redesigns are required if a doubling of potential conflicts is to be managed by the human controller.

From an engineering stand-point, traffic demand is an uncontrollable variable that will continue to grow in the coming decades. To ameliorate problems associated with increased traffic levels, it is conjectured that there is a pending need to aid the air traffic controller through improved conflict-detection and conflict-resolution decision-support tools. Improved conflict-detection systems require improved trajectory prediction and position sensing of aircraft. Thus, while traffic intensity cannot be controlled, efforts to improve trajectory prediction and position sensing systems will likely enable a reduction in aircraft spacing distances, D_{sep}^{min}. Results in this chapter indicate that by reducing aircraft spacing, air traffic controllers can make use of improved trajectory prediction tools to reduce the number of potential conflicts that need resolution commands. With a reduction in the excessive labeling of potential conflicts, the conflict-resolution taskload will surely decrease. However, there are limits to which decreased separation standards can aid the air traffic controller. Because of the potential for failure of position-reporting systems and the ever-present error in trajectory prediction (even if uncertainty is reduced), aircraft are likely to continue to be spaced at greater distances then necessary. Continuing onward, conflict-resolution tools may be able to supplement the gains made by reducing aircraft spacing.

One primary advantage of advisory conflict-resolution tools is that they transfer part of the mental taskload from the human controller to a computer system. Such transference is not only important in mitigating the workload implications of increasing conflict-resolution taskload, but also because potential conflicts will become increasingly complex with greater traffic intensity. Potential conflict-clusters will evolve to contain greater numbers of aircraft and interactions. While previously the conflict-resolution process could be solved in

a sequential pairwise fashion with little worry, greater traffic loads generating more complex conflict inactions will inhibit this strategy. Computer decision-support tools have the potential to play a key role in resolving these more complex conflict scenarios. However, the design of conflict-resolution algorithms to form the foundation of an advisory system should not be considered straight forward. As demonstrated in this chapter, any conflict-resolution tool will be presented with handling a wide arrays of problems, conflict types (e.g ascending/descending aircraft, mixed aircraft models and capabilities) and geometries.

In summary, this chapter has indicated a great need for improvement in the polices, practices, and technologies used in current day air traffic control. Barring any adjustments, the human controller will be overwhelmed with the requirements of managing and separating aircraft in an airspace. Thus, there is a strong imperative to improve conflict-detection systems and introduce advisory conflict-resolution tools into the repertoire of air traffic controllers.

CHAPTER IV

GRAPH REPRESENTATIONS OF POTENTIAL CONFLICTS

Through simulations of uncontrolled air traffic at high intensities, the previous chapter revealed that the rate of potential conflicts, in addition to other air traffic control events (e.g. acknowledgements, clearances), will one day hinder the ability of the air traffic controller in maintaining situational awareness and safety. To prevent unsafe operating conditions, air navigation service providers are taking action by developing next generation conceptual systems such as NextGen and SESAR. These next generation systems adjust policies and practices, and integrate new technologies. Researchers and air navigation service providers hypothesize that these adjustments will aid air traffic controllers by reducing and transforming controller workload, thereby enabling increases in airspace capacity to support future air traffic demand. While there are numerous components to these next generation conceptual frameworks, semi- and fully-automated conflict-detection and resolution systems is a promising component. This chapter begins to explore the design, capability, and implementation of conflict-detection and resolution systems to better understand their interactions with human air traffic controllers in an advisory human-in-the-loop framework. As stated previously, the metric of interest for this thesis is the taskload associated with the conflict-resolution process, specifically, the number of advisory resolution commands an air traffic controller issues when acting in conjunction with an advisory conflict-detection and resolution system.

Addressing controller taskload is fundamental to establishing the amount of effort required by air traffic controllers to manage and separate aircraft within an airspace. While taskload measures are unable to account for hidden mental processes in the daily work of air traffic controllers, the measures do provide a metric by which to understand controller effort [62].

Taskload can correspond to a number of actions (acknowledgements, hand-offs, clearances, etc); however, the metric studied here is the number of resolution commands issued to ensure aircraft separation. Consequently, the term *conflict-resolution taskload* is coined.

Definition 3. ***Conflict-resolution taskload*** *is the number of resolution commands used in ensuring separation between aircraft during a given time period.*

Already, one study has supported the theoretical framework of addressing conflict-resolution taskload as a proxy for controller workload when operating under an advisory conflict-detection and resolution system. For the multi-stage human-in-the-loop study described in [53], air traffic controllers were tasked with managing a high traffic intensity airspace using current practices and procedures, and later using an advisory conflict-detection and resolution system. In a second set of experiments, using an advisory conflict-detection and resolution tool, human simulation studies were performed under a 4D trajectory based operations (TBO) framework with an automated acknowledgement and hand-off system. Both these operational settings are representative of other proposed concepts within next generation air traffic systems [50]. Trajectory-based operations is an alternative to the current clearance-based procedures now found in air traffic control. Instead of aircraft being cleared to make individual movements along a trajectory (e.g. heading or altitude changes), complete trajectories are approved by the air traffic controller. In contrast to the current piecewise planning of clearance-based control, longer-term planning is expected to enable greater consideration of pilot/carrier preferences and optimal airspace system performance [50]. Automated acknowledgement and hand-off systems also represent a significant departure from current practice in which air traffic controllers and pilots are expected to make vocal contact with one another. By off-loading these tasks to an automated system, human controllers are able to focus their attention on their primary task of ensuring safety through conflict-detection and resolution. Studies have already demonstrated the benefits of automated hand-offs and trajectory-based operations, including significant reductions in controller workload [12, 13, 53, 68, 79].

A number of key results emerged from the study in [53]. First, when working under

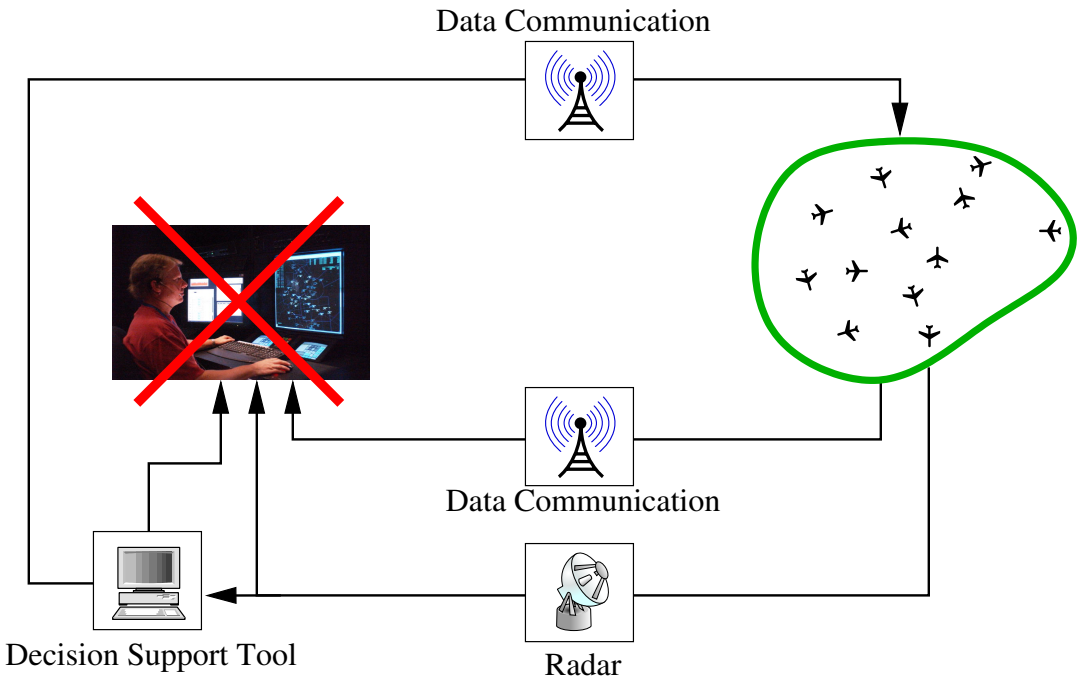


Figure 47: Air traffic controller deferring to an advisory conflict-detection and resolution system.

current procedures, air traffic controllers are barely able to mange 2X traffic intensities, and fail at 3X traffic levels. More pertinent to this chapter is that, when operating under an advisory conflict-detection and resolution system (with TBO and automated hand-offs), the measure most correlated to controller workload is the number of pending conflicts an air traffic controller has to resolve, i.e. the conflict-resolution taskload.

Having established that conflict-resolution taskload is worthy of study, it is of value to note one key statement made by the authors of [53]: "At the 3X level controllers and students accepted almost all advisories ($\sim$ 98%) due to time pressure." Effectively, the authors have indicated that at high traffic levels, air traffic controllers are unable to appropriately use the advisory system in the manner it was intended. Instead of checking trajectory solutions for safety, and potentially rejecting and deriving alternatives, the human controller operates as an open gate, automatically accepting all proposed solutions when overloaded. This situation is depicted in Figure 47. Implicitly, the advisory human-in-the-loop system reverts towards an automated system with the conflict-detection and resolution tool directly managing and separating aircraft. When considering that in the study the conflict-detection

and resolution system failed to identify or resolve all potential conflicts, then such deference behavior is of great concern. As best put by the author of [30], "There is a long history of cases in which operators are reportedly unaware of automation failures and do not detect critical system state changes when acting as monitors of automated systems." Further, "since the systems to be monitored continue to increase in complexity with the addition of automation, an increased trend towards large catastrophic failures often accompanies the incorporation of automation."

In response to the results above, the research goal of this thesis is to enable the design of advisory conflict-detection and resolution tools to be more consistent with the capabilities of human air traffic controllers. By making the best use of information, tools can be designed to reduce the number of resolution commands required to separate aircraft in order to manage time pressures. By studying the implementation, capabilities, and policies behind an advisory conflict-detection and resolution system, the discovery of best practices is possible.

To better understand the relationship between advisory systems and conflict-resolution taskload, a three step process is taken. First, in this chapter, graph based models are introduced and developed to describe the conflict-detection and resolution process. The graph models are an extension of the work introduced in [41]. In later chapters, human-in-the-loop conflict-detection and resolution decision-support tools are abstracted to model their behavior and characteristics. Finally, the abstracted models are applied to the previously generated traffic scenarios to understand how their implementations can reduce to conflict-resolution taskload.

4.1 Assumptions

To support the modeling of the conflict-detection and resolution process utilizing graphs, the following assumptions are required:

- The conflict detection and resolution process can be modeled according to a discrete-time system with a fixed step-size.
- Conflict relationships are deterministic.

- A dynamically feasible resolution command always exists to resolve potential conflicts.
- Resolution commands are implemented exactly as required and without delay.

4.2 Representing aircraft and potential conflicts through graphs

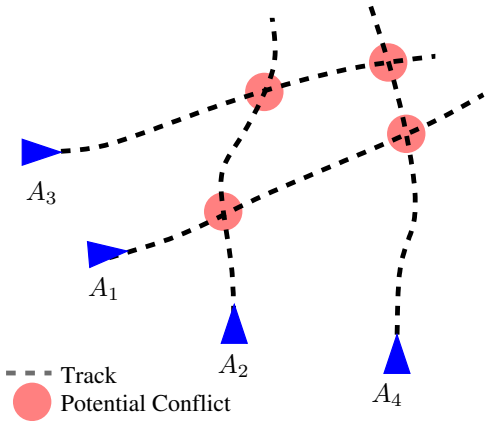
Consider a set of M aircraft, $A = \{A_1 ... A_M\}$, traveling through an airspace, as illustrated in Figure 48a. Aircraft trajectories are assumed to occur in 3D space across multiple flight-levels. According to aircraft intents, each aircraft has the potential to be in multiple conflicts if no control action is taken. For the en route environment, two aircraft are declared to be in conflict if they come within 5 NM laterally and 1,000ft vertically of each other. Federal Aviation Administration regulations require that air traffic controllers issue resolution commands to resolve any identified potential conflicts before they are realized. In the context of this thesis, a potential conflict is defined according to the definition below.

Definition 4. *A **potential conflict** between two aircraft is defined to exist if, according to the best available trajectory information and predictions, the two aircraft might come into conflict if no action is taken by the air traffic controller.*

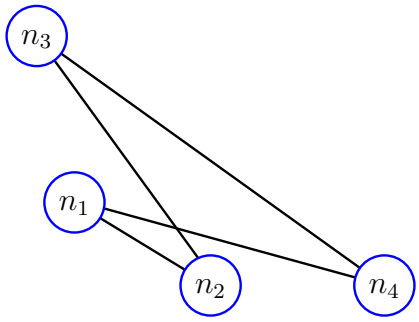
When and how a potential conflict is identified is addressed later in Section 5.3.

Aircraft and potential aircraft conflict relationships are represented by means of a graph. A representation of the potential conflicts for the example in Figure 48a is given by the undirected graph, $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, depicted in Figure 48b. Aircraft are represented by nodes in the vertex set $\mathcal{V} = \{n_1, ..., n_M\}$, where node n_i corresponds to aircraft A_i. Following regulations, for any pair of aircraft, (A_i, A_j), that are in potential conflict, the air traffic controller must issue a resolution command to at least one of the aircraft. Potential conflicts are indicated by an undirected edge in the edge set $\mathcal{E}$. That is, for the potential conflict between aircraft A_i and A_j, $(n_i, n_j) \in \mathcal{E}$. Adhering to the definitions above, the following definition for conflict graphs is provided.

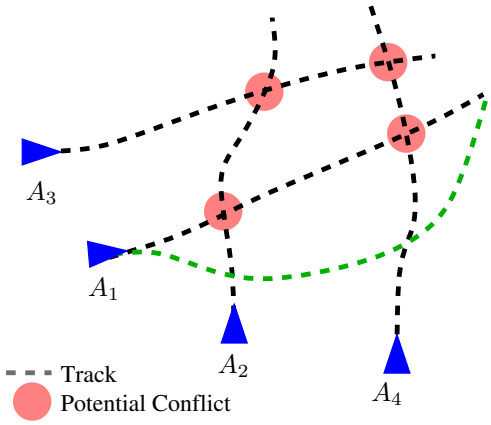
Definition 5. *A **conflict graph**, $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, is an abstract representation of aircraft and potential conflict relationships. Aircraft are represented by nodes in the vertex set $\mathcal{V}$, and potential conflicts are represented by edges in the edge set $\mathcal{E}$.*



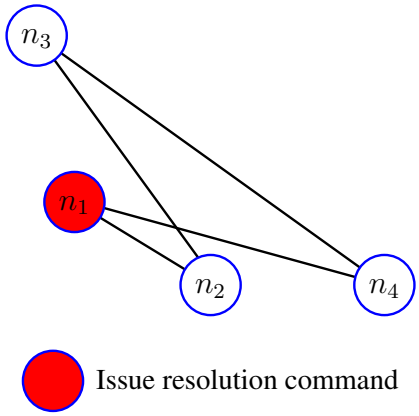
(a) Aircraft traversing airspace



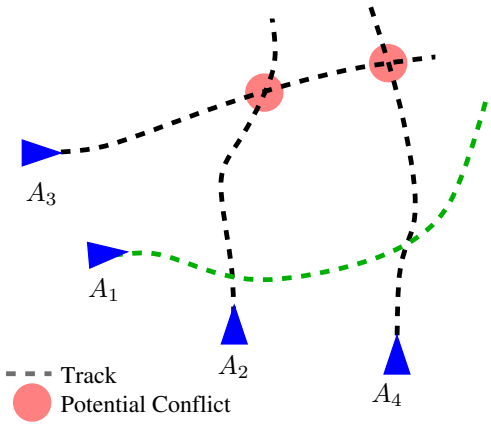
(b) Conflict graph representation



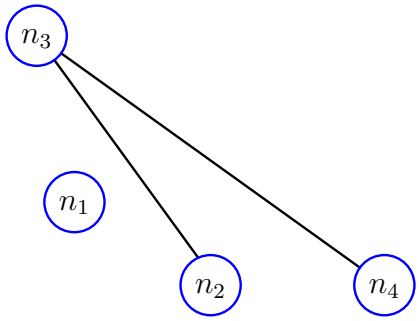
(c) Proposed solution



(d) Proposed solution in the graph space



(e) Proposed solution implemented



(f) Conflict graph updated according to new trajectory

Figure 48: Example graph representation of aircraft and potential conflict relationships.

Working in this framework, if the graph is completely disconnected, then the airspace is conflict-free and requires no action from the air traffic controller. Otherwise, edges exist within the graph and the air traffic controller should issue resolution commands to resolve potential conflicts. The act of issuing resolution commands to aircraft, assuming appropriate selection and implementation of the maneuvers, removes edges in the conflict graph. Through a series of controller actions, potential conflicts are resolved and correspondingly, edges in the conflict graph are removed.

Assertion 1. *An airspace is conflict-free if and only if the corresponding conflict graph is completely disconnected.*

A portion of the conflict-resolution process is illustrated in Figure 48 for the example provided. First, an aircraft is selected to be maneuvered, and a resolution command is proposed, as indicated in Figure 48c and Figure 48d. Following issuance of the resolution commands, potential conflicts are considered resolved and edges are removed from the graph $\mathcal{G}$, as depicted in Figure 48e and Figure 48f. If aircraft A_3 is also issued a resolution command, then it is possible to resolve all potential conflicts, such that the associated graph is completely disconnected.

The graph representation above is static; however, it can be extended to the dynamic case. A dynamic conflict graph is able to account for changes in the airspace as aircraft enter or exit and as potential conflicts are detected and resolved. Ultimately, a dynamic conflict graph represents the ongoing conflict-detection and conflict-resolution process.

To facilitate modeling of dynamic conflict graphs, a series of terms are introduced. Each term aids in describing if an aircraft belongs to the vertex set or is part of an element in the edge set. Furthermore, a distinction is made to indicate if an aircraft is able to accept a resolution command to prevent conflict.

The dynamic conflict graph is indicated by the discrete-time system, $\mathcal{G}(k) = (\mathcal{V}(k), \mathcal{E}(k))$. The set $\mathcal{V}(k)$ corresponds to the set of nodes belonging to the graph at time-step k. Similarly, $\mathcal{E}(k)$ contains the set of edges.

To construct dynamic conflict graphs, the definitions below are necessary. In accordance

with the purpose of the thesis, relevant terms are defined given the existence of an advisory decision-support tool to help automate the conflict-detection and resolution process.

Definition 6. *An aircraft is **visible** when the decision-support tool is aware of the aircraft's presence within or nearby the relevant airspace. If an aircraft A_i is visible at time-step k, then $n_i \in \mathcal{V}(k)$.*

Definition 7. *An aircraft is **controllable** if the air traffic controller is allowed to issue the aircraft a resolution command. In this case, the decision-support tool is permitted to generate a resolution command for the aircraft and propose a solution trajectory to the air traffic controller. According to standard operating procedures, an aircraft is controllable when it is located within the airspace of the managing air traffic controller or handed-off by the prior controller of the adjacent sector.*

Definition 8. *A potential conflict between two aircraft, (A_i, A_j), is **visible** if the decision-support tool predicts and identifies the potential for loss of separation. In this case, if a potential conflict between A_i and A_j is visible at time-step k, then $(n_i, n_j) \in \mathcal{E}(k)$.*

Definition 9. *A potential conflict between two aircraft, (A_i, A_j), is **resolvable** at time-step k if either A_i or A_j is controllable for the time-step.*

In the dynamic case, the conflict graph grows and shrinks incrementally with the arrival and departure of aircraft, and the identification and resolution of potential conflicts. Beginning with the conflict graph $\mathcal{G}(k) = (\mathcal{V}(k), \mathcal{E}(k))$, the graph is updated with the occurrence of each event (e.g. arrival, departure, identification of a potential conflict) during each time-step. For the time-step k, $\mathcal{V}^+(k)$ corresponds to the set of new aircraft arriving into the airspace over the considered interval of time. Any new potential conflicts identified within the same time-step are described by the edge set $\mathcal{E}^+(k)$. The new edges in $\mathcal{E}^+(k)$ include both potential conflicts generated by the arriving aircraft in $\mathcal{V}^+(k)$, and any new conflicts detected from aircraft already in the airspace. As such, $\mathcal{E}^+(k)$ may contain nodes (aircraft) that were already in the graph (airspace) prior to time-step k. During the conflict-detection process, it is also possible that updated information and trajectory predictions indicate that

a potential conflict will no longer occur. The set of unrealizable potential conflicts is designated by $\mathcal{E}^-(k)$. Following similar notation, $\mathcal{V}^-(k)$ denotes the set of nodes corresponding to the aircraft exiting the airspace.

For the dynamic graph model, resolution commands are not issued at the beginning of time-step k, but rather throughout it. Thus, the resolution process during time-step k is not considered completed until the end of the interval. Consider a temporary graph, $\mathcal{G}^T(k) = (\mathcal{V}^T(k), \mathcal{E}^T(k))$, just prior to time-step $k+1$, representing the airspace before any resolution commands have been issued, but taking into account other events within the airspace. In this case, the temporary graph is given by

$$\begin{aligned} \mathcal{V}^T(k) &= (V(k) \cup \mathcal{V}^+(k)) \\ \mathcal{E}^T(k) &= (E(k) \cup \mathcal{E}^+(k)) \setminus \mathcal{E}^-(k). \end{aligned} \tag{15}$$

The temporary vertex set $\mathcal{V}^T(k)$ contains aircraft already in the airspace, and new aircraft entering the airspace. The edge-set $\mathcal{E}^T(k)$ is representative of existing unresolved potential conflicts and new potential conflicts that have been identified.

Suppose that conflict-resolution actions act on the temporary graph $\mathcal{G}^T(k)$. Let CRP represent an arbitrary conflict-resolution procedure (CRP). The procedure takes as part of its input $\mathcal{G}^T(k)$ and returns $\mathcal{M}$, the set of nodes corresponding to maneuvering aircraft. The set $\mathcal{M}$ is dictated by the underlying conflict-resolution policy such that

$$\mathcal{M} = CRP(\mathcal{G}^T(k)). \tag{16}$$

Based on the conflict-resolution procedure, the edge set is updated to provide the graph $\mathcal{G}(k+1)$ at the next time-step. For a resolution command issued to aircraft A_i to prevent a conflict with A_j, where $n_i \in \mathcal{M}$, assume that the new advisory trajectory also prevents future conflicts according to the information encoded in the graph $\mathcal{G}^T(k)$. So if A_i is issued a resolution command, then the edge (n_i, n_j) is removed from the temporary graph. Additionally, it is possible that other edges that satisfy $(n_i, *) \in \mathcal{E}(k)$ are removed as well, depending on the resolution command selected. Accordingly, let $\mathcal{E}^r(k)$ represent any removed edges by means of the conflict-resolution procedure such that $\mathcal{G}(k+1)$; the graph at the next time-step is defined by

$$\begin{aligned} \mathcal{V}(k+1) &= \mathcal{V}^T(k) \setminus \mathcal{V}^-(k) \\ \mathcal{E}(k+1) &= \mathcal{E}^T(k) \setminus \mathcal{E}^r(k). \end{aligned} \tag{17}$$

The process described above does not require that all potential conflicts detected by the advisory system be resolved at the current time-step. But, at the very least, pending potential conflicts that will become realized in the next time-step without action must be resolved to maintain a safe airspace.

Note that the modeling here makes no distinction between trajectory-based operations and clearance-based control of aircraft. While the modeling procedure accurately accounts for the number of commands issued to resolve potential conflicts, in the case of clearance-based control, any subsequent maneuvers to place the aircraft back on its intended path are not accounted for when using lateral maneuvers (e.g. vectoring/heading commands). A more detailed discussion on the topic is provided in Section 5.7.

Ultimately, by following the discrete-time dynamic model dictated by the graph $\mathcal{G}(k)$, modeling the conflict-detection and resolution process is possible.

4.3 Review of Conflict Graphs

The need for decision-support tools to aid air traffic controllers in the conflict-detection and resolution process has motivated many researchers to create systems that provide advisory support. While they transform controller practice and reduce controller workload, there is limited understanding of how best to design and implement advisory decision-support tools. The design of an advisory system is particularly important when one considers that no guaranteed safe conflict-resolution algorithms exist to separate aircraft within an airspace. Therefore, before designing a conflict-resolution algorithm, it is important to understand the workload implications of any such system. This chapter takes a first step towards understanding the relationship between advisory tools and conflict-resolution taskload by modeling the conflict-detection and conflict-resolution process using graphs. In the ensuing chapters, the graph modeling is exploited to assess the amount of effort required by air traffic controllers using a given decision-support tool to resolve potential conflicts.

While modeling potential conflicts through graphs is based on fundamental relationships, the temporal dynamics rely on a number of assumptions and process constraints. Because the dynamic graphs are discrete-time systems, the process by which aircraft arrive and depart and potential conflicts are identified and resolved within each time-step are grouped together. As a result of the modeling, the order in which each task is processed within a time-step is not defined, however, some basic ordering is implicit. In the case when an aircraft enters an airspace, the aircraft should be visible to the conflict-detection and resolution system prior to being controllable - otherwise, conflict-resolution operations on a non-visible aircraft is nonsensical. Regardless, the task ordering is considered to be an aspect of the implemented conflict-detection and resolution program and thus, is left as general as possible. Furthermore, while not directly addressed in this chapter, future modeling and implementation of the conflict-detection and resolution processes on the graphs assume a constant time-step between subsequent values of k. While most research in the current literature on automated conflict-detection and conflict-resolution tools make use of fixed time-steps or event-based modeling [16, 83, 100, 101], there are no requirements for such implementations.

CHAPTER V

ABSTRACTION OF CONFLICT-DETECTION AND RESOLUTION TOOLS

In the previous chapter, a dynamic graph model, $\mathcal{G}(k) = (\mathcal{V}(k), \mathcal{E}(k))$, representing aircraft and their potential conflict relationships was introduced: aircraft appear as nodes, and potential conflicts are indicated by edges. At each time-step, the vertex and edge sets are updated as aircraft enter and exit the airspace, and potential conflicts are identified and resolved. Additionally, a generic conflict-resolution procedure (CRP) was introduced to act on the conflict graph as an advisory controller. There is a remaining need to detail how potential conflicts are identified, and how conflict-resolution algorithms resolve potential conflicts and prevent future secondary conflicts. Instead of specifying the exact implementation of an advisory conflict-detection and resolution tool, this chapter focuses on developing abstractions of the behaviors and capabilities of the tool acting on the dynamic conflict graphs. Through proper abstraction and parameterization, conflict-resolution taskload properties for many types of advisory systems can be explored. In this chapter, a framework for abstracting and parameterizing conflict-detection and resolution systems is presented.

Abstracting the behaviors and characteristics of conflict-detection and resolution tools enables high-level assertions on their performance. Ideally, this abstraction of advisory decision-support tools serves as an alternative to exhaustively designing tools, implementing them in high-fidelity simulations, and analyzing their conflict-resolution taskload. Such an approach of simulating specific conflict-detection and resolution systems limits the type of conclusions that can be drawn concerning the design of more generic algorithms: any results concerning conflict-resolution taskload are predicated on a specific implementation. By abstracting the properties of conflict-detection and resolution decision-support tools,

and implementing simulations in a framework consistent with their abstracted representations, broader statements on the relationship between the design of algorithms and conflict-resolution taskload, as well as best practices, can be investigated. Based on the discovery of best practices, a more guided approach can then be taken in the design of future conflict-detection and resolution systems.

In this thesis, conflict-detection and resolution systems are abstracted and parameterized according to the following properties:

1. The aircraft spacing distance used to identify potential conflicts.
2. The growth rate of uncertainty in predicting aircraft trajectories.
3. The policy a conflict-resolution algorithm uses in deciding which aircraft to maneuver.
4. The amount of future trajectory information used in making conflict-resolution decisions.
5. The length of time resolution trajectories can be guaranteed to be conflict-free.
6. The solve-time of the conflict-resolution algorithm. And by extension, the rate at which trajectory information is updated in the conflict-resolution process.

Properties 1 and 2 relate strictly to the conflict-detection process. Specifically, they correspond to the capability of position sensing systems and trajectory prediction tools used to measure and forecast aircraft positions. Based on the capabilities and limitations of trajectory prediction and measurement systems, policies and regulations that set aircraft spacing distances are established. The next three properties are primarily associated with conflict-resolution algorithms; however, they also depend on trajectory prediction tools. Furthermore, while the policy and information used to make decisions is a design property with some leeway, the amount of conflict-free time a resolution system can guarantee is an active area of research strongly correlated to the relative "strength" of an algorithm. Finally, the solve-time is a coupling factor between the conflict-detection and conflict-resolution processes.

It is worth noting that the assessment of conflict-resolution taskload (i.e. the number of resolution commands used to separate aircraft) for a conflict-resolution tool cannot be made independent of a conflict-detection system. Conflict-resolution decision-support tools require information concerning which aircraft are in the airspace, what are their intents (i.e. flight-plans), where potential conflicts are located, and which conflicts require resolution immediately. Because conflict-resolution tools do not have access to this information natively, they must rely on conflict-detection probes for information when generating solution trajectories. Thus, when analyzing advisory decision-support tools for conflict-resolution, both the detection process and the resolution process must be considered concurrently.

Further, for conflict-detection and resolution systems, the manner in which potential conflicts are identified is critical. Minimally, a conflict-resolution algorithm should be capable of resolving near-term pending potential conflicts that are passed to the algorithm as constraints by the conflict-detection system. Potential conflicts that are too far into the future and contain too much uncertainty can be temporarily ignored until more accurate information is available. Otherwise, if all identified potential conflicts are immediately forwarded to a conflict-resolution tool and required to be resolved, then the number of advisory resolution commands can result in superfluous conflict-resolution taskload. That is, poor handling of uncertainties in aircraft trajectories can identify false potential conflicts that never arise.

For the remainder of the chapter, the abstraction of conflict-detection and resolution systems is made in regards to the properties listed above. In the next sections, a receding-horizon control model, consistent with the dynamic conflict graph representation, is proposed for describing the process by which the conflict-detection and resolution occurs; a framework for abstracting how potential conflicts are identified is described; and finally, the capabilities of conflict-resolution systems are then parameterized. In abstracting the implementation, capabilities, and strategies of the conflict-detection and conflict-resolution systems, the advisory decision-support tools can be parameterized according to how they identify and resolve potential conflicts.

5.1 Assumptions

Modeling the abstraction of conflict-detection and resolution tools requires a number of assumptions to enable consistent comparison between implementations. Moreover, minimum standards for behaviors are required. The assumptions supporting the abstraction of advisory conflict-detection and resolution tools are as follows:

- The conflict-detection and resolution process, as well as the actual implementation of a conflict-detection and resolution advisory system, can be modeled according to a discrete-time system with a fixed step size operating according to a receding-horizon control framework.
- Potential conflict relationships between aircraft are deterministic.
- Potential conflicts, once labeled as not requiring resolution due to updated position information and trajectory predictions of aircraft, do not reappear as requiring resolution later in time.
- Uncertainty in forecasting aircraft positions grows isotropically at a constant rate in the horizontal plane. There is no uncertainty in the prediction of aircraft positions in the vertical direction. Based on the uncertainty model, a cylindrical safety region around aircraft grows radially.
- A dynamically feasible conflict-free trajectory of predetermined time length exists and is generated by the conflict-resolution tool for each potential conflict. Furthermore, aircraft resolution commands ensure that potential conflicts are cleared, and not just avoided (see Section 5.7).
- Uncertainty in trajectory prediction does not affect the ability of conflict-resolution tools in generating conflict-free trajectory solutions.
- The conflict-resolution algorithm has the most recent trajectory prediction information when it begins resolving any potential conflicts.

- The advisory resolution trajectories are considered to be acceptable to the air traffic controller and pilot, because resolution commands are guaranteed conflict-free for a finite time period,

- The air traffic controller implements resolution commands as suggested by the advisory decision-support tool.

- Advisory resolution commands are communicated and implemented exactly as required and without delay by the air traffic controller and pilot.

The last three assumptions concerning the implementation of advisory commands require a brief discussion. Despite assuming the air traffic controller implements each advisory resolution solution as proposed by the conflict-resolution tool, that does not imply that the air traffic controller lacks the time to verify the safety of each resolution command. Furthermore, the case of rejecting advisory resolution commands is currently ignored. As one assumption states, advisory resolution commands are assumed to be acceptable to the air traffic controller. Such a property is only possible through a well-designed conflict-detection and resolution system.

5.2 Receding-Horizon Control

For the dynamic conflict-graph model presented in Section 4.2, the process of iteratively identifying and resolving potential conflicts corresponds to a receding-horizon control framework. As used in this thesis, Figure 49 illustrates the corresponding implementation schematic for advisory conflict-detection and resolution tools using receding-horizon control.

Starting at the current time t, a conflict-detection tool looks at least T_R minutes into the future to check for any potential conflicts. This advanced planning is to ensure that if resolution commands are issued, then they remain dynamically feasible and safe, even if there is a delay in implementation by the pilot. Additionally, the buffer time provides the air traffic controller with sufficient time to generate an alternative resolution command if he or she elects to reject the advisory commands. The selection of T_R allows for some relaxation of the last three assumptions in Section 5.1.

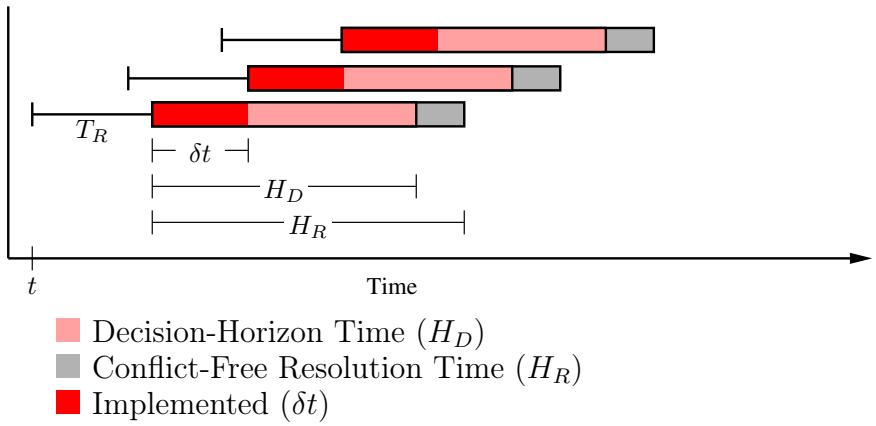


Figure 49: Conflict-resolution problem solved in a receding-horizon control framework.

When potential conflicts are detected, the conflict-resolution tool can limit its scope to only consider a subset of potential conflicts. Specifically, those pending conflicts that occur within a finite time window of length H_D. Limiting the scope prevents unnecessary consideration of uncertain potential conflicts that are too far into the future. Furthermore, reducing the scope enables real-time computation of resolution commands for a small set of aircrat and potential conflicts. Based on the potential conflicts within the H_D time window, the conflict-resolution problem is solved. If H_D is too large, then the conflict-resolution problem can be computationally difficult to solve.

According to the conflict-resolution algorithm, a set of advisory solution trajectories are generated. The modeling procedure assumes that each trajectory solution is guaranteed to be conflict-free for at least H_R minutes beyond the initial T_R if implemented by the controller.

In the case of receding-horizon control, only the resolution commands associated with the potential conflicts occurring within a δt time-window are implemented. Resolution of potential conflicts beyond the $[t + T_R, t + T_R + \delta t]$ implementation window are postponed until future time-steps when more accurate trajectory information is available.

The complete conflict-detection and resolution process is then repeated δt minutes later, ad infinitum. At each δt time step, the positions of aircraft are updated. This ensures that up-to-date information on potential conflicts and aircraft trajectories is always available, thereby preventing any unnecessary resolution commands due to uncertainty. The meaning

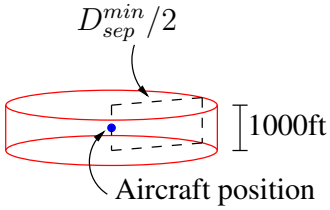


Figure 50: Safety region around an aircraft (not to scale)

and implementation behind the parameters H_D, H_R and δt are later explored in Section 5.3 and Section 5.4 in relation to the abstraction of conflict-detection and resolution tools.

As a requirement for the receding horizon-control framework, $H_R \geq H_D \geq \delta t$. Otherwise, potential conflicts are not identified with sufficient time to resolve them before they become realized. Also, because the implemented resolution commands are guaranteed to exist for at least $T_R + H_R$ minutes, potential conflicts cannot occur between current time t and $t + T_R$.

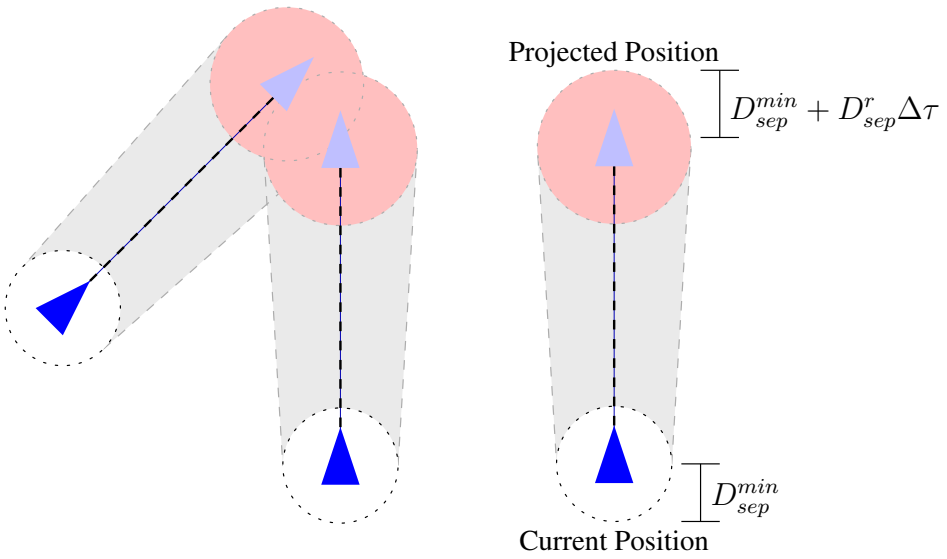
5.3 Parameterization of Conflict-Detection Systems

Before applying any conflict-resolution algorithms, potential conflicts must first be detected. The conflict-detection process is abstracted according to the following properties:

- How potential conflicts are identified.
- How uncertainty affects trajectory predictions.

Based on these characteristics, two parameters D_{sep}^{min} and D_{sep}^{r} are identified to describe conflict-detection tools.

Aircraft spacing requirements can be thought of as placing a cylindrical safety region around each aircraft that no other aircraft's safety region is permitted to penetrate. For an aircraft's current position, Figure 50 depicts the safety region surrounding it. The radius of the safety region is set by the lateral aircraft spacing requirement of D_{sep}^{min}, while the vertical buffer is set to 500 ft in each direction. Safety regions are used to classify if potential conflicts exists. When trajectory predictions forecast that the safety regions of two aircraft will overlap at some point in time, the aircraft are said to have a potential conflict. Figure 51a depicts an example of a potential conflict between two aircraft flying



(a) Potential conflicts are indicated by overlapping safety regions

(b) Trajectory prediction with uncertainty

Figure 51: Parameters determining labeling of potential conflicts.

at the same flight-level. To consider uncertainty in trajectory-prediction algorithms, the radius of the safety region is modeled to grow linearly in time, as depicted in Figure 51b. For a constant growth rate of D^r_{sep}, the radius of the safety region, when projected $\Delta\tau$ minutes in advance, grows from D^{min}_{sep} at the current time, to $D^{min}_{sep} + D^r_{sep}\Delta\tau$ in the future. Again, potential conflicts are identified according to overlapping safety regions. Growing the safety regions over time around the predicted aircraft positions is a robust approach to handling uncertainty in aircraft positions.

The values D^{min}_{sep} and D^r_{sep} are closely related to the capabilities of aircraft position sensing systems and trajectory-prediction tools. As stated previously, high altitude en route regulations state that aircraft should always maintain a minimum separation of 5 NM. In practice however, air traffic controllers begin to issue resolution commands when it appears aircraft trajectories will come within 8-9 NM of each other. While 4 NM, and even 2 NM separations are still far from a mid-air collision, uncertainty in radar systems inhibit air traffic controllers from spacing aircraft at these tight distances. For common en route radar systems, position updates arrive approximately every 12 seconds from a filtered radar data stream [76]. Because radar position reports degrade with distance, over 250 NM, "aircraft less than 5 miles apart could theoretically swap positions on radar monitors [76]." So while

radar systems may indicate clearance at 5 NM, noises in sensor measurements can obscure the fact that aircraft are much closer, even if filtering algorithms (e.g. alpha-beta filters) are present to estimate aircraft positions [102]. Ultimately, future reductions in aircraft spacing requirements will require a safety assessment that considers radar error, as well as other factors such as potential stresses and failures (airspace structure, controller/aircrew workload, communication systems), in order to limit the acceptable risk for a mid-air collision [60]. Further, air traffic controllers add their own safety buffers to ensure they adhere to safety regulations. In designing advisory conflict-detection and resolution tools, the conflict-detection sub-system will be required to mimic the same behavior as air traffic controllers. While D_{sep}^{min} is initially tied to aircraft separation requirements (i.e. 5 NM), it is best to think of the term in regards to how conflict-detection and resolution systems space aircraft. That is, if an advisory system detects a potential conflict according to the D_{sep}^{min} parameter, then it advises a resolution trajectory to the air traffic controller. Note, however, in practice the air traffic controller is not required to implement the proposed trajectory if they feel it is unnecessary. This potential case occurs when the conflict-detection and resolution system predicts a potential conflict, yet the air traffic controller is confident one will not occur.

The growth rate model used to represent uncertainties in trajectory prediction is a simplification based on prior studies. Recent works have indicated that the along-track error in trajectory position grows linearly along straight paths [34, 42, 74] - hence, the use of a linear model based on the parameter D_{sep}^{r}. The short-coming of such an approach is that uncertainty is assumed to grow isotropically in the X-Y plane. Additionally, the model assumes no uncertainty in the vertical direction. This simplified model does not replicate the types of errors associated with position reporting systems and trajectory-prediction tools [96, 67, 39]. Nonetheless, the parameters D_{sep}^{min} and D_{min}^{r} enable a basic model to understand how potential conflicts are labeled according to aircraft spacing and uncertainty.

Based on the two parameters D_{sep}^{min}, the aircraft spacing distance indicating advisement of a resolution command; and D_{sep}^{r} the growth rate in position uncertainty, the behavior of the conflict-detection process can be parameterized. The parameter D_{sep}^{min} is the aircraft

spacing distance used to label potential conflicts that require advisories. Based on the labeling of potential conflicts, the conflict-resolution algorithm then generates resolution trajectories.

The following definitions are now provided.

Definition 10. *The parameter D_{sep}^{min} is the **aircraft spacing** used in defining safety regions around each aircraft. When the safety regions encircling the forecast positions of the two aircraft overlap in space-time, the conflict-detection tool dictates that according to current information the conflict-resolution tool advise a resolution command. The parameter D_{sep}^{min} should be set to a value greater than or equal to the minimum required separation distance.*

Definition 11. *The **uncertainty** parameter D_{min}^{r} represents the growth in trajectory prediction errors in the X-Y plane according to a linear model.*

5.4 Parameterization of Conflict-Resolution Systems

Much like conflict-detection tools, the capabilities and behaviors of conflict-resolution algorithms can be abstracted and parameterized. Here, the abstraction of conflict-resolution tools focuses on the following properties:

- How much information is used in decision making.
- The decision-making policy used for selecting which aircraft to maneuver.
- Tool capability for finding conflict-free resolution commands.
- How often the conflict-resolution problem is solved.

The abstraction introduces two additional parameters, H_D and H_R, henceforth referred to as the decision-horizon time and the conflict-free resolution time. Additionally, the abstraction also makes use of δt, the solve-time parameter. The parameters H_D, H_R, and δt are first described assuming the presence of a conflict resolution procedure (CRP), such as the one introduced in Chapter 4. Next, the decision policies forming the backbone of conflict-resolution procedures are detailed.

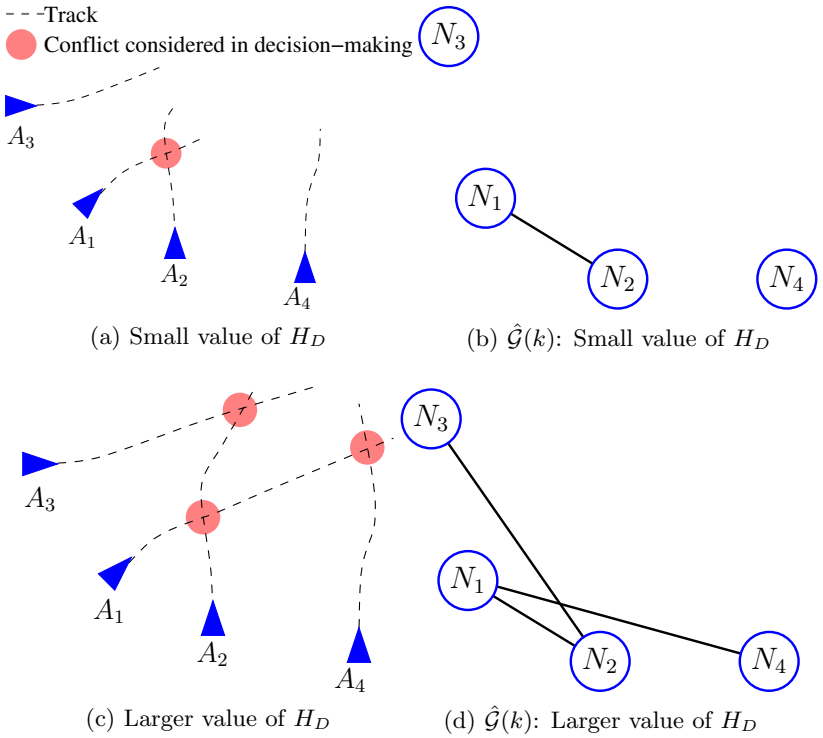
5.4.1 Implementations and Capabilities: H_D, H_R, and δt

H_D: Decision-Horizon Time. In Chapter 4, a conflict-resolution procedure acting on a dynamic conflict graph is described according to Equation 16. Taking as its input the temporary conflict graph at the current time step, $\mathcal{G}^T(k)$, the conflict-resolution procedure, CRP, outputs resolution commands to a set of aircraft $\mathcal{M}$. That is, $\mathcal{M} = CRP(\mathcal{G}^T(k))$. It is important to note that the decision of which aircraft to maneuver need not be based on all information supplied to the conflict-resolution procedure. In fact, if a conflict-resolution procedure did consider all aircraft and potential conflicts, even those far into the future, there exists the possibility that the conflict-resolution problem would no longer be computationally tractable. For example, when considering simultaneous conflict-resolution in the horizontal plane, for n aircraft involved in potential conflicts, the number of possible discrete decisions scales according to $n(n-1)/2$. At 2X traffic levels, it is not uncommon to have up to 40 aircraft in a sector. When considering discrete decision variables, and any other necessary calculations (fuel-burn, topologically similar yet alternative routes), the conflict-resolution problem is challenging to solve simultaneously for all 40 aircraft in implementation-time, even with simple algorithms [100]. Section 5.5 has a more detailed discussion on the issue of solution times.

To simplify the search for resolution commands, the conflict-resolution procedure can trim $\mathcal{G}^T(k)$ to limit the amount of information it considers in generating resolution trajectories. Trimming of temporary conflict graph $\mathcal{G}^T(k)$ yields a sub-graph referred to as the decision graph, denoted $\hat{\mathcal{G}}(k)$. The size of the decision-graph is established by the parameter H_D, the decision-horizon time. Now, instead of considering all potential conflicts, only a subset of them are considered when solving the conflict-resolution problem. More formal definitions of the decision-horizon time and decision-graphs are provided below.

Definition 12. *The parameter H_D, the **decision-horizon time**, as measured in minutes, indicates the time range over which conflict-resolution algorithms consider potential conflicts that might result in multiple aircraft resolutions.*

Definition 13. *The **decision-graph**, $\hat{\mathcal{G}}(k) = (\hat{\mathcal{V}}(k), \hat{\mathcal{E}}(k))$, at time-step k, for the current*



(a) Small value of H_D

(b) $\hat{\mathcal{G}}(k)$: Small value of H_D

(c) Larger value of H_D

(d) $\hat{\mathcal{G}}(k)$: Larger value of H_D

Figure 52: Representation of potential conflicts that are considered in the decision-making process.

time t, and parameterized according to the decision-horizon time H_D, is constructed by

$$\hat{\mathcal{V}}(k) = \mathcal{V}^T(k) \tag{18}$$

$$\hat{\mathcal{E}}(k) = \{(n_i, n_j) | (n_i, n_j) \in \mathcal{E}^T(k) \text{ and } t^c_{i,j} \in [t, t + H_D]\}, \tag{19}$$

where the values $t^c_{i,j}$ are the predicted potential conflicts times given by Equation 4.

A better understanding of the use of decision graphs and the decision-horizon time can be gained through the examples illustrated in Figure 52. When H_D is small, the conflict-resolution procedure only considers a single potential conflict between aircraft A_1 and A_2 (see Figure 52a and Figure 52b). Consequently, the conflict-resolution procedure generates a resolution command for one of the aircraft assuming that aircraft A_3 and A_4 continue on their intended trajectories. When H_D is larger, the decision-graph is expanded to include the potential conflicts between aircraft A_1 and A_4, and between aircraft A_2 and A_3. Thus, when deciding how to resolve the potential conflict between A_1 and A_2, the future resolution actions of A_3 and A_4 are considered, assuming that they are required to

maneuver. In contrast, for the prior case with a small value of H_D, the conflict-resolution algorithm assumes aircraft continue on their intended paths.

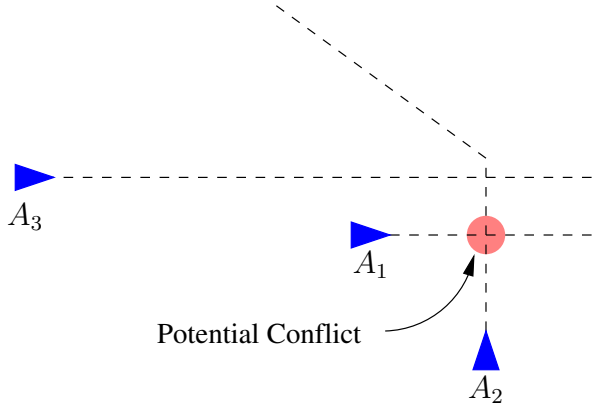
As noted previously, even the internal calculations of conflict-resolution tools require interaction with conflict-detection tools beyond the initial labeling of potential conflicts. Because the value of H_D relates to how far in advance potential conflicts are considered, selecting a value of H_D must be within the capabilities of the trajectory-prediction tools; potential conflicts cannot be considered if they cannot even be identified.

H_R: Conflict-Free Resolution Time. The capability of conflict-resolution algorithms is perhaps most related to the ability to ensure separation between aircraft. If an algorithm is unable to generate resolution commands that maintain proper spacing, then its value as an advisory decision-support tool is quite limited. Furthermore, if a conflict-resolution algorithm poorly routes aircraft, thereby generating secondary conflicts, then the advisory tool might create more work than it was intended to resolve. In parameterizing the strength of conflict-resolution algorithms, the guaranteed conflict-free resolution time, H_R, is introduced. The value of H_R represents the amount of time a conflict-resolution algorithm can guarantee that a resolution command is conflict-free beyond the initial T_R minutes.

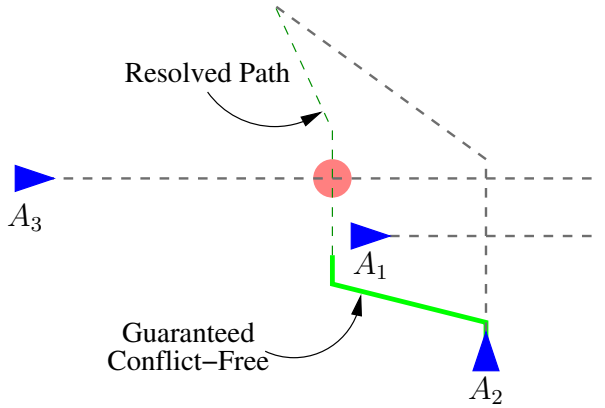
The importance of the conflict-free resolution time H_R is illustrated in Figure 53. When resolving a potential conflict, there is little value in issuing a resolution command to prevent one conflict if it generates another. As such, the relative size of H_R is important. Furthermore, H_R can be related to issues of safety. If conflict-free resolution times can only be guaranteed for short time-horizons, then air traffic controllers might be required to continuously resolve conflicts.

Definition 14. *The parameter H_R, the **conflict-free resolution time**, as measured in minutes, indicates the amount of time conflict-resolution algorithms guarantee resolution commands to be conflict-free beyond the initial T_R minutes.*

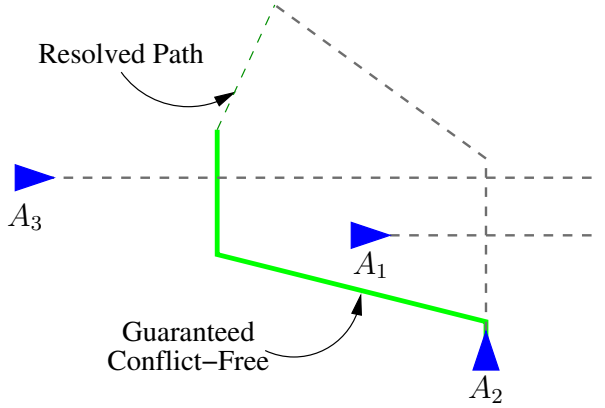
While the conflict-free time is stated as guaranteed, this is an assumption used only for



(a) Original traffic scenario



(b) Resolved scenario with small H_R generating secondary conflict



(c) Resolved scenario with larger H_R avoiding secondary conflict

Figure 53: Larger values of the guaranteed conflict-free time, H_R, help to prevent secondary conflicts.

mathematical modeling purposes. There are no guaranteed conflict-free algorithms. Furthermore, the lack of provably safe algorithms is one argument supporting the implementation of advisory systems. Therefore, as an alternative conceptualization, the guaranteed conflict-free time can be thought of as an expected value that is satisfied with probability $1 - \epsilon$, where $\epsilon \to 0$.

δt: Solve-Time. The parameter δt is used to describe how often the conflict-resolution problem is solved. The solve-time of a conflict-resolution algorithm is relevant because of how uncertainty in trajectory predictions manifests in excessive labeling of potential conflicts. As modeled in Section 5.3, uncertainty in trajectory predictions leads to the identification of potential conflicts which might not occur. While a rapid update rate for trajectory prediction is helpful in detecting potential conflicts, the conflict-resolution tool can only make use of information available when it begins solving the conflict-resolution problem.

Consider the case when the conflict-detection system prompts a conflict-resolution algorithm to resolve a number of potential conflicts. Also suppose that while the conflict-resolution problem is being solved, the conflict-detection system updates the state of potential conflicts such that one potential conflict is identified to no longer occur. Because conflict-resolution algorithm is running, the updated information cannot be considered, and thus, a resolution advisory is generated for the no-longer existent potential conflict.

The conflict-resolution algorithm gets snap-shots of the airspace, and must make decisions using the best available information. An illustration of this idea is provided in Figure 54. From the perspective of the conflict-resolution tool, the current positions and trajectories of aircraft are updated every δt minutes. And likewise, the safety region around each aircraft is reset.

The following definition of δt, the solve-time, is provided.

Definition 15. *The parameter δt corresponds to the **solve-time**, or how often the conflict-resolution problem is solved. Accordingly, aircraft position and trajectory prediction, and potential conflict information is updated within each implementation of the conflict-resolution*

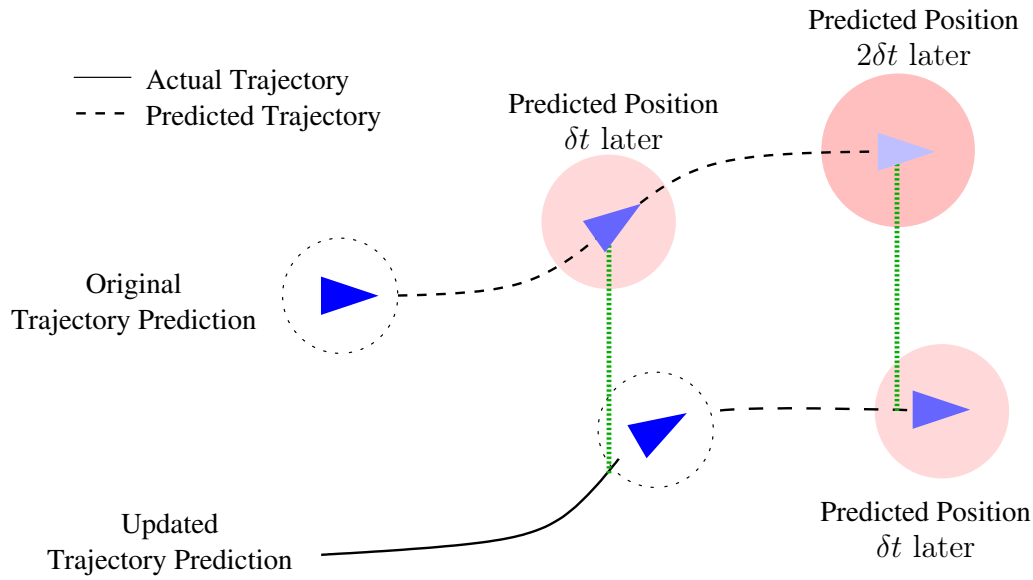


Figure 54: With solve-times of δt, trajectory predictions are updated for each implementation of the conflict-resolution tool.

algorithm.

In Section 5.5 a discussion of solution times and the parameters H_D and H_R is provided.

5.4.2 Conflict-Resolution Policies

Once potential conflicts are identified, the role of the advisory conflict-resolution algorithm is to select which aircraft should receive resolution commands. This section details three underlying policies for selecting aircraft. The first policy implements a simple first-come, first-served (FCFS) heuristic. The second policy randomly selects aircraft to maneuver until all potential conflicts are resolved. The final policy, is the Minimum Conflict-Resolution Taskload (MCRT) policy, which minimizes the number of resolution commands used to deconflict traffic.

Other conflict-resolution policies are detailed in Appendix A, all of which are ruled-based heuristics policies like the first-come, first-served and random policies.

The three policies highlighted in this section are specifically chosen to better understand the bounds on conflict-resolution taskload. The MCRT policy minimizing the number of resolution command proposed at each time-step. Thereby approaching a lower bound on the conflict-resolution taskload when implemented over long time periods. The FCFS policy is

representative of other generic rule-based heuristic policies. And finally, the random policy serves as a pseudo-upper bound and baseline standard for determining if other policies result in excessive conflict-resolution taskload due to their formulation.

Each conflict-resolution procedure, operating in accordance to a specific policy and within the receding-horizon framework, receives as part of its input a description of the conflict-relationships identified by the conflict-detection system. Specifically, the input includes the temporary conflict-graph, $\mathcal{G}^T(k)$, for the current time-step. Following the trimming process detailed in Section 5.4.1, parameterized by H_D, the conflict-resolution procedure then forms the decision graph, $\hat{\mathcal{G}}(k) = (\hat{\mathcal{V}}(k), \hat{\mathcal{E}}(k))$. Aircraft that are visible to the automated system at the time-step k, are represented by the vertex set $\hat{\mathcal{V}}(k)$. Potential conflicts are represented by the edge set $\hat{\mathcal{E}}(k)$.

A working definition of conflict-priority ordering is required.

Definition 16. *A conflict-resolution policy with* ***conflict-priority*** *ordering resolves potential conflicts according to the order in which they are predicted to occur.*

The decision policies presented in this thesis are representative of non-cooperative conflict-resolution policies that maneuver only aircraft involved in potential conflicts. While there does not exist a formal definition for cooperative or non-cooperative conflict-resolution policies, a working explanation is provided here.

Consider a potential conflict between aircraft A_i and A_j that requires at least one aircraft to maneuver. Let C_i and C_j represent the cost of maneuvering one aircraft or the other. Additionally, let $C_{i,j}$ be the cost of maneuvering both aircraft to resolve the potential conflict. For cooperative conflict-resolution algorithms, the relationship

$$C_{i,j} \leq C_i + C_j \tag{20}$$

can hold. Accordingly, in the context of an optimization framework, maneuvering both aircraft is the preferred option over maneuvering a single aircraft. Conversely for non-cooperative decision policies, it is a necessary condition that

$$C_{i,j} < C_i + C_j \tag{21}$$

for each pairwise conflict. Therefore, in the case of a single pairwise conflict, only one aircraft is maneuvered. When conflict-priority ordering is present, Equation 21 is a sufficient condition for non-cooperative decision policies.

First-come, first-served. The first-come, first-served policy (FCFS) is a straightforward heuristic that uses sequential path planning to generate safe aircraft trajectories. When implemented with conflict-priority ordering, conflicts are resolved sequentially until none remain. For each conflict-pair, (A_i, A_j), the aircraft arriving first into the sector is given priority, and thus, the aircraft arriving later is issued an advisory resolution command.

The algorithm for implementing the FCFS policy with conflict-priority ordering at time-step k is found in Procedure 2. Procedure 2 is initialized with the decision graph, $\hat{\mathcal{E}}(k)$, then potential conflicts are processed in the order of predicted occurrence. The predicted time of occurrence for each potential conflict between aircraft A_i and A_j is denoted by the variable $t^c_{i,j}$. Once the nearest pending unresolved potential conflict is identified, the arrival times for the aircraft are compared to determine which aircraft to maneuver. In Procedure 2, the temporary decision vector, m^T, contains the ordered set of aircraft that are issued resolution commands according to the FCFS policy with conflict-priority ordering. The value of m^T_q refers to the q^{th} aircraft to be issued a resolution command. Because the receding-horizon control framework only issues resolution commands to those aircraft with potential conflicts occurring in the $[t+T_R, t+T_R+\delta t]$ time interval, later potential conflicts are left to be solved in the future. The final step of the procedure is to segregate the aircraft associated with near-term potential conflicts. The output is $\mathcal{M}$, the set of aircraft the advisory conflict-detection and resolution system will advise rerouting.

Procedure 2 First-Come, First-Served with Conflict-Priority Ordering

$\mathcal{E}^T \leftarrow \hat{\mathcal{E}}(k),\ q = 0$

while $card(\mathcal{E}^T)$ **do**

$q = q + 1$

$(i^*, j^*) = \underset{i,j}{\text{argmin}}\ t^c_{i,j}$

$$m^T_q = \begin{cases} n_{i^*} & \text{if } t^a_{j^*} < t^a_{i^*} \\ \\ n_{j^*} & \text{else} \end{cases}$$

$\mathcal{E}^T \leftarrow \mathcal{E}^T \setminus (m^T_q, *)$

end while

$I = [t + T_R, t + T_R + \delta t]$

$\mathcal{M} = \{m \mid m \in m^T \cap \exists t^c_{m,*} \in I\}$

The removal of the conflict-priority ordering requirement results in a slightly different process for resolving potential conflicts; the procedure for FCFS without conflict-priority ordering is provided in Appendix A.

The primary advantages of the FCFS policy is that it provides a straightforward procedure to decide which aircraft are issued resolution commands. In regards to queuing for a service, the idea of prioritizing aircraft according to entry times appears sensible. However, one disadvantage of the FCFS policy is that there exists the possibility of consistently favoring aircraft traveling along specific routes. For example, consider the jetroutes illustrated in Figure 55 for ZMP42. For east-bound traffic along route J64, and north-bound traffic along route J25, a common point along the jetroutes promotes the possibility of potential conflicts. When operating under a FCFS policy, in the case of a potential conflict, the aircraft traveling along J64 are given priority over aircraft on J25 because they arrive into the sector earlier (as estimated by the distance between the aircraft entry points into the sector and the potential conflict). Through the FCFS decision mechanism, aircraft flying along specific routes are consistently favored over others, which could be considered as 'unfair' by air traffic carriers that heavily traffic the unfavored routes. While there is no direct

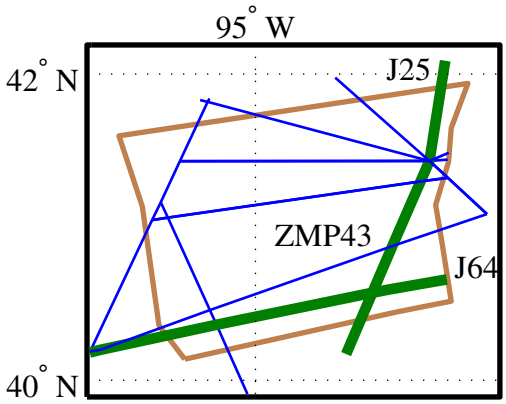


Figure 55: Sector map of ZMP42 with jetroutes

statistical evidence supporting persistent bias in automated conflict-resolution, the idea of fairness is often noted by designers of algorithms [2, 15, 45, 97, 101].

Random Policy. The introduction of a random policy to select maneuvering aircraft serves two purposes. First, in contrast to the FCFS policy, a random policy for assigning which aircraft are maneuvered does not contain biases that are favorable to a particular subset of aircraft. Second, a random policy serves as a reference standard to compare with the conflict-resolution taskload of other heuristic policies. For example, if the conflict-resolution taskload of the FCFS policy is greater than that for the random policy, then it can be stated that the FCFS policy yields excessive conflict-resolution taskload.

Procedure 3 Random with Conflict-Priority Ordering

$\mathcal{E}^T \leftarrow \hat{\mathcal{E}}(k),\ q = 0$

while $card(\mathcal{E}^T)$ **do**

$q = q + 1$

$(i^*, j^*) = \underset{i,j}{\text{argmin}}\ t^c_{i,j}$

$$m^T_q = \begin{cases} n_{i^*} & \text{if } rand < .5 \\ \\ n_{j^*} & \text{else} \end{cases}$$

$\mathcal{E}^T \leftarrow \mathcal{E}^T \setminus (m^T_q, *)$

end while

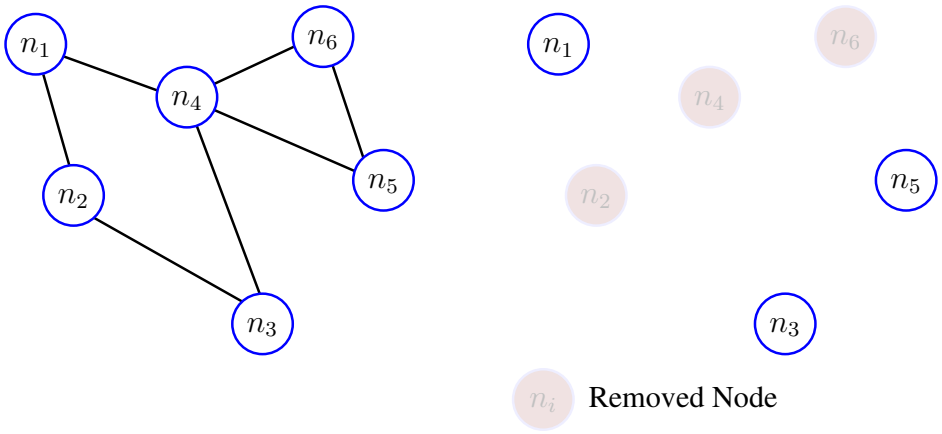
$I = [t + T_R, t + T_R + \delta t]$

$\mathcal{M} = \{m \mid m \in m^T \cap \exists t^c_{m,*} \in I\}$

One possible random policy is described by Procedure 3. The policy randomly selects which aircraft are issued advisory resolution commands according to conflict-priority ordering. Another random policy is proposed in Appendix A.

Minimum Conflict-Resolution Taskload Policy. Thus far, the policies introduced have relied on fixed procedures to determine which aircraft should be issued a resolution command. Further, none of the policies explicitly consider the potential repercussions on controller conflict-resolution taskload. The Minimum Conflict-Resolution Taskload policy (MCRT), overcomes this short-coming. Furthermore, the policy seeks to established a lower-bound on the amount of effort required to manage an airspace according to the capabilities of the conflict-detection and resolution system.

Addressing lower bounds on controller conflict-resolution taskload is fundamental to establishing the amount of effort required by air traffic controllers to manage and separate aircraft within an airspace when using an advisory decision-support tool. Currently, the effort required to resolve conflicts in an airspace remains a relatively ill-defined metric. Through a graph-based approach, tools from graph theory are applied to determine a lower



(a) Initial graph.

(b) Resulting graph.

Figure 56: Example application of the minimum vertex cover problem.

bound on the number of resolution commands required to separate aircraft.

Determining the minimum number of resolution commands is equivalent to applying the minimum vertex cover problem for graphs. According to [104], the minimum vertex cover problem (MVCP) asks: 'What is the minimum number of nodes that can be removed from a graph, such that the remaining graph is completely disconnected?' The corresponding act of removing a node n_i from the graph $\mathcal{G}$, is equivalent to issuing the aircraft A_i a resolution command. An example application of the minimum vertex cover problem is shown in Figure 56. Following removal of the nodes, the remaining graph is completely disconnected as illustrated in Figure 56b. (Note, the minimum vertex cover problem can have multiple solutions.) In the case of applying the minimum vertex cover problem to conflict-resolution, instead of removing nodes from the graph, any edges associated with the node are removed from the edge set. That is, if aircraft A_i is issued a resolution command, then after applying the conflict-resolution procedure, $(n_i, *) \notin \hat{\mathcal{E}^T}(k)$.

The minimum vertex cover problem can be expressed as a Mixed Integer Linear Program (MILP). While, there exist efficient large-scale methods for solving the NP-complete minimum taskload problem [18, 19, 88], given the relative size of the conflict graphs found in air traffic (i.e. less than 20 aircraft per conflict cluster, see Chapter 3 and [41]), the MILP formulation can be solved with sufficient speed (i.e. 0.02 seconds) by most standard integer

program solvers (e.g. CPLEX or LPSolve, and even Matlab).

For the decision graph $\hat{\mathcal{G}}(k) = (\hat{\mathcal{V}}(k), \hat{\mathcal{E}}(k))$ with M aircraft, let R_i be a binary variable indicating if aircraft A_i is to be issued a resolution command. If any two aircraft have nodes within the edge set $\hat{\mathcal{E}}(k)$, then at least one of the corresponding binary variables must be 1 to indicate a resolution command is issued to the aircraft. The minimum number of resolution commands is equivalent to minimizing the sum of the binary variables, R_i. The corresponding MILP is:

$$\begin{aligned} \min \quad & \textstyle\sum_{i=1}^{M} R_i \\ \text{s.t.} \quad & R_i + R_j \geq 1 \quad \forall\ (n_i,\ n_j) \in \hat{\mathcal{E}}(k) \\ & R_i \ \in \{0,1\}\ \forall\ i = 1 \ldots M \end{aligned} \tag{22}$$

The solution to the MVCP for conflict resolution is then given by the indices of the non-zero elements of the vector R.

The optimization problem in Equation 22 serves as the foundation for the minimum conflict-resolution taskload policy. The procedure for the MCRT policy is given in Procedure 4. The function $MVCP(\hat{\mathcal{E}}(k))$ represents the application of the minimum vertex cover problem. Following the nomenclature previously established, $\mathcal{M}$, is the set of aircraft to be issued advisory resolution commands during the current time-step.

Procedure 4 Minimum Conflict-Resolution Taskload Policy

$m^T = MVCP(\hat{\mathcal{E}}(k))$

$I = [t + T_R, t + T_R + \delta t]$

$\mathcal{M} = \{m \mid m \in m^T \cap \exists t^c_{m,*} \in I\}$

5.5 Solution Times

As it stands, the parameters H_D, H_R and δt, have no limitations as formalized. However, from a practical perspective, the ability to design an algorithm with large values of H_D and H_R, and a small value of δt is challenging. That is, considering large amounts of information and generating long conflict-free trajectories require longer computation times.

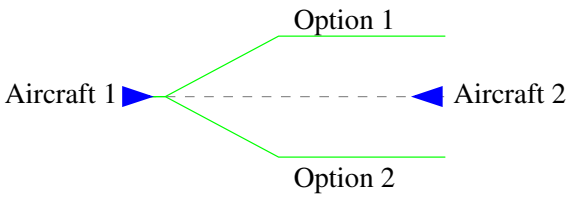


Figure 57: In the planar case, there exist two topologically distinct options for each resolution command.

Consider a potential conflict between two aircraft, as illustrated in Figure 57. Assuming reasonable behavior of aircraft routings (e.g. no loops), there exist two topologically distinct options for one aircraft maneuver around the other. For the case presented in the figure, these options are colloquially posed as "left or right." Recognizing that "left or right" is a relative term, it does not matter which aircraft is actually issued the maneuver. And fundamentally, for the planar case there are only two options regardless of the aircraft configuration (e.g. trailing aircraft: pass or stay behind; crossing aircraft: pass in front or pass behind) that represent two disconnected solution spaces. When extended to the simultaneous conflict-resolution problem with an arbitrary set of n aircraft, the number of combinations for the "left or right" option is given by $n(n-1)/2$, i.e. n choose 2.

For optimal formulations, the only way to ensure an optional solution is to exhaustively search through all possible decision-options (e.g. using branch and bound methods [80]). Thus, as the number of aircraft grows large, the disconnected solution space makes optimization difficult. This process can be quite time consuming when other options and constraints are considered. For example, constraints associated with aircraft dynamics restrict the size of each disconnected solution space, thereby making it challenging to even find initial feasible solutions to begin optimizing over (e.g. applying gradient descent methods). Furthermore, when relaxing which aircraft are allowed to maneuver, the "left or right" disconnected spaces are further subdivided into at least three options for each aircraft pair: maneuver aircraft 1, maneuver aircraft 2, maneuver both aircraft, maneuver no aircraft (assuming this option is allowable). With each disconnected sub-space, solution times generally increase. In the worst case, solution times grow exponentially.

Rule-based heuristic methods for conflict-resolution attempt to overcome solution-time

issues associated with optimal formulations by providing a fixed search procedure through a library of possible trajectory solutions. Because of the fixed library size, and a prescribed and dictated heuristic search procedure, rule-based methods often run in real-time [32, 73]. However, the capability of rule-based algorithms in finding feasible solutions is directly related to the heuristic search procedure and the size of the solution library. Furthermore, because of the difficulty of encoding descriptions of multi-aircraft conflicts into a rule-based search, these methods only consider pairwise conflicts when generating resolution commands. Thus, when resolving one potential conflict, possible secondary effects related to other potential conflicts are not considered.

There are key distinctions between rule-based and optimal formulations in their solution times and implementations. The rule-based methods require that $H_D = 0$ minutes, as only individual potential conflicts are considered at each time-step. Optimal formulations allow H_D to be set at any desired value. However, as H_D grows, the number of potential conflicts considered in the optimal formulation increases. As such, the solution space becomes increasingly disconnected (scaled by $n(n-1)/2$), thereby making it difficult to solve in real-time. So, from a practical perspective, increasing H_D is not always feasible. Of course attempts around this problem have been made by a number of researchers, particularly through the application of evolutionary optimization algorithms to rapidly explore the disconnected solution space [28].

In the degenerate case when $H_D = 0$ for the rule-based heuristics, a set of n problems is solved with $2(n-1)$ binary decision-options. While requiring a greater number of binary decisions than the simultaneous problem ($n(n-1)/2$ total binary decisions), the complete set of smaller problems can be solved faster than the simultaneous formulation [4].

Increasing the value of H_R is similarly constrained by the requirement for advisory tools to issue resolution commands in real-time. Consider the dynamic routing problem illustrated in Figure 58, in which an aircraft must find a path through moving weather (moving weather is a proxy for other aircraft). At each point along the aircraft trajectory, $x(t)$, the constraint $x(t) \notin \mathcal{W}(t)\ \forall t$ (where $\mathcal{W}(t)$ represents weather blockages) must be satisfied at each time-step. For the path planning problem, as H_R increases, the number of

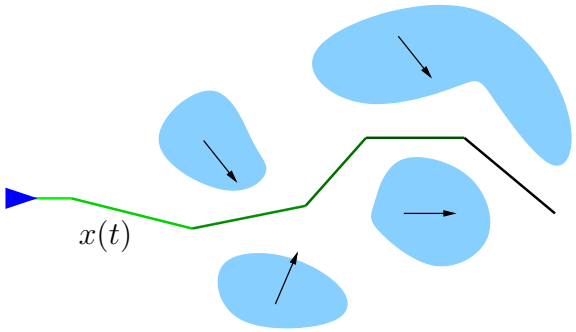


Figure 58: Finding solution to the motion planning problem becomes increasingly difficult as time horizons increase.

weather blockage constraints scales linearly. Meanwhile, additional constraints that ensure trajectory solutions satisfy aircraft dynamics (e.g. turn rates and stall speeds) are included. Furthermore, what makes the path planning problem even more difficult is that trajectories around moving weather (or aircraft) are inherently nonlinear, nonconvex, and multi-segmented. These properties prevent application of fast linear optimization algorithms (e.g. simplex) in solving the routing problem, and they require any rule-based algorithm to have a large library of solutions that include multi-segment routing.

In summary, large values of H_R and H_D are preferred for improved optimality and longer conflict-free solutions, but computation limitations prevent their implementation in practice. In many cases, with current computational constraints, the solution times for formulations with large values of H_R and H_D cannot be generated in real-time, or at least within the δt solve-rate.

5.6 Special Cases

Certain sets of implementation parameters δt, H_D, H_R, D^r_{sep}, and D^{min}_{sep} correspond to special cases of interest. These special cases are representative of absolute lower bounds, or common implementation frameworks proposed by previous researchers.

When $H_D = H_R = \infty$, and $D^r_{sep} = 0$, the parameterization represents a system with perfect knowledge and capabilities: all potential conflicts are correctly identified far into the future without the need for conservatively safe buffers, and all resolution commands are guaranteed to be conflict-free indefinitely. When the minimum conflict-resolution taskload

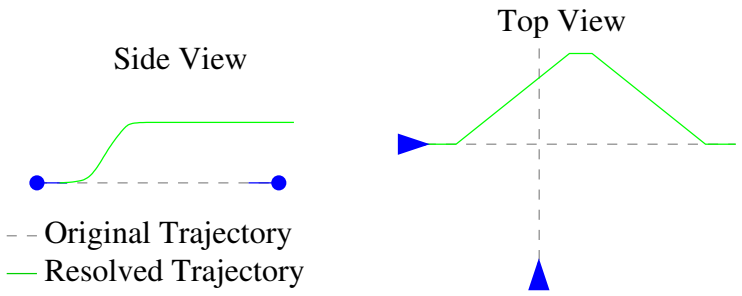
policy is applied with these parameters, the solution represents the minimum number of resolution commands used to separate aircraft. The solution serves as an absolute lower bound for all policies for a given traffic scenario. Thus, based on the solution to this special implementation, all other implementations in the remaining parameter space and policy space, can be compared against a reference standard. Additionally, when $\delta t = 0$, the implementation approximates an event-based conflict-detection and resolution tool: when potential conflicts are identified, they are resolved immediately without delay in computation.

Assigning $H_D = 0$ and $\delta t = 0$ is an implementation of interest that approximates pairwise sequential conflict resolution common to rule-based heuristic policies. The conflict-resolution process is considered to be pairwise, because as $H_D \to 0$, the decision-graph only contains a single potential conflict (assuming simultaneity of potential conflicts does not exist). To ensure feasibility of the implementation, $\delta t \to 0$ at an equal or greater rate than H_D; otherwise, within each time-interval not all conflicts are detected. Taking this case to an extreme, when the guaranteed conflict-free resolution time, H_R, is small (e.g. 1 minute), the system behaves reactively ,solving conflicts as they appear and only in the short-term. Such behavior is similar to the Traffic Collision Avoidance System (TCAS), which is designed to resolve any last-second conflicts that were not identified and resolved by the air traffic controller. TCAS represents the last line of safety, and its use is considered to be a failure. However, a difference between the previous policies described and TCAS, is that TCAS requires both aircraft to maneuver.

5.7 Clearance-Based Control versus Trajectory-Based Operations

At this stage, with the exception of the specification of conflict-free times, no assumptions are made concerning the nature of the resolution commands provided by the decision-support tool. That is, each resolution command can consist of any number and type of maneuvers (heading, speed, or altitude changes) of arbitrary magnitude. Moving forward, a distinction is needed between a resolution command and a resolution maneuver.

As an example, consider a potential conflict between two aircraft flying at level flight. Two possible resolution solutions include a flight level change and a lateral change for one of



(a) Vertical resolution solution.

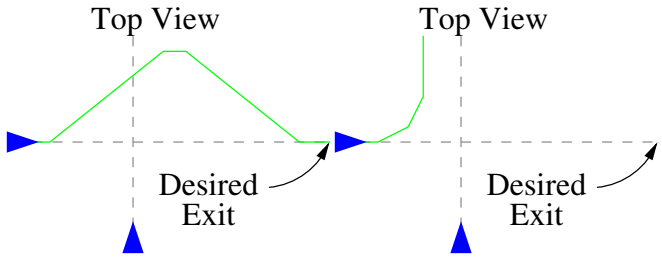
(b) Lateral resolution solution.

Figure 59: Two example resolution solutions for a potential conflict

the aircraft. As illustrated in Figure 59, in the latter case, a lateral resolution command can consist of multiple heading changes. Lateral maneuvers contain multiple heading resolutions to ensure that the aircraft returns to its original path. This type of path is refer to as a 'closed loop' trajectory. In other cases, if the heading change aligns with a future point in the flight plan, then a direct routing can be issued. This bypasses segments of the original flight plan. Because of the distinction between a resolution command and a resolution maneuver, the discussion of conflict-resolution taskload requires care.

Under current clearance-based control operations, the air traffic controller issues vocal clearances for each resolution maneuver by an aircraft. So, the conflict-detection and resolution model proposed in this thesis may account for each resolution maneuver used to prevent a conflict, but future maneuvers to ensure a close-loop trajectory are not considered. As such, the number of vocal communications associated with routing is under-represented by the conflict-resolution taskload. Thus, the conflict-resolution taskload represents a lower-bound. When only vertical maneuvers are used to deconflict traffic, the number of maneuvers issued by the controller approaches the conflict-resolution taskload.

Under the future NextGen system, the manner in which aircraft are issued resolution commands is likely to change. A major component of the NextGen system is the inclusion of 4D Trajectory-Based Operations (TBO) as an alternative to clearance-based control. Instead of approving individual resolution maneuvers, particularly in the case of potential conflicts, a complete trajectory is approved and communicated to the aircraft. The transmission of the new trajectory information can be communicated in multiple ways; however,



(a) Closed-loop resolution command. (b) Avoidance maneuver.

Figure 60: Difference between avoidance maneuvers and closed-loop resolution commands.

the most likely manner will be through digital data channels. Adherence to the amended trajectory is made possible through improved navigation aids such as GPS. Assuming aircraft conform to their trajectories, then air traffic controllers are only required to track aircraft that have potential conflicts and to provide additional services (e.g. congestion and weather alerts) to aircraft as requested. For trajectory-based operations, the conflict-resolution taskload of each policy more closely approximates the amount of effort required by the controller. Only in the case of amendments to a trajectory, is the air traffic controller required to verify it.

The use of closed-loop trajectory solutions is important to many conflict-resolution algorithms. As previously stated in Section 5.1, the model here assumes existence of resolution trajectories that clear potential conflicts, not just avoid them, as there is a critical distinction between the two cases. A visual representation that distinguishes between the avoidance maneuvers and resolution commands (particularly closed-loop trajectories), is illustrated in Figure 60. For the closed-loop resolution solution in Figure 60a, the aircraft reaches its destination. In contrast, for avoidance maneuvers there is no guarantee that aircraft are guided to their desired destinations. Instead, by constantly avoiding conflict, the aircraft is moved further away from its destination, as illustrated in Figure 60b.

5.8 Review

In a major literature review by Kuchar and Yang [56], the authors created a classification system to describe conflict-detection and resolution systems. The taxonomy they utilized

considered whether the algorithms were probabilistic, deterministic, or robust; the types of resolution maneuvers allowed (vertical, heading, speed); the solution methodology (prescribed, optimized, force field, manual); and whether the formulations were pairwise or global. In classifying algorithms, some properties and concerns of the conflict-detection and resolution tools can be readily identified and compared (e.g. deterministic prescribed algorithms fail to capture uncertainty, however they can be solved rapidly). In a similar fashion, the abstraction and parameterization of conflict-detection and resolution tools described in this chapter attempt to capture the capabilities of the algorithms. In doing so, assessments of the conflict-resolution taskload associated with each implementation are possible.

Building upon the conflict-graph models described in Chapter 4, a framework is developed to describe how conflict-detection and resolution algorithms operate in real-world implementation. First, a receding-horizon framework is proposed, in which the conflict-detection and resolution problem is continuously solved. At each time-step, potential conflicts are identified and resolved. The resolution commands for potential conflicts occurring in the near future are immediately advised to the air traffic controller by the decision-support tool. Potential conflicts that occur later in the future are still identified and resolved in the context of immediately pending conflicts; however, advisory commands are not proposed to the controller. As trajectory information is updated at each time-step, the conflict-detection and resolution process is repeated.

Following a description of the receding-horizon framework, parameters describing the behaviors and capabilities of the conflict-detection and resolution systems are provided. Conflict-detection systems are described according to how potential conflicts are labeled as requiring resolution. Conflict-resolution systems are parameterized according to how far in advance potential conflicts are considered and for how long resolution commands are guaranteed to be conflict-free.

Without actually delineating how the resolution maneuvers are selected, the framework and parameterization of the conflict-detection and resolution tools has been extended beyond that conceived by Kuchar and Yang. By characterizing advisory tools according to key model parameters and policies, a common methodology for analyzing the conflict-resolution

taskload of algorithms is now possible.

CHAPTER VI

SIMULATION MODELING FOR ABSTRACTED TOOLS

In the previous chapters, a model for generating uncontrolled traffic scenarios is proposed, a graph-based representation of the conflict-detection and resolution process is described, and finally an abstraction of conflict-detection and resolution tools is provided. To understand the relationship between conflict-resolution taskload and the design and implementation of conflict-detection and resolution systems, an appropriate simulation environment is required to extract relevant information. In this chapter, the simulation model used to implement the abstracted conflict-detection and resolution tools is described. Based on simulations, the conflict-resolution taskload for each traffic scenario and implementation is recorded, and a sensitivity analysis is performed to clarify best practices.

The primary difficulty in constructing a reasonable simulation environment, consistent with the prior abstractions, is in representing the conflict-detection and resolution process without specifying the exact trajectory solutions used in resolving potential conflicts. When resolution trajectories are not specified, the manner by which secondary conflicts are generated is ill-defined. Prior to describing the process for simulating the abstracted conflict-detection and resolution tools, some additional properties related to the conflict-resolution process must be stated such that approximations can be placed into context.

When an air traffic controller or conflict-resolution tool issues a command, ideally, the resulting aircraft trajectory reduces or removes the possibility of future potential conflicts in the near-term. Let $P^o_{con}(A_i|t^c_{i,*} \in [t, t+\Delta\tau])$ denote the probability for aircraft A_i to have a potential conflict in the time interval $[t, t+\Delta\tau]$ prior to any resolution commands. And let $P^+_{con}(A_i|t^c_{i,*} \in [t, t+\Delta\tau])$ denote the probability of potential conflict after aircraft A_i receives a resolution command. Then the relationship

$$P^+_{con}(A_i|t^c_{i,*} \in [t, t+\Delta\tau]) < P^o_{con}(A_i|t^c_{i,*} \in [t, t+\Delta\tau]) \tag{23}$$

should be satisfied for some value of $\Delta\tau \in \mathcal{R}^+$. It is desired that all possibility for a

potential conflict disappears after implementing a resolution command, such that

$$P^{+}_{con}(A_i | t^{c}_{i,*} \in [t, t + \Delta\tau]) = 0. \tag{24}$$

According to the abstraction of conflict-resolution systems provided in Chapter 5, for the guaranteed conflict-free time H_R, Equation 24 becomes $P^{+}_{con}(A_i | t^{c}_{i,*} \in [t, t + T_R + H_R]) = 0$. Now, what occurs beyond $T_R + H_R$ minutes is of particular interest in this chapter. Conflict-resolution solutions are modeled not to satisfy Equation 24 for $\Delta\tau > (T_R + H_R)$, so there always exists some probability of secondary conflicts. As it stands, the dynamic graph model and the abstractions of conflict-detection and resolution tools do not directly indicate how secondary conflicts are taken into account in a simulation environment. In practice, to generate a secondary conflict, a specified resolution trajectory is required. However, the act of generating specified resolution commands is exactly what the abstraction of conflict-detection and resolution tools aims to avoid. This chapter presents a method for mimicking secondary conflicts.

6.1 Creation of Complete Conflict-Graphs

To generate traffic scenarios for simulating the abstracted advisory conflict-detection and resolution tools, the conflict relationships described by the complete uncontrolled graphs $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ introduced in Chapter 3 are used. The uncontrolled graph approximates potential conflict relationships that would occur if aircraft flew irrespective of each other and without any input from an air traffic controller. For the purposes of simulation, the uncontrolled graph is expanded to include potential conflicts beyond the standard 5 NM minimum separation requirement. Instead, all aircraft pairs, (A_i, A_j), that come within 15 NM and 1000 vertical feet of each other are included as part of the complete edge set $\mathcal{E}$. Also, encoded as part of the graph is the minimum miss-distance $D^{miss}_{i,j}$, as well as the predicted time of potential conflict, $t^{c}_{i,j}$, given by Equation 3 and Equation 4.

6.2 Simulation Implementation

As noted previously, the use of uncontrolled conflict graphs, $\mathcal{G}$, is not directly applicable to the dynamic graphs, $\mathcal{G}(k)$, modeled in Chapter 4. For simulations, a proxy for the dynamic

graph is required because the realization of secondary conflicts is not directly possible without simulating an actual conflict-resolution algorithm. To approximate dynamic graphs, the uncontrolled conflict graph is incrementally and deterministically sampled. The deterministic sampling of the uncontrolled conflict graph adds and removes nodes and edges to the proxy dynamic graph, thereby mimicking entering and exiting aircraft and the detection and resolution of potential conflicts.

The simulation procedure to approximate dynamic graphs using uncontrolled graphs is provided in Procedure 5. In step 1, each traffic scenario is initiated by the uncontrolled graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. The conflict-detection and resolution process is applied at discrete time-steps until the simulated time is completed.

At time-step k, the conflict-detection process is applied, as indicated in step 5. Next, according to the current time, look ahead value T_R, and the parameters D_{sep}^{min} and D_{sep}^{r}, the conflict-detection procedure, indicated by the function $CD(\mathcal{G}, t, T_R, D_{sep}^{min}, D_{r}^{min})$, determine all potential conflicts requiring resolution during the time-window $[t + T_R, t + T_R + H_D]$. The output of the function is the graph $\tilde{\mathcal{G}}(k)$. The graph $\tilde{\mathcal{G}}(k)$ represents an approximation to the dynamic graph $\mathcal{G}(k)$, and is constructed by

$$\tilde{\mathcal{V}}(k) = \{v_i | v_i \in \mathcal{V} \cap A_i \in \text{visible and controllable}\} \tag{25}$$

$$\tilde{\mathcal{E}}(k) = \{(n_i, n_j) | D_{i,j}^{miss} < (D_{sep}^{min} + (t_{i,j}^{c} - t - T_r) D_{sep}^{r}) \cap (n_i, n_j) \in \mathcal{V} \cap (n_i, n_j) \in \text{visible}\}. \tag{26}$$

Aircraft A_i, as represented in the approximated dynamic graph, is considered visible and controllable if its arrival time, t_i^a, is greater than $t + T_R$. Furthermore, all potential conflicts occurring after $t + T_R$ are visible.

The next steps correspond to the conflict-resolution procedure. Based on the approximated dynamic graph $\tilde{\mathcal{G}}(k)$, an approximated decision-graph, $\tilde{\tilde{\mathcal{G}}}(k)$, is constructed to include only those potential conflicts within $H_D + T_R$ of the current time. The conflict-decision policy (e.g. FCFS, MCRT, random) is applied through the function $CR(\tilde{\tilde{\mathcal{G}}}(k))$ in step 7. Implicit in the function call is that any other additional information required in deciding which aircraft to maneuver (e.g. arrival times, flight phase, etc) is included. After applying the decision policy, the output of the function is the set of maneuvering aircraft $\mathcal{M}$. Based

on which aircraft are maneuvered, the uncontrolled conflict-graph, $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, is updated. For an aircraft A_i that is issued a resolution command at time t, all future potential conflicts between $[t+T_R, t+T_R+H_R]$, represented in the edge set $\mathcal{E}$, are removed. The procedure is then repeated until the time-span for the traffic scenario, as defined by t_{sim}, is completed.

Procedure 5 Simulating Conflict-Detection and Resolution Process

1: $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

2: $k = 0, t = 0$

3: **while** $t < t_{sim}$ **do**

4: $\quad k = k + 1, t = t + \delta t$

5: $\quad \tilde{\mathcal{G}}(k) = CD(\mathcal{G}, t, D_{sep}^{min}, D_r^{min})$

6: $\quad \tilde{\hat{\mathcal{G}}}(k) = (\tilde{\hat{\mathcal{V}}}(k), \tilde{\hat{\mathcal{E}}}(k))$ where

$$\tilde{\hat{\mathcal{V}}}(k) = \tilde{\mathcal{V}}(k)$$

$$\tilde{\hat{\mathcal{E}}}(k) = \{(n_i, n_i) | (n_i, n_i) \in \tilde{\mathcal{E}}(k) \cap t_{i,j}^c \leq (t + T_R + H_R)\}$$

7: $\quad \mathcal{M} = CR(\tilde{\hat{\mathcal{G}}}(k))$

8: $\quad \mathcal{E}^r = \{(A_i, A_j) | A_i \in \mathcal{M}, t_{i,j}^c \leq t + T_R + H_R\}$

9: $\quad \mathcal{E} = \mathcal{E} \setminus \mathcal{E}^r$

10: **end while**

The final steps 8 and 9 of the simulation procedure are key to approximating secondary conflicts. When a resolution command is guaranteed to be conflict-free, the probability of conflict during the interval $[t, t + T_R + H_R]$ is zero. In other words, Equation 24 holds true for $\Delta\tau = T_R + H_R$. Following $T_R + H_R$ minutes, potential conflicts represented in the uncontrolled conflict graph $\mathcal{G}$ are maintained even after resolution commands are implemented. In this way, the simulation keeps the probability of conflict outside the time interval $[t, t+T_R+H_R]$ constant when a resolution command is issued. Not only does the probability of potential conflict remain statistically similar for the open interval $(t + T_R + H + R, \infty)$, but so does the set of aircraft for which potential conflicts might involve. In this sense, the procedure maintains the relationship

$$P_{con}^{o}(A_i | t_{i,*}^c \in (t + T_R + H_R, \infty)) = P_{con}^{+}(A_i | t_{i,*}^c \in (t + T_R + H_R, \infty)) \tag{27}$$

when applying a wide range of conflict-detection and resolution policies, without actually dictating specific resolution commands.

Resolution commands often take aircraft off their intended paths, either due to altitude changes or lateral movements. As such, the aircraft maneuvers might cause interactions that would not have occurred if aircraft maintained their desired routes. However, there is currently no experimental validation supporting either claim that conflict probabilities $T_R + H_R$ minutes after a resolution command remain the same or change. Therefore, the assumption represented by Equation 27 is maintained.

Note, the implementation described in Procedure 5 is invariant to the value of T_R, when assuming that conflict-free resolution commands of length H_R always exists. As such, future simulations do not use the term.

6.3 Parameter Settings and Traffic Scenarios

Using the simulation described in Procedure 5, a wide variety of parameter settings and traffic cases are considered in the execution of the abstracted conflict-detection and resolution tools. In line with the intent of supporting a 3X traffic demand by 2025, simulations consider 50 traffic scenarios for intensities from 1X to 3X current traffic levels. The scenarios are selected from the 100 uncontrolled traffic scenarios generated in Chapter 2, and parsed down to consider only the 10 hour window between 10AM and 8PM. Explicitly the traffic intensities, I, come from the set, $\{1, 1.25, 1.5, \ldots, 2.75, 3\}$X for the three sectors, ZMP12, ZMP16, and ZMP42. A description of the traffic and a primary conflict analysis of the considered airspaces was provided Chapter 3. The three sectors chosen for simulation are representative of distinct classes of airspaces commonly found in en route centers. ZMP42 is a high altitude sector with the most traffic flowing east-bound or west-bound; the majority of traffic in ZMP16 is beginning its initial descent into Minneapolis St-Paul International Airport; and ZMP12 is a large high-altitude sector with numerous crossing routes. Figure 61 illustrates each sector's location and traffic within the airspace.

The parameter or configuration space, $(I \times D^{min}_{sep} \times D^{r}_{sep} \times \delta t \times H_D \times H_R)$, for each traffic scenario and decision policy covers a wide range, from best-case to a slightly degraded

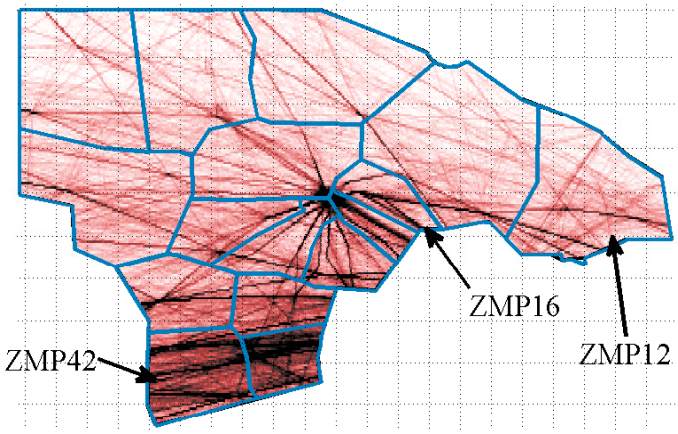


Figure 61: Traffic density map of ZMP center, highlighting sectors of interest.

version of current-day operations. The range of values for each parameter are as follows:

- $D_{sep}^{min} \in \{4, 5, 6, 7, 8, 9, 10\}$ NM
- $D_{sep}^{r} \in \{0, 1/6, 1/3, 1/2\}$ NM/min
- $\delta t \in \{0, 1, 2, 3, 4\}$ minutes
- $H_D \in \{0, 1, 2, 4, 6, 8, 10, 15, 20, \infty\}$ minutes
- $H_R \in \{0, 1, 2, 4, 6, 8, 10, 15, 20, \infty\}$ minutes

In addition to the three decision policies described in Chapter 5, the 5 policies in Appendix A are included as part of the simulations. A total of 45,662,400 simulations were run, covering a range of airspaces, traffic intensities, decision policies, and configurations.

CHAPTER VII

SIMULATION ANALYSIS

Using the abstracted models in Chapter 5, and the simulation environment in Chapter 6, the conflict-resolution taskload resulting from advisory conflict-detection and resolution tools are analyzed. The examination of conflict-resolution taskload focuses on a sensitivity analysis of the parameters describing the advisory decision-support tools.

The abstraction of conflict-detection systems considers how potential conflicts are identified. The corresponding parameters are D_{sep}^{min} and D_{sep}^{r}. The value D_{sep}^{min}, the aircraft spacing distance, defines a safety region around each aircraft that no other aircraft is allowed to penetrate. If two safety regions are projected to overlap in time, then a resolution command is advised to at least one of the aircraft. Uncertainty in trajectory prediction and the conflict-detection process is parameterized by D_{sep}^{r}, which describes the temporal growth rate of the safety regions.

The abstraction and parameterization of conflict-resolution tools considers the decision policy used to select which aircraft are maneuvered; how much information is used in the decision-making process; how long resolution commands are guaranteed conflict-free; and how often the conflict-resolution problem is solved. Example decision policies include a simple first-come, first-served policy, a policy that maneuvers randomly selected aircraft, and an optimal decision policy that uses the least possible number of resolution commands.

Adjusting each of the parameters (D_{sep}^{min}, D_{sep}^{r}, δt, H_D, and H_R) and changing the polices, simulations are executed at different traffic intensities. Following the completion of the simulations, the conflict-resolution taskload is extracted for each traffic scenario and configuration, and scaled according to the traffic intensity (see Hypothesis 1 in Chapter 2). Aggregate statistics are computed.

The next section explores basic relationships between each parameter and the conflict-resolution taskload. A series of sensitivity analysis studies are then performed. The goal is

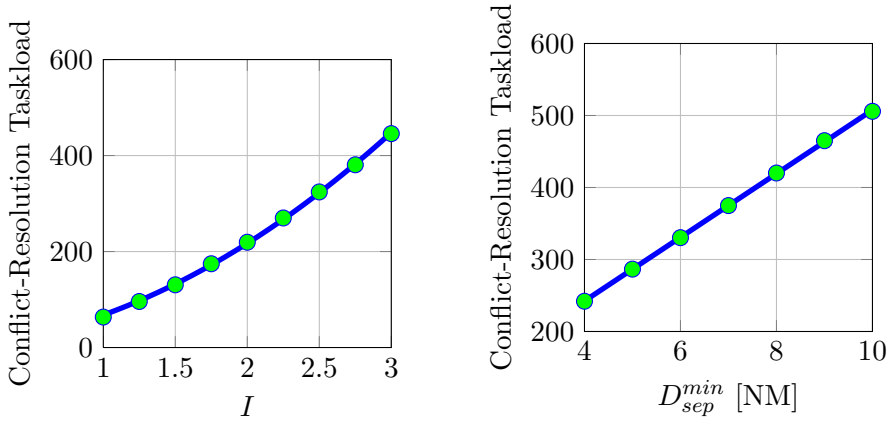
to extract best practices, enabling the design of advisory conflict-detection and resolution tools.

The first sensitivity analysis seeks to understand the potential benefits of improving a single parameter. Understanding the potential impact of a single parameter and its behavior with regards to conflict-resolution taskload, helps motivate the subsequent studies. In Section 7.3, a statistical comparison of decision-policies is performed to understand their relationships to conflict-resolution taskload. The next section addresses the relative value of reduced aircraft spacing as compared to improved conflict-resolution tools. Section 7.5 follows with a study that considers the benefit of long-term guaranteed conflict-free resolution commands. Finally, the last analysis examines how uncertainty and solve-times should be managed. Based on results from each of the studies, suggestions for the design of advisory conflict-detection and resolution tools are provided in Section 7.7.

7.1 Aggregate Behaviors

In this section a preliminary examination of the conflict-resolution taskload is provided to motivate future studies. Isolating an individual parameter (i.e. I, D_{sep}^{min}, D_{sep}^{r}, δt, H_D, and H_R), the expected conflict-resolution taskload $E[N]$ is plotted as a function of the considered term.

Traffic Intensity and Aircraft Spacing. Figure 62 illustrates the quadratic and linear behavior of the expected conflict-resolution taskload as a function of the traffic intensity, I, and the aircraft spacing, D_{sep}^{min}. Regardless of the policy, sector, or remaining configuration description, the quadratic and linear models hold. Averaging 50 simulation runs for each configuration and applying a least-squares fit, errors between predicted averages and simulation averages closely match. Table 4 contains the worst-case average and maximum errors over all implementations, policies, sectors, and traffic intensities. As indicated in the table, the quadratic model ($E[N] = cI^2$) for predicting the conflict-resolution taskload as a function of traffic intensity has a maximum error of 5.6 resolution commands. Likewise, the linear model for D_{sep}^{min} ($E[N] = cD_{sep}^{min}$) is always within 1.8 resolution commands



(a) Traffic Intensity

(b) Aircraft Separation

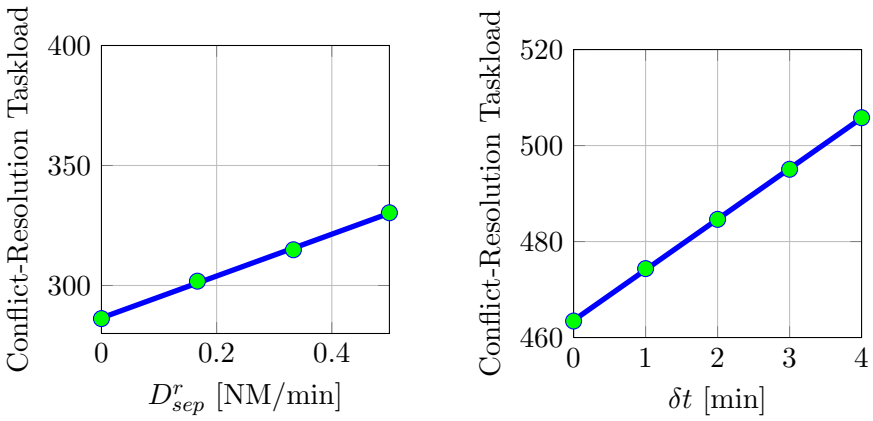
Figure 62: Qualitative behavior of conflict-resolution taskload with regards to the traffic intensity and aircraft spacing.

Table 4: Worst-case model errors between simulation averages and predicted averages over all configurations, traffic intensities, and sectors.

Parameter	Max Abs. Error	Mean Abs. Error	Max % Error	Mean % Error
I	5.6	3.19	8.3%	4.4%
D_{sep}^{min}	1.8	0.2	1.1%	0.2%
D_{sep}^{r}	11.6	1.3	6.8%	1.4%
δt	4.7	0.4	1.5%	0.3%

of the expected conflict-resolution taskload generated by the simulations. The model describing the conflict-resolution taskload according to the intensity, I, and aircraft spacing, D_{msep}^{min}, remains the same with controlled or uncontrolled traffic. Recall that a prior study in Chapter 3 demonstrated that the number of uncontrolled conflicts can be described by the model $cI^2D_{sep}^{min}$, where c is a sector specific constant.

Uncertainty and Solve-Times. Isolating the terms D_{sep}^{r} and δt individually, the expected conflict-resolution taskload grows linearly with each parameter, as shown in Figure 63. The corresponding models predicting the conflict-resolution taskload are $E[N] = c_1 + c_2 D_{sep}^{r}$ and $E[N] = c_3 + c_4\delta t$. For some implementations, the slope of the line of best-fit for the conflict-resolution taskload approaches or equals zero. That is, $c_2 \sim 0$ or $c_4 \sim 0$. Later studies indicate that the terms D_{sep}^{r} and δt are coupled. When δt is small, the effect of uncertainty on the conflict-resolution taskload is small, and vice versa. The linear models



(a) Prediction Uncertainty Growth Rate

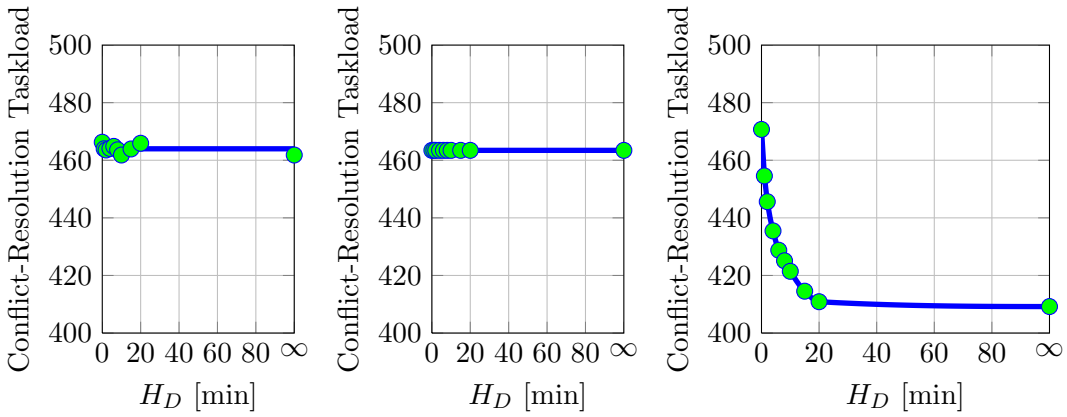
(b) Solve-Time

Figure 63: Qualitative behavior of conflict-resolution taskload with regards to the uncertainty and solve-time.

match closely to the expected conflict-resolution taskload of the simulations, as indicated by the small errors in Table 4.

Decision-Horizon Time. The decision-horizon H_D, exhibits different behaviors in comparison to the other parameters. For the decision-horizon time, when using a rule-based heuristic policy with conflict-priority ordering, the conflict-resolution taskload is independent of H_D, as shown by the constant trend-lines in Figure 64a and Figure 64b. The independence between H_D and the conflict-resolution taskload is apparent through examination of the procedures describing rule-based policies. For the rule-based heuristic policies with conflict-priority ordering, each potential conflict is considered independently of others; the selection of which aircraft to maneuver does not depend on the decision-graph. Thus, the size of the decision-graph, dictated by H_D, does not change which aircraft are maneuvered. In contrast, the minimum conflict-resolution taskload policy (MCRT) does not exhibit this property, as shown in Figure 64c. With larger values of H_D, the decision-graph increases in size, thereby providing the policy with more information to make optimal decisions. There is however a limitation to the benefit of increasing H_D. In fact, the reduction in conflict-resolution taskload, $E[N]$, decreases as $H_D \to \infty$. And furthermore,

$$\lim_{H_D \to \infty} \frac{\delta E[N]}{\delta H_D} \to 0 \tag{28}$$

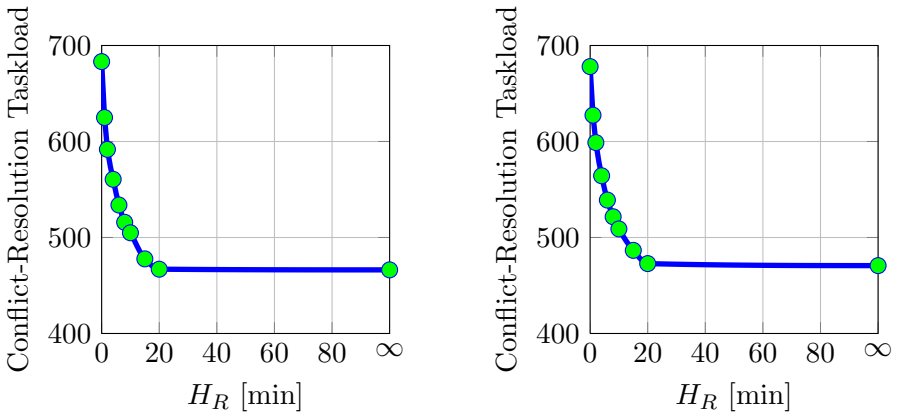


(a) Random (b) First-Come First-Serve (c) Minimum Conflict-Resolution Taskload

Figure 64: Qualitative behavior of conflict-resolution taskload with regards to the decision-horizon time.

Conflict-Free Resolution Time. The plots in Figure 65 demonstrate the value of improving conflict-resolution algorithms through the parameter H_R. A larger value of H_R increases the conflict-free resolution time, which prevents secondary conflicts. The plots represent the random policy and MCRT policy for the same setting and parameter configuration $(I \times D^{min}_{sep} \times D^{r}_{sep} \times \delta t, H_D)$. For both decision policies, as H_R increases, the expected conflict-resolution taskload decreases. Similar to the previous case with H_D, there are diminishing returns on improving the conflict-free time. It is worth noting that for the same configuration, the conflict-resolution taskload for both the random policy and the MCRT policy closely match. Minor differences in the conflict-resolution taskload between the optimal and random polices suggests that the choice in decision-policy has limited effect on the conflict-resolution taskload. This topic is later explored in Section 7.3.

Complete Conflict-Resolution Taskload Model. Based on the relatively simple relationships indicated by Figure 62 and Figure 63, there is a temptation to formulate the polynomial model, $E[N] = f(I, D^{min}_{sep}, D^{r}_{sep}, \delta t)$ around constant values of H_D and H_R,



(a) Random

(b) Minimum Conflict-Resolution Taskload

Figure 65: Qualitative behavior of conflict-resolution taskload with regards to conflict-free resolution time.

where

$$\begin{aligned} f(I, D^{min}_{sep}, D^{r}_{sep}, \delta t) = {} & c_1(I^2)(D^{min}_{sep})(D^{r}_{sep}) + c_2(I^2)(D^{min}_{sep})(D^{r}_{min})(\delta t) \\ & + c_3(I^2)(D^{r}_{min}) + c_4(I^2)(D^{r}_{min})(\delta t) \\ & + c_5(I)(D^{min}_{sep})(D^{r}_{min}) + c_6(I)(D^{min}_{sep})(D^{r}_{min})(\delta t) \\ & + c_7(I)(D^{min}_{sep}) + c_8(I)(D^{min}_{sep})(\delta t). \end{aligned} \tag{29}$$

A complete model enables prediction of the conflict-resolution taskload for any given traffic scenario and conflict-detection and resolution system. However, besides providing the conflict-resolution taskload, such a complete model provides little insight. Because traffic in each sector is unique (as demonstrated in Chapter 3), any best-fit coefficients are not consistent across airspaces. Therefore, study of the best-fit coefficients for the complete model does not allow for more general conclusions on the relationship between the parameters and the conflict-resolution taskload. Furthermore, if the conflict-resolution taskload model in Equation 29 is expanded to consider H_D and H_R, then a simple polynomial model fails. A polynomial model is unable to capture the asymptotic behavior of the conflict-resolution taskload with regards to the parameters H_D and H_R. Also, the relationships indicated in Figure 64c and Figure 65 do not correspond to simple functions (e.g. $1/x$, e^{-x}). Certainly, it is possible to truncate the results over the range $H_D \in [0, 20]$ minutes and $H_R \in [0, 20]$

minutes for model fitting. Doing so allow a 4^{th}-order polynomial in H_D and H_R to describe the conflict-resolution taskload; lower order polynomials result in a poor fit. A complete model that considers all parameters, including H_D and H_R over the truncated domain, results in a 13 degree polynomial. Such a high-degree polynomial makes drawing conclusions from the coefficients increasingly difficult, even when focusing on a single sector.

Instead of focusing on mathematical models to prediction the conflict-resolution taskload, alternative methods for understanding the relative importance of each parameter are preferred.

7.2 Initial Parameter Sensitivity Analysis

An initial parameter sensitivity analysis was performed on the conflict-resolution taskload simulation data. The goal of the sensitivity analysis is to understand how improving one particular parameter can aid in reducing the conflict-resolution taskload. More specifically, what are the upper and lower bounds of improvement. Along these lines, the following example question can be asked:

> *If the conflict-free resolution time is improved from* $H_R = 0$ *minutes to* $H_R = \infty$ *minutes, then what is the greatest expected percent reduction and the smallest expected percent reduction in the conflict-resolution taskload, regardless of the remaining configuration* ($I \times D^{min}_{sep} \times D^{r}_{sep} \times \delta t \times H_D$)*?*

Replacing H_R with another parameter and updating the remaining configuration, answering a more generic form of the question above provides upper and lower bounds of improvement according to a single parameter.

To support the sensitivity analysis, some additional notation is required. Let $\mathcal{C}_P$ denote the subset of configurations with equal parameter values, excluding the parameter P. So for all $c_1, c_2 \in \mathcal{C}_P$, where

$$\begin{aligned} c_1 &= (I_1, D^{min}_{sep,1}, D^{r}_{sep,1}, \delta t_1, H_{D,1}, H_{R,1}), \\ c_2 &= (I_2, D^{min}_{sep,2}, D^{r}_{sep,2}, \delta t_2, H_{D,2}, H_{R,2}), \end{aligned} \tag{30}$$

it is required that all parameters are equal except the parameter corresponding to P. For example, C_{H_D} corresponds to the space of configurations in which the parameter H_D varies, while all others remain constant.

Following this notation, $E[N]_{C_{H_D}}$, denotes the expected conflict-resolution taskload as a function of H_D, for a specific configuration with constant values of I, D^{min}_{sep}, D^{r}_{sep}, δt, H_R, and a set decision policy.

For the more generic question, the greatest possible improvement and the smallest possible improvement for an arbitrary parameter P are given by

$$R^{max}_P = \max_{c_P \in C_P} \left(\frac{\max(E[N]_{c_P}) - \min(E[N]_{c_P})}{\max(E[N]_{c_P})} \right) \tag{31}$$

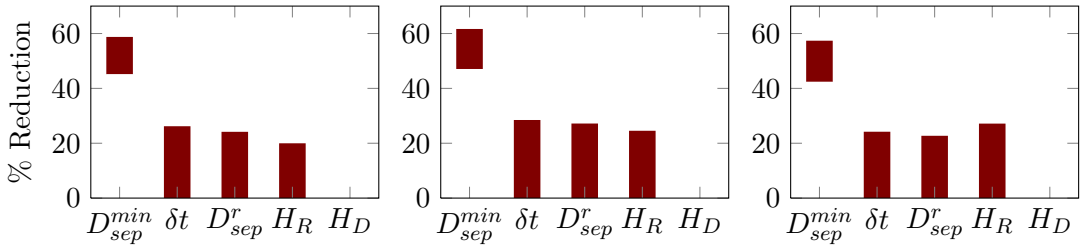
and

$$R^{min}_P = \min_{c_P \in C_P} \left(\frac{\max(E[N]_{c_P}) - \min(E[N]_{c_P})}{\max(E[N]_{c_P})} \right), \tag{32}$$

respectively. The values R^{max}_P and R^{min}_P are henceforth referred to as the dynamic range of improvement for the parameter P.

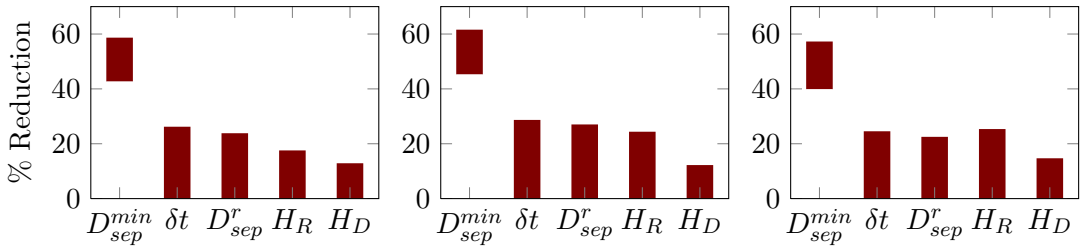
The dynamic range for the parameters D^{min}_{sep}, D^{r}_{sep}, δt, H_R, and H_D are shown in Figure 66 and Figure 67 for the FCFS and MCRT policies over the sectors ZMP12, ZMP16, and ZMP42. All parameters consider the complete range of simulation values (see Section 6.3), except for D^{min}_{sep}. Instead, for D^{min}_{sep}, the potential improvement from changing aircraft spacing from 9 NM to 4 NM is tested to better reflect the change from current operations to future operations. The dark bars indicate the range of improvement for a single variable. For example, for the FCFS policy in ZMP12, improvement of the conflict-free time, H_R yields a maximum reduction in conflict-resolution taskload between $R^{min}_{H_R} = 0\%$ and $R^{max}_{H_R} = 20\%$.

The dynamic ranges in Figure 66 and Figure 67 indicate that the largest reduction in conflict-resolution taskload comes from reducing aircraft spacing. The best-case improvements show that reducing aircraft spacing can decrease the conflict-resolution taskload approximately 60%, regardless of the sector or decision policy. Moreover, even the lower bound of improvement for D^{min}_{sep} is large, ($R^{min}_{D^{min}_{sep}} \sim 45\%$), and exceeds the best-case improvement for any other parameter.



(a) ZMP12 (b) ZMP16 (c) ZMP42

Figure 66: FCFS parameter sensitivity analysis: Range of percent reduction in conflict-resolution taskload.



(a) ZMP12 (b) ZMP16 (c) ZMP42

Figure 67: MCRT parameter sensitivity analysis: Range of percent reduction in conflict-resolution taskload.

Compared to each other, the remaining parameters δt, D^r_{sep}, and H_R, show similar reductions in conflict-resolution taskload. However, there are variations between the sectors. For example, the improvement of increasing H_R shows greatest potential benefit in sector ZMP42, over δt, and D^r_{sep}. That is, $R^{max}_{H_R} \geq R^{max}_{\delta t} \geq R^{max}_{D^r_{sep}}$. However, the ordering of the best-case improvements is not the same for ZMP12 and ZMP16. For the MCRT and FCFS policies, Figure 66 and Figure 67 show that $R^{max}_{\delta t} \geq R^{max}_{D^r_{sep}} \geq R^{max}_{H_R}$ for ZMP12 and ZMP16.

For both policies, and all sectors, the sensitivity analysis indicates that the decision-horizon time, H_D, has the least impact on the conflict-resolution taskload. In the case of the FCFS policy, $R^{max}_{H_D} = 0$, indicating that the decision-horizon has no effect on the conflict-resolution taskload. This result is the same result illustrated in Figure 64b of Section 7.1.

Perhaps more interesting is that there exist configurations where improvement to a single parameter does not reduce the conflict-resolution taskload. For $P = \delta t$, D^r_{sep}, H_R, and H_D, the lower bound improvement is $R^{min}_P = 0$ for both policies across all three sectors.

Such a result is surprising. However, it might indicate that at lower traffic levels, the benefits of conflict-resolution tools are limited. Furthermore, optimizing the conflict-resolution taskload over a subset of parameters allows others to be less than ideal. For example, if $H_R = 20$ minutes and $\delta t = 0$, then an uncertainty growth in trajectory prediction given by $D^r_{sep} = 0.5$ NM/min might be inconsequential to the conflict-resolution taskload. Thus, such a result might imply that instead of improving trajectory-prediction tools, systems can be designed to quickly detect and resolve potential conflicts to overcome limitations as a result of uncertainty.

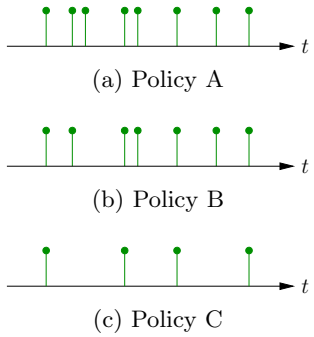
7.3 Policy Comparison

Results in Section 7.1 indicate the potential for only limited gains when implementing the MCRT policy over other rule-based heuristic policies (recall the comparison of the conflict-resolution taskload curves in Figure 65). In this section, a more detailed consideration of the expected conflict-resolution taskload as a function of the decision policy is taken. Using statistical tests, A-B comparisons are made to determine the value of minimizing the conflict-resolution taskload. Ultimately, hypothesis testing demonstrates that the decision policy has at most a minor effect on the conflict-resolution taskload.

The first set of tests considers the conflict-resolution taskload over the 10 hour time period for each of the 50 traffic scenario. Using analysis of variance (ANOVA) and Tukey's Honestly Significant Difference (HSD) test, multiple comparisons are made to test if there is a statistical difference between any two decision policies for a set configuration [82]. The degrees of freedom for Tukey's HSD test is 392 ($8 \times 50 - 8$). For a specific configuration ($I \times D^{min}_{sep} \times D^r_{sep} \times \delta t \times H_D \times H_R$), let N^A_i and N^B_i denote the conflict-resolution taskload for traffic scenario i, and decision policies A and B. The mean conflict-resolution taskload is given by $E[N^A]$ and $E[N^B]$. The Tukey's HSD test ($\alpha = .05$) is applied with the following hypothesis between each pair of decision policies :

$$H_o: \quad E[N^A] = E[N^B]$$
$$H_a: \quad E[N^A] \neq E[N^B]$$

When the null hypothesis, H_o, is accepted, then the conflict-resolution taskloads of the



(a) Policy A

(b) Policy B

(c) Policy C

Figure 68: Representative timeline of conflict-resolution advisories.

two policies are similar. Otherwise, acceptance of the alternative hypothesis implies that the policies result in different conflict-resolution taskloads.

ANOVA and Tukey's HSD testing requires that the observed conflict-resolution taskload data come from normal distributions. Additionally, it is required that data samples are independent from each other, i.e. N_i^A is independent from N_j^A for $i \neq j$. Both properties are assumed to be satisfied.

As an example, consider the representative timelines in Figure 68 for three arbitrary decision policies. Each pulse along the timelines indicates that a resolution command is advised. When the conflict-resolution taskloads are similar, the number of pulses closely match, indicating no significant difference between the two policies. For the advisory timelines of Policy A and Policy B, matched testing would indicate no difference between the policies' conflict-resolution taskload. However, if there is a significant difference in the conflict-resolution taskload, as illustrated by the differences between Policy B and Policy C, then hypothesis testing rejects the null hypothesis.

The hypothesis testing is applied between all policies, matching traffic configurations and scenarios. A summary of the test results is provided in Table 5, Table 6, and Table 7 for sectors ZMP12, ZMP16, and ZMP42, respectively. Policies 1-7 correspond to rule-based heuristic policies, while Policy 8 corresponds to the MCRT policy. In each table entry, the total number of configurations that are statistically different according to the testing procedure is listed. As a reference, there are 48,384 configurations for each policy and sector covering a range of values for the settings I, D_{sep}^{min}, D_{sep}^{r}, δt, H_D, and H_R.

Table 5: Policy Comparisons: ZMP12

Policy	1	2	3	4	5	6	7	8
1	-	0	0	0	0	0	0	**1580**
2	-	-	0	0	0	0	0	**2665**
3	-	-	-	0	0	0	0	**2665**
4	-	-	-	-	0	0	0	**2379**
5	-	-	-	-	-	0	0	**1616**
6	-	-	-	-	-	-	0	**2382**
7	-	-	-	-	-	-	-	**2268**
8	-	-	-	-	-	-	-	-

Table 6: Policy Comparisons: ZMP16

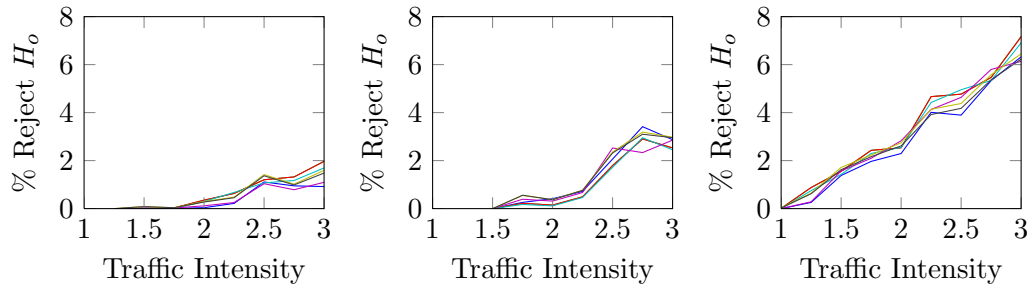
Policy	1	2	3	4	5	6	7	8
1	-	0	0	0	0	0	0	**4715**
2	-	-	0	0	0	0	0	**3913**
3	-	-	-	0	0	0	0	**3913**
4	-	-	-	-	0	0	0	**3810**
5	-	-	-	-	-	0	0	**4400**
6	-	-	-	-	-	-	0	**4944**
7	-	-	-	-	-	-	-	**4869**
8	-	-	-	-	-	-	-	-

A major result of the analysis demonstrates that there is no statistical difference between any of the rule-based heuristic policies. As shown in the three tables, between the seven policies, all of the configurations accept the null hypothesis. Based on this result, the following conjecture is made:

Conjecture 1. *Excluding cooperative conflict-resolution, there is no significant difference in the conflict-resolution taskload between any two rule-based heuristic policies.*

Table 7: Policy Comparisons: ZMP42

Policy	1	2	3	4	5	6	7	8
1	-	0	0	0	0	0	0	**12322**
2	-	-	0	0	0	0	0	**14261**
3	-	-	-	0	0	0	0	**14261**
4	-	-	-	-	0	0	0	**13861**
5	-	-	-	-	-	0	0	**13298**
6	-	-	-	-	-	-	0	**13508**
7	-	-	-	-	-	-	-	**12937**
8	-	-	-	-	-	-	-	-

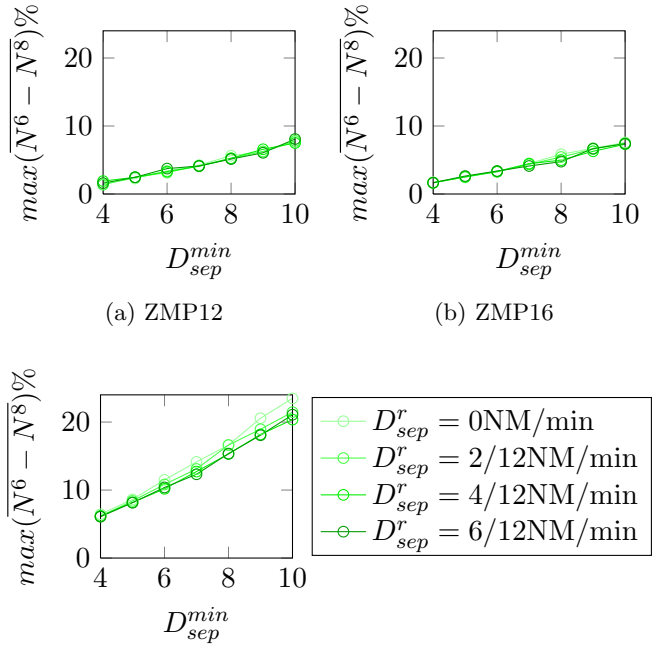


(a) ZMP12 (b) ZMP16 (c) ZMP42

Figure 69: Percent of configurations for which the MCRT policy is statistical different from a rule-based heuristic policy (Reject H_o).

The hypothesis testing reveals that in many cases there is a statistically relevant difference between the seven rule-based heuristic policies and the MCRT policy (see the last column in each table). However, this statistical difference does not exist for all configurations. In fact, the number of configurations that accept the alternative hypothesis grows with traffic intensity. Figure 69 illustrates this point. Each line in the plots represents one of the seven rule-based heuristic policies. The trend lines show that as the traffic intensity increases, a greater fraction of configurations become statistically different from the MCRT policy. While not shown, a similar trend exists as the aircraft spacing distance increases. The implication of both results is that the MCRT policy becomes progressively more effective as the number of potential conflicts increases.

While the MCRT policy is deemed statistically different from the other policies, the exact benefit is not yet stated. Figure 70 illustrates the maximum difference in expected conflict-resolution taskload between the MCRT policy and the random policy. Each line corresponds to a subset of configurations parametrized by the value D^r_{sep}, and considers every possible implementation of $(\delta t \times H_D \times H_R)$ at the 3X traffic level. According to the results in Figure 70, for ZMP12 and ZMP16, the largest average difference in conflict-resolution taskload between the MCRT policy and the random policy is 9 less maneuvers over a 10 hour time period. For ZMP42, the reduction is just over 10 when aircraft are spaced at 6NM. Such a reduction only accounts for an average difference of 1 less maneuver per hour. Considering that there is far greater variance in conflict-resolution taskload



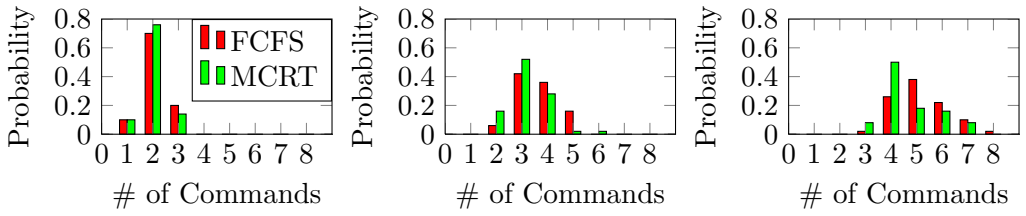
(a) ZMP12 (b) ZMP16

(c) ZMP42

Figure 70: Maximum difference in the expected conflict-resolution taskload between the MCRT policy and the random policy over a range of configurations.

per hour ($5 \leq \sigma^2 \leq 30$), it is unlikely an air traffic controller would notice such a small reduction in the conflict-resolution taskload. At lower traffic intensities, the differences in the conflict-resolution taskload, even when statistically different, are marginal. Overall, when comparing the maximum difference in conflict-resolution taskload between the MCRT policy and any other policy, the potential benefits appear limited.

A similar analysis can be extended to shorter time-windows of 2.5 minutes. While 10 less resolution commands over 10 hours may appear insignificant, if the deductions are concentrated during a short-burst of activity, then there might be some benefit of the MCRT policy in reducing controller workload. Figure 71 contains example distributions of the number of resolution commands used to space aircraft during the busiest 2.5 minute time periods for the 1X, 2X, and 3X traffic levels over ZMP42 (configuration: $D_{sep}^{min} = 6$, $D_{sep}^{r} = 0$, $\delta t = 0$). Using a two-sample Kolmogorov-Smirnov goodness-of-fit test (KS test) with $\alpha = .01$, the conflict-resolution taskload distributions are tested to see if they come from different populations. Accordingly, the following null hypothesis and alternative



(a) 1X: Accept H_o (b) 2X: Accept H_o (c) 3X: Reject H_o

Figure 71: Probability distributions of the conflict-resolution taskload for the busiest 2.5 minute time period in each traffic scenario.

hypothesis are set:

H_o: Samples are drawn from the same distribution

H_a: Samples are drawn from different distributions

If the null hypothesis is accepted, then there is no distinction in the conflict-resolution taskload between the two policies.

Application of the KS test reveals that there is no statistical difference between the MCRT policy and the random policy at the 1X and 2X traffic intensities for the configuration corresponding to Figure 71. However, at the 3X traffic level, the null hypothesis is rejected, implying that there is a difference between the distributions of the conflict-resolution taskload. Figure 71c illustrates that the distribution of the conflict-resolution taskload for the MCRT policy is skewed to the left in comparison to the random policy.

When applying the same testing procedure to ZMP12 and ZMP16, there are no differences found for any configurations with $D_{sep}^{min} \leq 6$ NM, regardless of the traffic intensity. In the case of ZMP42, the only configurations for which the MCRT policy outperforms the random policy is when the solve-times is $\delta t = 0$ minutes. A complete list of configurations for which the null hypothesis of the KS test is rejected at the 3X traffic level, when $D_{sep}^{min} \leq 5$ NM, is provided in Table 8. Also included as part of the table is the probability that the MCRT policy reduces the conflict-resolution taskload by at least two resolution commands.

For most cases when the null hypothesis is rejected, the guaranteed conflict-free time, H_R, is greater than or equal to 15 minutes. Likewise, the decision-horizon times are large. By current standards of performance, $H_R = 15$ minutes is quite large, especially when one considers that the look ahead time for potential conflicts is typically greater than 5 minutes

Table 8: Configurations with $D_{sep}^{min} \leq 5NM$ for which there is a statistical difference in the conflict-resolution taskload distribution over 2.5 minutes for ZMP42

D_{sep}^{min} [NM]	D_{sep}^{r} [NM/min]	δt [min]	H_D [min]	H_R [min]	$P(\lvert N_i^A - N_i^B \rvert \geq 2)$
5	0	0	10	15	0.18
5	0	0	15	15	0.10
5	0	0	15	20	0.16
5	0	0	15	100	0.10
5	0	0	20	100	0.14
5	0.167	0	15	20	0.18
5	0.167	0	15	100	0.16
5	0.167	0	20	100	0.12
5	0.333	0	15	15	0.10
5	0.500	0	8	10	0.12
5	0.500	0	10	20	0.12
5	0.500	0	15	20	0.14

(i.e. $T_R \geq 5$ minutes). State-of-the-art trajectory prediction tools are optimistically capable of detecting potential conflicts 20 minutes into the future [5]. It is also worth noting that the solve-time of $\delta t = 0$ refers to event-based conflict-detection and resolution implementation. For event-based implementations, as soon as a potential conflict is identified, it is immediately resolved. For many conflict-resolution algorithms, the ability to quickly generate resolution commands for aircraft, especially when the decision-horizon and conflict-free times are large, poses quite a challenge.

According to Table 8, there is only a marginal fraction of traffic scenarios (< 0.20) for which the MCRT policy reduces the number of advisory commands by two or more. Additional analysis indicates that between 35%-53% of the traffic scenarios show no improvement in conflict-resolution taskload across the configurations.

When considering traffic intensities less than 3X, there are no cases with $D_{sep}^{min} \leq 5$ NM and $H_R \leq 20$ minutes for which the MCRT policy has a statistically different conflict-resolution taskload distribution from the random policy. Thus it can be argued that the MCRT policy does not pose a substantial benefit over any other rule-based heuristic policy.

Differences between the other rule-based polices and the MCRT policy reflect the findings above. Furthermore, extending the KS test to the other rule-based policies, there is no statistical difference found between them.

Working with the assumption that improvements in position sensing and trajectory prediction, enabled by ADS-B, will allow for reductions in aircraft spacing ($D_{sep}^{min} \leq 5$ NM) and uncertainty, and that complete airspace redesigns will occur before 3X traffic, an even stronger conjecture can be made:

Conjecture 2. *There are no significant differences in the conflict-resolution taskload between any two decision-policies (excluding cooperative conflict-resolution).*

The previous conjecture can be applied to understanding the value of the decision-horizon time, H_D. As it stands, the MCRT policy is the only policy that benefits by extending the decision-horizon time. In contrast, an increase in the value of H_D has no effect on conflict-resolution taskload of the rule-based heuristic policies with conflict-priority ordering. Differences in the conflict-resolution taskload between the MCRT policy and any other policy is small for the same configuration (regardless of the value of H_D). Thus, it can be stated that any increase in H_D, while potentially resulting in statistical differences over 10 hour time windows, does not effectively reduce the conflict-resolution taskload. Hence the corollary below.

Corollary 1. *The air traffic controller is unlikely to detect any benefit in reduced conflict-resolution taskload due to an increase in the decision-horizon time, H_D.*

The major result of this section is that the decision policy and the decision-horizon time has limited or imperceivable effects on the conflict-resolution taskload of an advisory decision-support tool.

7.4 Relative value between D_{sep}^{min} and H_R

The preliminary results in Section 7.2 indicate that a decrease in aircraft spacing can yield far greater reductions in conflict-resolution taskload in comparison to adjusting other parameters. As it stands, next generation air traffic systems (e.g. NextGen and SESAR) all include the introduction of Automatic Dependent Surveillance-Broadcast (ADS-B) to report aircraft positions using global positioning systems (GPS). Furthermore, improved trajectory prediction and 4D trajectory-based operations are considered imperatives to managing

congestion and increasing throughput, especially in terminal areas [38, 46, 90]. Researchers hypothesize that implementing ADS-B and improving trajectory predictions will allow for closer aircraft spacings [31, 81]. Thus, it is likely that the reductions in conflict-resolution taskload identified in Section 7.2 will one day be realized. However, there are still unanswered questions concerning the relative importance of reducing aircraft spacing in comparison to the introduction and improvement of conflict-resolution algorithms. More specifically, how should a research and development portfolio be developed to provide equitable support according to potential benefits? In this section the relative value of D_{sep}^{min} is compared to H_R in reducing conflict-resolution taskload.

The value of improved trajectory prediction and position measurement systems enabling a reduction in aircraft spacing is tested through the parameter D_{sep}^{min}. Larger values of D_{sep}^{min}, beyond minimum required separations (e.g. 5 NM in current-day operations), represent increased uncertainty in the current and future positions of aircraft. This uncertainty often leads to excessive labeling of potential conflicts and to issuing resolution commands that promote overly-safe and conservative actions by air traffic controllers. The importance of conflict-resolution algorithms is evaluated through the parameter H_R. Increasing the value of H_R simulates improving the capability of conflict-resolution algorithms to ensure conflict-free trajectories for longer periods of time.

Because the previous section indicates that the value of H_D and the underlying decision policy is insignificant to determining the conflict-resolution taskload, this study strictly considers the FCFS policy and the configuration setting $H_D = 0$ and $\delta t = 0$. Later in Section 7.6, analysis demonstrates that the value of D_{sep}^{r} does not affect the conflict-resolution taskload when $\delta t = 0$. As such, the uncertainty parameter is arbitrarily set to $D_{sep}^{r} = 0$ in this section.

Applying the FCFS policy to 10 hours of traffic in ZMP42, the expected conflict-resolution taskload counts for three traffic intensities (1X, 2X, 3X) are depicted in Figure 72. As the traffic intensity increases, so does the expected conflict-resolution taskload across all values of H_R and D_{sep}; each surface representing a single traffic intensity is strictly greater than the previous. Figure 72 clearly illustrates that as aircraft spacing is reduced,

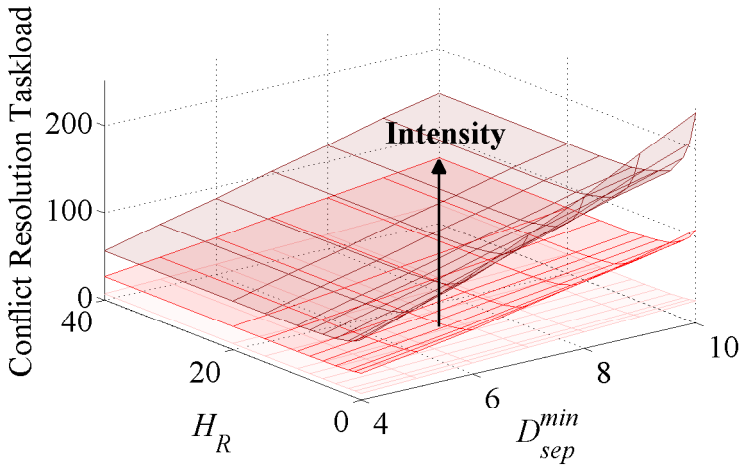


Figure 72: Expected number of advisory resolution commands for the FCFS policy at 1X, 2X, and 3X traffic intensities.

the conflict-resolution taskload decreases. Also, increasing the conflict-free time, H_R, for the conflict-resolution algorithm can significantly reduce the conflict-resolution taskload, albeit with diminishing returns.

To begin the study comparing the relative value of D_{sep}^{min} and H_R to the conflict-resolution taskload, a reference standard is required. Assume that air traffic controllers (with or without assistance from conflict-resolution tools) are able to issue resolution commands that are guaranteed conflict-free for an average of 4 minutes beyond the look ahead time T_R. Thus, if a potential conflict is identified to occur in 5 minutes, and it is immediately resolved, then the associated resolution command ensures that the aircraft is conflict-free for 9 minutes. Depending on the flight-plans of other aircraft, there is the potential for resolved trajectories to be conflict-free even longer.

The relative benefits of aircraft spacing compared to the advancement of conflict-resolution algorithms is assessed by determining the value of H_R required to achieve the same decrease in conflict-resolution taskload when aircraft spacing is reduced. For example, consider the following question:

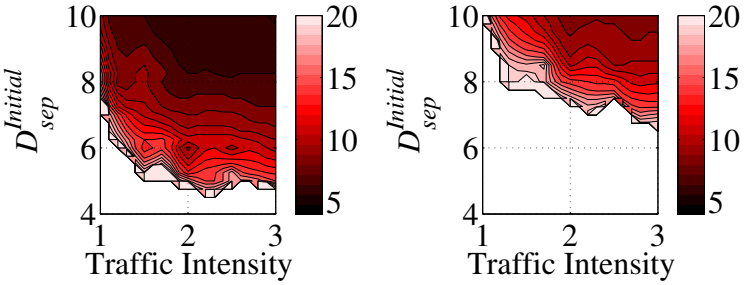
Let improved trajectory prediction and reporting allow aircraft spacing to decrease by ΔD_{sep}^{min}, from $D_{sep}^{initial}$ to D_{sep}^{final}. For resolution commands that are guaranteed to be conflict-free for $H_R = 4$ minutes, the expected conflict-resolution taskload decreases from $E[N^{initial}]$ to $E[N^{final}]$. What is the required conflict-free time, H_R, to achieve the same decrease in conflict-resolution taskload when aircraft spacing is held constant (i.e. $D_{sep}^{initial} = D_{sep}^{final}$)?

By evaluating the conflict-resolution taskload of the simulations and interpolating between results, the above question can be answered for a wide range of traffic intensities and initial and final aircraft spacings. These results are illustrated in Figure 73 and Figure 74 for sectors ZMP12 and ZMP42. As an example, from Figure 74b, to match the decrease in conflict-resolution taskload associated with a 0.5 NM reduction in aircraft spacing, from $D_{sep}^{initial} = 8$ NM to $D_{sep}^{final} = 7.5$ NM, at the 2.5X traffic intensity, a guaranteed conflict-free time of $H_R = 9.5$ minutes is required.

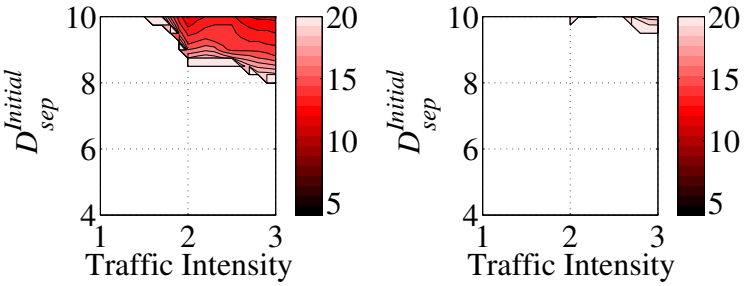
The figures illustrate that in order to match greater reductions in aircraft spacing, increasingly larger values of H_R are needed. For example, at the 3X traffic intensity, to match the decrease in conflict-resolution taskload when aircraft spacing is reduced by a 0.25 NM ($D_{sep}^{initial} = 8$ NM), a conflict-free time of $H_R = 6$ minutes is needed. However, for a larger decrease of 1 NM, a 20 minute conflict-free time is required.

Furthermore, as the reduction in aircraft spacing increases, from 0.25 NM to 1 NM, the space of intensities and initial aircraft spacings for which a matching value of H_R exists grows smaller. Notice that from Figure 73a to Figure 73d (and Figure 74a to Figure 74d), the no-solution whitespace grows larger and larger.

When slower solve-times are present for the conflict-detection and resolution process, the value of H_R becomes more relevant. Comparing Figure 75a and Figure 75b, for a slower solve-time given by $\delta t = 2$ minutes, the set of intensities and aircraft spacings for which improving the conflict-free time can match spacing reductions is larger than the $\delta t = 0$ case. This is also true when uncertainty is introduced, as illustrated in Figure 75d. However, that is not to say that increasing δt is beneficial. On the contrary, Section 7.1 shows that a greater

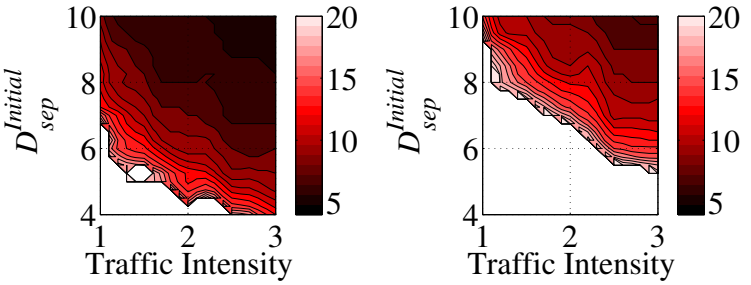


(a) Spacing reduction of 0.25 NM. (b) Spacing reduction of 0.50 NM.

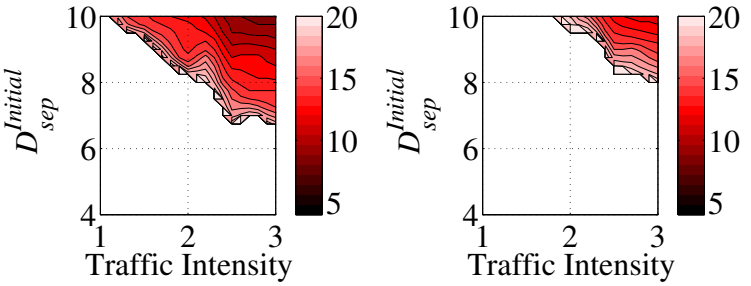


(c) Spacing reduction of 0.75 NM. (d) Spacing reduction of 1.0 NM.

Figure 73: Required value of H_R to match a reduction in aircraft spacing (ZMP12).

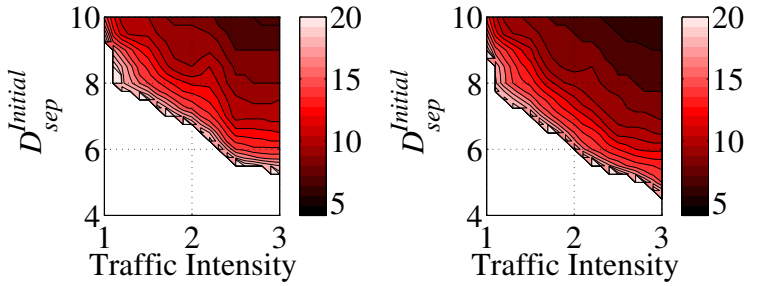


(a) Spacing reduction of 0.25 NM. (b) Spacing reduction of 0.50 NM.

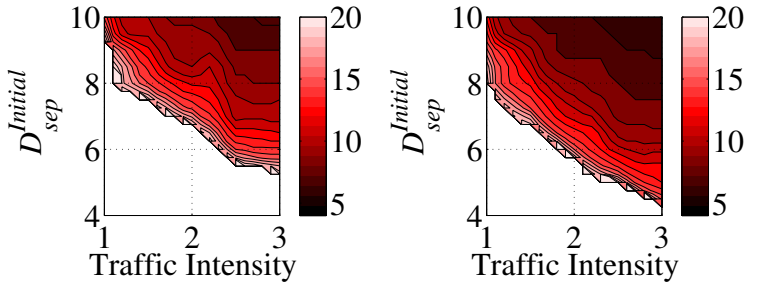


(c) Spacing reduction of 0.75 NM. (d) Spacing reduction of 1.0 NM.

Figure 74: Required value of H_R to match a reduction in aircraft spacing (ZMP42).



(a) $\delta t = 0$min, $D_{sep}^{r} = 0$ NM/min (b) $\delta t = 2$min, $D_{sep}^{r} = 0$ NM/min



(c) $\delta t = 0$min, $D_{sep}^{r} = 4/12$ NM/min (d) $\delta t = 2$min, $D_{sep}^{r} = 4/12$ NM/min

Figure 75: Required value of H_R to be equivalent to a reduction in the separation requirements of $\Delta D_{sep}^{min} = 0.5$ NM (ZMP42).

value of δt results in increased conflict-resolution taskload. The figures imply that if large uncertainty is present and solve-times are slow, then both a reduction in aircraft spacing and improved conflict-free times help to decrease the conflict-resolution taskload.

Overall, the results indicate that in order to reduce conflict-resolution taskload, the most effective approach is to reduce aircraft spacing. While a reduction in spacing is not a straightforward achievement, the introduction of ADS-B as part of next generation air traffic systems will surely aid in limiting sensor measurement noise in the position information provided to controllers. Additionally, the introduction of 4D trajectory-based operations in conjunction with performance standards to ensure aircraft stay on desired paths will provide a framework to improve trajectory projections.

Keeping all other variables constant, reductions in the effective aircraft spacing has a significant impact on conflict-resolution taskload. In fact, for the ZMP42 simulations, the reduction is linear over all intensities and conflict-free times H_R, as shown in Figure 72. When compared to a reference standard of $D_{sep}^{initial} = 9$ NM, the percent reduction in

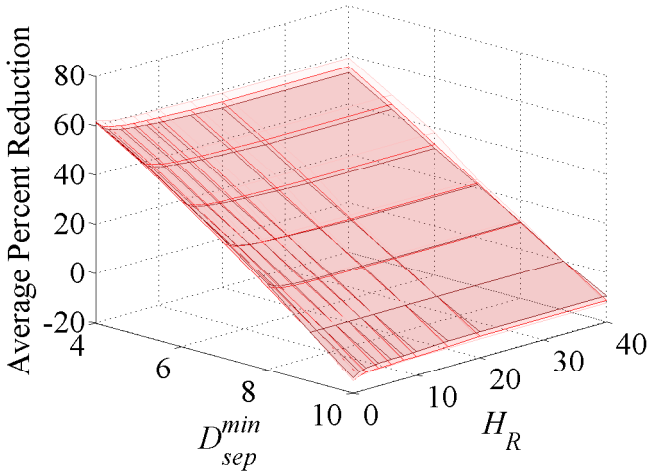
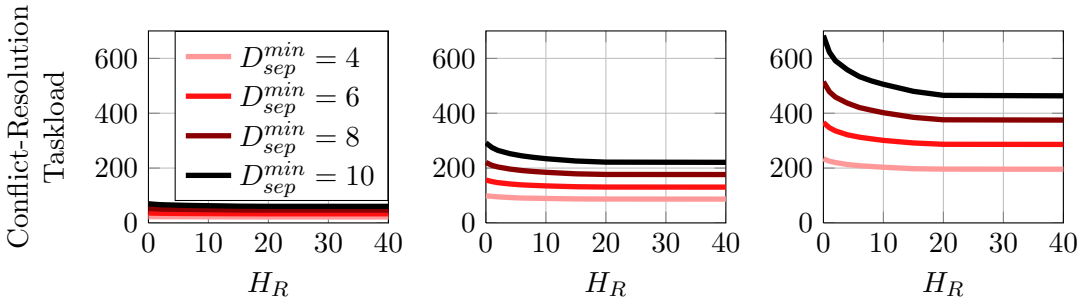


Figure 76: Percent decrease in the conflict-resolution taskload by reducing aircraft spacing when compared to $D_{sep}^{initial} = 9$ NM. (At the 1X, 2X, and 3X traffic intensities)

conflict-resolution taskload can be quite significant. According to Figure 76, the number of potential conflicts can be reduced approximately 60% by decreasing spacing standards to 4 NM, regardless of the value of H_R. This improvement is consistent for all traffic intensities, airspaces, and configurations.

The research presented here suggests that the consequences of improved trajectory prediction and position sensing for conflict detection can result in greater decreases in controller effort (implicitly through the reduction in conflict-resolution taskload) than the improvement of advisory systems for conflict-resolution. That is not to say that advisory conflict-resolution systems are not valuable, but rather, they are not the most promising approach for reducing conflict-resolution taskload. Thus, research, development, and policy efforts should place greater focus on gaining universal presence of ABS-B on all aircraft, and on improving aircraft trajectory predictions. In the next section, the benefits of conflict-resolution systems are studied to better understand what might be considered a satisfactory conflict-free resolution time.



(a) 1X Traffic (b) 2X Traffic (c) 3X Traffic

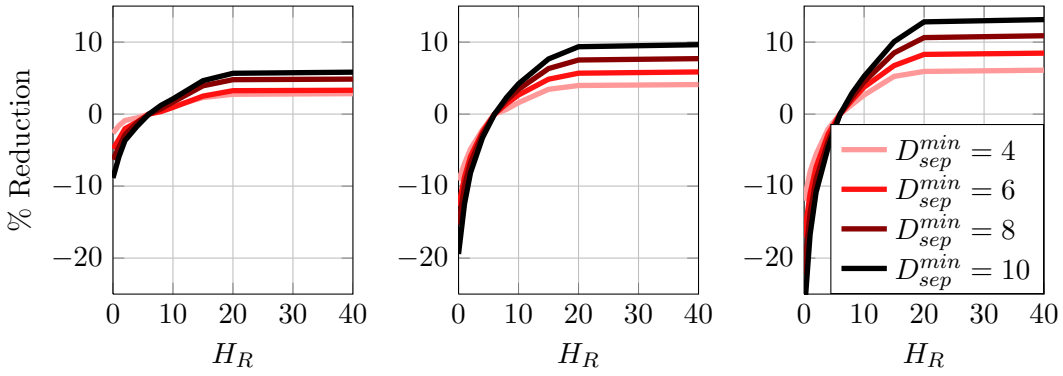
Figure 77: Conflict-resolution taskload for ZMP42 (FCFS, $\delta t = 0$, $D_{sep}^{r} = 0$).

7.5 Benefit of Improved Conflict-Resolution

While not nearly as effective as reducing aircraft spacing, the introduction and improvement of conflict-resolution algorithms can certainly reduce the conflict-resolution taskload. However, the potential gains are limited. Therefore, instead of striving to achieve the perfect conflict-resolution algorithm that can guarantee conflict-free travel for indefinite periods, it is advisable to design simple and robust algorithms that achieve a satisfiable conflict-resolution taskload. The study in this section focuses on the diminishing returns associated with the parameter H_R.

The conflict-resolution taskload for the FCFS policy with $\delta t = 0$ minutes is illustrated in Figure 77 for different traffic intensities and aircraft spacings. As the traffic intensity and aircraft spacing increases, the conflict-resolution taskload grows, regardless of the value of H_R. However, for a fixed traffic intensity and aircraft spacing, as the conflict-free time is increased, the controller receives fewer advisory resolution commands. Similar behavior is exhibited in sectors ZMP12 and ZMP16 as well.

Figure 78 illustrates the potential benefits of increasing the conflict-free time according to a reference standard of $H_R = 6$ minutes. For example, for a spacing requirement of 6 NM, improving the guaranteed conflict-free resolution time to 20 or more minutes yields a maximum reduction in conflict-resolution taskload of 8.5% at the 3X intensity level. This decrease corresponds to an average of 27 less resolution commands issued over the 10 hours time period. Smaller reductions occur at lower traffic intensities.



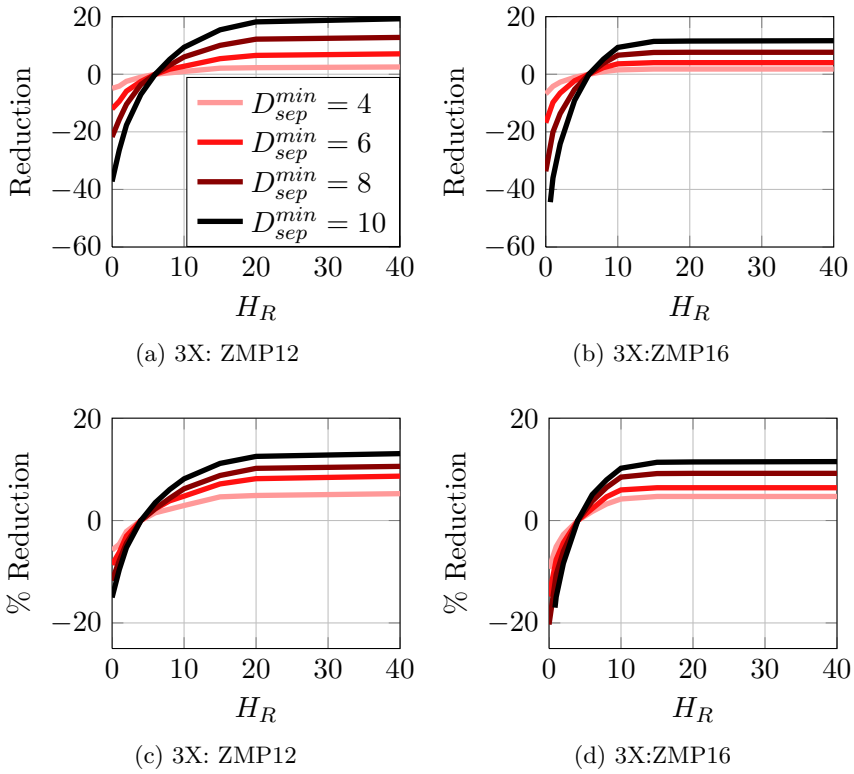
(a) 1X (b) 2X (c) 3X

Figure 78: Percent reduction in conflict-resolution taskload compared to $H_R = 6$ minutes (ZMP42).

At $D_{sep}^{min} = 4$ NM, the largest improvement from an increase in H_R is about a 6.1% decrease in conflict-resolution taskload, corresponding to an average reduction of 13 resolution commands over 10 hours. The decrease corresponds to 1 less resolution command every 50 minutes. From the perspective of the air traffic controller, this difference is probably imperceptible, especially because the variance in the conflict-resolution taskload within one hour is far greater. Similarly, diminishing returns exist for ZMP12 and ZMP16. Figure 79 illustrates that even at the 3X traffic level, for $D_{sep}^{min} = 6$ NM, improvement to the conflict-resolution algorithm only reduces the conflict-resolution taskload by 10 commands ($\leq 8\%$).

This analysis demonstrates that the benefit of increasing the conflict-free time to reduce controller taskload has diminished returns, especially for $H_R > 20$ minutes. One reason for the limited improvement associated with significant increases in H_R is that sector sizes are finite. Sectors are also designed to have intersecting regions spaced far apart from each other.

The distributions of aircraft transit times through the sectors are shown in Figure 80. For ZMP42, 65% of all aircraft take 20 minutes or less to pass through the airspace, and virtually all take less than 25 minutes. For ZMP16, the vast majority of aircraft exit the airspace within 10 minutes. In the context of transit times, the diminishing returns for $H_R \geq 20$ minutes for ZMP42 makes sense. Likewise, for ZMP16, because transits times tend to be less than 10 minutes, Figure 79b and Figure 79d illustrates that there is little



(a) 3X: ZMP12

(b) 3X:ZMP16

(c) 3X: ZMP12

(d) 3X:ZMP16

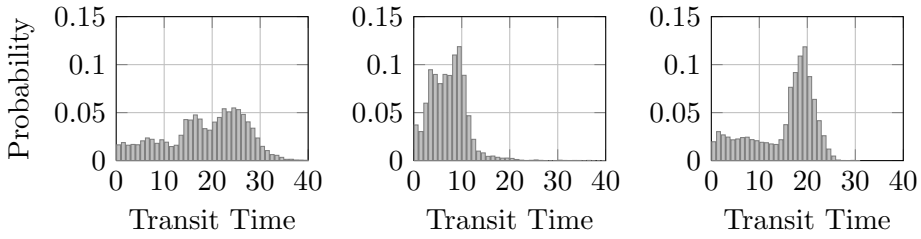
Figure 79: Relative reductions in conflict-resolution taskload for ZMP12 and ZMP16.

benefit to increasing H_R beyond this time.

The sector ZMP12 is a bit different from ZMP16 and ZMP42. Because of its size, a large portion of traffic takes more than 20 minutes to traverse the airspace, as shown in Figure 80a. However, the plots in Figure 79 show that the benefit of extending H_R beyond 15 or 20 minutes is relatively small.

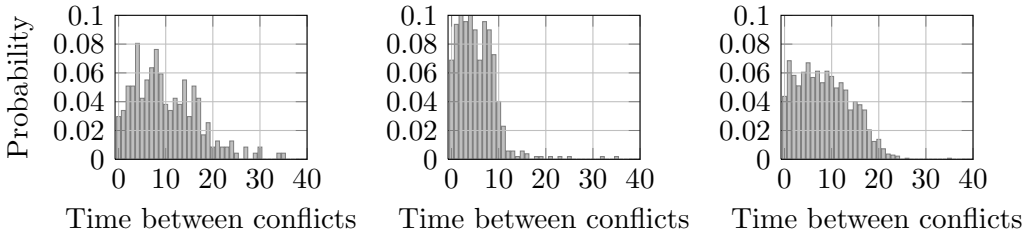
Understanding why a conflict-free time of $H_R = 15$ minutes is satisfactory for ZMP12 can best be understood through analysis of the inter-arrival conflict times. Figure 81 contains the distribution of inter-arrival times between an aircraft's first potential conflict and its last potential conflict within a sector. In the case of ZMP12, 91% of inter-arrival times are less than 20 minutes, despite the fact that only 50% of all aircraft traverse the sector in the same time period.

Comparing the three sectors, 90% of all inter-arrival conflict times occur within 20 minutes, 10 minutes, and 16 minutes for ZMP12, ZMP16, and ZMP42, respectively. Because each sector is designed according to a specific purpose, there is no one-size fits all value of



(a) ZMP12 (b) ZMP16 (c) ZMP42

Figure 80: Distribution of aircraft transit times [minutes] through sector (1 minute increments).



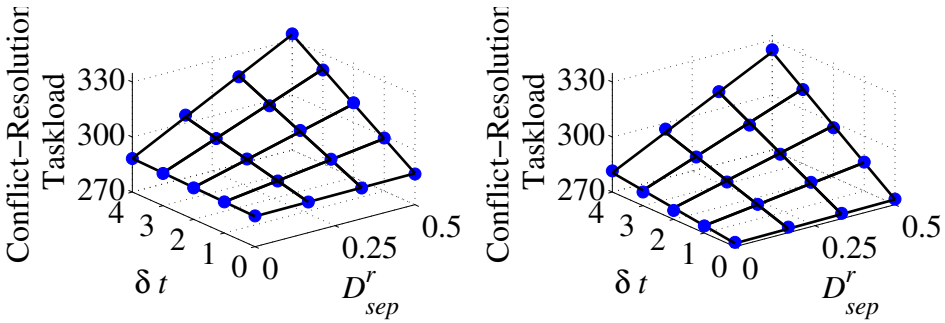
(a) ZMP12 (b) ZMP16 (c) ZMP42

Figure 81: Distribution of inter-arrival times [minutes] between the first and last potential conflict of an aircraft (1 minute increments).

H_R, however even with a value of $H_R = 6$ minutes the majority of secondary conflicts can be prevented.

7.6 Managing Uncertainty and Solve-Times

Uncertainty in the conflict-detection process can pose quite a challenge to air traffic controllers and advisory conflict-resolution systems. As a precautionary measure, the need to manage future workload sometimes prompts air traffic controllers to resolve potential conflicts that are unlikely to ever become realized [52]. In the case of future systems, with the introduction of trajectory-based operations and automated hand-offs, many routine and mundane tasks are offloaded to computer systems. This transference of function provides the air traffic controller with more time to dedicate to resolving potential conflicts. However, there is still a need to answer how often conflict-resolution problems should be resolved when uncertainty is present, and furthermore, when is it beneficial to sacrifice rapid solve-times for improved conflict-free resolution times.



(a) FCFS (b) MCRT

Figure 82: Conflict-resolution taskload as a function of the solve-time and uncertainty parameters.

For systems that simultaneously solve the conflict-resolution problem for multiple aircraft, computation times can grow quite large [100]. These slow solve-times prevent rapid information updates and imply that decisions are made on old information. In the case of uncertainty, when computation time prevents event-based resolution of potential conflicts, advisory systems are required to plan in advance and hedge accordingly - much like human air traffic controllers. Conservative behavior then results in an increase in the number of advisory resolution commands.

Study of the solve-times and uncertainty, parameterized by the terms δt and D^r_{sep}, demonstrate a coupling as shown in Figure 82. When either δt or D^r_{sep} is small, the effect of the other term on the conflict-resolution taskload is diminished. For a given airspace and decision-policy, with fixed values of traffic intensity and configuration $(D^{min}_{sep}, H_D, H_R)$, the expected conflict-resolution taskload over 10 hours can be modeled as a function of the uncertainty and solve-time parameters D^r_{sep} and δt:

$$E[N] = c_1 D^r_{sep}\delta t + c_2\delta t + c_3 D^r_{sep} + c4. \tag{33}$$

Note the coupling term $D^r_{sep}\delta t$.

Using least-squares optimization to solve for the coefficients, the error of the model is bounded by 2% for all configurations, regardless of policy, airspace, intensity, or remaining parameters.

For all considered policies, except MCRT, $c_3 = 0$. As long as the conflict-detection and

resolution problem is solved in an approximate event-based implementation, there is little effect of uncertainty on the conflict-resolution taskload. For the MCRT policy, the coefficient c_3 is non-zero but bounded between $0 < c_3 < 12$. Even with an event-based implementation of the MCRT policy, uncertainty has a minor effect on the conflict-resolution taskload. The additional conflict-resolution taskload is a consequence of the MCRT policy making decisions on uncertain potential conflicts that do not become realized. In the worst case scenario, for $\delta t = 0$ and $D^r_{sep} = 0.5$ NM/min, the maximum average increases in expected conflict-resolution taskload is 6 more resolution commands during the 10 hour simulation period.

In current operations, there already exist conflict-resolution algorithms capable of solving even complex conflict configurations in under a minute. [28] and [32] are two such examples taking different approaches. In [28], the authors make use of genetic algorithms to simultaneously generate resolution commands for sets of aircraft (# of aircraft ≤ 15). Aircraft trajectories are represented by a genes sequence representative of motion primitives (i.e. heading, speed or altitude changes). The conflict-resolution algorithm in [28] can also be implemented sequentially, resolving potential conflicts individually in order of occurrence [4]. Sequential conflict-resolution degrades the problem to path planning for a single aircraft in a dynamic environment. Relative to simultaneous conflict-resolution, sequential path planning can be solved faster for the complete set of aircraft. While not always capable of generating feasible solutions for both simultaneous and sequential implementations, there is no reason why the gene library of potential trajectory solutions cannot be expanded to overcome this limitation. As it stands, the gene library does not consider complicated closed-loop trajectories with multiple lateral off-set, speed, and altitude changes. Furthermore, with the advent of multi-core and graphics processing unit computing, genetic algorithms can take advantage of parallelized implementations to ensure that feasible solutions can be found, even when large libraries are used [43]. Likewise, the rule-based heuristic method proposed in [32] can take a similar approach.

For present-day traffic levels, there exist conflict-resolution formulations that can be solved within 1 minute or in an event-based implementation. However, as traffic intensities

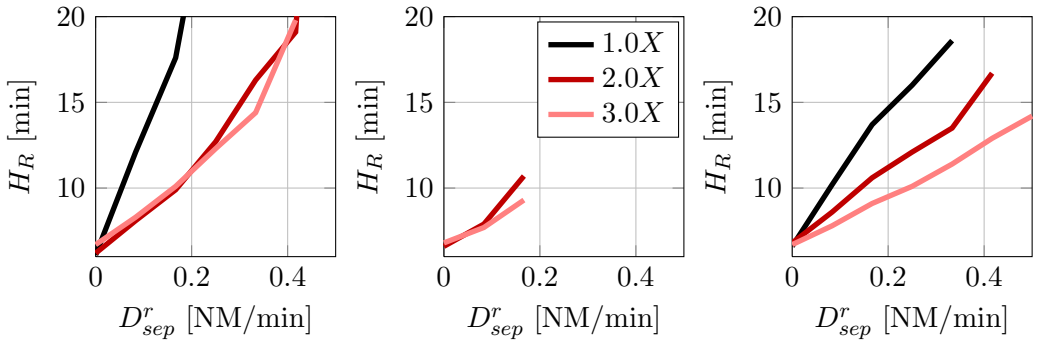
and conflict complexities increase, the issue of solve-times will become more pertinent. It is certainly true that faster solves-times are preferred, as longer solve-times have the potential to create ancillary problems. For example, in the case of a system-crash or the rejection of an advisory command, time is needed either to recover the system or to request a new advisory. A more problematic circumstance arises when the air traffic controller is unaware of a system failure and continues to wait for advisory resolution commands. For these cases, systems must be designed to reboot quickly, and generate any necessary resolution commands before problems occur. Therefore, it is assumed and asserted that conflict-resolution tools must be capable of resolving the conflict-resolution problem in under 1 minute to ensure they are useful to air traffic controllers.

The primary question then becomes: Is it preferred to have rapid solution-times (i.e. $\delta t = 0$ minutes) to help manage uncertainty, or is it preferred to sacrifice solve-times to ensure longer conflict-free times, H_R?

Similar to the benefit analysis provided in Section 7.4, solve-times and conflict-free times are now compared. In this case, the following question is asked:

> *Let improved computation (through hardware or software implementation) result in faster solve times from $\delta t = 1$ minute to $\delta t \sim 0$ minutes. Additionally, let the same conflict-resolution system be able to guarantee conflict-free trajectories for $H_R = 6$ minutes. The improvement to a faster solve-time limits the effect of uncertainty, resulting in a decrease in the expected conflict-resolution taskload from $E[N^{initial}]$ to $E[N^{final}]$. If instead solve-times are held constant, but efforts are taken to strengthen the conflict-resolution algorithms, what is the required conflict-free time, H_R, to achieve the same decrease in conflict-resolution taskload?*

Figure 83 answers the question for sectors ZMP12, ZMP16, and ZMP42 at different traffic intensities according to the uncertainty parameter D^r_{min} (FCFS decision policy).



(a) ZMP12 (b) ZMP16 (see text) (c) ZMP42

Figure 83: Required value of H_R to match faster solve-times.

Using Figure 83c as an example, at the 2X traffic level, if $D^r_{sep} = 0.25$ NM/min, a conflict-free time of at least $H_R \sim 12$ minutes is required to equal the reduction in the conflict-resolution taskload due to an improvement in the solve-time. In some cases, for specific values of traffic intensity or uncertainty, no value of H_R can match the change in conflict-resolution taskload. This is reflected by the lack of a solution (i.e. line) in Figure 83b; there is no matching H_R for the 1X traffic intensity, and at the 2X and 3X traffic levels, only a limited range of uncertainty values have solutions. Consistent across the three sectors, as traffic intensity increases, smaller improvements to H_R are required.

Problematic with this analysis is that there is no consistent answer to which is preferred: faster solve-times or improved conflict-free times. As Figure 83 demonstrates, the equivalent value of H_R is different for each sector. Furthermore, changes in the required H_R between traffic intensities are not consistent. For example, for ZMP42, there is a significant separation in the equivalent values of H_R between the 2X and 3X traffic intentsity lines. However, for ZMP12 when $D^R_{sep} \leq 0.33$ [NM/min], there is little distinction between the 2X and 3X traffic intensity lines.

Ultimately, there does not exist a standard best practice for the selection of H_R and δt. Rather, each parameter setting needs to be tuned according to the traffic within the sector and the uncertainty in trajectory prediction. However, considering that uncertainty in identifying potential conflicts is a function of the trajectory prediction tools, an alternative approach is to improve these systems. Just as reductions in D^r_{sep} provided large decreases in

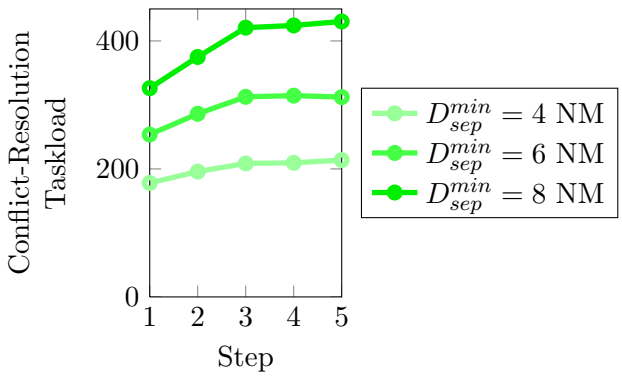


Figure 84: Representative summary of analysis.

the conflict-resolution taskload when compared to H_R, again, the conflict-detection side of the advisory tools can be improved to make up for short comings in the conflict-resolution algorithms.

7.7 Review

The results in this chapter can best be understand through Figure 84, which follows a sequence of steps corresponding to reductions in the performance capabilities of an advisory conflict-detection and resolution system. **Step 1** represents the best-case implementation of an advisory system for a given value of D_{sep}^{min}: the decision policy is MCRT, $D_{sep}^{r} = 0$ NM/min, $\delta t = 0$ minutes, $H_D = \infty$ minutes, and $H_R = \infty$ minutes. In this case, on average the least number of advisory resolution commands is generated. **Step 2** reflects a change in the decision policy from MRCT to FCFS. With the change, there is an increase in the expected conflict-resolution taskload. In **Step 3**, the conflict-free time is reduced to 6 minutes, which brings with it another increase. In **Step 4** and **Step 5**, the solve-time is slowed to $\delta t = 1$ minute, and the uncertainty parameter is changed to $D_{sep}^{r} = 0.167$ NM/min. The last configuration is representative of an advisory system that could realistically be implemented.

The most noticeable characteristic of the figure is that the conflict-resolution taskload is most sensitive to the aircraft spacing parameter D_{sep}^{min}. When $D_{sep}^{min} = 4$ NM, even the degraded performance setting of the advisory system results in a lower conflict-resolution taskload than the best-case implementation for an aircraft spacing of $D_{sep}^{min} = 6$ NM.

Therefore, to reduce conflict-resolution taskload, the most effective measure is to decrease the aircraft spacing requirement. As stated before, the introduction of ABS-B will most likely go a long way in enabling this reduction. Furthermore, improved trajectory prediction allows for greater certainty in labeling potential conflicts. In fact, results from Section 7.4 and Section 7.6 imply that short comings in conflict-resolution algorithms can be overcome through better measurement and prediction of current and future aircraft positions. What makes such a result particularly interesting is that the computation times required to improve trajectory predictions and position measurements are relatively small when compared to improvements in conflict-resolution algorithms. However, from a infrastructure perspective, the cost of improved conflict-resolution tools depends on software improvements coupled with more processing power. In contrast, to significantly improve the position measurement of aircraft through ADS-B, related infrastructure costs are estimated to be upwards of $7 billion [1], which excludes the development of software systems to take advantage of the improvements.

For the remainder of the parameters and settings, the sensitivity analysis indicates there is some leeway in their selection and implementation, assuming a reduction in aircraft spacing. One of the more prominent results in this chapter indicates that the underlying decision policy by which aircraft are selected to maneuver plays little role in determining the conflict-resolution taskload over long periods of time. And as a corollary, the decision-horizon time, H_D, has little perceivable influence on the conflict-resolution taskload. Even at high traffic intensities, the short-term benefits of the MCRT policy are not significant until the 3X traffic intensity for the busiest sector in Minneapolis center.

Because selection of the decision policy is somewhat flexible, consideration of other properties might be more relevant than just the conflict-resolution taskload. In an advisory based setting, the air traffic controller is required to approve the safety of each resolution command. It then might be beneficial for advisory systems to provide clear resolution commands, that is, commands that can be readily identified as safe by the air traffic controller. Doing so may allow the air traffic controller to continue processing potential conflicts in a more efficient and safe manner, as less time is spent reviewing each advisory command.

Also of interest, is the result that conflict-resolution algorithms need not provide conflict-free travel for long periods of time. In fact, beyond guaranteed conflict-free times of $H_R = 6$ minutes, there is limited benefit to improving conflict-resolution algorithms. An extension to this result is that for some traffic intensities, it is better to improve solve-times to help manage uncertainty than it is to increase conflict-free times. However, the division in trade-off between conflict-free times and solve-times is case dependent on the traffic intensity and traffic pattern within a sector.

Given that only a limited set of requirements is needed from a conflict-resolution algorithm (i.e. a satisfactory conflict-free time H_R, and a rapid solve-time δt), it is perhaps within the best interest of air navigation service providers to push the development and completion of a single advisory algorithm. Ultimately, the inclusion of an automated tool will require significant testing and verification; in order to prevent delays, selection of a single algorithm should be considered an imperative and occur as soon as possible.

CHAPTER VIII

CONCLUSION

8.1 Summary

Historically, engineers, researchers, and designers of conflict-resolution algorithms for air traffic systems have largely and implicitly ignored human factors issues by designing algorithms to replace, rather than support air traffic controllers. Despite the advances in automated conflict-resolution systems, there still exist a number of issues that are unaddressed or unsolved. In particular, current formulations are unable to guarantee safe and dynamically feasible trajectory solutions.

The research presented in this thesis focuses on an alternative to automating tactical air traffic control: the inclusion of advisory conflict-detection and resolution decision-support tools to aid air traffic controllers, without replacing them. In a human-in-the-loop framework, advisory conflict-detection and resolution systems identify potential conflicts and then propose resolution commands for the air traffic controller to verify and issue to aircraft. Much like automated or highly computerized solutions, advisory conflict-detection and resolution tools can be designed to handle large traffic loads and provide solutions in real-time. Since the air traffic controller is still part of the decision process and has the option to accept or reject proposed solutions, there remains a safety fall-back.

While human factors and cognitive engineering researchers have highlighted key aspects and requirements for the successful design of decision-support tools, there has been little actualization of these concepts into mathematically rigorous algorithms. The major contribution of this study is to bring together more formally these fields through the introduction and analysis of a model to better understand the conflict-resolution taskload associated with an advisory decision-support tool. The research presented here seeks to understand how the formulation, capabilities, and implementation of conflict-detection and resolution tools affect the controller taskload associated with the conflict-resolution process, and implicitly

controller workload.

Based on an analysis of a data-driven traffic model, and a sensitivity study of advisory conflict-detection and resolution tools, the following insights are drawn:

• When traffic throughput increases, not only will the number of potential conflicts multiply, but so will the complexity of conflict scenarios. Simple pairwise conflicts between two aircraft will become less likely. Instead, conflict scenarios will contain multiple aircraft that conflict with each other. (refer to Section 3.2)

• Without change to policies, practices, and technologies, air transportation demand will one day exceed the capacity of the unaided air traffic controller. As such, there is a need to provide some level of automation in air traffic control. (refer to Section 3.3)

• Currently, no guaranteed safe conflict-resolution algorithms exist. Therefore, an air traffic system that relies on complete automation or supervisory control is not a tenable option. Instead, advisory control might provide an acceptable alternative framework. Using advisory systems, air traffic controllers are able to leverage the strength of automated systems, while providing a safety fall back in the case of failure.

• The conflict-resolution taskload of an advisory conflict-detection and resolution system is most sensitive to aircraft spacing requirements. To achieve the greatest reductions in conflict-resolution taskload, air navigation service providers must make strong efforts towards improving aircraft position measurements and trajectory prediction tools. One avenue to achieving these goals is to make ADS-B utilization ubiquitous. (refer to Section 7.2 and Section 7.4)

• For policies considered, studies demonstrate that the decision policy of a conflict-resolution algorithm for selecting which aircraft to maneuver has an imperceivable effect on the conflict-resolution taskload. Because of this insensitivity, there is leeway in the design of conflict-resolution algorithms. So instead of spending effort to select a decision policy that minimizes conflict-resolution taskload, research efforts can best be placed into other areas of interest (e.g. minimizing fuel-costs). (refer to Section 7.3)

• Once aircraft spacing distances are reduced, the introduction of any conflict-resolution algorithm that is capable of generating conflict-free trajectories for 6 minutes or longer

is satisfactory in managing conflict-resolution taskload (refer to Section 7.5). The lack of strict requirements for an advisory conflict-detection and resolution system suggests that any well specified and functioning algorithm can be used. To prevent delay in formulating a complete tool, air navigation service providers and engineers should continue diligently by proposing and testing other requirements needed for a conflict-resolution tool (e.g. fuel-optimal, human-centric).

• No standard best practice exists for selecting the conflict-free resolution time and solve-time. Instead, the selection of each depends on the prevailing traffic pattern within an airspace and the traffic intensity. (refer to Section 7.6)

8.2 Contributions

According to the models and studies provided, the following contributions are contained within this thesis:

• Based on generated traffic scenarios, a quadratic model that considers traffic intensity and separation requirements is proposed to predict the number of uncontrolled conflicts within an airspace. Similar to previous studies, the predicted number of uncontrolled conflicts is proportional to the square of the traffic intensity. The major contribution of the quadratic model is to indicate that the number of uncontrolled conflicts is proportional to aircraft separation requirements. (refer to Section 3.2)

• Leveraging previous research representing aircraft and potential conflicts through static conflict graphs, this thesis extends the graph model to the dynamic case. Furthermore this thesis includes modeling of the conflict-detection and resolution process acting on dynamic conflict graphs. (refer to Chapter 4)

• The term conflict-resolution taskload is coined. In regards to previous studies, the meaning of conflict-resolution taskload is distinguished from the number of potential conflicts a conflict-resolution tool solves. In this way, the conflict-resolution taskload is related to the effort an air traffic controller makes in separating aircraft. (refer to Chapter 4)

• To support conflict-resolution taskload analysis of advisory decision-support tools, an abstraction of the behaviors and characteristics of conflict-detection and resolution tools

is proposed. More specifically, the abstraction of conflict-detection tools considers how potential conflicts are detected and resolved. Conflict-detection tools are parameterized by aircraft spacing requirements and uncertainty in trajectory prediction. The abstraction and parameterization of conflict-resolution tools considers the decision policy used to select which aircraft are maneuvered; how much information is used in the decision making process; the length of time for which resolution trajectories are guaranteed conflict-free; and how often the conflict-resolution problem is solved. In addition to the abstraction of advisory tools, a major contribution of this thesis is to provide a simulation framework in which to implement them without specifying their exact behavior. (refer to Chapter 5)

• Application of the abstracted conflict-detection and resolution tools demonstrate that for a controlled airspace, the conflict-resolution taskload is most sensitive to the traffic intensity and the effective aircraft spacing. According to specified abstracted parameters for a conflict-detection and resolution tool, the proposed model indicates that the conflict-resolution taskload is proportional to $I^2 D_{sep}^{min}$, where I and D_{sep}^{min} refer to the traffic intensity and aircraft spacing. (refer to Section 7.1)

• Demonstration that improvements to conflict-detection systems would provide the greatest benefit in reducing conflict-resolution taskload. (refer to Section 7.2 and Section 7.4)

• Based on a sensitivity analysis, a series of best and satisfactory practices is noted. One specific result concludes that the decision policy for selecting which aircraft to maneuver has little perceivable influence on the conflict-resolution taskload of an advisory conflict-resolution system (refer to Section 7.3). Additionally, the thesis demonstrates that increasing the conflict-free time of a conflict-resolution algorithm has diminishing returns for reducing the conflict-resolution taskload (refer to Section 7.5).

8.3 Future Work

It is necessary to note that the research presented in this thesis does not consider all aspects vital to determining conflict-resolution taskload. Furthermore, the results contained in this thesis suggest additional avenues of research that will provide greater insight into the interaction between air traffic controllers and advisory decision-support tools.

Improving Models. There is the potential for future work to incorporate a more flexible and realistic representation of the conflict-detection and resolution process. First, the conflict-detection model can be expanded to consider non-isotropic growth in uncertainty as well as variable growth rates. Such a model would be more reflective of errors in radar systems and trajectory prediction tools. Furthermore, to be more consistent with engineered conflict-detection and resolution algorithms, potential conflicts and resolution commands should be considered to be probabilistic. That is, potential conflicts are identified by their probability of occurrence, and conflict-free times for resolution trajectories are stochastic.

System-wide Conflict Resolution. Future research should address the system-wide effects of advisory conflict-resolution systems. Specifically, how managing a single airspace might result in downstream conflicts in other sectors. Or conversely, how conflict-resolution tools can be designed to resolve potential conflicts not only within one sector, but future sectors. The problem of considering adjacent sectors in the conflict-resolution process is a complex task, however most likely needed. As traffic intensities increase, aircraft interactions will become increasingly coupled through potential conflicts, not only at the sector-level but at the center-level.

In this thesis, analysis of the conflict-free time, H_R, relied on two assumptions. The traffic generation procedure assumed that the aircraft trajectories and potential conflicts generated for each airspace are independent of prior control actions taken in other sectors. Furthermore, when generating resolution commands, the conflict-resolution model assumes that solution trajectories are only conflict-free for H_R minutes within the current sector. Thus, the research as implemented in this thesis does not consider the multi-sector problem.

If a multi-sector approach towards conflict-resolution is desired then a serious discussion of workload issues is needed. Any resolution command that adjusts trajectories in future airspaces requires acceptance by the corresponding air traffic controller. In such a framework, controllers do not act as independent units. They are required to propose and approve trajectories with other controllers. While a system-wide approach towards conflict-resolution might further reduce conflict-resolution taskload, there is uncertainty as

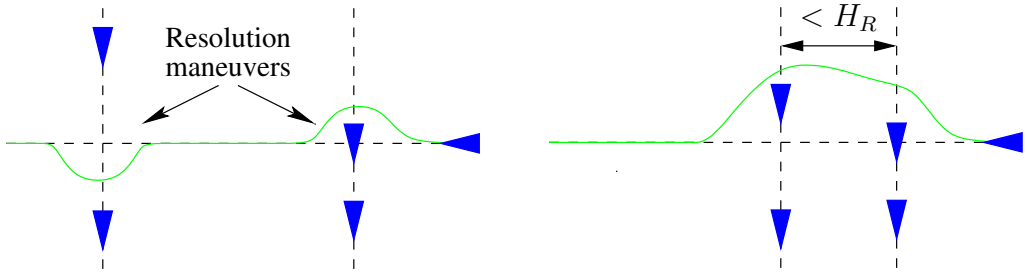
to how it will affect the workload of controllers. To enable such a system-wide approach requires an information distribution system similar to Nextgen's System Wide Information management (SWIM) concept [50].

Guarantee conflict-free travel. Developing algorithms that function at high traffic levels is still a concern. Even at present traffic levels there do not exist systems that can guarantee conflict-free travel. Future research is required to strength conflict-resolution algorithms.

Human-Centric Advisory Commands. The analysis in this thesis strictly focused on the conflict-resolution taskload of decision-support tools. This approach was taken with the understanding that it is a significant driver of controller workload when working with trajectory based operations. Further, this thesis makes the assumption that all advisory resolution commands affect controller workload equally. In fact, this assumption is probably not true, especially during high traffic volumes or when advisory resolution trajectories are difficult to decipher. Because the responsibility of ensuring safety lies with the air traffic controller, the more complex an advisory solution is, the more effort a controller must make in ensuring its safety and feasibility. So while a set of potential conflicts might be resolvable with a small number of maneuvers, the complexity of advisory trajectories might require significant mental effort from an air traffic controller. In these cases it might be preferred to issue a sub-optimal number of resolution commands that the air traffic controller can easily verify to be safe and feasible. Future studies are required to better understand how controllers respond to complex trajectories.

There is a need to considered the ramifications of mixed aircraft equipage in regards to ADS-B. Two of the underlying assumptions found in this thesis are that aircraft are spaced uniformly, and that aircraft are equally capable of receiving resolution trajectories consistent with trajectory based operations. For scenarios when ADS-B is not universal, complications may arise in the conflict-resolution process. As such, additional studies on the design of conflict-resolution algorithms in mixed-equipage environments is required.

Furthermore, the conflict-detection and resolution process proposed in this thesis does



(a) Current routes (b) Future routes

Figure 85: Taking full advantage of conflict-resolution systems may require a system-wide airspace redesign.

not need to be conducted in a receding-horizon control framework constrained to fixed time-steps and horizon times. Based on traffic conditions, the horizon times and the conflict-free times can be adjusted to meet the needs of the controller. In control theory there is already a stream of research that considers adaptive receding-horizon control [27, 65].

Airspace Redesign for Automated Conflict-Resolution. Finally, there is a need to take a fundamentally different approach to understanding the implementation of advisory tools. This thesis focused on analyzing the control system (both sensing and control) in an effort to manage conflict-resolution taskload. An alternative approach is to redesign the plant, e.g. the airspace, to be more consistent with advisory conflict-detection and resolution systems. In current-day operations, major traffic flows are spatially separated to space the time between potential conflicts for a single aircraft. Large distances between intersections provide the air traffic controller with sufficient time between potential conflicts to generate any necessary resolution commands. With trajectory based operations, if conflict-resolution algorithms are able to guarantee conflict-free flight for H_R minutes, then to take advantage of advisory conflict-detection and resolution systems, sequential conflict regions should be within H_R minutes of each other. As illustrated in Figure 85, to reduce the number of resolution commands, closely spaced flows might be beneficial. Otherwise, if intersection are spaced far apart, aircraft might be required to re-plan trajectories each time a potential conflict is identified.

APPENDIX A

DECISION POLICIES

In addition to the decision policies detailed in Chapter 5, five other decision policies are considered in the simulation analysis. They are detailed below.

A.1 First-Come, First-Served without Conflict-Priority Ordering

The First-Come, First-Served (FCFS) policy is adjusted to remove conflict-priority ordering. The FCFS policy without conflict-priority ordering results in a slightly different process for resolving potential conflicts when compared to the original policy described in Procedure 2. Instead of processing and resolving potential conflicts according to the order they occur, aircraft are first sorted by their arrival time into the airspace. The most recent unresolved aircraft involved in potential conflict is selected to receive a resolution command. The temporary edge-set $\mathcal{E}^T$ is updated accordingly. The processes repeats until the edge set $\mathcal{E}^T$ is empty. The final set of maneuvered aircraft is given by the set $\mathcal{M}$. The complete process is provided in Procedure 6.

Procedure 6 First-Come, First-Served without Conflict-Priority Ordering

$\mathcal{E}^T \leftarrow \hat{\mathcal{E}}(k),\ q = 0$

while $card(\mathcal{E}^T)$ **do**

$q = q + 1$

$i^* = \underset{i}{\text{argmax}}(t_i^a \mid (n_i, *) \in \mathcal{E}^T)$

$m_q^T = n_{i^*}$

$\mathcal{E}^T \leftarrow \mathcal{E}^T \setminus (m_q^T, *)$

end while

$I = [t + T_R, t + T_R + \delta t]$

$\mathcal{M} = \{m \mid m \in m^T \cap \exists t_{m,*}^c \in I\}$

A.2 First-Exit, First-Served

The first-exit, first-served policy (FEFS) is similar to the FCFS policy; however, instead of prioritizing aircraft based on their arrival time into the sector, aircraft are given priority according to when they are scheduled to exit the sector.

Using the same notation found in the procedure for the FCFS policy (see Procedure 2 and Procedure 6), the procedures for FEFS, with and without conflict-priority ordering, are given by Procedure 7 and Procedure 8. In the case of the FEFS policy, aircraft are ordered according to the original scheduled exit time from the sector, t_i^x. In the case when an aircraft receives a resolution maneuver, potentially adjusting the expected exit time, the scheduled exit time remains the same.

Procedure 7 First-Exit, First-Served with Conflict-Priority Ordering

$\mathcal{E}^T \leftarrow \hat{\mathcal{E}}(k),\ q = 0$

while $card(\mathcal{E}^T)$ **do**

$q = q + 1$

$(i^*, j^*) = \underset{i,j}{\text{argmin}}\ t^c_{i,j}$

$$m^T_q = \begin{cases} n_{i^*} & \text{if } t^x_{j^*} < t^x_{i^*} \\ n_{j^*} & \text{else} \end{cases}$$

$\mathcal{E}^T \leftarrow \mathcal{E}^T \setminus (m^T_q, *)$

end while

$I = [t + T_R, t + T_R + \delta t]$

$\mathcal{M} = \{m \mid m \in m^T \cap \exists t^c_{m,*} \in I\}$

Procedure 8 First-Exit, First-Served without Conflict-Priority Ordering

$\mathcal{E}^T \leftarrow \hat{\mathcal{E}}(k),\ q = 0$

while $card(\mathcal{E}^T)$ **do**

$q = q + 1$

$i^* = \underset{i}{\text{argmin}}(t^x_i \mid (n_i, *) \in E^T)$

$m^T_q = n_{i^*}$

$\mathcal{E}^T \leftarrow \mathcal{E}^T \setminus (m^T_q, *)$

end while

$I = [t + T_R, t + T_R + \delta t]$

$\mathcal{M} = \{m \mid m \in m^T \cap \exists t^c_{m,*} \in I\}$

Like the FCFS policy, there is always the potential of favoring aircraft traveling along specific routes. However, a benefit of the procedure is that because aircraft closer to sector boundaries are prioritized, the advisory system is less likely to issue resolution commands to aircraft about to transitioning between sectors. From the human-factors perspective, such a property may be preferred, as maneuvering aircraft near boundaries requires additional coordination effort between the controllers in each sector.

A.3 Knowledge-Based Policy

This policy is based on a knowledge-based policy called Resolution Aircraft and Maneuver Selector (RMAS). RAMS forms the backbone of the Automated Airspace Concept [32], a tool designed for performing automated conflict-resolution. Based on the characteristics of a pairwise conflict, RAMS selects which of the two aircraft to maneuver, as well as the preferred resolution command, according to a rule-set. RAMS is based on standard practices derived from human controllers and operational insights and analytical studies. In such a framework, potential conflicts are resolved according to conflict-priority ordering.

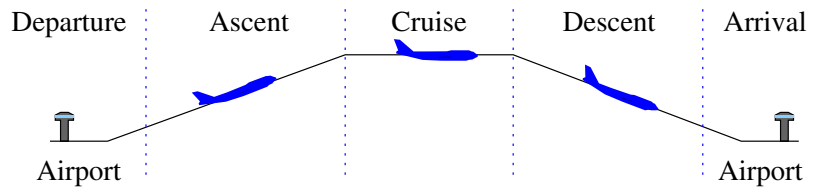


Figure 86: Different phases of flight considered in the knowledge-based policy.

The RAMS' rules dictating the preferred aircraft to maneuver for each type of potential conflict are similar to those provided in Table 9. In the original formulation of RAMS, an ordering of preferred solutions is included as part of the policy. Each potential conflict is defined according to the phase of flight each aircraft is in at the time of occurrence, not the time of detection. The major phases of flight include: ascent, cruise, and descent. The flight phases are illustrated in Figure 86. Table 9 provides a simplification of the rules given in [32]. As part of the RAMS system, additional fidelity differentiated two types of arrivals: cruise arrivals and descending arrivals. For the RAMS system, an arrival aircraft is defined as any aircraft within 200 NM or 20 minutes of its TRACON arrival fix. Because information concerning the distance and time to TRACON arrival fixes is not available in the PDARS data used to generate the traffic simulations, some of the rules are collapsed together to still allow for an approximation of RAMS. In some cases, the simplification loses resolution procedures that prevent reconstruction or inversion of the original rules.

Table 9: Knowledge-Based Policy Rules

Flight Phase of Aircraft		
Aircraft 1	**Aircraft 2**	**Preferred Aircraft to maneuver**
Cruise	Cruise Large crossing angle	Aircraft furthest from airspace boundary or top of descent
Cruise	Cruise Small crossing angle or aircraft in trail	Faster aircraft
Cruise	Ascent	Ascending aircraft
Cruise	Descent	Cruising aircraft
Ascent	Ascent	Lower aircraft
Ascent	Descent	Ascending aircraft
Descent	Descent	Trailing aircraft

The procedure for the knowledge-based policy with conflict-priority ordering is provided in Procedure 9. The function $KBR(A_i, A_j)$, included as part of the policy procedure, enacts

the knowledge-based rules provided in Table 9. The input declaration of the aircraft pair (A_i, A_j) into the function $KBR(A_i, A_j)$ assumes that classification information is available.

Procedure 9 Knowledge-Based Policy with Conflict-Priority Ordering

$\mathcal{E}^T \leftarrow \hat{\mathcal{E}}(k),\ q = 0$

while $card(\mathcal{E}^T)$ **do**

$q = q + 1$

$(i^*, j^*) = \underset{i,j}{\operatorname{argmin}}\ t^c_{i,j}$

$m^T_q = KBR(A_{i^*}, A_{j^*})$

$\mathcal{E}^T \leftarrow \mathcal{E}^T \setminus (m^T_q, *)$

end while

$I = [t + T_R, t + T_R + \delta t]$

$\mathcal{M} = \{m \mid m \in m^T \cap \exists t^c_{m,*} \in I\}$

A.4 Random Policy with Maneuver Balancing

The second random procedure, described by Procedure 10, takes a more balanced approach towards deciding which aircraft are issued resolution commands. In particular, the policy considers which aircraft have been previously maneuvered through the vector M^P. The value of M^P_i indicates the number of times aircraft A_i has been issued a resolution command. For the potential conflict between aircraft A_i and A_j, the values M^P_i and M^P_j are compared. If $M^P_i = M^P_j$, that is, both aircraft have been maneuvered the same number of times, then the policy randomly selects which aircraft to maneuver. In the case when $M^P_i \neq M^P_j$, the aircraft with less maneuvers is issued a resolution command. After the corresponding aircraft are issued resolution commands, the vector M^P is updated.

By taking into account the number of times an aircraft has been maneuvered prior to any potential conflicts, the second policy acts to balance the number of times any one aircraft receives trajectory perturbations. In the context of fuel consumption, if each maneuver

entails an extra deviation or fuel-cost, then by balancing the distribution of maneuvered aircraft, the fuel consumption is better balanced across aircraft.

Procedure 10 Random with Prior-Resolution Balancing and Conflict-Priority Ordering

$\mathcal{E}^T \leftarrow hat\mathcal{E}(k),\ q = 0$

while $card(\mathcal{E}^T)$ **do**

$q = q + 1$

$(i^*, j^*) = \underset{i,j}{\text{argmin}}\ t^c_{i,j}$

$$m^T_q = \begin{cases} n_{i^*} & \text{if } (M^P_{i^*} < M^P_{j^*}) \cup \\ & (M^P_{i^*} = M^P_{j^*} \cap rand < .5) \\ n_{j^*} & \text{if else} \end{cases}$$

$\mathcal{E}^T \leftarrow \mathcal{E}^T \setminus (m^T_q, *)$

end while

$I = [t + T_R, t + T_R + \delta t]$

$\mathcal{M} = \{m \mid m \in m^T \cap \exists t^c_{m,*} \in I\}$

$M^P_i \leftarrow M^P_i + card(i \in \mathcal{M})$

REFERENCES

[1] "Final regulatory impact analysis, final rule, Automatic Dependent Surveillance Broadcast (ADS-B) Out, performance requirements to support air traffic control service," Tech. Rep. FAA200729305, Federal Aviation Administration, Office of Aviation Policy and Plans, August 18, 2009.

[2] Agogino, A. and Tumer, K., "Evolving distributed agents for managing air traffic," in Conference on Genetic and Evolutionary Computation.

[3] Alliot, J., Bosc, J., Durand, N., and Maugis, L., "CATS: A complete air traffic simulator," in Digital Avionics Systems Conference, 1997.

[4] Archambault, N. and Durand, N., "Scheduling heuristics for on-board sequential air conflict solving," in Digital Avionics Systems Conference, 2004.

[5] Arthur, W. and McLaughlin, M., "User request evaluation tool (URET) interfacility conflict probe performance assessment," in USA/Europe Air Traffic Management R&D Seminar, 1998.

[6] Averty, P., "Conflict Perception by ATCS Admits Doubt but not Inconsistency," in USA/Europe Air Traffic Management R&D Seminar, 2005.

[7] Averty, P., "Elements for prioritizing between conflict resolutions in air traffic control," in Digital Avionics Systems Conference, 2008.

[8] Barnier, N. and Allignol, C., "4D-trajectory deconfliction through departure time adjustment," in USA/Europe Air Traffic Management R&D Seminar, 2009.

[9] Billings, C. and Woods, D., "Concerns about adaptive automation in aviation systems," Human performance in automated systems: Current research and trends, pp. 264–269, 1994.

[10] Bilmoria, K., Banavar, S., Chatterji, G., Sheth, K., and Grabbe, S., "FACET: Future ATM concepts evaluation tool," Air Traffic Control Quarterly, vol. 9, no. 1, 2000.

[11] Brooker, P., "Control workload, airspace capacity and future systems," Human Factors and Aerospace Safety, vol. 3, no. 1, pp. 1–23, 2003.

[12] Calderón-Meza, G. and Sherry, L., "Analysis of stakeholder benefits of nextgen trajectory-based operations," in Integrated Communications Navigation and Surveillance Conference, 2010.

[13] Calderón-Meza, G. and Sherry, L., "Establishing an Upper-Bound for the Benefits of NextGen Trajectory-Based Operations," in International Conference on Research in Air Transportation, 2010.

[14] Cardosi, K., "An Analysis of En Route Controller-Pilot Voice Communications," Tech. Rep. DOT/FAA/RD-93/11, US Department of Transportation, Federal Aviation Administration, 1993.

[15] Chaloulos, G., Hokayem, P., and Lygeros, J., "Distributed hierarchical MPC for conflict resolution in air traffic control," in American Control Conference.

[16] Chaloulos, G., Roussos, G., Lygeros, J., and Kyriakopoulos, K., "Ground Assisted Conflict Resolution in Self-Separation Airspace," in Guidance, Navigation and Control Conference and Exhibit, 2008.

[17] Chatterji, G. and Sridhar, B., "Neural network based air traffic controller workload prediction," in American Control Conference, 1999.

[18] Chvatal, V., "A greedy heuristic for the set-covering problem," Mathematics of Operations Research, pp. 233–235, 1979.

[19] Clarkson, K., "Modification of the greedy algorithm for vertex cover.," Information Processing Letters, vol. 16, no. 1, pp. 23–26, 1983.

[20] Committee for a Review of the En Route Air Traffic Control Complexity and Workload Model, "Air Traffic Controller Staffing in the En Route Domain : A Review of the Federal Aviation Administrations Task Load Model," tech. rep., National Research Council of the National Academies, 2010.

[21] Corker, K., Gore, B., Fleming, K., and Lane, J., "Free Flight and the Context of Control: Experiments and Modeling to Determine the Impact of Distributed Air-ground Air Traffic Management on Safety and Procedures," in USA/Europe Air Traffic Management R&D Seminar, 2000.

[22] Cruck, E. and Lygeros, J., "Subliminal Air Traffic Control: Human Friendly Control of a Multi-Agent System," in American Control Conference, 2007.

[23] Davis, T., Krzeczowski, K., and Bergh, C., "The final approach spacing tool," in Symposium on Automatic Control in Aerospace, 1994.

[24] Delahaye, D. and Puechmorel, S., "Air traffic complexity map based on non linear dynamical systems," in EUROCONTROL Innovative Research Workshop, 2005.

[25] Dougui, N., Delahaye, D., Puechmorel, S., and Mongeau, M., "A New Method for Generating Optimal Conflict Free 4D Trajectory," in International Conference on Research in Air Transportation, 2010.

[26] Dowek, G., Muñoz, C., and Carreño, V., "Provably Safe Coordinated Strategy for Distributed Conflict Resolution," in Guidance Navigation, and Control Conference and Exhibit, 2005.

[27] Droge, G. and Egerstedt, M., "Adaptive look-ahead for robotic navigation in unknown environments," in International Conference on Intelligent Robots and Systems, 2011.

[28] Durand, N. and Alliot, J.-M., "Optimal resolution of en route conflicts," in USA/Europe Air Traffic Management R&D Seminar, 2001.

[29] Ehrmanntraut, R., "The Potential of Speed Control," in Digital Avionics Systems Conference, 2004.

[30] Endsley, M., "Automation and situation awareness," Automation and human performance: Theory and applications, pp. 163–181, 1996.

[31] Ennis, R. and Zhao, Y., "A formal approach to the analysis of aircraft protected zone.," Air Traffic Control Quarterly, vol. 12, no. 1, pp. 75–102, 2004.

[32] Erzberger, H., "Automated Conflict Resolution for Air Traffic Control," in International Congress of the Aeronautical Sciences, 2006.

[33] Erzberger, H. and Paielli, R., "Concept for next generation air traffic control system.," Air Traffic Control Quarterly, vol. 10, no. 4, pp. 355–378, 2002.

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

[35] Federal Aviation Administration, "FAA Aerospace Forecast: Fiscal Years 2010-2030," tech. rep., U.S. Department of Transportation, 2010.

[36] Federal Aviation Administration, "Facility Orientation Guide: Minneapolis ARTCC." www.stuckmic.com/faa-facility-orientation-guides.html, September 2011.

[37] Galdino, A., Muñoz, C., and Ayala, M., "Formal Verification of an Optimal Air Traffic Conflict Resolution and Recovery Algorithm," in Workshop on Logic, Language, Information and Computation, 2007.

[38] Gangel, M., Mehta, G., Muhammad, C., Sekhavat, S., Vora, G., and Arkind, K., "Terminal area capacity enhancement concept," in Systems and Information Engineering Design Symposium, 2004.

[39] Garcia, J., Besada Portas, J., Soto Jaramillo, A., and Miguel Vela, G., "Opportunity trajectory reconstruction techniques for evaluation of ATC systems," International Journal of Microwave and Wireless Technologies, vol. 1, no. 3, pp. 231–238, 2009.

[40] Gawinowski, G., Garcia, J.-L., Guerreau, R., Weber, R., and Brochard, M., "ERASMUS: A new Path for 4D Trajectory-Based Enablers to Reduce the Traffic Complexity," in Digital Avionics Systems Conference, 2007.

[41] Granger, G. and Durand, N., "A traffic complexity approach through cluster analysis," in USA/Europe Air Traffic Management R&D Seminar, 2003.

[42] Green, S., Vivona, R., Grace, M., and Fang, T., "Field evaluation of descent advisor trajectory prediction accuracy for en route clearance advisories," in Guidance, Navigation, and Control Conference and Exhibit, 1998.

[43] Harding, S. and Banzhaf, W., "Fast genetic programming on GPUs," Genetic Programming, pp. 90–101, 2007.

[44] Hinton, D., Charnock, J., and Bagwell, D., "Design of an aircraft vortex spacing system for airport capacity improvement," in Aerospace Sciences Meeting & Exhibit, 2000.

[45] Hoekstra, J., van Gent, R., and Ruigrok, R., "Designing for safety: the free flight air traffic management concept," Reliability Engineering & System Safety, vol. 75, no. 2, pp. 215–232, 2002.

[46] Hull, J., Barmore, B., and Abbott, T., "Technology-enabled airborne spacing and merging," in Digital Avionics Systems Conference, 2004.

[47] Irvine, R., "A simplified approach to conflict probability estimation," in Digital Avionics Systems Conference, 2001.

[48] Jardin, M., "Air traffic conflict models," in Aviation Technology, Integration and Operations Conference, 2004.

[49] Joint Planning and Development Office, "Next Generation Air Transportation System Integrated Work Plan: A Functional Outline," tech. rep., Joint Planning and Development Office, Washington, DC, Sept. 2008.

[50] Joint Planning and Development Office, "Concept of Operations for the Next Generation Air Transportation System," tech. rep., Joint Planning and Development Office, Washington, DC, Sept. 2010.

[51] Kirwan, B., "The role of the controller in the accelerating industry of air traffic management," Safety Science, vol. 37, no. 2-3, pp. 151–185, 2001.

[52] Kirwan, B. and Flynn, M., "Towards a controller-based conflict resolution tool - a literature review," Tech. Rep. ASA.01.CORA.2.DEL04-A.LIT, EUROCONTROL, 2002.

[53] Kopardekar, P., Prevot, T., and Jastrzebski, M., "Traffic complexity measurement under higher levels of automation and higher traffic densities," Air Traffic Control Quarterly, vol. 17, no. 2, pp. 125–148, 2009.

[54] Kopardekar, P., Schwartz, A., Magyarits, S., and Rhodes, J., "Airspace complexity measurement: An air traffic control simulation analysis," in USA/Europe Air Traffic Management R&D Seminar, 2007.

[55] Kopardekar, P. and Magyarits, S., "Dynamic density: Measuring and predicting sector complexity," in Digital Avionics Systems Conference, 2002.

[56] Kuchar, J. and Yang, L., "A Review of Conflict Detection and Resolution Modeling Methods," IEEE Transactions on Intelligent Transportation Systems, vol. 1, pp. 179–189, Dec. 2000.

[57] Lange, M., Hjalmarsson, J., Cooper, M., Ynnerman, A., and Duong, V., "3d visualization and 3d and voice interaction in air traffic management," in The annual SIGRAD Conference, 2003.

[58] Laudeman, I., Shelden, S., Branstrom, R., and Brasil, C., "Dynamic density: An air traffic management metric," Tech. Rep. NASA/TM-1998-112226, 1998.

[59] Loft, S., Sanderson, P., Neal, A., and Mooij, M., "Modeling and predicting mental workload in en route air traffic control: Critical review and broader implications," Human Factors: The Journal of the Human Factors and Ergonomics Society, vol. 49, no. 3, p. 376, 2007.

[60] Loghides, C., Paulson, G., and Philipp, W., "Guidelines for the application of the ecac radar separation minima," Tech. Rep. ASM.ET1.ST18.1000 -REP-01.00, EUROCONTROL, 1998.

[61] Maddalon, J., Butler, R., Geser, A., and Muñoz, C., "Formal verification of a conflict resolution and recovery algorithm," Tech. Rep. NASA/TP-2004-213015, NASA Langley Research Center, NASA LaRC,Hampton VA 23681-2199, USA, April 2004.

[62] Manning, C., Mills, S., Pfleiderer, E., Fox, C., and Mogilka, H., "Relationships among observer ratings of mental workload and taskload measures derived from routinely recorded air traffic control data," Focusing Attention on Aviation Safety, 2001.

[63] Manning, C., Mills, S., Fox, C., Pfleiderer, E., and Mogilka, H., "Using Air Traffic Control Taskload Measure and Communication Events to Predict Subjective Workload," tech. rep., Federal Aviation Administration, Office of Aerospace Medicine, 2002.

[64] Metzger, U. and Parasuraman, R., "The role of the air traffic controller in future air traffic management: An empirical study of active control versus passive monitoring," Human Factors: The Journal of the Human Factors and Ergonomics Society, vol. 43, no. 4, pp. 519–528, 2001.

[65] Michalska, H. and Mayne, D., "Robust receding horizon control of constrained nonlinear systems," IEEE Transactions on Automatic Control, vol. 38, no. 11, pp. 1623–1633, 1993.

[66] Mogford, R., "The complexity construct in air traffic control: a review and synthesis of the literature," Tech. Rep. DOT/FAA/CT-TN92/22, Department of Transportation / Federal Aviation Administration Technical Center, 1995.

[67] Mondoloni, S., Paglione, M., and Green, S., "Trajectory modeling accuracy for air traffic management decision support tools," in Congress of International Council of the Aeronautical Sciences, 2002.

[68] Mueller, E. and Lozito, S., "Experimental evaluation of an integrated datalink and automation-based strategic trajectory concept," in Aviation Technology, Integration, and Operations Conference, 2007.

[69] Niedringhaus, W., "Maneuver Option Manager: Automated Simplification of Complex Air Traffic Control Problems," IEEE Transactions on Systems, Man and Cybernetics, vol. 22, no. 5, pp. 1047–1057, 1991.

[70] Niedringhaus, W., "Stream Option Manager (SOM): Automated Integration of Aircraft Separation, Merging, Stream Management, and other Air Traffic Control Functions," IEEE Transactions on Systems, Man and Cybernetics, vol. 25, no. 9, pp. 1269–1280, 1995.

[71] Nijhuis, H., "RHEA, Role of the Human in the Evolution of ATM systems: Final Report," tech. rep., National Aerospace Laboratory, 1998.

[72] Ore, O., Theory of Graphs. American Mathematical Society, 1962.

[73] Ozgur, M. and Cavcar, A., "A knowledge-based conflict resolution tool for rn-route air traffic controllers," Aircraft Engineering and Aerospace Technology, vol. 80, no. 6, pp. 649–656, 2008.

[74] Paielli, R. and Erzberger, H., "Conflict Probability Estimation for Free Flight," Journal of Guidance, Control, and Dynamics, vol. 20, no. 3, pp. 588–596, 1997.

[75] Parasuraman, R. and Riley, V., "Humans and automation: Use, misuse, disuse, abuse.," Human Factors, vol. 39, no. 2, 1997.

[76] Perry, T., "In search of the future of air traffic control," Spectrum, IEEE, vol. 34, no. 8, pp. 18–35, 1997.

[77] Porterfield, D., "Evaluating controller communication time as a measure of workload," International Journal of Aviation Psychology, vol. 7, no. 2, pp. 171–182, 1997.

[78] Prandini, M., Putta, V., and Hu, J., "A probabilistic measure of air traffic complexity in three-dimensional airspace," International Journal of Adaptive Control and Signal Processing, vol. 1, pp. 1–25, 2009.

[79] Prevot, T., Lee, P., Callantine, T., Smith, N., and Palmer, E., "Trajectory-oriented time-based arrival operations: Results and recommendations," in USA/Europe Air Traffic Management R&D Seminar, 2003.

[80] Rardin, R. L., Optimization in Operations Research. Prentice Hall, 1997.

[81] Reynolds, T. and Hansman, R., "Analysis of separation minima using a surveillance state vector approach," in USA/Europe Air Traffic Management R&D Seminar, 2000.

[82] Ropella, K., Introduction to Statistics for Biomedical Engineers. Morgan and Claypool Publishers, 2007.

[83] Roussos, G., Chaloulos, G., Kyriakopoulos, K., and Lygeros, J., "Control of multiple non-holonomic air vehicles under wind uncertainty using model predictive control and decentralized navigation functions," in Conference on Decision and Control.

[84] Salaün, E., Gariel, M., Vela, A., Feron, E., and Clarke, J., "Airspace Complexity Estimations Based on Data-Driven Flow Modeling," in Guidance, Navigation, and Control Conference, 2010.

[85] Schmidt, D., "On the conflict frequency at air route intersections," Transportation Research, vol. 11, no. 5, pp. 351–355, 1977.

[86] SESAR Consortium, "The ATM Target Concept: D3," Tech. Rep. DLM-0612-001-02-00, SESAR Consortium, Toulouse, France, Sept. 2007.

[87] Shingledecker, C. A., "Subsidiary radio communications tasks for workload assessment in R&D simulations," tech. rep., Air force aerospace medical research laboratory, 1982.

[88] Shyu, S., Yin, P., and Lin, B., "An ant colony optimization algorithm for the minimum weight vertex cover problem," Annals of Operations Research, vol. 131, no. 1, pp. 283–304, 2004.

[89] Slattery, R. and Green, S., "Conflict free trajectory planning for air traffic control automation," Tech. Rep. TM-108790, NASA Ames Research Center, 1994.

[90] Smith, A. and Bateman, H., "Management of holding patterns: A potential ads-b application," in Digital Avionics Systems Conference, 2008.

[91] Smith, P., McCoy, C., and Layton, C., "Brittleness in the design of cooperative problem-solving systems: The effects on user performance," IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans, vol. 27, no. 3, pp. 360–371, 2002.

[92] Sperandio, J., "Variation of Operator's Strategies and Regulating Effects on Workload," Ergonomics, vol. 14, no. 5, pp. 571–577, 1971.

[93] Sridhar, B., Sheth, K., and Grabbe, S., "Airspace complexity and its application in air traffic management," in USA/Europe Air Traffic Management R&D Seminar, 1998.

[94] Stein, E., "Human Operator Workload in Air Traffic Control," Human Factors in Air Traffic Control, pp. 155–184, 1998.

[95] Swenson, H., Barhydt, R., and Landis, M., "Next Generation Air Transportation System (NGATS) Air Traffic Management (ATM)-Airspace Project," tech. rep., National Aeronautics and Space Administration, 2006.

[96] Swierstra, S. and Green, S., "Common trajectory prediction capability for decision support tools," in USA/Europe Air Traffic Management R&D Seminar, 2003.

[97] Tomlin, C., Pappas, G., Lygeros, J., Godbole, D., and Sastry, S., "Hybrid control models of next generation air traffic management," Hybrid systems IV, pp. 378–404, 1997.

[98] Transforming the NAS: The next generation air traffic control system, "Erzberger, h.," in International Congress of the Aeronautical Sciences, 2005.

[99] Transportation Safety Board of Canada, "Aviation reports - 2008 - a08w0007." http://www.tsb.gc.ca/eng/rapports-reports/aviation/2008/a08w0007/a08w0007.asp, 9 2011.

[100] Vela, A., Solak, S., Clarke, J., Singhose, W., Johnson, E., and Barnes, E., "Near Real-Time Fuel-Optimal En Route Conflict Resolution," IEEE Transactions on Intelligent Transportation Systems, vol. 11, no. 4, pp. 826–837, 2010.

[101] Vela, A., Solak, S., Feron, E., Feigh, K., Singhose, W., and Clarke, J., "A fuel optimal and reduced controller workload optimization model for conflict resolution," in Digital Avionics Systems Conference, 2009.

[102] Wang, H., Kirubarajan, T., and Bar-Shalom, Y., "Precision large scale air traffic surveillance using imm/assignment estimators," IEEE Transactions on Aerospace and Electronic Systems, vol. 35, no. 1, pp. 255–266, 1999.

[103] Wanke, C., "Using Air-Ground Data Link to Improve Air Traffic Management Decision Support System Performance," in USA-Europe ATM R&D Seminar, 1997.

[104] West, D. B., Introduction to Graph Theory (2nd Edition). Prentice Hall, Aug. 2000.