Autonomous & Resilient Countermeasures for Emergent System Disruptions with Application to Air Traffic Management

by Curtis J. Marshall

B.S. in Electrical/Mechanical Engineering Technology, May 2008, Rochester Institute of Technology
M.S. in Manufacturing & Mechanical Systems Integration, November 2008, Rochester Institute of Technology
M.E. in Systems Engineering, May 2013, Stevens Institute of Technology

A Dissertation submitted to

The Faculty of
The School of Engineering and Applied Science
of The George Washington University
in partial fulfillment of the requirements
for the degree of Doctor of Philosophy

January 10, 2020

Dissertation directed by

Thomas Holzer
Professorial Lecturer of Engineering Management and Systems Engineering

The School of Engineering and Applied Science of The George Washington University certifies that Curtis J. Marshall has passed the Final Examination for the degree of Doctor of Philosophy as of November 15, 2019. This is the final and approved form of the dissertation.

Autonomous & Resilient Countermeasures for Emergent System Disruptions with Application to Air Traffic Management

Curtis J. Marshall

Dissertation Research Committee:

Thomas Holzer, Professorial Lecturer of Engineering Management and Systems Engineering, Dissertation Director

Thomas Mazzuchi, Professor of Engineering Management and Systems Engineering & of Decision Sciences, Committee Member

Shahram Sarkani, Professor of Engineering Management and Systems Engineering, Committee Member

Amir Etemadi, Professorial Lecturer of Engineering Management and Systems Engineering, Committee Member

Shahyar Sarkani, Professorial Lecturer of Engineering Management and Systems Engineering, Committee Member

© Copyright 2020 by Curtis J. Marshall
All rights reserved

Acknowledgements

I thank all of my academic advisors and instructors who imparted a wealth of knowledge and wisdom throughout the execution of this research, especially Dr. Thomas Holzer, Dr. Blake Roberts, and Dr. Michael Grenn.

I would like to express my appreciation to my professional colleagues for their unrelenting encouragement and support, notably Dr. Keith Stuart, Dr. Stephen Marley, Dr. LeAnn Miller, Dr. Johnathon Huff, Dr. David Wheeler, Dr. Ireena Erteza, Dr. Arnoldo Muyshondt, Mr. Carlos Tafoya, Mr. Salvador Martell, Mr. Aaron Hillhouse, Mr. Brian Geery, Mr. Ronald Kulju, Mr. Steven Rohde, Ms. Deanna Montoya, Mr. Ronald Espinoza, Mrs. Amber Cantwell, Mr. Timothy Wiseley, Mr. Ronald Baker, Mr. Michael Emerson, and Ms. Kathryn Naassan.

I am especially grateful for the patience and understanding of family and friends over many years while executing this research. I am exceptionally fortunate for their encouragement and support during the major setbacks and achievements alike.

Abstract of Dissertation

Autonomous & Resilient Countermeasures for Emergent System Disruptions with Application to Air Traffic Management

Engineered systems are susceptible to the disruption of intended functionality when unanticipated operating environments and constraints emerge during mission execution. To safeguard intended system functionality, resilient systems which integrate disruption avoidance and mitigation measures are needed. System automation methods have been successfully adopted for disruption avoidance and performance optimization based on anticipated needs and risks. However, given the development of unpredicted or emergent issues, human intervention is often relied on as an operational contingency to system failure or degraded performance. The purpose of this research was to develop and analyze a model for evaluating and selecting system responses to mitigate emergent disruptions without human intervention. The scope of this research addresses risk management within the system lifecycle with focus on system adaptation to unpredictable changes in the operational environment or functional constraints. The *Disruption Resilience and Adaptation Model (DREAM)* was formulated as an autonomous decision-making and performance evaluation process for mitigating unforeseen or unavoidable system disruptions. Evaluating the system as a whole, the model integrates diagnostic and prognostic heuristics for performance feedback and regulating system responses to actual or potential disruptions. The suitability of rule and utility-based automation methods as system adaptation techniques was also investigated.

Via modeling and simulation, the DREAM and existing system automation methods were applied to an Air Traffic Management (ATM) problem and compared based on: (i)

mission reliability; (ii) incident rate; (iii) system efficiency; (iv) standard deviation of efficiency; (v) and system stability. Simulation results demonstrate the DREAM yielded statistically significant reductions in the frequency and severity of ATM system disruptions in comparison to the existing rule-based standard and the leading utility-based method.

As demonstrated in the ATM case study performed in this research, the DREAM can reduce the frequency and severity of system disruptions that are unanticipated or unpredictable. Furthermore, the DREAM can autonomously enable preventive and corrective action in response to unexpected system disruptions, increasing the likelihood of achieving intended system objectives without the need for human intervention. In practical terms, use of the DREAM could increase the operational availability of safety- and security-critical systems, such as ATM systems, for which disruptions can be catastrophic in nature. Beyond the ATM application in this research, the DREAM is targeted towards engineered systems which are susceptible to inevitable, yet unavoidable disruptions, such as natural disasters, intentional attacks, and human errors.

Table of Contents

Acknowledgements ..... **iv**

Abstract of Dissertation ..... **v**

Table of Contents ..... **vii**

List of Figures ..... **xii**

List of Tables ..... **xv**

List of Acronyms ..... **xvii**

Chapter 1: Introduction **1**

1.1 Problem Statement 1

1.2 Problem Background 2

1.2.1 CIKR Resilience Initiatives 2

1.2.2 Application Area – Air Traffic Management Systems 3

1.3 Research Objectives 4

1.4 Research Questions 5

1.5 Research Hypotheses 6

1.6 Research Contributions 9

1.7 Relationship to Prior Studies 12

1.8 Dissertation Outline 13

Chapter 2: Literature Review 15

2.1 Resilience Engineering 15

2.1.1 Sources of System Disruption 18

2.1.2 Differentiation between Resilience, Reliability, and Robustness 23

2.2 Artificial Intelligence and System Automation 25

2.2.1 System Automation Methods 27

2.3 Critical Infrastructure and Key Resources 35

2.3.1 ATM System Modernization Efforts 37

2.4 Related Studies 39

2.4.1 Krozel & Peters 39

2.4.2 Resmerita & Heymann 40

2.4.3 Ouedraogo, Enjalbert & Vanderhaegen 41

2.5 The Theory of Constraints 42

2.5.1 Evaporating Cloud Technique 44

2.6 Assessment of System Resilience Constraints 46

2.6.1 Forecasting Constraints in Emergent Environments 46

2.6.2 Resource Deconfliction Constraints 47

2.6.3 Manpower Constraints 48

2.7 Literature Search Summary 50

Chapter 3: Methodology 53

3.1 Implementation of the DREAM 55

3.1.1 Performance Monitoring and Appraisal Process 55
3.1.2 Resilience-Centered Analysis of Alternatives 58
3.1.3 Response Prioritization and Selection Process 60
3.1.4 Summary of DREAM Benefits 61
3.2 ATM Case Study 62
3.2.1 ATM System HASS 62
3.2.2 ATM System Performance Measures 63
3.2.3 Free Flight Modeling & Simulation 68
3.2.4 Monte Carlo Simulation 71
3.2.5 Agent-based Modeling and Simulation 73
3.4 HASS Experiment #1 82
3.4.1 System-level Test Case 83
3.4.2 Experiment #1 Setup and Execution 84
3.5 HASS Experiment #2 86
3.5.1 DRA Test Case 86
3.5.2 Experiment #2 Setup and Execution 87
Chapter 4: Results & Analysis 89
4.1 Results for Experiment #1 (HASS of λ and R_M) 90
4.1.1 Comparison of TCAS II and DREAM Results for Experiment 1 91
4.1.2 Comparison of D* Lite and DREAM Results for Experiment 1 92
4.2 Results for Experiment 2 (HASS of E, D_E, and SS) 93
4.2.1 Comparison of TCAS II and DREAM Results for Experiment 2 96

4.2.2 Comparison of D* Lite and DREAM Results for Experiment 2 97

4.3 Analysis of Results 98

4.3.1 Analysis and Findings for Research Question 1 99

4.3.2 Analysis and Findings for Research Question 2 102

4.3.3 Analysis and Findings for Research Question 3 105

4.3.4 Analysis and Findings for Research Question 4 108

4.3.5 Analysis and Findings for Research Question 5 111

4.3.6 Analysis Summary 114

Chapter 5: Conclusion 115

5.1 Summary of DREAM Capabilities and Benefits 115

5.2 Research Application and Findings 117

5.3 Limitations 118

5.4 Areas of Future Research 119

5.5 Conclusion 121

References 123

Appendices 141

Appendix A: Additional Background on Resilience Engineering 141

Appendix B: Overview of the U.S. National Airspace System 144

Appendix C: Random Number Generation for Monte Carlo Simulation 147

Appendix D: Normal Probability Plots of Paired Differences for Experiment 1 149

Appendix E: Normal Probability Plots of Paired Differences for Experiment 2 151

List of Figures

Figure 1-1 Relationship Between Research Questions and Hypotheses 8

Figure 2-1: Example of Deadlock due to Resource Contention 20

Figure 2-2: Quality of Service vs. Quality of Protection vs. Quality of Resilience 25

Figure 2-3: 2017 Trends in Artificial Intelligence & System Automation 27

Figure 2-4: Free Flight Concept of Operations for ATM Moderation 38

Figure 2-5: Generic EvC Diagram 45

Figure 2-6: EvC for Forecasting Constraints 47

Figure 2-7: EvC for Resource Deconfliction Constraints 48

Figure 2-8: EvC for Manpower Constraints 49

Figure 3-1: Research Framework 54

Figure 3-2: DREAM Inputs, Outputs and Processes 55

Figure 3-3: FFOA Model 70

Figure 3-4: Autonomous Aircraft Model 71

Figure 3-5: Monte Carlo Sampling Process for Random Waypoint Selection 72

Figure 3-6: Modular Simulation Architecture and Implementation Logic 73

Figure 3-7: TCAS II Thresholds for Collision Avoidance Maneuvers [RTCA, 2008] 74

Figure 3-8: Simulation Logic for TCAS II Implementation 74

Figure 3-9: Simulation Logic for D* Lite Implementation 75

Figure 3-10: Simulation Logic for DREAM Implementation 76

Figure 3-11: Overview of Analysis for Matched Data-sets 78

Figure 3-12: Example of System-level HASS Test Case 84

Figure 3-13: Test Model for System-level HASS 85

Figure 3-14: Example of DRA Test Case Simulation 87
Figure 3-15: Test Model for SoS-level HASS 88
Figure 4-1: Mean and SE of Incident Rate based on 20 simulations 90
Figure 4-2: Mean and SE of Mission Reliability 91
Figure 4-3: Mean and SE of the System Efficiency 94
Figure 4-4: Mean and SE of the Standard Deviation of Efficiency, D_E 95
Figure 4-5: Mean and SE of System Stability, *SS* 96
Figure 4-6: Analysis Outputs for Hypothesis 1A 100
Figure 4-7: Analysis Outputs for Hypothesis 1B 100
Figure 4-8: Analysis Outputs for Hypothesis 2A 103
Figure 4-9: Analysis Outputs for Hypothesis 2B 103
Figure 4-10: Analysis Outputs for Hypothesis 3A 106
Figure 4-11: Analysis Outputs for Hypothesis 3B 106
Figure 4-12: Analysis Outputs for Hypothesis 4A 109
Figure 4-13: Analysis Outputs for Hypothesis 4B 109
Figure 4-14: Analysis Outputs for Hypothesis 5A 112
Figure 4-15: Analysis Outputs for Hypothesis 5B 112
Figure A-1: Example of Performance Degradation with Full Recovery 141
Figure A-2: Example of a System Failure with Partial Recovery 141
Figure A-3: Example of System Stabilization after Disruption without Recovery 142
Figure A-4: Example of System or Mission Failure due to Insufficient Resilience 142
Figure A-5 : System Performance with and without Graceful Degradation 143
Figure B-1: Summary of NAS Facility Types & Functions [FAA, 2009] 145

Figure B-2: FAA Free Flight Tools and Application Areas [GAO, 1998, 2001] 146

Figure C-1: Examples of the Marsaglia Effect on Cartesian Coordinates 147

Figure D-1: Normal Probability Plot of $d_i s$ for Incident Rate (TCAS II & DREAM) ... 149

Figure D-2: Normal Prob. Plot of $d_i s$ for Mission Reliability (TCAS II & DREAM) ... 149

Figure D-3: Normal Probability Plot of $d_i s$ for Incident Rate (D* Lite & DREAM) 150

Figure D-4: Normal Prob. Plot of $d_i s$ for Mission Reliability (D* Lite & DREAM) 150

Figure E-1: Normal Prob. Plot of $d_i s$ for System Efficiency (TCAS II & DREAM) 151

Figure E-2: Norm. Prob. Plot of $d_i s$ for Std. Dev. of Efficiency (TCAS II & DREAM)... 151

Figure E-3: Normal Prob. Plot of $d_i s$ for System Stability (TCAS II & DREAM) 152

Figure E-4: Normal Prob. Plot of $d_i s$ for System Efficiency (D* Lite & DREAM)..... 152

Figure E-5: Norm. Prob. Plot of $d_i s$ for Std. Dev. of Efficiency (D* Lite & DREAM) 153

Figure E-6: Normal Prob. Plot of $d_i s$ for System Stability (D* Lite & DREAM)..... 153

List of Tables

Table 2-1: Resilience Design Principles for Engineered Systems 17

Table 2-2: Common Causes of System Disruption and Process Variation 18

Table 2-3: Basic Types of Human Error Modes [Hollnagel, 1998] 19

Table 2-4: LOA Scale. Adapted from [Endsley & Kiris,1995; Sheridan, 2011] 27

Table 2-5: Rule Types Applied in RBA [Moore et al, 2011; Lera et al, 2010] 30

Table 2-6: Common Critical Infrastructure Sectors and Examples 36

Table 2-7: Common Key Resource Categories and Examples 36

Table 3-1: Free Flight Model Parameters [Adapted from Kochenderfer et al., 2010] 71

Table 3-2: Paired t-test Hypotheses and Inferences on the Mean of Paired Differences . 81

Table 3-3: Bonferroni Correction for Paired *t*-test 82

Table 3-4: Model Inputs for System-level HASS 85

Table 3-5: Model Inputs for SoS-level HASS 88

Table 4-1: MEAN AND SE OF INCIDENT RATE 90

Table 4-2: MEAN AND SE OF MISSION RELIABILITY 91

Table 4-3: Percent Change in Results for Experiment 1– TCAS II vs. DREAM 92

Table 4-4: Percent Change in Results for Experiment 1– D* Lite vs. DREAM 93

Table 4-5: MEAN AND SE OF SYSTEM EFFICIENCY 94

Table 4-6: MEAN AND SE OF THE STANDARD DEVIATION OF EFFICIENCY, D_E 95

Table 4-7: MEAN AND SE OF SYSTEM STABILITY, *SS* 96

Table 4-8: Percent Change in Results for Experiment 2 – TCAS II vs. DREAM 97

Table 4-9: Percent Change in Results for Experiment 2 – D* Lite vs. DREAM 98

Table 4-10: Summary of Hypothesis Testing for Research Question 1 101

Table 4-11: Summary of Hypothesis Testing for Research Question 2 104

Table 4-12: Summary of Hypothesis Testing for Research Question 3 107

Table 4-13: Summary of Hypothesis Testing for Research Question 4 110

Table 4-14: Summary of Hypothesis Testing for Research Question 5 113

Table 4-15: Summary of Results for Research Hypotheses 114

List of Acronyms & Abbreviations

A/C	Aircraft
AAS	Autonomous Aircraft System
ABM	Agent-based Model
ABMS	Agent-based Modeling and Simulation
ACAS	Aircraft Collision Avoidance System
AI	Artificial Intelligence
AOA	Autonomous Operations Area
AS	Autonomous System
ATM	Air Traffic Management
AWS	Autonomous Weapon System
C2	Command and Control
CBA	Context-based Autonomy
CFR	Code of Federal Regulations
CI	Confidence Interval
CIKR	Critical Infrastructure and Key Resources
CoG	Center of Gravity
CONOPS	Concept of Operations
COOP	Continuity of Operations
D*	Dynamic A-star algorithm
DHG	Dynamic Heuristic Generation
DHS	Department of Homeland Security
DREAM	Disruption Resilience & Adaptation Model
EOE	Exceedance of Envelope
EUROCONTROL	European Organization for the Safety of Air Navigation
EvC	Evaporating Cloud
FAA	Federal Aviation Administration

FEMA	Federal Emergency Management Administration
FFOA	Free Flight Operation Area
FS	Fuel Shortage
FSM	Failsafe Maneuver
GNC	Guidance, Navigation and Control
HASS	Highly Accelerated Stress Screening
HITL	Human-in-the-Loop
HSC	Homeland Security Council
INCOSE	International Council on Systems Engineering
KPI	Key Performance Indicator
KPP	Key Performance Parameter
LBA	Learning-based Autonomy
LOA	Level of Automation
LoSS	Loss of Safe Separation
MAC	Mid-Air Collision
MBA	Model-based Autonomy
MEF	Mission Essential Function
MOA	Managed Operations Area
MOE	Measure of Effectiveness
NAS	National Airspace System
NMAC	Near Mid-Air Collision
nmi	Nautical Miles
Norm.	Normal
PIAC	Peak Instantaneous Aircraft Count
PIF	Performance Impact Factor
POOGI	Process of Ongoing Improvement
Prob.	Probability
QoP	Quality of Protection

QoR	Quality of Resilience
QoS	Quality of Service
RAMS	Reliability, Availability, Maintainability and Safety
RBA	Rule-based Autonomy
RIF	Resilience Impact Factor
RTCA	Radio Technical Commission for Aeronautics
SESAR	Single European Sky ATM Research
SoH	State of Health
SoI	System of Interest
SoS	System of Systems
Std. Dev.	Standard Deviation
TCAS	Traffic Alert and Collision Avoidance System
TOC	Theory of Constraints
TPP	Technology and Public Policy
U.S.	United States
UAS	Unmanned Aircraft System
UBA	Utility-based Autonomy

Chapter 1: Introduction

Unpredictable natural and man-made operating environments are inherent risks for the disruption of engineered systems. Disruptions include any events or conditions that adversely impact a system's *mission essential functions* which are capabilities that must be provided continuously or resumed rapidly after interruption to fulfill system requirements [Rucks et al., 2011; DHS, 2009]. The failure or degradation of mission essential functions can result in undesirable effects and/or catastrophic implications for the system and its stakeholders. For example, the disruption of safety- and security-critical systems can result in development of hazardous conditions or potential loss of life. Lacking the ability to anticipate and avoid all disruptions to be encountered, the need for *resilient systems* which integrate disruption prevention and mitigation measures is recognized in systems research and development [INCOSE, 2014, 2015; Haimes, 2012]. In practical terms, resilient systems are able to "perform their intended function in the face of changing circumstances or invalid assumptions" [INCOSE, 2014]. In the context of systems providing safety-critical and security-critical functionality, the resilience challenge is motivated by the potentially catastrophic implications of system disruption such as the loss of life [Setola et al., 2016; DHS, 2013].

1.1 Problem Statement

Disruptions to critical infrastructure systems have jeopardized public safety and economic stability worldwide resulting in thousands of deaths, millions of injuries, and billions in economic losses each year for the last two decades [Obolensky et al., 2019]. The U.S. Department of Homeland Security [2010] highlighted that "[d]espite our best efforts, some attacks, accidents, and disasters will occur. Therefore, the challenge is to

foster a society that is robust, adaptable, and has the capacity for rapid recovery." That is, disruptions are inevitable and diverse in origin, including causes that are intentional, inadvertent, naturally occurring, and man-made. In recent years, the leading causes of disruptions to *critical infrastructure and key resources* (*CIKR*) which are relied upon for public safety, security, and economic stability include [Schwab, 2017; Setola et al., 2016]: (i) aging systems with declining reliability and availability due to maintenance needs; (ii) increasingly complex systems that elude precise monitoring and control; (iii) a high degree of interdependence amongst systems contributing to cascading failures; (iv) growing scarcity of natural resources, such as fossil fuel; and (v) volatile social and economic conditions which cause rapid surges in demand for goods and services, such as refugee crises. Naturally occurring events and conditions, such as extreme weather, are also a perennial source of disruptions to CIKR. As the population grows, an increasingly target-rich environment for human casualties and economic losses are an inherent risk of CIKR disruption.

1.2 Problem Background

1.2.1 CIKR Resilience Initiatives

Historically, CIKR system management has narrowly focused on disruption prevention and recovery activities, such as the deterrence of attacks, preventive maintenance, and emergency preparedness/response [DHS, 2009; Hemme, 2015]. As the population grows and uncertainty surrounding natural and man-made resources increases, CIKR management priorities have shifted towards three concurrent priorities [Hemme, 2015; Arsenault & Sood, 2007]:

- *Efficiency optimization:* avoiding or minimizing the waste of resources;

- *Stakeholder equity:* ensuring fairness with respect to resource utilization amongst stakeholders; and
- *System resilience:* the ability to avoid, adapt to, withstand or recover from anticipated and emergent disruptions.

Furthermore, *system automation* is increasingly being adopted in CIKR systems. System automation enables the partial or complete replacement of a human with a machine for system *command and control (C2)*, encompassing planning, directing, or authorizing the execution of tasks to meet targeted goals and objectives [Sheridan, 2011]. More broadly, system automation has successfully supported performance optimization and disruption mitigation for engineered systems within the scope of anticipated needs. However, existing automation methods generally lack adaptability when emergent issues arise, often requiring human intervention as a contingency to system failure or unacceptable performance degradation. As applied to CIKR systems, automation methods that can mitigate both anticipated and emergent disruptions without human intervention are sought.

1.2.2 Application Area – Air Traffic Management Systems

The challenges of CIKR management are exemplified in Air Traffic Management (ATM) systems given a combination of: (1) high susceptibility to weather disruptions; (2) finite airspace as a resource for routing and deconflicting aircraft; and (3) potentially catastrophic consequences of failure or degraded performance such as aircraft collisions. [Marshall et al., 2018a]. As the demand for finite airspace grows, the aviation engineering and regulatory communities have targeted improvements to ATM safety, flexibility, and efficiency by increasing autonomous navigation and collision avoidance

capabilities, collectively referred to as *Free Flight initiatives* (detailed in Section 2.3) [GAO, 1998, 2011]. The long-term objective of Free Flight initiatives is fully autonomous aircraft navigation without pre-coordinated flight plans or human air traffic controllers for airspace deconfliction. This real-world problem motivated the selection of Free Flight ATM as the application area for this research. This research considered the ATM system holistically, rather than its software and hardware elements individually, addressing the physical, logical, technical and administrative requirements and constraints which are susceptible to disruption.

1.3 Research Objectives

The primary objective of this research was to provide a model for adapting system responses to emergent disruptions without human intervention. Several sub-objectives were established for this research:

1. Identify and model cause and effect relationships for the disruption of engineered systems in emergent operating environments;
2. Propose countermeasures to overcome or offset constraints on anticipatory knowledge for the prevention of system disruptions;
3. Formulate and implement a system C2 model that enables proposed countermeasures without a human in the loop;
4. Characterize and evaluate the C2 model in comparison to existing system automation standards and conventions;

The overarching research objective and sub-objectives was applied to Free Flight ATM as the system of interest in this research (as introduced in Section 1.2.2). However, the

potential extensibility of research findings and notional objectives for future research are explored (as discussed in Chapter 5, Section 5.4).

1.4 Research Questions

The experiments performed as a part of this research addressed the following questions:

1. Does the developed model reduce the frequency of system disruptions in emergent operating environments when compared to rule-based and utility-based system automation methods?
2. Does the developed model improve mission reliability in emergent operating environments when compared to rule-based and utility-based system automation methods?
3. Does the developed model improve system efficiency in emergent operating environments when compared to rule-based and utility-based system automation methods?
4. Does the developed model reduce the standard deviation of system efficiency in emergent operating environments when compared to rule-based and utility-based system automation methods?
5. Does the developed model improve system stability when compared to rule-based and utility-based system automation methods?

Each research question is aligned with one of the CIKR management priorities (discussed in Section 1.2.1). The first three research questions are associated with system resilience

capabilities, including disruption avoidance, withstanding, adaptation, and recovery. The fourth research question is aligned with the aim of efficiency optimization in CIKR systems that manage or utilize limited resources. The fifth research question addresses the priority of stakeholder equity in engineered systems that manage or utilize shared resources for mission execution.

1.5 Research Hypotheses

As listed below and illustrated in Figure 1-1, a total of 10 hypotheses are associated with research questions presented in Section 1.4:

***H_{1A}**: There is a statistically significant decrease in the frequency of system disruptions via the DREAM in comparison to TCAS II for Free Flight ATM.*

***H_{1B}**: There is a statistically significant decrease in the frequency of system disruptions via the DREAM in comparison to the D* Lite for Free Flight ATM.*

***H_{2A}**: There is a statistically significant increase in mission reliability achieved via the DREAM in comparison to TCAS II for Free Flight ATM.*

***H_{2B}**: There is a statistically significant increase in mission reliability achieved via the DREAM in comparison to D* Lite for Free Flight ATM.*

***H_{3A}**: There is a statistically significant increase in system efficiency via the DREAM in comparison to TCAS II for Free Flight ATM.*

***H_{3B}**: System efficiency via the DREAM is reduced, but comparable to the level achieved by D* Lite for Free Flight ATM.*

***H_{4A}**: There is a statistically significant decrease in the standard deviation of system efficiency via the DREAM in comparison to TCAS II for Free Flight ATM.*

***H_{4B}**: The standard deviation of system efficiency via the DREAM is increased, but comparable to the level achieved by D* Lite for Free Flight ATM.*

***H_{5A}**: System stability via the DREAM is reduced in comparison to TCAS II for Free Flight ATM, but a net decrease in system disruptions is still achieved.*

***H_{5B}**: There is a statistically significant increase in system stability via the DREAM in comparison to D* Lite for Free Flight ATM.*

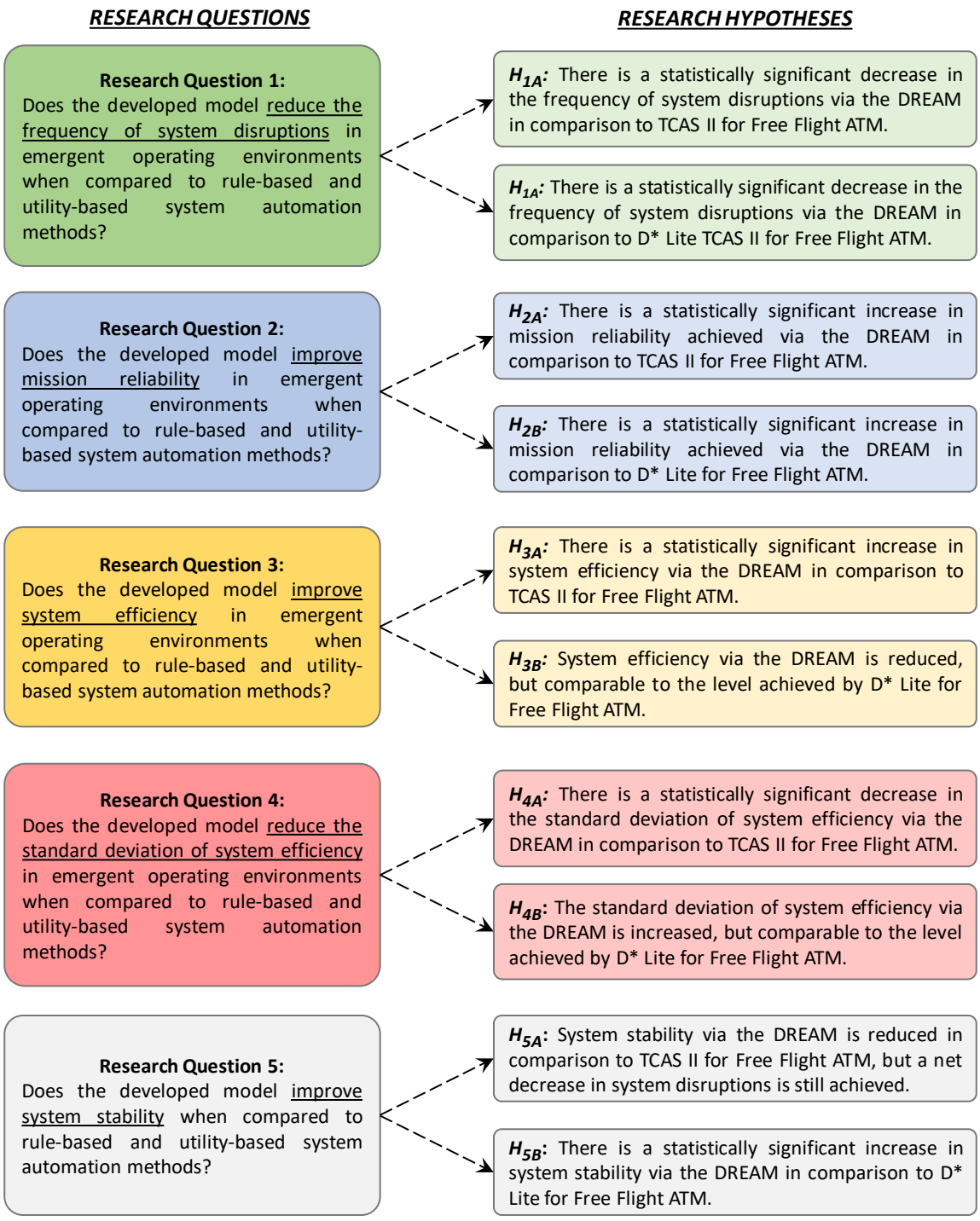


Figure 1-1 Relationship Between Research Questions and Hypotheses

1.6 Research Contributions

This research contributes to the body of knowledge surrounding resilience engineering by providing an adaptive and autonomous C2 model for responding to emergent system disruptions. Existing research on system resilience largely focuses on strategies and techniques for disruption avoidance through anticipation, such as failure prognostics, vulnerability testing, and heuristics for system design based on historical precedents. The model presented in this research complements and expands upon the existing body of knowledge by addressing the following constraints and limitations:

a. Reliance on human intervention as a contingency for inoperable or ineffective systems is recognized as a best-practice, however, increasing system complexity and human factors can limit or invalidate a human in the loop for system command and control;

b. System automation has been successfully adopted for disruption avoidance and performance optimization within the scope of anticipated needs and risks, but existing implementation methods lack adaptability when emergent issues and risks arise;

c. Existing resilience metrics and assessment methodologies are often domain-specific which limits their extensibility.

The model integrates satisficing, coopetition, and autonomic control functionality to provide the following benefits in comparison to existing automation methods in response to emergent system disruptions:

- System automation does not rely on specific, pre-planned actions and reactions, whereas rule-based methods strictly define and enforce all system behavior without flexibility to deviate;
- System responses are not dependent on previously observed patterns of success or failure based on training and/or calibration periods in the target operating environment, contrasting with learning-based methods; *and*
- System availability measures are used for performance feedback and prognostics which are tailorable to the system of interest, unlike hyper-heuristic and metaheuristic methods which rely on metrics specified for existing problem archetypes and solutions.

The contributions of this research are also presented in multiple conference papers and peer-reviewed journal articles:

- A conference paper presented at the 2017 Conference on Systems Engineering Research [Marshall, Roberts & Grenn, 2017a] discusses the potential for optimization processes to inadvertently generate sub-optimal outcomes with respect to performance quality and efficiency. A conceptual decision model was introduced that targeted system self-regulation between optimization-seeking behavior and satisficing behavior based on performance feedback. The key contributions of this paper are a causal analysis of the research problem and an analysis of alternatives for conceptual development of the DREAM, as discussed in Chapter 3;

- A conference paper presented at the 2017 International Conference on Control, Automation, and Robotics [Marshall, Roberts & Grenn, 2017b] addresses the inherent uncertainty in anticipating and avoiding system disruptions before system deployment. Resilience engineering principles were applied to conceptualize a system C2 model for disruption mitigation without preplanning or human intervention as an operational contingency. The contributions associated with this paper are discussed in Chapter 3 with respect to architecture development and an analysis of alternatives for implementation of the DREAM;

- An article in *IEEE Systems Journal* [Marshall, Roberts & Grenn, 2018] analyzes existing and emerging approaches for implementing system automation with focus on adaptability under operational uncertainty. A C2 model for autonomous systems operating in emergent environments is presented, including the development of satisficing heuristics based on system availability. The C2 model is applied to the Free Flight ATM problem using agent-based modeling and simulation and performance is compared to existing standards for ATM automation. The results and analysis for this study are presented in Chapter 4, Section 4.1. The major contribution of this publication is a reference model for C2 of systems jointly concerned with disruption avoidance and adaptability to enhance mission reliability in emergent operating environments.

- A journal article currently in-review [Marshall, Roberts, Grenn & Holzer, 2019] focuses on the risk of system disruption due to availability conflicts for

shared resources amongst uncoordinated stakeholders. Leveraging coopetition principles, a C2 model for deconflicting shared resources without complete anticipatory knowledge of stakeholder demands is presented. The Free Flight ATM problem is used to evaluate model performance via ABMS and results are compared to existing alternative methods. The findings and conclusions associated with this study are presented in Chapter 4, Section 4.2. This publication provides a reference model for C2 of systems that rely on shared resources for mission execution in which efficiency and fairness of resource utilization are concurrent priorities.

This research is timely given ever increasing uncertainty and volatility of natural and man-made environments that have resulted in unanticipated disruptions to CIKR systems relied on for public safety and security [Schwab, 2017; Setola et al, 2016].

1.7 Relationship to Prior Studies

This research complements and expands upon prior studies addressing risk management and system resilience in emergent operating environments. The major thrusts of prior studies are summarized below:

- *Resilience Quantification & Assessment:* diagnostic and prognostic procedures and metrics to characterize system performance before, during, and after disruptions;
- *Resilient Architecture & Design:* the development and application of system design methods, heuristics, and patterns focused on disruption avoidance and mitigation;

- *Autonomous Fault Management:* enabling the detection and response to anomalous, non-conforming and hazardous conditions in engineered systems without human intervention.

The literature review presented in Chapter 2 elaborates on the related research thrusts and substantiates the need for additional research beyond prior studies.

1.8 Dissertation Outline

This dissertation is organized into five chapters as follows.

Chapter 1 presents the research problem, specifies research objectives and introduces the system of interest selected for research application. A summary of research contributions to the bodies of knowledge surrounding risk management and system resilience is also provided, including a summary of publications.

Chapter 2 provides an exhaustive literature review of the core subjects and prior studies related to the research. Major topics include: (i) resilience engineering objectives; (ii) sources of system disruptions; and (iii) system automation methods. The chapter concludes with the presentation of the *Disruption Resilience and Adaptation Model (DREAM)* which implements proposed countermeasures to emergent system disruptions based on research objectives and findings.

Chapter 3 presents the testing and evaluation methodology used to assess the DREAM, including discussion of the measures of effectiveness (MOEs) for system resilience in the context of the ATM application. The development of a modeling and simulation capability to apply the DREAM to the Free Flight ATM problem (requiring autonomous system C2 in an emergent operational environment) is also detailed.

Chapter 4 presents simulation results and analysis, including comparison of the DREAM to ATM standards and conventions based on existing system automation approaches. Statistical analysis of results via paired *t*-testing and the construction of *t*-confidence intervals are also presented.

Chapter 5 summarizes research findings and suggests areas of future research related to the application of the DREAM and broader challenges surrounding risk management, emergence, and resilience across the system lifecycle.

Chapter 2: Literature Review

The literature review for this research focuses on: (1) system resilience motivations; (2) resilience engineering objectives and techniques; and (3) system automation techniques as an enabler of system resilience. Relevant findings on system resilience motivations were presented in Chapter 1 as part of the research introduction and background. Findings on the remaining areas of concentration for the literature search are presented in this chapter in addition to a summary of relevant prior studies.

2.1 Resilience Engineering

Resilience engineering focuses on the proactive protection of organizations and engineered systems from disruption, extending beyond failure mitigation to prevention of any unacceptable performance degradation or perturbation [Madni & Jackson, 2009]. As explained by Arsenault & Sood [2007], the overarching goal in resilience engineering is to increase the likelihood of a system "thriving, not just surviving," when disruptions are encountered. Resilience engineering can also be considered at varying scales and levels of abstraction of a system, such as an individual component, sub-system, system-of-systems, or the system architecture.

A diverse set of perspectives on resilience are observed in the literature, such as psychological resilience to stress, business resilience to market competition and biological resistance to disease [Wang, 2018]. Within the context of engineered systems, Madni & Jackson [2009] define resilience based on four capabilities for disruption mitigation:

- *Avoidance:* proactive action taken to circumvent a known or anticipated disruption;

- *Adaptation:* reactive measures taken to circumvent an unexpected or unanticipated disruption;
- *Withstanding*: passive and active measures for enduring and/or resisting disruptions;
- *Recovery:* partial or full restoration of system functionality following a disruption.

Appendix A presents notional disruption scenarios that illustrate and distinguish system performance with and without resilience capabilities.

Jackson & Ferris [2013] specify four system attributes targeted by resilience engineering which complement and supplement the core resilience capabilities:

- *Capacity:* the degree to which the system can resist disruption before functionality is adversely impacted;
- *Flexibility:* the feasibility and ease with which a system can be adjusted before, during or after a disruption;
- *Tolerance:* the degree of system insensitivity during a disruption and repairability after encountering a disruption;
- *Cohesion:* integration of system elements such that required functionality is achieved with sufficient independence or isolation to mitigate cascading failures.

Building on the resilience attributes, Jackson & Ferris (2013) outlined fourteen design principles to support the development of resilient engineered systems (Table 2-1). The design principles can be applied independently or in combination based on the needs and constraints of the system of interest.

Table 2-1: Resilience Design Principles for Engineered Systems [Jackson & Ferris, 2013]

Resilience Attribute	Design Principles
Capacity	The *absorption design principle* calls for the system to be designed with adequate margin to withstand a design-level threat.
	The *physical redundancy* principle states that resilience is enhanced when critical components are physically redundant.
	The *functional redundancy* principle calls for critical functions to be duplicated using different means.
	The *layered defense* principle states that single points of failure should be avoided.
Flexibility	The *reorganization* principle states that the system should be able to change its own architecture before, during, or after the encounter with a threat. This design principle is applicable particularly to human systems.
	The *human in the loop* principle states that there should be a human in the system there is a need for human cognition (especially when unprecedented threats are involved). However, this principle has the caveat that it will only be effective if a human will be in a position to assimilate the relevant information and perform an appropriate action quickly enough to be effective.
	The *complexity avoidance* principle calls for the minimization of complex elements, such as software and humans, except where they are essential or beneficial (e.g. the human in the loop principle).
	The *drift correction* principle states that detected threat conditions should be corrected before encountering the threat.
Tolerance	The *localized capacity* principle states that, when possible, the functionality of a system should be concentrated in individual nodes of the system and stay independent of other nodes.
	The *loose coupling* principle states that cascading failures should be counteracted by inserting pauses between the nodes.
	The *neutral state* principle states that systems should be brought into a neutral state before actions are taken.
	The *reparability* principle states that systems should be reparable to bring the system back to full or partial functionality.
Cohesion	The *inter-node interaction* principle requires that nodes (i.e. elements) of a system be capable of communicating, cooperating, and collaborating with each other.
	The *reduce hidden interactions* principle states that efforts should be made to assure that potentially harmful interactions between nodes of the system should be reduced.

2.1.1 Sources of System Disruption

Madni and Jackson [2009] characterize the sources of system disruption as having "many faces" and the ability to "take a variety of forms". In other words, system disruptions stem from numerous and diverse causes which are anticipated, unexpected, inadvertent, man-made and naturally-occurring in origin. As presented in Table 2-2, the common causes of system disruption and process variation are associated with six general categories, including: *manpower*, *machines*, *materials*, *methods*, *measurements*, and the *environment* [Tague, 2005]. Madni and Jackson [2009] also distinguish *systemic disruptions* which arise from the failure or degradation of a function provided by the system from *external disruptions* which are caused by factors beyond they system's control. The literature identifies errors and uncertainty associated with manpower, and more specifically a human in the loop, as a major source of systemic disruptions. With respect to external disruptions, uncertainty in the environment and growing demand for limited natural and man-made resources are endemic resilience challenges for engineered systems [Madni & Jackson, 2009; Setola et al., 2017].

Table 2-2: Common Causes of System Disruption and Process Variation [Tague, 2005]

Disruption/Variation Category	Failure, Degradation or Limitation related to…
Manpower	Human involvement in system operation and sustainment, e.g., human error, fatigue, distraction.
Machines	Tools and equipment, e.g., scheduled or unscheduled maintenance of tools
Materials	Raw materials, consumable materials, and information.
Methods	Processes and procedures, e.g., software error, use of the incorrect procedure
Measurements	Quantitative/qualitative inspection, timing and/or synchronization capability
Environment	Naturally occurring phenomenon and man-made required or imposed on the system, e.g., weather, facilities

2.1.1.1 Manpower-related Disruptions

Maintaining a human-in-the-loop for engineered systems is a generally desirable practice in resilience engineering, especially in safety-critical and security-critical applications [Jackson & Ferris, 2013]. However, human involvement for system monitoring and control can be constrained or precluded due to: (1) physiological stressors and limitations inherent to the system or operating environment; (2) system complexity which can exceed cognitive limitations; and (3) time constraints for preventive or corrective action that cannot be met due to inefficiency [Dekker, 2005; Madni et al., 1985]. Additionally, Hollnagel [1998] identifies eight basic "error modes" associated with human behavior that can cause or exacerbate system disruptions Table 2-3. As elaborated by Madni & Jackson [2009]:

> *"When it comes to disruptions, human agents are a double-edged sword. On the one hand, they can act unpredictably…On the other hand, their ingenuity is what makes them most adaptable and capable of responding to disruptions...Despite extensive training, preparation and/or experience humans can still act in a completely unpredictable fashion."*

Given the susceptibility of a human in the loop to systemic and external disruptions, human factors such as ergonomics and human-machine interaction have been extensively researched in the resilience engineering domain.

Table 2-3: Basic Types of Human Error Modes [Hollnagel, 1998]

Error Mode	Human Error Results in…
Timing	Action performed too early or too late
Duration	Action performed too briefly or for too long
Distance	Object or control moved too short or too far
Direction	Action performed in the wrong direction
Speed	Action performed too slowly or too fast
Force/Power/Pressure	Action performed with too little or too much force
Object	Action performed on the wrong object (or control)
Sequence	Two or more actions performed in the wrong order

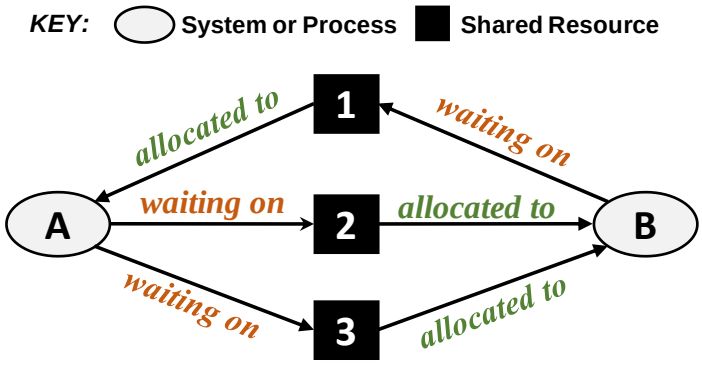
2.1.1.2 Resource Contention

Systems that depend on one or more shared resources for mission execution can be disrupted when resource demand exceeds the available supply. For example, the required type or amount of manpower, machines, materials or facilities needed to satisfy requirements may already be in-use or unavailable due to an unforeseen disruption. *Resource contention* occurs when conflicts over resource access or usage occur between its multiple stakeholders, such as that observed during power shortages and roadway congestion. Three general types of disruption are associated with resource contention:

a. *Resource starvation*: one or more stakeholders are perpetually denied access to a shared resource due to monopolization by other stakeholders;

b. *Livelock*: negotiation for the use of a resource in conflict amongst multiple stakeholders occurs in perpetuity, without any stakeholder gaining access;

c. *Deadlock*: the availability of a one or more shared resources is contingent upon mutually exclusive events or conditions, resulting in the disruption of all dependent systems or processes (Figure 2-1).



Note: System A and B require resources 1, 2, and 3 to execute their intended functionality. No alternatives are available as substitutes for resources 1, 2, or 3.

Figure 2-1: Example of Deadlock due to Resource Contention

To avoid or mitigate disruptions due to resource contention, two general strategies are commonly adopted for conflict resolution [Gianetto & Heydari, 2015; Dagnino, 2009]:

a. *Competitive strategies* involve the use of adversarial or hostile behavior amongst stakeholders to gain or maintain access to required resources when conflicts arise;

b. *Cooperative strategies* require teaming and collaboration and amongst stakeholders that require compromise and consensus to resolve conflicts.

When a competitive strategy is used, Tanenbaum [2001] highlights the increased risk for system disruption due to resource starvation. In contrast, the use of a collaborative strategy is susceptible to livelocks. Notably, both conflict resolutions strategies are subject to the deadlock scenario [Powell, 2007; Winston & Goldberg, 2004]. The concept of *coopetition* seeks to integrate the use of competitive and cooperative strategies amongst competing entities such that mutual benefits exceed what can be achieved via a competition or collaboration alone [Dagnino, 2009; Dagnino & Padula, 2002]. The concept of coopetition has been applied in a system-of-systems context in the literature with focus on the adaptation and evolution of systems to manage "conflicts between local versus global goals, motivations, and behaviors" [Gianetto & Heydari, 2015].

2.1.1.3 Emergent Operating Environments

The environment of a system consists of "a set of elements and their relevant properties, which are not part of the system but a change in any of which can produce a change in the state of the system" [Ackoff, 1971]. Consequently, engineered systems are susceptible to the disruption of their intended functionality when unanticipated changes in the operating environment are encountered. Both physical and non-physical changes in

a system's operating environment can result in disruptions, such as a natural disaster or a change in regulation.

Unanticipated changes in a system's operating environment result in three types of uncertainty for performance optimization and risk mitigation [Milliken, 1987]: (i) *state uncertainty* which reflects an incomplete or inaccurate understanding of how the environment has changed or is changing; (ii) *effect uncertainty* which is the inability to predict the impact of environmental changes on future states of the system; and (iii) *response uncertainty* defined as the lack of knowledge of response options and/or the an inability to predict the likely consequences of a response. To mitigate state uncertainty, effect uncertainty and response uncertainty, characterization of engineered systems can be performed based on the potential environments and constraints that may be encountered. The literature differentiates amongst several types of system testing and characterization methods which target the reduction of uncertainty associated with the naturally occurring or man-made operating environments [McLean, 2009; Natella, Cotroneo & Medeira, 2016]:

- *Highly Accelerated Life Testing:* repetitively testing the operation of a product or system at its design limitations until failure with the goal of proactively implementing and validating preventive or corrective actions to increase reliability. Common design specifications and failure modes assessed via highly accelerated life testing include: (i) cycling between upper and lower operating temperature and humidity levels; (ii) mechanical vibration and/or shock; and (iii) continuous operation under the maximum specified mechanical or electrical load.

- *Highly Accelerated Stress Screening:* testing the operation of a product or system at levels that increasingly exceed its intended design limitations to characterize the margin for failure and/or degradation of performance. Unlike highly accelerated life testing, the objective of highly accelerated stress screening is to expose and characterize unknown system failure modes and performance degradation, rather than assessing system reliability in response to known failure modes;

- *Fault Injection Testing:* software focused testing which is used to characterize system responses to invalid inputs or unexpected user behavior. Common design specifications and failure modes assessed with Fault Injection Testing include [Tao, Ezhilchelvan & Shrivastava; 1999]: (i) omission faults in which an expected action or response to a system input does not occur; (ii) value faults in which invalid or erroneous response from the system is observed; (iii) timing faults for which the system responds outside to the required timeframe; and (iv) arbitrary faults in which the system deviation stems from a combination of an omission, value or timing fault.

Environmental testing and characterization methods can be implemented via instrumented testing of the actual system, modeling and simulation techniques, or emulation techniques which combine actual and simulated elements of an engineered system and its environment for performance evaluation.

2.1.2 Differentiation between Resilience, Reliability, and Robustness

The literature clarifies that resilience is differentiated from the concepts of system reliability and robustness, which are often conflated in the discussion of risk and

uncertainty [Woods, 2015; Capano & Woo, 2018]. Resilience is distinguished from reliability and robustness by Zhang & Luttervelt [2011] as follows:

> *"The [system's] resilience differs from the reliability in that the reliability is a property of the system to measure the life of the system (i.e., sustaining success of the system), and the [system's] resilience differs from the robustness in that the latter is a property of the system to measure the insensitivity to disturbances of the system when the system is in operation."*

Based on this distinction, the concept of resilience can be viewed as complementary to: (a) reliability with respect to the avoidance of systemic disruptions; (b) robustness with respect to withstanding disruptions based on system capacity and tolerance attributes; and (c) both reliability and robustness in terms of the system's ability to sustain an acceptable level of performance. However, resilience is also uniquely concerned with the ability of the system to recover from disruptions. Tapolcai et al. [2005] established an alternative characterization of resilience that demonstrates the distinction from reliability and robustness based on three complementary perspectives of system quality (Figure 2-2):

a. *Quality of Service* is concerned with reliability and robustness which enable the system to remain operable and maintain an acceptable level of performance for its intended functions;

b. *Quality of Protection* focuses on system prevention and recoverability from unacceptable performance degradation or failure that would result in the system becoming inoperable or otherwise unable to fulfill its intended functions;

c. *Quality of Resilience* jointly considers the system's quality of service and quality of protection which aligns with the resilience paradigm of enabling prevention, survival and recovery from disruptions.

When considered independently, the quality of service and quality of protection perspectives provide an incomplete or misleading characterization of system performance with respect to disruption mitigation. The quality of resilience perspective of system performance is comprehensive in that it considers both preventive and corrective action in response to disruptions [Tapolcai et al., 2005].

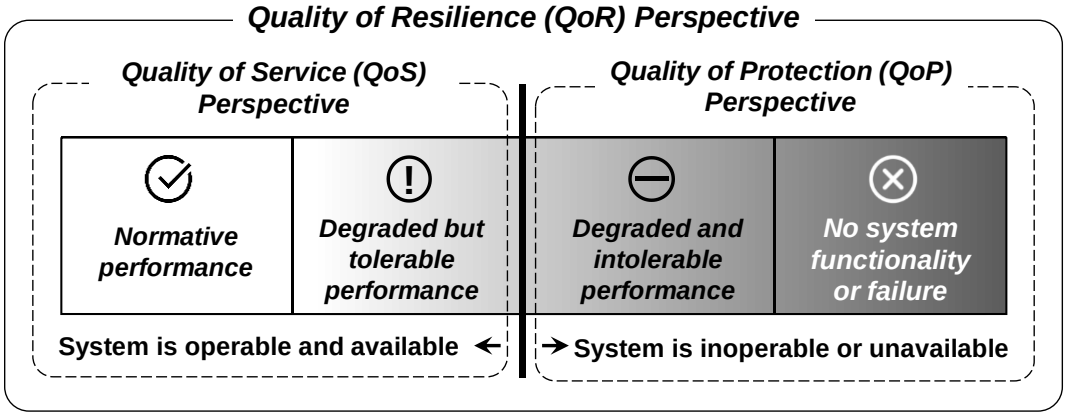


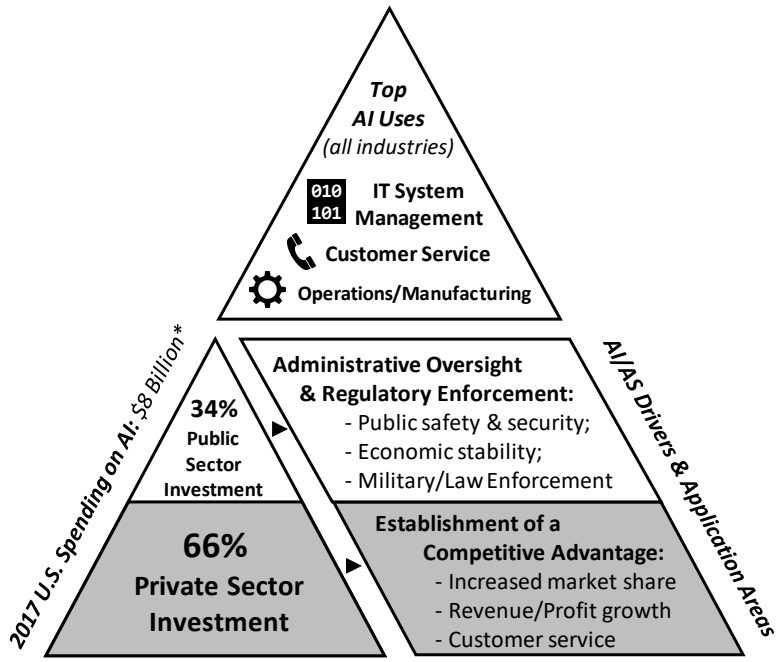
Figure 2-2: Comparison of Quality of Service vs. Quality of Protection vs. Quality of Resilience

2.2 Artificial Intelligence and System Automation

Artificial intelligence is "the science and engineering of making intelligent machines" McCarthy [2007]. Sauter [2010] clarifies that artificial intelligence involves the emulation of human expertise and decision-making without the disadvantages of forgetfulness, bias, fatigue or distractions that humans are prone to. Artificial intelligence has been broadly adopted across the public and private sectors for *system automation* which enables the partial or complete replacement of human operators by machines

(Figure 2-3) [Daquila et al, 2017, Ransbotham et al, 2017]. Automation has been employed for all aspects of system *command and control (C2)* consisting of planning, directing, authorizing, supervising and executing tasks or activities to fulfill targeted goals and objectives.

The literature distinguishes two general types of system automation [Dorf & Bishop, 2011; Pachter & Chandler, 1998]: (a) *automated control* which involves machine execution of prescribed/preset tasks or procedures without human intervention and (b) *autonomous control* that enables machine self-direction and execution of tasks or procedures without human involvement. The *level of automation (LOA)* indicates the allocation of responsibility and authority for system C2 between a human and machine element [Sheridan, 2011; Endsley & Kiris, 1995]. As shown in Table 2-4, a system's LOA can span between the extremes of manual control where a human performs all C2 and autonomous control for which all C2 tasks are performed by machine. Between these extremes, several automated C2 schemas with shared allocation of responsibility and authority between humans and machines exist [Sheridan, 2011]. Stone et al. [2016] observed that the majority of system automation is "highly tailored to particular tasks" and enabled via automated control. Automated control has historically excelled in supporting efficiency optimization and failure mitigation within the scope of required or anticipated system needs, such as automated fire detection and suppression systems. However, given the development of unanticipated or emergent conditions, system automation may be unsupported or unreliable, as exemplified by failures of automated speech-recognition systems and self-driving vehicles operating outside of expected environments [Kalra & Paddock, 2016; Goldwater, Jurafsky & Manning, 2010].



Note: On a geographic basis, 80% of all spending on AI and cognitive systems is based in the U.S. Total spending worldwide on AI and cognitive systems in 2017 was estimated at $12 Million [Daquila et al, 2017].

Figure 2-3: 2017 Trends in Artificial Intelligence & System Automation

Table 2-4: LOA Scale. Adapted from [Endsley & Kiris,1995; Sheridan, 2011]

Level of Automation		Roles and Responsibilities	
		Decision-making	Action Execution
1	None (Manual)	Human	Human
2	Decision Support	Shared	Human
3	Consensual AI	Shared	Shared
4	Monitored AI	Machine/AI	Shared
5	Full Automation	Machine/AI	Machine/AI

2.2.1 System Automation Methods

The implementation of system automation scales in complexity with the targeted LOA and objectives in replacing human decision making or operational control. The higher the degree of autonomous control required or uncertainty surrounding functional requirements and constraints, the greater the complexity of implementation. Furthermore,

implementation methods for system automation can generally be classified into two categories: (a) *rules-based approaches* define specific system actions and reactions to be executed under a given set of circumstances and (b) *principles-based approaches* specify desired outcomes for the system with a minimal set of rules imposed to mitigate unintended outcomes. Polacek et al. [2012] provided the following comparison of rules-based and principles-based approaches:

> *"A rules-based approach controls the actions that an agent should take when interacting with other agents or the environment. This greatly constrains the autonomy of agents (which in fact is the intent of the rules)…In contrast, a principles-based approach influences desired outcomes or goals that each agent or group of agents pursues as opposed to specifying their actions. Principles preserve the autonomy of agents, leading to greater adaptability to unforeseen circumstances."*

Existing methods for implementing system automation via the rules-based and principles-based approaches are reviewed in the following sections.

2.2.1.1 Rules-based Methods for System Automation

The rules-based approach for system automation has many well-established methods and conventions [Ingrand & Ghallab, 2017; Krupitzer et al., 2015; Lalanda et al., 2014]:

a. *Rule-based Autonomation (RBA)* establishes pre-determined responses to system inputs or internal conditions based on arbitrary or empirically-informed rules, policies, or constraints;

b. *Utility-based Automation (UBA)* scores and ranks alternative actions based on their actual or perceived preferability with respect to targeted objectives. The most preferable actions available, as assessed, are selected and executed;

c. *Model-based Autonomation (MBA)* abstracts the system and its environment using functional or mathematical models to enable decision analysis and execution on a diagnostic and/or prognostic basis;

d. *Learning-based Autonomation (LBA)* identifies and exploits beneficial patterns/trends based on historical system performance and/or observation of the operating environment, optimizing system efficiency and efficacy over time.

Rules-based methods have successfully supported a wide range of system optimization and failure prevention applications. For example, automated manufacturing, healthcare robots, credit card fraud monitoring, and autonomous vehicles are largely enabled by rules-based methods [Stone et al. 2018; Nof, 2009]. Notably, RBA and UBA are the predominant methods adopted for system automation [Lalanda et al., 2014].

2.2.1.1.1 RBA Implementation

The RBA method is implemented using "conditional specifications that instruct, permit, trigger, and inhibit processes, functions, and actions" based on system inputs, operational states or environmental conditions, referred to as *rules* or *policies*. Four general rule types are applied in RBA [Moore, Hu & Jackson, 2011]:

a. *Derivation rules*: specify conditional relationships between information such as sensor readings and a defined system operating state;

b. *Integrity constraints:* rules used to verify system conformance to requirements and constraints and make inferences about the existence of error conditions;

c. *Transformation rules*: establish conditional relationships between rules based on the transitive property of equality, e.g., if $A = B$ and $B = C$, then $A = C$;

d. *Reaction rules*: trigger an action once a specified condition or set of conditions are detected.

The definition and enforcement of rules for RBA is typically achieved using Boolean logic and programming constructs such as IF-THEN statements and DO-WHILE loops (Table 2-5).

Table 2-5: Rule Types Applied in RBA [Moore et al, 2011; Lera et al, 2010]

Rule Type	Description	Example Implementation
Derivation	Specification of a conditional relationship	*Rule 1:* **IF** *condition A* is **TRUE, THEN** *condition B* is **TRUE**
Integrity Constraint	Verification of system conformance to specified rules	*Given rule 1 (in the example above), the following rule is established:* *Rule 2:* **IF** *condition A* is **TRUE AND** *condition B* is **FALSE THEN** an *error condition* has occurred
Transformation	Establish relationships between rules based on the transitive property	*Given rule 1 and rule 2 (in examples above), the following rule can be established:* *Rule 3:* **IF** *condition A* is **TRUE, THEN** *condition C* is **TRUE**
Reaction	Trigger an action once a specified condition or set of conditions is met	*Rule 4:* **IF** *condition C* is **TRUE, THEN EXECUTE** *action 1*

2.2.1.1.2 UBA Method

UBA methods are implemented with *utility functions* which assess the actual or perceived preferability of decision alternatives to drive decision-making. Utility functions are typically developed on the basis of economic factors, e.g., expected cost, return on investment, or efficiency. Two utility models are commonly applied in UBA, as follows.

The *maximin model* is used to select the decision alternative that maximizes potential gains while assuming the worst case for potential losses, represented by the *maximin value* (v^*) [Mashcler, Sloan & Zamir, 2013]. The maximin value is given by:

$$v^* = \max_{p \in P} \min_{s \in S(p)} U(p, s) \tag{2.1}$$

where P denotes the set of all potential actions that can be taken by the decision-maker, $S(p)$ denotes the set of conditions and constraints associated with a potential action p, and $U(p, s)$ is the utility function used to assess the known or perceived value of outcomes associated with action p given state s.

The *minimax model* is used to select the decision alternative that minimizes the maximum potential losses, represented by the *minimax value* (v°) [Mashcler, Sloan & Zamir, 2013]. The maximin value is given by:

$$v^\circ = \min_{p \in P} \max_{s \in S(p)} U(p, s) \tag{2.2}$$

where all variables are identical to those in (Eqn. 2.1). However, the order of operations for minimization and minimization of the utility function is reversed.

For decision scenarios involving multiple, competing stakeholders, as in DRA, the maximin model can be adapted as follows to determine the maximin value for an individual stakeholder ($\underline{v_i}$), given by:

$$\underline{v_i} = \max_{a_i} \min_{a_{-i}} U_i\,(a_i, a_{-i}) \tag{2.3}$$

where i is the index of stakeholder of interest, $-i$ denotes all other stakeholders except the stakeholder of interest, a_i is the action taken by stakeholder of interest, a_{-i} denotes the actions taken by all other stakeholders and U_i is the utility function of stakeholder i.

Likewise, the minimax model can be adapted as follows to determine the maximin value for an individual stakeholder ($\overline{v_i}$):

$$\overline{v_i} = \min_{a-_i} \max_{a_i} U_i\,(a_i, a_{-i}) \qquad (2.4)$$

where all variables are identical to those presented in Eqn. 2.3, but the sequence of maximization and minimization of the utility function is reversed.

2.2.1.1.3 MBA Method

MBA utilizes quantitative and qualitative models of the system of interest to analyze system performance and drive autonomous decision-making. In general, three model types are used for MBA implementation [Krupitzer et al., 2014]: (a) *system models* which represent of the system's state or behavior (such as configuration settings or degree of conformance to requirements); (b) *goal models* which represent the system's decision-making processes for self-adjustment; and (c) *environment models* which represent the operating conditions that effect the system of interest, e.g., physical constraints, functional interdependencies, administrative controls). Furthermore, model development and application for MBA can support system diagnostics which assess a system's current state or performance and system prognostics which estimating the system's future state or performance [Poongodai & Bhuvaneswari, 2013].

2.2.1.1.4 LBA Method

LBA is generally accomplished via three methods [Russell & Norvig, 2010]: (a) *supervised learning* in which a machine is supplied with corresponding sets of expected inputs and desired outputs (collectively referred to as training data) until the machine can independently infer or estimate desired outputs with sufficient accuracy; (b) *unsupervised*

learning in which a machine independently seeks to identify logical relationships, statistical relationships, and patterns amongst a set of data elements without the aid of training data; and (c) *reinforcement learning* which involves a machine performing exhaustive trial-and-error testing within its targeted operating environment to generate positive and negative feedback which is used to incrementally optimize system performance.

2.2.1.2 Principles-based Methods for System Automation

A standard set of methods and conventions for principles-based automation has yet to emerge in the literature. However, research targeting adaptive and improvisational automation for complex systems with high emergence is evident [Krupitzer et al., 2015; Ingrand & Ghallab, 2017], referred to as *context-based automation (CBA)* in this research. Research surrounding CBA focuses on six dimensions of uncertainty with respect to system self-adaptation, known as the *5W + 1H* questions [Salehie & Tahvildari, 2009]: (1) When to adapt?; (2) Why is adaption required?; (3) Where does the change need to be implemented?; (4) What kind of change is needed?; (5) Who has to perform the change?; and (6) How will the change be performed? The overarching goal of CBA research is to maximize the system LOA in analyzing and acting on the 5W + 1H questions to satisfy unanticipated needs.

Two general methods for CBA implementation are evident in the literature [Burke et al., 2003, 2013]:

a. *Metaheuristic methods* seek to characterize a given problem based on existing, well-understood archetypes then apply known solutions or heuristics to satisfy requirements and constraints;

b. *Hyper-heuristic methods* seek to decompose a given problem into a set of sub-problems for which known solutions or heuristics exist then apply those heuristics in an effective sequence towards a satisfactory solution.

Both metaheuristic and hyper-heuristic methods rely on the availability of existing solutions or heuristics for problem solving, resulting in failure when existing solutions are unknown or invalid for a given problem [Burke et al., 2003, 2009, 2013].

2.2.1.2.1 CBA Method

Metaheuristic and hyper-heuristic methods for CBA are commonly implemented as a *Boolean satisfiability solver* algorithm which attempts to solve hard structured problems with multiple variables and constraints using Boolean logic [Gomes, Kautz & Selman, 2008]. In the context of Boolean satisfiability solvers, the terms 'solve' and 'solution' refer to the ability to identify one or more solutions that satisfy the requirements and constraints associated with the problem as represented by Boolean logic. A Boolean satisfiability solver can be modeled generically using the tuple $\langle X, B, C \rangle$, where $X = \{x_1, \dots, x_n\}$ is a set of problem or decision variables, $B = \{B_1, \dots, B_n\}$ is a set of sets that bound or specify values which variables can assume, and $C = \{c_1, \dots, c_m\}$ is a set of constraints which satisfy solution objectives [Marshall et al., 2019]. Each constraint is specified as a Boolean expression involving one or more variables. Constraints that are satisfied are assigned the value $\{1\}$, whereas unsatisfied constraints are assigned the value $\{0\}$. Given $\langle X, B, C \rangle$, brute force enumeration or linear programming can be used to identify combinations of variables that partially or completely satisfy solution constraints, for example:

$$\max \sum_{c \in C} z_c \quad s.t. \begin{cases} z_c \in \{0,1\} & \forall c \in C \\ x_i \in B_i & i = 1, \dots, n \end{cases} \tag{2.5}$$

where z_c denotes the value assigned to a constraint $c \in C$, representing satisfaction or lack thereof. If more than one suitable solution is identified, a single solution may be selected at random amongst the tied solutions. Alternatively, the model in Eqn. 2.5 can be extended to further constrain solutions, for example:

$$\max \sum_{c \in C} z_c \quad s.t. \begin{cases} z_c \in \{0,1\} & \forall c \in C \\ x_i \in B_i & i = 1, \dots, n \\ c_k = 1 & \forall k \in K \end{cases} \tag{2.6}$$

where $K = \{k_1, \dots, k_m\}$ is a set of index values for constraints that are preferred, required, or that otherwise bound the desired solution.

2.3 Critical Infrastructure and Key Resources

Critical infrastructure and key resources (CIKR) consist of shared natural and man-made assets that are relied on for public safety, security, and economic stability [Setola et al., 2016]. While varying definitions and specifications of CIKR exist from nation to nation, *critical infrastructure* (CI) commonly includes man-made systems, facilities, and services associated with 9 industrial sectors: (1) agriculture; (2) commerce; (3) communication; (4) energy; (5) government; (6) security/emergency services; (7) public health; (8) transportation; and (9) water and sanitation (Table 2-6) [Schwab, 2016; Setola et al. 2016; Hemme, 2015]. *Key resources* (KR) include a nation's natural resources, geographic features/territory, major industries and special assets essential to operation of the government or economy (Table 2-7) [DHS 2009, Moteff & Parfomak, 2004]. In recent years, catastrophic failures in the energy, transportation and water/sanitation sectors have highlighted the criticality of CIKR to maintaining public welfare.

Table 2-6: Common Critical Infrastructure Sectors and Examples

CI Sector	Description
Agriculture	Production and distribution of raw and processed foods, e.g., crops, game, livestock, and seafood.
Commerce	Banking and financial systems, e.g., currency transaction, imports/exports, stock exchanges.
Communication	Information and telecommunication systems, e.g., postal systems, telephone/internet utilities, radio frequency transmission systems.
Energy	Production, storage, and transmission of energy (in raw and consumable forms), e.g., petroleum, natural gas, renewable energy, and nuclear power
Government	Administrative and logistical functions required to meet fiduciary obligations/responsibilities (on national, regional or municipal levels), e.g., elections, regulatory systems, diplomatic affairs.
Security/Emergency Services	Law enforcement, military, and emergency response to natural & man-made disasters.
Public Health	Provision and/or regulation of health and welfare services, e.g., housing, medical care, and education.
Transportation	Provision and/or regulation of terrestrial, maritime, and aerial transportation, e.g., ports, roadways, railways, waterways, and airspace.
Water & Sanitation	Production, storage, and distribution of potable water (including wastewater management).

Table 2-7: Common Key Resource Categories and Examples

KR Category	Description/Examples
Natural Resources	Renewable and non-renewable raw materials, e.g., precious metals/minerals, natural gas, oil, wood and water.
Geographic Features & Territory	Naturally occurring or man-made landforms, structures, and bodies of water providing geo-political, military and/or economic benefits.
Major Industries & Special Assets	*Examples:* Tourism/special events, entertainment industry, manufacturing and/or technology sectors (including the defense industrial base, chemical industry, and nuclear industry), national monuments/heritage sites.

2.3.1 ATM System Modernization Efforts

Since 1977 global aviation traffic has doubled every 15 years [ICAO, 2016]. As the demand for finite airspace continues to grow, aviation regulators and the broader aviation community have championed *Free Flight initiatives* which target improved operational capacity, efficiency, and safety through increased system automation [GAO, 1998, 2011]. The ultimate objective of Free Flight initiatives is to enable fully autonomous guidance, navigation and control of aircraft without preplanning or human air traffic controllers for all phases of flight, including: (i) *surface operations* which consist of all aircraft movement of the ground between airport taxiways and runways; (ii) *terminal operations* which include aircraft takeoffs, landings, and sequencing of holding maneuvers within vicinity of an airport; and (iii) *en route operations* which encompass continuous flight path deconfliction between aircraft and rerouting due to weather and/or special use airspace constraints, such as no fly zones. Collision avoidance between the aircraft, static obstacles (such as terrain or man-made structures), and moving obstacles (such as airport ground support vehicles or other aircraft) is a cross-cutting for ATM that applies to all phases of flight.

Despite the demand for autonomous ATM infrastructure, a fully autonomous concept of operations for ATM is infeasible as long as a mixture of aircraft with and without autonomous control capabilities remain in operation. As an incremental step towards the ultimate objective of Free Flight, plans to segregate airspace into Free Flight Operation Areas for autonomous-capable aircraft and Managed Operation Areas for manually piloted aircraft are evident in ATM modernization efforts worldwide (Fig. 2-4) [FAA/EUROCONTROL, 2006]. Free Flight Operation Areas would enable fully

autonomous aircraft operation during the en route phase of flight within specifically designated areas of airspace. Managed Operation Areas would provide command and control for manually piloted aircraft via a human in the loop for all phases of flight, while only supporting surface operations and terminal operations for aircraft capable of autonomous operation. A buffer zone between operation areas would facilitate transitions of autonomous aircraft between the Free Flight and managed operations areas, coinciding with the transitions between terminal operations and en route operations.

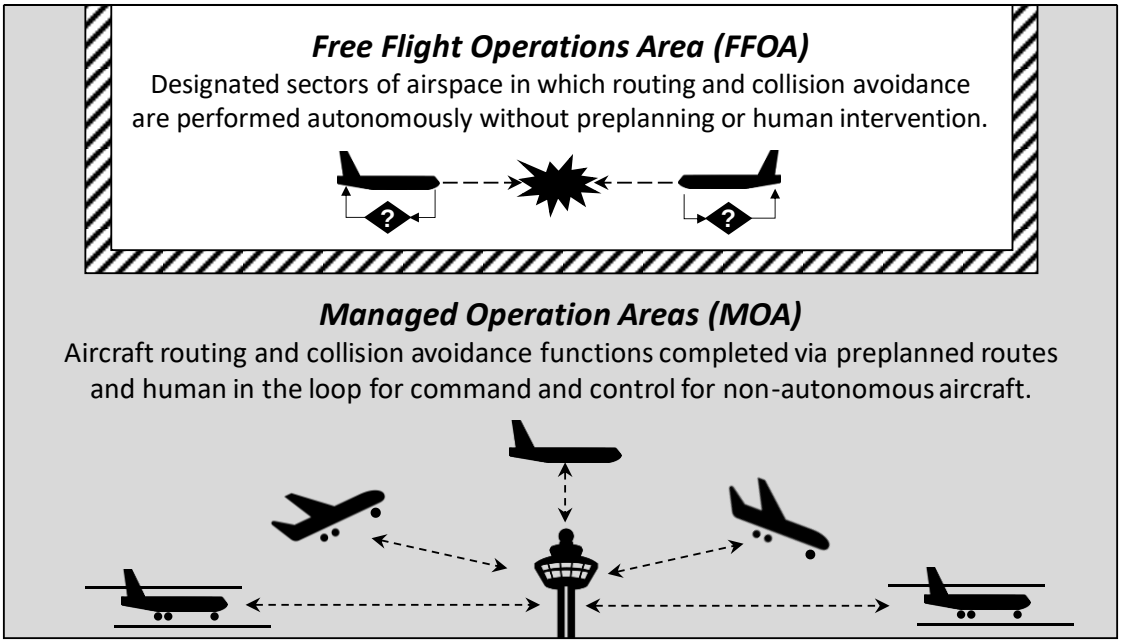


Figure 2-4: Free Flight Concept of Operations for ATM Moderation

In the United States alone, over 540,000 unmanned and autonomous-capable aircraft are estimated to be in service by the year 2020, compared to an estimated 7,200 manually piloted aircraft for passenger and cargo transport in the same year [FAA, 2016]. Consequently, the safe integration of unmanned and autonomous-capable aircraft with manually piloted aircraft is a preeminent challenge for the aviation community [FAA,

2012]. Additional background on Free Flight initiatives for ATM system modernization in the U.S. is presented in Appendix B.

2.4 Related Studies

There is a considerable amount of related research surrounding the resilience of safety-critical and security-critical systems. However, most studies were limited in scope to disruption avoidance or reliance on human intervention as a contingency to unavoidable system disruptions. Studies by Krozel & Peters [1997a, 1997b], Resmerita & Hyemann [2003], and Ouedraogo, Enjalbert & Vanderhaegen [2013] were the most relevant to this research. The remainder of this section reviews the key objectives and limitations of these studies.

2.4.1 Krozel & Peters

Research by Krozel & Peters [1997a,1997b] provided several critical insights and advancements for automated conflict detection and avoidance with application to Air Traffic Management:

a. a procedure for iterative identification and prioritization of system disruption risks based on safety criticality and time constraints for system response;

b. a decision support tool that provided guaranteed recommendations for conflict resolution;

c. demonstrated integration of rule-based and utility-based automation techniques.

Krozel & Peter's methods were adapted for DRA in the DREAM. However, the following issues and limitations of these methods were identified:

a. a human in the loop was required to down-select and execute recommended decisions;

b. precise anticipatory knowledge of conflicts was required for disruption avoidance and/or mitigation measures;

c. cooperation between all system stakeholders is required for disruption avoidance or mitigation measures;

d. extensibility of the conflict detection method was limited to system's concerned with spatial constraints and other types of physical interference in the operating environment.

The DREAM was architected, designed and tested to overcome these functional and non-functional limitations.

2.4.2 Resmerita & Heymann

Resmerita & Heymann [2003] detailed a methodology for conflict resolution of "mutually exclusive" resources which cannot be used by more than one system simultaneously. Similar to Krozel & Peter's approach [1997a], the methodology consists of iterative processes for risk identification, prioritization, and the analysis of alternatives for conflict resolution. However, the method relies on preplanned contingencies and functional redundancies implemented before system deployment for disruption mitigation, specifically:

a. a decentralized decision-making process which mitigates a single-point of failure;

b. a non-cooperative strategy to mitigate functional interdependencies that can result in livelock and deadlock scenarios;

c. an automated procedure for down-selection and execution of system response to emergent disruptions which replaces a human in the loop.

The DREAM incorporates elements of Resmerita & Heymann's methods to implement the functional redundancy and layered defense principles for system resilience. Key shortcomings of this approach include:

a. susceptibility to resource starvation which can deteriorate stakeholder equity;

b. precise forecasts of resource demand and conflicts are required for disruption avoidance;

c. no mechanism for system adaption or recovery from unanticipated disruptions is provided.

Countermeasures for each these shortcomings were developed and incorporated in the DREAM.

2.4.3 Ouedraogo, Enjalbert & Vanderhaegen

A functional architecture for adapting system behavior based on learning from prior successes and failures was proposed by Ouedraogo, Enjalbert & Vanderhaegen [2013]. The architecture integrates an iterative performance feedback process with a "knowledge database" that enables identification of potentially erroneous knowledge, refinement of existing knowledge, and capturing new knowledge over time. The major benefits and contributions of this research included:

a. a set of metrics for assessing system resilience on a diagnostic and prognostic basis;

b. a recursive trial-and-error process to fill knowledge gaps or validate assumptions;

c. demonstrated implementation of a decision support tool for system adaptation in emergent environments and conditions.

The following constraints and limitations were identified:

a. disruption avoidance and mitigation measures are limited to disruptions the system has previously encountered and survived.

b. system disruptions are assessed as independent events without consideration of combined effects that can result in unique implications;

c. extensibility is limited to systems and applications that include a human in the loop.

The DREAM incorporates Ouedraogo, Enjalbert & Vanderhaegen's concept of integrating performance diagnostics and prognostics for system adaptation. However, formulation of alternative performance metrics and evaluation procedures was required to avoid the limitations of the proposed approach.

2.5 The Theory of Constraints

The Theory of Constraints (TOC) asserts that the performance of a system is limited by one or more *constraints* defined as "anything that limits or prevents a system from achieving higher performance relative to its purpose" [ReVelle, 2002]. The TOC specifies two general types of constraints [Goldratt & Cox, 1984]:

a. *Physical Constraints:* Resources that physically limit the system from meeting its goals;

b. *Policy Constraints:* "Rules and measures that inhibit the system's ability to continue to improve."

To resolve or improve physical and policy constraints, the TOC prescribes the *Process of On-Going Improvement (POOGI)* consisting of seven major steps [Ronen & Spector, 1992; Goldratt & Cox, 1984]:

1. *Define the System's Goal;*

2. *Determine the Global Performance Measures:* Identify which metrics and outcomes will be used to assess system's ability to meet defined goals and objectives:

3. *Identify the System Constraints*;

4. *Exploit the System Constraints*: identify and seize opportunities to transform the constraint into an advantage;

5. *Subordinate Everything the Constraints*: change existing rules and practices that conflict with exploitation the constraint;

6. *Elevate the Constraint*: introduce new rules, resources, and processes to supplement and expand on the gains from constraint exploitation and subordination;

7. *Prevent Inertia from Becoming the Constraint*: reassess and reprioritize constraints to enable continuous improvement of the system (i.e. return to step 1 and repeat the process).

The TOC asserts that a limited subset of the system's constraints, with as few as one, need to be overcome to achieve system goals and objectives [ReVelle, 2001]. The purpose of the POOGI is to prioritize and focus on constraints that have the most impact on system goals.

2.5.1 Evaporating Cloud Technique

The *Evaporating Cloud (EvC)* technique [Goldratt, 1990] is a graphical ontology for representing and reasoning about conflicts encountered within the POOGI. Within the TOC, conflicts represent scenarios that require a tradeoff between competing requirements based on actual or perceived constraints of the system. The EvC technique supports conflict resolution in two stages. First, actual or perceived conflicts are modeled by developing an EvC diagram (Figure 2-5) consisting of the following elements:

a. *Common Objective:* a desired or intended outcome of the system impacted by the conflict;
b. *Requirements:* a set of requirements or goals that support the common objective but may potentially be in conflict;
c. *Prerequisites:* conditions or constraints driven by system requirements that directly conflict with each other;
d. *Connectors:* relationships between elements in the EvC diagram are established using one of two connector types. Straight arrows map the association between the prerequisites, system requirements and the common objective. Jagged arrows

are used to indicate a conflict exists between a set of or requirements or prerequisites.

The second stage of the EvC technique requires the development of one or more *injections* which are actions that can eliminate or mitigate conflicts between system requirements or prerequisites. Consequently, the EvC technique “does not strive to reach a compromise solution, rather it concentrates on invalidating the problem itself" [Goldratt, 1990]. However, injections can also result in conceptualization of partial solutions which improve rather than resolve conflicts.

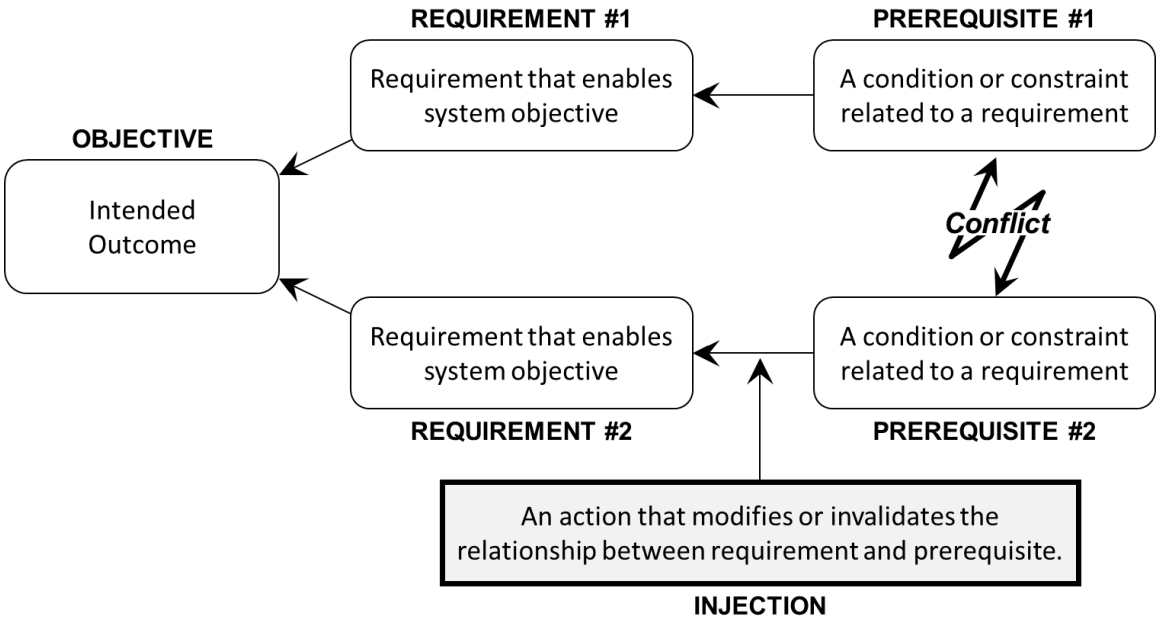


Figure 2-5: Generic EvC Diagram

As informed by the findings of the literature review, the TOC and EvC technique were applied to: (1) identify and prioritize a subset of system resilience constraints to assess; (2) model underlying physical conflicts and policy conflicts associated with prioritized system constraints; and (3) conceptualize countermeasures to underlying conflicts to improve or resolve system constraints.

2.6 Assessment of System Resilience Constraints

2.6.1 Forecasting Constraints in Emergent Environments

Emergent conditions in man-made and naturally occurring operating environments represent an inherent risk for the disruption of engineered systems. Lacking the ability to anticipate and avoid all potential disruptions, disruption prevention and mitigation are both required to satisfy system resilience objectives. Despite being complementary needs with respect to resilience, disruption prevention and mitigation can conflict in emergent environments with respect to underlying assumptions and constraints (Figure 2-6). Avoidance is an optimization-seeking strategy that presumes disruptions can be accurately predicted and prevented. In contrast, mitigation accommodates the potential for sub-optimal system performance due a presumption of unforeseen or unavoidable disruptions. Consequently, the prioritization of prevention and mitigation actions in emergent environments can be ambiguous, representing a policy constraint in the TOC. For example, should an avoidance or mitigation action be taken for a given disruption when outcomes are uncertain? Similarly, should system recovery from a past disruption or the avoidance of a future disruption be prioritized? Given the unique objectives and constraints of the system of interest, the prioritization of disruption avoidance and mitigation may be a conditional rather than an absolute matter.

To address the forecasting constraint in emergent environments, application of the *satisficing principle* was proposed as an injection to the EvC. The satisficing principle prioritizes the adequate and sufficient fulfillment of system requirements overachieving ideal or optimal outcomes [Simon, 1955]. Notably, satisficing can mitigate inadvertent errors and anomalous behavior that can result from pursuing optimal outcomes under

uncertainty. For example, a quality paradox can be observed in which sub-optimal global behavior is achieved despite the application of locally optimal decisions. Similarly, an efficiency paradox can occur in which the resources applied for decision making or problem resolution exceed the benefits gained.

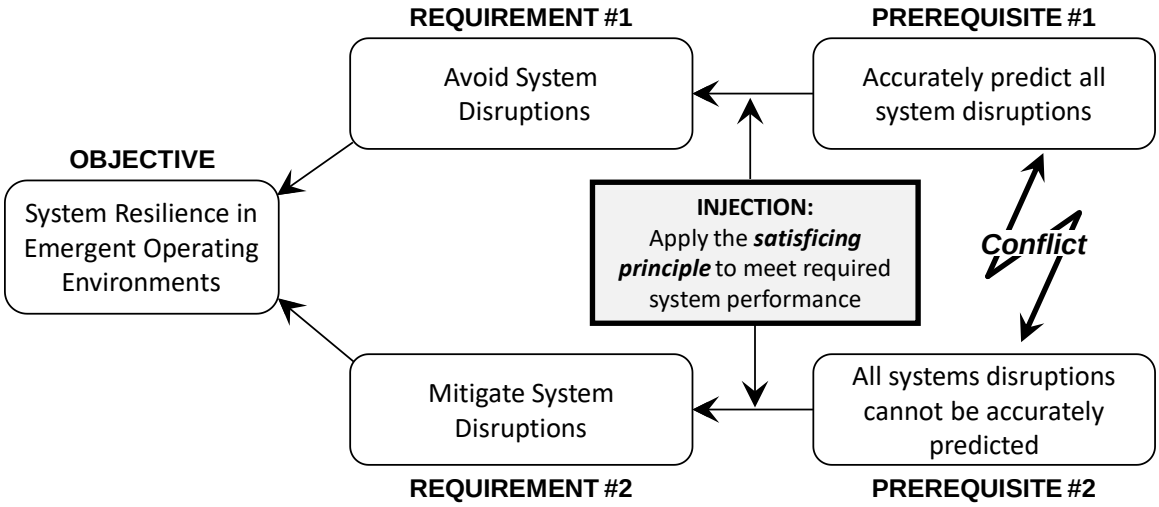


Figure 2-6: EvC for Forecasting Constraints

2.6.2 Resource Deconfliction Constraints

Availability conflicts over shared resources that are needed to fulfill system requirements are a prevalent source of disruptions (as discussed in Chapter 2, Section 2.1.1.2). Lacking the ability to increase the supply or limit demand for a given resource, deconfliction are largely constrained to two priorities [de Weck, Roos & Magee, 2016]:

a. *Efficiency optimization:* avoiding or minimizing the waste of resources;
b. *Stakeholder equity:* fairness with respect to resource utilization amongst stakeholders;

While efficiency optimization and stakeholder equity are complementary to the objective of resource deconfliction, these aims often motivate competitive or collaborative strategies that can conflict (Figure 2-7).

To alleviate deconfliction constraints involving shared resources, application of a *coopetition strategy* was proposed as an EvC injection. Coopetition involves the exploitation of collaborative behavior between competitors when mutually beneficial outcomes can be achieved or if the performance impact does not adversely affect objectives [Gianetto & Heydari, 2015; Dagnino, 2009]. In contrast, the application of strictly competitive or collaborative strategies can unnecessarily result in sub-optimal or detrimental outcomes given a lack of flexibility.

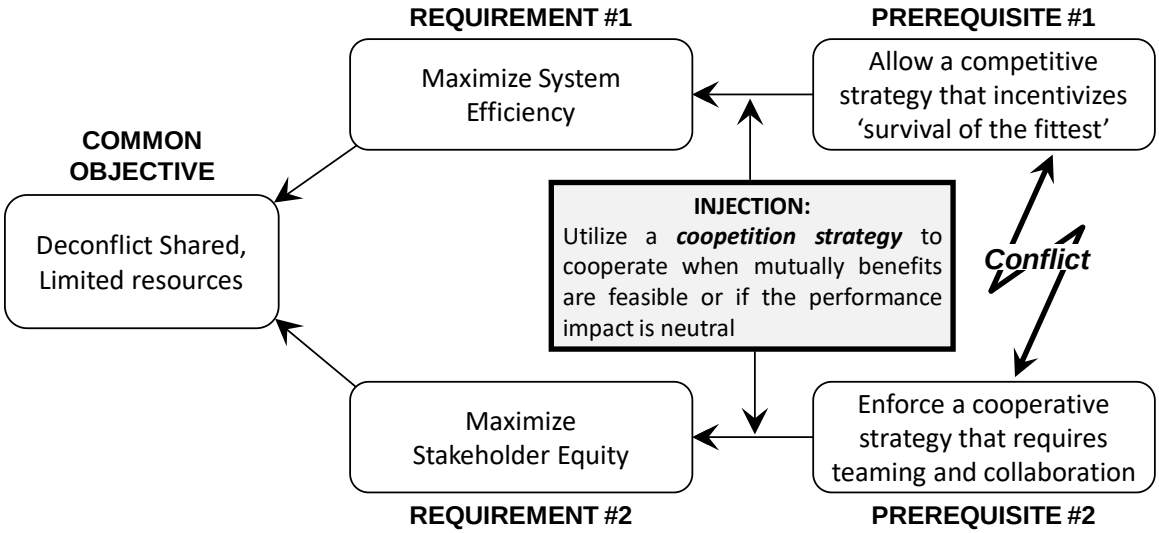


Figure 2-7: EvC for Resource Deconfliction Constraints

2.6.3 Manpower Constraints

Human intervention is often relied on as an operational contingency to engineered systems that fail or provide unsatisfactory performance. Resilience design principles also

incorporate maintaining a human in the loop of engineered systems to enable flexibility in response to emergent disruptions [Jackson & Ferris, 2013]. However, human error and susceptibility to disruptions can limit or preclude human intervention in many cases (as discussed in Section 2.1.1.1). Consequently, reducing dependence on human intervention also supports operational contingency when manpower constraints arise. The benefits and limitations of human intervention as an operational contingency are represented as a conflict in Figure 2-8.

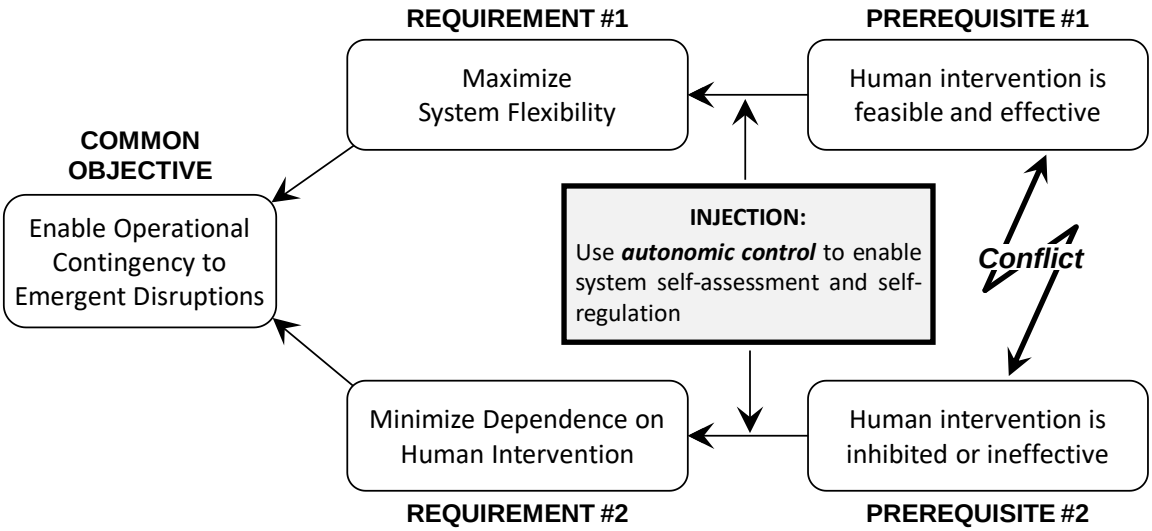


Figure 2-8: EvC for Manpower Constraints

To address the manpower conflict for enabling operational contingency, the use of *autonomic control* was proposed as an injection to the EvC. Autonomic control is system automation technique which integrates of two overarching capabilities [Kephart & Chess, 2003; Lalanda et al., 2014]:

- *System self-assessment:* the ability of a system to monitor and evaluate its own performance with respect to mission requirements;

- *System self-regulation:* the ability of a system to adapt its own performance within the scope of overarching mission priorities and constraints. That is, the ability to execute preventive and/or corrective actions to maintain satisfactory performance without human-involvement.

Autonomic control processes are enabled by system objectives and requirements which are specified for the system without prescribing exact procedures or instructions for their fulfillment. The system is given full authority and flexibility to adapt its own behavior in order to meet or exceed intended outcomes based on performance feedback.

2.7 Literature Search Summary

The literature search identified the following key findings and implications which were used to align research objectives with gaps in the body of knowledge surrounding resilient engineered systems:

1. The motivation for system resilience is the mitigation of anticipated and emergent system disruptions of all degrees of severity which encompasses any unacceptable degradation in performance or outright failure.

2. Resilience engineering represents a paradigm-shift in risk management for engineered systems which treats system disruptions as inevitable rather than potential outcomes. Likewise, the scope of resilience engineering extends beyond the mitigation of anticipated threats and vulnerabilities to emergent risks encountered across the system lifecycle.

3. The existing body of knowledge surrounding system resilience is narrowly focused on disruption avoidance, resistance and recovery with the following themes observed:

    a. Applying lessons learned and best practices observed from historical successes and failures of engineered systems to prevent future failures or performance degradation;

    b. Continuous monitoring of deviations between actual and target performance for early detection and response to potentially adverse trends based on diagnostic assessments of system performance;

    c. Domain specific resilience metrics, strategies and tools that lack extensibility.

4. The existing body of knowledge surrounding system resilience lacks generalizable strategies and techniques for mitigating emergent disruptions that are unknown until encountered during mission execution, chiefly:

    a. Reliance on human intervention as a contingency for inoperable or ineffective systems is recognized as a best-practice, however, increasing system complexity and human factors can limit or invalidate a human in the loop for system command and control;

    b. System automation has been successfully adopted for disruption avoidance and performance optimization, but existing implementation methods are largely rule-based and lack the flexibility to adapt to emergent issues and risks;

c. Existing resilience metrics and assessment methodologies are narrowly focused on diagnostics which are lagging performance indicators that largely support reactive rather than proactive measures for disruption mitigation.

With respect to relevant application areas for this research, safety-critical systems and applications were abundant in the resilience literature. In keeping with that theme, the growing demand for increasingly scarce natural and man-made resources relied on for public safety, security and economic stability was identified as a problem of historical and current significance in the resilience engineering domain. Based on this pragmatic motivation and the identified gaps in the literature, an opportunity to contribute to the body of knowledge surrounding the resilience of autonomous systems operating in emergent operating environments was apparent.

Chapter 3: Methodology

Research was conducted using the experimental method and organized in four progressive phases (Figure 3-1):

1. *Problem Definition and Scoping* consisted of three tasks: (1) establish contextual background surrounding the research problem; (2) prioritize research goals and objectives; and (3) identify a system of interest for research application. This phase of research was accomplished by performing a literature review (presented in Chapter 2);

2. *Problem Analysis* focused on the identification of limitations and gaps in the existing risk management and resilience engineering methodology. Subsequently, the Theory of Constraints was used to identify and reason about cause-and-effect relationships surrounding emergent system disruptions (as presented in Chapter 2, Section 2.5).

3. *Concept Formulation and Implementation* involved development of the DREAM as an experimental solution to mitigate emergent system disruptions based on proposed countermeasures established via the Evaporating Cloud Technique (detailed in Chapter 2, Section 2.6). The methods applied and developed in this research to implement proposed countermeasures are presented in Section 3.1.

4. *Test & Evaluation* applied the DREAM to the Free Flight ATM problem using modeling and simulation. The DREAM was compared to the performance of existing standards and conventions for autonomous ATM. Statistical analyses of results based on paired t-testing were then performed on results to evaluate

research hypotheses. Modeling and simulation methods used to execute research experiments are specified in Section 3.2, followed by the presentation of the statistical analyses methods used to evaluate experiment results in Section 3.3. All results and analyses associated with the test and evaluation phase of this research are presented in Chapter 4.

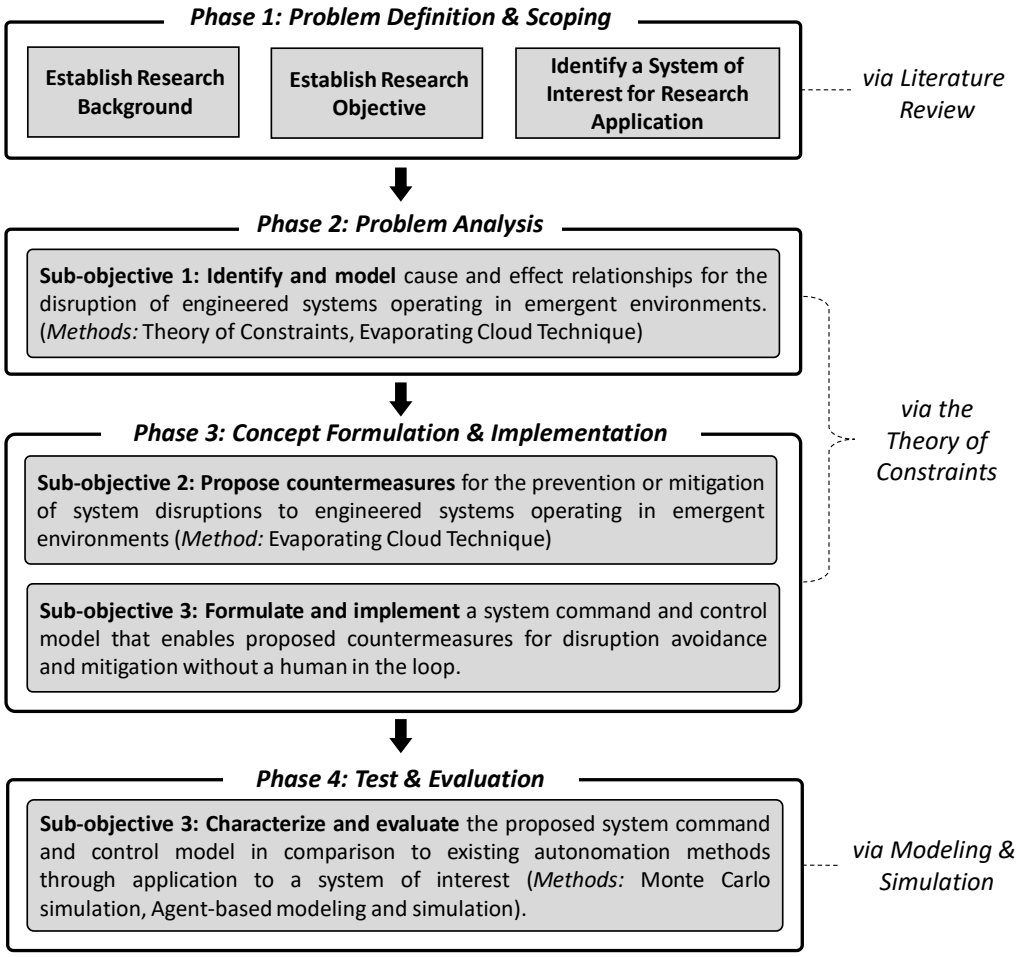


Figure 3-1: Research Framework

The remainder of this chapter details the methods used to implement and evaluate the performance of the DREAM based on application to the Free Flight ATM problem, corresponding with research sub-objectives 3 and 4 (as specified in Chapter 1, Section 1.3).

3.1 Implementation of the DREAM

The DREAM integrates satisficing, coopetition and autonomic control functionality into a system C2 architecture (Figure 3-2) based on constraint resolution concepts developed via the TOC and EvC technique (presented in Chapter 2). The implementation of the model components and processes is presented in the following sections.

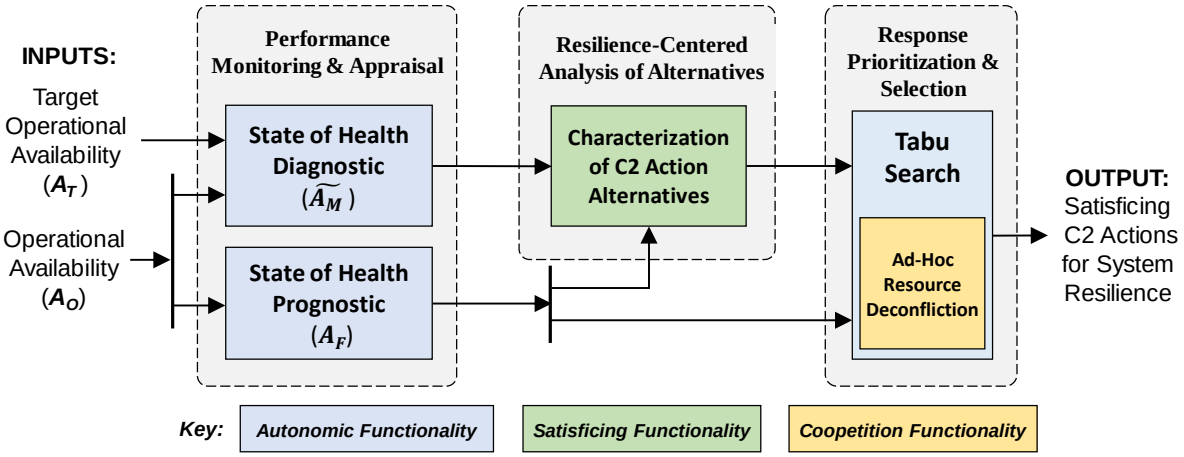


Figure 3-2: DREAM Inputs, Outputs and Processes

3.1.1 Performance Monitoring and Appraisal Process

The performance monitoring and appraisal process enables the autonomic control function of system self-assessment. Continuous evaluation of the system's *state of health (SoH),* which characterizes system performance relative to its required or intended performance, is executed on a diagnostic and prognostic basis as follows.

For the diagnostic assessment of SoH, *operational availability* (A_O) is used to determine proportion of total operating time that the system has functioned at or above an acceptable level of performance based on actual usage [Blanchard, 2004]. Operational availability is given by

$$A_O = \frac{Uptime}{Uptime + Downtime} \tag{3.1}$$

where $Uptime$ is the cumulative time that the system is operable and functioning at or above an acceptable level of performance and $Downtime$ is the total length of time the system provides unacceptable performance or is inoperable. Applying the *principle of factor sparsity* as an analytical heuristic, i.e., the observation that roughly 80% of effects are generated by approximately 20% of causes in many natural and man-made phenomenon [Newman, 2005], the *moving median of operational availability* ($\widetilde{A_M}$) over the last 20% of the system's operating time is used to monitor the trend in historical performance. The moving median of operational availability is given by

$$\widetilde{A_M} = \frac{\min(A_{80}, A_{90}, A_{100}) \; + \; \max(A_{80}, A_{90}, A_{100})}{2} \tag{3.2}$$

where A_{80}, A_{90}, A_{100} represent the operational availability achieved at 80%, 90%, and 100% of the system's current operating time, respectively, as calculated using Eqn. 3.1. The procedure developed in this research for diagnostic SoH evaluation is presented in Algorithm 1, below, based on A_O, $\widetilde{A_M}$, and a *target availability level* (A_T) which is a value specified for the system representing the ideal or desired system availability. The procedure outputs one of four *SoH Levels* that characterize the system's demonstrated ability to meet the target availability level:

- *SoH Level 1*: the targeted operational availability is being met or exceeded;
- *SoH Level 2*: infrequent periods of downtime are being experienced;
- *SoH Level 3*: frequent or extended periods of downtime are being experienced;
- *SoH Level 4*: mission failure and/or system failure is expected without remediation.

Algorithm 1 – Procedure for System SoH Diagnostic

input: operational availability (A_O); moving median of operational availability ($\widetilde{A_M}$); target availability level (A_T)

output: *system SoH level*

procedure:

1: **if** $\widetilde{A_M} \geq A_T$ **AND** $A_O \geq A_T$ **then** $SoH\ Level = 1$
2: **else if** $\widetilde{A_M} \geq A_T$ **AND** $A_O < A_T$ **then** $SoH\ Level = 2$
3: **else if** $\widetilde{A_M} < A_T < A_O$ **then** $SoH\ Level = 3$
4: **else** $SoH\ Level = 4$ ▷ *Note:* $\widetilde{A_M} < A_T$ and $A_O < A_T$
5: **end if**
6: **return** ($SoH\ Level$)

As a prognostic measure of system SoH, an adaptation of the operational availability metric presented in Eqn. 3.1 was developed to provide an estimate of the system's future operational availability given the execution of a specific action (or inaction) at the present, termed *forecasted availability* (A_F). Forecasted availability is given by

$$A_F = \frac{Uptime^*}{Uptime^* + Downtime^*} \qquad (3.3)$$

where $Uptime^*$ and $Downtime^*$ are respectively the estimated uptime and estimated downtime associated with a future C2 action or inaction. Algorithm 2, below, presents the procedure developed in this research to characterize the expected trend in system SoH based on A_F and A_O, resulting in one of four *Performance Impact Factor (PIF)* levels:

- *PIF Level 1*: operational availability is expected to improve if the specified action is executed.
- *PIF Level 2*: operational availability is not expected to increase or decrease after executing the specified action;
- *PIF Level 3*: operational availability is expected to decrease if the specified action is executed;

- *PIF Level 4*: estimated operational availability is unavailable or unknown for the specified action.

Algorithm 2 – Procedure for System SoH Prognostic

input: operational availability (A_O); forecasted availability (A_F)
output: *PIF Level*
procedure:
1: **if** $Uptime^*$ **OR** $Downtime^*$ is undefined **then** $PIF\ Level = 4$
2: **else if** $A_F < A_O$ **then** $PIF\ Level = 3$
3: **else if** $A_F = A_O$ **then** $PIF\ Level = 2$
4: **else** $PIF\ Level = 1$ ▷ *Note:* $A_F > A_O$
5: **end if**
6: **return** ($PIF\ Level$)

3.1.2 Resilience-Centered Analysis of Alternatives

The resilience-centered analysis of alternatives supports satisficing principles in the DREAM by providing a schema for classifying decision alternatives based on the concurrent objectives of disruption avoidance and mitigation. Given the uncertainty of outcomes in emergent operating environments, the resilience-centered analysis of alternatives was developed as part of this research to provide a heuristic for assessing expected impacts on system resilience based on system SoH and PIF levels (Algorithm 3). The procedure results in one of five *Resilience Impact Factor (RIF)* levels which designate the anticipated trend in system resilience if the associated prevention or mitigation action is executed:

- *RIF Level 1*: the specified action is expected to sustain and/or improve system resilience;
- *RIF Level 2*: temporary degradation of system resilience is expected with rapid recovery from disruptions;

- *RIF Level 3:* an extended period of degraded system resilience is expected with slow recovery from disruptions;
- *RIF Level 4:* permanent degradation of system resilience is expected;
- *RIF Level 5:* permanent system disruption is expected if the specified action is executed.

Algorithm 3 – Procedure for RIF Encoding

input: *SoH Level*; *PIF Level* for an available C2 action
output: *RIF Level* for an available C2 action
procedure:

if $SoH\ Level = 1$ **then**
  if $PIF\ Level = 1$ **OR** $PIF\ Level = 2$ **then** $RIF\ Level = 1$
  else if $PIF\ Level = 3$ **then** $RIF\ Level = 2$
  else $RIF\ Level = 3$
  end if
else if $SoH\ Level = 2$ **then**
  if $PIF\ Level = 1$ **OR** $PIF\ Level = 2$ **then** $RIF\ Level = 2$
  else $RIF\ Level = 3$
  end if
else if $SoH\ Level = 3$ **then**
  if $PIF\ Level = 1$ **OR** $PIF\ Level = 2$ **then** $RIF\ Level = 3$
  else $RIF\ Level = 4$
  end if
else ▷ $Note: SoH\ Level = 4$
  if $PIF\ Level = 1$ **then** $RIF\ Level = 3$
  else if $PIF\ Level = 3$ **then** $RIF\ Level = 4$
  else $RIF\ Level = 5$
  end if
end if
return ($RIF\ Level$)

3.1.3 Response Prioritization and Selection Process

The response prioritization and selection process enables the autonomic control function of system self-adaptation in the DREAM. Via the self-adaptation process, the coopetition principle is supported by using performance feedback to regulate the competitive or collaborative behavior for the deconfliction of shared resources required for mission execution.

Following the categorization of available action alternatives by RIF level, down-selection is performed using a *tabu search* process which integrates a set of prioritized performance goals and constraints with performance feedback for decision making [Glover, 1985]. As informed and implemented within this research, two constraints were established and enforced during the down-selection of alternatives, referred to as *tabus*:

- *Tabu 1 (Deficiency Mitigation):* Actions categorized as RIF level 5 are prohibited from selection to mitigate the sustained decline of system resilience capabilities;
- *Tabu 2 (Course Filtering):* Actions categorized with a RIF level higher than the minimum RIF level available in the option-space are prohibited from selection to narrow the solution-space while mitigating premature optimization in the decision process.

Subsequently, with tabus enforced, four rules for down-selection are sequentially applied to the unprohibited actions based on their RIF level and forecasted availability:

1. *Failsafe Triggering Rule:* If all C2 actions in the solution-space are prohibited, then initiate a failsafe action to mitigate an imminent disruption of the system,

e.g., alerting critical system stakeholders or mobilizing an emergency response team.

2. *Remediation Triggering Rule:* If RIF level 3 is the minimum available in the solution-space, then randomly select and execute any C2 action with RIF level 3;
3. *Optimization Rule:* Select the unprohibited C2 action with the lowest RIF level and highest forecasted availability;
4. *Tie-Breaking Rule:* In execution of the optimization rule, if multiple C2 actions exist with an equivalent RIF level and forecasted availability, randomly select a C2 action.

The logic of the response prioritization and selection process enables the identification and selection of C2 actions that satisfice emergent system constraints and resilience objectives via automation.

3.1.4 Summary of DREAM Benefits

The DREAM mitigates several limitations of existing automation approaches with respect to system resilience:

a. System automation does not rely on specific, pre-planned actions or reactions, unlike RBA that defines and enforces explicit rules for system behavior without the ability to deviate;
b. Satisficing and optimization-seeking behaviors are supported based on system SoH, contrasting with the persistent application of optimization-seeking processes in UBA that can inadvertently degrade overall performance quality and/or system efficiency over the long-term.

c. C2 actions are driven by system SoH and operational constraints at decision-time, as opposed to LBA which is driven by previously observed patterns of success or failure that may be invalid for current needs;
d. System availability is used as a performance diagnostic and prognostic which is tailorable to the system of interest, unlike hyper-heuristic and metaheuristic methods for CBA which rely on a library of metrics based on existing problem archetypes and solutions.

In order to demonstrate and validate the benefits of the DREAM in comparison to RBA and UBA methods, this research applied the model to autonomous aircraft C2 for Free Flight ATM.

3.2 ATM Case Study

To apply the DREAM and evaluate research hypotheses, a case study on Free Flight ATM was performed. The case study used modeling and simulation to compare the performance of the DREAM with existing RBA and UBA methods for autonomous navigation and collision avoidance. The ATM case study employs the *Highly Accelerated Stress Screening (HASS)* method which intentionally exposes a system of interest to operating conditions and environments beyond anticipated requirements to characterize its performance [McLean, 2009]. The remainder of this chapter details the HASS test scenario, measures of effectiveness utilized, modeling and simulation methods employed and experimental setup for evaluating research hypotheses.

3.2.1 ATM System HASS

HASS experiments were based on a Free Flight ATM test scenario introduced by Krozel et al. [2001]. In the test scenario, two or more autonomous aircraft must

simultaneously navigate between randomly assigned starting waypoints and ending waypoints within a Free Flight Operation Area (FFOA) without exchanging information for flight-path deconfliction or collision-avoidance. The objective for each aircraft is to fly the most direct route between their respective starting and ending waypoints without disrupting other aircraft in the FFOA. Each aircraft is equipped with sensors that enable detection of other aircraft within a limited range, including their current location, direction of travel and relative speed. All aircraft in the FFOA are restricted to fly at an equivalent altitude and equivalent speed. To remain in the FFOA, only left and right turns can be executed for navigation and collision avoidance. Based on these objectives and constraints, ATM system efficacy and efficiency via autonomous C2 is evaluated based on the measures of effectiveness specified in Section 3.2.2.

3.2.2 ATM System Performance Measures

The ATM case study used five measures of effectiveness (MOEs) to evaluate and compare the performance of the DREAM with existing RBA and UBA methods for autonomous navigation: (1) incident rate; (2) mission reliability; (3) system efficiency; (4) the standard deviation of efficiency; and (5) system stability. In the following sections, each MOE is defined and discussed in the context of CIKR management and the ATM application.

3.2.2.1 Incident Rate

Incident rate (λ) measures the frequency of system disruptions resulting in degraded performance or system failure. The incident rate metric was adapted from the *failure rate* metric used in reliability analysis [DoD, 2005] which narrowly focuses on system failures that result in the system becoming inoperable. In the context of system resilience, the

incident rate is an indicator of the efficacy of disruption avoidance. Consequently, the minimization of the incident rate is desirable as it demonstrates the ideal behavior of circumventing system disruptions altogether. λ is given by

$$\lambda = \frac{Total\ Incidents}{Total\ Operating\ Time} \tag{3.4}$$

where $Total\ Incidents$ is a count of unique system disruptions and $Total\ Operating\ Time$ is the elapsed duration of time over which system disruptions were observed. All system disruptions considered in the calculation of the incident rate within the ATM case study are defined in Section 3.2.3.

3.2.2.2 Mission Reliability

Mission reliability (R_M) measures the likelihood of a system performing its required functions throughout the duration of a specified mission profile [DoD, 2005]. R_M is an indicator of the system's ability to support intended outcomes as opposed to a measure of the system's ability to provide required functional capabilities over time, i.e. *system reliability*, as discussed in Section 2.1.2. R_M is given by

$$R_M = A_O \times MCR \tag{3.5}$$

where A_O is given in Eqn. 3.4 and the *Mission Capability Rate (MCR)* is the proportion of missions completed to the missions attempted by the system [DoD, 2005]. *MCR* is given by

$$MCR = \frac{Missions\ Completed}{Missions\ Attempted} \tag{3.5}$$

In the ideal case $R_M = 1$, representing a system that is never disrupted and always achieves its intended outcomes. In contrast, an $R_M = 0$ represents a system that is either

persistently unavailable or incapable of completing its intended mission, irrespective of A_O achieved.

3.2.2.3 System Efficiency

System Efficiency (E) is a measure of the degree to which a system achieves an ideal or expected level of performance. In practical terms, system efficiency indicates the degree of waste or losses in a system with respect to the resources consumed by or expended to operate the system, such as time, money, or materials. Within the context of CIKR, the minimization of waste is desired to increase resource longevity and accessibility for potential beneficiaries. For the autonomous ATM application, system efficiency represents the degree to which aircraft are able to fly a direct path between a selected point of origin and a selected destination, respectively termed a starting waypoint and ending waypoint [Archibald et al., 2008]. The individual efficiency of a single aircraft (E_i) is given by

$$E_i = \frac{t_i}{t_{d_i} + t_i} \tag{3.6}$$

where i is the index of the aircraft, t_i is the ideal flight time via a straight-line flight path without deviations, and t_d is the delay time that exceeds the ideal flight time, if any. The *system efficiency*, which takes into account all aircraft, is given by

$$E = \frac{1}{N}\sum_{i=1}^{N} E_i \tag{3.7}$$

where N is the total number of aircraft in the designated ATM system. Under ideal circumstances, all aircraft can fly direct routes without deviations between their starting and ending waypoints, resulting in 100% efficiency ($E = 1$). In practicality, disruptions

such as inclement weather and routing conflicts require aircraft to deviate from a direct path, resulting in decreased system efficiency.

3.2.2.4 Standard Deviation of Efficiency

The *standard deviation of efficiency* (D_E) is a measure of the variation in efficiency amongst individual systems competing for shared resources such as an aircraft competing for shared, limited airspace. As a measure of variation, the standard deviation of efficiency is also an indicator of the degree of equity amongst stakeholders [Archibald et al., 2008]. In the context of DRA and CIKR management, minimizing the standard deviation of efficiency for a given resource is not only desirable, but legally mandated and continuously audited for many public safety and security resources, e.g., response times for emergency responders. For the ATM application, the standard deviation of efficiency is given by

$$D_E = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(E_i - E)^2}. \qquad (3.8)$$

Ideally, no variation in efficiency would exist ($D_E = 0$), indicating that all aircraft experience an equivalent level of efficiency with respect to airspace utilization. However, the absence of variation should not be conflated with an acceptable or desirable level of efficiency. For example, a standard deviation of efficiency of zero would apply to all aircraft experiencing no delay time or an equally high amount of delay time. Therefore, minimizing the standard deviation of efficiency while meeting or exceeding required system efficiency levels is the overarching ideal in CIKR management.

3.2.2.5 System Stability

System stability (SS) is a measure of the degree to which a system mitigates error propagation and cascading failures due to the so-called *domino effect* or *butterfly effect*, *i.e.*, when small localized changes in a system result in global/large scale effects due to system dependencies and sensitivities [Archibald et al., 2008; Krozel et al., 2001]. In practical terms, system stability is an indicator of the efficacy of preventive and corrective actions for disruption mitigation. Within the context of DRA and CIKR management, maximization of system stability is desired to mitigate domino effects that can incidentally disrupt the system. System stability is given by

$$SS = \frac{L_1}{L_2} \tag{3.9}$$

where L_1 is the set of all system disruptions that would occur if no preventive or correction actions are taken and L_2 is the set of all system disruptions that occur when preventive or corrective actions are taken. In the calculation of system stability, both L_1 and L_2 are measured with respect to an equivalent set of operating conditions and scenarios. A value of $SS = 1$ indicates that an equal number of disruptions occurs with and without preventive and/or corrective action for system disruptions. In other words, an identical amount of disruptions occurred with and without preventive and/or corrective action. Values of $SS < 1$ indicate that system stability worsens when preventive and corrective actions are executed. For example, $SS = 0.5$ indicates that corrective and preventive actions resulted in twice the number of disruptions than without. Values of $SS > 1$ indicate that system stability is improved with preventive and corrective actions.

For example, $SS = 2.0$ indicates that half the number of disruptions occurred with preventive and corrective action than without.

In the context of the ATM application, system stability is calculated based on the number of disruptions when aircraft fly direct flight paths between waypoints with no collision avoidance maneuvers performed (as L_1) and the number of disruptions observed when collision avoidance maneuvers are performed for airspace deconfliction (as L_2).

3.2.3 Free Flight Modeling & Simulation

Free Flight simulations were based on a combined physical and dynamic model of a FFOA and autonomous aircraft. The FFOA component of the model consists of a two-dimension graph representing a 100 nautical mile by 100 nautical mile sector of airspace with 100 waypoints evenly distributed in a 10 by 10 grid (Figure 3-3). Each waypoint represents a location that an autonomous aircraft can enter or exit the FFOA. The autonomous aircraft component of the model integrates the physical attributes, dynamic attributes, and functional constraints of aircraft operating within the FFOA (Figure 3-4). A total of seven parameters constrain the Free Flight test model:

- *Air Traffic Density* (ρ): the number of aircraft operating within a designated sector of airspace, such as an FFOA;
- *Collision Span* (C): the maximum span of aircraft susceptible to a collision, e.g., aircraft wingspan or fuselage length;
- *Detection Range* (D): the distance at which aircraft can detect other aircraft via airborne sensor equipment;

- *Fuel Reserve* (F): the amount of flight time associated with contingency fuel that is beyond the minimum fuel required for an aircraft to reach its intended destination via the most efficient route;
- *Safe-separation Distance* (S): minimum horizonal clearance required between aircraft to provide safety margin;
- *Aircraft Speed* (V): forward speed required to be maintained by aircraft in the FFOA;
- *Turn Rate* (ω): maximum speed at which aircraft can execute left and right turns.

The collision span, safe separation distance, and detection range parameters correspond with three safety-critical regions surrounding each aircraft:

- *Collision Region:* the physical area surrounding the aircraft that must be free of obstacles to prevent a collision;
- *Safe Separation Region:* an administratively controlled physical area surrounding the aircraft collision region that provides a physical buffer for collision avoidance;
- *Detection Region:* the physical area surrounding the aircraft within which other aircraft can be detected.

Based on parameter settings, the following disruptions to the ATM system are feasible:

- *Loss of Safe Separation (LoSS):* operation of an aircraft at a distance less than the specified safe separation distance from another aircraft;
- *Near Mid Air Collision (NMAC)*: operation of an aircraft at a distance of 500 feet or less from another aircraft.

- *Mid Air Collision (MAC)*: physical contact made by two or more aircraft, i.e., the separation distance between two or more aircraft is less than the collision span;
- *Fuel Shortage (FS)*: an aircraft runs out of fuel before reaching its assigned end waypoint;
- *Exceedance of Envelope (EOE)*: autonomous operation of an aircraft outside the FFOA boundary;
- *Failsafe Maneuver (FSM): a change in aircraft altitude is required for collision avoidance.*

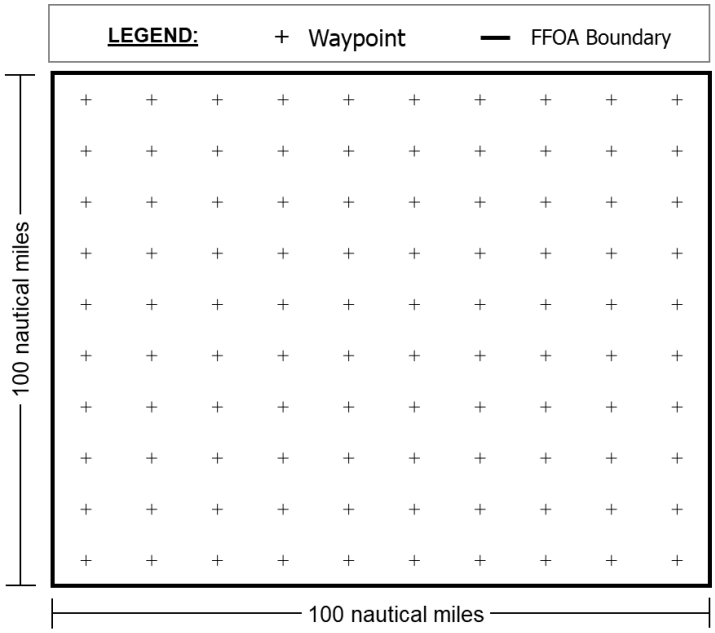


Figure 3-3: FFOA Model

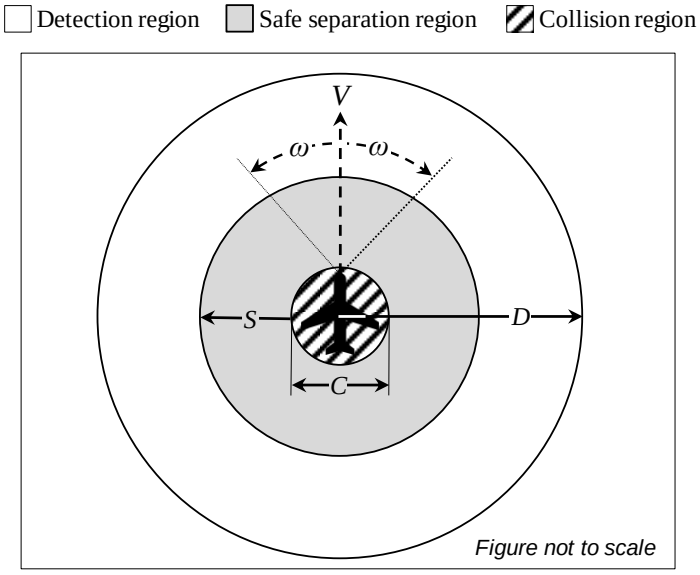


Figure 3-4: Autonomous Aircraft Model

Table 3-1: Free Flight Model Parameters [Adapted from Kochenderfer et al., 2010]

Parameter	Description	Units
ρ	Air Traffic Density; Number of aircraft operating at the same altitude level within the same sector of airspace.	aircraft per 10,000 sq. nautical miles
C	Collision span; Maximum span of aircraft susceptible to a collision (e.g. aircraft wingspan or fuselage length).	feet (ft)
D	Detection Range; Distance at which aircraft can be reliably detected (via airborne radar equipment).	nautical miles (nmi)
F	Fuel Reserve; Equivalent flight time of contingency fuel (beyond the minimum amount required for an aircraft to reach its intended destination via the most efficient route)	minutes (min)
S	Safe-separation Distance; Minimum horizonal clearance required between aircraft (to provide safety margin)	feet (ft)
V	Forward Horizontal Speed required to be maintained by aircraft.	nautical miles per hour (nmi/hr)
ω	Turn Rate; Maximum speed at which aircraft can execute course change maneuvers (i.e. left and right turns).	degrees/sec (°/sec)

3.2.4 Monte Carlo Simulation

Monte Carlo sampling was used to randomly select starting and ending waypoints for each aircraft in the simulated FFOA. Sampling was based on a discrete uniform

distribution, with each waypoint in the FFOA having an equal likelihood of being selected as a starting waypoint (Figure 3-5). Before selection of the ending waypoint for each aircraft, the sample space is censored to remove the selected starting waypoint for the aircraft, with all remaining waypoints in the FFOA (99 total after censoring) having an equal likelihood of selection.

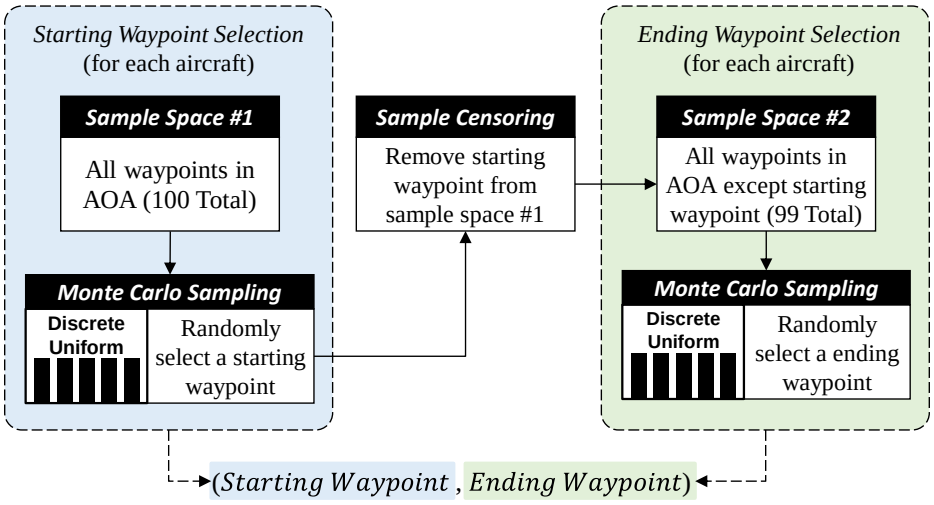


Figure 3-5: Monte Carlo Sampling Process for Random Waypoint Selection

Random number generation for Monte Carlo sampling was performed using the MRG32k3a algorithm [L'Ecuyer, 1999], providing a statistically robust method for sampling of ordered data elements with high uniformity and low periodicity [L'Ecuyer, 1999, 2001]. An extended discussion of random number generation alternatives considered and the selection of the MRG32k3a for this research is presented in Appendix C. This implementation contrasts with the conventional use of Bayesian and Markov models derived from empirical observations of ATM radar data that limits test cases to a subset of those feasible [Kochenderfer et al., 2010]. The Monte Carlo sampling approach used in this research enables the simulation of emergent air traffic patterns including any feasible flight path geometries, aircraft separation distances, or relative speeds amongst aircraft within the scope of the model constraints.

3.2.5 Agent-based Modeling and Simulation

Agent-based modeling and simulation was used to observe the behavior of aircraft via the RBA, UBA, and the DREAM for system C2 based on identical input parameters and waypoint assignments from Monte Carlo simulation. As shown in Figure 3-6, a modular simulation architecture was developed which integrated: (i) a set of baseline functionality applied to all simulated aircraft for interfacing with the simulated FFOA environment and tracking disruptions, (ii) approach-specific C2 logic for route deconfliction and collision avoidance in the FFOA, and (iii) calculation of MOEs.

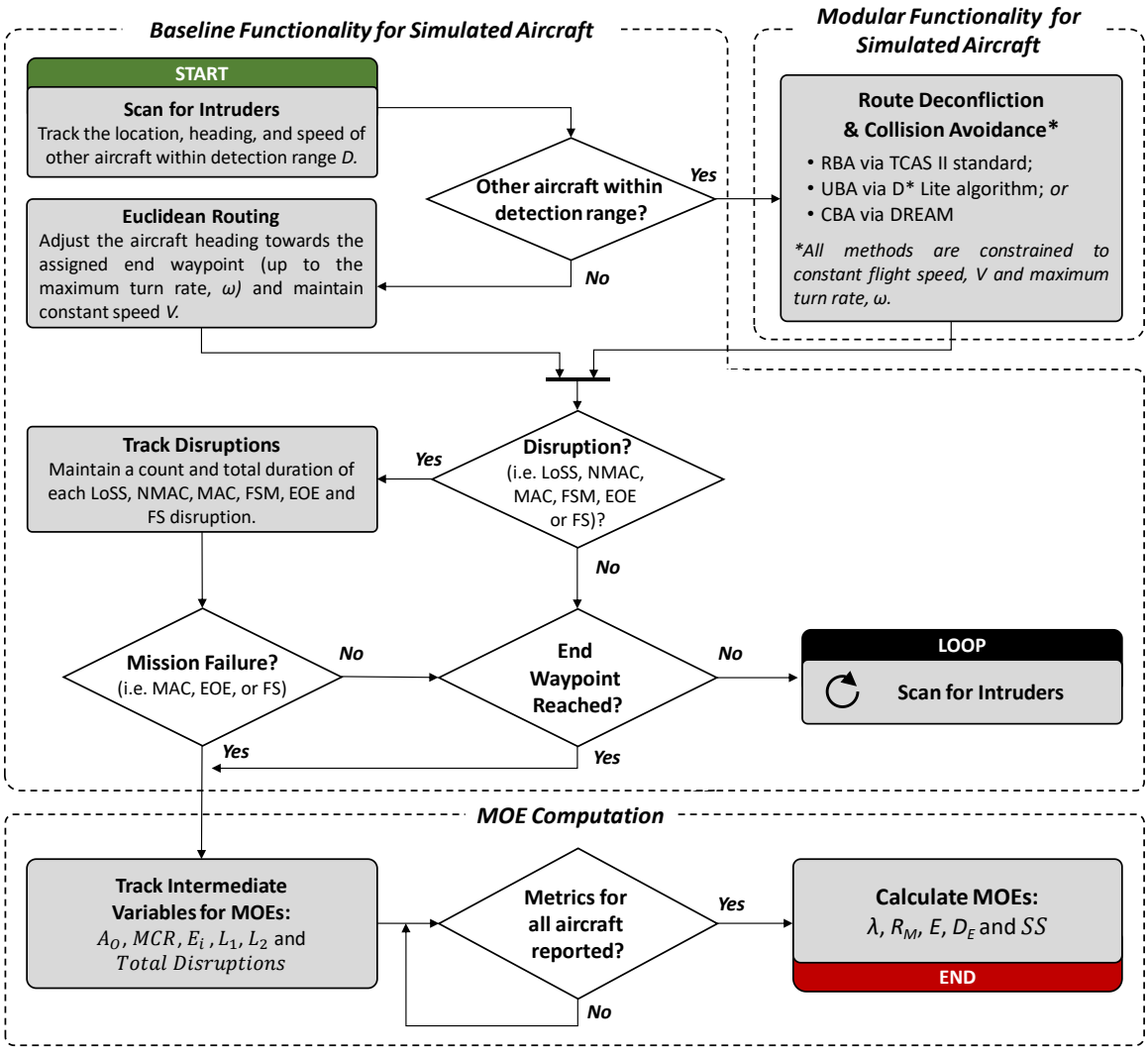


Figure 3-6: Modular Simulation Architecture and Implementation Logic

For the RBA method, simulated aircraft applied the Traffic Alert and Collision Avoidance System (TCAS II) standard [RTCA, 2008]. TCAS II specifies thresholds for physical and temporal proximity of aircraft at which collision avoidance maneuvers are triggered (Figure 3-7). Simulation logic for implementation for TCAS II is shown in Figure 3-8.

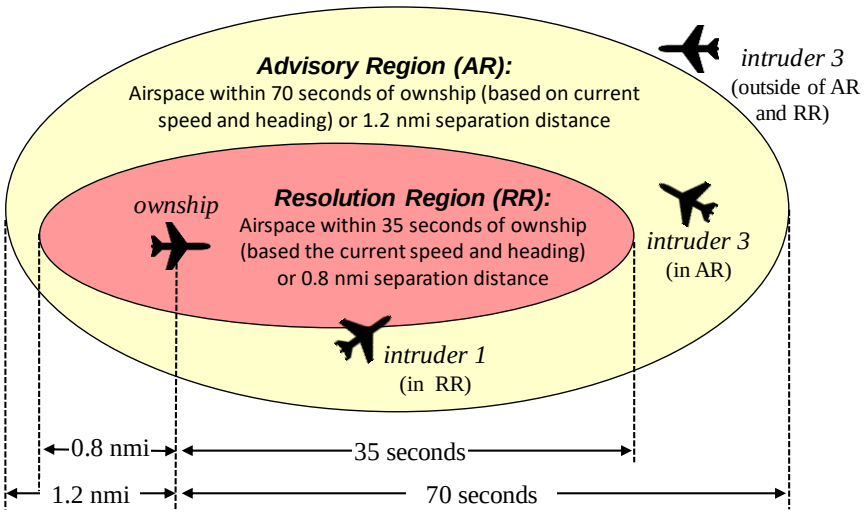


Figure 3-7: TCAS II Thresholds for Collision Avoidance Maneuvers [RTCA, 2008]

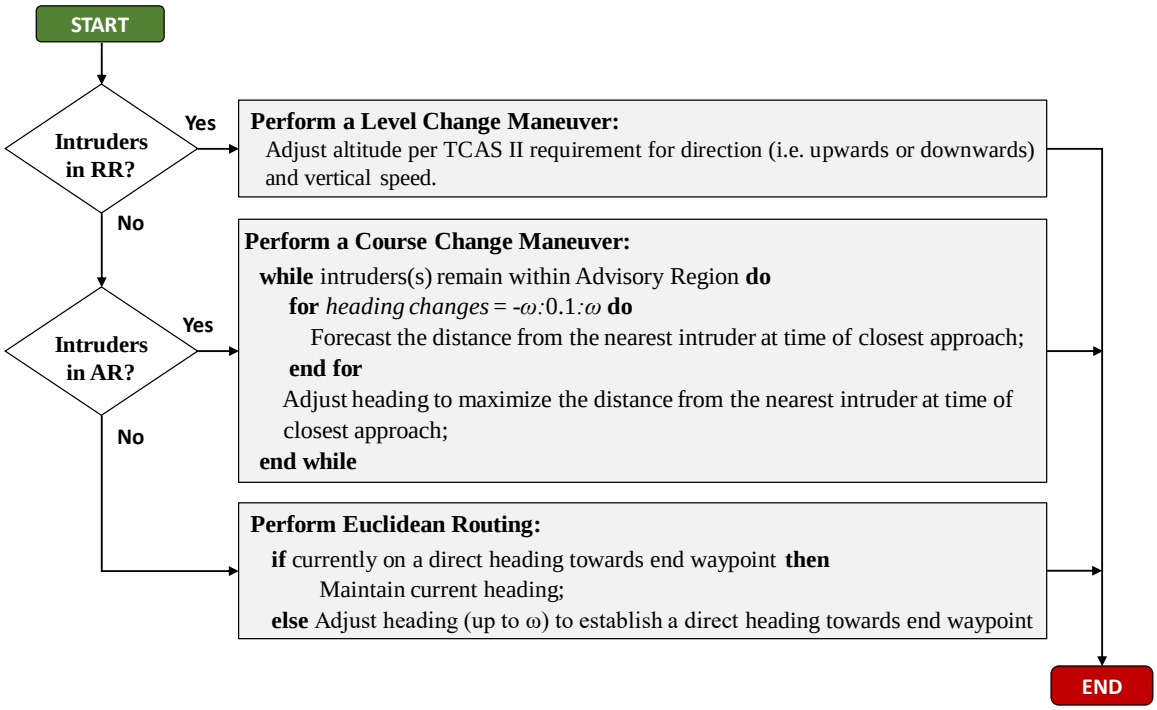


Figure 3-8: Simulation Logic for TCAS II Implementation

For the UBA method, simulated aircraft apply the Dynamic A-star Lite (D* Lite) algorithm [Koenig & Likhachev, 1998] which has been broadly adopted for vehicle automation in uncertain environments [Ferguson et al., 2005; Kortenkamp et al., 1998]. D* Lite employs utility functions to minimize total travel distance and flight time while avoiding collisions. Simulation logic for implementation of D* Lite is shown in Figure 3-9.

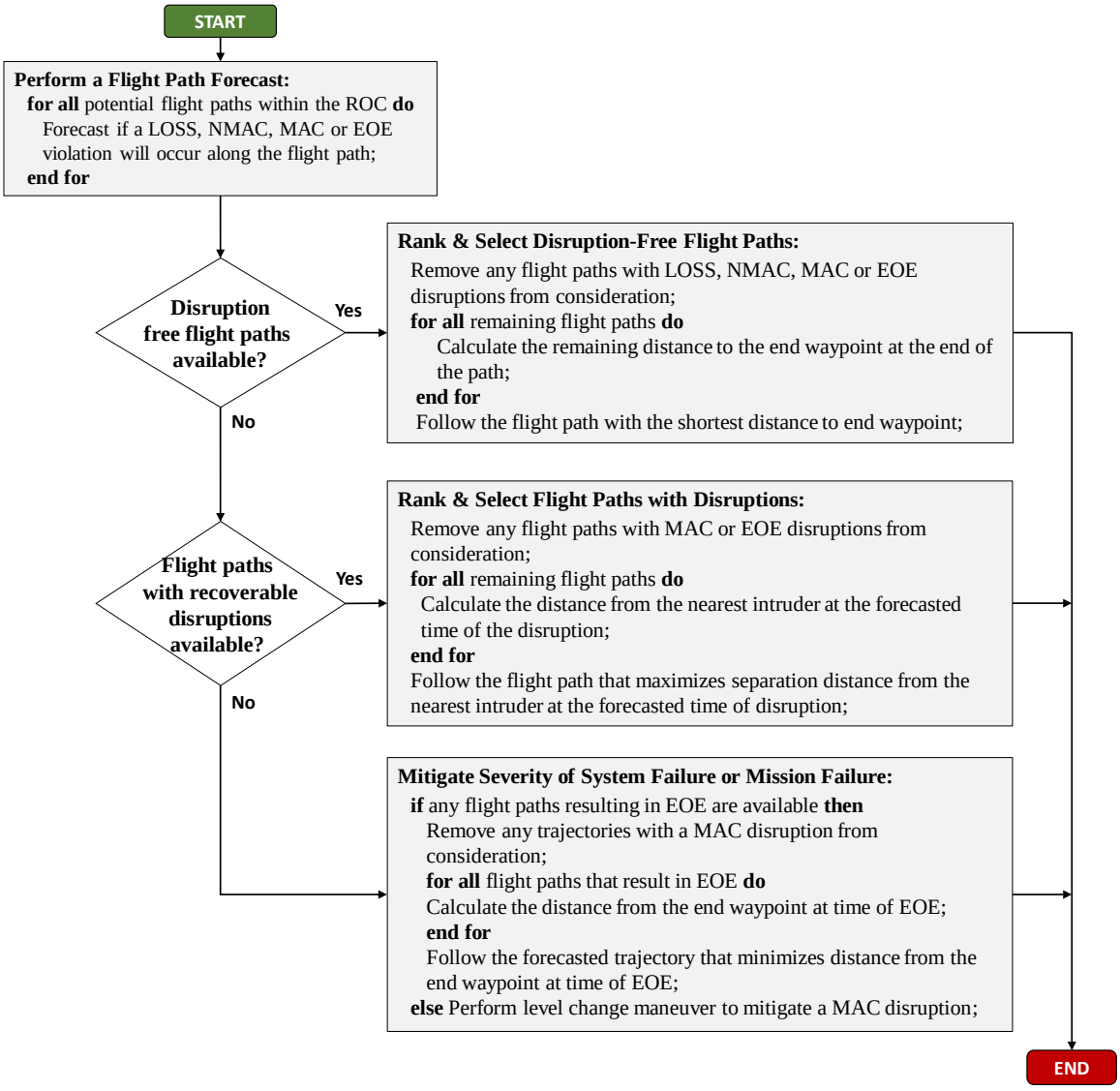


Figure 3-9: Simulation Logic for D* Lite Implementation

The DREAM is applied for Free Flight ATM using the FAA's target operational availability of 99.999% for ATM processes [FAA, 2008]. The DREAM is utilized to select turn maneuvers or trigger level change maneuvers as a failsafe action for aircraft navigation and collision avoidance Simulation logic for implementation of the DREAM for Free Flight ATM is shown in Figure 3-10.

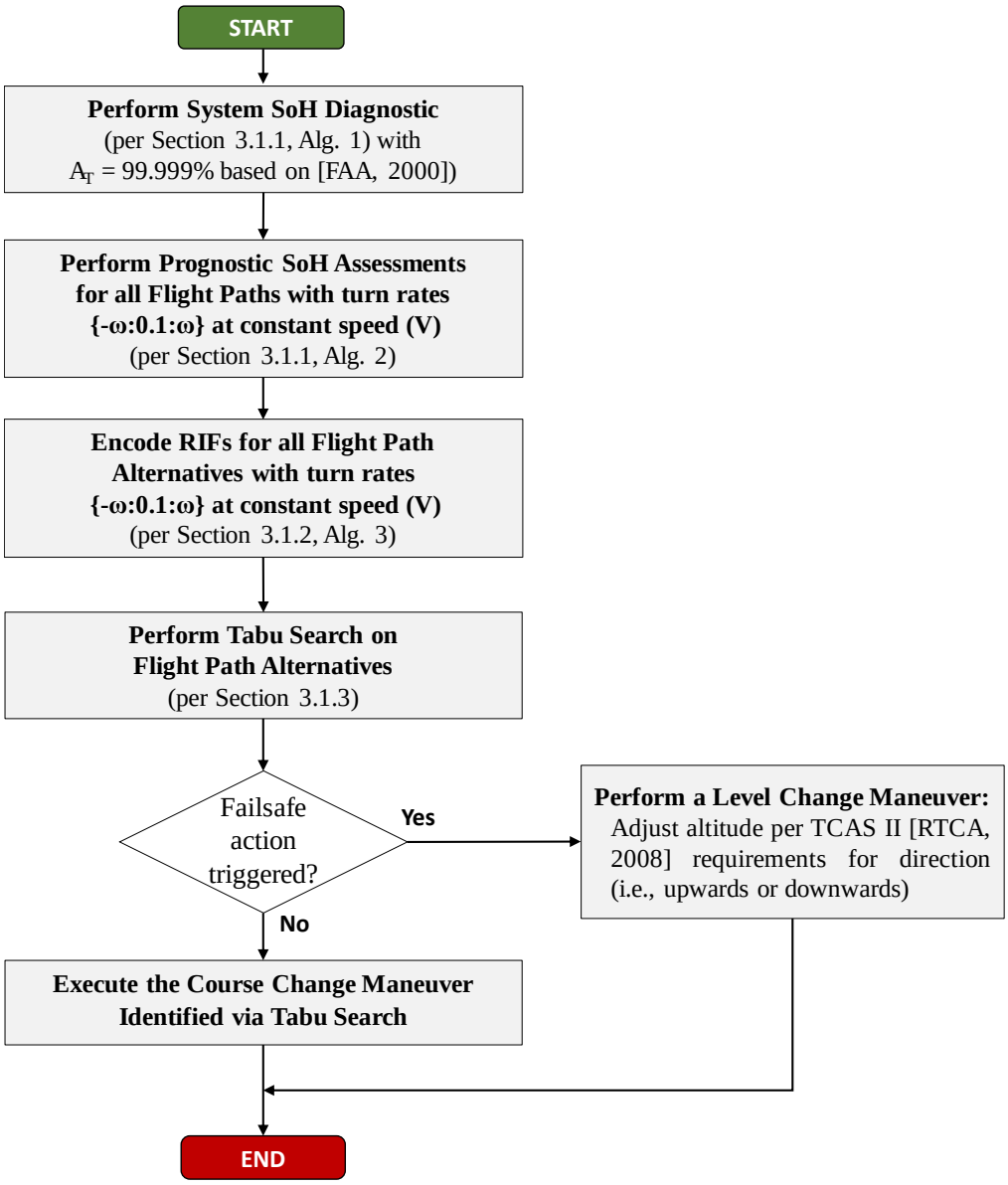


Figure 3-10: Simulation Logic for DREAM Implementation

The agent-based modeling and simulation process outputs a matched dataset of MOEs for the DREAM, TCAS II and D* Lite which enables comparison of results based on identical test HASS input parameters and Monte Carlo-generated starting and ending waypoints for aircraft.

3.3 Statistical Analyses of Matched Data-sets

Analyses of simulation outputs took into account the lack of independence between simulation outputs for the DREAM, TCAS II and D* Lite given identical test cases. Accordingly, *paired t-test* were used to evaluate the significance of the *mean difference* (μ_d) in MOEs observed between: (1) the DREAM and TCAS II; and (2) the DREAM and D* Lite. μ_d is given by

$$\mu_d = \mu_1 - \mu_2 \tag{3.10}$$

where μ_1 and μ_2 are the respective means of the two approaches being compared. For evaluation of each MOE, μ_1 was assigned as the mean of the DREAM's output and μ_2 was assigned as the mean of either TCAS II or D* Lite. As shown in Figure 3-11, the paired *t*-test procedure entails the following steps [Ott & Longnecker, 2015]:

1. Calculation of paired differences for the matched data set (detailed in Section 3.3.1);
2. Testing the distribution of paired differences for normality as critical assumption for validity of *t*-tests (detailed in Section 3.3.2);
3. Calculation of the *t* test statistic for hypotheses on μ_d (detailed in Section 3.3.3);
4. Construction of confidence intervals for μ_d (detailed in Section 3.3.4).

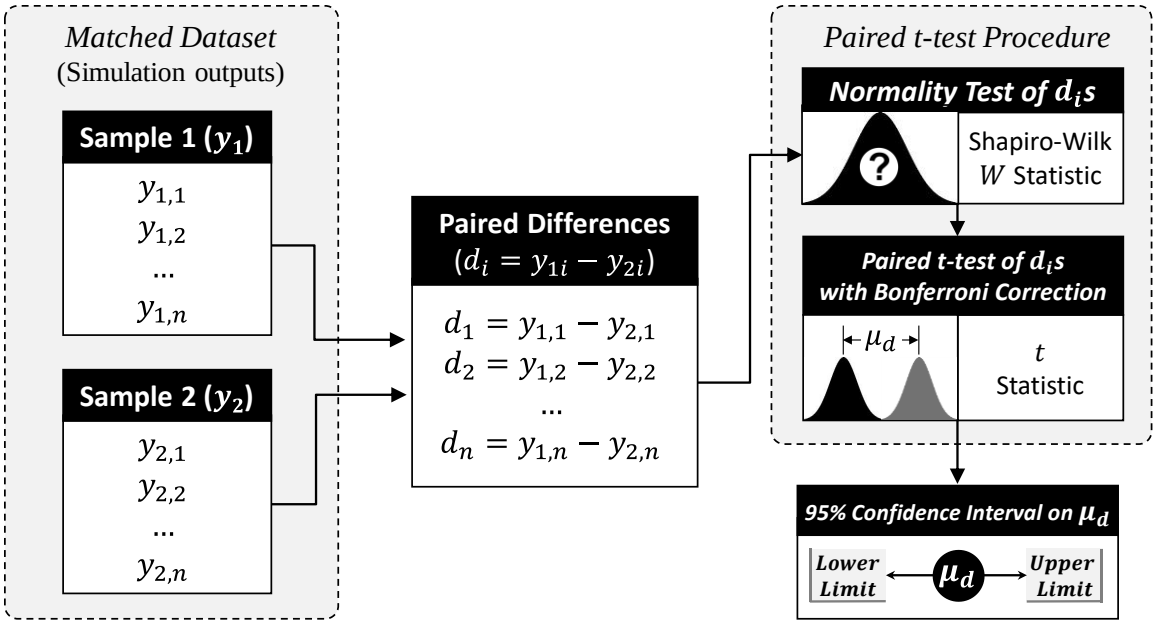


Figure 3-11: Overview of Analysis for Matched Data-sets

3.3.1 Paired Difference Calculations for MOEs

Paired differences ($d_i s$) were computed for all MOEs between (1) the DREAM and TCAS II results and (2) the DREAM and D* Lite results, using the following formula:

$$d_i = y_{1i} - y_{2i} \tag{3.11}$$

where y_1 represents sample 1 values and y_2 represents sample 2 values indexed by i. For all analyses performed, y_1 consists of outputs for the DREAM and y_2 consists of the outputs for an either TCAS II or D* Lite.

Based on d_i values, the mean of paired differences ($\bar{d}$) was then calculated using the following:

$$\bar{d} = \frac{\sum_{i=1}^{n} d_i}{n} \tag{3.12}$$

where d_i is given in (4.9) and n is the total number of paired outputs between y_1 and y_2. Using d_i, $\bar{d}$, and n, the standard deviation of paired differences (s_d) was calculated using the following formula:

$$s_d = \sqrt{\frac{\sum_{i=1}^{n}(d_i - \bar{d})^2}{n-1}} \qquad (3.13)$$

3.3.2 Normality Test of Paired Differences for MOEs

The *Shapiro-Wilk (SW) normality test* [Shapiro & Wilk, 1965] was used to assess the distribution of d_i values for all MOEs. The SW normality test establishes a null hypothesis (H_0) that the sampling distribution (with sample size n) is normal and an alternate hypothesis (H_a) that the sampling distribution is non-normal. The W test statistic is used to evaluate hypotheses and is given by

$$W = \frac{b^2}{SSD} \qquad (3.14)$$

where b is an estimate of the sample's variance from the expected values of a standard normal distribution and SSD is the sum of squared deviations of sample values from the sample mean. b is given by

$$b = \sum_{i=1}^{m} a_i(x_{n+1-i} - x_i) \qquad (3.15)$$

where a_i are the Shapiro-Wilk coefficients for a given sample size (n) [Shapiro & Wilk, 1965: pp. 603-604], x_i represent the values in the sample as ordered from lowest to highest, and $m = n/2$ when n is even, otherwise $m = (n-1)/2$ if n is odd. SSD is given by

$$SSD = \sum_{i=1}^{n} (x_i - \bar{x})^2 \qquad (3.16)$$

where $\bar{x}$ is the mean of the sample values. For the experiments presented in this study $\bar{x} = \bar{d}$, calculated using Eqn. 3.12.

Critical values for W based on sample size and confidence level (1−α) were used to determine the *p*-value (Shapiro & Wilk, 1965: p. 605). SW normality test for experiments presented in this dissertation were conducted at the 95% confidence level (i.e. α = 0.05) for a sample size of 20 (i.e. $n = 20$), with a critical value of 0.905. Accordingly, the H_0 for SW normality tests were rejected for $W \leq 0.905$, indicating that the sampling distribution is non-normal, whereas $W > 0.905$ indicates the sampling distribution is normal. Interpolation of *p*-values was performed for the critical values of W between 0.868 and 0.988, which are the minimum and maximum W values specified in the SW criticality tables, corresponding with p-values of 0.01 and 0.99 respectively. [Shapiro & Wilk, 1965: p. 605].

3.3.3 Paired *t*-test of μ_d

Paired *t*-tests were performed to assess the significance of μ_d between (1) the DREAM and TCAS II for all MOEs and (2) the DREAM and D* Lite for all MOEs. The paired t-test uses the *t* statistic to evaluate test hypotheses and is given by

$$t = \frac{\bar{d} - D_0}{s_d/\sqrt{n}} \qquad (3.17)$$

where $\bar{d}$ is calculated using Eqn. 3.12, D_0 is a specified value for which a difference in sample means to be detected, s_d is calculated using Eqn. 3.13, and *n* is the number of paired observations between samples. For all analyses presented in this dissertation D_0 =

0, enabling the detection and evaluation of significance for any change in μ_d for all MOEs. As presented in Table 3-2, paired *t*-tests were used to make inferences about μ_d based on t and a specified α-level (i.e. $\alpha = 0.05$ for all paired *t*-test performed). Hypotheses on the μ_d for all MOEs are presented in Chapter 1, Section 1.5.

Table 3-2: Paired t-test Hypotheses and Inferences on the Mean of Paired Differences

Analysis Question	Null Hypothesis (H_0)	Alternate Hypothesis (H_a)	Inferences about μ_d based on *t* and α=0.05
$\mu_1 > \mu_2$?	$\mu_d \leq 0$	$\mu_d > 0$	Reject H_0 if $t \geq t_\propto$
$\mu_1 < \mu_2$?	$\mu_d \geq 0$	$\mu_d < 0$	Reject H_0 if $t \leq -t_\propto$
$\mu_1 = \mu_2$?	$\mu_d = 0$	$\mu_d \neq 0$	Reject H_0 if $\|t\| \geq t_{\propto/2}$

$t_\propto$ is the critical value of t for single-tailed tests at a given α-level and $n-1$ degrees of freedom.
$t_{\propto/2}$ is the critical value of t for two-tailed tests at a given α-level and $n-1$ degrees of freedom.

Given that multiple hypothesis tests on μ_d were required for each measure of effectiveness, the *Bonferroni method* [Schaffer, 1995; Ott & Longnecker, 2015] was used to control the Type I error associated with results of the paired *t*-tests (i.e. the probability of rejecting the null hypothesis when it is in fact true). The Bonferroni method limits the Type I error of the individual hypothesis test performed. Doing so guarantees a specified confidence level (α) which represents the cumulative Type 1 error that is acceptable for the analysis. The Type I error for an individual hypothesis test (α_I) using the Bonferroni correction method is given by

$$\alpha_I = \frac{\alpha}{m} \qquad (3.18)$$

where *m* is the total number of hypothesis tests being performed on the sample mean [Ott & Longnecker, 2015]. Table 3-3 presents the uncorrected and Bonferroni-corrected analysis parameters used for paired *t*-testing based on a significance level of $\alpha = 0.05$

and two hypotheses ($m = 2$) for each measure of effectiveness in the ATM case study (as presented in Chapter 1, Section 1.5). All paired *t*-tests were performed using the corrected α-level of 0.025 which corresponds with a critical value of 2.093 for the *t* statistic. Minitab statistical software (Release 18.1) [Minitab, 2017] was used to perform all paired *t*-testing.

Table 3-3: Bonferroni Correction for Paired *t*-test

Uncorrected Parameters		Corrected Parameters	
Significance Level *(α)*	**Critical Value of *t****	**Bonferroni-corrected Significance Level** (α_I)	**Critical Value of *t****
0.05	1.729	0.025	2.093

*Critical value of *t* is for 19 degrees of freedom given that paired difference calculations for each measure of effectiveness are based on 20 samples.

3.3.4 Confidence Intervals for μ_d

Construction of CIs for μ_d were performed at the 95% confidence level to estimate the magnitude of differences in MOE results. CIs for μ_d are constructed using the following formula:

$$\bar{d} \pm t_{\propto/2} \frac{s_d}{\sqrt{n}} \qquad (3.19)$$

where $\bar{d}$ is calculated using Eqn. 3.12, s_d is calculated using Eqn. 3.13, and $t_{\propto/2}$ is the critical value of t for at a specified α-level and $n - 1$ degrees of freedom [Ott & Longnecker, 2015].

3.4 HASS Experiment #1

To evaluate research hypotheses 1, 2, 3 and 4 (presented in Chapter 1, Section 1.5), a HASS experiment was developed to compare the DREAM, TCAS II, and the D* Lite

algorithm with respect to λ and R_M. This section presents the HASS test case and the experimental setup for the simulation.

3.4.1 System-level Test Case

The HASS experiment for λ and R_M entailed one aircraft in the simulation designated as the system of interest, referred to as *ownship.* All other aircraft in the simulation, referred to as *intruders,* were automated to fly direct routes between their assigned starting waypoints and ending waypoints. For each intruder reaching its ending waypoint, a new intruder is generated within the FFOA to maintain a constant air traffic density level throughout the simulation while ownship was en-route. Ownship and intruders are *uncooperative,* meaning that no information is exchanged between ownship and intruders for route deconfliction. However, all intruders are cooperative with other intruders and inherently deconflicted in the simulation. Nominally, ownship is permitted to make left and right turns for navigation and collision avoidance. As a failsafe, *level-change maneuvers* which change the aircraft's altitude can be triggered for collision avoidance but result in mission failure. Ownship is continuously monitored for LoSS, NMAC, MAC, FS, and EOE disruptions (per Section 3.2.3) and the ability to reach its assigned ending waypoint, i.e. mission completion. Figure 3-12 illustrates the system-level HASS test case.

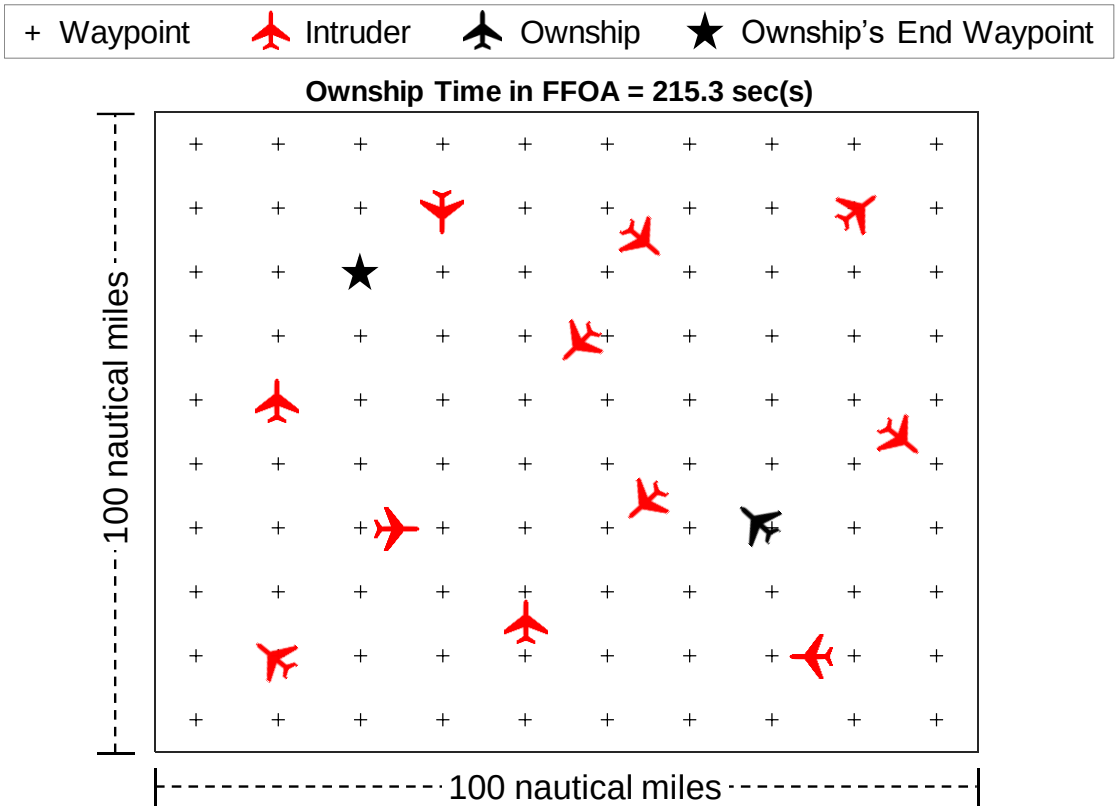


Figure 3-12: Example of System-level HASS Test Case

MATLAB® simulation showing ownship and 11 intruders (i.e. ρ = 12 aircraft/10,000 nmi^2) navigating between randomly assigned starting and ending waypoints within an FFOA covering 10,000 square nautical miles. Figure is not to scale.

3.4.2 Experiment #1 Setup and Execution

The experiment flow for the system-level test case is shown in Figure 3-13. Model input parameters including rationale are presented in Table 3-4. For each ρ level, 20 simulations were executed with each consisting of 1000 Monte Carlo iterations to capture stochastic effects. Average results for λ and R_M are output for the DREAM, TCAS II, and D* Lite for each simulation performed. Simulation results and analysis are presented in (Section 4.1).

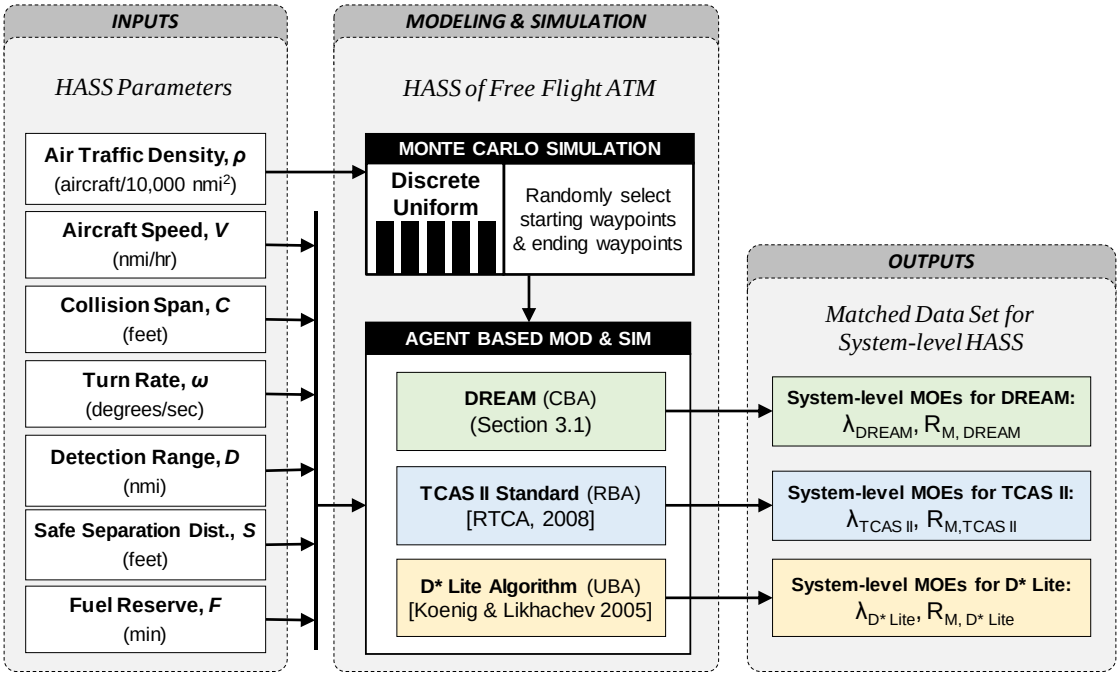


Figure 3-13: Test Model for System-level HASS

Table 3-4: Model Inputs for System-level HASS

Parameter	Setting(s)	Rationale
ρ	{5, 10, 15, 20, 25, 30, 35, 45} aircraft/10K nmi^2	2030 PIAC estimates for the NAS [Henriksen, 2008; EUROCONTROL/FAA, 2006].
C	115 ft	Representative wingspan of the Airbus A320 [Airbus, 2017; FAA, 2017a] and Boeing 737 [Boeing, 2017; FAA, 2017a] aircraft prevalent in high-altitude operations targeted for AOA implementation.
D	14 nmi	Reliable detection range for required ACAS equipment for commercial aircraft operating in the NAS [RTCA, 2008].
F	45 minutes	U.S. Code of Federal Regulations for Instrument Flight Rules [CFR, 2000].
S	4000 ft	RTCA DO-365 standard for horizontal safe separation of large UAS in the NAS [RTCA, 2017].
V	450 nmi/hr	Representative cruise speed of the Airbus A320 [Airbus, 2017] and Boeing 737 [Boeing, 2017] aircraft prevalent in high-altitude operations targeted for AOAs.
ω	3°/sec	FAA standard rate turn [FAA, 2017b].

3.5 HASS Experiment #2

To evaluate research hypotheses 4 through 10 (presented in Chapter 1), a HASS experiment was developed to compare the DREAM, TCAS II, and the D* Lite algorithm based on system efficiency, standard deviation of efficiency and system stability. This section presents the associated HASS test case and the experimental setup for the simulation.

3.5.1 DRA Test Case

The HASS experiment for E, $D_{E,}$ and SS addresses the collective performance of all aircraft in a FFOA, representing a DRA problem for a system of systems (SoS). All aircraft in this simulation are uncooperative when navigating between their randomly assigned starting waypoint and ending waypoints. Nominally, all aircraft are permitted to make left and right turns for navigation and collision avoidance. As a failsafe, level-change maneuvers which change the aircraft's altitude can be triggered for collision avoidance. Ownship is continuously monitored for LoSS, NMAC, MAC, FSM, and EOE disruptions (as detailed in Section 3.2.3) and total flight time required to reach its assigned ending waypoint. The fuel reserve parameter and FS disruptions were excluded in the modeling and simulation for this test case given a focus on airspace as a shared finite resource rather than individually controlled resources or constraints. Figure 3-14 illustrates the HASS modeling and simulation for the DRA test case.

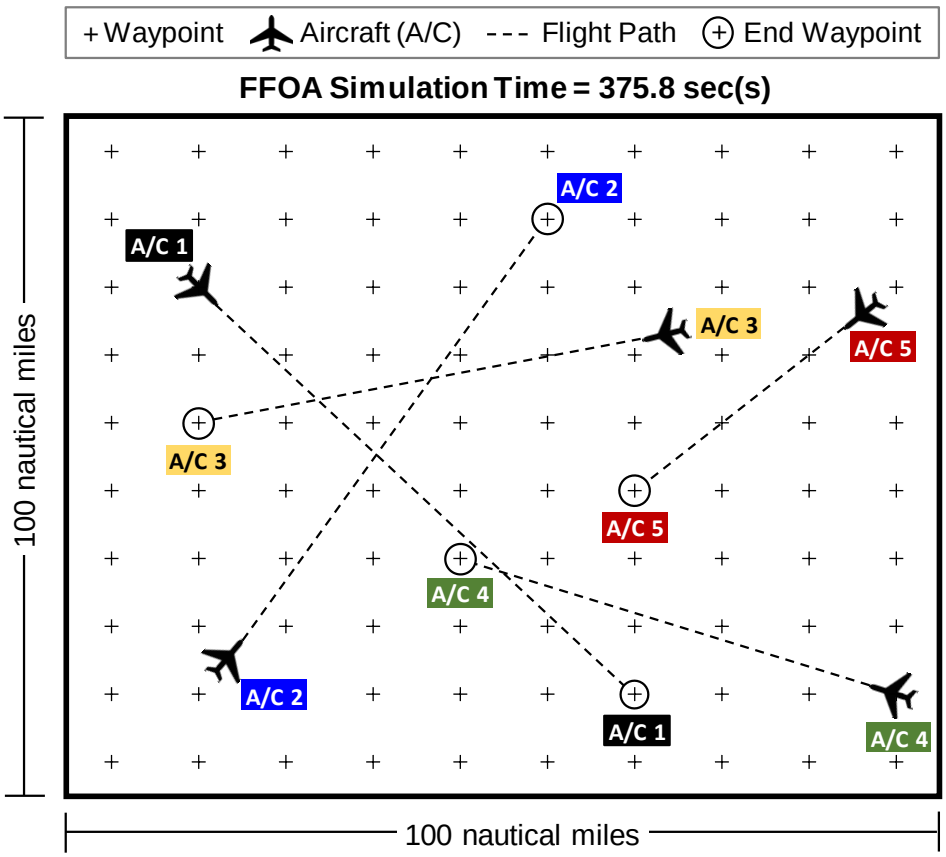


Figure 3-14: Example of DRA Test Case Simulation

MATLAB® simulation showing 5 aircraft navigating between randomly assigned starting and ending waypoints within an AOA covering 10,000 square nautical miles (i.e. ρ = 5 aircraft/10,000 nmi^2). Figure is not to scale.

3.5.2 Experiment #2 Setup and Execution

The experiment flow for DRA test case is shown in Figure 3-15. Model input parameters including rationale are presented in Table 3-5. For each ρ level, 20 simulations were executed with each consisting of 1000 Monte Carlo iterations to capture stochastic effects. Average results for E, D_E and SS are output for the DREAM, TCAS II, and D* Lite for each simulation performed. Simulation results and analysis are presented in Section 4.2.

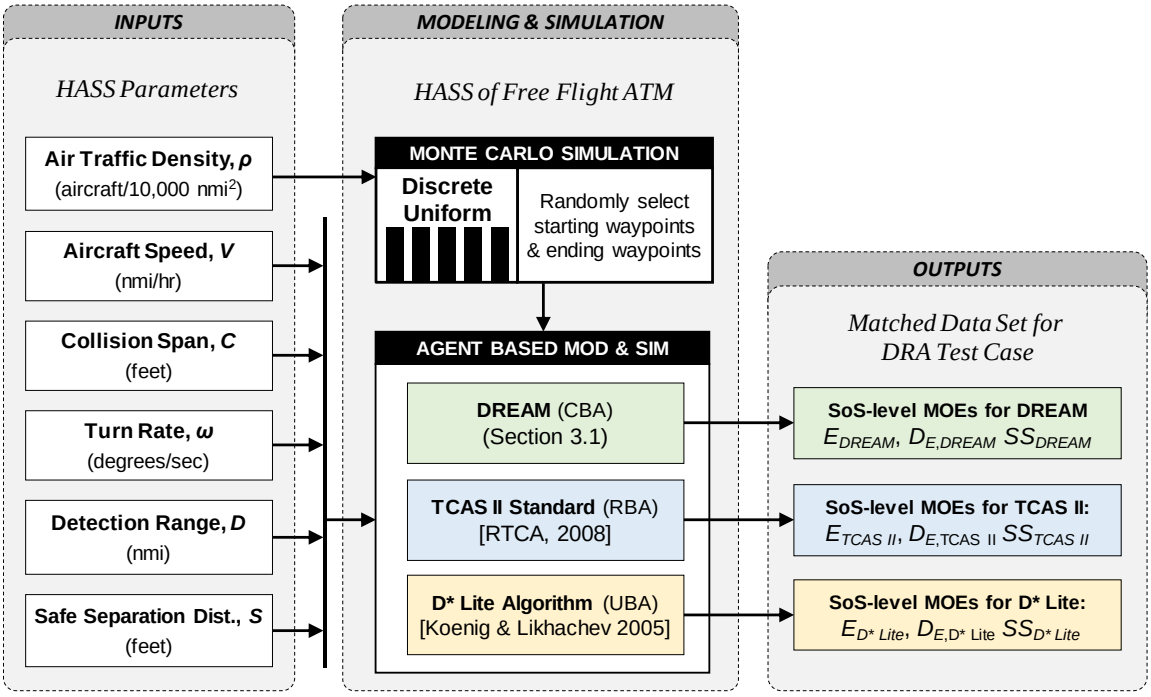


Figure 3-15: Test Model for SoS-level HASS

Table 3-5: Model Inputs for SoS-level HASS

Parameter	Setting(s)	Rationale
ρ	{5, 10, 15, 20, 25, 30, 35, 45} aircraft/10K nmi^2	2030 PIAC estimates for the NAS [Henriksen, 2008; EUROCONTROL/FAA, 2006].
C	115 ft	Representative wingspan of the Airbus A320 [Airbus, 2017; FAA, 2017a] and Boeing 737 [Boeing, 2017; FAA, 2017a] aircraft prevalent in high-altitude operations targeted for AOA implementation.
D	14 nmi	Reliable detection range for required ACAS equipment for commercial aircraft operating in the NAS [RTCA, 2008].
S	4000 ft	RTCA DO-365 standard for horizontal safe separation of large UAS in the NAS [RTCA, 2017].
V	450 nmi/hr	Representative cruise speed of the Airbus A320 [Airbus, 2017] and Boeing 737 [Boeing, 2017] aircraft prevalent in high-altitude operations targeted for AOAs.
ω	3°/sec	FAA standard rate turn [FAA, 2017b].

Chapter 4: Results & Analysis

The results of the HASS modeling and simulation as described in Chapter 3 are presented in the following sections along with the findings of statistical analyses to evaluate research questions and hypotheses (as presented in Section 1.4). The results of Experiment 1, which performed a HASS of incident rate and mission reliability, are presented in Section 4.1. Results for Experiment 2, which executed a HASS of system efficiency, standard deviation of efficiency, and system stability, are presented in Section 4.2. The chapter concludes with the analysis of results associated with research questions in Section 4.3. This chapter is outlined as follows:

- **Section 4.1 – Results of Experiment 1 (HASS of λ and R_M)**
  - *Section 4.1.1* – Comparison of TCAS II and DREAM Results for λ and R_M
  - *Section 4.1.2* – Comparison of D* Lite and DREAM Results for λ and R_M
- **Section 4.2 – Results of Experiment 2 (HASS of E, D_E, and SS)**
  - *Section 4.2.1* – Comparison of TCAS II and DREAM Results for E, D_E, and SS
  - *Section 4.2.2* – Comparison of D* Lite and DREAM Results for E, D_E, and SS
- **Section 4.3 –Analysis of Results**
  - *Section 4.3.1* – analysis and findings for hypotheses associated with Research Question 1 ($\boldsymbol{H_{1A}}$ and $\boldsymbol{H_{1B}}$)
  - *Section 4.3.2* – analysis and findings for hypotheses associated with Research Question 2 ($\boldsymbol{H_{2A}}$ and $\boldsymbol{H_{2B}}$)
  - *Section 4.3.3* – analysis and findings for hypotheses associated with Research Question 3 ($\boldsymbol{H_{3A}}$ and $\boldsymbol{H3_B}$)
  - *Section 4.3.4* – analysis and findings for hypotheses associated with Research Question 4 ($\boldsymbol{H_{4A}}$ and $\boldsymbol{H_{4B}}$)

- *Section 4.3.5* – analysis and findings for hypotheses associated with Research Question 5 ($\boldsymbol{H_{5A}}$ and $\boldsymbol{H_{5B}}$)

4.1 Results for Experiment #1 (HASS of λ and R_M)

Based on the HASS experiment presented in Section 3.4, Free Flight ATM simulations were executed to evaluate and compare performance of the DREAM, TCAS II and the D* Lite algorithm with respect to Incident Rate (λ) and Mission Reliability (R_M). Simulation results reflecting the mean and standard error of the mean (SE) for λ and R_M based on 20 simulations at each air traffic density level are presented graphically in Figures 4-1 and 4-2, with corresponding numerical summaries presented in Tables 4-1 and 4-2.

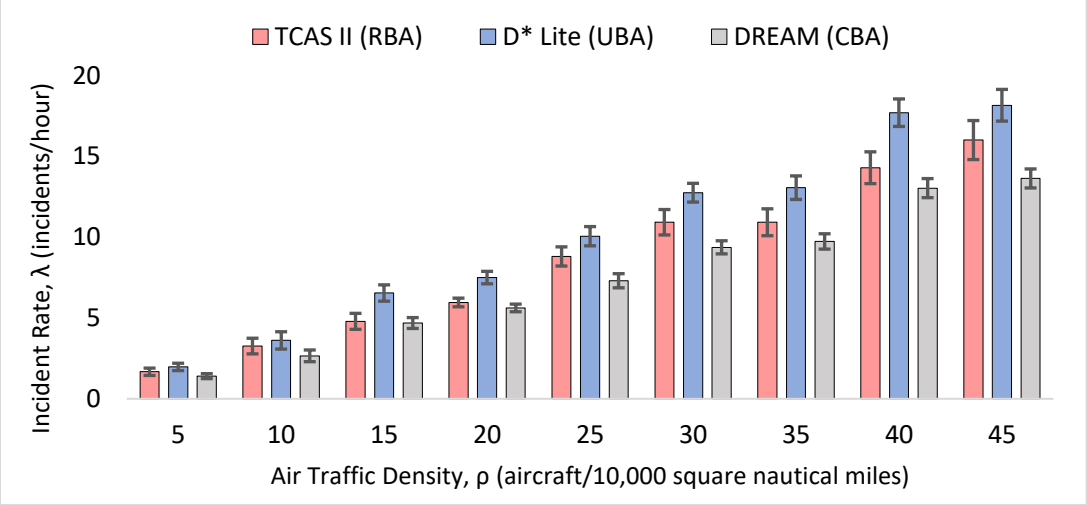


Figure 4-1: Mean and SE of Incident Rate based on 20 simulations

Table 4-1: MEAN AND SE OF INCIDENT RATE

ρ (aircraft per 10000 sq. nmi)	TCAS II (RBA)		D* Lite (UBA)		DREAM (CBA)	
	$\bar{\lambda}$	$SE_{\bar{\lambda}}$	$\bar{\lambda}$	$SE_{\bar{\lambda}}$	$\bar{\lambda}$	$SE_{\bar{\lambda}}$
5	1.67	0.23	1.97	0.23	1.40	0.15
10	3.26	0.48	3.61	0.54	2.66	0.36
15	4.79	0.49	6.54	0.51	4.69	0.34
20	5.96	0.27	7.50	0.39	5.62	0.24
25	8.81	0.59	10.06	0.60	7.31	0.44
30	10.92	0.79	12.75	0.58	9.37	0.41
35	10.92	0.83	13.06	0.73	9.74	0.48
40	14.30	0.98	17.70	0.85	13.04	0.59
45	16.01	1.21	18.16	0.98	13.64	0.59

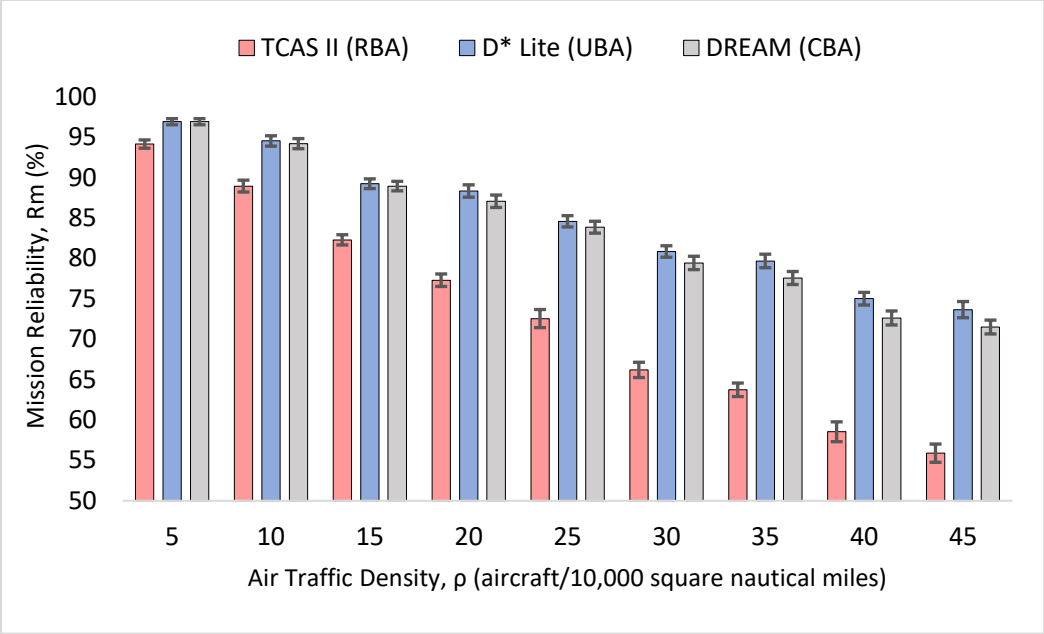


Figure 4-2: Mean and SE of Mission Reliability

Table 4-2: MEAN AND SE OF MISSION RELIABILITY

ρ (aircraft per 10000 sq. nmi)	TCAS II (RBA)		D* Lite (UBA)		DREAM (CBA)	
	$\overline{R_M}$	$SE_{\overline{R_M}}$	$\overline{R_M}$	$SE_{\overline{R_M}}$	$\overline{R_M}$	$SE_{\overline{R_M}}$
5	94.15	0.51	96.92	0.38	96.92	0.38
10	88.95	0.73	94.54	0.64	94.20	0.63
15	82.29	0.63	89.24	0.60	88.95	0.59
20	77.30	0.77	88.34	0.77	87.07	0.77
25	72.55	1.12	84.59	0.70	83.86	0.73
30	66.20	0.95	80.85	0.71	79.44	0.83
35	63.74	0.84	79.68	0.84	77.57	0.81
40	58.54	1.23	75.01	0.78	72.63	0.87
45	55.89	1.13	73.66	1.00	71.51	0.86

4.1.1 Comparison of TCAS II and DREAM Results for Experiment 1

A direct comparison of TCAS II and DREAM results for Experiment 1 is presented in Table 4-3, showing the percent change in incident rate and mission reliability based on identical model inputs and constraints. For all air traffic density levels tested, the DREAM yielded the lowest incident rate and highest mission reliability in comparison to TCAS II, averaging a 12% decrease in the frequency of system disruptions and a 15%

increase in the satisfaction of system performance objectives. In the context of the Free Flight ATM application, these results reflect that the DREAM provides the following benefits in comparison to the existing TCAS II standard: (i) fewer Loss of Safe Separation (LoSS), Near Mid-air Collision (NMAC), Mid-air Collision (MAC), Exceedance of Envelope (EOE), and Fuel Shortage (FS) disruptions observed, resulting in improved safety; (ii) a higher proportion of completed flights without precoordinated flight paths, indicating increased operational efficacy and flexibility; and (iii) improved operational contingency without human intervention.

Table 4-3: Percent Change in Results for Experiment 1– TCAS II vs. DREAM

ρ (aircraft/10000 nmi^2)	$\Delta\lambda$ (%)	ΔR_M (%)
5	-18.32	3.04
10	-17.23	5.79
15	-0.65	7.64
20	-4.81	12.92
25	-17.14	15.61
30	-16.88	19.68
35	-10.85	20.94
40	-8.54	23.83
45	-13.54	28.07
Avg.	*-12.00*	*15.28*

Percent change calculated as:

$\Delta\%$ = (CBA Result – RBA Result) / (RBA Result) * 100

= (DREAM Result – TCAS II Result) / (TCAS II Result) * 100

4.1.2 Comparison of D* Lite and DREAM Results for Experiment 1

Experiment 1 results for the D* Lite algorithm and the DREAM are directly compared in Table 4-4, showing the relative change in incident rate and mission reliability between methods. Across all air traffic density levels, the DREAM yielded the lowest incident rate in comparison to D* Lite, averaging over 25% fewer incidents per hour. In terms of mission reliability, the DREAM averaged nearly 1.5% fewer mission

completions than D* Lite. In pragmatic terms, results demonstrate the DREAM provided improved disruption avoidance for autonomous aircraft in the FFOA in comparison to D* Lite, with fewer LoSS, NMAC, MAC, EOE, and FS disruptions observed.

Table 4-4: Percent Change in Results for Experiment 1– D* Lite vs. DREAM

ρ (aircraft/10K nmi^2)	Δλ (%)	ΔR_M (%)
5	-28.84	0.00
10	-26.29	-0.40
15	-28.39	-0.27
20	-24.58	-1.42
25	-27.95	-0.69
30	-26.77	-1.69
35	-25.31	-2.78
40	-26.42	-2.98
45	-24.21	-3.04
Avg.	-26.53	-1.48

Percent change calculated as:
Δ% = (CBA Result – UBA Result) / (UBA Result) * 100
= (DREAM Result – D* Lite Result) / (D* Lite Result) * 100

4.2 Results for Experiment 2 (HASS of E, D_E, and SS)

Based on the HASS experiment presented in Section 3.5, Free Flight simulations were executed to evaluate and compare performance of the DREAM, TCAS II and D* Lite with respect to system efficiency (E), standard deviation of efficiency (D_E) and system stability (SS). Simulation results reflecting the mean and standard error of the mean for E, D_E and SS based on 20 simulations at each air traffic density level are presented graphically in Figures 4-3 through 4-5, with numerical summaries presented in Tables 4-5 through 4-7.

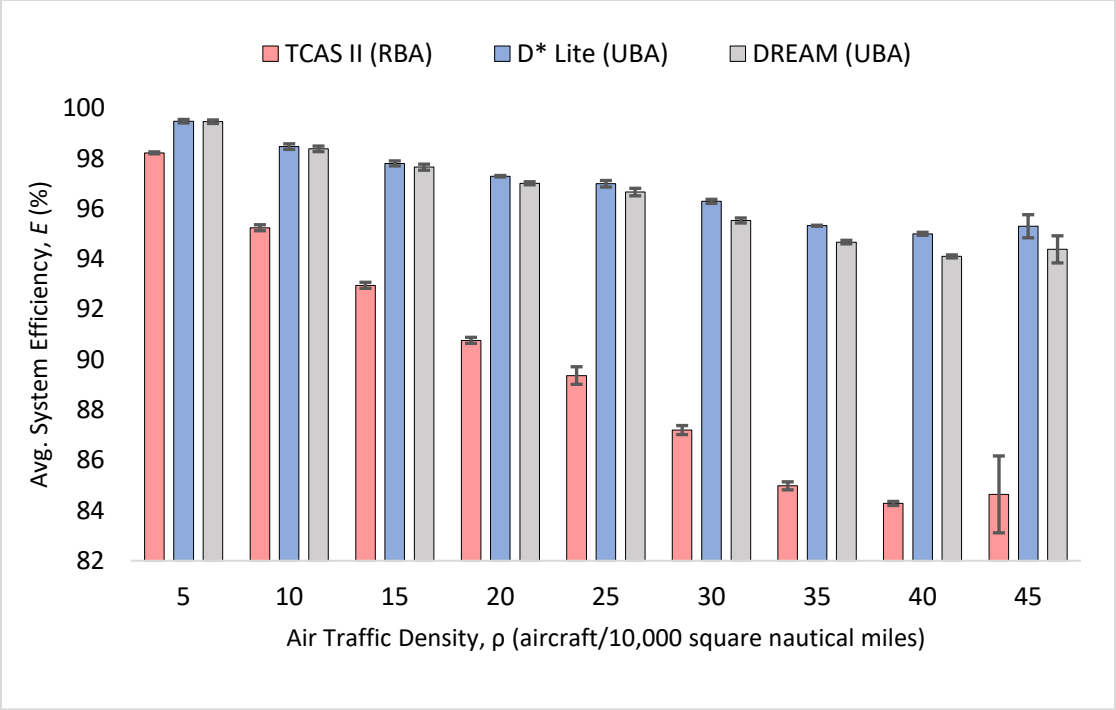


Figure 4-3: Mean and SE of the System Efficiency

Table 4-5: MEAN AND SE OF SYSTEM EFFICIENCY

ρ (aircraft per 10000 sq. nmi)	TCAS II (RBA)		D* Lite (UBA)		DREAM (CBA)	
	$\bar{E}$	$SE_{\bar{E}}$	$\bar{E}$	$SE_{\bar{E}}$	$\bar{E}$	$SE_{\bar{E}}$
5	98.23	0.04	99.49	0.07	99.47	0.07
10	95.25	0.12	98.48	0.11	98.39	0.11
15	92.96	0.12	97.81	0.1	97.66	0.12
20	90.77	0.12	97.30	0.03	97.02	0.06
25	89.37	0.35	97.00	0.13	96.67	0.15
30	87.20	0.18	96.30	0.08	95.54	0.10
35	84.98	0.16	95.33	0.02	94.68	0.07
40	84.28	0.08	95.01	0.06	94.11	0.06
45	84.64	1.53	95.31	0.46	94.39	0.54

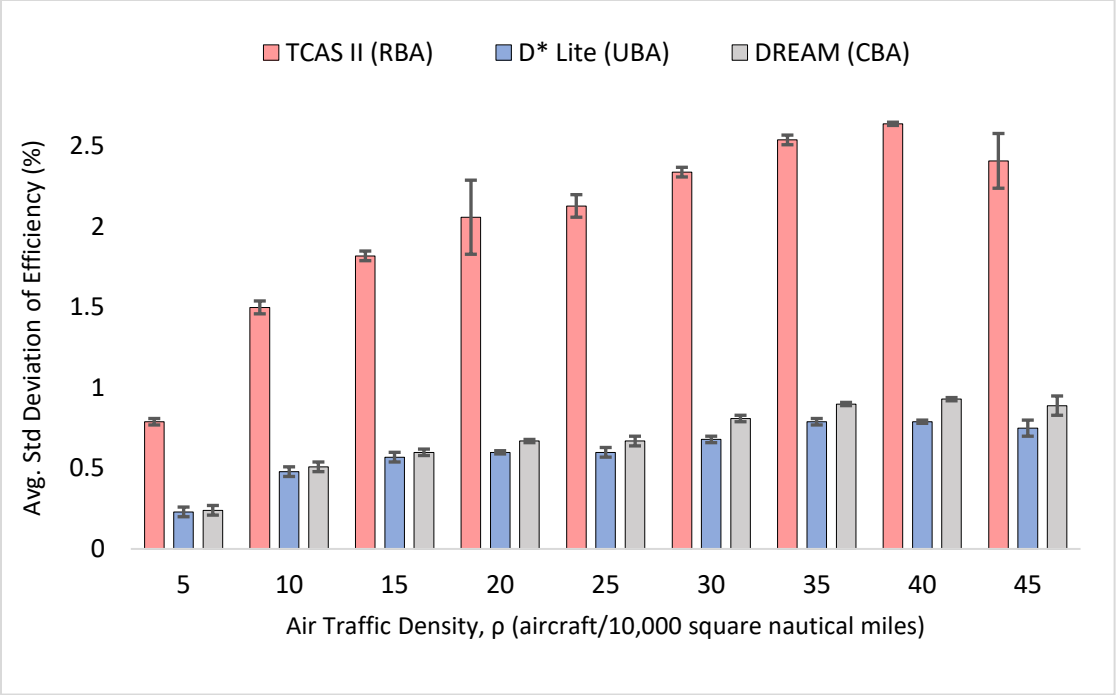


Figure 4-4: Mean and SE of the Standard Deviation of Efficiency, D_E

Table 4-6: MEAN AND SE OF THE STANDARD DEVIATION OF EFFICIENCY, D_E

ρ (aircraft per 10000 sq. nmi)	TCAS II (RBA)		D* Lite (UBA)		DREAM (CBA)	
	$\overline{D_E}$	$SE_{\overline{D_E}}$	$\overline{D_E}$	$SE_{\overline{D_E}}$	$\overline{D_E}$	$SE_{\overline{D_E}}$
5	0.79	0.02	0.23	0.03	0.24	0.03
10	1.50	0.04	0.48	0.03	0.51	0.03
15	1.82	0.03	0.57	0.03	0.60	0.02
20	2.06	0.23	0.60	0.01	0.67	0.01
25	2.13	0.07	0.60	0.03	0.67	0.03
30	2.34	0.03	0.68	0.02	0.81	0.02
35	2.54	0.03	0.79	0.02	0.90	0.01
40	2.64	0.01	0.79	0.01	0.93	0.01
45	2.41	0.17	0.75	0.05	0.89	0.06

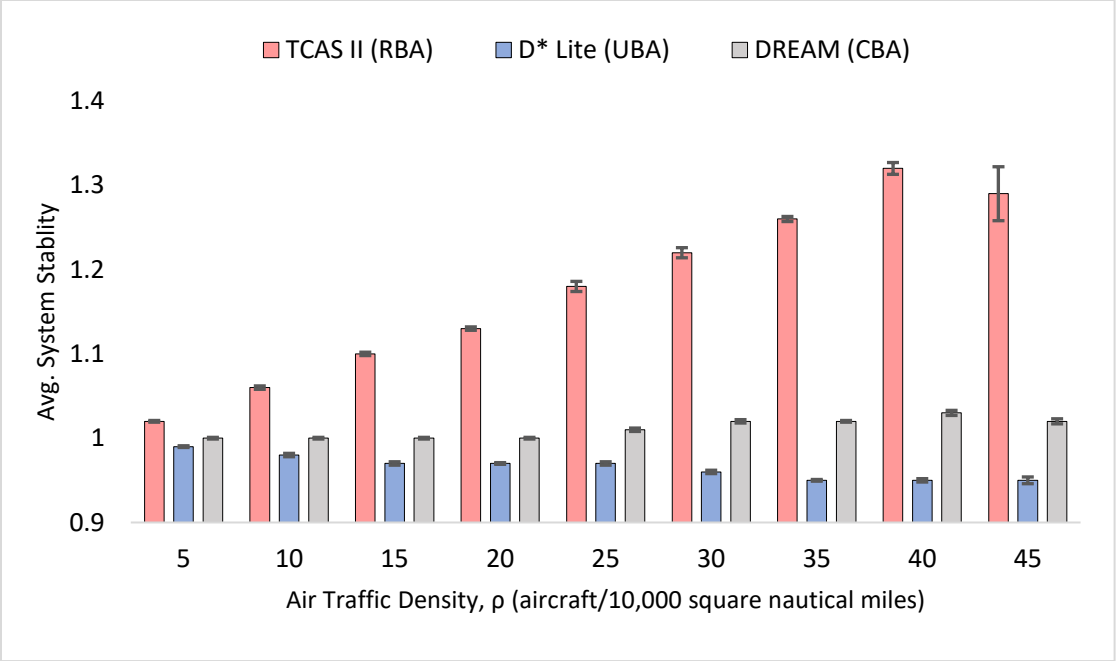


Figure 4-5: Mean and SE of System Stability, *SS*

Table 4-7: MEAN AND SE OF SYSTEM STABILITY, *SS*

ρ (aircraft per 10000 sq. nmi)	TCAS II (RBA)		D* Lite (UBA)		DREAM (CBA)	
	$\overline{SS}$	SE_{SS}	$\overline{SS}$	SE_{SS}	$\overline{SS}$	$SE_{\overline{SS}}$
5	1.02	0.001	0.99	0.001	1.00	0.001
10	1.06	0.002	0.98	0.002	1.00	0.001
15	1.10	0.002	0.97	0.002	1.00	0.001
20	1.13	0.002	0.97	0.001	1.00	0.001
25	1.18	0.006	0.97	0.002	1.01	0.002
30	1.22	0.006	0.96	0.002	1.02	0.002
35	1.26	0.003	0.95	0.001	1.02	0.001
40	1.32	0.007	0.95	0.002	1.03	0.003
45	1.29	0.032	0.95	0.004	1.02	0.003

4.2.1 Comparison of TCAS II and DREAM Results for Experiment 2

A direct comparison the TCAS II and DREAM results for Experiment 2 is presented in Table 4-8, reflecting the percent change in E, D_E and SS based on identical model inputs and constraints. Across all air traffic density levels tested, the DREAM resulted in higher E and lower D_E than TCAS II, averaging a 7.5% reduction in flight delay time and

a 66% reduction in the variation of delay time amongst aircraft. Conversely, the DREAM resulted in lower system stability than TCAS II for all air traffic density levels tested, with 13% fewer system disruptions avoided on average.

Table 4-8: Percent Change in Results for Experiment 2 – TCAS II vs. DREAM

ρ (aircraft/10K nmi^2)	ΔE (%)	ΔD_E (%)	ΔSS (%)
5	1.27	-70.37	-1.99
10	3.29	-66.10	-5.23
15	5.06	-66.80	-8.89
20	6.89	-67.69	-11.09
25	8.17	-68.68	-13.82
30	9.57	-65.20	-17.10
35	11.41	-64.60	-19.16
40	10.95	-62.34	-19.28
45	11.51	-63.18	-21.02
Avg.	7.57	-66.11	-13.07

Percent change calculated as:

$\Delta\%$ = (CBA Result – RBA Result) / (RBA Result) * 100

= (DREAM Result – TCAS II Result) / (TCAS II Result) * 100

4.2.2 Comparison of D* Lite and DREAM Results for Experiment 2

Experiment 2 results for D* Lite and the DREAM are directly compared in Table 4-9, showing the relative change in E, D_E and SS based on identical model inputs. For all air traffic density levels, the DREAM resulted in decreased E with increased D_E when compared to D* Lite, with a 0.5% increase in flight delay time and a 12% increase in the variation of delay times between aircraft observed. In contrast, the DREAM provided increased SS in comparison to D* Lite, averaging over 4.5% fewer LoSS, NMAC, MAC, EOE and FSM disruptions observed in the FFOA.

Table 4-9: Percent Change in Results for Experiment 2 – D* Lite vs. DREAM

ρ (aircraft/10K nmi^2)	ΔE (%)	ΔD_E (%)	ΔSS (%)
5	-0.02	3.78	0.68
10	-0.09	5.56	1.81
15	-0.15	6.78	3.14
20	-0.29	10.49	3.84
25	-0.34	10.94	4.83
30	-0.78	20.28	5.69
35	-0.68	13.80	6.71
40	-0.95	18.07	7.91
45	-0.97	19.06	6.86
Avg.	-0.47	12.09	4.61

Percent change calculated as:

$\Delta\%$ = (CBA Result – UBA Result) / (UBA Result) * 100

= (DREAM Result – D* Lite Result) / (D* Lite Result) * 100

4.3 Analysis of Results

In order to assess the significance of the results and evaluate hypotheses associated with research questions, statistical analyses were performed that accounted for the lack of independence in the outputs generated by Free Flight modeling and simulation (as specified in Chapter 3, Section 3.3) The following sections present the analysis and findings associated with the research questions and hypotheses (presented in Chapter 1, Sections 1.4 and 1.5), including the following:

1. *Paired Difference Calculations* for each measure of effectiveness, as specified in Section 3.3.1;

2. *Normality Testing of Paired Differences* for each measure of effectiveness as a pre-requisite for validity of *t*-tests, as specified in Section 3.3.2;

    - *Note:* Normal probability plots of paired differences are presented in Appendix D for Experiment 1 results (i.e. λ and R_M) and Appendix E for Experiment 2 results (i.e. E, D_E, and SS).

3. *Paired t-tests* with Bonferroni correction applied to evaluate hypotheses on the mean difference for each measure of effectiveness, as specified in Section 3.3.3; and

4. *Construction of 95% Confidence Intervals (CIs)* for the mean of paired differences, as specified in Section 3.3.4.

4.3.1 Analysis and Findings for Research Question 1

The first research question is as follows (as outlined in Section 1.4):

> ***Research Question 1:*** *Does the developed model reduce the frequency of system disruptions in emergent operating environments when compared to rule-based and utility-based system automation methods?*

The associated hypotheses for Research Question 1 are as follows (as outlined in Section 1.5):

> $\boldsymbol{H_{1A}}$: *There is a statistically significant decrease in the frequency of system disruptions via the DREAM in comparison to TCAS II for Free Flight ATM.*
>
> $\boldsymbol{H_{1B}}$: *There is a statistically significant decrease in the frequency of system disruptions via the DREAM in comparison to the D* Lite algorithm for Free Flight ATM.*

Analysis outputs for hypotheses $\boldsymbol{H_{1A}}$ and $\boldsymbol{H_{1B}}$ are presented in Figures 4-6 and 4-7, respectively, based on Experiment 1 results for the incident rate (as presented in Section 4.1). Each set of analysis outputs summarizes the required calculations for the paired *t*-test procedure as presented in Section 3.3.

Analysis of the Mean Difference in Disruption Frequency – TCAS II vs. DREAM

Paired Difference Calculations:

Mean of paired differences (Eqn. 3.12): $\bar{d} = -2.114$

Std. Dev. of paired differences (Eqn. 3.13): $s_d = 4.202$

Normality Test of Paired Differences:

Critical value of W (n = 20, α = 0.05): $W_{Critical} = 0.905$

Calculated W (Eqn. 3.14): $W = 0.913$

p-value for calculated W: $p\text{–}value = 0.076$

Normal distribution of paired differences? *Yes*

Paired t-testing of the Mean Difference:

Null Hypothesis (H_0): $\mu_d \geq 0$ *Alternate Hypothesis (H_{1A}):* $\mu_d < 0$

Critical value of t-statistic (n = 20, $\alpha_I = 0.025$): $t_\alpha = 2.093$

Calculated t-statistic (Eqn. 3.17): $t = -2.25$

p-value for calculated t: $p - value = 0.018$

Paired- t-test conclusion: *Reject H_0*

95% Confidence Interval on the Mean Difference (Eqn. 3.19):

Lower Bound on μ_d: -4.079 *Upper Bound on μ_d:* -0.145

Figure 4-6: Analysis Outputs for Hypothesis 1A

Analysis of the Mean Difference in Disruption Frequency – D* Lite vs. DREAM

Paired Difference Calculations:

Mean of paired differences (Eqn. 3.12): $\bar{d} = -4.307$

Std. deviation of paired differences (Eqn. 3.13): $s_d = 2.107$

Normality Test of Paired Differences:

Critical value of W (n = 20, α = 0.05): $W_{Critical} = 0.905$

Calculated W (Eqn. 3.14): $W = 0.947$

p-value for calculated W: $p\text{–}value = 0.381$

Normal distribution of paired differences? *Yes*

Paired t-testing of the Mean Difference:

Null Hypothesis (H_0): $\mu_d \geq 0$ *Alternate Hypothesis (H_{1B}):* $\mu_d < 0$

Critical value of t-statistic (n = 20, $\alpha_I = 0.025$): $t_\alpha = 2.093$

Calculated t-statistic (Eqn. 3.17): $t = -9.14$

p-value for calculated t: $p - value < 0.001$

Paired- t-test conclusion: *Reject H_0*

95% Confidence Interval on the Mean Difference (Eqn. 3.19):

Lower Bound on μ_d: -5.294 *Upper Bound on μ_d:* -3.322

Figure 4-7: Analysis Outputs for Hypothesis 1B

The results of the analysis of hypothesis testing associated with Research Question 1 are discussed below and summarized in Table 4-10:

$\boldsymbol{H_{1A}}$: The DREAM is found to provide a statistically significant decrease in the frequency of system disruptions in comparison to TCAS II for Free Flight ATM, averaging 2.11 fewer incidents per hour (corresponding with $t = -2.25$ and a $p - value = 0.018$ for the paired t-test). A 95% CI places the mean incident rate for the DREAM between 0.15 and 4.1 incidents per hour lower than for TCAS II, representing an improvement in system disruption avoidance with practical significance.

$\boldsymbol{H_{1B}}$: The DREAM is found to provide a statistically significant decrease in the frequency of system disruptions in comparison to D* Lite for Free Flight ATM, averaging over 4 fewer incidents per hour (corresponding with $t = -9.14$ and a $p - value < 0.001$ for the paired *t*-test). A 95% CI constructed on the mean difference places the incident rate for the DREAM between 3.3 and 5.3 incidents per hour lower than D* Lite, reflecting a significant improvement in disruption avoidance.

Table 4-10: Summary of Hypothesis Testing for Research Question 1

Hypothesis	Description	Supported?
H_{1A}	There is a statistically significant decrease in the frequency of system disruptions via the DREAM in comparison to TCAS II for Free Flight ATM.	Yes
H_{1B}	There is a statistically significant decrease in the frequency of system disruptions via the DREAM in comparison to the D* Lite algorithm for Free Flight ATM.	Yes

4.3.2 Analysis and Findings for Research Question 2

The second research question is as follows (as outlined in Section 1.4):

> ***Research Question 2:*** *Does the developed model improve mission reliability in emergent operating environments when compared to rule-based and utility-based system automation methods?*

The associated hypotheses for Research Question 2 are as follows (as outlined in Section 1.5):

> $\boldsymbol{H_{2A}}$: *There is a statistically significant decrease in mission reliability achieved via the DREAM in comparison to TCAS II for Free Flight ATM.*
>
> $\boldsymbol{H_{2B}}$: *There is a statistically significant decrease in mission reliability achieved via the DREAM in comparison to the D* Lite algorithm for Free Flight ATM.*

Analysis outputs for hypotheses $\boldsymbol{H_{2A}}$ and $\boldsymbol{H_{2B}}$ are presented in Figures 4-8 and 4-9, respectively, based on Experiment 1 results for mission reliability (as presented in Section 4.1). Each set of analysis outputs summarizes the required calculations for the paired *t*-test procedure as presented in Section 3.3.

Analysis of the Mean Difference in Mission Reliability – TCAS II vs. DREAM

Paired Difference Calculations:

Mean of paired differences (Eqn. 3.12): $\bar{d} = 15.70\%$

Std. Dev. of paired differences (Eqn. 3.13): $s_d = 3.89\%$

Normality Test of Paired Differences:

Critical value of W (n = 20, α = 0.05): $W_{Critical} = 0.905$

Calculated W (Eqn. 3.14): $W = 0.966$

p-value for calculated W: $p\text{–}value = 0.634$

Normal distribution of paired differences? *Yes*

Paired t-testing of the Mean Difference:

Null Hypothesis (H_0): $\mu_d \leq 0$ *Alternate Hypothesis (H_{2A}):* $\mu_d > 0$

Critical value of t-statistic (n = 20, $\alpha_I = 0.025$): $t_\alpha = 2.093$

Calculated t-statistic (Eqn. 3.17): $t = 18.05$

p-value for calculated t: $p - value < 0.001$

Paired- t-test conclusion: *Reject H_0*

95% Confidence Interval on the Mean Difference (Eqn. 3.19):

Lower Bound on μ_d: 13.9% *Upper Bound on μ_d:* 17.5%

Figure 4-8: Analysis Outputs for Hypothesis 2A

Analysis of the Mean Difference in Mission Reliability – D* Lite vs. DREAM

Paired Difference Calculations:

Mean of paired differences (Eqn. 3.12): $\bar{d} = -2.25\%$

Std. Dev. of paired differences (Eqn. 3.13): $s_d = 1.38\%$

Normality Test of Paired Differences:

Critical value of W (n = 20, α = 0.05): $W_{Critical} = 0.905$

Calculated W (Eqn. 3.14): $W = 0.959$

p-value for calculated W: $p\text{–}value = 0.496$

Normal distribution of paired differences? *Yes*

Paired t-testing of the Mean Difference:

Null Hypothesis (H_0): $\mu_d \leq 0$ *Alternate Hypothesis (H_{2B}):* $\mu_d > 0$

Critical value of t-statistic (n = 20, $\alpha_I = 0.025$): $t_\alpha = 2.093$

Calculated t-statistic (Eqn. 3.17): $t = -7.29$

p-value for calculated t: $p - value = 1.000$

Paired- t-test conclusion: *Fail to Reject H_0*

95% Confidence Interval on the Mean Difference (Eqn. 3.19):

Lower Bound on μ_d: -2.89% *Upper Bound on μ_d:* -1.60%

Figure 4-9: Analysis Outputs for Hypothesis 2B

The results of the analysis of hypothesis testing associated with Research Question 3 are discussed below and summarized in Table 4-11:

H_{2A}: The DREAM is found to provide a statistically significant increase in mission reliability in comparison to TCAS II, with an average increase of 15.7% observed (corresponding with $t = 18.05$ and a $p - value < 0.001$ for the paired *t*-test). A 95% CI places the mean difference in mission reliability between 13.8% and 17.5% higher via the DREAM in comparison to TCAS II, representing a significant improvement in system's combined operational availability and mission completion rate.

H_{2B}: The DREAM was not found to provide a statistically significant increase in mission reliability when compared to the D* Lite algorithm. On average the DREAM resulted in 2.2% lower mission reliability than D* Lite given identical test cases (corresponding with $t = -7.29$ and a $p - value = 1.000$ for the paired t-test). However, a 95% confidence interval places the mean difference in mission reliability between 1.6% and 3% lower for the DREAM than D* Lite. In practical terms, however, the observed difference in mission reliability may be negligible, if not outweighed by the benefits expected from the significant reduction in incident rate realized via the DREAM as compared to D* Lite (discussed in Sections 4.1.1 and 4.3.1).

Table 4-11: Summary of Hypothesis Testing for Research Question 2

Hypothesis	Description	Supported?
H_{2A}	There is a statistically significant increase in mission reliability achieved via the DREAM in comparison to TCAS II for Free Flight ATM.	Yes
H_{2B}	There is a statistically significant increase in mission reliability achieved via the DREAM in comparison to D* Lite for Free Flight ATM.	No

4.3.3 Analysis and Findings for Research Question 3

The third research question is as follows (as outlined in Section 1.4):

> ***Research Question 3:*** *Does the developed model improve system efficiency in emergent operating environments when compared to rule-based and utility-based system automation methods?*

The associated hypotheses for Research Question 3 are as follows (as outlined in Section 1.5):

> $\boldsymbol{H_{3A}}$: *There is a statistically significant increase in system efficiency via the DREAM in comparison to TCAS II for Free Flight ATM.*
>
> $\boldsymbol{H_{3B}}$: *System efficiency via the DREAM is reduced, but comparable to the level achieved by D* Lite for Free Flight ATM.*

Analysis outputs for hypotheses $\boldsymbol{H_{3A}}$ and $\boldsymbol{H_{3B}}$ are presented in Figures 4-10 and 4-11, respectively, based on Experiment 2 results for system efficiency (as presented in Section 4.2). Each set of analysis outputs summarizes the required calculations for the paired *t*-test procedure as presented in Section 3.3.

Analysis of the Mean Difference in System Efficiency – TCAS II vs. DREAM

Paired Difference Calculations:

Mean of paired differences (Eqn. 3.12): $\bar{d} = 9.74\%$

Std. Dev. of paired differences (Eqn. 3.13): $s_d = 4.43\%$

Normality Test of Paired Differences:

Critical value of W (n = 20, α = 0.05): $W_{Critical} = 0.905$

Calculated W (Eqn. 3.14): $W = 0.934$

p-value for calculated W: $p\text{–}value = 0.303$

Normal distribution of paired differences? *Yes*

Paired t-testing of the Mean Difference:

Null Hypothesis (H_0): $\mu_d \leq 0$ *Alternate Hypothesis (H_{3A}):* $\mu_d > 0$

Critical value of t-statistic (n = 20, $\alpha_I = 0.025$): $t_\alpha = 2.093$

Calculated t-statistic (Eqn. 3.17): $t = 9.83$

p-value for calculated t: $p - value < 0.001$

Paired- t-test conclusion: *Reject H_0*

95% Confidence Interval on the Mean Difference (Eqn. 3.19):

Lower Bound on μ_d: 7.67% *Upper Bound on μ_d:* 11.81%

Figure 4-10: Analysis Outputs for Hypothesis 3A

Analysis of the Mean Difference in System Efficiency – D* Lite vs. DREAM

Paired Difference Calculations:

Mean of paired differences (Eqn. 3.12): $\bar{d} = -0.92\%$

Std. Dev. of of paired differences (Eqn. 3.13): $s_d = 0.46\%$

Normality Test of Paired Differences:

Critical value of W (n = 20, α = 0.05): $W_{Critical} = 0.905$

Calculated W (Eqn. 3.14): $W = 0.954$

p-value for calculated W: $p\text{–}value = 0.450$

Normal distribution of paired differences? *Yes*

Paired t-testing of the Mean Difference:

Null Hypothesis (H_0): $\mu_d \geq 0$ *Alternate Hypothesis (H_{3B}):* $\mu_d < 0$

Critical value of t-statistic (n = 20, $\alpha_I = 0.025$): $t_\alpha = 2.093$

Calculated t-statistic (Eqn. 3.17): $t = -8.94$

p-value for calculated t: $p - value < 0.001$

Paired- t-test conclusion: *Reject H_0*

95% Confidence Interval on the Mean Difference (Eqn. 3.19):

Lower Bound on μ_d: -1.14% *Upper Bound on μ_d:* -0.71%

Figure 4-11: Analysis Outputs for Hypothesis 3B

The results of hypothesis testing associated with Research Question 3 are discussed below and summarized in Table 4-12:

H_{3A}: The DREAM is found to provide a statistically significant increase in system efficiency in comparison to TCAS II, with an average increase of 9.47% observed (corresponding with $t = 9.83$ and a $p - value < 0.001$ for the paired *t*-test). A 95% CI places the mean difference in system efficiency between 7.7% and 11.8% higher via the DREAM in comparison to TCAS II, representing a significant improvement in system's efficiency given identical operating conditions and constraints.

H_{3B}: The DREAM was found to yield comparable, albeit reduced system efficiency when compared to the D* Lite algorithm. On average the DREAM resulted in 0.92% lower system efficiency the D* Lite given identical test cases (corresponding with $t = -8.94$ and a $p - value < 0.001$ for the paired t-test). A 95% confidence interval places the mean difference in system efficiency between -1.2% and -0.71%. In practical terms, however, the observed difference in system efficiency is negligible, if not outweighed by the benefits expected from the significant reduction in incident rate realized via the DREAM as compared to D* Lite (discussed in Sections 4.1.1 and 4.3.1).

Table 4-12: Summary of Hypothesis Testing for Research Question 3

Hypothesis	Description	Supported?
H_{3A}	There is a statistically significant increase in system efficiency via the DREAM in comparison to TCAS II for Free Flight ATM.	Yes
H_{3B}	System efficiency via the DREAM is reduced, but comparable to the level achieved by D* Lite for Free Flight ATM.	Yes

4.3.4 Analysis and Findings for Research Question 4

The fourth research question is as follows (as outlined in Section 1.4):

> ***Research Question 4:*** *Does the developed model reduce the standard deviation of system efficiency in emergent operating environments when compared to rule-based and utility-based system automation methods?*

The associated hypotheses for Research Question 4 are as follows (as outlined in Section 1.5):

> $\boldsymbol{H_{4A}}$: *There is a statistically significant decrease in the standard deviation of system efficiency via the DREAM in comparison to TCAS II for Free Flight ATM.*
>
> $\boldsymbol{H_{4B}}$: *The standard deviation of system efficiency via the DREAM is increased, but comparable to the level achieved by D* Lite for Free Flight ATM.*

Analysis outputs for hypotheses $\boldsymbol{H_{4A}}$ and $\boldsymbol{H_{4B}}$ are presented in Figures 4-12 and 4-13, respectively, based on Experiment 2 results for the standard deviation of system efficiency (as presented in Section 4.2). Each set of analysis outputs summarizes the required calculations for the paired *t*-test procedure as presented in Section 3.3.

Analysis of the Mean Difference in Std. Deviation of Efficiency – TCAS II vs. DREAM

Paired Difference Calculations:

Mean of paired differences (Eqn. 3.12): $\bar{d} = -1.52\%$

Std. Dev. of paired differences (Eqn. 3.13): $s_d = 0.52\%$

Normality Test of Paired Differences:

Critical value of W (n = 20, α = 0.05): $W_{Critical} = 0.905$

Calculated W (Eqn. 3.14): $W = 0.945$

p-value for calculated W: $p\text{–}value = 0.353$

Normal distribution of paired differences? *Yes*

Paired t-testing of the Mean Difference:

Null Hypothesis (H_0): $\mu_d \geq 0$ *Alternate Hypothesis (H_{4A}):* $\mu_d < 0$

Critical value of t-statistic (n = 20, $\alpha_I = 0.025$): $t_\alpha = 2.093$

Calculated t-statistic (Eqn. 3.17): $t = -13.07$

p-value for calculated t: $p - value < 0.001$

Paired- t-test conclusion: *Reject H_0*

95% Confidence Interval on the Mean Difference (Eqn. 3.18):

Lower Bound on μ_d: -1.76% *Upper Bound on μ_d:* -1.28%

Figure 4-12: Analysis Outputs for Hypothesis 4A

Analysis of the Mean Difference in Std. Deviation of Efficiency – D* Lite vs. DREAM

Paired Difference Calculations:

Mean of paired differences (Eqn. 3.12): $\bar{d} = 0.14\%$

Std. Dev. of paired differences (Eqn. 3.13): $s_d = 0.06\%$

Normality Test of Paired Differences:

Critical value of W (n = 20, α = 0.05): $W_{Critical} = 0.905$

Calculated W (Eqn. 3.14): $W = 0.976$

p-value for calculated W: $p\text{–}value = 0.842$

Normal distribution of paired differences? *Yes*

Paired t-testing of the Mean Difference:

Null Hypothesis (H_0): $\mu_d \leq 0$ *Alternate Hypothesis (H_{4B}):* $\mu_d > 0$

Critical value of t-statistic (n = 20, $\alpha_I = 0.025$): $t_\alpha = 2.093$

Calculated t-statistic (Eqn. 3.17): $t = 10.43$

p-value for calculated t: $p - value < 0.001$

Paired- t-test conclusion: *Reject H_0*

95% Confidence Interval on the Mean Difference (Eqn. 3.18):

Lower Bound on μ_d: 0.11% *Upper Bound on μ_d:* 0.17%

Figure 4-13: Analysis Outputs for Hypothesis 4B

The results of the analysis of hypothesis testing associated with Research Question 4 are discussed below and summarized in Table 4-13:

H_{4A}: The DREAM is found to provide a statistically significant decrease in the standard deviation of system efficiency in comparison to TCAS II, with an average decrease of 1.52% observed (corresponding with $t = -13.07$ and a $p - value < 0.001$ for the paired *t*-test). A 95% CI places the standard deviation of efficiency between 1.3% and 1.8% lower via the DREAM in comparison to TCAS II, representing a practical improvement in stakeholder equity.

H_{4B}: The DREAM was found to yield an comparable, albeit increased standard deviation of system efficiency when compared to the D* Lite algorithm, averaging just 0.14% higher across identical test cases (corresponding with $t = 10.43$ and a $p - value < 0.001$ for the paired t-test). A 95% confidence interval places the mean difference in the standard deviation of efficiency between -0.11% and -0.17% lower for the DREAM than D* Lite, representing a negligible differentiation in stakeholder equity in practical terms.

Table 4-13: Summary of Hypothesis Testing for Research Question 4

Hypothesis	Description	Supported?
H_{4A}	There is a statistically significant decrease in the standard deviation of efficiency via the DREAM in comparison to TCAS II for Free Flight ATM.	Yes
H_{4B}	The standard deviation of system efficiency via the DREAM is increased, but comparable to the level achieved by D* Lite for Free Flight ATM.	Yes

4.3.5 Analysis and Findings for Research Question 5

The fifth and final research question is as follows (as outlined in Section 1.4):

> ***Research Question 5:*** *Does the developed model improve system stability in emergent operating environments when compared to rule-based and utility-based system automation methods?*

The associated hypotheses for Research Question 5 are as follows (as outlined in Section 1.5):

> $\boldsymbol{H_{5A}}$: *System stability via the DREAM is reduced in comparison to TCAS II for Free Flight ATM, but a net decrease in system disruptions is still achieved.*
>
> $\boldsymbol{H_{5B}}$: *There is a statistically significant increase in system stability via the DREAM in comparison to the D* Lite algorithm for Free Flight ATM.*

Analysis outputs for hypotheses $\boldsymbol{H_{5A}}$ and $\boldsymbol{H_{5B}}$ are presented in Figures 4-14 and 4-15, respectively, based on Experiment 2 results for system stability (as presented in Section 4.2). Each set of analysis outputs summarizes the required calculations for the paired *t*-test procedure as presented in Section 3.3.

Analysis of the Mean Difference in System Stability – TCAS II vs. DREAM

Paired Difference Calculations:

Mean of paired differences (Eqn. 3.12): $\bar{d} = -0.27$

Std. Dev. of paired differences (Eqn. 3.13): $s_d = 0.13$

Normality Test of Paired Differences:

Critical value of W (n = 20, α = 0.05): $W_{Critical} = 0.905$

Calculated W (Eqn. 3.14): $W = 0.956$

p-value for calculated W: $p\text{–}value = 0.474$

Normal distribution of paired differences? *Yes*

Paired t-testing of the Mean Difference:

Null Hypothesis (H_0): $\mu_d \geq 0$ *Alternate Hypothesis (H_{4A}):* $\mu_d < 0$

Critical value of t-statistic (n = 20, $\alpha_I = 0.025$): $t_\alpha = 2.093$

Calculated t-statistic (Eqn. 3.17): $t = -9.29$

p-value for calculated t: $p - value < 0.001$

Paired- t-test conclusion: *Reject H_0*

95% Confidence Interval on the Mean Difference (Eqn. 3.19):

Lower Bound on μ_d: -0.21% *Upper Bound on μ_d:* -0.33%

Figure 4-14: Analysis Outputs for Hypothesis 5A

Analysis of the Mean Difference in System Stability – D* Lite vs. DREAM

Paired Difference Calculations:

Mean of paired differences (Eqn. 3.12): $\bar{d} = 0.07$

Std. Dev. of paired differences (Eqn. 3.13): $s_d = 0.03$

Normality Test of Paired Differences:

Critical value of W (n = 20, α = 0.05): $W_{Critical} = 0.905$

Calculated W (Eqn. 3.14): $W = 0.958$

p-value for calculated W: $p\text{–}value = 0.490$

Normal distribution of paired differences? *Yes*

Paired t-testing of the Mean Difference:

Null Hypothesis (H_0): $\mu_d \leq 0$ *Alternate Hypothesis (H_{4B}):* $\mu_d > 0$

Critical value of t-statistic (n = 20, $\alpha_I = 0.025$): $t_\alpha = 2.093$

Calculated t-statistic (Eqn. 3.17): $t = 10.43$

p-value for calculated t: $p - value < 0.001$

Paired- t-test conclusion: *Reject H_0*

95% Confidence Interval on the Mean Difference (Eqn. 3.19):

Lower Bound on μ_d: 0.06 *Upper Bound on μ_d:* 0.08

Figure 4-15: Analysis Outputs for Hypothesis 5B

The results of the analysis of hypothesis testing associated with Research Question 5 are discussed below and summarized in Table 4-14:

$\boldsymbol{H_{5A}}$: The DREAM and TCAS II are both found to yield a net decrease in the number of system disruptions experienced. However, the DREAM results in a lower level of system stability than TCAS II, with an average difference of 0.27 observed (corresponding with $t = -9.29$ and a $p - value < 0.001$ for the paired *t*-test). A 95% CI places the mean difference of system stability between 0.21 and 0.33 lower via the DREAM in comparison to TCAS II, representing a significant difference in efficacy with respect to the mitigation of the domino effect (as discussed in Section 3.2.2.5). In practical terms, however, experiment results reflect that both the DREAM and TCAS II achieved a stabilizing effect ($SS \geq 1$) which reduced the total number of system disruptions observed as a result of preventive and corrective action.

$\boldsymbol{H_{5B}}$: The DREAM was found to yield a statistically significant increase in system stability when compared to the D* Lite algorithm, averaging 0.07 higher across identical test cases (corresponding with $t = 10.43$ and a $p - value < 0.001$ for the paired t-test). A 95% confidence interval places the mean difference in system stability between 0.06 and 0.08 lower for the DREAM than D* Lite, representing a small differentiation in quantitative terms. However, experiment results reflect that the DREAM achieved a stabilizing effect ($SS \geq 1$) whereas the D* Lite resulted in a destabilizing effect given identical scenarios.

Table 4-14: Summary of Hypothesis Testing for Research Question 5

Hypothesis	Description	Supported?
H_{5A}	System stability via the DREAM is reduced in comparison to TCAS II for Free Flight ATM, but a net decrease in system disruptions is still achieved.	Yes
H_{5B}	There is a statistically significant increase in system stability via the DREAM in comparison to D* Lite for Free Flight ATM.	Yes

4.3.6 Analysis Summary

The results of hypothesis testing for all research questions are summarized in Table 4-15. The implications of these findings for application of the DREAM, limitations of the research, and recommended areas for future research are discussed in Chapter 5.

Table 4-15: Summary of Results for Research Hypotheses

Hypothesis	Description	Supported?
H_{1A}	There is a statistically significant decrease in the frequency of system disruptions via the DREAM in comparison to TCAS II for Free Flight ATM.	Yes
H_{1B}	There is a statistically significant decrease in the frequency of system disruptions via the DREAM in comparison to the D* Lite algorithm for Free Flight ATM.	Yes
H_{2A}	There is a statistically significant increase in mission reliability achieved via the DREAM in comparison to TCAS II for Free Flight ATM.	Yes
H_{2B}	There is a statistically significant increase in mission reliability achieved via the DREAM in comparison to D* Lite for Free Flight ATM.	No
H_{3A}	There is a statistically significant increase in system efficiency via the DREAM in comparison to TCAS II for Free Flight ATM.	Yes
H_{3B}	System efficiency via the DREAM is reduced, but comparable to the level achieved by D* Lite for Free Flight ATM.	Yes*
H_{4A}	There is a statistically significant decrease in the standard deviation of system efficiency via the DREAM in comparison to TCAS II for Free Flight ATM.	Yes
H_{4B}	The standard deviation of system efficiency via the DREAM is increased, but comparable to the level achieved by D* Lite for Free Flight ATM.	Yes*
H_{5A}	System stability via the DREAM is reduced in comparison to TCAS II for Free Flight ATM, but a net decrease in system disruptions is still achieved.	Yes*
H_{5B}	There is a statistically significant increase in system stability via the DREAM in comparison to D* Lite for Free Flight ATM.	Yes

*Despite a statistically significant difference in performance, the mean difference and a 95% CI on the mean difference are representative of comparable performance in practical terms.

Chapter 5: Conclusion

The need for resilient systems which are capable of avoiding, withstanding, adapting to and recovering from disruptions continues to grow given ever increasing uncertainty surrounding natural and man-made operating environments [INCOSE, 2014, 2015; Madni & Jackson, 2009]. This research focused on the mitigation of system disruptions due to: (a) the lack of complete anticipatory knowledge of threats and vulnerabilities a system will encounter; (b) the prevention of human involvement for system operation or as an operational contingency; and (c) resource contention amongst multiple systems or processes. The *Disruption Resilience and Adaptation Model (DREAM)* was developed as an adaptive and autonomous system command and control model for engineered systems subject to emergent operating environments and constraints. This chapter summarizes the DREAM's capabilities and benefits, limitations of the research, areas of future research and concludes with a discussion on the practical implications of this research. The chapter is outlined as follows:

- *Section 5.1* – Summary of DREAM Capabilities and Benefits
- *Section 5.2* – Research Application and Findings
- *Section 5.3* – Limitations of Research
- *Section 5.4* – Areas of Future Research
- *Section 5.5* – Conclusion

5.1 Summary of DREAM Capabilities and Benefits

The DREAM integrates multiple tools and countermeasures for system disruptions that cannot be precisely anticipated and/or prevented, including:

- *Resilience Quantification & Assessment Metrics:* diagnostic and prognostic resilience metrics based on system availability are used to continuously monitor and adapt system behavior in lieu of the ability to anticipate and avoid disruptions. By using system availability as an indirect indicator of disruption rather than detection of a specific event or condition, the need for preventive or corrective action can be identified even without anticipatory knowledge of disruption;

- *Autonomous Disruption Avoidance & Mitigation Procedures*: to mitigate the risks associated with relying on human control or back-ups to engineered systems, the DREAM is designed for autonomous identification and selection of preventive and corrective actions in response to unanticipated disruptions. Automation is enabled via an autonomic control process, enabling system self-monitoring and self-regulation to satisfice mission requirements without a human involvement. The use of autonomic control for automation contrasts with the prevalence of automated control processed which are limited to the execution of preplanned actions and responses without the ability to deviate;

- *Support for Dynamic Resource Allocation:* coopetition behavior is enforced in operational environments that require limited resources to be shared amongst multiple systems. Reinforcement of coopetition behavior is enabled by modulating the system's tolerance for resource disruption based on its operational availability. Coopetition behavior is achieved autonomously through system self-monitoring and self-regulation without the need for

preplanned deconfliction or centralized command and control over shared resources.

The capabilities and benefits of the DREAM were demonstrated and evaluated on the basis of an individual system of interest and from the system of systems perspective, as summarized in the following section.

5.2 Research Application and Findings

Application of the DREAM to the Free Flight concept of operations for Air Traffic Management (ATM) was performed via modeling and simulation, as presented in Section 3.2. Two Highly Accelerated Stress Screening (HASS) experiments were performed to assess ATM system resilience to disruptions and the degradation of operational efficiency from the perspective of the individual aircraft (representing a single system of interest) and multiple aircraft operating in shared airspace (representing a system of systems). HASS experiments enabled the performance of the DREAM to be compared with the TCAS II standard for aircraft collision avoidance and the D* Lite algorithm which is the predominant method for autonomous vehicle navigation in uncertain environments.

From the perspective of an individual aircraft, HASS of ATM resilience focused on assessing the system incident rate (λ) and mission reliability (R_M) as defined in Sections 3.2.2.1 and 3.2.2.2 respectively. Simulation results and analysis demonstrate that the DREAM concurrently reduced the incident rate and increased mission reliability in comparison to the TCAS II standard for aircraft collision avoidance. As compared to the D* Lite algorithm, a reduction of the system incident rate was achieved, with comparable mission reliability observed for identical test cases.

From the system of systems perspective, HASS of ATM assessed system efficiency (E), the standard deviation of efficiency (D_E), and system stability (SS) as defined in Sections 3.2.2.3 through 3.2.2.5. Results and analysis demonstrate that the DREAM improved the resilience and equity amongst aircraft in the ATM system in comparison to the TCAS II standard, with increased system efficiency, a reduction in the standard deviation of efficiency, and comparable system stability achieved as the demand for finite airspace increased. In comparison to D* Lite, the DREAM provided improved system stability while achieving comparable system efficiency and standard deviation of efficiency.

5.3 Limitations

As noted by Box [1979], "all models are wrong, but some are useful". In other words, it's important to acknowledge that models are a "simplification or approximation of reality and hence will not reflect all of reality" [Burnham & Anderson, 2003]. The following caveats and limitations apply to the modeling and simulation performed in this research. Critically, ATM modeling did not include all potential failure modes and types of performance degradation that can arise from emergent operational environment. Rather, a subset of ATM system failure modes and susceptibilities related to operational safety, efficacy, and efficiency were used to demonstrate the concept. Additionally, modeling was limited in scope to the enroute phase of flight, encompassing aircraft routing and collision avoidance functions of ATM systems. Aircraft ground operations and transitions (from air-to-ground and ground-to-air) were excluded from this study for model simplification. While these operational scenarios represent a small proportion of overall system operating time compared to the enroute flight phase, they involve unique

and potentially substantive risks for ATM disruption that are not captured in this study, (e.g., runway closures due to foreign object debris). Furthermore, naturally-occurring sources of disruption were not considered in this study. This is a significant caveat, given the wide-ranging scale and variety of effects of natural phenomena that can disrupt ATM systems, such as the weather, bird strikes, and electromagnetic interference of aircraft sensors due to solar-radiation. While the inclusion of natural-phenomena would have increased the fidelity of the ATM model, implementation would have increased complexity of the modelling effort and exceeded the objectives that bounded this research. Subsequently, it is unknown if the results of this study are applicable to naturally occurring ATM disruptions or to scenarios involving a mixture of natural and man-made sources of disruption.

5.4 Areas of Future Research

While the DREAM is notionally extensible to engineered systems and processes other than ATM, future research is needed to validate the extensibility of the model for other systems. Prospective research questions surrounding model extensibility include, but are not limited to, the following:

- What are the implications on model implementation and performance for systems with centralized vs. distributed command and control processes?
- Are the diagnostic and prognostic metrics developed for assessing system resilience effective when system efficacy and efficiency are based on discrete performance milestones (rather than continuous targets or thresholds)?
- How scalable is the model with respect to its application in the system of systems context?

Furthermore, validation of model extensibility can be performed beyond the scope of safety- and security-critical systems and processes such as CIKR. For example, the model could be applied for disruption mitigation in engineered systems used for customer-service or business management applications that focus on customer satisfaction or responsiveness to market demand as critical functions.

The limitations of this research (presented in Section 5.3) also present several opportunities to expand and/or improve upon the body of knowledge surrounding resilience engineering and Free Flight ATM. With respect to modeling and simulation of emergent operating environments, the ATM model developed for this research can be expanded to incorporate the following use cases and potential sources of disruption: (i) ground-operations phase of flight, such as aircraft taxing and runway maneuvers; (ii) transitional phases of flight including take-off, landing, and holding maneuvers surrounding airports; (iii) naturally-occurring phenomenon, such as meteorological conditions or natural terrain that may pose a collision risk; (iv) man-made obstacles other than aircraft, such as buildings and restricted use airspace; and (v) failure or degraded performance of the aircraft's flight critical equipment or sub-subsystems, such as the airframe, engines or avionic instruments.

While the existing ATM model is implemented in two-dimensional space, implementing the model in three dimensions or more would increase model's physical fidelity with respect to aircraft maneuvering for routing and collision-avoidance, (e.g., adjustment of altitude, roll angle or pitch angle or combinations thereof with respect to ground). Likewise, the scope of HASS testing for ATM could be expanded by modeling additional administrative or environmental constraints in the simulated airspace. Namely,

the introduction of special use airspace, such as no fly zones and restricted airspace, could be used assess ATM resilience to emergent changes in the boundaries of navigable airspace.

Other prospective areas of research surround the relationship between system resilience, system sustainment tasks and the retirement phase of the systems lifecycle, such as following questions:

- How does system resilience impact the decommissioning, dismantlement, disposal of engineered systems when intended?
- What criteria should be used in tradeoffs between system availability in the short-term versus survivability in the long-term? In other words, what type of decision rationality should be applied when a mutually exclusive choice between a temporary and permanent loss of a mission essential functions is required.
- How can resilient system architecture and design support the reuse or reapplication of engineered systems in-lieu of system disposal?

Examining these prospective research thrusts and key questions complement the contributions of this research towards the development and sustainment of resilient systems.

5.5 Conclusion

Autonomous functionality continues to become more commonplace for performance monitoring and optimization in engineered systems. This research provides an autonomous command and control model that provides countermeasures for

unanticipated disruptions that jeopardize intended system functionality. The research confirms the model can concurrently reduce the frequency and severity of system disruptions in comparison to existing rule-based automation methods that rely on anticipatory knowledge of disruptions. The model can be applied as part of a risk mitigation and system resilience strategy for disruption avoidance, withstanding, adaptation and recovery for engineered systems subject to emergent operating environments.

While the DREAM was applied and demonstrated for Free Flight ATM in this research, the metrics and algorithms developed are extensible to other engineered systems that are susceptible to unforeseeable and/or unavoidable disruptions, such as the power grid and public transportation systems. Adaptation of the model is enabled by the ability to tailor system availability metrics and parameters to a specific problem or system of interest, including the specification of: (i) uptime and downtime definitions for operational availability; (ii) estimated uptime and estimated downtime calculations for forecasted availability; and (iii) the target availability level for the system.

The research presented here addresses the growing need for risk management and system resilience strategies beyond the ideal scenario of avoiding disruptions through anticipation or relying on human intervention as an operational contingency. As the ability to anticipate and avoid disruptions is overcome by emergent needs and constraints, the DREAM demonstrates an adaptive command and control capability that can enhance the resilience of engineered systems.

References

Airbus S.A.S. (2017). Airbus Family Figures. Retrieved from http://www.airbus.com/content/dam/corporate-topics/publications/backgrounders/Airbus-Family-Figures-booklet.pdf. Site accessed on September 18, 2017.

Ackoff, R. L. (1971). Towards a system of systems concepts. *Management Science*, 17 (11), 661-671.

Allen, C., Wallach, W., & Smit, I. (2006). Why machine ethics? *IEEE Intelligent Systems*, *21*(4), pp. 12-17.

Allspaw, J. (2012). Fault injection in production. *Communications of the ACM*, *55*(10), pp. 48-52.

Anderson, M., & Anderson, S. L. (2007). The status of machine ethics: a report from the AAAI Symposium. *Minds and Machines*, *17*(1), pp. 1-10.

Archibald, J. K., Hill, J. C., Jepsen, N. A., Stirling, W. C., & Frost, R. L. (2008). A satisficing approach to aircraft conflict resolution. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, *38*(4), pp. 510-521.

Arkin, R. C., Ulam, P., & Wagner, A. R. (2012). Moral decision making in autonomous systems: Enforcement, moral emotions, dignity, trust, and deception. *Proceedings of the IEEE*, *100*(3), pp. 571-589.

Arnold, R. D., & Wade, J. P. (2015). A definition of systems thinking: a systems approach. *Procedia Computer Science*, *44*, pp. 669-678.

Arsenault, D., & Sood, A. (2007). Resilience: a systems design imperative. *Critical Thinking: Moving from Infrastructure Protection to Infrastructure Resilience*, pp. 87-95.

Barnard, A. (2012). Ten things you should know about HALT & HASS. In *Proceedings of the 2012 Annual Reliability and Maintainability Symposium* (pp. 1-6). IEEE.

Bell, D., & Gana, L. (2012). Algorithmic trading systems: A multifaceted view of adoption. *2012 45th Hawaii International Conference on System Science (HICSS)*, IEEE, pp. 3090-3099.

Bettini, C., Brdiczka, O., Henricksen, K., Indulska, J., Nicklas, D., Ranganathan, A., & Riboni, D. (2010). A survey of context modelling and reasoning techniques. *Pervasive and Mobile Computing*, *6*(2), pp. 161-180.

Blanchard, B.S. (2004). *Logistics Engineering and Management*, 6th ed. Prentice Hall: Saddle River, NJ, pp. 72-73.

Boeing Corporation (Boeing). (2017). Next Generation 737 Technical Specs. Available: http://www.boeing.com/commercial/737ng/#/technical-specs. Site accessed September 18, 2017.

Boeing Corp. (Boeing). (2017). 737 MAX Technical Specs. Available: http://www.boeing.com/commercial/737max/#/technical-specs. Site accessed September 18, 2017.

Box, G. E. (1979). Robustness in the strategy of scientific model building. *Robustness in statistics*. R. Launer and G. Wilkinson (eds.), Academic Press, pp. 201–236.

Buede, D. And Miller, W. (2016). The engineering design of systems: models and methods. Wiley, pp. 295-297 .

Burke, E., Hyde, M., Kendall, G., Ochoa, G., Özcan, E., & Woodward, J. R. (2010). A classification of hyper-heuristic approaches. *Handbook of metaheuristics,* Springer: Boston, MA, pp. 449-468.

Burke, E., Gendreau, M., Hyde, M., Kendall, G., Ochoa, G., Özcan, E., & Qu, R. (2013). Hyper-heuristics: A survey of the state of the art. *Journal of the Operational Research Society*, *64*(12), pp. 1695-1724.

Burke, E., Kendall, G., Newall, J., Hart, E., Ross, P., & Schulenburg, S. (2003). Hyper-heuristics: An emerging direction in modern search technology. *Handbook of metaheuristics*, Springer: Boston, MA, pp. 457-474.

Burnham, K. P., & Anderson, D. R. (2003). *Model selection and multimodel inference: a practical information-theoretic approach*. Springer-Verlag, pp. 20 – 23.

Capano, G., & Woo, J. J. (2018). Designing policy robustness: Outputs and processes. Policy and society, 37(4), 422-440.

Cook, A., Blom, H. A., Lillo, F., Mantegna, R. N., Micciche, S., Rivas, D., ... & Zanin, M. (2015). Applying complexity science to air traffic management. *Journal of Air Transport Management*, *42*, 149-158.

Dagnino, G. B. (2009). Coopetition strategy: a new kind of interfirm dynamics for value creation. *Coopetition strategy*, G.B. Dagnino and E. Rocco (eds.), Routledge, pp. 45-63.

Daquila, M., Schubmehl, D., Anand, N., Bisht, A., Kuban, M., Membrila, R., Glennon, M., & Iwamoto, N. (2017). *Worldwide Semiannual Cognitive/Artificial Intelligence Systems Spending Guide Taxonomy*, International Data Corporation (IDC), Framingham, MA. Retrieved from

https://www.idc.com/getdoc.jsp?containerId=US43607318. Site accessed on February 16, 2018.

Debreu, G. (1954). Representation of a preference ordering by a numerical function. Decision processes, 3, 159-165.

Dekker, S. (2004). *Ten questions about human error: A new view of human factors and system safety*. CRC Press.

De Weck, O. L., Ross, A. M., & Rhodes, D. H. (2012). Investigating relationships and semantic sets amongst system lifecycle properties (ilities). MIT: Cambridge, MA. (Unpublished manuscript). Retrieved from https://dspace.mit.edu/handle/1721.1/102927. Site accessed on January 5, 2018.

De Weck, O. L., Roos, D., & Magee, C. L. (2011). *Engineering systems: Meeting human needs in a complex technological world*. MIT Press: Cambridge, MA, pp. 69 – 96.

Dill, K., Pursel, E. R., Garrity, P., Fragomeni, G., & Quantico, V. (2012). Design patterns for the configuration of utility-based AI. In Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC), pp. 1-12.

Dorf, R., & Bishop, R. (2011). Modern control systems. Pearson., pp. 5-48.

Endsley, M. R., & Kiris, E. O. (1995). The out-of-the-loop performance problem and level of control in automation. *Human factors*, *37*(2), pp. 381-394.

Entacher, K. (1997). *A collection of selected pseudorandom number generators with linear structures*. Austrian Center for Parallel Computation. (Unpublished paper). Retrieved from http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.53.3686. Site accessed on January 10, 2018.

Espinosa, H. E. P., & Ayala-Solares, J. R. (2016). The power of natural inspiration in control systems. *Nature-Inspired Computing for Control Systems,* Springer: Cham, pp. 1-10.

European Organization for the Safety of Air Navigation /Federal Aviation Administration (EUROCONTROL/FAA). (2006). Communications Operating concept and Requirements for the Future Radio System, COCR Ver. 1.0. Retrieved from https://faaco.faa.gov/index.cfm/attachment/download/7844. Site accessed on October 6, 2017.

Federal Aviation Administration (FAA). (2017a). Aircraft Characteristics Database. Retrieved from https://www.faa.gov/airports/engineering/aircraft_char_database/. Site accessed on October 6, 2017.

Federal Aviation Administration (FAA). (2017b) Aeronautical Information Manual (AIM). Washington, D.C. Retrieved from https://www.faa.gov/air_traffic/publications/media/AIM_Basic_dtd_10-12-17.pdf. Site accessed on November 10, 2017.

Federal Aviation Administration (FAA). (2012). Integration of Civil Unmanned Aircraft Systems (UAS) in the National Airspace System (NAS) Roadmap (2012-AJG-502), FAA: Washington, D.C. Retrieved from https://www.faa.gov/uas/media/uas_roadmap_2013.pdf. Site accessed on November 10, 2017.

Federal Aviation Administration (FAA). (2016). Aerospace Forecast: Fiscal Years 2016-2036 (TC16-0002), Washington, D.C., 2016. Retrieved from https://www.faa.gov/data_research/aviation/aerospace_forecasts/. Site accessed on November 10, 2017.

Federal Aviation Administration (FAA). (2000). System safety handbook. U.S. Dept. of Transportation: Washington, DC: Retrieved from https://www.faa.gov/regulations_policies/handbooks_manuals/aviation/risk_management/ss_handbook/. Site accessed on November 10, 2017.

Federal Aviation Administration (FAA). (2009).Traffic Flow Management in the National Airspace System. U.S. Dept. of Transportation: Washington, DC: Retrieved from https://www.fly.faa.gov/Products/Training/Traffic_Management_for_Pilots/TFM_in_the_NAS_Booklet_ca10.pdf. Site accessed on November 10, 2017.

Ferguson, D., Likhachev, M., & Stentz, A. (2005). A guide to heuristic-based path planning. *Proceedings of the international workshop on planning under uncertainty for autonomous systems, international conference on automated planning and scheduling (ICAPS)*, pp. 9-18.

Goldwater, S., Jurafsky, D., & Manning, C. D. (2010). Which words are hard to recognize? Prosodic, lexical, and disfluency factors that increase speech recognition error rates. *Speech Communication*, *52*(3), pp. 181-200.

Goldratt, E. M., & Cox, J. (1984). *The Goal: A Process of Ongoing Improvement*. North River Press.

Goldratt, E.M. (1990) *What is this thing called the Theory of Constraints and how is it implemented?* North River Press.

Gianetto, D. A., & Heydari, B. (2015). Catalysts of cooperation in system of systems: the role of diversity and network structure. *IEEE Systems Journal*, *9*(1), pp. 303-311.

Glover, F. (1989). Tabu search—part I. *ORSA Journal on computing*, *1*(3), pp. 190-206.

Gomes, C. P., Kautz, H., Sabharwal, A., & Selman, B. (2008). Satisfiability solvers. *Foundations of Artificial Intelligence*, 3, pp. 89-134.

Government Accountability Office (GAO). (1998). National Airspace System: FAA Has Implemented Some Free Flight Initiatives, but Challenges Remain, GAO/RCED-98-246. U.S. Government Printing Office, Washington, D.C.

Government Accountability Office (GAO). (2001). National Airspace System: Free Flight Tools Show Promise, but Implementation Challenges Remain, GAO-01-932. U.S. Government Printing Office, Washington, D.C.

Haimes, Y. Y. (2012). Modeling complex systems of systems with phantom system models. *Systems Engineering*, *15*(3), pp. 333-346.

Hayes-Roth, F. (1985). Rule-based systems. *Communications of the ACM*, *28*(9), pp. 921-932.

Hemme, K. (2015). Critical infrastructure protection: Maintenance is national security. *Journal of Strategic Security*, *8*(3), pp. 25-39.

Henriksen, S. (2008). Unmanned aircraft system control and ATC communications bandwidth requirements (NASA/CR-2008-214841), NASA: Hanover, MD. Retrieved from https://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/20080010686.pdf. Site accessed December 15, 2017.

Hollnagel, E. (1998). *Cognitive reliability and error analysis method*. Elsevier : Oxford, UK, pp. 164 – 165.

Hosseini, S., Barker, K., & Ramirez-Marquez, J. E. (2016). A review of definitions and measures of system resilience. *Reliability Engineering & System Safety*, *145*, pp. 47-61.

International Civil Aviation Organization (ICAO). (2016). 2016 – 2030 Global Air Navigation Plan (Doc 9750-AN/963). Retrieved from https://www.icao.int/

airnavigation/ Documents/GANP-2016-interactive.pdf. Site accessed December 13, 2017.

International Council on Systems Engineering (INCOSE). (2014). *A world in motion: systems engineering vision 2025.* INCOSE: San Diego, CA, pp. 28-35.

Ingrand F, Ghallab M. Deliberation for Autonomous Robots: A Survey. *Artificial Intelligence*. 2017; 247: 10-44.

International Council on Systems Engineering (INCOSE). (2015) *Systems Engineering Handbook (4.0),* INCOSE: Hoboken, NJ, pp. 229-260.

Jackson, S. (2009). *Architecting resilient systems: Accident avoidance and survival and recovery from disruptions*, 66, Wiley: Hoboken, NJ, pp. 174

Jackson, S., & Ferris, T. L. (2013). Resilience principles for engineered systems. *Systems Engineering*, *16*(2), pp. 152-164.

Kalra, N., & Paddock, S. M. (2016). Driving to safety: How many miles of driving would it take to demonstrate autonomous vehicle reliability?. *Transportation Research Part A: Policy and Practice*, *94*, pp. 182-193.

Kephart, J. O., & Chess, D. M. (2003). The vision of autonomic computing. *Computer*, (1), pp. 41-50.

Kochenderfer, M. J., M. Edwards, M. W., Espindle, L. P., Kuchar, J. K., & Griffith, J. D. (2010). Airspace encounter models for estimating collision risk. *Journal of Guidance, Control, and Dynamics*, *33*(2), pp. 487-499.

Koenig, S., & Likhachev, M. (2005). Fast replanning for navigation in unknown terrain. *IEEE Transactions on Robotics*, *21*(3), pp. 354-363.

Kortenkamp, D., Bonasso, R. P., & Murphy, R. (1998). *Artificial intelligence and mobile robots: case studies of successful robot systems*. MIT Press: Cambridge, MA.

Krozel, J., & Peters, M. (1997). Strategic conflict detection and resolution for free flight. In *Proceedings of the 36th IEEE Conference on Decision and Control* (Vol. 2), pp. 1822-1828). IEEE.

Krozel, J., & Peters, M. (1997). Conflict detection and resolution for free flight. *Air Traffic Control Quarterly*, 5(3), pp. 181-212.

Krozel, J., Peters, M., Bilimoria, K. D., Lee, C., & Mitchell, J. S. (2001). System performance characteristics of centralized and decentralized air traffic separation strategies. *Air Traffic Control Quarterly*, *9*(4), pp. 311-332.

Krupitzer, C., Roth, F. M., VanSyckel, S., Schiele, G., & Becker, C. (2015). A survey on engineering approaches for self-adaptive systems. *Pervasive and Mobile Computing*, *17*, pp. 184-206.

Lalanda, P., Diaconescu, A., & McCann, J. (2013). *Autonomic Computing: Principles, Design and Implementation*. Springer, pp. 109-120.

L'ecuyer, P. (1999). Good parameters and implementations for combined multiple recursive random number generators. *Operations Research*, *47*(1), pp. 159-164.

L'Ecuyer, P. (2001). Software for uniform random number generation: Distinguishing the good and the bad. *Proceedings of the 33rd Winter Simulation* Conference, IEEE Computer Society: Washington, D.C., pp. 95-105.

L'Ecuyer, P., & Simard, R. (2007). TestU01: A C library for empirical testing of random number generators. *ACM Transactions on Mathematical Software (TOMS)*, *33*(4), 22.

Madni, A. M. (2007). Designing for resilience. *ISTI Lecture Notes on Advanced Topics in Systems Engineering*.

Madni, A. M., Brenner, M. A., Costea, I., MacGregor, D., & Meshkinpour, F. (1985). Option generation: Problems, principles, and computer-based aiding. *Proceedings of*

the 1985 International Conference on Systems, Man, and Cybernetics. IEEE. pp. 757-760.

Madni, A. M., & Jackson, S. (2009). Towards a conceptual framework for resilience engineering. *IEEE Systems Journal*, *3*(2), pp. 181-191.

Marsaglia, G. (1968). Random numbers fall mainly in the planes. *Proceedings of the National Academy of Sciences*, *61*(1), pp. 25-28.

Marshall, C. J., Roberts, B., & Grenn, M. (2018a). Adaptive and Automated Reasoning for Autonomous System Resilience in Uncertain Worlds. *Disciplinary Convergence in Systems Engineering Research.* Springer: Cham, pp. 799-812.

Marshall, C. J., Roberts, B., & Grenn, M. (2017b). Intelligent control & supervision for autonomous system resilience in uncertain worlds. *Proceedings of the 3rd International Conference on Control, Automation and Robotics (ICCAR),* IEEE, pp. 438-443.

Marshall, C. J., Roberts, B., & Grenn, M. (2018). Context-driven autonomy for enhanced system resilience in emergent operating environments. *IEEE Systems Journal*.

Marshall, C. J., Roberts, B., Grenn, M & Holzer, T. (2019). Adaptive cooperation and competition for autonomous system resilience. (Manuscript submitted for publication)

Maschler M, Solan E, Zamir S. *Game Theory*. Cambridge University Press; 2013:176–180.

MathWorks. (2017). Creating and Controlling a Random Number Stream. Retrieved from https://www.mathworks.com/help/matlab/math/creating-and-controlling-a-random-number-stream.html#brvku_2. Site accessed on December 20, 2017.

MathWorks. (2016). MATLAB and Statistics Toolbox Release 2016b [Computer software], Natick, Massachusetts: The MathWorks Inc.

Matsumoto, M., & Nishimura, T. (1998). Mersenne twister: A 623-dimensionally equidistributed uniform pseudo-random number generator. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, *8*(1), pp. 3-30.

McCarthy, J. What is artificial intelligence? Technical report, Stanford University, 2007. http://www-formal.stanford.edu/jmc/whatisai.pdf

McLean, H. W. (2009). HALT, HASS, and HASA explained: accelerated reliability techniques. ASQ Quality press.

Michalski, R. S., Carbonell, J. G., & Mitchell, T. M. (Eds.). (2013). Machine learning: An artificial intelligence approach. Springer Science & Business Media.

Milliken, F. J. (1987). Three types of perceived uncertainty about the environment: State, effect, and response uncertainty. Academy of Management review, 12(1), 133-143.

Minitab, Inc (Minitab). (2017). *Minitab 18 Statistical Software*. [Computer software], Minitab: State College, PA.

Mintzberg, H. (1994). Rethinking strategic planning part I: Pitfalls and fallacies. *Long Range Planning*, *27*(3), pp. 12-21.

Moore, P., Hu, B., & Jackson, M. (2011). Rule strategies for intelligent context-aware systems: The application of conditional relationships in decision-support. 2011 International Conference on Complex, Intelligent and Software Intensive Systems (CISIS), pp. 9-16. IEEE.

Morgan, M. G. (2001). Carnegie Mellon's department of engineering and public policy. *International Journal of Technology, Policy and Management*, *1*(2), pp. 138-150.

Moteff, J., & Parfomak, P. (2004). Critical infrastructure and key assets: definition and identification. Congressional Research Service, Washington, D.C. Retrieved from https://fas.org/sgp/crs/RL32631.pdf. Site accessed January 10, 2018.

Natella, R., Cotroneo, D., & Madeira, H. S. (2016). Assessing dependability with software fault injection: A survey. ACM Computing Surveys (CSUR), 48(3), 44.

Neufville, R. D. (2001). The emerging curriculum for engineering. *International Journal of Technology, Policy and Management*, *1*(1), pp. 117-127.

Newman, M. E. (2005). Power laws, Pareto distributions and Zipf's law. *Contemporary physics*, *46*(5), pp. 323-351.

Nof, S., (2009). Industrial Automation. *Springer Handbook of Automation*, Springer, pp. 837-1058.

Obolensky, M. A. B., Erman, A. E., Rozenberg, J., Maruyama Rentschler, J. E., Avner, P., & Hallegatte, S. (2019). *Infrastructure Disruptions: How Instability Breeds Household Vulnerability* (No. 8902). The World Bank.

O'Rourke, T. D. (2007). Critical infrastructure, interdependencies, and resilience. *The BRIDGE*, *37*(1), pp. 22-29.

Oracle. (2017). Java Platform Standard Edition 8 Documentation – Class Random. Retrieved from https://docs.oracle.com/javase/8/docs/api/java/util/Random.html. Site accessed on December 20, 2017.

Ott, R. L., & Longnecker, M. T. (2015). *An introduction to statistical methods and data analysis*. Nelson Education: Boston, MA. pp. 325 – 329.

Ouedraogo, K. A., Enjalbert, S., & Vanderhaegen, F. (2013). How to learn from the resilience of Human–Machine Systems? *Engineering Applications of Artificial Intelligence*, 26(1), pp. 24-34.

Pachter, M., & Chandler, P. R. (1998). Challenges of autonomous control. IEEE Control Systems, 18(4), 92-97.

Passino, K. M. (1995). Intelligent control for autonomous systems. IEEE Spectrum, 32(6), 55-62.

Polacek GA, Gianetto DA, Khashanah K, Verma D. On Principles and Rules in Complex Adaptive Systems: A Financial System Case Study. *Systems Engineering*. 2012;15(4):433-447.

Poongodai, A. & Bhuvaneswari, S. (2013). AI technique in diagnostics and prognostics. *International Journal of Computer Applications*, pp. 1 - 4.

Powell, W. B. (2007). *Approximate Dynamic Programming: Solving the curses of dimensionality*, Wiley-Interscience: Hoboken, N.J., pp. 387-432.

Project Management Institute [PMI]. (2004). *A Guide to the Project Management Body of Knowledge (PMBoK)*. PMI: Newtown Square, PA, pp. 303-309.

Raj, R., Wang, J. W., Nayak, A., Tiwari, M. K., Han, B., Liu, C. L., & Zhang, W. J. (2015). Measuring the Resilience of Supply Chain Systems Using a Survival Model. *IEEE Systems Journal*, *9*(2), pp. 377-381.

Ransbotham, S., Kiron, D., Gerbert, P., & Reeves, M. (2017). Reshaping Business With Artificial Intelligence: Closing the Gap Between Ambition and Action. *MIT Sloan Management Review*, *59*(1).

Roberts, B., Mazzuchi, T., & Sarkani, S. (2016). Engineered resilience for Complex systems as a predictor for cost overruns. *Systems Engineering*, *19*(2), pp. 111-132.

Radio Technical Commission for Aeronautics (RTCA). (2017). *Detect and Avoid Minimum Operational Performance Standards Phase 1 (DO-365)*. RTCA: Washington, D.C.

Radio Technical Commission for Aeronautics (RTCA). (2008). *Minimum Operational Performance Standards for Traffic Alert and Collision Avoidance System II (DO-185B)*, RTCA: Washington, D.C.

Resmerita, S., & Heymann, M. (2003). Conflict resolution in multi-agent systems. In *Proceedings of the 42nd IEEE International Conference on Decision and Control* (Vol. 3), pp. 2537-2542). IEEE.

ReVelle, J. B. (2001). *Manufacturing Handbook of Best Practices: An Innovation, Productivity, and Quality Focus*. CRC Press, pp. 383 – 397.

Ronen, B., & Spector, Y. (1992). Managing system constraints: a cost/utilization approach. *The International Journal of Production Research*, 30(9), pp. 2045-2061.

Rucks, A. C., Ginter, P. M., Duncan, W. J., & Lesinger, C. (2011). A continuity of operations planning template: Translating public policy into an effective plan. *Journal of Homeland Security and Emergency Management*, *8*(1).

Rukhin, A., Soto, J., Nechvatal, J., Smid, M., & Barker, E. (2010). A statistical test suite for random and pseudorandom number generators for cryptographic applications. NIST SP-800-22. Booz-Allen and Hamilton Inc. Mclean Va. Retrieved from https://nvlpubs.nist.gov/nistpubs/legacy/sp/nistspecialpublication800-22r1a.pdf. Site accessed on December 20, 2017.

Rumsfeld, D. (2002). DoD News Briefing [Transcript]. Retrieved from http://archive.defense.gov/Transcripts/Transcript.aspx?TranscriptID=2636. Site accessed on January 12th, 2018.

Russell, S. and Norvig, P. (2010). Artificial Intelligence: A Modern Approach. Pearson: Upper Saddle River, NJ.

Salehie, M., & Tahvildari, L. (2009). Self-adaptive software: Landscape and research challenges. *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, 4(2), 14, pp. 1-42.

Sauter, V. L. (2014). *Decision support systems for business intelligence*. Wiley: Hoboken, NJ, pp. 5-31.

Shaffer, J. P. (1995). Multiple hypothesis testing. *Annual review of psychology*, *46*(1), pp. 561-584.

Schwab, K. (2017). *The fourth industrial revolution*. Crown Business.

Schwab, K. (2016). *The global competitiveness report 2016–2017*, World Economic Forum: Geneva. Retrieved from https://www.weforum.org/reports/the-global-competitiveness-report-2016-2017-1. Site accessed October 20, 2017.

Setola, R., Luiijf, E., & Theocharidou, M. (2016). Critical infrastructures, protection and resilience. *Managing the Complexity of Critical Infrastructures*. J. Kacprzyk, R. Setola, V. Rosato, E. Kyriakides, and E. Rome (eds.), Vol. 90, Springer: Cham, pp. 1-18.

Shapiro, S. S., & Wilk, M. B. (1965). An analysis of variance test for normality (complete samples). *Biometrika*, *52*(3/4), pp. 591-611.

Sheridan, T. B. (2011). Adaptive automation, level of automation, allocation authority, supervisory control, and adaptive control: distinctions and modes of adaptation. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, *41*(4), pp. 662-667.

Simon, H. A. (1955). A behavioral model of rational choice. *The quarterly journal of economics*, *69*(1), pp. 99-118.

Sood, A., & Tellis, G. J. (2005). Technological evolution and radical innovation. *Journal of Marketing*, *69*(3), pp. 152-168.

Stone, P., Brooks, R., Brynjolfsson, E., Calo, R., Etzioni, O., Hager, G., ... & Leyton-Brown, K. (2016). *Artificial intelligence and life in 2030: One hundred year study on artificial intelligence*. Stanford University: Stanford, CA. Retrieved from https://ai100.stanford.edu/. Site accessed on January 10, 2018.

Tague, N. R. (2005). *The quality toolbox* (Vol. 600). ASQ Quality Press: Milwaukee, WI, pp. 247–24.

Tanenbaum, Andrew (2001). *Modern Operating Systems.* Prentice Hall, pp. 184–185.

Tao, S., Ezhilchelvan, P. D., & Shrivastava, S. K. (1995). Focused fault injection testing of software implemented fault tolerance mechanisms of Voltan TMR nodes. Distributed Systems Engineering, 2(1), 39.

Tapolcai, J., Cholda, P., Cinkler, T., Wajda, K., Jajszczyk, A., Autenrieth, A., ... & Svinnset, I. E. (2005). Quality of resilience (QoR): NOBEL approach to the multi-service resilience characterization. *Proceedings of the 2nd International Conference on Broadband Networks (BroadNets 2005).* IEEE. pp. 1328-1337.

Tucker, J. B., ed. *Innovation, Dual Use, and Security: Managing the Risks of Emerging Biological and Chemical Technologies*. MIT Press, 2012.

Tversky, A., & Kahneman, D. (1974). Judgment under uncertainty: Heuristics and biases. *science*, *185*(4157), pp. 1124-1131.

U.S. Code of Federal Regulations [CFR]. Air Traffic and General Operating Rules, 14 C.F.R § 91.167 Code of Federal Regulations. (2000). Retrieved at https://www.gpo.gov/fdsys/granule/CFR-2012-title14-vol2/CFR-2012-title14-vol2-sec91-167. Site accessed on October 4, 2017.

U.S. Department of Defense (DoD). (2005). *DoD guide for achieving reliability, availability, and maintainability*, DoD: Washington, D.C. Retrieved from https://www.dsiac.org/sites/default/files/reference-documents/DoD%20RAM%20Guide%202005%20-%20Modified.pdf. Site accessed on November 10, 2017.

U.S. Department of Homeland Security (DHS). (2009). *National Infrastructure Protection Plan (NIPP) 2009: Partnering to Enhance Protection and Resiliency*, DHS: Washington, D.C. Retrieved from https://www.dhs.gov/publication/nipp-2009-partnering-enhance-protection-resiliency. Site accessed on January 15, 2018.

U.S. Department of Homeland Security (DHS). (2013). *National Infrastructure Protection Plan (NIPP) 2013: Partnering for Critical Infrastructure Security and Resilience*, DHS: Washington, D.C. Retrieved from https://www.dhs.gov/publication/nipp-2013-partnering-critical-infrastructure-security-and-resilience. Site accessed on January 15, 2018.

Wang, J., Dou, R., Muddada, R. R., & Zhang, W. (2018). Management of a holistic supply chain network for proactive resilience: Theory and case study. *Computers & Industrial Engineering*, 125, pp. 668-677.

Watson, H. A. (1961). *Launch control safety study*. Bell Laboratories: Murray Hill, NJ, 1961. (Unpublished Report)

Winston, W. L., & Goldberg, J. B. (2004). *Operations research: applications and algorithms* (Vol. 3). Thomson Brooks/Cole, pp. 974-985.

Woods, D. D. (2006). Essential characteristics of resilience. *Resilience engineering: Concepts and precepts.* E. Hollnagel, D.D. Woods, and N. Leveson (eds.), Ashgate: Aldershot, pp. 21-34.

Woods, D. D. (2015). Four concepts for resilience and the implications for the future of resilience engineering. *Reliability Engineering & System Safety*, 141, 5-9.

Zanin, M., & Lillo, F. (2013). Modelling the air transport with complex networks: A short review. *The European Physical Journal Special Topics*, *215*(1), pp. 5-21.

Zhang, W. J., & Van Luttervelt, C. A. (2011). Toward a resilient manufacturing system. *CIRP Annals-Manufacturing Technology*, *60*(1), pp. 469-472.

Appendices

Appendix A: Additional Background on Resilience Engineering

Notional disruption scenarios that illustrate system performance with and without resilience capabilities and attributes are presented in Figures A-1 through and A-4.

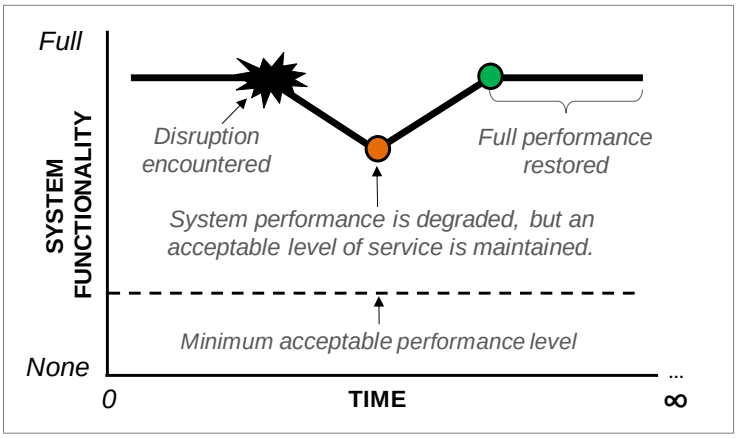


Figure A-1: Example of Performance Degradation with Full Recovery after Disruption

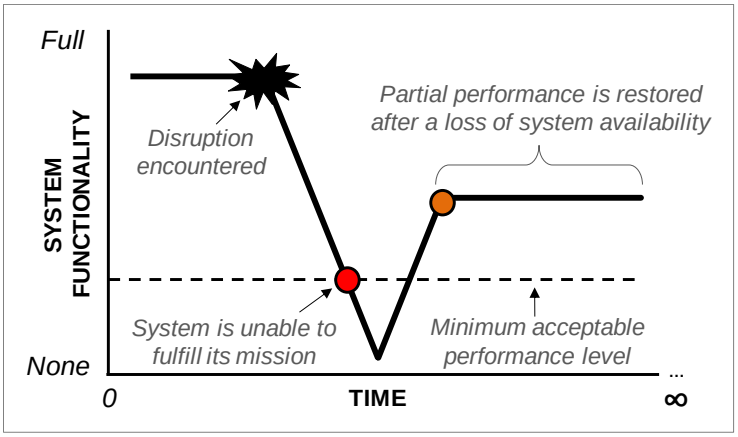


Figure A-2: Example of a System Failure with Partial Recovery after Disruption

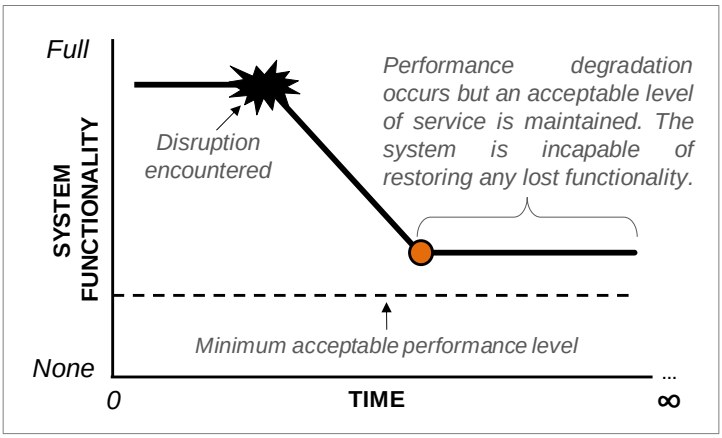


Figure A-3: Example of System Stabilization after Disruption without Recovery

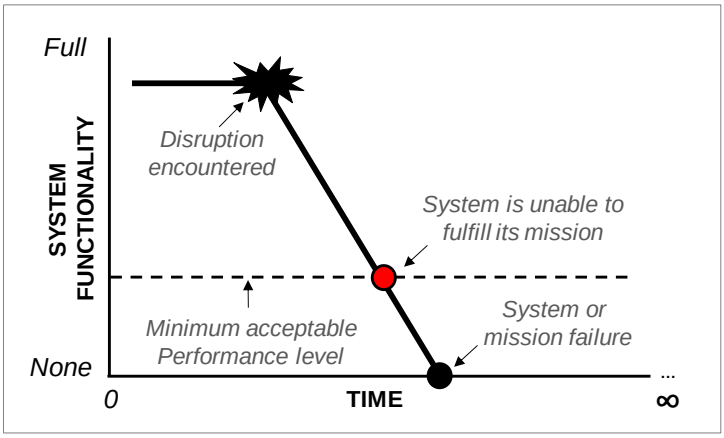


Figure A-4: Example of System or Mission Failure due to Insufficient Resilience

When system disruption is unavoidable, *graceful degradation* which enables the gradual and predictable loss of system functionality is desirable, as opposed to the rapid or instantaneous loss of functionality [Jackson, 2009] (illustrated in Figure A-5). The benefits of graceful degradation include: (i) providing system end-users with time to address safety and security impacts; (ii) maximizing the time for corrective action to enable system recovery or stabilization; (iii) maximizing system availability (when system recovery or stabilization are not feasible).

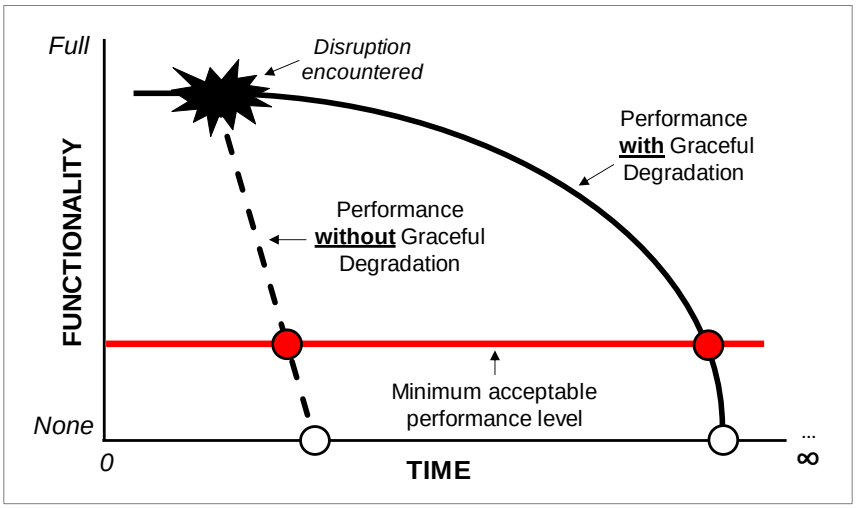


Figure A-5 : System Performance with and without Graceful Degradation after Disruption

Appendix B: Overview of Conventional and Free Flight Air Traffic Management in the U.S. National Airspace System

The National Airspace System (NAS) incorporates human air traffic controllers and technological aids to continuously monitor and direct aircraft operating in U.S. airspace. Automated information-processing, communication, navigation, surveillance, and weather equipment allow air traffic controllers to see the location of aircraft, aircraft flight plans, and prevailing weather conditions, as well as to communicate with pilots. FAA controllers are primarily located in three types of facilities with narrowly defined roles and responsibilities [FAA, 2009]:

- **Airport towers** control the flow of aircraft—before landing, on the ground, and after take-off—within 5 nautical miles of the airport and up to 3,000 feet above the airport. A combination of technological and visual surveillance is used by air traffic controllers to direct aircraft departures and approaches, as well as to communicate instructions and weather-related information to pilots.
- **Terminal Radar Approach Control (TRACON) facilities**—sequence and separate aircraft as they approach and leave busy airports, beginning about 5 nautical miles and extending to about 50 nautical miles from the airport and up to 10,000 feet above the ground.
- **Air Route Traffic Control Centers (ARTCC)—**or en route centers—direct aircraft in transit over the continental United States. Aircraft are controlled through regions of airspace by en route centers responsible for the regions. Control is passed from one en route center to another as aircraft move across a region until it reaches TRACON airspace. Most of the en route centers' controlled airspace extends above 18,000 feet for commercial aircraft. Beyond the radars' sight, controllers must rely on periodic radio communications through a third party—Aeronautical Radio Incorporated (ARINC), a private organization funded by the airlines and FAA to operate radio stations—to determine aircraft locations.

An overview of NAS functions by facility showing the transition of command and control functions between aircraft departure to aircraft arrival is provided in Figure B-1.

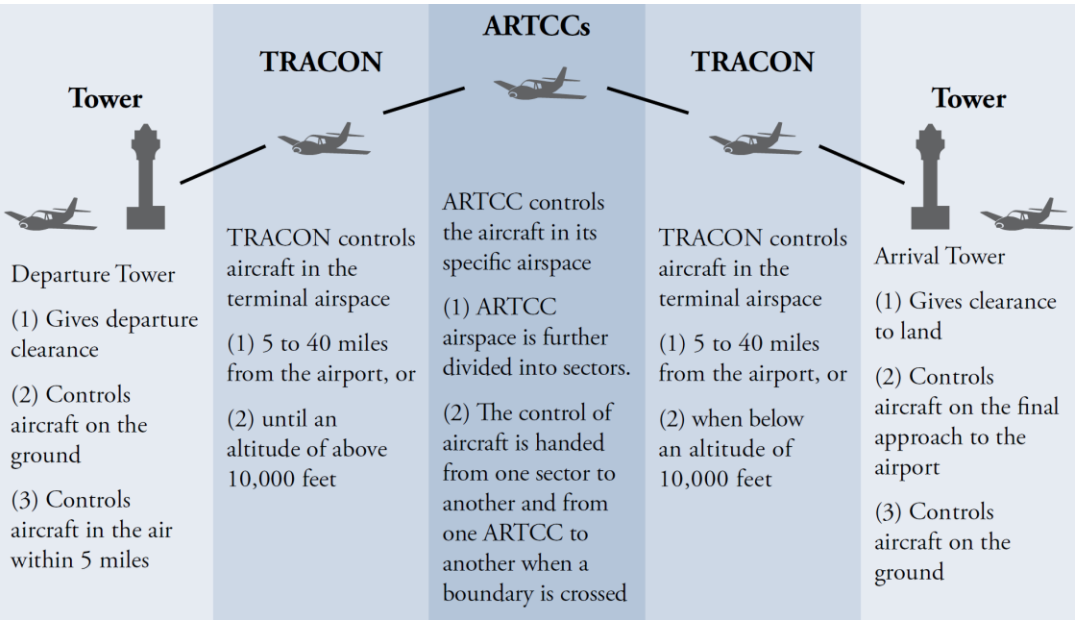


Figure B-1: Summary of NAS Facility Types & Functions [FAA, 2009]

FAA Free Flight initiatives seek to improve the safety, flexibility and efficiency of every phase of flight through the increased use of artificial intelligence and autonomous functionality [GAO, 1998, 2001]. Figure B-2 summarizes the automated and semi-autonomous tools implemented as part of Phase 1 of FAA Free Flight initiatives in the U.S. completed between 1995 and 2002, including:

- **Surface Movement Advisor (SMA)** to provide more efficient movement of aircraft on the ground, including movement between taxiways and runways;
- **User Request Evaluation Tool (URET)** which supports the identification of conflicts in planned aircraft routes and can automatically respond to pilots' request for route changes;
- **Traffic Management Advisor (TMA)** to enable more efficient scheduling of aircraft departures and arrivals at capacity-constrained airports; *and*
- **Collaborative Decision Making (CMD) tool** which automates the electronic exchange and analysis of flight status and airspace capacity information between the FAA and airlines operating in the NAS;
- **Passive Final Approach Spacing Tool (pFAST)** supports more efficient scheduling and sequencing of aircraft final approaches to the destination airport and assignment of runways for landing.

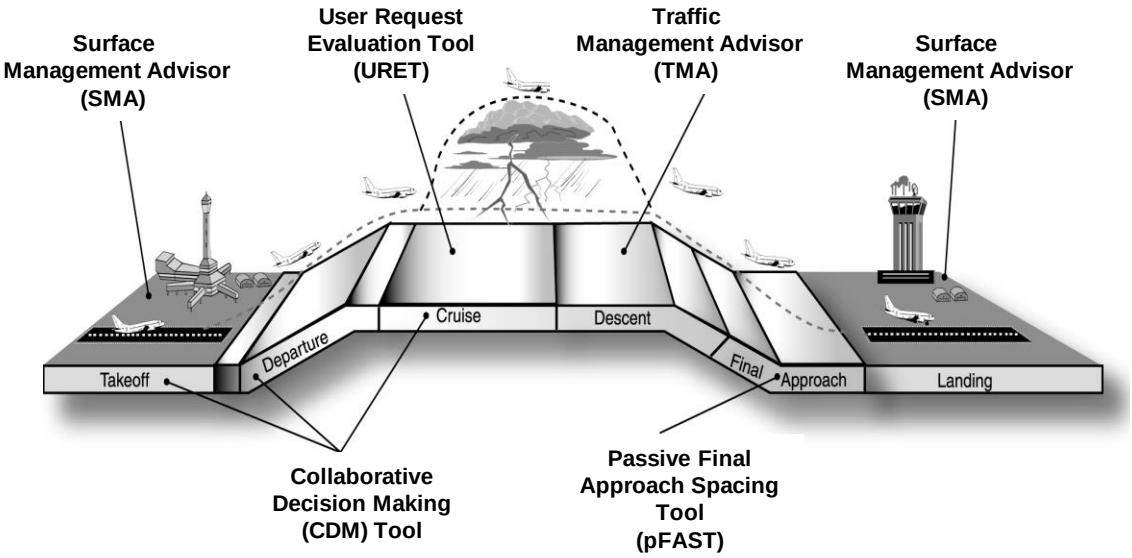
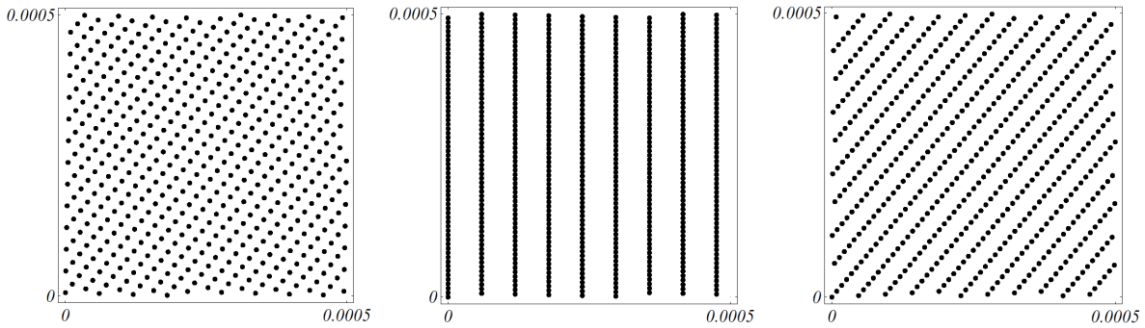


Figure B-2: FAA Free Flight Tools and Application Areas [GAO, 1998, 2001]

Appendix C: Random Number Generation for Monte Carlo Simulation

Default random number generators (RNGs) for many popular software programs and programming languages (e.g. Excel, MATLAB, Mathematica, Java standard library, R, etc.) fail the best-known public-domain statistical tests for uniformity and periodicity of outputs including the DIEHARD test package and National Institute of Technology (NIST) Test Suite [L'ecuyer & Simard, 2007]. Modeling and simulation for the experiments presented in this dissertation was initially implemented using the Java programing language, which uses a Linear Congruential Generator (LCG) algorithm as the default RNG [Oracle, 2017]. Poor uniformity of LCG outputs has been demonstrated when generating *n*-tuples (i.e. ordered data-sets of consisting of n elements) with consecutively drawn random numbers [Marsaglia, 1968], also referred to as the *Marsaglia effect*. Examples of the Marsaglia effect which graphically demonstrate patterns generated by plotting successively drawn cartesian coordinates from an LCG are presented in Figure C-1.



(a) Sparse pattern generated by the ANSIC LCG

(b) Vertical pattern generated by the MINSTD LCG

(c) Diagonal pattern generated by the URN12 LCG

Figure C-1: Examples of the Marsaglia Effect on Cartesian Coordinates (Entacher, 1997)

To mitigate the Marsaglia effect on Monte Carlo sampling of starting and ending waypoints for Free Flight ATM simulation, alternative RNG methods were sought in the literature. For stochastic simulation and statistical experiments, RNGs with multiple independent streams and high periodicity are recommended, such as Multiple Recursive Generators (MRGs) and multiplicative lagged-Fibonacci generators (MLFGs) [L'ecuyer, 2001; L'ecuyer & Simard 2007]. These RNG methods mitigate the issue of low periodicity and poor uniformity by combining multiple LCGs for recursion. Subsequently, the test model was re-implemented in MATLAB, which offers a proven

MRG with excellent efficiency – 'mrg32k3a' [MathWorks, 2017; L'ecuyer, 1999].[1] The 'mrg32k3a' MRG is enabled as part of the simulation initialization script before any Monte Carlo simulation processes are performed (in-lieu of the default MATLAB RNG, i.e. the 'mt19937ar' *Mersenne Twister* algorithm [Matsumoto & Nishimura 1998]).

[1] While alternative methods for RNG can be implemented in Java (including the 'mrg32k3a' RCG), significant computational overhead and efficiency penalties would be incurred (resulting in slower simulation runtimes), prompting the transition of modeling and simulation for the study to MATLAB.

Appendix D: Normal Probability Plots of Paired Differences for Experiment 1

Appendix D.1 – Paired Differences of Experiment 1 Results for TCAS II and DREAM

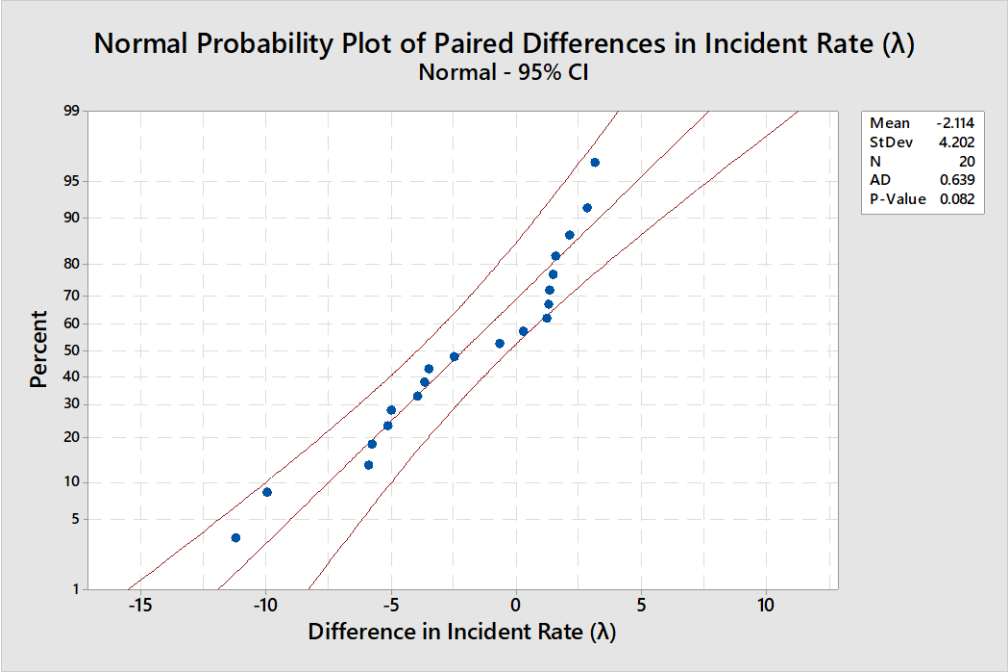


Figure D-1: Normal Probability Plot for Differences in Incident Rate (TCAS II & DREAM)

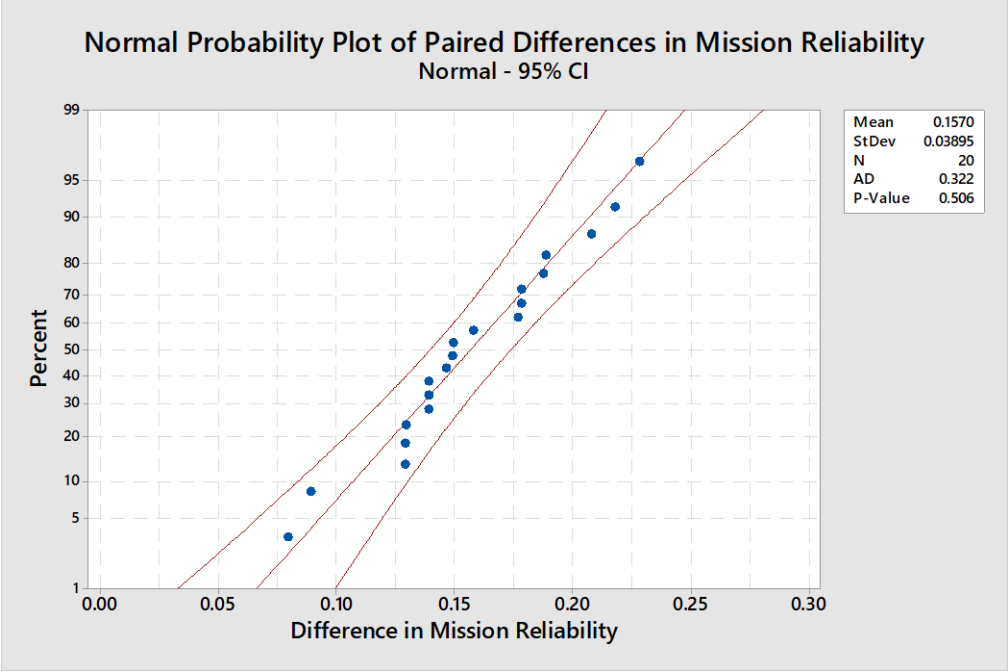


Figure D-2: Normal Probability Plot for Differences in Mission Reliability (TCAS II & DREAM)

Appendix D.2 – Paired Differences of Experiment 1 Results for D Lite and DREAM*

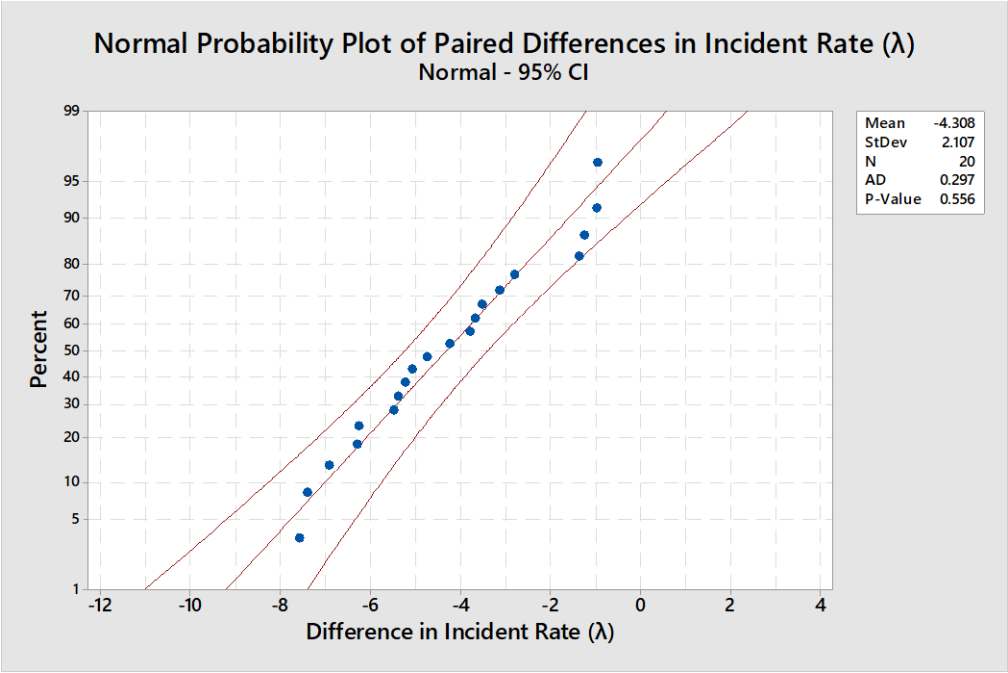


Figure D-3: Normal Probability Plot for Differences in Incident Rate (D* Lite & DREAM)

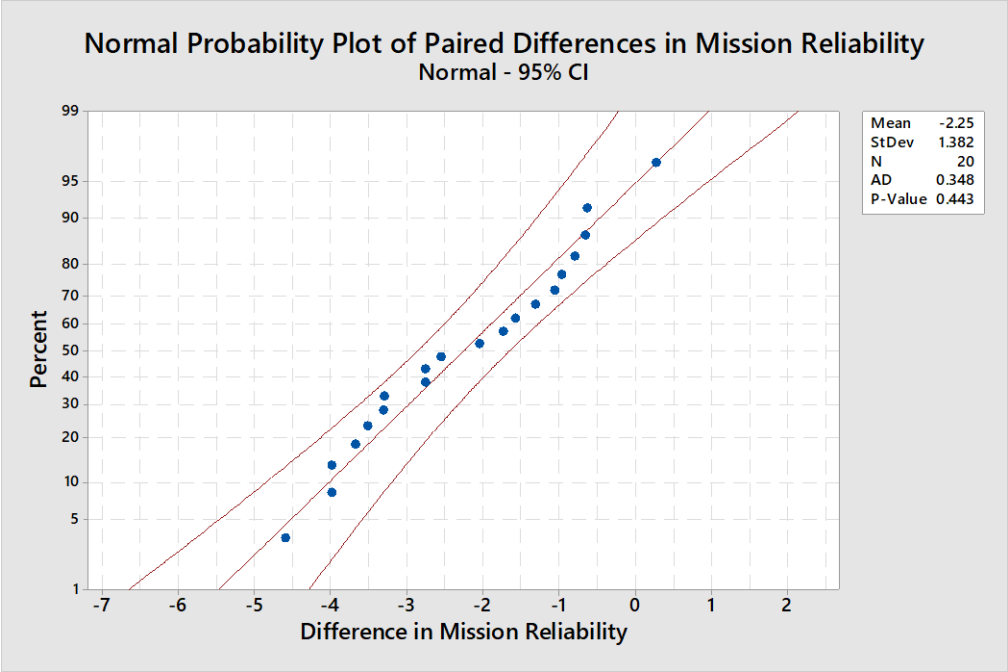


Figure D-4: Normal Probability Plot of Differences in Mission Reliability (D* Lite & DREAM)

Appendix E: Normal Probability Plots of Paired Differences for Experiment 2

Appendix E.1 – Paired Differences of Experiment 2 Results for TCAS II and DREAM

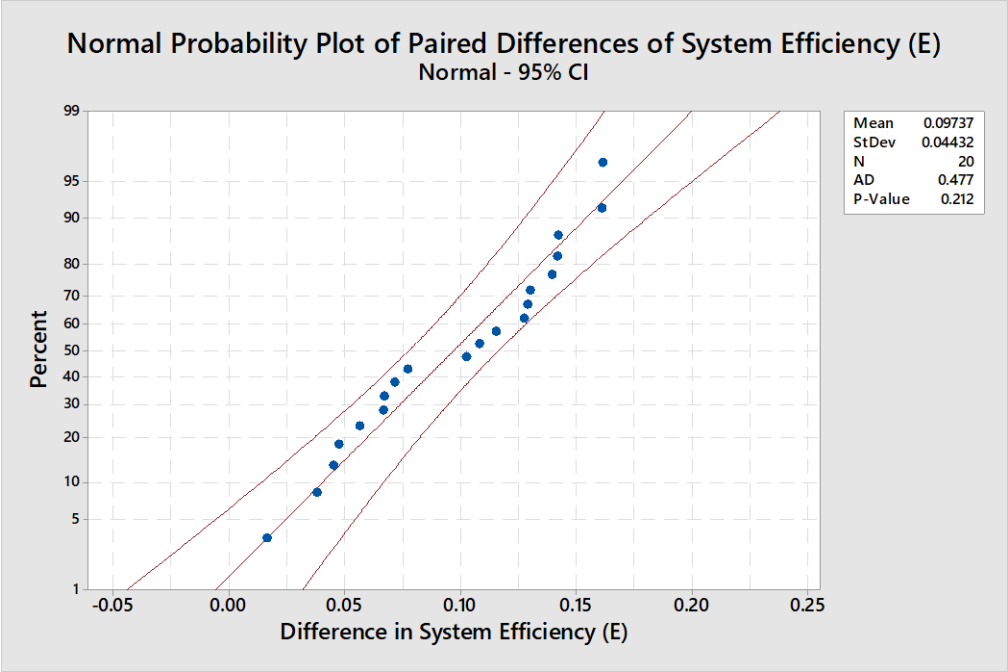


Figure E-1: Normal Probability Plot of Differences in System Efficiency (TCAS II & DREAM)

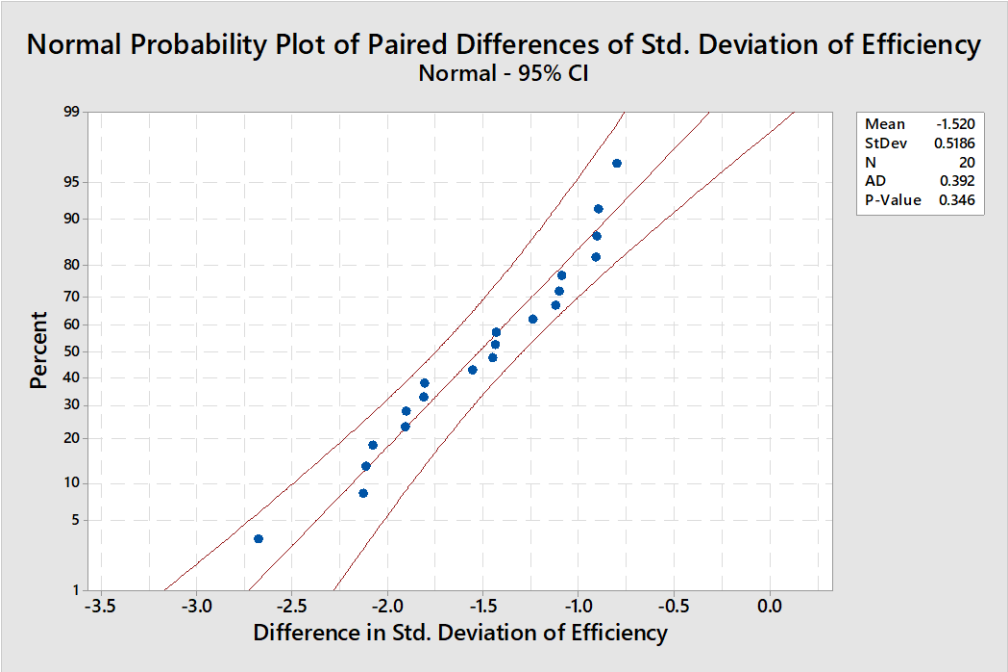


Figure E-2: Normal Probability Plot of Differences in Std. Dev. of Efficiency (TCAS II & DREAM)

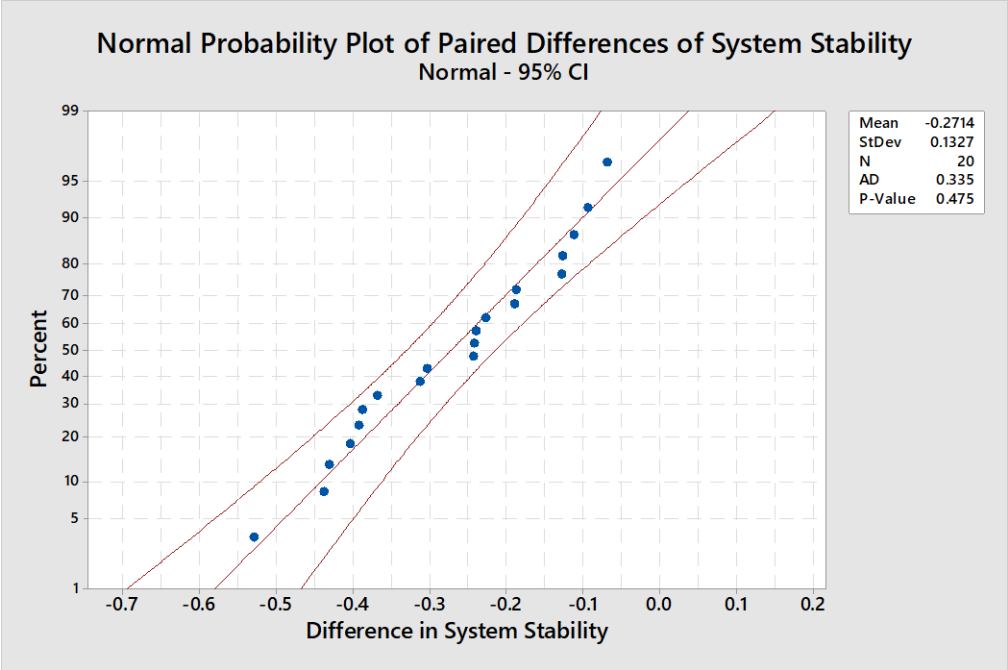


Figure E-3: Normal Probability Plot of Differences in System Stability (TCAS II & DREAM)

Appendix E.2 – Paired Differences of Experiment 2 Results for D Lite and DREAM*

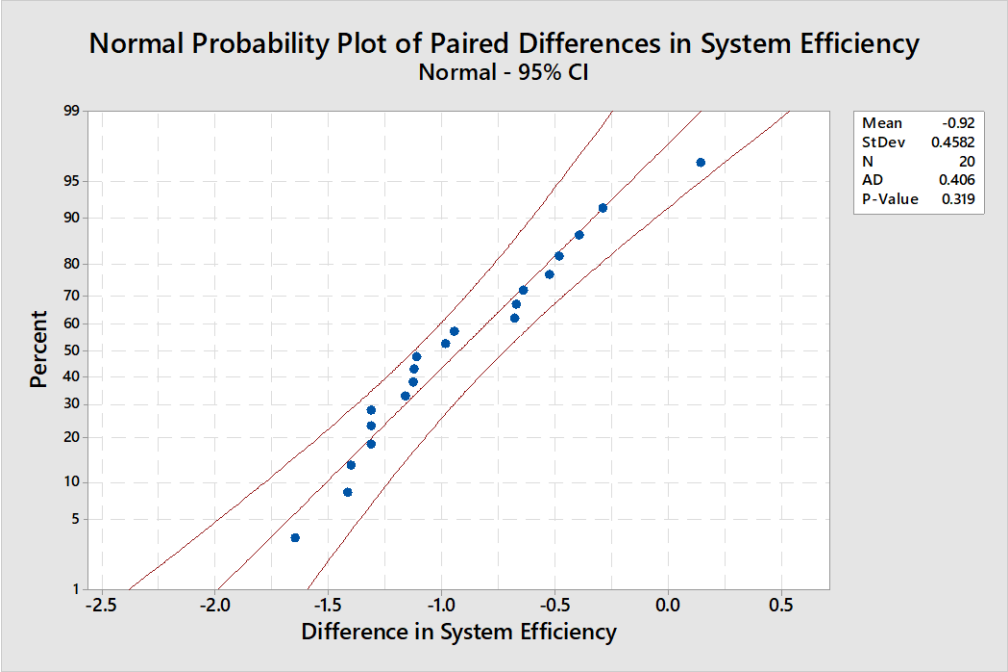


Figure E-4: Normal Probability Plot of Differences in System Efficiency (D* Lite & DREAM)

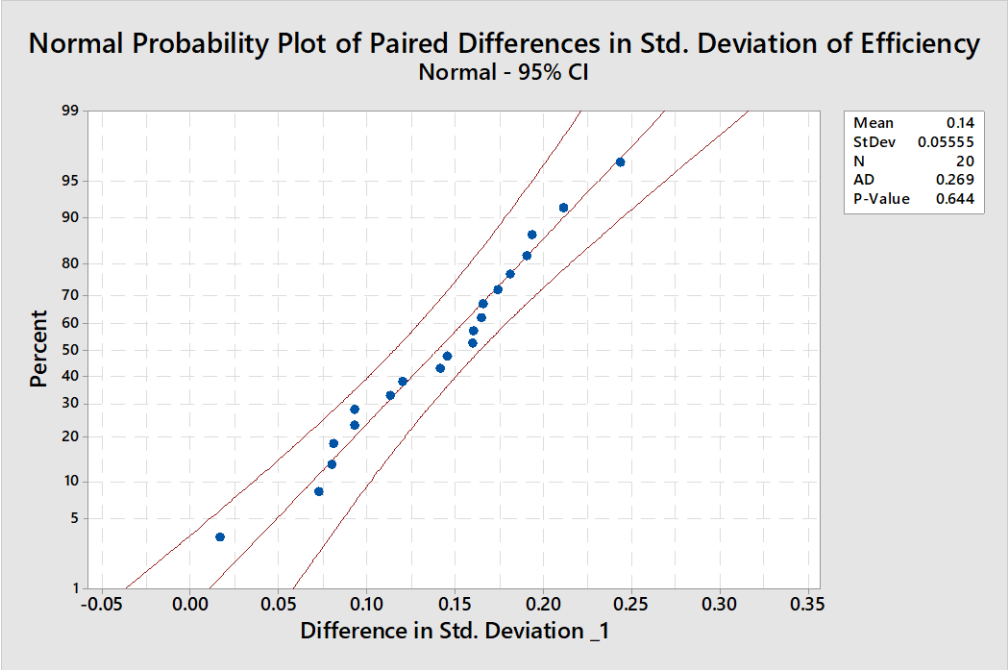


Figure E-5: Normal Probability Plot of Differences in Std. Dev. of Efficiency (D* Lite & DREAM)

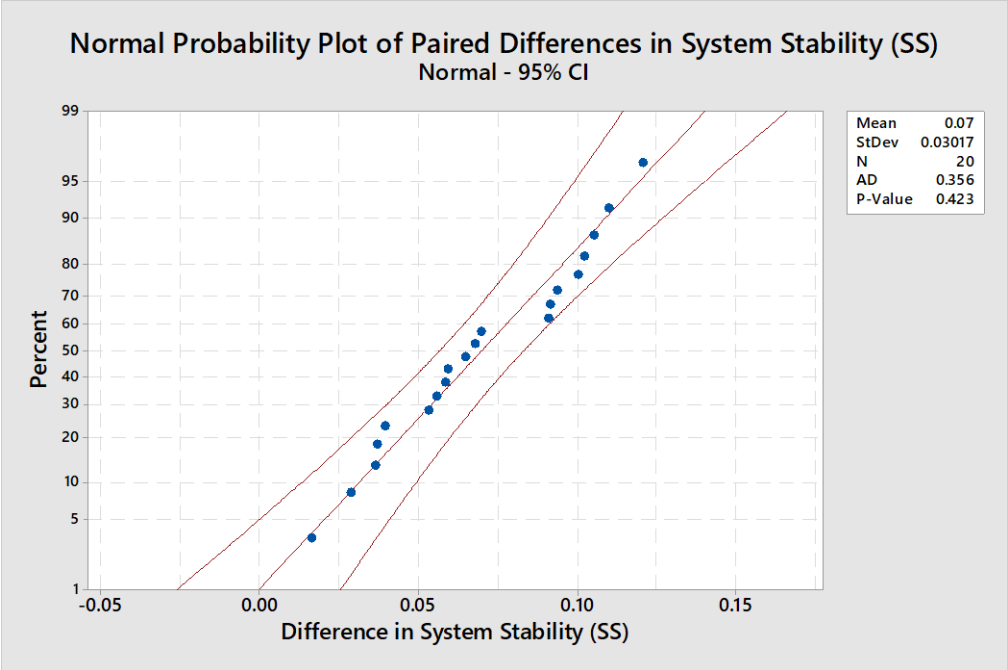


Figure E-6: Normal Probability Plot of Differences in System Stability (D* Lite & DREAM)