ASSESSING THE PERFORMANCE OF HUMAN-AUTOMATION COLLABORATIVE PLANNING SYSTEMS

by

Jason C. Ryan

B. S. Aerospace Engineering
University of Alabama, Tuscaloosa, AL, 2007

Submitted to the Department of Aeronautics and Astronautics
in partial fulfillment of the requirements for the degree of

Master of Science in Aeronautics and Astronautics
at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2011

Author ……………………………………………………………………………………
Department of Aeronautics and Astronautics
May 19th, 2011

Certified by ……………………………………………………………………………
Mary L. Cummings
Associate Professor of Aeronautics and Astronautics
Thesis Supervisor

Accepted by ……………………………………………………………………………
Eytan H. Modiano
Associate Professor of Aeronautics and Astronautics
Chair, Graduate Program Committee

ASSESSING THE PERFORMANCE OF HUMAN-AUTOMATION COLLABORATIVE PLANNING SYSTEMS

by

Jason C. Ryan

Submitted to the Depart of Aeronautics and Astronautics
on May 19th, 2011 in partial fulfillment of the requirements for the
degree of Master of Science in Aeronautics and Astronautics.

ABSTRACT

Planning and Resource Allocation (P/RA) Human Supervisory Control (HSC) systems utilize the capabilities of both human operators and automated planning algorithms to schedule tasks for complex systems. In these systems, the human operator and the algorithm work collaboratively to generate new scheduling plans, each providing a unique set of strengths and weaknesses. A systems engineering approach to the design and assessment of these P/RA HSC systems requires examining each of these aspects individually, as well as examining the performance of the system as a whole in accomplishing its tasks. An obstacle in this analysis is the lack of a standardized testing protocol and a standardized set of metric classes that define HSC system performance. An additional issue is the lack of a comparison point for these revolutionary systems, which must be validated with respect to current operations before implementation.

This research proposes a method for the development of test metrics and a testing protocol for P/RA HSC systems. A representative P/RA HSC system designed to perform high-level task planning for deck operations on United States Naval aircraft carriers is utilized in this testing program. Human users collaborate with the planning algorithm to generate new schedules for aircraft and crewmembers engaged in carrier deck operations. A metric class hierarchy is developed and used to create a detailed set of metrics for this system, allowing analysts to detect variations in performance between different planning configurations and to depict variations in performance for a single planner across levels of environment complexity. In order to validate this system, these metrics are applied in a testing program that utilizes three different planning conditions, with a focus on validating the performance of the combined Human-Algorithm planning configuration.

Experimental result analysis revealed that the experimental protocol was successful in providing points of comparison for planners within a given scenario while also being able to explain the root causes of variations in performance between planning conditions. The testing protocol was also able to provide a description of relative performance across complexity levels.

The results demonstrate that the combined Human-Algorithm planning condition performed poorly for simple and complex planning conditions, due to errors in the recognition of a transient state condition and in modeling the effects of certain actions, respectively. The results also demonstrate that Human planning performance was relatively consistent as complexity increased, while combined Human-Algorithm planning was effective only in moderate complexity levels. Although the testing protocol used for these scenarios and this planning algorithm was effective, several limiting factors should be considered. Further research must address how the effectiveness of the defined metrics and the test methodology changes as different types of planning algorithms are utilized and as a larger number of human test subjects are incorporated.

Thesis Supervisor: Mary L. Cummings
Title: Associate Professor of Aeronautics and Astronautics

ACKNOWLEDGEMENTS

I owe a great many of you a great deal of thanks.

To my Mom and Dad, you have supported me in every decision that I have ever made and stuck by me through thick and thin. When I decided to leave home for college for an engineering discipline we knew little about, you were behind me 100%. When I left for Atlanta for a difficult job in small business, again you were there. When I decided to leave my well-paying job to go back to graduate school to further my education, you again supported me completely. The two of you have influenced me more than I can explain and I have only come to appreciate that since I have left home. I love you both, and I would not be here without your love and support.

To the Office of Naval Research, for funding my research and giving me the opportunity to attend this prestigious institution, for the exciting travel experiences, the ability to meet new people. This has been a wonderful project to be a part of, and I look forward to a few more years of continuing research.

To Jim and Nick, I would not have made it to MIT without you. Although I thought I was a hard working, dependable person, working with you taught me more lessons about engineering, responsibility, and how to manage ones life and resources than I could ever have imagined. We had our ups and downs, but I am forever thankful to you, Jim, for giving me a chance to work with your company. I will be forever grateful for everything that you have done for me.

To Missy, I owe you a great debt of thanks for giving me the opportunity to work with you these past two years (and for the next few more). I know there were times were I struggled with things, was hopelessly lost, or made major mistakes, but you always pointed me in the right direction (with an appropriate amount of force). Working in this lab the past two years has made me realize that this is exactly the type of research I have always been interested in, but never knew what it was. Simply for that, you have my eternal thanks.

To Yves, my thesis reader, I'm sorry for putting you through all of this, and I thank you for your patience, your guidance, and your sense of humor as we went through this process. You've been equal parts friend, mentor, and torturer through this process, but it's all for the better. I could not have finished this thesis without you, and expect to get a bottle of wine from me soon.

To Olinda, my other thesis reader, thanks for taking some time out of your busy schedule to read over my thesis. Also, thanks for showing me a bit around San Diego while I was there. Hopefully, I can make it back at there at some point and have some more time to look around the area.

To the rest of the DCAP Team – Nick, Jon, Emilio, Randy, Ashis, Yale, Raj, Buddy, Susie, Bruno, Jonathan, and Morrisa – thanks for all the hard work in putting the system together. This thesis would not have happened without the combined efforts of all of us to create the DCAP system. I am the major beneficiary of your hard work, so I thank you all.

To those of you in the HAL Lab whom I have come to know and love – Jackie, Daryl, Farzan, Dave, Kris, Luca, Jason, Ryan, Kim, Andrew, Armen, Fei, Paul, Christin, Thomas, Pierre, Luisa, Birsen, Brian, Tuco – thanks for all the good times, the laughs, the beer, the hockey, the volleyball, for eating my cooking, for the late nights and random movies, for dragging me out of my bad moods, for leading me around Boston and Cambridge, and for all the other ways that you made my past two years here in Cambridge an absolute joy! I am delighted to call all of you friends. I know some of you are leaving soon, and to you, I wish you the best of luck with what you are doing. To the rest of you, I still get to see you for at least another year, and I can't wait to see what happens.

To the crews at Tech Catholic Community and Hope In Action – thank you both for reminding what is important to me outside of my research and work life. I feel more grounded now than I have since the end of my undergrad days at Alabama. None of you really know the extent of that story, but sufficed to say, you have my gratitude for it.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

AA	Airborne Aircraft
AAE	Attention Allocation Efficiency
ABE-Auto	Autonomous platform Behavior Efficiency – Autonomy
ANOVA	ANalysis Of VAriance
ASP	Aircraft Schedule Panel
B	Baseline planning condition
B-HA	Comparison of Baseline and Human-Algorithm planning conditions
B-HO	Comparison of Baseline and Human-Only planning conditions
CAG	Carrier Air Wing Commander
CTA	Cognitive Task Analysis
CW	Crew Working (variable in the Variable Ranking Tool)
C1L	Carrier 1 Launches
C1LR	Carrier 1 Launch Rate
C2L	Carrier 2 Launches
C2LR	Carrier 2 Launch Rate
C3L	Carrier 3 Launches
C3LR	Carrier 3 Launch Rate
C4L	Carrier 4 Launches
C4LR	Carrier 4 Launch Rate
DA	Deck Aircraft (variable in the Variable Ranking Tool)
DCAP	Deck operations Course of Action Planner
DL	Decision Ladder
DR	Display Requirement
DRT	Deck Resource Timeline
DS	Deck Support vehicles (variable in the Variable Ranking Tool)
DVT	Disruption Visualization Tool
EART	Emergency Aircraft Recovery Time
EFD	Event Flow Diagram
FMAC	Fast Manned AirCraft
FMAC_6_AAT	Fast Manned AirCraft #6 Aircraft Active Time
FMAC_6_HFR	Fast Manned AirCraft #6 Hydraulic Fluid Remaining
FMAC_6_EART	Fast Manned AirCraft #6 Emergency Aircraft Recovery Time
FR	Functional Requirement
FUAV	Fast Unmanned Aerial Vehicle
FV	Fuel Violation
GPS	Global Positioning System
HA	Human-Algorithm (planning condition)
HBE-IPE	Human Behavior Efficiency – Information Processing Efficiency subclass
hCTA	Hybrid Cognitive Task Analysis
HFV	Hydraulic Fluid Violation

HO	Human-Only (planning condition)
HO-HA	Comparison between Human-Only and Human-Algorithm planning conditions
HRI	Human-Robot Interaction
HSC	Human Supervisory Control
HV-D	Halo Violation – Duration
HV	Halo Violation
ILP	Integer Linear Program
IPE	Information Processing Efficiency
IR	Information Requirement
LP	Linear Program
LZ	Landing Strip/Landing Zone
LZFT	Landing Zone Foul Time
MCO	Mars Climate Observer
MD	Mission Duration
MDP	Markov Decision Process
ME-C	Mission Efficiency – Coverage subclass
ME-E	Mission Efficiency – Error subclass
ME-T	Mission Efficiency – Time subclass
MS	Marshal Stack
P/RA	Planning/Resource Allocation
PCA	Principal Components Analysis
RRT	Rapidly-exploring Random Tree
SA	Situational Awareness
SAGAT	Situational Awareness Global Assessment Tool
SAR	Situational Awareness Requirement
SART	Situational Awareness Rating Technique
SMAC	Slow Manned AirCraft
SMAC_2_AAT	Slow Manned AirCraft #2 Aircraft Active Time
SMAC_2_EART	Slow Manned AirCraft #2 Emergency Aircraft Recovery Time
SMAC_2_EFR	Slow Manned AirCraft 32 Emergency Fuel Remaining
SME	Subject Matter Expert
STO	Scenario Task Overview
SUAV	Slow Unmanned Aerial Vehicle
TAT	Total Active Time
TAAT	Total Aircraft Active Time
TATT	Total Aircraft Taxi Time
TCAT	Total Crew Active Time
TCLR	Total Catapult Launch Rate
TUAT	Total Unmanned ground vehicle Active Time
TV	Total Violations
UAV	Unmanned Aerial Vehicle
UIT	User Interaction Time
UU	User Utilization
VRT	Variable Ranking Tool
WTAI	Wait Time due to operator Attention Inefficiencies

WTI	Wait Time due to operator Interaction
WTO	Wait Time due to Operator
WTP	Wait Time due to Processing
WTQC	Wait Time in Queue at Catapult
WTQCrew	Wait Time in Queue due to Crew
WTQMS	Wait Time in Queue in Marshal Stack

1. INTRODUCTION

Sheridan defined Human Supervisory Control (HSC) systems to be those in which “one or more human operators are intermittently programming and continually receiving information from a computer that itself closes an autonomous control loop through artificial effectors to the controlled process or task environment" [1]. While Sheridan’s original work considered the teleoperation of robots, HSC systems can also include systems that utilize automated algorithms to schedule task assignments or perform path planning for various agents [2-6]. These will be referred to as Planning and Resource Allocation, or P/RA, HSC systems, a model of which is provided in Figure 1 (adapted from Sheridan’s original HSC model in [1]).

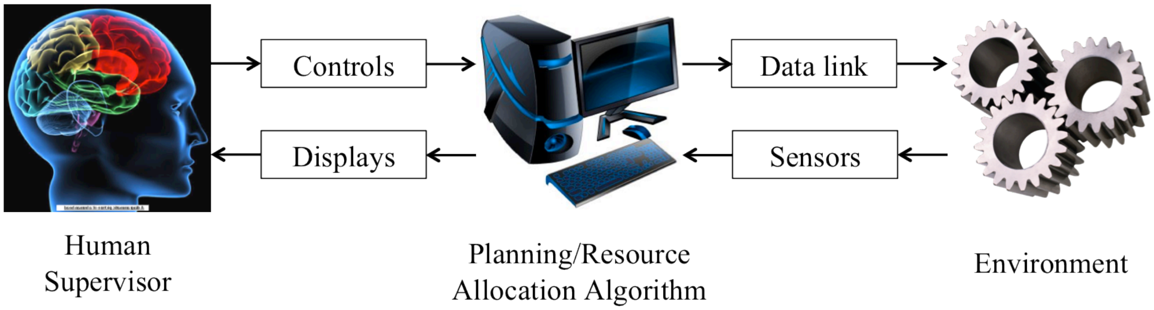


Figure 1. Human supervisory control diagram for P/RA HSC systems, modified from Sheridan [1].

Within a P/RA HSC system, the human operator engages a planning algorithm through a set of control interfaces in order to create a feasible plan of action. Once the plan has been deemed acceptable, it is transmitted to and implemented by the agents in the environment. The planning algorithm then monitors the execution of this plan via sensors in the environment, relaying information back to the operator through a set of display interfaces. A simple, but common form of this is the automobile GPS system, in which drivers input a destination and a set of preferences to an automated planning algorithm, which returns a suggested driving path to the driver. After route acceptance, the system then continually updates the driver on the status of the route, sensed through a GPS receiver, and relayed through a visual display and auditory alerts.

In more complex planning domains, such as military command and control environments [7, 8], the value of P/RA HSC systems lies in the complementary capabilities of human and automated planners. Automated planning algorithms are capable of processing and incorporating vast amounts of incoming information into their solutions. However, these algorithms are brittle and unable to account for conditions that are outside the programmed parameters, especially in uncertain environments. [9]. Also, despite the speed at which algorithms can process information, human operators retain superiority in pattern recognition and the ability to adapt to changing conditions [10, 11]. The human ability to satisfice, or to provide feasible solutions that only address a subset of the overall problem, has also been shown to be highly effective [12, 13]. Recent research has shown that by properly allocating functions between human operators and automated systems, performance superior to either entity alone can be achieved [4, 14, 15]. In the context of P/RA HSC systems, human planners can rely on their experience to determine the factors most important to system performance (as they would otherwise do when satisficing). Communicating these factors aids the algorithm in the development of a local solution that often outperforms the solutions generated by the human or algorithm individually.

The design of P/RA HSC systems requires a systems engineering approach, which addresses the performance of both the human operator and the algorithm, the interactions between them, and their ability to function together in executing system tasks [16]. This approach stems from the belief that successful system performance is a product of both effective component design and effective component integration. The Mars Climate Orbiter (MCO), for example, was destroyed on entry into the Martian atmosphere due to a difference in measurement units between two subcomponents [17]. Although the individual units tested properly, the error in unit consistency went undetected, resulting in a total mission loss. While the MCO case is an extreme result, it highlights the necessity of evaluating interactions between components within the system. Regardless of the performance of the human operator and the algorithm within a P/RA system, if the two cannot effectively communicate in order to execute tasks, the overall effectiveness of the system will likely be diminished.

Viewing this from a systems engineering perspective, several models provide guidance for the development of P/RA HSC systems. Two of these models, the Waterfall [18] and "V" models [19], only address the highest level of process task definition (e.g., *Analysis* and *Design* in the waterfall model). This thesis will focus on a third model, the spiral model [20, 21], which divides these high level tasks into multiple phases of planning, requirements definition, risk analysis, and testing. This set of four steps is continually repeated throughout the process. Figure 2 shows a spiral model for the development of a generic software system. As the spiral moves outward, the design process moves from lower to higher levels of abstraction, beginning with a basic definition of the concept of operations in the center and concluding with final acceptance testing and implementation. The construction of the spiral model also provides guidance to the designer as to where to move if a test shows deficient system performance. For example, the *Integra-*

tion and Test stage includes the final assembly of system components and tests of their ability to interact effectively. Should this test fail, the engineering process should likely return to the *Design Validation and Verification* stage to adjust component design parameters, or to the *Integration and Test Plan* stage if the method of component interfacing requires alteration.

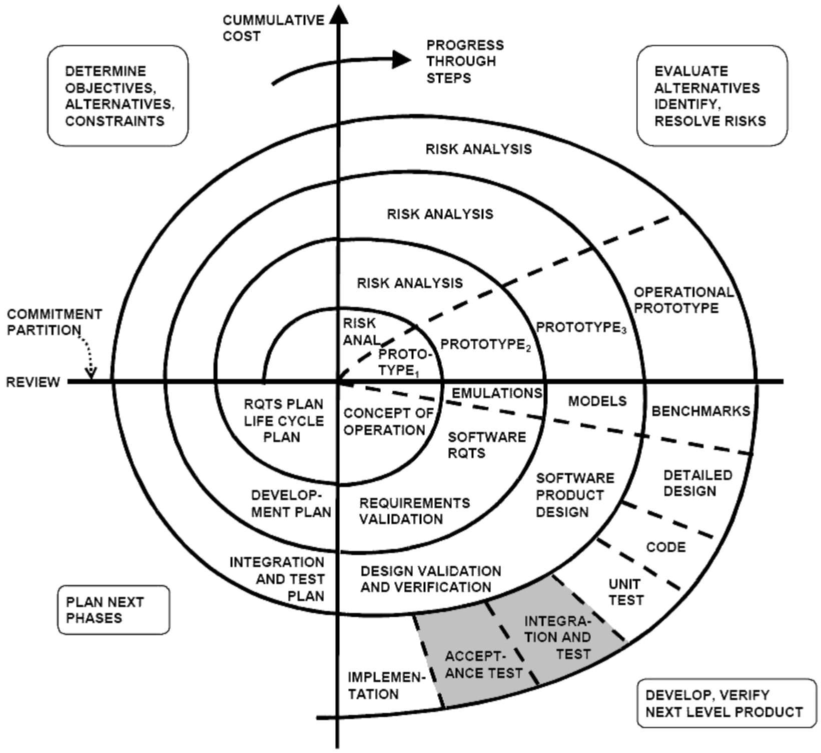


Figure 2. Systems engineering spiral model as adapted to software engineering [20, 21].

The spiral model in Figure 2 is used as a basis for discussion throughout the remainder of this thesis. Specifically, this thesis will address the two final test steps, highlighted in grey in Figure 2 – the *Integration and Test* and *Acceptance Test* stages. The former addresses the effectiveness with which the human operator and the algorithm interact within the system, while the latter addresses the ability of the combined system to effectively perform tasks in the environment. This thesis will address the development of measurement metrics and a testing protocol for evaluating the performance of P/RA HSC systems in these two test steps, addressing both the human and algorithmic components of the system. The testing protocol and measurement metrics should also be generalizable to a wide range of P/RA HSC system domains and algorithm formats. The metrics developed for this protocol are both descriptive and diagnostic, providing empirical comparison points between systems while also identifying the properties of a single system that led to its efficiency (or inefficiency).

1.1. Problem Statement

A systems engineering approach to the evaluation of P/RA HSC systems requires a holistic, comprehensive testing protocol. An obstacle to the creation of this protocol is a lack of both standardized metrics and a standardized methodology of metric definition. While standardized metrics and frameworks exist for defining the performance of both human operators [22, 23] and automated planning algorithms [24, 25], no standardized frameworks are currently in place for the interaction between humans and automated P/RA systems, or for system (mission) performance overall.

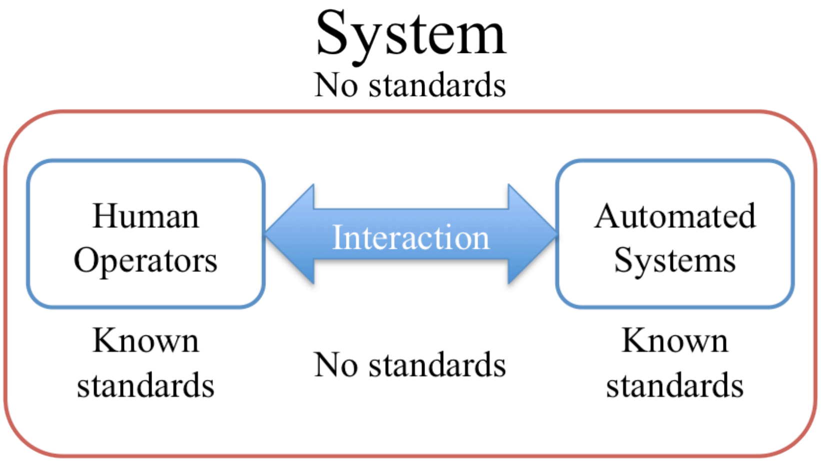


Figure 3. Existence of standardized metrics for HSC systems.

The goal of this thesis is to understand how a set of metrics should be defined for and applied to a P/RA HSC system, and how an analysis of the resulting data can provide insight into the strengths and weaknesses of a human-automation collaborative system. A metric class hierarchy from prior literature [26, 27] is used to guide the creation of metrics for a representative P/RA HSC system, the Deck operations Course of Action Planner (DCAP). The DCAP system utilizes an automated scheduling algorithm to aid operators in replanning tasks in the aircraft carrier deck environment, which is generalizable to a large number of planning and resource allocation HSC systems. Metrics are defined for this system and utilized in an experimental simulation testbed that examines performance over varying complexity levels. The discussion of these testing results addresses both the comparison of system performance within each testing scenario, as well as the performance of the systems across complexity levels. The next section of this chapter details the specific research questions that will be addressed in this thesis.

1.2. Research Questions

This thesis addresses three specific questions:

1. What metrics are required for the evaluation of a Planning and Resource Allocation Human Supervisory Control system as compared to manual planning?
2. How can these metrics assess the variations in performance of human and combined human-algorithm planning agents?
3. How can these metrics predict system feasibility and highlight possible design interventions?

1.3. THESIS OVERVIEW

This thesis is organized into six chapters. Chapter 1, Introduction, describes the motivation and research objectives for this thesis. Chapter 2, Prior Work, details prior research concerning planning and resource allocation algorithms and the creation of metrics for human performance, automated algorithm performance, and the interaction between these elements. Chapter 3, the Deck Operations Course of Action Planner (DCAP), explains the features of the DCAP system and its embedded automated algorithm. Chapter 4, Performance Validation Testing, describes the creation of the metrics used in the analysis of the DCAP system, the testing scenarios, the creation of a set of operator planning heuristics, and subsequent testing of the system. Chapter 5, Results and Discussion, details the results of the application of the defined metrics to the resultant simulation data and the information gained from this process. Chapter 6, Conclusions and Future Work, reviews the contributions of this research in regards to the defined research questions and also addresses future research questions.

2. Prior Work

This chapter provides a review of metrics previously used in validating the performance of planning algorithms and HSC systems (including both P/RA and more generic HSC systems). The first section in this chapter presents a framework for HSC metric classification taken from prior literature. This framework is used as an organizational tool for the remaining sections of the chapter, which provide details on the specific types of metrics utilized in prior literature.

2.1. The HSC Metric Hierarchy

Several non-standardized metric class hierarchies have been developed for HSC systems [28-31]. Metrics can be differentiated into classes according to their attributes, primarily in terms of the object of the application. P/RA HSC metrics can be separated into classes for Human, Automation, and Mission performance as well as Human-Automation Interaction. Mission Performance metrics describe the ability of the system, as a whole, to accomplish its goals in the environment. Automation Performance describes the ability of the automated components – such as automated algorithms or sensors – to perform their specific tasks. Measures of Human-Automation Interaction typically describe the active processes the operator uses to input commands to or acquire information from the system (e.g. mouse clicks), while Human Performance measures typically describe features native to the human operator (such as fatigue or stress). Table 1 provides a brief representation of four prior metric hierarchies according to this metric class structure.

Table 1. Metric classes from prior work.

	Human Performance	Automation Performance	Human-Automation Interaction	Mission Performance
Olsen and Goodrich [28]		X	X	X
Steinfeld *et al*. [29]	X	X	X	X
Crandall and Cummings [30]	X	X	X	
Scholtz [31]	X		X	

Olsen and Goodrich's hierarchy [28] focused almost exclusively on quantifying robot performance, with most metric classes excluding measures for the human operator; those that considered human operators examined only their interaction with the autonomous platform. Steinfeld *et al*. [29] included classes addressing both the human and automated aspects of the system, as well as the effectiveness of the overall system. However, the class of human performance metrics did not differentiate between human-automation interaction and individual human performance measures. Steinfeld's hierarchy also lacked depth in the definitions for human and system performance (only three metrics appear in each category), but did provide numerous metrics for automation performance.

Crandall and Cummings [30] created metrics for single-robot and multi-robot systems and created additional measures addressing human interaction. This hierarchy did not include direct measures of system performance, although it did provide a differentiation between human performance and human-automation interaction. Scholtz's [31] hierarchy addressed measures dealing primarily with the human operator and their interaction with the system, but lacked metrics for system performance and automation performance.

While each of these four hierarchies is lacking in some manner, they combine to address each of the aspects of P/RA HSC systems as depicted in Figure 3. However, as a whole, only the automation performance class contains a large number and variety of example metrics; the remaining categories only include few, if any, examples. These deficiencies were also noted by Pina *et al.* [26, 27], who incorporated the work of these (and other) authors in creating an expanded and detailed categorical structure for HSC metrics. The five main categories developed from this work, with additional subcategories, are shown in Table 2.

Table 2. Pina et al.'s [26, 27] metric classes and subclasses.

Mission Efficiency	Autonomous Platform Behavior Efficiency	Human Behavior Efficiency	Human Behavior Precursors	Collaborative Metrics
• Time based • Error based • Coverage based	• Adequacy • Autonomy • Usability • Self-awareness	• Attention Allocation Efficiency • Information Processing Efficiency	• Cognitive Precursors • Physical Precursors	• Between Humans • Between Autonomous systems • Between Human and Automation

Pina *et al.*'s [26, 27] five metric classes consider each of the various aspects of a general HSC system and encompass much of the previous metric hierarchies, while also providing additional detail in the definition of subclasses for each category. Pina *et al.* also include an additional class of Collaborative measures, which is an additional perspective on human-automation interaction. Collaborative measures address the sociological aspects of the system, considering the automation to be a member of the "team" of operators performing system tasks. These measures may address the effectiveness of collaboration between multiple human operators, multiple automated agents, or between human and automation.

Mission Efficiency metrics measure the performance of the system *as a whole* as it performs tasks within its domain – a critical issue in the *Acceptance Test* stage in the spiral model. The remaining categories address the performance of individual subcomponents and their efficiency of interaction, supporting the *Integration and Test* stage in the spiral model. Autonomous Platform Behavior Efficiency contains measures for the effectiveness of an algorithm in its computations and its capability to support the human operator in his/her tasks. Human Behavior Efficiency measures address the performance of the human operator as he or she engages the system through both cognitive (information extraction) and physical (command input) means. Human Behavior Precursor metrics examine the endogenous factors that affect human interactions (such as physical and mental fatigue or operator situational awareness). The final category, Collaborative Metrics, addresses the degree to which human users and automated agents are able to work together to accomplish tasks. Figure 4 highlights how Pina *et al*.'s classes of metrics apply to the P/RA HSC metric hierarchy originally shown in Figure 1.

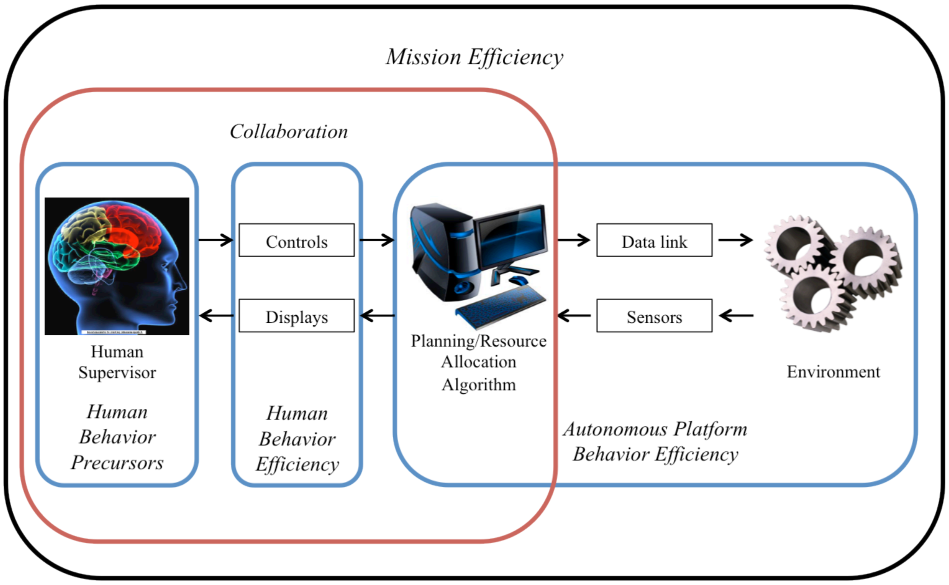

Mission Efficiency
Collaboration
Controls
Displays
Data link
Sensors
Human Supervisor
Human Behavior Precursors
Human Behavior Efficiency
Planning/Resource Allocation Algorithm
Environment
Autonomous Platform Behavior Efficiency

Figure 4. HSC diagram highlighting Pina's metric classes

The metrics included in these categories can fulfill both descriptive and diagnostic roles. All metrics are descriptive with respect to some aspect of the system. For Planning/Resource Allocation systems, descriptive metrics document the objective performance of the system and its subcomponents (the human operator and algorithm). For a path planning system, a descriptive Mission Performance measure may address the total travel time on the path or the cost of the path (e.g. total work). Descriptive measures for the algorithm may address the total time required to replan or make take the form of a scoring function applied to the solution. A descriptive measure for the human operator may include a rating of their situational awareness or trust in the system.

These same measures can also be used in a diagnostic manner, explaining the performance of other metrics. While total mission completion time is a widely used descriptive measure, it has no ability to explain the conditions within the environment that lead to its final value. This can only be revealed by additional diagnostic metrics that illuminate specific details of the performance of the system. For instance, a metric noting that the human operator required more time to execute replanning tasks may provide one explanation for high values of mission completion time. Metrics demonstrating that the system performed poorly on a single mission subtask may provide an alternate explanation for this same factor. Additionally, a second round of diagnostic measures can be applied to each of these cases in order to determine why the human operator required more time to replan (longer time to perform a replanning subtask) or why the mission subtask required more time (deadlock in path planning or unnoticed failure). This can continue iteratively until a definite root cause explanation is obtained.

The ultimate goal of the system is to provide maximum effectiveness in performing tasks in the environment, which is revealed primarily by measures of Mission Efficiency. However, in cases where poor mission performance is seen, descriptive measures may not effectively identify the mechanisms leading to problems. Thus, a combination of metrics addressing each of these factors – the mission, human operator, algorithm, and human-automation interaction classes– is needed to provide a full analysis of the system [26, 27]. The remaining sections of this chapter will address measures for each of these aspects individually as they relate to P/RA HSC systems.

2.2. METRICS FOR MISSION EFFICIENCY

Measures of Mission Efficiency address the ability for the complete HSC system to perform tasks in the world, and exact definitions for these measures depend on the environment in which the HSC system acts. Pina *et al.* [26, 27] differentiated measures of Mission Efficiency measures into Error-based, Time-based, and Coverage-based measures. This section will address these measures and provide examples used in prior studies, focusing on those used in P/RA systems.

Error measures identify the number of errors that occur in the execution of the P/RA system solution. Errors can be attributed to either the human or the automation performing inappropriate actions (errors of commission) or not fulfilling desired objectives (errors of omission). For a path planning P/RA system, the returned solution may be a path that avoids specific areas (such as threat zones) while minimizing costs or collisions [4, 32, 33]. Error measures for such a path planning system may track how many collisions occur or how much threat is accrued by flying into unacceptable areas (both are errors of commission). In other cases, the inability to address tasks within a given time window [34, 35] can be considered as errors of omission (failing to perform certain tasks). These measures are descriptive and diagnostic only with respect to the performance of the returned solution. The identification of the specific actions on the part of the human operator or the algorithm that lead to this performance can appear in other metric classes.

Time-based measures include all temporal measures, primarily addressing the total time of solution execution (the mission time or mission duration) [36-40]. By definition, however, these are limited to systems with a temporal component. For P/RA HSC systems that perform time-independent task allocations, these measures may not be important.

Coverage-based metrics can also be included in some cases [3, 41, 42]. A common application of military P/RA planning systems is in target destruction tasks, where an algorithm supports a human operator in identifying, tracking, and destroying hostile targets. In some cases, the number of targets may outnumber the available resources, making the destruction of every target impossible. The percentage of total targets destroyed can be used as a measure of overall performance for the system [41]. Additionally, a measure of missiles fired per enemy target destroyed is descriptive in terms of the actions of the algorithm but may also be diagnostic in revealing the efficiency of system actions. In this case, high values of missiles fired per target destroyed can explain poor overall performance (e.g., the system did not effectively utilize its resources or the missiles had difficulty in reaching and destroying targets) [42].

These Mission Efficiency measures examine the effectiveness of the generated solution in light of the system objectives, but they are not the sole indicator of a well-performing system [16]. These measures are, however, the primary descriptive metrics of the system and are often the primary criterion on which system implementation is based. These measures are also affected by the individual performance of the human operator and the algorithm and the quality of interaction between the two, each of which must be considered in the course of system evaluation. The next section will address one of these aspects – the ability of the algorithm to support the mission, described by measures of Autonomous Platform Behavior Efficiency.

2.3. Metrics for Autonomous Platform Behavior Efficiency

In P/RA HSC systems, an algorithm does not necessarily exist as an independent agent and may interact with a human operator in order to perform system tasks. In this regard, the algorithm must perform adequately within the system and provide sufficient support for mission operations. The ability of the algorithm and the associated interface to accomplish these objectives was included as part of Autonomous Platform Behavior Efficiency in Pina *et al.*'s metric hierarchy [26, 27]. However, in selecting metrics that define the performance of the automated algorithm, the type of algorithm and the domain of application will determine the number of applicable metrics. Before providing examples of Autonomous Platform Behavior Efficiency, this section will review common types of algorithms used in P/RA HSC systems and the domains to which they are applied.

2.3.1. Common Algorithms for Planning and Resource Allocation

Several different forms of planning algorithms have been proposed for, or implemented in P/RA HSC systems [8, 15, 32, 33, 36, 40, 43-45]. However, these various algorithms can be grouped into three categories based upon the assumptions made and actions taken by the algorithms. These three classes are Deterministic, Probabilistic, and Heuristic algorithms [46, 47]. Table 3 provides a brief comparison of these algorithm classes.

Table 3. Comparison of Algorithm Classes.

	Deterministic Algorithms	*Probabilistic Algorithms*	*Heuristic Algorithms*
Assumed knowledge of the world	Complete	Incomplete	Incomplete
Decision basis	Fully defined cost functions and constraints	Probability density functions	Heuristic rules
Examples	MILPs, Dynamic Programming, Clustering	MDPs, Kalman/particle filtering, RRTs	Tabu search Hill-climbing algorithms

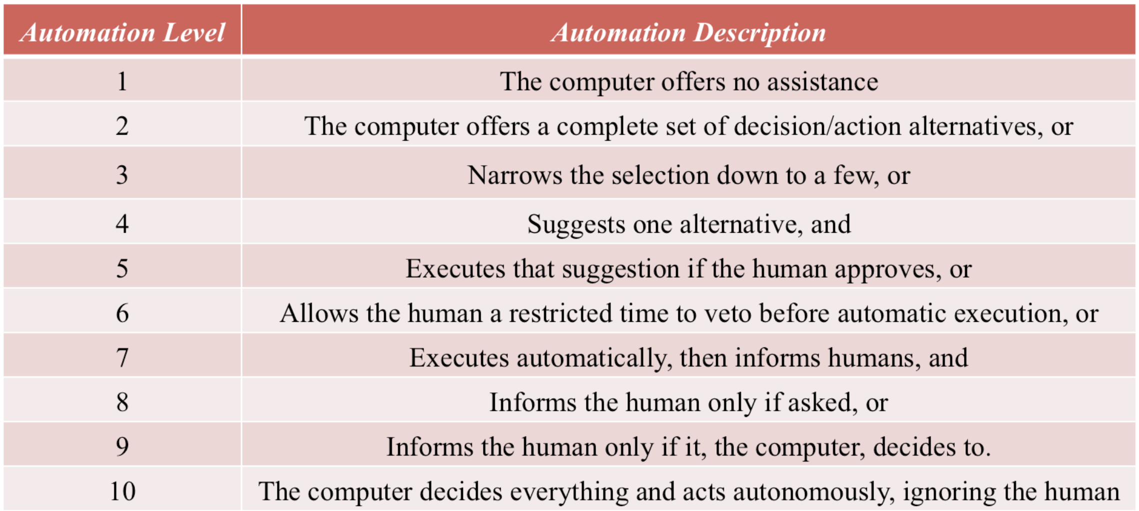
Deterministic algorithms utilize explicit cost functions and constraint models to perform an exhaustive search of the domain. For correct solutions, these algorithms require access to all information relating to these cost and constraint models. If this information is accessible, these algorithms will return an optimal solution (if one exists). This class includes Mixed-Integer Linear Programs [32, 44, 45], graph exploration algorithms such as Breadth- and Depth-first search, potential fields [36, 40], and many other forms [8, 15, 33, 43].

In contrast with deterministic algorithms, probabilistic algorithms [48-51], such as Markov Decision Processes (MDPs), assume incomplete knowledge about the world and calculate responses based on probability models. Kalman and particle filters also fall within this class, but instead use mathematical filtering techniques to reduce the level of incompleteness of information. For instance, systems performing target tracking may not know the exact location of the target to be tracked, but may be able to build a probability density function describing the likelihood of the target's location on the map [49].

Heuristic algorithms [37, 41, 52-59] also assume incomplete information about the world, but do not rely on probabilities in order to make choices. Instead, heuristic algorithms rely on a set of heuristics – "rules of thumb" [60] – to calculate responses. For these cases, the exact equation describing optimality may not be known, due to either problem complexity or to the inability to model certain constraints accurately. In this case, a heuristic function, such as a scoring metric [61, 62], can be used to judge the relative performance of the system. This class of algorithms includes Tabu search [41, 52-54] and hill-climbing algorithms [61, 62], as well as several other forms [37, 55-59].

The amount and types of data available from the environment can influence the choice of algorithm for the system. Deterministic algorithms typically require complete data on the state of the world. For a tracking task, the algorithm requires precise information on the terrain, the current position of the target, and information about the current state of tracking vehicle and its capabilities. Probabilistic algorithms can compensate for cases where the system does not have precise information on the location of the target, as noted above. The system's solution could be optimal on average but may not be optimal for any one case. Heuristic algorithms can be used when the characteristics of an optimal solution or the world are not known, or if the problem space is complex enough that complete, feasible solutions are not expected. In this case, the system iterates through possible solutions, selecting candidates for the next iteration based on a set of basic rules. These algorithms do not guarantee optimality, and instead seek solutions that reach a certain threshold of performance – a concept known as "satisficing" [63].

By definition, HSC systems require the presence and interaction of human operators in order to accomplish tasks. The required level of interaction may vary as described by Sheridan's ten levels of automation, listed in Figure 5.



Automation Level	*Automation Description*
1	The computer offers no assistance
2	The computer offers a complete set of decision/action alternatives, or
3	Narrows the selection down to a few, or
4	Suggests one alternative, and
5	Executes that suggestion if the human approves, or
6	Allows the human a restricted time to veto before automatic execution, or
7	Executes automatically, then informs humans, and
8	Informs the human only if asked, or
9	Informs the human only if it, the computer, decides to.
10	The computer decides everything and acts autonomously, ignoring the human

Figure 5. Sheridan and Verplank's Ten Levels of Automation (adapted from [64]).

In one extreme (Level 1), the human operator performs all tasks without any aid from the automated system. At the other extreme (Level 10), the automated system performs all tasks, requiring no assistance from (and offering no notifications to) the human operator. The remaining eight levels comprise the majority of Human Supervisory Control systems, with operator workload and input gradually decreasing as level increases. For the case of P/RA systems, many exist in the range between Level 3 (the automated system provides several options) and Level 6 (the system executes a suggested solution unless vetoed by human operator). In each of these cases, the human operator and automated system work collaboratively in order to perform a shared task, with the operator possibly providing suggestions to the system or selecting (or vetoing) one of several suggested solutions. In this context, the performance of the P/RA algorithm requires measuring the ability of the algorithm to support these collaborative mission-replanning tasks. These measures of Automation Platform Behavior Efficiency are discussed in the next subsection.

2.3.2. Measures of Autonomous Platform Behavior Efficiency

Pina *et al*. divided measures for Autonomous Platform Behavior Efficiency into four categories (Table 2). Measures of Adequacy address the ability of the algorithm to support the mission computationally, focusing on the accuracy and reliability of the algorithm. Originally, this category only considered qualitative measures for the entire HSC system and did not differentiate between the interface and the algorithm. For P/RA systems, this can be expanded to include traditional algorithm performance measures such as runtime and error (infeasibility/incompleteness) rates (see [24, 25, 65-70] for examples). Within P/RA systems, ideal algorithms would be both highly reliable and highly accurate, accepting user inputs and creating valid plans with no errors.

In the context of P/RA systems, measures of Usability address the human operator's subjective opinions of the algorithm's ability to support the mission, as well as the ability of the user to understand how to interact with the algorithm through the display and control interface. This may involve a subjective evaluation of the system by the user [37, 71] or asking the user to explain their thought processes during interaction [72, 73]. In other cases, these qualitative measures have been operationalized into quantitative measures [74-77], such as tracking the order and duration of user's interactions with the system and comparing these to the actions of an expert user [74-76]. With regards to P/RA systems, these measures may also ask the user to evaluate the performance of and their ability to understand the algorithm through surveys [3] or other means. The goal of these measures is to determine the ease of use of the algorithm – that the user is able to understand its actions, able to predict its performance, and understand how to appropriately interact with the algorithm.

Autonomy refers to how well the system is able to function without operator interaction. This measure has its roots in Human Robot Interaction (HRI) research [6, 28, 30, 35, 78-80], where the performance of one or more robots can degrade over time. This resulted in measures of "neglect" [28, 78] that judged how long the system could maintain performance above a certain threshold without operator input. This may not be an applicable measure for every P/RA algorithm. If the current plan or schedule does not degrade in performance until an exogenous event occurs, and the algorithm *requires* human interaction in order to replan, neglect tolerance is dependent on the occurrence of the exogenous event and not on the algorithm. For all systems that do not exhibit these two conditions, typical measures of neglect tolerance apply.

Self-awareness is primarily intended for autonomous systems that can independently self-monitor and self-diagnose their performance. In the context of P/RA systems, this would involve the ability of the algorithm to relate the actual performance of its solutions to the predicted performance. This could potentially lead to the algorithm adjusting its parameters as time continues (known as “on-line” learning; see [81]). For a P/RA system, the inclusion of this ability would be beneficial to the embedded algorithms. However, this is still an area of emerging research [82-84] and is beyond the scope of this present work.

Metrics within the class of Autonomous Platform Behavior Efficiency address the performance of the automated algorithm and its ability to support the P/RA HSC system in its tasks. While these measures address the planning algorithm and its responses to human input, they do not address the performance of the human operator as he or she interacts with the system. These measures for human performance are divided into categories of Human Behavior Efficiency and Human Behavior Precursors and are addressed in the next section of this chapter.

2.4. METRICS FOR HUMAN PERFORMANCE

Human performance can be characterized along two dimensions. Measures for Human Behavior Efficiency address the effectiveness with which the human operator actively engages the system through the interface elements, acquiring information and executing tasks. Human Behavior Precursors address how the efficiency of this interaction is affected by certain endogenous factors inherent to the human operator. These precursors simultaneously both influence and are influenced by Human Behavior Efficiency. Each of these factors will be discussed individually in the following subsections, beginning with Human Behavior Efficiency.

2.4.1. Metrics for Human Behavior Efficiency

For P/RA HSC systems, physical interaction with the algorithm is mediated by a display interface. Thus, measures of interaction efficiency address the ease with which the user assimilates information from and executes control inputs through the system interface. Pina *et al.* divided these measures into two separate categories – measures for Attention Allocation and Information Processing.

The ability of the user to successfully distribute their attention across multiple competing tasks in order to assimilate incoming information is referred to as Attention Allocation Efficiency (AAE). For P/RA systems, this involves the operator's ability to monitor the overall state of the world, to monitor the state of individual priority tasks, to be aware of failure and status messages, and to execute replanning actions. Measuring the operator's efficiency in doing these tasks requires understanding what aspects of the interface the user is interacting with at a given time and with what purpose. These measures can be taken by tracking mouse cursor location [77], eye-tracking [85, 86] or through verbal reporting on the part of the user [73, 87, 88]. The end goal of these measures is to determine the effectiveness of the user's information acquisition process and to highlight any necessary design changes that can aid the operator in attending to the correct information at the correct times.

The efficiency of attention allocation with the system can be viewed as how well operators process their internal queue of tasks. In this view, the operator is a processing server in which incoming tasks received, are processed, and then removed from the queue [89]. As such, the efficiency of this interaction assesses how long tasks wait to be added to the queue and, once there, how long they must wait to be addressed. These measures are defined as Wait time due to operator Attention Inefficiencies (WTAI) [90] and Wait Time due to Operator (WTO), respectively [79]. Ideally, these measures would be minimized, indicating that the user has sufficient attentional resources to identify necessary tasks and work through them quickly.

Information Processing Efficiency (IPE) metrics are aimed at determining how effectively the user interacts with the system while performing a single control action. These measures are highly influenced by research in the field of Human-Robot Interaction (a subset of HSC), and include tracking explicit interactions with the interface, the time of interaction with interface segments, and the rate of decisions made or actions performed [26, 27]. Highly efficient operators will require a minimum number and duration of interactions in order to perform system tasks.

Similarly to AAE measures, IPE measures may also involve tracking user interaction with the interface. For a P/RA system, IPE measures denote the overall time and number of interactions required to complete the replanning task (and individual subtasks). Here, the outright number of mouse clicks and total time of activity in each interface is of concern (as opposed to AAE measures, which address the *order* in which these tasks and subtasks are performed).

Combined, these measures of Human Behavior Efficiency address the operator's ability to perform one or more tasks, and how well they manage switching between competing tasks. These may be both descriptive, allowing comparisons of user interaction times across different interface types, and diagnostic, providing guidance to designers on what aspects of the interface require a redesign. These measures are jointly influenced by the design of the system interface and endogenous human factors, such as fatigue and situational awareness, which affect the user. The latter, termed Human Behavioral Precursors, is discussed in the next section.

2.4.2. Metrics for Human Behavior Precursors

Human Behavioral Precursors are underlying factors that affect the performance of users in an HSC system. These can either be physical factors, such as physical fatigue and sleeplessness, or cognitive factors, such as mental workload and situational awareness (SA). Several references provide discussions on this topic (see [22, 23] for examples), but for P/RA HSC systems, the predominant factors are the cognitive precursors and their affect on human decision making. For P/RA systems, SA measures will address the ability of the operator to maintain awareness about the current operational state of the overall schedule, of individual schedules for vehicles (or other entities), of the presence of errors in the system, and the ability of the operator to forecast future system states.

Mental workload has been quantified by surveys [3, 34, 37, 42], user interaction measures [3, 39, 91-94], or measures of utilization [89, 95], which returns user interaction time as a percentage of the total mission duration. These are primarily descriptive measures, where excessively high values may lead to poor decision-making and an inability of the human operator to perform at optimal efficiency. For P/RA systems, the mental effort required to understand the actions of the algorithm might be addressed through a survey, while the effort required to implement actions in the system could be tracked by measures of utilization.

The measures discussed in the previous two sections address the specific interaction between the human operator and the P/RA HSC system. Measures of Human Behavior Efficiency focused on the specific interactions between the user and the automation, while measures of Behavior Precursors addressed endogenous factors that affect this interaction. A third aspect of interaction addresses the effectiveness of the human operator and the algorithm in working as a team. These measures of Human-Automation Collaboration are discussed in the next section.

2.5. METRICS FOR HUMAN-AUTOMATION COLLABORATION

Measures of Human-Automation Collaboration consider the automation to be an active participant in system performance; the automation is treated as a teammate working with the human operator. This class of metrics includes the user's trust in the automated system [96-104] and the adequacy of the user's mental model of the automation [105-108]. For P/RA systems, this addresses the ability of the user to understand the planning and/or resource allocation strategies of the algorithm and the effects of user input on these strategies. Trust denotes how much confidence the user has in the ability of the algorithm to aid in creating a viable solution. In cases where a human has the choice between a manual and automated planning system, a lack of trust in the system or the inability to form accurate mental models may lead the user to return to manual planning and reject the automation, regardless of its performance. For a P/RA HSC system, measures of trust typically correlate to the willingness of the operator to accept the plans generated by the automation, or if automation is optional, may explain the operator's utilization of the automated system [109, 110].

Mental models are “the mechanisms whereby humans are able to generate descriptions of system purpose and form, explanations of system functioning and observed system states, and predictions (or expectation) of future system states” [105] and may take many forms [107]. When a human operator has a highly accurate mental model, they are better able to understand and predict the performance of the automation. This engenders trust in the user, which continues to build until the system exhibits unexpected behavior. Inaccurate mental models can be a product of the operator’s inability to understand the system or of unreliable system performance and may result in major accidents or abandonment of the system [105, 106, 108]. For P/RA HSC systems, this class of measures concerns how well the user is able to understand the processes of the automated algorithm. For instance, Rapidly-exploring Random Tree (RRT) algorithms randomly select points in space when creating paths in an environment. This randomness may make it difficult for a human operator to understand the planner’s actions, build a mental model, and predict system behavior. A less random algorithm, such as an ILP that has a set cost function, may be more predictable for the operator.

2.6. Chapter Summary

This chapter has reviewed a metric class hierarchy put forth in previous literature [26, 27] and discussed its application to P/RA systems. The first section of the chapter reviewed the contents of this hierarchy, followed by four sections describing its five metric categories. The second section addressed measures of Mission Efficiency, which describe the functionality of the system as a whole. This was followed by a section discussing common algorithm types for P/RA systems, which classified algorithms in terms of three basic categories of functionality. Limitations on applicability and appropriate metrics for each were also discussed in this section, as were measures of Autonomous Platform Behavior Efficiency. These measures serve to describe the ability of the algorithm to support operations within the HSC system. The third section of this chapter discussed two categories of human performance measures. Human Behavior Efficiency addresses the operator's active engagement of the interface, while Human Behavior Precursors measure the passive influences that affect this engagement. The final section of the chapter discussed overall measures of the collaboration between the human and the automated system. Together, this set of five metric classes allows for an analysis of the performance of the system as a whole (Mission Efficiency), as well as the performance of individual subcomponents (the remaining four classes). As noted in this chapter, the specific definition of metrics within these classes is dependent on the characteristics of the automated planner and the domain in which the system functions. The next chapter will provide a description of the representative system used in this thesis, the Deck operations Course of Action Planner. DCAP is a representative P/RA HSC system requiring the interaction of a human operator and an embedded planning algorithm to reschedule operations on the aircraft carrier deck.

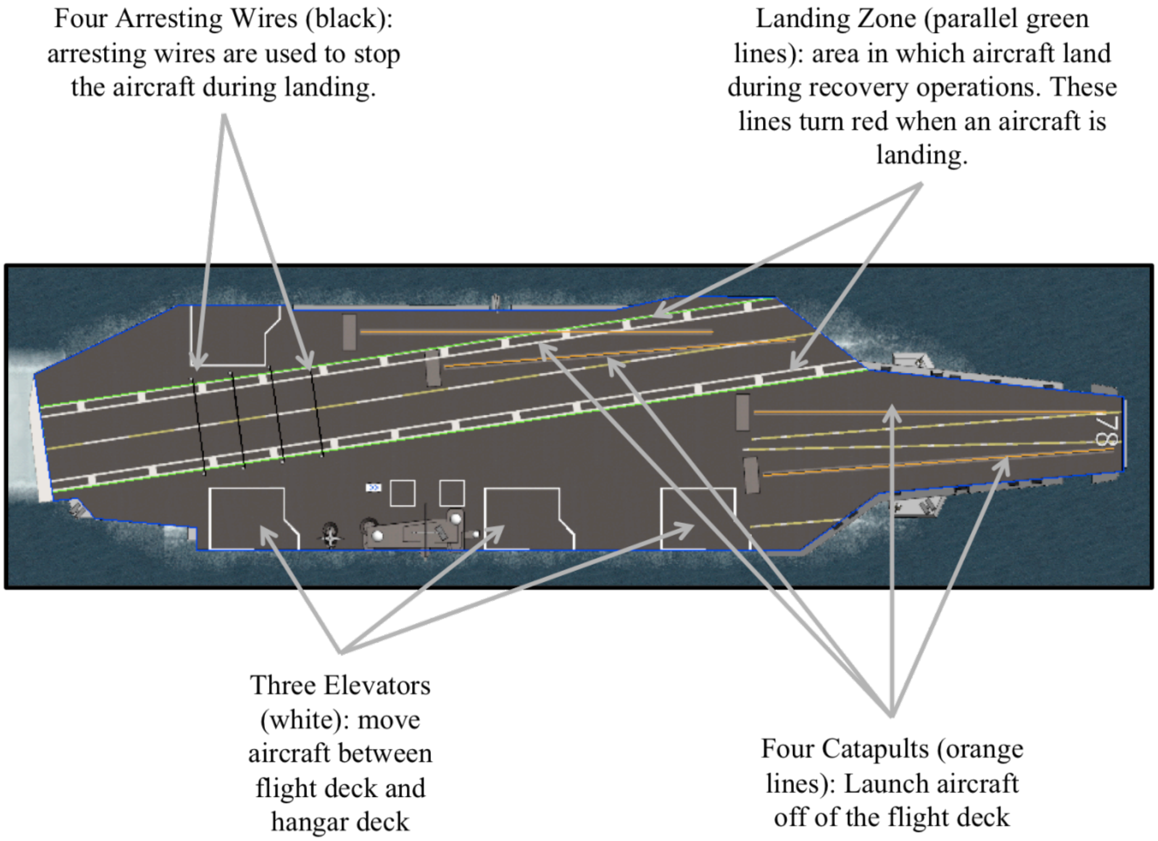
3. The Deck Operations Course of Action Planner (DCAP)

The DCAP system was developed for the purposes of exploring planning under uncertainty for heterogeneous manned-unmanned environments. DCAP is a representative P/RA HSC system, as DCAP requires input from both a human operator and an algorithm in order to perform scheduling tasks. The system focuses on a simulated aircraft carrier deck environment, where numerous crewmembers and aircraft act simultaneously in conducting flight operations (actual en route mission and strike planning are outside the scope of this simulation). In this regard, the problem is one of resource allocation – the set of aircraft on the deck must accomplish specific tasks that are enabled through the use of limited resources, such as launch catapults, elevators, fuel stations, and a landing strip. While the DCAP simulation is specific to carrier operations, there is little conceptual difference between the task allocation performed in this system and that done in airport traffic routing, supply chain management, or other logistics and supply chain management problems.

In DCAP, the operator has the choice of when to engage the system to replan the schedule. The operator may choose to do so due to the occurrence of a failure in the system (e.g., a catapult fails during launch operations) or due to the operator's dissatisfaction with the current schedule. During the replanning process, the human operator provides guidance to the planning and scheduling algorithm through a set of displays. The algorithm returns a proposal for a new, theoretically near optimal operating schedule that incorporates the operator's instructions while also accounting for additional system constraints (e.g., compensating for any failures that are present and ensuring that no aircraft runs out of fuel). This proposed schedule is presented to the operator through modifications to the display in which the inputs were created. This simulation environment also includes an embedded vehicle routing system, implementing some collision avoidance capabilities. This chapter describes the simulation environment and the DCAP system components.

3.1. THE SIMULATION ENVIRONMENT

The DCAP simulation environment is intended to replicate operations on United States aircraft carriers. The deck environment is identical in layout to the current fleet of Nimitz-class carriers and is shown in Figure 6.


Four Arresting Wires (black): arresting wires are used to stop the aircraft during landing.

Landing Zone (parallel green lines): area in which aircraft land during recovery operations. These lines turn red when an aircraft is landing.

Three Elevators (white): move aircraft between flight deck and hangar deck

Four Catapults (orange lines): Launch aircraft off of the flight deck

Figure 6. Basic Areas of the Aircraft Carrier Deck.

There are four forward facing catapults, oriented in forward and aft pairs. Within each pair, aircraft launch from the catapults in an alternating fashion; between the pairs, aircraft may launch simultaneously. After launching, aircraft proceed to a mission area. After mission completion, aircraft return to a holding pattern, known as the Marshal Stack (MS), several miles away from the ship. Aircraft remain in the holding pattern until they are given clearance to land. When clearance is given, aircraft exit the Marshal Stack individually with consistent spacing. On landing, aircraft must catch one of the restraining cables with a "tailhook," which extends backwards from the bottom of the aircraft. If the tailhook does not catch a wire, the aircraft must cycle back to the holding pattern before attempting a second landing. It can also be seen in Figure 6 that the aft catapults (Catapults 3 and 4) are collocated with the landing strip, applying an additional constraint to operations. This area can be used for either landing or launching aircraft, but not both. A time penalty is also incurred when changing the deck configuration between launch and landing, as the landing cables must be replaced or removed by personnel on the deck. The simulation includes four different generic aircraft forms, modeled from realistic platforms. These four vehicles are variations on fast/slow and manned/unmanned aircraft and are listed in Table 4.

Table 4. Types of aircraft modeled in the DCAP simulation.

	Fast (higher flight speed; requires weapons)	*Slow (lower flight speed; no weapons)*
Manned	Fast Manned Aircraft (FMAC, based on F18) Lowest endurance	Slow Manned Aircraft (SMAC, based on C2 Greyhound) High endurance
Unmanned	Fast UAV (FUAV, based on X-47B Pegasus) Medium endurance	Slow UAV (SUAV, based on the MQ-1 Predator) Highest endurance

Fast aircraft have higher flight speeds and require weapons to be loaded before taking off. They also have lower endurance (total possible flight time) than the slow aircraft. Slow aircraft have lower maximum flight speeds, so they have a far lower fuel consumption rate. Both UAV types have longer endurances than their manned counterparts, with the SUAV having the highest endurance overall. The FMAC has the lowest endurance but represents the largest proportion of the fleet.

Despite these differing characteristics, all aircraft taxi across the deck at the same speeds, roughly equivalent to human walking speed. This is a safety constraint on operations; taxi speed is limited due to the high number of crew on deck (typically over 100 individuals on the 18,210 m^2 deck). Aircraft are the driving elements within the system schedule; every other entity on the deck, including crew, can be seen as resources utilized by the aircraft to perform tasks. In the simulation, aircraft tasks describe the high-level actions of the aircraft, such as “Taxi to parking spot” or “Takeoff from Catapult 2.” Tasks are not given defined start and stop deadlines as system complexity and constraints may not permit actions to occur at precisely defined times. For example, as the schedule executes, variations in process times and traffic congestion on deck lead to delays in the schedule. A task that was originally given a start time of t might only be able to begin at $t + n$ due to limitations on the rate of fueling, transit, and other actions[1]. Instead, advancement to the next task is based on satisfying state conditions (e.g., the taxi task ends when the aircraft reaches the desired final location). This accounts for the variety of interactions that constrain operations on deck, such as the replanning of taxi routes, delays due

[1] Planners (human or algorithm) are not allowed to command changes in task execution rates. The rate at which a task occurs is either a set property of the resource or an inviolable safety constraint (e.g. taxi speed for aircraft).

to lack of crew escorts, or constraints on launching aircraft simultaneously at adjacent catapults. Additionally, in the simulation, several tasks are given variable processing times sampled from Gaussian distributions[2] in order to model the variations seen in real-life operations.

As noted earlier, crewmembers are resources for aircraft to utilize, and their presence is required for a number of operations. Seven different subsets of crew are modeled in the simulation, each represented by the color of their uniform in the real world (Table 5).

Table 5. List of crew groups (by color) and roles.

Personnel Uniform Color	*Role*
Yellow	Escorts and guides aircraft on deck
Brown	Oversees plane maintenance; aids in escorting aircraft on deck
Blue	Responsible for deck equipment; aids in escorting aircraft on deck
Red	Handles weapons loading and unloading
Purple	Responsible for fueling aircraft
Green	Operates catapult and landing strip arresting wires
White	Safety officers – serve only a monitoring role

In the DCAP simulation, each aircraft requires 1 yellow-, 1 brown-, and 2 blue-shirted crewmembers to be present in order to taxi, a set of 5 green-shirted crewmembers present to operate a given catapult, and ten (the same ten assigned to Catapults 3 and 4) present to operate the landing cables[3]. Additionally, some wheeled vehicles – Deck Sup-

[2] Mean times of operation and distributions for fueling, landing, and takeoff procedures were taken from interviews with subject matter experts including nearly two dozen instructors at a U. S. Naval training base, each with several years of experience in deck operations. See Appendix A for further information on these distributions.

[3] In actual operations, only three crewmembers are required for the landing cables. This constraint was modified within the simulation to serve a secondary function of moving these crew out of the landing zone to prevent landing conflicts.

port Vehicles – also exist in the simulation, just as in the actual deck environment. In reality, these vehicles would be tow trucks used for relocating aircraft. In this simulation, these are modeled as futuristic unmanned weapons loaders, which assist red-shirted crew in performing their tasks on deck.[4] The movement of these entities (aircraft, crew, and Deck Support Vehicles) is animated for the user within a main display window, termed the *Carrier Display* window; Figure 7 shows a close-up view of the deck from this window. The *Carrier Display* serves as the foundation around which the remainder of the DCAP system is built. The next section will detail the specific display elements in the system.

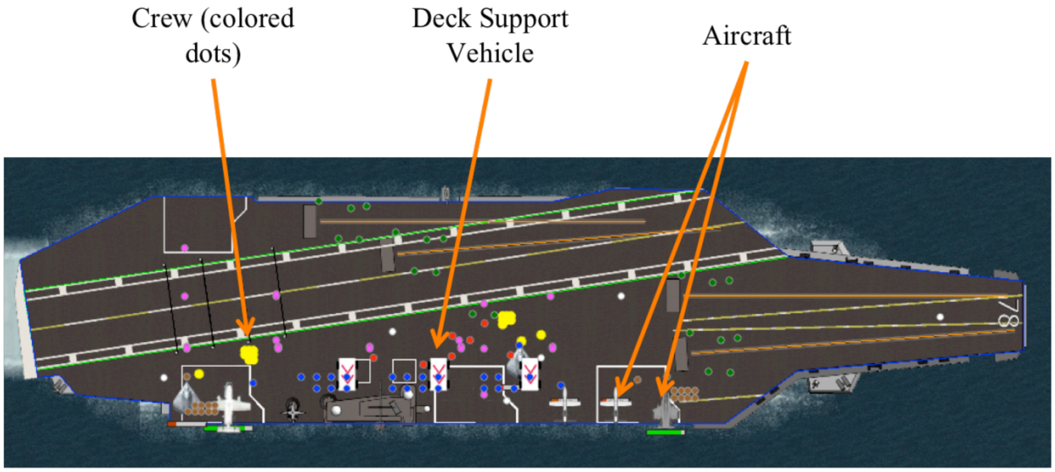


Figure 7. View of the Carrier Deck, showing crew, Aircraft, and Unmanned Weapons Loaders.

3.2. Display Elements

The DCAP system utilizes a set of display elements to display information about the current operating schedule, aircraft states, and system failures to the operator. The operator then interacts with the automated system in order to create at a feasible plan of operations that he or she finds acceptable. Bruni *et al*. provide a model for this collaborative

[4] This difference was influenced by the goals of the overall research program and testing the inclusion of unmanned systems being included in the Naval environment.

human-automation decision making, defining both the process (Figure 8) and roles for entities in the system [111, 112].

The process begins with the acquisition of data from the world (*Data Acquisition* block), which is then used by the Generator in a *Data Analysis* process. The result of this data is used in an *Evaluation* step. This is guided by a Moderator who describes elements of the solution to the Generator, makes sub-decisions that require refinement of the solutions, and may request further data analysis. When the Moderator has created an acceptable solution option (or set of options) it is sent to the Decider for approval. The solution is then either accepted or rejected.

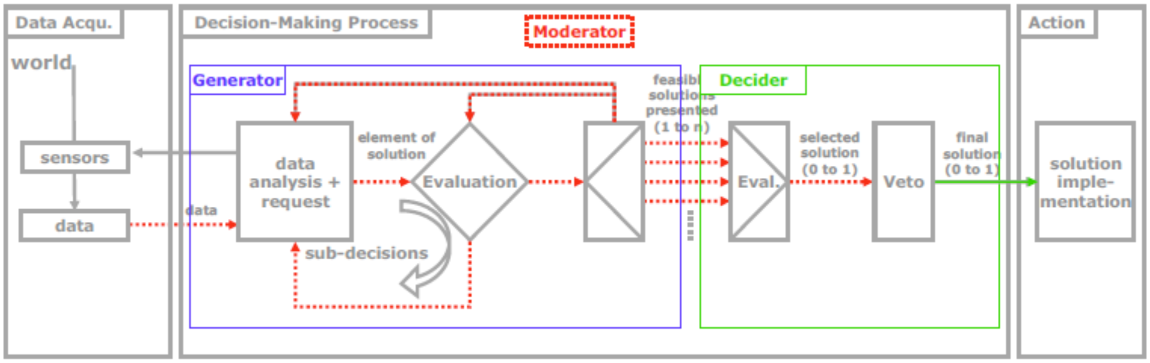


Figure 8. Model of Human-Automation Collaborative Decision Making [111].

Applying this model to DCAP, the algorithm plays the role of solution Generator, while the human operator plays the roles of Moderator and Decider. This model of interaction influenced the creation of three separate interface configurations for the DCAP system – an Information Display configuration (*Data Acquisition*), a Plan Creation configuration (*Data Analysis and Request* and sub-decisions for the Moderator), and a Proposal Review configuration (*Evaluation* and *Veto* for the Decider). A Hybrid Cognitive Task Analysis (hCTA) was used to generate specific function and information requirements for each of these three interfaces. The hCTA process involves the creation of theorized process flow diagrams of high-level operator actions within the system (e.g., "Monitoring" or "Replanning."). These process flow diagrams include segments for operator processes, decisions, assumptions, and iterative loops. From the decision blocks, decision trees can be created that further detail the decision making process, illuminating the specific information required through the decision making process [113].

The result of the DCAP hCTA process was a set of functional and information requirements that guided the development of three different display configurations[5]. In the Information Display configuration, the display serves as an information acquisition and display tool to support operator monitoring of the system and the decision on when to create a new schedule ("replan"). This configuration directly supports the operator in the *Data Analysis +Request* step. The second configuration, the Plan Creation configuration, allows the operator to specify inputs and constraints for the new schedule to the algorithm, supporting the operator in the role of Moderator. The third and final configuration, the Proposal Review configuration, supports the operator in their role as the Decider, while also allowing the user to return to the Moderator stage to alter or to provide additional inputs to the algorithm. The following subsections will address each of these display configurations individually.

3.2.1. Information Display Configuration

The Information Display configuration is the main configuration of the interface (Figure 9). The *Carrier Display* shows the current location of all vehicles and crew on deck. This frame can show either a close-up view of the deck (Figure 10), or a zoomed

[5] Details concerning the DCAP hCTA can be found in Appendix B. A tutorial on using the interface can be found in Appendix C.

out view for monitoring flight operations (Figure 11). The *Marshal Stack Display* (Figure 9) shows the current list of aircraft waiting to land and the landing order. Individual aircraft schedules appear in the *Aircraft Schedule Panel* (ASP), a vertical list on the right side of the screen (Figure 9). The *Deck Resource Timeline* (DRT) at the bottom of the screen shows the allocation of tasks for the four catapults and the landing strip (Figure 9). These two sections of the interface also convey information on aircraft and deck resource failures to the user. The remaining features of the interface are supporting features, such as sort options and legends. The user initiates replanning by pressing the "Request Schedule" button at the upper right corner of the screen. This shifts the display interface to the Plan Creation Configuration.

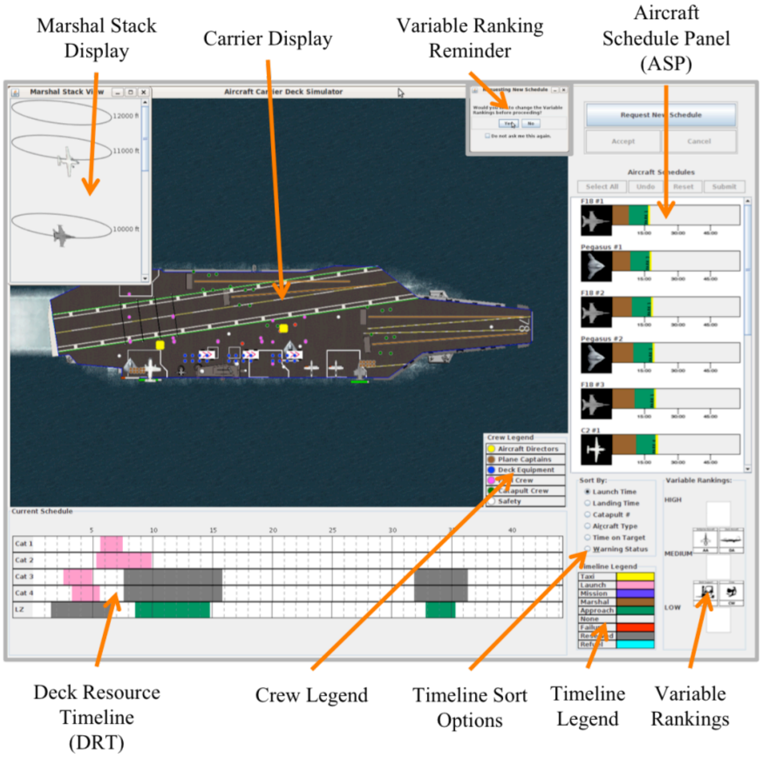


Figure 9. Information Display Configuration of the DCAP Interface

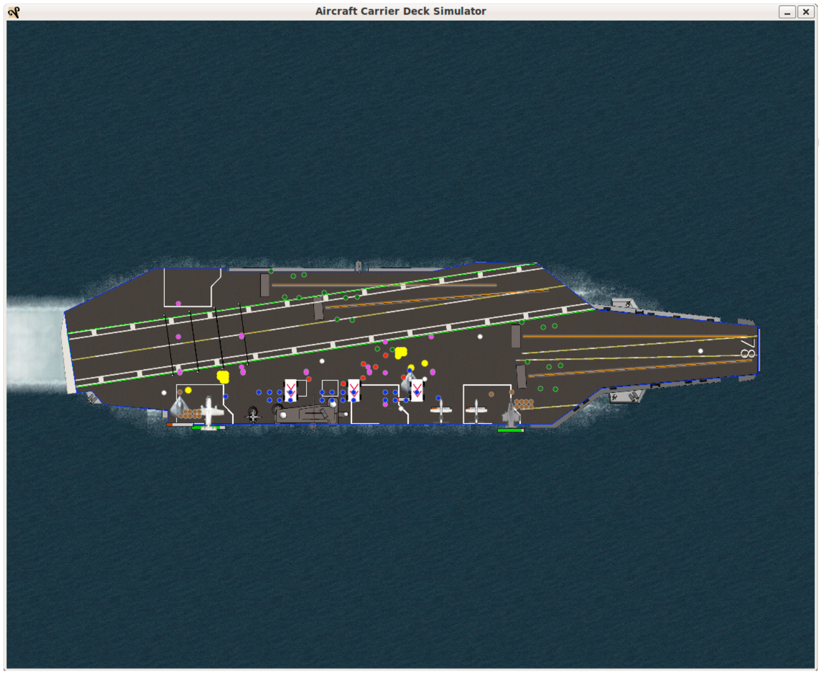


Figure 10. Full image of Carrier Display, "Deck View."”

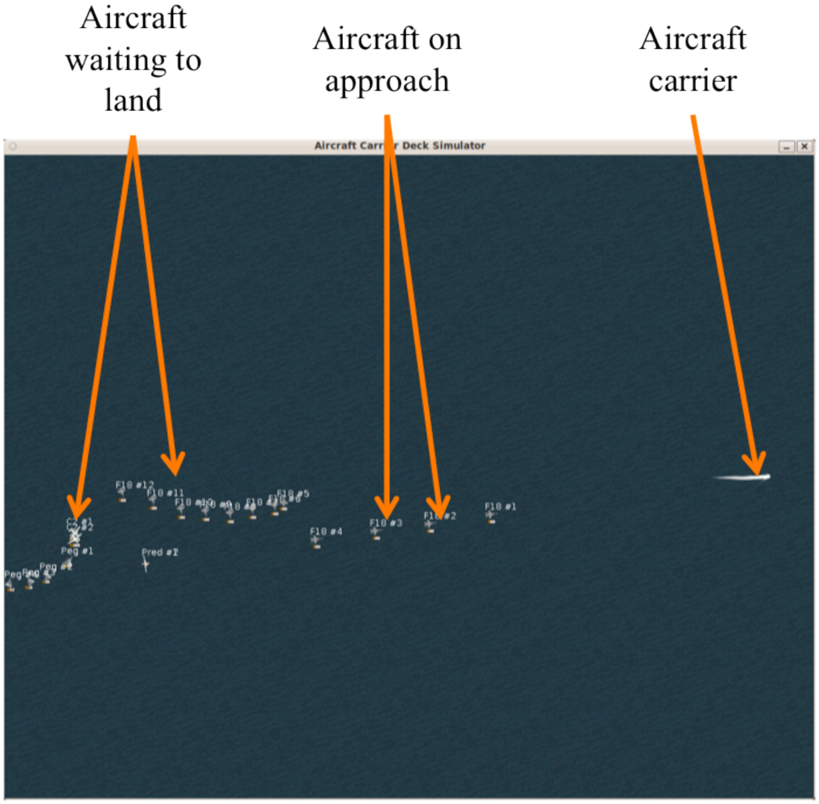


Figure 11. Full image of Carrier Display, "General Overview."

3.2.2. Plan Creation Configuration

The Plan Creation configuration allows the user to define weights for the planner's objective function as well as a set of additional constraints on the solution. The creation of objective function weights is done by ranking the relative priority of a set of personnel groups within the environment (e.g. deck aircraft or crewmembers). For example, in the aircraft carrier environment, the mission focus alternates between launching and landing aircraft. Planning priorities can be continually adjusted to reflect these changes. At other times, concerns for the workload of the crew and support vehicles on deck may arise and further modify mission priorities. Having a single, consistent definition of priority levels does not effectively capture the complexity of the environment.

Constraints are created by assigning priority ratings to specific aircraft, then defining a desired schedule for each aircraft. These two actions (ranking personnel groups and assigning aircraft priority designations) can be done in any order, but both must be done before the automated planner can begin its computations. This section will describe the ranking of the personnel groups first, and then will describe the definition of individual aircraft priority.

3.2.2.1. Relative Priorities in the Variable Ranking Tool

Entering the Plan Creation Configuration first allows the operator the option to bring up an additional frame – the *Variable Ranking Tool* (Figure 12) – to define a set of priorities for four personnel groups on deck. These four groups are defined to be Airborne Aircraft (AA), Deck Aircraft (DA), Crew Working on deck (CW), and Deck Support vehicles (DS).

The goal of this ranking process is to allow the operator to specify the relative importance of each of these groups to the planning algorithm and is done via a drag-and-drop interface using five levels of priority. Including this as a step in the plan creation process allows the operator flexibility in modifying the algorithm's objective function in light of changing conditions on the aircraft carrier deck. This ranking can occur in any manner the operator desires – placing variables on separate levels, all on a single level, and any variation in between. The level of ranking corresponds to a numerical weight for the objective function – the highest ranked variables receive a weight of 5, the lowest ranked receive a weight of 1. The operator clicks "Submit" to save the personnel group rankings and transmit them to the algorithm.

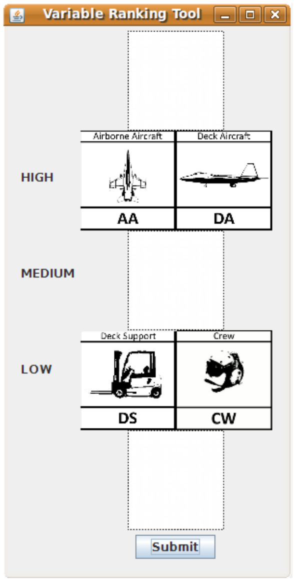


Figure 12. Ranking interface for the four system variables.

3.2.2.2. Individual Priorities in the Aircraft Schedule Panel

This configuration also allows the user to specify aircraft-specific constraints in the *Aircraft Schedule Panel* (ASP, Figure 12). Pressing "Request Schedule" causes checkboxes to appear next to each aircraft box in the ASP. Clicking a checkbox designates the corresponding aircraft as a priority consideration for the automated planner, as seen in Figure 13. Additionally, this action causes the aircraft timeline to split horizontally into halves. The upper half displays the aircraft's current operating timeline, while the bottom half can be manipulated by the operator and shows a projected schedule based on the operator's preferences. Figure 13 provides an example of an aircraft that has been designated as having priority status (the checked box on the left) with a suggestion to significantly delay operations (the bottom timeline has been moved to the right). In this figure, the Predator UAV is about to begin taxi operations on deck in preparation for mission (the upper timeline). In the bottom timeline, the operator has requested that this aircraft delay these operations for an additional 15 minutes.

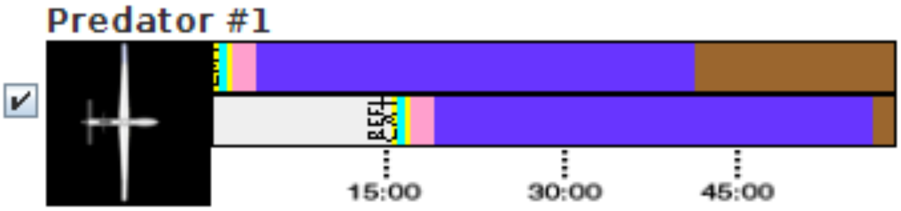
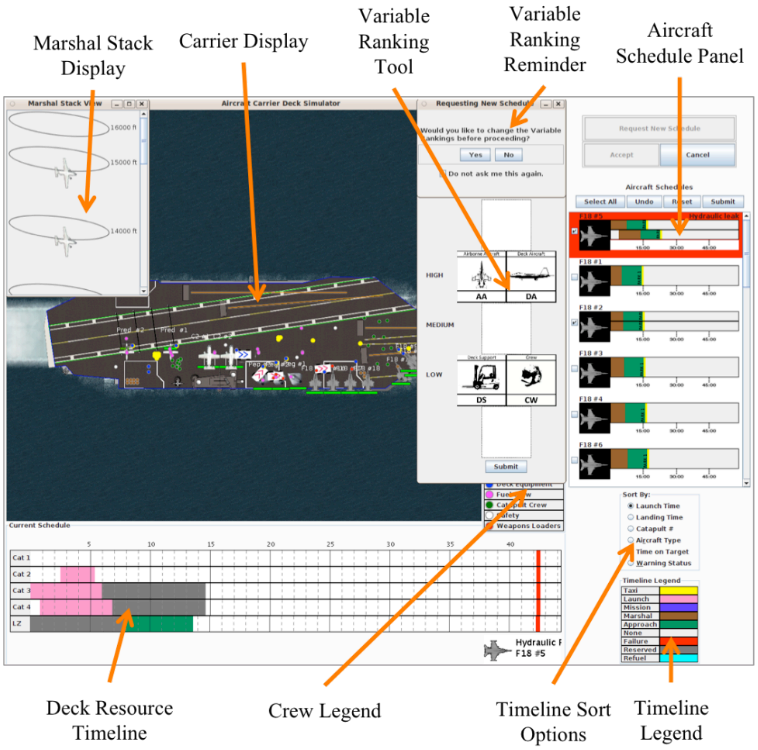


Figure 13. Example of priority definition and suggestion of an operating schedule for an aircraft in the ASP.

The operator has the flexibility to specify as many or as few priority aircraft as desired, and may adjust the schedules of all or none of these aircraft. Once all desired changes are made, the operator presses “accept” to submit this information to the automated planning algorithm. The inputs from the VRT and ASP are then utilized simultaneously. The overall schedule is optimized according to the weights from the VRT while also satisfying the constraints on aircraft schedules. After the automated planning algorithm has finished its computations, the proposed schedule is returned to the display elements and shown in the Proposal Review Configuration, discussed in the next section.

3.2.3. Proposal Review Configuration

After finishing its computations, the automated algorithm returns a proposed schedule to the system to be displayed for operator approval (Figure 14).



Would you like to change the Variable Rankings before proceeding?

Figure 14. Proposal Review configuration.

The proposed schedule is shown using modifications of the basic display elements. The convention used is similar to that of the Plan Creation Configuration, in which the human operator is allowed to make suggestions in the lower half of each aircraft's timeline while the upper continues to show the current operating schedule. In the Proposal Review Configuration, aircraft timelines in the ASP remain split into upper and lower halves. The upper still continues to show the current operating schedule, but the lower now shows the algorithm's proposed schedule for this aircraft. Additionally, a second Deck Resource Timeline appears below the first, utilizing the same convention – the upper timeline shows current operations while the lower shows the proposed schedule. This allows the human operator to easily identify the differences between current and proposed schedules for each of these timelines.

An additional display window in this configuration is the *Disruption Visualization Tool* (DVT, Figure 15). This configural display [114-116] displays comparisons of active operating time for the four variable groups (Airborne Aircraft, Deck Aircraft, Crew Working, and Deck Support vehicles). Each quadrant of the diamond maps to the ratio of active time for the proposed schedule to the active time required for the current schedule for an individual variable group. The dashed line denotes a ratio value of one – no change occurred between the proposed and current schedules. Lower ratios (smaller green triangles, whose edge is inside the dashed line) imply that the algorithm was able to schedule tasks for that group more efficiently. Higher ratios (larger red triangles, whose edge is outside the dashed line) imply that the algorithm was unable to do so, due either to operator specifications or a degraded system state (such as an accumulation of delays in the schedule). For the image in Figure 15, the proposed schedules for both the Airborne (upper left) and Deck Aircraft (upper right) are more efficient than the current schedules. The proposed schedule for the Crew (bottom left) is much less efficient, while the proposed schedule for the Deck Support vehicles is only marginally less efficient. This display does not include error and warning messages, such as exceeding the total acceptable working time, and is only meant to provide a simple, easy-to-understand depiction of relative cost of the new plan. Such warning and alerting displays are left for future work.

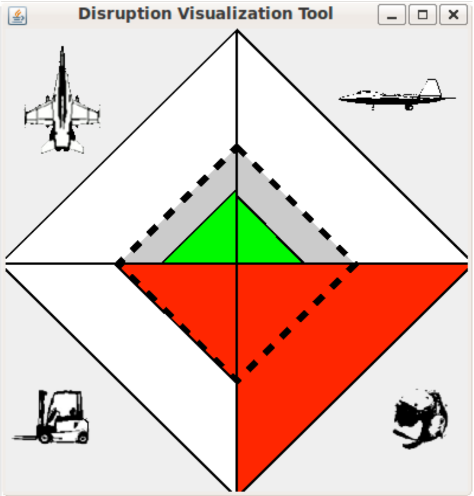


Figure 15. Disruption Visualization Tool (DVT).

The goal of the Proposal Review Configuration is to display sufficient information to the operator to determine whether the proposed schedule should be accepted, modified, or rejected. When the user decides that the plan is worthy of acceptance, the proposed schedule and the system reset to the Information Display configuration. The preceding sections have discussed the actions taken by the operator, but have not discussed the automated algorithm and how it handles these inputs. A brief discussion of this is provided in the next section.

3.3. THE AUTOMATED PLANNER

The current automated algorithm in use in the DCAP system[6] is an Integer Linear Program (ILP) [45]. Generally, Linear Programming (LP) algorithms function by minimizing a given cost function while simultaneously satisfying a set of constraints defined for the problem space. The cost function is generally a summation of a set of variables, each assigned a different scoring weight. Constraint satisfaction is typically modeled by defining an upper bound for several summations (e.g., sum of all *x* should be less than 1). An example ILP formulation appears below:

$$\begin{aligned} &\textit{minimize} \quad c^T x \\ &\textit{subject to} \quad Ax = b \\ &x \geq 0 \end{aligned} \tag{1}$$

where c^T is a matrix of weighting values and x is a matrix of system variables. The variables A and b are matrices used to define constraints in the system. In the case of DCAP, c^Tx is a function that minimizes total operational time. This was selected since minimiz-

[6] The DCAP system is modular with respect to the automated algorithm. Any algorithm can be utilized in the system, as long as it is adapted to accept the appropriate inputs and outputs. Future testing and validation will utilize MDPs and Queuing Network-based policy generators.

ing active time also minimizes fuel consumption (fuel is a limited resource) and a maximization of safety (less time in operations implies fewer chances for accidents to occur)[7]. The matrix c^T is populated by the rankings in the *Variable Ranking Tool* (Airborne Aircraft, Deck Aircraft, etc.). The corresponding entries in *x* contain the total active time of each variable group (i.e., the total man-hours of labor for Deck Aircraft). The matrices *A* and *b* consist of additional weights on *x* and bounds on values, respectively. A constraint applied to a single aircraft's fuel level at landing may take the form of

$$x \geq 0.20 \tag{2}$$

where *A* is equal to 1 and *b* is equal to 0.20 (20%). This constraint dictates that the aircraft's fuel level at landing (a member of *x*) should be at least 20% of the maximum fuel level. This would be an example of a "hard" constraint utilized by the planning algorithm[8].

Inputs from the *Aircraft Scheduling Panel* are used as "soft" constraints on the system. The heavily constrained nature of the system implies that an operator's desired schedule for an aircraft may not be possible, as changes to a single aircraft's schedule could affect the entire system. To account for this, the planning algorithm treats the suggested schedule as a soft constraint – the algorithm attempts to minimize the total difference between the desired task times (as input by the user) and those returned in the new schedule solution. Treating this as a hard constraint would force the system to incorporate

[7] This concern is also reflected in interviews with Naval personnel. When schedules degrade and require replanning, the personnel stated that their main concern is executing aircraft tasks as fast as possible while maintaining safe operations.

[8] Hard constraints should never be violated by the planning algorithm. For instance, if the specification is to remain less than or equal to 0.20, the value should never reach 0.21. Soft constraints are more flexible, guiding the algorithm to a certain objective but not requiring the objective to be satisfied. As noted in the test, one form of this is to minimize a value without placing bounds on the value.

these suggestions *exactly* as specified, which may not be possible due to the complexity of the system and the dynamics of the environment. The user, unable to accurately predict the evolution of the system, would then be suggesting a series of infeasible schedules to the algorithm. Minimizing the overall difference in start times between the suggested and the returned schedule allows the algorithm to provide a feasible schedule that adheres as closely to the original schedule as possible. Although the system does not currently highlight instances of infeasibility to the user, this will be a topic of future work.

A formal testing of the algorithm on several sample problems, as well as comparisons to additional well-known LP solvers, can be found in [45]. While the measures included in Banerjee *et al.* [45] suffice for analyzing the performance of the algorithm on its own, this testing must be repeated once the algorithm is integrated with the P/RA system. This is due both to a change in the specific problem domain, which has characteristics different from the test problem, and to the inclusion of the human operator in the system, whose inputs of priorities and constraints will affect algorithm performance. An analysis of algorithm performance under these circumstances appears in Chapter 5.3.

3.4. CHAPTER SUMMARY

This chapter has provided a description of the simulation environment, described the layout of the interface and how an operator interacts with the system to create a new plan, and provided a brief description of the current automated planning algorithm. The goal of this chapter was to describe the characteristics of the environment and the methods of human interaction in order to motivate the metrics and testing program developed for the DCAP system. The following chapter describes the definition of these metrics.

4. Performance Validation Testing

In focusing on validating the performance of the system, and specifically the adequacy of the algorithm in supporting the system, a comparison between the DCAP Human-Algorithm (HA) planning system and its real-world counterpart is required. This comparison is difficult to perform due to limitations in access to detailed operational logs in the aircraft carrier environment, as well as differences between the simulation environment and real-world operations. This latter confound is difficult to avoid, as DCAP is a revolutionary system – it has no real-world predecessor and thus must be modeled in a simulated environment. However, a comparison to real-world operations can still be performed by comparing the performance of DCAP's HA-generated plans to plans generated by real-world Subject Matter Experts (SMEs) that work in the aircraft carrier environment. These SME-based plans are generated without the assistance of the planning algorithm and are referred to as Human-Only (HO) plans. By executing these HO plans within the DCAP simulation environment, the decision-making strategies of the users are preserved while also ensuring that these plans operate under the same environmental constraints and limitations (due to the simulation environment) as the HA-generated plans.

This chapter will first provide an overview of the testing protocol, which utilizes a single Expert User who applies a set of SME heuristics to guide his or her interactions with both the HO and HA planning conditions. This section will also provide definitions for the scenarios used in the testing program, as well as a statistical power analysis to determine the number of required trials. The final sections detail the measurement metrics defined for the system and the testing apparatus utilized in this experimental program.

4.1. Testing Protocol

In order to compare the performance of the DCAP system to the human-generated, SME-based plans, a series of realistic test scenarios was created. As one major purpose of the DCAP system is to replan schedules in the case of a disruptive failure of aircraft or deck resources, each scenario included at least one form of failure. Additionally, as testing across varying complexity levels is an important aspect of algorithm validation [24, 25], three scenarios addressing different levels of complexity were designed and are discussed in Chapter 4.1.2. Applying the human-generated, SME-based (the Human-Only, or HO, planning condition) planner and the DCAP planner (the Human-Algorithm, or HA, planning condition) to these scenarios allows for a relative comparison performance of the two, but provides no objective comparison point to ground the analysis. A third planning condition – the no-replan Baseline condition, B – provides this perspective. In this planning condition, each scenario happens as scheduled without replanning. This provides an objective, independent measuring point for establishing planner performance and internal validity[9] within the testing scenarios. In the case of the latter, there is no guarantee that the Baseline schedules, as designed, are near optimal. If the Baseline schedules are not near optimal, the possibility exists that the HO and HA planners may submit schedules that outperform the Baseline in critical metrics. Measuring all three cases allows analysts to determine the level of validity of the Baseline cases as designed, and poor results may lead to changes in the testing scenarios.

[9] An experiment exhibits internal validity if the tests performed truly measure the variables of interest and the results cannot be produced from other spurious, uncontrolled factors.

In conducting these tests, the inclusion of multiple users – even if all are guided by the same SME planning strategies – causes a confound in the examination of the performance of the planning algorithm. In this case, it becomes difficult to analyze the performance of the algorithm on its own, as variations in user input and strategies may directly cause variations in algorithm performance. The utilization of a single individual minimizes the variability in interaction that would be seen with a large group of human test subjects and allows for a more precise inspection of algorithm performance in the DCAP system. Even so, a single individual's actions may vary among different trials. In order to remove these variations, the Expert User's actions were scripted and based upon a defined set of SME Heuristics, developed from interviews with Naval personnel. These are detailed in the following section.

4.1.1. Subject Matter Expert Heuristics

Throughout the design process of the DCAP system, a variety of Naval personnel were consulted. This included over two dozen individuals, encompassing former Naval aviators, a former member of an Air Wing Commander's planning staff, and two commanders of a training base for deck crewmen. In meetings that occurred in person, participants were presented with example scenarios that could occur in real-life operations and were asked what their responses to the situations would be. Through these guided interviews, the DCAP research team was able to identify relative consistency in solution generation despite a lack of standardized training for replanning carrier operations [117]. These rules, or heuristics, are shaped by human experience and are used to simplify the problem at hand, allowing users to come to solutions quickly [13, 118, 119]. The list of heuristics appears in Table 6, grouped according to three general categories (Generic, Deck, and Airborne) but not in order of importance.

Table 6. Aircraft Carrier expert operator heuristics.

General	Deck	Airborne
(1) Minimize Changes (2) Cycle aircraft quickly, but maintain safety (3) Halt operations if crew or pilot safety is compromised	(4) Maintain an even distribution of workload across the deck (5) Make available as many deck resources as possible (6) When moving aircraft on deck, maintain orderly traffic flow through the center of the deck	(7) Marshal Stack populated according to fuel burn, fuel level, then miscellaneous factors (8) Park aircraft for maximum availability next cycle (9) "True" vs. "Urgent" Marshal Stack emergencies

General heuristics are applied to any and all replanning scenarios. These *General* heuristics include – minimize changes in the schedule (Heuristic 1), work quickly and safely (Heuristic 2), and halt operations if any human being is placed in immediate physical danger (Heuristic 3).

For *Deck* heuristics, the concerns are to balance workload on deck (Heuristic 4) due to concerns of crew workload and the maintainability of the deck equipment, to ensure maximum flexibility in operations by keeping all resources available, if possible (Heuristic 5), and to keep orderly motion on the deck by focusing movement in the interior of the deck (Heuristic 6).

Airborne heuristics deal with the ordering of aircraft in the landing order (Heuristic 7), where they should be parked after landing (Heuristic 8), and how to handle failures for airborne aircraft (Heuristic 9). Applying Heuristic 9 to an airborne aircraft requires understanding the nature of the failure and its criticality. True emergencies must be dealt with immediately, as they endanger the pilot and the aircraft. Urgent emergencies are of concern, but if compensating for these failures causes further schedule degradation or requires numerous changes on deck, operators may delay action until a more satisfactory time.

These expert heuristics were reviewed by the previously-interviewed Naval personnel in the form of a teach-back interview [120]. That is, the interviewees were presented with a problem scenario, to which the interviewer applied the heuristic in question. The interviewer would describe the heuristics and what their resulting plan would be. The interviewee would then validate proposed action, possibly suggesting further details or a slight differentiation in the heuristic. The final set of heuristics thus allows a non-expert user to generate approximately the same solutions as a more experienced subject matter expert.

4.1.2. Scenario Definition

Complexity is known to cause variations in algorithm performance, due in part to the brittleness inherent to automated algorithms [9]. As such, testing across a range of complexity is considered a necessity for the full validation of an algorithm [24, 25], even if this only establishes bounds on algorithm operation. The complexity of a system can be described either objectively (through some standardized, logical method) [121-123] or subjectively (through the views of separate individuals) [124-126]. Scalability [127], particularly load scalability, can also be used as a form of complexity for the system. This involves testing over a range of load sizes (for DCAP, the load size is the number of aircraft). However, due to physical space constraints, the aircraft carrier environment has a hard upper bound on the number of aircraft (as well as crew and deck support vehicles) that can exist at any given time. Subjective evaluations also may vary widely; therefore an objective description based on the number of applied SME heuristics was used in order to provide a stable, common definition of complexity. The following subsections describe the three scenarios defined for the testing process (Simple, Moderate, Complex) and list the applicable heuristics required for each (a total of 4, 5, and 7, respectively). A description of the actions taken by the user in replanning for these scenarios appears in Appendix D.

4.1.2.1. Simple Scenario

The Simple scenario models the occurrence of a catapult failure on deck during launch operations and has four applicable expert user heuristics, detailed below. Twenty aircraft (2 SMAC, 2 SUAV, 12 FMAC, 4 FUAV) are fueled and have weapons loaded while parked on the deck. Aircraft then proceed to launch catapults, queuing in lines of no more than three (similar to real operations) at each launch catapult. After launching from the catapult, aircraft proceed to a mission area.

Aircraft launch assignments are initially distributed across catapults. Catapult 1 remains inaccessible for the entirety of the scenario due to several aircraft parked in the immediate vicinity. Exact times are not predictable due to the stochasticity in processing times of fueling and launching aircraft, as noted earlier in Chapter 3. While estimates of mean time and standard deviation for each these Gaussian processing times can be obtained and summed to form a new Gaussian model, additional variability and stochasticity exists due to the route planning system. The route planner's actions are guided by the location of aircraft at each point and cannot be adequately modeled as a Gaussian distribution. As such, the processing times can be highly variable and do not exhibit a standard distribution form.

A failure occurs in the system 555 seconds after simulation start (total scenario length approximately 1800 seconds), occurring in the window of time after the launch of the SUAV aircraft from Catapult 4, but before the launch of the SMAC aircraft from Catapult 3. This failure incapacitates Catapult 3 for the remainder of the simulation. Replanning after this failure should address the reassignment of aircraft to the remaining accessible and operational catapults. The scenario terminates when all aircraft have departed the carrier deck and reached their mission location.

This scenario is identified as Simple as replanning for the system requires the application of only four expert user heuristics. Heuristics 1 (Minimize changes to the schedule) and 2 (Cycle aircraft quickly, but maintain safety at all times) apply to most situations. Additionally, Deck heuristics 4 (Maintain an even distribution of workload on the deck) and 6 (When moving aircraft on deck, maintain orderly flow through the center of the deck) also apply. This results in SMEs moving all aircraft from the failed Catapult 3 forward to Catapult 2 while also attempting to balance the number of aircraft at the two remaining functional catapults (Catapults 2 and 4). The naïve user action might simply move aircraft to the closest catapult; in this case, aircraft at the failed Catapult 3 would be sent to Catapult 4. This not only overloads Catapult 4, but it is also more difficult for the crew to manage the turning and movement of the aircraft aft than to taxi forward. In the minds of the SMEs, moving the aircraft forward minimizes the complexity of and risk associated with reassigning aircraft catapult assignments on the deck.

4.1.2.2. Moderate Scenario

The Moderate scenario involves the application of five expert heuristics and models a recovery task. In this scenario, all aircraft begin at their mission location and immediately begin returning to the Marshal Stack to land. This scenario also utilizes twenty aircraft (2 SMAC, 2 SUAV, 12 FMAC, 4 FUAV), which are timed to enter the Marshal Stack with a very tight spacing based on Rule 7 in the expert heuristics (populate the Marshal Stack according to fuel burn rate, fuel level, and maintenance requirements). FMAC aircraft entered first, followed by SMAC aircraft, then FUAV aircraft, then SUAVs. Two failures are introduced just before aircraft enter the Marshal Stack – an FMAC has a hydraulic failure, while an SMAC has a fuel leak. Replanning should lead to a reordering of aircraft in the Marshal Stack that ensures both aircraft land before encountering a limit violation on their hydraulic fluid and fuel, respectively. Replanning for this scenario should also address Rules 1, 2, 3 (The safety of pilots and crew overrides all, even if it requires stopping operations momentarily) and 9 (Differentiate between "True" emergencies, which must be handled immediately, and "Urgent" emergencies, which could be delayed if needed). In this case, the SMEs move the SMAC (fuel leak) forward in the Marshal Stack to minimize the risk of this aircraft running out of fuel. However, the nature of the hydraulic failure increases the possibility of the FMAC crashing on landing. This would disable the landing strip for an extended period of time while crew cleared the wreckage and prepared the landing strip for operation. Moving the FMAC backwards in the Marshal Stack allows for additional aircraft to land and thus minimizes the potential repercussions of a crash, if one occurs. The naïve user may not understand this constraint and may instead send both failed aircraft forward in the Marshal Stack, increasing the chance of causing major disruptions in task execution for the remaining aircraft.

4.1.2.3. Complex Scenario

The Complex scenario models aspects of a mixed launch/landing event that requires the application of seven expert user heuristics. The two previous test scenarios focused on only one aspect of the launch and landing (recovery) of aircraft in the aircraft carrier environment. The Complex scenario focuses on both aspects, addressing a case where emergency launches are requested in the midst of landing operations. This scenario begins similarly to the Moderate scenario, with twenty aircraft (2 SMAC, 10 FMAC, 6 FUAV) returning from mission. The order of entry is slightly different from that of the Moderate scenario; here, FUAVs enter the Marshal Stack first, followed by FMACs and SMACs. In the midst of return operations, a supervisor requests the launch of additional reconnaissance aircraft. Also, aircraft begin the scenario with lower fuel levels as compared to the Moderate scenario, which greatly increases the chances of encountering low fuel emergency conditions in this Complex scenario.

In this case, two additional SUAVs launch from the flight deck. In launching these aircraft, only Catapults 2, 3, and 4 are available (just as in the Simple Scenario, aircraft are parked over Catapult 1, making it inaccessible). Just as this request is fielded, a fuel leak arises in a SMAC just arriving in the Marshal Stack. This creates conflicting priorities for scheduling – the Carrier Air Wing Commander (CAG) has requested that these aircraft be launched immediately, but the fuel leak must also be addressed relatively quickly. However, the use of Catapults 3 and 4 may lead to conflicts with aircraft incoming to land. Addressing this scenario will require the application of heuristics 1, 2, 3, 4, 6, 7, and 9 (Table 11). The naïve user solution in this case may be sending one aircraft to separate forward and aft catapults (e.g., to Catapults 2 and 4). While this may guarantee a faster launch time, any potential delays in the launch at the aft catapult increase the likelihood of an aircraft on approach being forced to abandon landing. This incurs greater fuel cost and increases the risk associated with this aircraft. The SME solution requires two actions – moving the failed SMAC forward in the landing order (to minimize the chance of running out of fuel) and sending the launching aircraft to the forward catapult (Catapult 2) *only*. Utilizing only this catapult ensures that, regardless if the time required to launch, aircraft on approach do not experience any interference in the landing strip area. In this case, efficiency in launching is sacrificed to minimize the risk for the airborne aircraft.

4.1.3. Statistical Power Analysis

Power tests performed for the DCAP system resulted in a sample size of 30 trials per planning condition and scenario combination. This resulted in a total of 270 required trials (30 trials x 3 scenarios x 3 planning conditions), with two-thirds of these requiring direct intervention by a human operator. The HO planning condition requires an individual to apply the SME planning heuristics to the scenario, while the HA planning conditions requires an individual to interact with the DCAP planning algorithm to affect a replan in the system. The final third (the Baseline plan) represents a nominal schedule with no failures or need to replan.

4.2. Definition of the Metrics

The five metric classes developed by Pina *et al.* [26, 27] provide the framework for defining metrics to measure the performance of the DCAP system. These originally addressed measures of Mission Efficiency, Algorithm Behavior Efficiency, Human Behavior Efficiency, Human Behavior Precursors, and Collaborative Metrics. As the examination of the multiple user case is outside of the scope of this research, several metric subclasses were removed. Metrics still exist for four of the five classes defined by Pina *et al.* (only Collaborative metrics are removed entirely, due to the use of only a single Expert User who is intimately familiar with the system). Detailed definitions of these metrics will be discussed in the following subsections.

4.2.1. Mission Efficiency Metrics

Pina *et al.* divided mission performance metrics into three categories of time-based, error-based, and coverage-based metrics. For the aircraft carrier environment, these respectively address the Expediency with which operations occur, the level of Safety under which they occur, and the Efficiency of task performance. The full list of Mission Efficiency measures appears in Table 7 and is discussed in the subsequent subsections.

Table 7. DCAP Mission Performance Metrics.

Safety (Error-based)	Expediency (Time-based)	Efficiency (Coverage-based)
• Number of limit violations o Fuel o Time ▪ Pilot ▪ Crew • Foul Deck Time for the Landing Strip • Time to recover emergency aircraft • Safety Margin on landing o Time to fuel threshold (e-fuel or zero fuel)	• Total Time on Deck o Aircraft in transit o Crew o UGVs • Mission Duration • Delays o WTCrew o WTQ-Catapult o WTQ-MS	• Excess capacity/ overload measurement • Resource Usage Rates

4.2.1.1. Safety (Error-based) Metrics

The Safety category includes measures tracking various possible failure and error conditions within the system. There are two sources of explicit failures in the system – aircraft-specific failures and deck resource failures. Failures for the deck concern the failure of specific items, such as Catapult 1. Aircraft failures occur for one or more aircraft in the system, currently modeled as some form of fluid leak. Table 8 includes a list of possible failures currently modeled in the system.

Table 8. Possible Failures in the DCAP Simulation

	Failure		
	Catapult Failure	**Aircraft Fuel Leak**	**Aircraft Hydraulic Leak**
Entities Affected	Deck aircraft assigned to that catapult	Individual aircraft	Individual aircraft
Duration	Few minutes to permanent	Until repaired or fuel reaches zero	Until repaired or hydraulic fluid level reaches zero
Solution requires…	New catapult assignments for affected aircraft	Lading the aircraft before fuel level becomes critical (<20%)	Landing the aircraft before hydraulic fluid level becomes critical (<20%)

Because fuel and hydraulic fluid are finite resources, the occurrence of a leak requires that action be taken to land the affected airborne aircraft before these fluid levels reach a critical state. Such aircraft are labeled emergency aircraft. These critical states are defined as a 20% of the maximum fluid level. Breaching either of these thresholds (fuel or hydraulic fluid) is termed a *limit violation*. While the planner cannot control the occurrence of these failures, the subsequent schedule correction should minimize the occurrence of limit violations (a hard constraint on the planner).

In establishing the performance of planning corrections for aircraft experiencing failures, three values can be calculated. The first is the difference between the time of landing and the time of the failure – the Emergency Aircraft Recovery Time (EART), which should be minimized. Additionally, the remaining fuel/hydraulic fluid levels and the total flight time remaining may also be calculated. The latter values serve as diagnostics of the relative robustness of the new plan, depicting how much buffer time was afforded by the solution. These metrics should be statistically correlated, in that minimizing EART should maximize the level of fuel/hydraulic fluid remaining at landing. The remaining fluid level can also be used to determine the remaining excess flight time, describing the amount of time the aircraft could have spent in flight before a new schedule was required. This third value is likely also statistically correlated to the first two. While a single value could suffice for statistical analysis, the inclusion of all three provides additional diagnostic value for the system.

An additional, non-explicit error condition also exists in the D-CAP simulation. At certain times, crew or aircraft may move into the landing zone (LZ) during operations, which results in a fouled deck condition for the landing strip. In this state, no aircraft may land. Higher values of LZ Foul Time result in increased likelihood of an aircraft being "waved off" and forced to return to a holding pattern. If this occurs while an aircraft experiencing a fuel or hydraulic leak is attempting to land, the potential for losing the aircraft and pilot increases significantly. Thus, while this Foul Time is not a direct failure, higher values induce greater probabilities of failures into the system. This value should be minimized for recovery scenarios.

The focus of this section has been on error-related metrics addressing safety, noted as a priority by stakeholders interviewed during this research. This overriding priority led to the classification of several metrics as error-based metrics, even though they are time-based, because of their overriding safety value. There are several additional time metrics used in this study, used primarily as diagnostic measures of efficiency. These are discussed in the following section.

4.2.1.2. Expediency (Time-based) Metrics

Time-based measures for the DCAP system address the expediency with which operations are performed. Within the aircraft carrier operations environment, minimizing the time required to perform actions minimizes total aircraft fuel consumption while also minimizing risks to the crew, aircraft, and ground vehicles active in the system[10].

As a whole, measures for expediency have been previously used as measures of overall mission performance and as diagnostic measures for subcomponents and subtasks in the system. For the DCAP system, both forms are utilized. The overall Mission Duration is calculated as a measure of overall performance. It is defined as the elapsed time from the start of the simulation to the point that a terminal end condition (based on the scenario definition) is reached. Ideally, the system should execute the schedule in a minimum amount of time, launching aircraft as quickly as possible from the deck, or maximizing the rate at which airborne aircraft are allowed to land.

[10] It is the belief of the Naval personnel interviewed in this research that decreasing the active time of aircraft, crew, and support vehicles decreases the cumulative probability of that entity experiencing an accident.

For the expediency measures, “Active time” is defined as the total amount of time any person or vehicle is actively engaged in a task on the carrier deck. For example, an aircraft accumulates active time while it is fueling or is taking off, but not while it is parked or otherwise idle. These values can be calculated for individual aircraft, crew, or deck support vehicles, as well as summed for the entirety of each group. For crew, lower active time values equate to a lower likelihood of injury and a lower level of fatigue. The same is also true for ground vehicles, although injuries and fatigue are replaced by maintenance issues and fuel constraints, respectively. For aircraft, lower active times imply lower fatigue for the pilots and less fuel consumption, as well as a lowered risk of possible collisions. Aircraft are also given measures of Taxi Time, denoting the amount of time aircraft are engaged in taxi tasks, including time spent waiting for crew or clearance to move. This is a subset of the Active Time for aircraft and is included as a diagnostic measure to determine how well the system has allocated movement tasks on the deck. For all of these measures (individual or collective), lower values are desirable.

Additionally, three metrics measuring system delays are included, influenced by Cummings and Mitchell’s wait times for human interaction [35]– Wait Time in Queue at Catapult (WTQC), Wait Time in Queue for Crew (WTQCrew), and Wait Time in Queue for Marshal Stack (WTQMS). These track the total wait time aircraft incur while waiting in the processing queue at a catapult, waiting for a crewmember to arrive and perform a task, or waiting in the marshal queue for landing clearance. Higher values of WTQC and WTQCrew imply that aircraft are actively burning fuel while waiting for another aircraft or crewmember to complete a task. Ideally, aircraft would have only minimum wait times, saving fuel. Lower values of WTQMS are also desirable, as this value depicts the total time aircraft are in holding patterns waiting to land, consuming limited fuel resources. If these times are too high, pilots and aircraft are in danger of having insufficient fuel to land on the carrier deck.

The time measurements covered in this section function primarily as diagnostic measures of the efficiency of the holistic system. Additional diagnostic measures can also be incorporated for the deck resources, specifically for the deck catapults. These measures of coverage establish how well tasks were allocated between the four catapults during the course of a simulation and are covered in the following section.

4.2.1.3. Efficiency (Coverage-based) Metrics

Efficiency measures defined for this category address the distribution of tasks in the system, measuring the number of launches at each catapult as well as the launch rate (launches per mission duration). Due to the nature of the deck environment, it is desirable to have a balanced distribution of launch tasks between catapults. Launches cannot occur simultaneously for catapults within each of the forward and aft pairs (within Catapults 1 and 2 or within Catapults 3 and 4, respectively). However, launches can occur simultaneously across pairs (i.e., Catapult 3 may launch while either of Catapult 1 or 2 is launching). This implies that distributing tasks across the catapult pairs may increase launch efficiency. Additionally, an even distribution of launch tasks within a pair of catapults also creates a slight performance gain. If two aircraft are assigned to launch at neighboring catapults, the first aircraft to arrive will immediately begin launch operations. The second aircraft is allowed to taxi onto the neighboring catapult, saving some time in the takeoff process, even though it must wait to begin launch preparations. Due to these characteristics, it is desirable to balance launch tasks between the fore and aft catapult pairs as well as among catapults within a single pair. This assignment strategy should also maximize the overall launch rate of the system. Additionally, these measures of launch rate could also be applied according to queuing theory and establish a theoretical maximum for launch capabilities.

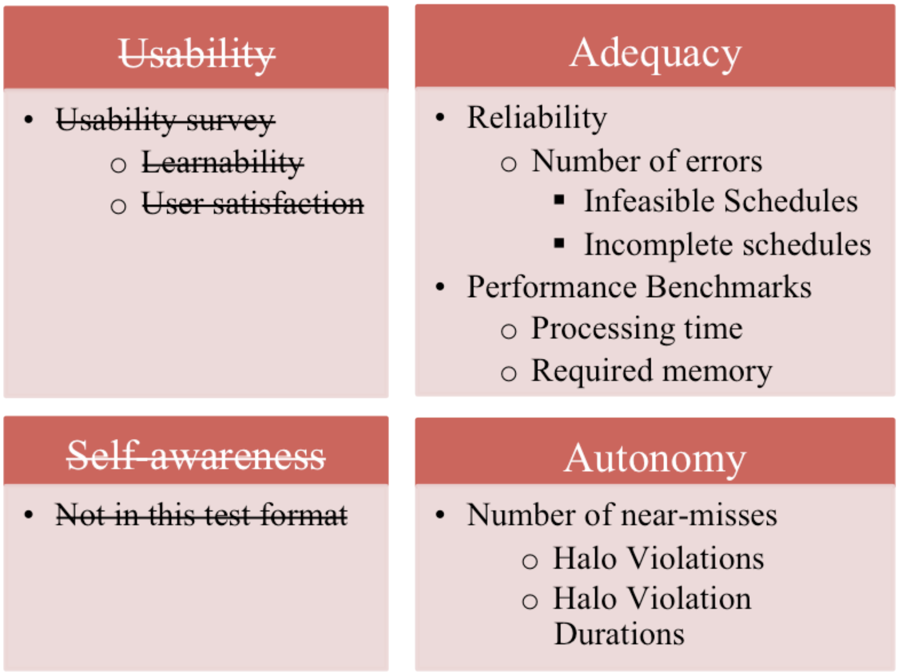
Combined, these measures of Error-, Time-, and Coverage-based efficiency serve to provide descriptive evidence to quantify the differences in performance between planning conditions and diagnostic support necessary for determining the mechanisms that created these differences. However, these measures only apply to mission tasks, and examine the performance of the solution as it is executed. Additional measures are needed to establish the effectiveness of the human operator and the algorithm in the schedule creation process. Measure for the algorithm fall into the class of Autonomous Platform Behavior Efficiency metrics, which are discussed in the next section.

4.2.2. Autonomous Platform Behavior Efficiency Metrics

Measures of Automation Behavior Efficiency address how well the algorithm supports system operations and includes subcategories of Usability, Adequacy, Autonomy, and Self-Awareness [26, 27]. Usability concerns the interaction of the algorithm and the human operator through the system interfaces. Adequacy addresses the computational efficiency of the algorithm, including speed of computation and error rates. Autonomy concerns how well the system works while not experiencing direct human interaction. Self-awareness addresses the capability of the algorithm to examine (and possibly correct) its own performance. The metrics defined for these classes are listed in Table 9 and explained in the remainder of this section. For the purposes of this system evaluation, only Autonomy and Adequacy are addressed, as the remaining two classes (Self-

Awareness and Usability) are not applicable. The algorithms currently in use in the DCAP system are not self-aware, thus this class of measures cannot be used in testing. Usability measures are best applied when using a varied user population; the use of the single Expert User would not give adequate information for judging usability, thus negating the use of this class. However, future evaluations of system usability are planned.

Table 9. DCAP Automation Behavior Efficiency Metrics [26, 27].



~~Usability~~

- ~~Usability survey~~
  - ~~Learnability~~
  - ~~User satisfaction~~

Adequacy

- Reliability
  - Number of errors
    - Infeasible Schedules
    - Incomplete schedules
- Performance Benchmarks
  - Processing time
  - Required memory

~~Self-awareness~~

- ~~Not in this test format~~

Autonomy

- Number of near-misses
  - Halo Violations
  - Halo Violation Durations

Many potential measures of algorithm Adequacy are interdependent with other measures since the quality of the returned solution is not solely dependent on the algorithm. The priority and constraint inputs by the operator affect the final solutions generated by the algorithm, such that neither the operator nor the algorithm is solely responsible for the resulting schedule. However, several measures of algorithm adequacy can be developed. The number of failures in the algorithm should be tracked in order to establish the reliability and stability of the algorithm. Processing time of the algorithm (Wait Time due to algorithm Processing, WTP) is included, as it is often used in analyses of algorithm performance [24, 25].

Measures of Autonomy include metrics that address the efficiency of the embedded vehicle router and track the occurrence and duration of proximity violations that may occur. Figure 16 provides a depiction of the collision avoidance “halo” that surrounds each aircraft. The system tracks the number of times the halo is violated and the duration of time this violation exists. The actual diameter of the halo is forty-four feet, equal to the wingspan of an F-18, the most common aircraft in the Naval fleet. Because the system treats aircraft as individual point masses (physical constraints are not currently modeled), this measurement was adapted to the other aircraft as well. In Figure 16, the crewmember’s (blue dot) act of crossing the halo violation line would increase the violation count by one. As long as this crewman is within the halo area, time is added to the Halo Violation Duration (HV-D) measure. Time is no longer added to the duration measure once the crewmember exits the halo area. However, if the crewmember reenters the halo area, the count is again increased by one and times resumes being added to the duration measure. The addition of time to the duration measure is agnostic of the number of crewmembers within the halo and is sensitive only to the presence of a violation.

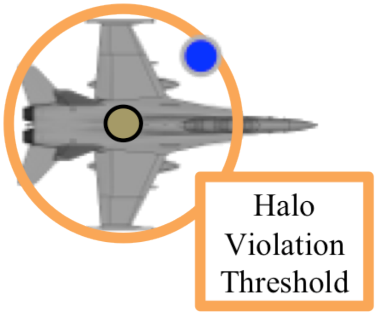


Figure 16. Collision avoidance "halo" for aircraft. Blue dot represents crew, gold dot with black ring signifies environment model of aircraft (point mass).

4.2.3. Human Behavior Efficiency Metrics

Human Behavior Efficiency metrics consider the physical and cognitive aspects of user interaction with the system displays and controls. Pina *et al*. [26, 27] divided these into two subcategories for Attention Allocation Efficiency (AAE, addressing cognitive aspects) and Information Processing Efficiency (IPE, addressing physical aspects). The list of Human Behavior Efficiency metrics applied to DCAP appears in Table 10.

AAE measurements seek to define how the user's attentional resources are allocated during system use. However, because the testing program is utilizing a single Expert User, these measures are not applicable. Future testing programs utilizing a variety of human users should incorporate these measures. IPE metrics, however, may still be applied to the system and measure the efficiency with which the operator inputs commands into the system. In this case, the actions of the Expert User, who is intimately familiar with the system, can be treated as an upper bound for future users. Pina *et al*. [26, 27] differentiated the IPE subclass into measures of Recognition, Decision, Action Implementation, and Task Efficiencies. The low number of decisions that are made in the DCAP system, as well as the use of a single Expert User, negate the first two subcategories of this class. The remaining two classes – Action Implementation and Task Efficiency – address the physical interaction of the user with the system, establishing how effectively the user translates decisions into actions. For DCAP, these metrics were defined to include the number of user interactions (mouse clicks or button presses) during replanning, the distance of mouse cursor travel, and the total time of interaction with the system. For instance, a user that is panicked due to increased time pressure may begin navigating incorrect measures in a rush to complete their tasks. Doing so will result not only in an increased time of activity, but backtracking through incorrect menus will also result in more mouse clicks in the interface and more cursor movement around the screen. Conversely, a user who is carefully deliberating their actions may also exhibit increased activity time, but will likely not exhibit increases in mouse clicks and cursor movement. Ideally, the user would make a decision in minimal time and with minimal actions (mouse clicks) and cursor movements.

Table 10. DCAP Human Behavior Efficiency Metrics [26, 27]

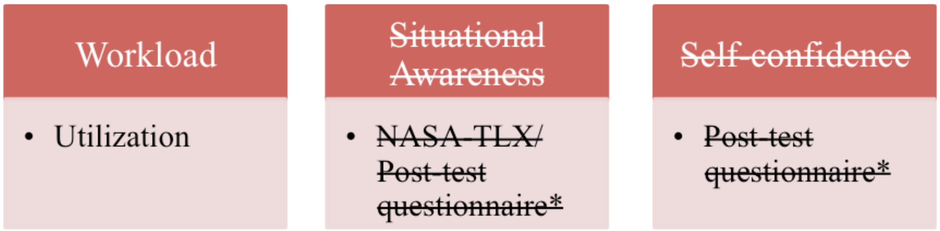
~~Attention Allocation Efficiency~~	Information Processing Efficiency
• ~~Not in this test format~~	• Action Implementation Efficiency ○ Number of interactions ○ Distance mouse cursor travels • Task Efficiency ○ Interaction Time

The measures in this section have described the active performance of the human operator, but have not characterized how the physical environment and the operator’s mental state affect this behavior. These measures of Human Behavior Precursors are found in the following section.

4.2.4. Human Behavior Precursors

Human Behavior Precursors include the psychological and physical states that affect operator performance. Pina *et al.* [26, 27] divided cognitive measures into those that address operator workload, situational awareness, and self-confidence. While a variety of measures of workload can be used [3, 39, 91-94], utilization [89, 95, 128, 129] is a direct, quantitative measurement of the time the user interacts with the system. Measures of utilization require the knowledge of total user interaction time with the system and total time of system execution. Utilization is then a measure of the percentage of total operating time in which the user is actively engaging the system. Higher utilization rates often result in increased mental fatigue on the part of the human operator, increasing the likelihood of errors in decision-making or task execution. This measure primarily involves user physical interaction with the system (e.g., the IPE measures), as detecting cognitive interaction is notoriously difficult.

Table 11. Human Behavior Precursor Metrics
(* requires naïve users).



Workload	~~Situational Awareness~~	~~Self-confidence~~
• Utilization	• ~~NASA-TLX/ Post-test questionnaire*~~	• ~~Post-test questionnaire*~~

As with metrics for Human Behavior Efficiency, the use of an Expert User negates certain measurement subclasses, namely Situational Awareness and Self-confidence. However, measures from these two classes should be included in future research programs that include a varied group of human users.

4.2.5. Section Summary

This section has discussed the creation of metrics for the analysis of the DCAP system across four major categories of Mission Efficiency, Algorithm Behavior Efficiency, Human Behavior Efficiency, and Human Behavior Precursors. Measures for Mission Efficiency address the overall performance of the system as it develops and implements schedules for the aircraft carrier environment. The remaining measures address the effectiveness of the human operator and the algorithm in supporting this process, as well as addressing the effectiveness of their interactions with each other. These measures are the primary mechanisms with which the performance of the DCAP system will be compared to the human-generated SME plans over a series of testing scenarios. These scenarios are based on realistic deck conditions, representing three different levels of complexity within the environment. The following section will detail the creation of the schedules and the determination of relative complexity for each.

4.3. TESTING APPARATUS

Testing was performed on a Lenovo Thinkpad W500 laptop (2.80 GHz Intel Core 2 Duo T9600 CPU, 8 GB RAM, Windows 7 64-bit operating system) using a Logitech M510 wireless RF mouse. The DCAP simulation was run in an Ubuntu 9.10 virtual machine run through VMWare Workstation. Within Ubuntu, the DCAP software (a Java^{TM} application) was executed through the Eclipse Galileo Java^{TM} IDE. Data were extracted by automated features embedded in the Java code. Events were logged at the time of each failure, at the time of replan completion, and upon scenario termination. Scenario termination was also automated to ensure no variation in end conditions. Data files were reformatted into Excel^{TM} spreadsheets, then analyzed in the SPSS^{TM} analytical software package. The results of this data analysis will be covered in the following chapter.

4.4. CHAPTER SUMMARY

This chapter has discussed the performance validation testing of the DCAP system, which focuses on examining the performance of the system in replanning tasks in the aircraft carrier environment. This chapter began with a discussion of the testing protocol defined for this program, including a description of the Expert User used to minimize variations in human input into the system. A first subsection explained the set of Subject

Matter Expert heuristics that would be used by this Expert User in replanning, followed by a subsection defining the three testing scenarios used in system testing, whose complexities were based on the number of SME heuristics required in replanning. A third subsection discussed the statistical power analysis that determined the number of requiring trials for the testing program. The second main section of this chapter defined the metrics used in establishing the performance of the three planning conditions, with individual subsections addressing the main categories of the metric hierarchy. This chapter's final section discussed the testing apparatus used in this experimental program.

5. RESULTS AND DISCUSSION

This chapter provides both quantitative and qualitative analysis of the testing results for the no-replan Baseline (B), Human-Only (HO), and Human-Algorithm (HA) planning conditions. For these three planning conditions and the three test scenarios (Simple, Moderate, and Complex), a total of nine different data sets were generated (gray blocks in Figure 17). This chapter compares the performance of the planners within each scenario (rows in Figure 17) as well as how planner performance varies across scenario levels (columns in Figure 17).

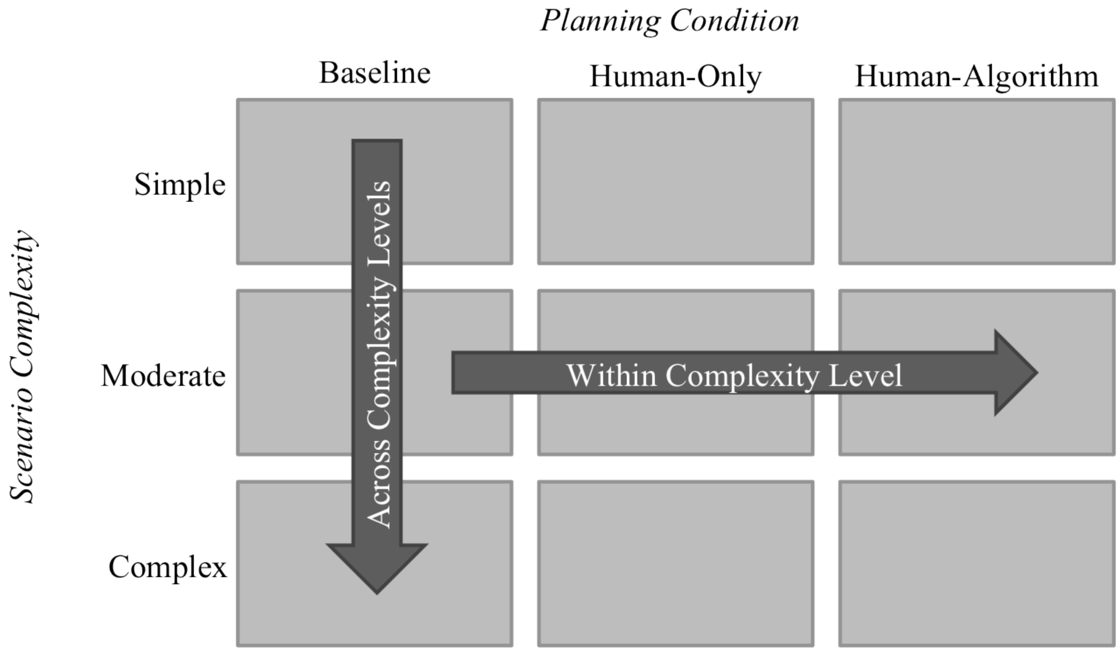


Figure 17. Visual depiction of the analyses performed in this testing program.

In total, thirty-nine different measurement metrics were defined for the DCAP testing program in Chapter 4.2. In using three planning conditions (B, HO, and HA) across three scenarios, a total of 324 possible pairwise comparisons exist. Performing all of these tests would significantly increase the experiment-wise error rate – the chance that the experiment *as a whole* accepts one alternative hypothesis that should be rejected. This risk can be mitigated by reducing the number of individual tests along with lowering the value of the significance level α used in each. An overview of these adjustments is discussed in the first section in this chapter, with the remaining sections presenting and discussing the results of data collection.

5.1. REDUCING FAMILY-WISE ERROR

In performing statistical testing on groups of data, tests can be divided into "families" of tests applied to subsets of independent data. For instance, within the DCAP test program, the data from each of the three test scenarios forms its own subset – tests applied to the Moderate scenario have no relation to the tests performed on data from the Simple scenario. In doing so, the *family-wise error rate*, α_{fw}, is formally defined as the "probability of making *at least* one Type I error in the family of tests when all the null hypotheses are true" [130]. As the number of statistical tests applied to the experimental data increases, the likelihood that at least one test in the family experiences Type I Error (accepting the alternative hypothesis when it should be rejected) also increases. For example, utilizing a significance level α of 0.05 on a single family of statistical tests (each test has only a 5% chance of Type I error) does not imply that the chance of *any* test in this family experiencing Type I error is 0.05. For a study with five statistical tests at an α of 0.05, the likelihood of at least one test experiencing Type I Error (α_{ew}) is 0.23; for 10 tests, 0.40; for 50 tests, 0.92 [131]. Decreasing the number of statistical tests performed or lowering the significance level α for the remaining tests, or a combination of both, will lower this experiment-wise error rate.

In order to decrease the chance of family-wise error in each of the three test scenarios, two filters were passed over the data in order to reduce the number of statistical tests being performed. In the first filter, metrics that were considered to have low external validity[11] were removed from consideration. For instance, Wait Time in Queue due to Crew (WTQCrew) is useful for determining the efficiency of crew allocation, but the HA planner does not plan tasks for the crew at this time. Crewmembers are assigned automatically in the simulation, based on their availability. Additionally, stakeholders in the aircraft carrier environment are less concerned with these measures than they are for other metrics, such as Total Aircraft Taxi Time (TATT). A list of the metrics removed and the reasons for doing so appears in Appendix E.

In the second filter, a Principal Components Analysis (PCA) was performed for each planner-scenario combination, yielding nine correlation matrices which identify statistically related data within a given planner-scenario combination. These matrices were analyzed within scenarios (across rows in Figure 17) in order to identify groups of cross-correlated metrics (sets of metrics that returned Pearson correlations of 0.7 or greater at a p-value of less than 0.001). For each group of metrics that correlated across planning conditions, a single metric was chosen for further analysis based on high external validity. For instance, PCA data for the Moderate scenario revealed that for all three planning conditions, Total Aircraft Active Time (TAAT), Total Active Time (TAT), and Wait Time in Queue in the Marshal Stack (WTQMS) were highly correlated. TAAT was retained, as it has the highest external validity; operators are highly concerned with the

[11] External validity here relates to the how the measure correlates with operators' views of system performance. Interviews revealed that operators do not consider Wait Time in Queue at Catapult, but have a high concern for Total Aircraft Active Time (TAAT). There are also levels of external validity; some measures are highly significant to operators, while others are useful, but not greatly valued.

value of this metric and do not currently calculate WTQMS or TAT. However, this does not imply that the two remaining metrics should be discarded. WTQMS is not calculated by operators currently, but was retained for its insight into the system's efficiency in landing aircraft. Tables denoting PCA results for each planner-scenario combination and the resulting cross-correlations appear in Appendix F. At the conclusion of this process, nineteen metrics remained, a list of which is found in Table 12, along with their applicable testing scenarios.

Table 12. Final list of DCAP Test metrics, with applicable scenarios (check marks, √, signify the test is applicable for the scenario).

Class	*Metric*	*Abbreviation*	*Simple*	*Moderate*	*Complex*
ME-Error	Fuel Violations	FV			✓
	Landing Zone Foul Time	LZFT	✓	✓	✓
	FMAC #6 Hydraulic Fluid Remaining	FMAC #6 HFR		✓	
	FMAC #6 Recovery Time	FMAC_6_EART		✓	
	SMAC #2 Active Time	SMAC_2_AAT		✓	✓
	SMAC #2 Fuel Remaining	SMAC_2_EFR		✓	✓
	SMAC #2 Recovery Time	SMAC_2_EART		✓	✓
ME-Time	Total Aircraft Taxi Time	TATT	✓	✓	✓
	Total Aircraft Active Time	TAAT	✓	✓	✓
	Total Crew Active Time	TCAT	✓	✓	✓
	Wait Time in Queue - Marshal Stack	WTQMS		✓	✓
	Mission Duration	MD	✓	✓	✓
ME-Coverage	Catapult 2 Launch Rate	C2LR	✓		✓
	Catapult 3 launch Rate	C3LR	✓		✓
	Catapult 4 Launch Rate	C4LR	✓		✓
	Total Catapult Launch Rate	TCLR	✓		✓
ABE-Auto	Halo Violations	HV	✓	✓	✓
	Halo Violation Duration	HVD	✓	✓	✓
HBE-IPE	User Interaction Time	UIT	✓	✓	✓
Total number of statistical comparison			34	40	49

As noted in the table, some scenarios did not require certain metrics. Fuel Violations (FV) did not occur in the Simple and Moderate scenarios, but did occur in the Complex scenario; thus, only the Complex case contains a check for FV. The Moderate case did not require any aircraft launches, thus no Catapult Launch Rate measures are required. For the metrics that remain, there typically exists three statistical comparisons to be performed – B vs. HO, B vs. HA, and HO vs. HA. The only metric that does not require three tests is the User Interaction Time metric, because no user interaction was performed in the B cases. This results in totals of 34, 40, and 49 metrics for the three scenarios (Simple, Moderate, and Complex, respectively). Given this number of tests, the comparison-wise significance level for each statistical test, α, can be calculated to preserve an overall family-wise significance level, α_{fw}. These values are found in Table 13.

Table 13. Test significance levels desired for family-wise significance level of 0.05.

Scenario	*Number of tests*	*Family-wise significance level*	*Test significance level*
Simple	34	0.05	0.0015
Moderate	40	0.05	0.0013
Complex	49	0.05	0.0010

ANalysis Of Variance (ANOVA) tests were desired for the statistical comparisons and require that the data exhibit both normality and homoskedasticity. Data were first tested for normality. For data that showed normality, Levene tests for heteroskedasticity were performed, with additional transformations attempted if tests showed heteroskedasticity. For data that tested both normal and homoskedastic, parametric ANOVA tests were acceptable and were utilized in the analysis. For all other cases, non-parametric Mann-Whitney U tests were used to compare distributions. Full outputs of the Kolmogorov-Smirnov tests for normality and Levene tests for heteroskedasticity appear in

Appendix G and Appendix H, respectively. The remaining sections in this chapter present the results of this statistical testing in tabular form; for brevity, boxplots for this data have been placed in Appendix I (Simple scenario), Appendix J (Moderate Scenario), and Appendix K (Complex Scenario).

5.2. Within-scenario Results

This section presents the results of statistical testing within each scenario. Comparisons of performance are drawn between metrics from all three planning conditions (Baseline, Human-Only, and Human-Algorithm), with a focus on the performance of the HA planner. Tables are presented providing the results of statistical testing, with the type of test performed and the resulting significance values reported for each test. Additional tables provide qualitative analyses of the relationships within the data. These two forms of analysis aid in the identification of differences in performance between the HA and HO planners, which are discussed at the end of each section. In these discussions, the metrics are also used to explain differences in the data, revealing not only errors on the part of the planner, but also shortcomings in the operator heuristics.

While there is a large amount of data that can be analyzed, each section focuses primarily on the metrics that support the identification of differences in performance between the HO and HA planners and their subsequent explanation. The remaining measures are noted in the Appendices for completeness, but may not be specifically discussed.

5.2.1. Simple Scenario

The Simple scenario (a launch scenario) included twenty aircraft launched from the carrier deck. During launch procedures, one of the aft catapults failed, requiring the reassignment of launch tasks among the remaining operational catapults. Examination of the performance of the planners in this scenario revealed that the Human-Only plans outperformed the Human-Algorithm plans, but, as expected, neither was able to reach the same level of performance as the Baseline condition.

The primary descriptive metric for performance in this scenario is Mission Duration (MD). Measures for the launch rates of Catapult 2 and Catapult 4 and the total launch rate (C2LR, C4LR, and TCLR, respectively) were used in a diagnostic role, aiding in identifying the causes of the HA planner's poor performance as compared to the HO planner. The following sections cover the statistical testing performed on these metrics, a qualitative analysis of the differences in these metrics, and a discussion of the implications of the results and how they revealed superior performance on the part of the HO planner.

5.2.1.1. Results of Statistical Testing

Table 14 presents a compilation of the statistical testing data for the Simple scenario. Within this table, the metric name and its desired magnitude (High or Low) are presented, followed by columns detailing the results of statistical testing between pairs of planning conditions. These columns list the statistical test applied, the results of the test, and the relative difference between the two conditions. Significant values ($p < 0.0015$) imply that the null hypothesis (h_0: distributions of the two planning conditions are identical) was rejected. The relative difference between two data sets takes one of three forms:

1. The distributions of Planners 1 and 2 were equivalent (h_0 could not be rejected in statistical testing).
2. The median value for Planner 1 was greater than that of Planner 2.
3. The median value for Planner 1 was less than that of Planner 2.

Given these relationships, the superiority of a planner for a given parameter is based on the desired magnitude of the metric. For instance, LZFT is desired to be lower; thus, because B < HO in the Simple scenario, the Baseline has better performance in this respect. Reviewing the data in Table 14, it can be seen that all but three comparisons (all for HV-D) were shown to be significantly different[12]. Within the remaining statistically significant results, the Baseline condition was shown to have superior performance for all measures except for certain catapult launch rates. The HO planner was shown to outperform the Baseline in measures of C2LR, C4LR, and TCLR, while the HA planner also outperformed the Baseline in C4LR (this seemingly counterintuitive result of superior launch rates *after* the occurrence of failures is discussed in the next section). Comparing the HO and HA plans, the HA planner was shown to be superior in terms of C4LR and UIT, with the HO planner maintaining superiority in all other metrics.

Table 14. Results of statistical testing for the Simple scenario (* signifies significance at α = 0.0015; NP = Non-Parametric Mann-Whitney U Test).

Metric	*Desired Magnitude*	*B vs. HO*			*B vs. HA*			*HO vs. HA*		
		Test	*p-value*	*Relation*	*Test*	*p-value*	*Relation*	*Test*	*Relation*	*p-value*
LZFT	LOW	NP	*	B<HO	NP	*	B<HA	NP	*	HO<HA
TATT	LOW	NP	*	B<HO	NP	*	B<HA	NP	*	HO<HA
TAAT	LOW	NP	*	B<HO	NP	*	B<HA	NP	*	HO<HA
TCAT	LOW	NP	*	B<HO	NP	*	B<HA	NP	*	HO<HA
MD	LOW	NP	*	B<HO	P	*	B<HA	NP	*	HO<HA
C2LR	HIGH	NP	*	B<HO	NP	*	B>HA	NP	*	HO>HA
C3LR	HIGH	-	-	-	-	-	-	-	-	-
C4LR	HIGH	NP	*	B<HO	NP	*	B<HA	NP	*	HO<HA
TCLR	HIGH	NP	*	B>HO	NP	*	B>HA	NP	*	HO>HA
HV	LOW	NP	*	B<HO	NP	*	B<HA	NP	*	HO<HA
HV-D	LOW	NP	P=0.005	B=HO	NP	p=0.906	B=HA	NP	p=0.535	HO=HA
UIT	LOW	NP	-	-	NP	-	-	NP	*	HO>HA

[12] Note that, due to the failure of Catapult 3, comparisons of C3LR were not performed.

5.2.1.2. Discussion

One interesting note from Table 14 is that the HO planner outperformed the Baseline in all launch rate values for Catapults 2 and 4 while the HA planner outperformed the Baseline only in C4LR, even with the HO and HA planners having an overall longer mission duration value. While this seems counterintuitive, recall that the Launch Rate (launches per minute) is determined by the following equation:

$$LaunchRate = \frac{n_{launches}}{MissionDuration} \tag{3}$$

where $n_{launches}$ is the number of launches assigned to that catapult and *MissionDuration* is the final calculated Mission Duration value. In this equation, increasing the number of launches at a single catapult over a given mission duration *X* will increase the Launch Rate of that catapult. An increase in Launch Rate will also occur for Mission Duration values slightly greater than *X,* but this does not continue unabated. If increases in mission duration continue unchecked, the value for the launch rate will begin to decrease. For instance, with an initial $n_{launches}$ of 5 and Mission Duration of 15 minutes, increasing the number of launches by three increases launch rate for all Mission Duration values less than 24 minutes. As a result of this, the reallocation of launch tasks by the HO and HA planners to Catapults 2 and 4 and to Catapult 4, respectively, provided sufficiently large increases in $n_{launches}$ to outweigh increases in Mission Duration (Figure 18 and Figure 19). However, because the number of launches is the system is fixed at twenty, these increases in Mission Duration cause detrimental affects in the Total Catapult Launch Rate (Figure 20).

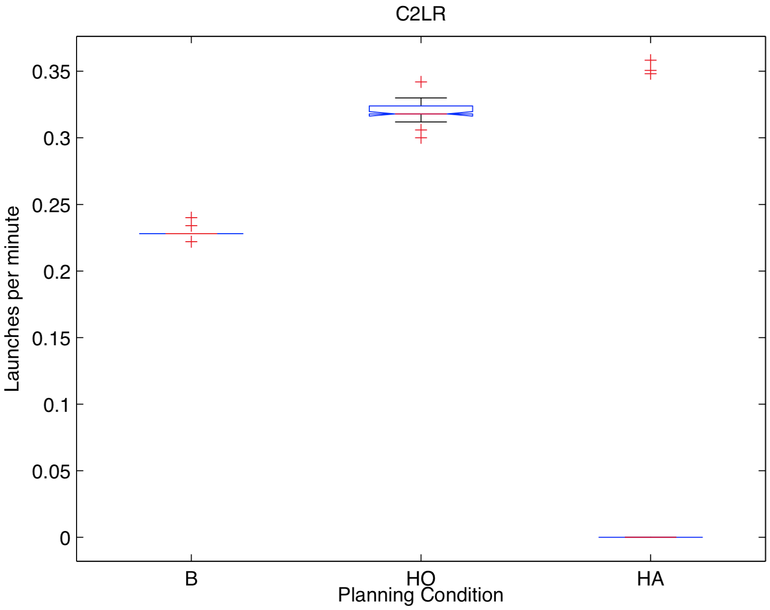


Figure 18. Simple scenario, Catapult 2 launch rate (launches per minute).

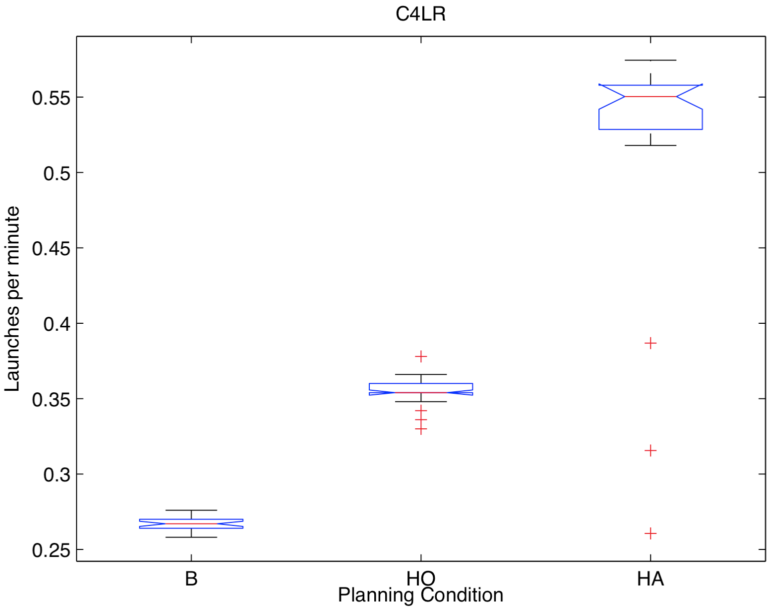


Figure 19. Simple scenario, Catapult 4 launch rate (launches per minute).

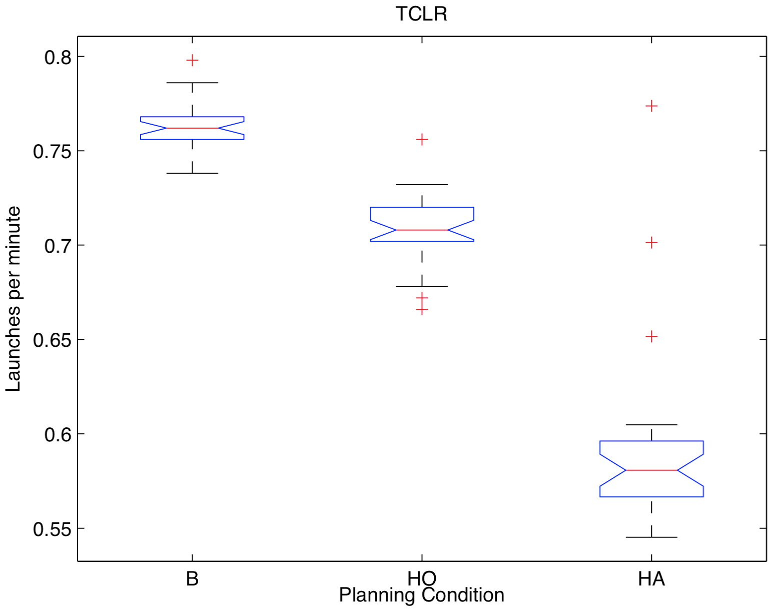


Figure 20. Simple scenario, Total Catapult Launch Rate (launches per minute).

In terms of a general performance comparison between the HO and HA planners, the most important point is that the HO planner performed better in Mission Duration. This measure directly addresses the total time required to launch all aircraft from the carrier deck and reach mission, a measure of primary importance in the wake of a launch catapult failure. In this case, catapult launch rates from this scenario also explain the changes in Mission Duration – the distribution of launch tasks between all remaining catapults by the HO planner resulted in a better total launch rate than the HA planner. In this case, a detailed discussion of the relative magnitudes of the individual catapult launch rates is not as of much concern as the fact that the HA planner did not assign aircraft to Catapult 2 in the majority of cases (Figure 18). The HO planner, based on SME heuristics, leveraged all available resources in order to achieve a faster processing time than the automated planner (signified by increases in TCLR in Figure 20).

While this review of quantitative data helps to identify the actions of the planner that created poor performance ratings, but they are still not sufficiently detailed to determine the specific failures in logic of the HA planner. However, this data supports the creation of two rival explanations as to why the HA planner made no assignments to Catapult 2. In the first explanation, the algorithm optimization may have shown that assigning all aircraft to Catapult 4 was the theoretical optimum for these cases. In the second, the algorithm state data could have returned a faulty value or made an incorrect assumption concerning the availability of Catapult 2 during the replanning stage, forcing the planner to make allocations to only a single catapult.

After investigation, the latter explanation was shown to be correct – the algorithm incorrectly considered a certain transient deck condition to be permanent, leading to the assumption that Catapult 2 was also unavailable. This then led to an inappropriate distribution of tasks on the deck, utilizing only a single resource. This unnecessary constraint on operations, leading to increased active time, taxi time, and mission duration values, is an example of the brittleness that often accompanies automated algorithms. Without prior coded knowledge concerning the transient nature of this condition, the algorithm was unable to properly compensate for its occurrence. Instead, the planner considered this event to be a permanent failure of the catapult, which is the only failure the algorithm could recognize. The planner then constructed a plan that was near optimal *given this faulty system information*. Correcting this error in state translation may allow the planning algorithm to generate plans as good or better than those developed by the HO planner in this round of testing.

This explanation would not have been reached without the inclusion of the additional launch rate metrics within the Mission Efficiency – Coverage subclass. Although these measures are not primary measures of mission performance, in this case, they were helpful in identifying certain inappropriate actions on behalf of the HA planner.

5.2.2. Moderate Scenario

The Moderate scenario (a recovery scenario) required the safe landing of twenty aircraft currently in flight. During landing procedures, two aircraft (SMAC #2 and FMAC #6) encountered failures (high priority fuel leak and low priority hydraulic leak, respectively). This required reassigning the landing order of aircraft to ensure that both of these aircraft landed before encountering a Fuel or Hydraulic Fluid Violation (FV or HFV), respectively. In examining the performance of the planners in this scenario, mixed results between the planning conditions were seen. The HA planner outperformed the HO planner in measures that addressed global performance (such as Mission Duration and Total Aircraft Active Time), while the HO planner maintained superior performance in measures addressing the high priority aircraft (measures for the failed SMAC). The following section contains the results of the statistical testing, which will be followed by a subsequent section discussing these results.

5.2.2.1. Results of Statistical Testing

Table 15 presents a compilation of the statistical testing data for the Moderate scenario. This table also lists the metric name, its desired magnitude (High or Low), and details the statistical testing between pairs of planning conditions. Significant values ($p < 0.0013$) imply that the null hypothesis (h_0: distributions of the two planning conditions are not different) was rejected. Testing revealed several instances of statistically equivalent performance, most notably for LZFT, HV, and HV-D, where no differences were seen between planning conditions. Additionally, the Baseline and HO planners were found to be equivalent for TAAT and TCAT, while the Baseline and HA planners were equivalent for TAAT and TCAT. The remaining measures all resulted in statistical differences between planning conditions, most importantly among Mission Duration and measures for SMAC #2 and FMAC #6.

Table 15. Results of statistical testing for the Moderate Scenario (* signifies significance at α = 0.0013; NP = Non-Parametric Mann-Whitney U Test; P = Parametric ANOVA).

Metric	*Desired Magnitude*	*B vs. HO*			*B vs. HA*			*HO vs. HA*		
		Test	*p-value*	*Relationship*	*Test*	*p-value*	*Relationship*	*Test*	*p-value*	*Relationship*
LZFT	LOW	P	p=0.733	B=HO	P	p=0.052	B=HA	P	p-=0.023	HO=HA
FMAC_6_HFR	HIGH	NP	*	B>HO	NP	*	B>HA	NP	*	HO<HA
FMAC_6_EART	LOW	NP	*	B<HO	NP	*	B<HA	NP	*	HO>HA
SMAC_2_AAT	LOW	P	*	B>HO	P	*	B>HA	P	*	HO<HA
SMAC_2_EFR	HIGH	P	*	B<HO	NP	*	B<HA	NP	*	HO>HA
SMAC_2_EART	HIGH	NP	*	B>HO	NP	*	B>HA	NP	*	HO<HA
TATT	LOW	NP	*	B>HO	P	p=0.032	B=HA	P	*	HO<HA
TAAT	LOW	P	p=0.132	B=HO	NP	*	B<HA	NP	*	HO>HA
TCAT	LOW	NP	p=0.018	B=HO	P	p=0.021	B=HA	P	*	HO<HA
WTQMS	LOW	NP	*	B<HO	NP	*	B<HA	NP	*	HO>HA
MD	LOW	NP	*	B<HO	NP	*	B<HA	NP	*	HO>HA
HV	LOW	NP	p=0.909	B=HO	NP	p=0.001	B>HA	NP	p=0.002	HO=HA
HV-D	LOW	P	p=0.005	B=HO	NP	p=0.046	B=HA	P	p=0.402	HO=HA
UIT	LOW	-	-	-	-	-	-	NP	*	HO<HA

Reviewing the relationships in this data, two major themes arise. First, the HO and HA planners developed schedules that addressed the two aircraft failures in similar, but different, manners. Both the HO and HA planners moved the SMAC aircraft (fuel leak) forward in the landing order, as demonstrated by lower values of SMAC_2_EART as compared to the Baseline[13]. Also, both the HO and HA planner moved the FMAC (hydraulic leak) backwards in the landing order, signified by larger values of FMAC_6_EART as compared to the Baseline. Comparing the HO and HA planners, it can be seen that the HO planner moved each aircraft to a greater degree than the HA planner (the HA moved the SMAC forward 1 slot in the landing order as opposed to 11 slots in the HO plan, while moving the FMAC backwards 6 slots as opposed to 7 in the HO plan). Secondly, the HA planner completed the mission in less time than the HO planner, as signified by superior performance in Mission Duration. The diagnostic measure WTQMS supports this view, showing that the HA planner required aircraft to be in the Marshal Stack holding pattern for less time overall.

5.2.2.2. Discussion

The results of testing for the Moderate scenario show mixed results for the performance of the HA planner as compared to the HO planner. Firstly, the HA planner differed in its approach to rescheduling the two failed aircraft. While the HA planner followed the instructions of the Expert User, moving the SMAC (fuel leak) forward in the landing order and the FMAC (hydraulic leak) backwards, this was done to a lesser extent than the HO planner. This resulted in better performance for the HA planner with respect to the FMAC but lower performance with respect to the SMAC. However, this launch order lead to a decrease in overall WTQMS and MD values and no increase in the number of Fuel or Hydraulic Fluid Violations (which did not occur for any case).

[13] Recall the EART is the duration of time from when the aircraft failure first occurs to the point that the aircraft lands. Thus, a lower EART for a given planner signifies that the aircraft landed earlier.

In this case, the determination as to which planning condition performed better in general depends on the perspective of the analyst. If viewed in terms of overall performance, the HA planner was better at optimizing the full system schedule, as revealed by its ability to decrease the overall Mission Duration and WTQMS. If viewed in terms of the number of FV and HFV, the planners performed equally well. However, if viewed in terms of adherence to operator heuristics, the HA planner may be viewed by operators as having inferior performance. Although the HA plan moved aircraft in the same manner as the HO plan, the magnitude of aircraft movements was of different magnitudes, especially in terms of the movement of the high priority SMAC aircraft. While the HA schedule was able to maximize its objectives of decreasing TAAT, WTQMS, and MD, the mismatch between the desired state of SMAC 2 (as judged by the HO planning action) and the actions of the planning algorithm may be undesirable.

Considering these three perspectives holistically provides additional insight with regards to the SME heuristics. The SME heuristic considering airborne aircraft failures maintains that severe failures (fuel leaks) be moved to the front of the landing order to minimize the chance the aircraft running out of fuel. Aircraft with less critical failures (hydraulic leaks) are moved to the end of the landing order to minimize the possibility of the aircraft crashing on landing, thus placing the other airborne aircraft in harm's way. In this testing scenario, the HA planner followed these suggestions, but not to the extent described in the heuristics. This less conservative planning strategy resulted in an overall decrease in total mission time while not incurring any more severe penalties in the form of FV or HFV. This suggests that the actions ordered by the heuristics may be overly conservative. This is also an instance of P/RA HSC systems working as desired – the human operator specified a set of high-level guidelines, within which the planning algorithm was able to develop a locally optimum plan.

These results could not have been seen if only a small set of metrics is reviewed in the course of examining performance. If the analysis had been limited only to the measures of primary concern for the stakeholders – FV, HFV, and measures for the SMAC and FMAC – the performance of the HA planner in optimizing the overall mission duration would have gone unnoticed. It would instead have been judged as moderately effective but still inferior to the HO planner. Only by the inclusion of additional metrics was the effectiveness of the HA planner in optimizing the *entire* system seen.

5.2.3. Complex Scenario

The Complex scenario (a mixed recovery and launch scenario) required the safe landing of twenty aircraft currently in flight while simultaneously launching two additional aircraft. During landing procedures, a single aircraft encountered a high priority emergency (SMAC #2 encountered a fuel leak). Solving this problem required balancing the need to land the aircraft immediately with the need to launch two others. The results of this scenario showed that the HO planner was able to address both the failures and the additional launches effectively, while the HA planner's solution further exacerbated problems in the system. This is primarily revealed through an analysis of the number of Fuel Violations (FV), total Mission Duration (MD), Total Aircraft Active Time (TAAT), the error measures for the emergency aircraft (SMAC #2), Landing Zone Foul Time (LZFT), and Wait Time in Queue in the Marshal Stack (WTQMS). Additional analysis of the launch rates for Catapults 2 through 4, as well as the Total Catapult Launch Rate

(TCLR), aided in understanding the planner actions that led to these differences in performance. The following two sections present the results of statistical testing, followed by a discussion of these results.

5.2.3.1. Results of Statistical Testing

Table 16 presents a compilation of the statistical testing data for the Complex scenario. This table also lists the metric name, its desired magnitude (High or Low), and details the statistical testing between pairs of planning conditions. Significant values ($p < 0.0010$) imply that the null hypothesis (h_0: distributions of the two planning conditions are identical) was rejected. In these results, it can be seen every statistical comparison involving the HA planning condition (compared to either the B or HO conditions) returned significance. The only cases where measures were seen to be statistically equivalent were found in the B vs. HO comparison (LZFT, TATT, MD, C2LR). Reviewing the relationships with statistically different data shows that there were differences in performance concerning the emergency aircraft SMAC #2 (which encountered a fuel leak). From the data it can be seen that the HO planner moved the SMAC forward (signified by a lower value of SMAC_2_EART as compared to the Baseline). The HA solution, however, either moved the SMAC *backwards* in the landing order, or its assignment of launch tasks delayed the landing of the aircraft. Additionally, the HA planner exhibited more Fuel Violations (FV) than either of the other conditions, with the HO planner demonstrating the lowest FV value overall. Also, the HA planner required more time to complete the mission than either of the other two cases (signified by poorer performance on MD). In handling the launches of the requested aircraft, differences in planning strategy also occurred, with the HO planner assigning aircraft only to Catapult 2 and the HA planner assigning aircraft to Catapults 3 and 4 (signified by higher launch rates for each case).

Table 16. Results of statistical testing for the Complex Scenario (* signifies significance at $\alpha = 0.001$; NP = Non-Parametric Mann-Whitney U Test; P = Parametric ANOVA).

Metric	*Desired Magnitude*	*B-HO*			*B-HA*			*HO-HA*		
		Test	*p-value*	*Relationship*	*Test*	*p-value*	*Relationship*	*Test*	*p-value*	*Relationship*
FV	LOW	P	n/a[14]	B>HO	NP	*	B<HA	NP	*	HO<HA
LZFT	LOW	P	p=0.478	B=HO	NP	*	B<HA	P	*	HO<HA
SMAC_2_AAT	LOW	NP	*	B>HO	NP	*	B<HA	NP	*	HO<HA
SMAC_2_EFR	HIGH	P	*	B<HO	P	*	B>HA	P	*	HO>HA
SMAC_2_EART	LOW	P	*	B>HO	P	*	B<HA	P	*	HO<HA
TATT	LOW	P	p=0.734	B=HO	P	*	B<HA	P	*	HO<HA
TAAT	LOW	P	*	B<HO	NP	*	B<HA	NP	*	HO<HA
TCAT	LOW	P	*	B>HO	P	*	B<HA	P	*	HO<HA
WTQMS	LOW	NP	*	B<HO	NP	*	B<HA	NP	*	HO<HA
MD	LOW	P	p=0.927	B=HO	NP	*	B<HA	NP	*	HO<HA
C2LR	HIGH	P	p=0.921	B=HO	P	*	B>HA	P	*	HO>HA
C3LR	HIGH	P	n/a[11]	B=HO=0	NP	*	B<HA	NP	*	HO<HA
C4LR	HIGH	P	n/a[11]	B=HO=0	NP	*	B<HA	NP	*	HO<HA
TCLR	HIGH	P	p=0.921	B=HO	NP	*	B>HA	NP	*	HO>HA
HV	LOW	NP	p=0.395	B=HO	NP	*	B>HA	NP	*	HO>HA
HV-D	LOW	NP	*	B>HO	NP	*	B>HA	NP	*	HO>HA
UIT	LOW	-	-	-	-	-	-	NP	*	HO<HA

5.2.3.2. Discussion

In the Complex scenario, the HA planner was outperformed by the HO planner in all of the Mission Efficiency metrics, most importantly in measures of Fuel Violations and Mission Duration. In fact, the only metrics in which the HA planner had better performance were launch rate values for Catapults 3 and 4, and this only signifies a difference in the assignment of launch tasks. These launch rate measures point to the root causes of poor HA planner performance.

[14] For these tests, values were either identical for all cases or zero for all cases, making statistical testing infeasible.

Reviewing the boxplots for the launch rates for Catapults 2-4 and the total launch rate (found in Figure K. 11 through Figure K. 14 in Appendix K), it can be seen that the HO planner made assignments only to Catapult 2, while the HA planner made assignments to only Catapults 3 and 4. For the majority of times, assigning aircraft to these aft catapults has no affect on airborne aircraft. However, for this scenario – where a set of airborne aircraft, low on fuel, will imminently land – this assignment can create significant repercussions. Recall that the aft catapults (Catapults 3 and 4) share deck space with the Landing Zone and that these resources cannot be operated simultaneously. Use of the aft catapults during landing operations may result in the waveoff of an aircraft on approach. This occurs if the landing strip is unavailable when an approaching aircraft reaches a certain threshold in its approach trajectory. If a waveoff is required, the incoming aircraft returns to the Marshal Stack before attempting a second landing[15]. This would then incur additional WTQMS, increase the total Mission Duration, and increase the likelihood of a fuel violation. In the case of the HA replanning, each of these conditions was seen, suggesting that the HA plan was incurring waveoffs due to its catapult assignments.

While several explanations can be developed concerning the reason for assigning launches to the aft catapults, the true cause is that the planning algorithm did not account for the interaction of the catapults and the landing strip and predicted that its assigned launch tasks were the fastest available option. In this case, however, the fastest launch configuration did not necessarily imply optimality. Because of the interactions of resources on the deck, the best option (as shown in the HO plan) would have been to vacate

[15] In reality, this would not occur in precisely this manner. Judgments on how to handle a waved off aircraft require knowledge of the fuel state of the aircraft and the availability of any airborne tankers for refueling. As the latter is not handled by the planning algorithm at this time, the method of handling waveoffs was altered for the time being.

the aft catapults to deconflict aircraft landings. The realization of this fault in the planner modeling could not have been reached if metrics had been limited only to Time-based (ME-T) measures or to metrics specifically related to SME heuristics (such as the handling of the SMAC). Limiting the analysis to only Coverage-based metrics would have revealed differences in planning strategy but would have failed to demonstrate how these differences affected the mission as a whole. The combination of the two led to a viable explanation for how the planner's actions were determined. In addition to comparing the performance of planners within an individual scenario, the performance of planners across scenarios can also be compared. This comparison addresses the performance of planners over the columns in Figure 17 and is discussed in the following section.

5.3. Performance Across Scenarios

The performance of systems across complexity levels allows analysts to detect potential issues with the brittleness of algorithms (their inability to account for possible inputs and environmental conditions), while also determining the limits of the human operator or algorithm in regards to increasing complexity levels. Typically, algorithms are expected to perform better in cases requiring rapid, complex mathematical optimizations [10], but brittleness may negate this likelihood. By testing across a variety of inputs and complexity levels, potential instances of brittleness – due to either improper environmental models or the inability to sense certain conditions – can be uncovered. For the case of the DCAP testing program, complexity was determined according to the number of required Subject Matter Expert heuristics required to replan, providing a consistency of complexity definition for three dissimilar test cases (one launch-only, one landing-only, one combined launch/landing).

With this definition of complexity in place, performance across scenario conditions can be judged in several manners. For the purposes of this discussion, performance is based on a scoring function relating how well two planners compared to each other in a given scenario (for instance, the HO versus the Baseline). This scoring function is given by the following equation:

$$PerformanceScore = \frac{N_{superior} - N_{inferior}}{N_{total}} \qquad (4)$$

where $N_{superior}$ is the number of metrics where the HO planner outperformed the Baseline, $N_{inferior}$ is the number of metrics where the HO planner underperformed with regards to the Baseline, and N_{total} is the total number of metrics used in statistical testing for that scenario (12, 14, and 17 metrics for the Simple, Moderate, and Complex scenarios, respectively). This returns a percentage score centered at 0, with +/- 100% denoting completely superior/inferior performance for a planner, respectively. This scoring metric was applied for all planner comparisons across all three scenarios, limited to the applicable metrics discussed in the previous sections. A visual representation of these resulting scores is presented in Figure 21, with the above equation lying on the vertical axis.

In this figure, the HO planner is seen to have superior performance with respect to the Baseline plan for the Complex scenario. Although this seems counterintuitive, it is due to the nature of the Baseline case and the actions of the HO planner. Recall that for the Baseline tests, no replanning occurred. For SMAC #2, which incurred a fuel leak, the HO planner will replan and move this aircraft forward in the landing order; by default, the HO plan will see superior performance for these metrics. In fact, these three metrics

(SMAC_2_ EART, SMAC_2_AAT, and SMAC_2_EFR) are specifically the cause of the net performance increase for the HO planner in the Complex scenario.

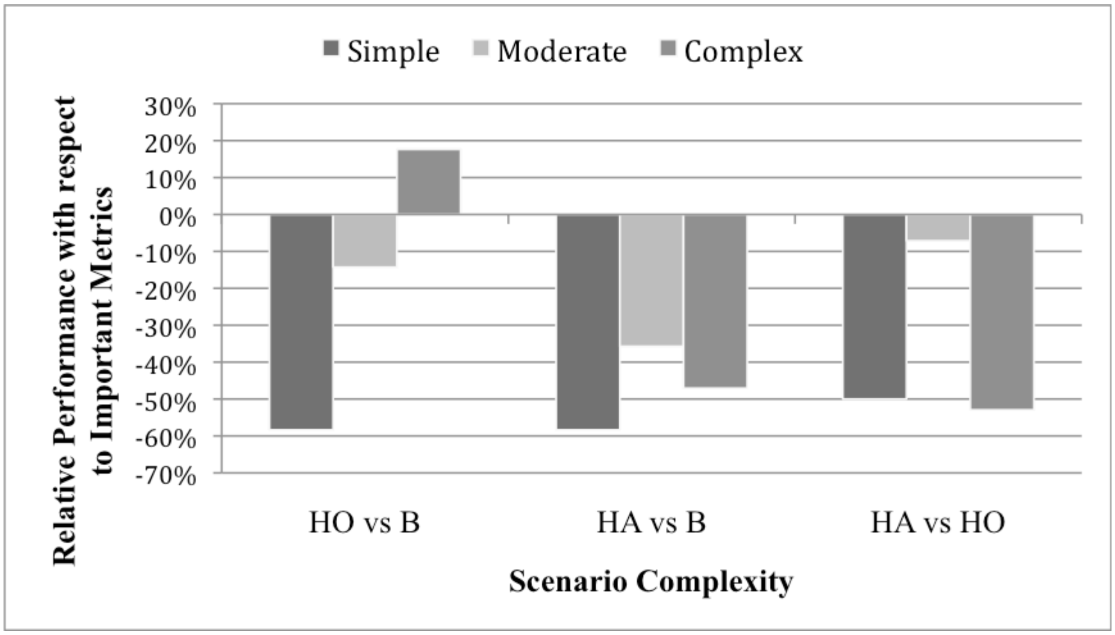


Figure 21. Relative performance comparison across scenarios.

Additionally, the contents of Figure 21 as a whole suggest that, for this P/RA system, the typical assumption of superior algorithm performance under increasing complexity is inappropriate. Observing the HO vs. B comparison, it can be seen that the performance of the HO planner improved as complexity increased. The HA planner formed an inverted “U” shape with respect to the Baseline, performing better in the Moderate scenario than in the Complex and Simple scenarios (the reasons for this will be discussed later in this section). This HA planner also exhibited similar performance with respect to the HO planner. Additionally, at no time did the performance of the HA planner become equal or superior to the Baseline or HO planning conditions. In this case, the HA planner was observed to perform poorly at both the higher and lower complexity levels. This data also suggests that the human heuristics, as applied in this context and in these scenarios, are adequate for addressing system complexity, as relative performance increased as complexity increased.

These results, however, are dependent on both the metrics used in the analysis and the definition of complexity, the latter of which will be discussed in a later paragraph. Concerning the metrics used in analysis, the previous paragraph describes performance in relation to *all* metrics used in testing (Figure 21). Figure 22 shows this graph again with only metrics *common* to all three scenarios. For instance, catapult launch rates – which do not exist for the Moderate scenario – have been removed, as have metrics for aircraft that experienced failures.

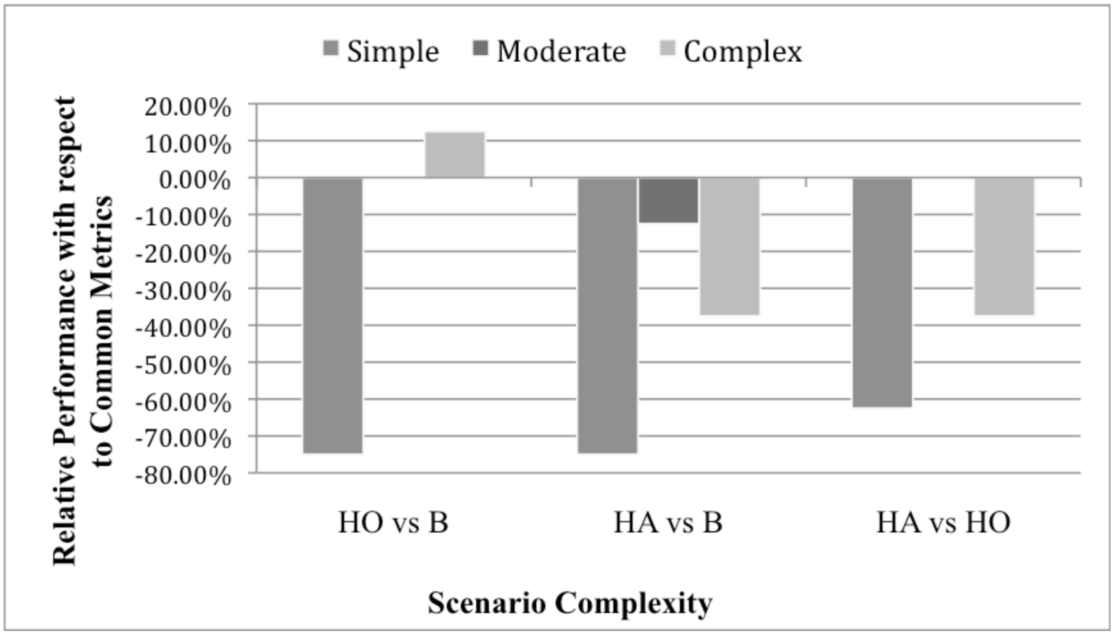


Figure 22. Relative performance comparison across Common metrics.

Between Figure 21 (all metrics) and Figure 22 (common metrics), the relative performance of each planner across scenarios is generally the same, with the HA planner still demonstrating the inverted "U" of performance. However, the removal of the error metrics for failed aircraft in the Moderate case have resulted in the HA planner having net even performance to the HO planner in this scenario; this was previously a case where the HA planner received a net negative rating on performance. The reasons for the inverted U of HA planner performance are the same as those discussed in the previous sections on individual scenarios and are recapitulated here – in the Simple scenario, improper state modeling led the HA planner to solve a more complex problem than actually existed, while in the Complex scenario, certain constraints modeling the complexity of interactions on deck were not included. However, for the Moderate case, the HA planner was able to achieve equivalent performance as the HO and Baseline planners, with the HA planner actually achieve superior performance in the important metrics of Mission Duration. In this case, with the system accurately modeling the complexity of the world, the planner's ability to forecast task durations and the interactions between vehicles allowed the planner to create a more efficient plan. These conclusions on complexity, however, are based on the definition and application of the SME heuristics; alternative definitions of complexity may not yield the same conclusions. Basing complexity on the maximum number of entities active in the system at any given time leads to a reordering of relative complexity (Table 17) with results of this shown in Figure 23.

Table 17. Alternative complexity definitions for the DCAP test scenarios.

Original Ordering	*Number of active entities*	*Order based on number of entities*
Simple	~100	Moderate
Moderate	20-36	Complex
Complex	22-46	Simple

As noted in Table 17, replanning for the Simple scenario involved altering the activity of a much larger number of personnel (crew, aircraft, or ground vehicles) than the other two scenarios. This is due to the fact that replanning the Simple scenario occurs when 18 aircraft are still on deck. Thus, with four crewmembers required per aircraft, five per catapult, and additional ground deck, approximately 100 personnel are affected by a single replan. Replanning for the Moderate scenarios involves no replanning for the crew or for UGVs, as all aircraft are still airborne. The Complex scenario differs from the Moderate only in that two aircraft will launch from the carrier deck, thus affecting only a small set of crew and support vehicles on deck. Using this perspective, the Simple scenario is actually the most complex scenario, while the Moderate scenario is the least complex. The relative performance ratings under this format (Figure 23) demonstrate that the HA planner performed best at the *minimal* complexity level (now the Moderate case) with performance decreasing as complexity increases, in direct opposition to the standard assumptions of increased performance.

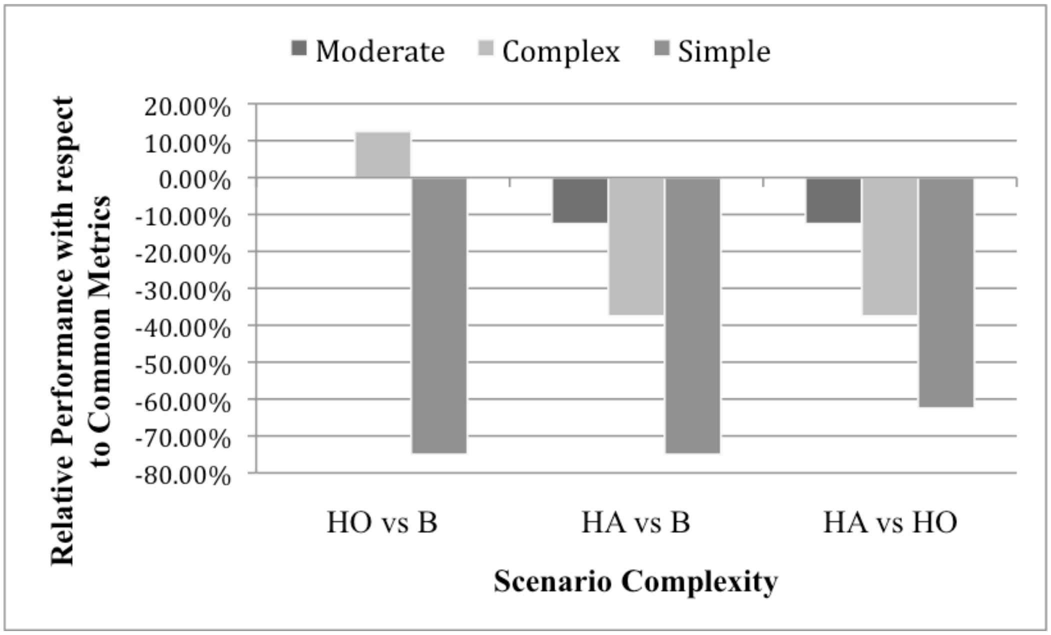


Figure 23. Relative performance comparison across Scenarios in the alternate complexity ordering (number of entities).

Additionally, the performance of the HA planner in adapting to complexity is also dependent on the specific deficiencies noted in the previous sections on individual test scenarios. For the case of the Simple scenario, the most likely cause of deficient performance is errors in data logging between the algorithm and the simulation environment for the acquisition of state data. This is not a direct error on the part of the human operator interacting with the planning algorithm, the latter of which calculated a theoretically near-optimal solution based on faulty state information and modeling. In this case, a transient delay condition was treated as a long-term failure, negating the assignment of tasks to a catapult. If this modeling assumption was corrected, or if the HO planner had treated this transient condition in the same manner, the comparative performance across complexity levels may have been much different. The same can be said for the Complex condition, where the system's modeling of the interaction between landings and takeoffs was shown to be deficient. For this scenario, the planner did not include a model for the interaction between the aft catapults and the landing strip, leading to an inappropriate assignment of resources that led to delays in landing airborne aircraft. The inclusion of a model for landing strip-catapult interaction should correct the failings of the planner in handling joint launch and recovery cases. However, simply instituting the design changes is insufficient; a second testing cycle should be utilized, as described by the Spiral Model in Figure 2. This round of testing should begin with the current scenarios in order to validate that the design changes properly address the conditions discussed in this chapter. Additionally, a set of new scenarios with new and different content should be created. This may allow for the identification of additional system constraints currently unknown to algorithm designers. Iteratively expanding the testing and introducing scenarios with new content will allow analysts and designers to continue to pinpoint possible refinements in the planning algorithm logic.

5.4. Chapter Summary

This chapter has reviewed the testing data from DCAP simulation testing, covering statistical analyses of the data and relative differences between planning conditions. The first three sections reviewed the results of planner performance within each scenario, with the fourth comparing the performance of the planners across complexity levels. Results from the scenarios originally designated as Simple and Complex demonstrated poor performance on the part of the HA system and provides system designers with points of investigation for further design changes. The Moderate scenario demonstrated that the HA planner has performance superior to the HO planner, taking similar actions in replanning but in a less conservative fashion. In this case, the actions of the HA planner, although they did not completely adhere to the SME heuristics, resulted in overall improvements in global measures of mission performance.

Examining performance across scenarios showed that the HO planner was capable of adapting to increased complexity within the scenarios, while the HA planner struggled with both the Simple and Complex scenarios. This runs contrary to Fitts' list [10] and the common belief that planning algorithms are better able to handle complex conditions. However, this may be due to the previously noted errors in the interfacing between the algorithm and the simulation. These errors resulted in the HO and HA planners viewing the state of the world in different manners. If the state information used by the automated planning algorithm had been more accurately modeled, these variations in performance may not have been seen. Furthermore, as shown in Chapter 5.3, these views on variation in performance are influenced by both the selection of metrics in analyzing performance and the definition of complexity under which this analysis occurs. By using only common metrics across all scenarios, the relative performance characteristics of the HO and HA planners changed slightly. These changes showed a mix of increases and decreases regarding the relative performance of planning conditions. Additionally, an alternate definition of complexity provided a very different view of system performance. The original definition (based on SME heuristics) showed the HA planner as having an inverted "U" of performance with respect to both the HO and B planning conditions. An alternate definition of complexity, based on the number of entities affected by replanning actions, showed that the HA planner had linearly decreasing performance as the level of complexity increased.

6. Conclusions and Future Work

The goal of this thesis was to establish what metrics and methods were required for the validation and verification of a P/RA HSC system and how these metrics can be used in the iterative systems engineering design process. This requires not only validating the performance of the individual subcomponents of the system (the human operator and the automation), but also validating the quality of interaction between these subcomponents and the effectiveness of their collaboration in forming solutions. These measurements concern two steps in the systems engineering spiral model depicted in Figure 2 – the *Integration and Test* and *Acceptance Test* stages. In addressing the *Integration and Test* step, the metrics and methodology seek to measure the quality of integration of the subcomponents (in this case, the human, the automated algorithm, and the display/control interface) and their effects on one another. For the *Acceptance Test* step, the metrics and testing protocol aid in the characterization of overall system performance, providing comparison points for different system design alternatives or in regards to a current system.

These objectives formed the basis of three specific research objectives for this thesis, which are reviewed and answered in the next subsection. The second section within this chapter highlights the limitations and future work related to this thesis.

6.1. Research Objectives and Findings

This thesis sought to address three specific research questions concerning the design and implementation of metrics and a testing protocol for the validation of P/RA HSC systems. This section reviews the three research questions posed in Chapter 2 and details the answers formulated in the course of this research.

What metrics are required for evaluating the performance of a Planning and Resource Allocation Human Supervisory Control system as compared to manual planning?

The analysis of the three DCAP scenarios revealed a necessity for providing both descriptive and diagnostic metrics in evaluating the performance of the HA planner as compared to the HO, manual plan. Descriptive metrics provide comparison points between planning conditions, allowing analysts to define explicit differences in performance between the systems (in this case, different planning conditions). In the case of DCAP, these measures were defined based on the context of operations within the system. For supervisors on the aircraft carrier deck, safety is of paramount importance and included measures such as the number of Fuel Violations and the ability to recover emergency aircraft quickly. The second primary measure is the speed of operations, specifically total Mission Duration, but additional measures addressing the time spent executing certain subtasks (e.g. taxiing on deck or waiting in the Marshal Stack) were also included. The emphasis on these two classes of measure may not be appropriate for other P/RA systems (e.g. airport traffic routing, hospital operating room allocation, etc.) and may result not only in entirely different sets of metrics, but a redefinition of priorities of individual metrics and metric classes.

Regardless of the specific nature of the system, the inclusion of primary metrics as comparison points allows analysts to quantify the variations in performance between planning conditions. However, these metrics may not allow a qualitative understanding of *why* these variations occurred. The inclusion of diagnostic measures, such as subcomponent task times (WTQMS, TATT, TAAT) and resource allocation metrics (catapult launch rates) allows analysts to understand how the resulting actions of the planner led to undesirable performance. In the DCAP analysis, the launch rate metrics for the Simple scenario were diagnostic in that they enabled analysts to determine that an unbalanced distribution of launch assignments led to a decrease in overall launch rate and an increase in final Mission Duration. A further review of the launch assignments then revealed an error on behalf of the planning algorithm in modeling a transient failure condition on the deck. In the case of the Complex scenario, catapult launch rates were again used as diagnostics in determining the root causes of poor planner performance. In this case, however, the launch rates revealed a significant deficiency in the algorithm's lack of modeling of interactions between the aft catapults and the landing strip. Although the algorithm took actions that were predicted to take the minimum time to launch, launches did not occur quickly enough. This resulted in conflicts in the landing strip and led to a series of aircraft waveoffs on landing.

How can these metrics assess the variations in performance of human and combined human-algorithm planning agents?

For the DCAP system, an analysis of variations in performance took two forms. The first addressed variations in performance for all planners within a specific scenario. These comparisons were primarily facilitated by the usage of descriptive measures of performance, applied consistently to each planner within each trial. Both qualitative and statistical analyses were utilized, each contributing to the determination of performance. Statistical analyses aid in identifying differences within data, but do not provide guidance in determining the form of the difference. In this regard, the inclusion of qualitative analyses to classify the type of difference significantly aided in distinguishing cases of superior and inferior performance. Only by including the additional qualitative review of differences in measures could variations in performance truly be depicted.

However, this testing protocol was purposefully limited in its ability to test variations in performance with respect to human input through the usage of a single Expert User. The scripting of this user's actions allowed the testing program to highlight specific deficiencies on the part of the planning algorithm and its responses to varying complexity levels at the cost of investigating system performance with respect to variations in human input. This limits the generalizability of the test results to the larger class of potential users. However, if this testing protocol were repeated with a single scenario (fixing the test protocol on this axis) and using a larger pool of human test users, a similar examination of performance could be performed. In this manner, the testing protocol can be viewed as existing in three axes – operator, algorithm, and scenario. Comparing the performance of any condition requires a fixing of at least two of these axes. In the current testing, the human axis was fixed, and comparisons occurred by either fixing on the scenario axis (comparing performance between planners) or the planner axis (comparing planner performance across scenarios).

The utilization of realistic scenarios that addressed not only differing complexity levels, but also different planning environments (launch-only, recovery-only, and mixed), allowed the identification of several variations in planner performance. For the Simple scenario, a deficiency in planning operations was seen; the planning algorithm did not properly model a transient failure on deck. This resulted in an underutilization of resources on deck, lowering the launch rate of the system and leading to an increase in overall mission duration. In the Moderate scenario, similar, yet different, performance was seen between the HO and HA planners. The HA planner made similar adjustments to the two emergency aircraft in the scenario, but did not move aircraft at the same magnitude as the HO planner. However, the HA planner's actions resulted in lower overall Mission Duration while not incurring any more Fuel or Hydraulic Fluid Violations. Lastly, in the Complex scenario, a deficiency different from that of the Simple case was seen. Here, the planning algorithm did not model the interactions between resources on the deck, instead selecting to launch aircraft from the aft catapult and creating conflicts with incoming aircraft attempting to land. While these actions were the result of attempting to launch aircraft in the minimum amount of time, which may be optimal for many cases, this choice of action was suboptimal in this scenario. Without the inclusion of all three cases, some significant information regarding the performance of the planners (both good and bad) would not have been discovered.

This testing program also revealed variations in the performance of the deck environment itself, despite the standardization of user replanning actions and initial scenario states. For instance, HV-D showed a large distribution for the Moderate, Baseline case while having a relatively small distribution for the Simple, HO case. This occurred even though the scenario utilized the same initial conditions and the user took the same replanning actions in every trial. This variation in metric values was also seen for the Baseline case, where the user took no replanning actions whatsoever. The dynamics of the carrier deck environment introduce further stochasticity into system performance, and the testing protocol must be such that the number of trials performed effectively captures this variability. Failing to do so may lead the data to exist at the tails of the distribution, leading to under- or over-predictions of performance. This may then lead to errors in the statistical comparisons and incorrect conclusions on planner performance.

How can these metrics predict system feasibility and highlight design flaws?

The prediction of system feasibility and the determination of design flaws is a qualitative determination on the part of the analyst, requiring knowledge of both the dynamics of the environment and the capabilities of the planning systems, supplemented by data generated in the testing phase. System feasibility is best served by descriptive measures of performance, while design flaws are best pinpointed by diagnostic measures.

Concerning system feasibility, the results of the descriptive performance measures, tempered by the analyst's understanding of system dynamics, provide evidence for or against system acceptance. The inclusion of multiple comparison points (in this case, the HO and B planning conditions) effectively grounds the analyst with regards to expected system performance. The results from the Moderate scenario suggested that the HA planner is indeed capable of performing as well as the HO planner in addressing failures, albeit through a different planning strategy. In this case, the HA planner took actions similar to the HO planner (making a similar reorganization of the landing order) but ordered less overall movement of the two emergency aircraft. This difference in strategy was revealed through the use a variety of metrics and by comparing the performance of the system to the HO planner. The results from the Simple scenario, taken at face value, suggest that the HA planner was completely infeasible for use in this manner. However, the definition of the Simple scenario and its method of execution implied that no planner would be able to achieve performance equal to or better than the Baseline. Without this understanding (either foreknowledge on the part of the designer, or through observing the performance of the state-of-the-art HO planner), this scenario would have demonstrated that neither the HA nor the HO planner is adequate for use in the real world. In this case, the inclusion of the HO planner as a comparison point grounded expectations for current system performance.

Design flaws are highlighted by a combination of the analyst's understanding of the system dynamics and the application of diagnostic measures seen in the system. In utilizing measures as diagnostics of performance, explanations for the variation in performance can be constructed and can guide analysts through the next design iteration. Again reviewing the results from the Simple scenario, the diagnostic measures of catapult launch rate revealed a possible error in how the planner viewed the availability of the catapults (the improper modeling of a transient failure), providing a specific point of investigation for the designers. This inadequacy is likely due to a failure in properly addressing the state information being acquired from the simulation data and is likely not due to actions of the human or algorithm specifically.

However, in the Complex scenario, the use of diagnostic measures for catapult launch rates revealed a possible logical flaw in the algorithm's model of system dynamics. The model used by the algorithm suggested that the actions in the proposed schedule would be optimal, providing maximum launch rate and minimum operational time. However, the execution of the schedule demonstrated that this was not the case, providing a point of investigation for the design team. In both of these cases, the results of the metrics applied to the system and the subsequent statistical and qualitative analyses provided evidence to support a hypothesis. The analysts' understanding of the system dynamics and the functions of the planning algorithm and the human operator then frame these explanations.

The poor performance of the HA planner in both the Simple and Complex scenarios highlights instances of brittleness within the algorithm and its data collection modules. Each of these discoveries was highly specific to the case presented – the errors in state information in the Simple scenario would not have been seen if crew had not been moving through Catapult 2's area one minute earlier or later. Similarly, the Complex scenario tested a unique boundary case that, while a realistic and important test case, may not be typical. The majority of operations are similar to the Simple and Moderate cases, neither of which would have revealed the failure of the algorithm to compensate for conflicts between the aft catapults and landing strip. For the DCAP system, the utilization of a variety of testing scenarios incorporating a broad range of possible circumstances aided in revealing specific brittle features of the algorithm.

6.2. LIMITATIONS AND FUTURE WORK

Although the presented methodology was successful in defining a set of metrics that allowed analysts to define the performance of the planners and to detect a series of design changes for the HA planning algorithm, it is not without limitations. The definition and use of large numbers of metrics (as was done here) can be time-consuming and expensive. The utilization of all metrics in a statistical analysis would contribute to a rapid degradation of family-wise significance, which can be ameliorated through the use of statistical reduction techniques such as Principal Components Analysis. However, performing a PCA for a large number of tests, scenarios, and planning conditions requires even further time and resources. Additionally, removing metrics based on strict adherence to the PCA may lead to heterogeneous data sets; that is, metrics that show high correlation in one test scenario may not show correlated in another. A possible extension of this work is improving the ability to identify key metrics early on in the process, preventing excess data collection and reducing the number of man-hours spent in the analytical process.

Additionally, this methodology has only been applied to a single P/RA system example, using a single deterministic resource allocation algorithm (an Integer Linear Program), operating in a unique environment and using a single Expert User in its testing program. These are all very specific conditions that occurred in the testing of this system, and the generalizability of the methodology may have suffered because of this.

The inclusion of only one single, deterministic algorithm for resource allocation limits its generalizability to pure path planning systems or to systems utilizing Heuristic or Probabilistic algorithms. Currently, ongoing research involves the creation of two additional planning algorithms within these categories. These algorithms are of different formats than the ILP used here (both are non-deterministic) and have different strengths and weaknesses concerning failure handling and guarantees of optimality. Future research should apply this methodology to this pair of algorithms to determine if any changes in approach are needed for these non-deterministic algorithms.

Additionally, this testing program used a single Expert User, a member of the design team intimately familiar with the interface. For the purposes of algorithm validation, this is an acceptable practice, as the inclusion of this user minimizes user error and standardizes the input to the system. The lack of relative "noise" in the input of the human operator allows for a direct assessment of the performance of the algorithm. While this limits the confounds that would have arisen from utilizing multiple users, much future work remains to be done to address the use of the DCAP system by the user population at large and in realistic circumstances.

Despite the fidelity of the simulation environment, it is difficult to precisely recreate the exact conditions under which these decisions are made. The physical environment, time pressure, and stress associated with operators performing these actions were not addressed in this testing program. Additionally, the final implementation of this P/RA HSC system will most likely include several different operators with various skill levels, experiences, and different planning heuristics. Before accepting the system for final implementation, a larger testing program including wide spectrum of potential users should be performed and should utilize many of the measures negated in Chapter 4.2 by the inclusion of the Expert User. This testing would specifically address measures of Collaboration, Human Behavior Precursors, and Human Behavior Efficiency. This testing would also address the performance of the planner and the system as a while across multiple users.

This testing protocol cannot truly be finalized until its application has been tested across a broad range and depth of applicable P/RA HSC systems. While this testing protocol has proved effective in allowing analysts to determine the differences between manual (HO) and combined human-automation (HA) planning for the DCAP system, the DCAP system was a unique and highly specific test case. A great deal of further work remains in validating and verifying this methodology across a larger sample size of P/RA HSC systems, algorithm formats, and user pools.

6.3. THESIS SUMMARY

This thesis has addressed the definition of metrics and a testing protocol for P/RA HSC systems, with a focus on validating the performance of the combined human-algorithm planning system. This thesis began with a review of an example HSC metric framework and three main classes of automated algorithms used in P/RA HSC systems. This thesis then reviewed various measures associated with the classes defined in the metric framework, providing examples of their application in various HSC systems or in algorithm validation and verification. After providing background on an example P/RA HSC system, the definition of specific metrics and the protocol for testing the DCAP system were provided. This included descriptions of three testing scenarios across varying levels of complexity and explanations of how an Expert User interacted with the system during the testing scenario. The results of system testing were presented and discussions of the performance of the various planning conditions were provided. Discussions of planner performance across levels of complexity were also offered. Lastly, the ability of this work to address three main research objectives was discussed, as were the limitations of and future work related to this research. While the metrics and protocol has proved successful in this application to this specific system, there is no guarantee that these metrics and the testing protocol are optimal for other systems. Repeated application of this methodology to a variety of other P/RA HSC system formats and algorithm forms will provide additional insight into the successes and limitations of the metrics and protocol used here.

APPENDIX A – GAUSSIAN PROCESSING TIMES FOR DCAP SIMULATION

Table A.1. List of Gaussian processing times for the DCAP simulation.

Task	*Description*	*Mean*	*Standard Deviation*
Fueling	Fuel flow rate	600 lb/min	136 lb/min
Takeoff	Attach Aircraft to Catapult	1 minute	15 second
	Accelerate to speed	3.5 seconds	0.5 seconds
Landing	Time to hit wire and decelerate	4 seconds	1 second

APPENDIX B – HYBRID COGNITIVE TASK ANALYSIS FOR DCAP

The goal of a Hybrid Cognitive Task Analysis (hCTA) is the creation of a set of information, display, and functional requirements for the interface of a complex system, beginning with a high-level scenario task description [113]. The process consists of five steps: 1) a Scenario Task Overview (STO), 2) a set of Event Flow Diagrams (EFDs), 3) generation of Situational Awareness Requirements (SARs), 4) generation of Decision Ladders (DLs) with corresponding display requirements, and lastly 5) generation of Information and Functional Requirements (IRs and FRs). The hCTA process is often used in cases where the system being designed is revolutionary (in that it has no prior predecessors) and has previously been used in the design of control interfaces for unmanned underwater vehicles (UUVs) [132], submarine surface collision avoidance [133], and interactive scheduling for commuter trains [134]. The following sections will describe the steps used in creating the DCAP hCTA, beginning with the Scenario Task Overview.

B.1 PRELIMINARY ANALYSIS

B.1.1 Scenario Task Overview (STO)

The Scenario Task Overview decomposes the full mission definition into a series of phases to be completed by the human operator. In most cases, this results in a relatively linear flow of phases from one task to the next. For instance, in the case the rail scheduling system in [134], which dealt with the management of the schedule of a single train traveling between two points, the phases were broken into *Pre-departure*, *Departure*, *En*

Route, and *Arrival* (termination) phases. For the aircraft carrier operating environment, the operator primarily engages in two roles – *Monitoring* the environment and *Replanning* when required. In this case, replanning consists of reassigning aircraft to either different resources or altering the order of assignment to a single resource (or a combination of both). At this time, en route mission and strike planning are not included, but may be included in future work. The system instead focuses on the replanning of tasks only for aircraft in the immediate vicinity of the aircraft carrier, focusing on tasks that require usage of resources on deck. Within these two phases of operation, a total of 19 subtasks were defined for management of aircraft carrier deck operations. These are presented in Table B. 1 (*Monitoring* phase) and Table B. 2 (*Replanning* phase).

Table B. 1. Scenario Task Overview - *Mission* phase and subtasks

Phase	***Task Number***	***Task***	***Related EFD Symbol***
Monitor	1	Observe crew motion on deck	L1
	2	Issue halt to operations if unsafe	P1
	3	Observe operations of airborne aircraft	L2
	4	Issue alert if failure occurs	P2
	5	Observe state of deck resources	L3
	6	Issue alert if failure occurs	P2
	7	Monitor total shift time of the crew	L3
	8	If over working time, document and account for in future personnel scheduling	P3
	9	Monitor total shift time of all pilots	L5
	10	If over working time, document and account in future mission scheduling	P4
	11	Monitor status of the schedule	L6
	12	If schedule degrades due to delay accumulation, judge need for replan.	P5
	13	If replan need arises, initiate replan	D1

Table B. 2. Scenario Task Overview - *Replanning* phase and subtasks.

Phase	***Task Number***	***Task***	***Related EFD Symbol***
Replan	1	Determine what item has failed	P6
	2	Determine the type/extent of failure	P7
	3	Determine affected aircraft	D2
	4	Reassign tasks for affected deck aircraft	D3
	5	Reassign landing positions for affected airborne aircraft	D4
	6	Communicate schedule changes to all personnel	P9

B.1.2 Event Flow Diagrams (EFDs)

An Event Flow Diagram places the subtasks that comprise the STO phases into a process flow diagram, highlighting the temporal constraints and relationships between subtask elements. The subtasks are broken into Processes, Decisions, Loops, Phases, and Assumptions with arrows denoting the transition between elements (Figure B. 1). These elements are then assigned alphanumeric labels for traceability. Thus, a future Information Requirement (IR1) can be linked to a specific Loop (L3), Decision (D5), or Process (P12). Three different EFDs were created, totaling six Loops, thirteen Processes, and seven Decision elements.

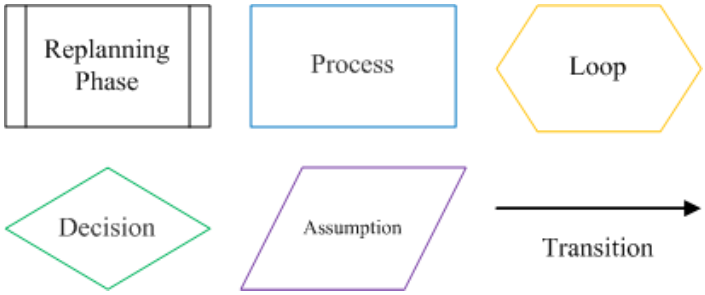


Figure B. 1. Elements used in Event Flow Diagrams.

The first EFD (Figure B. 2) describes the operator in the *Monitoring* phase, consisting of a set of concurrent monitoring Loops (L1-L6) and a single Decision element regarding the need to replan (D1). If this decision returns a "yes" answer, the operator moves to the replanning phase. The second EFD (Figure B. 3) describes the basic replanning process of operators in the current operating paradigm. Although there are few items within this diagram, the important note is the existence of three decision-making loops (Decision blocks contained within the loop symbol). First, for all *i* aircraft in the system, the operator must decide the extent to which a failure conditions affects each aircraft. This provides the operator with a list of potential aircraft to reschedule. The next two decision loops separate this list into groups of *j* deck and *k* airborne aircraft, which experience different failures, have different concerns, and different methods of redress given the failure in the system. For each aircraft, the operator must determine if a new plan must be generated for the aircraft. Once all plans are generated, these are transmitted to the personnel on deck for implementation (P9).

For each of the two EFDs presented here, Decision Ladders (DLs) were created in order to determine specific informational and functional requirements for the decision-making process. These DLs also guided the inclusion of the automated planning system within DCAP, which required the creation of an additional STO Phase, a third EFD, and further DLs, all of which are detailed in the following sections.

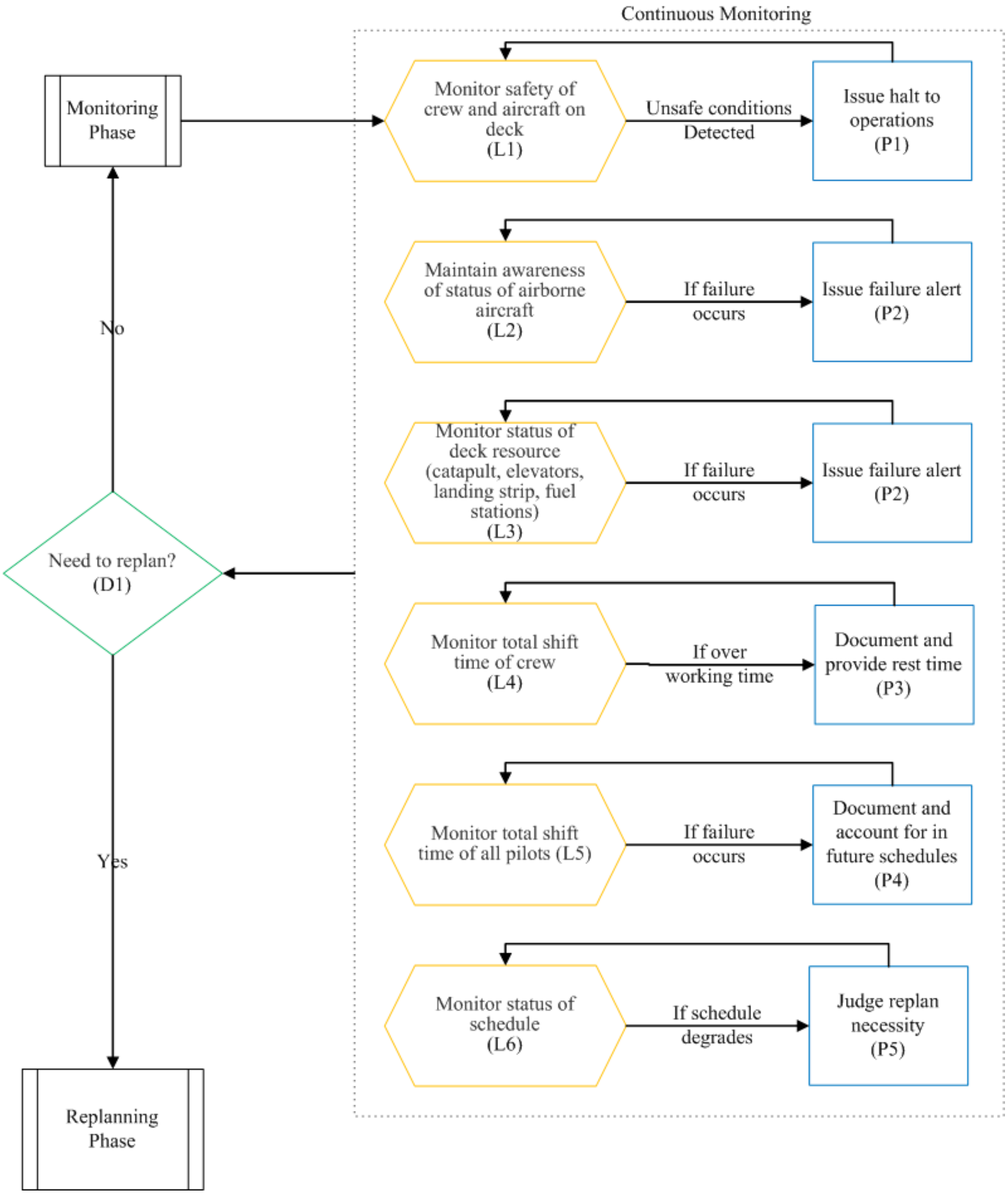

Continuous Monitoring
Monitoring Phase
Monitor safety of crew and aircraft on deck (L1)
Unsafe conditions Detected
Issue halt to operations (P1)
Maintain awareness of status of airborne aircraft (L2)
If failure occurs
Issue failure alert (P2)
Monitor status of deck resource (catapult, elevators, landing strip, fuel stations) (L3)
If failure occurs
Issue failure alert (P2)
No
Need to replan? (D1)
Monitor total shift time of crew (L4)
If over working time
Document and provide rest time (P3)
Yes
Monitor total shift time of all pilots (L5)
If failure occurs
Document and account for in future schedules (P4)
Monitor status of schedule (L6)
If schedule degrades
Judge replan necessity (P5)
Replanning Phase

Figure B. 2. Event Flow Diagram 1 - *Monitoring*.

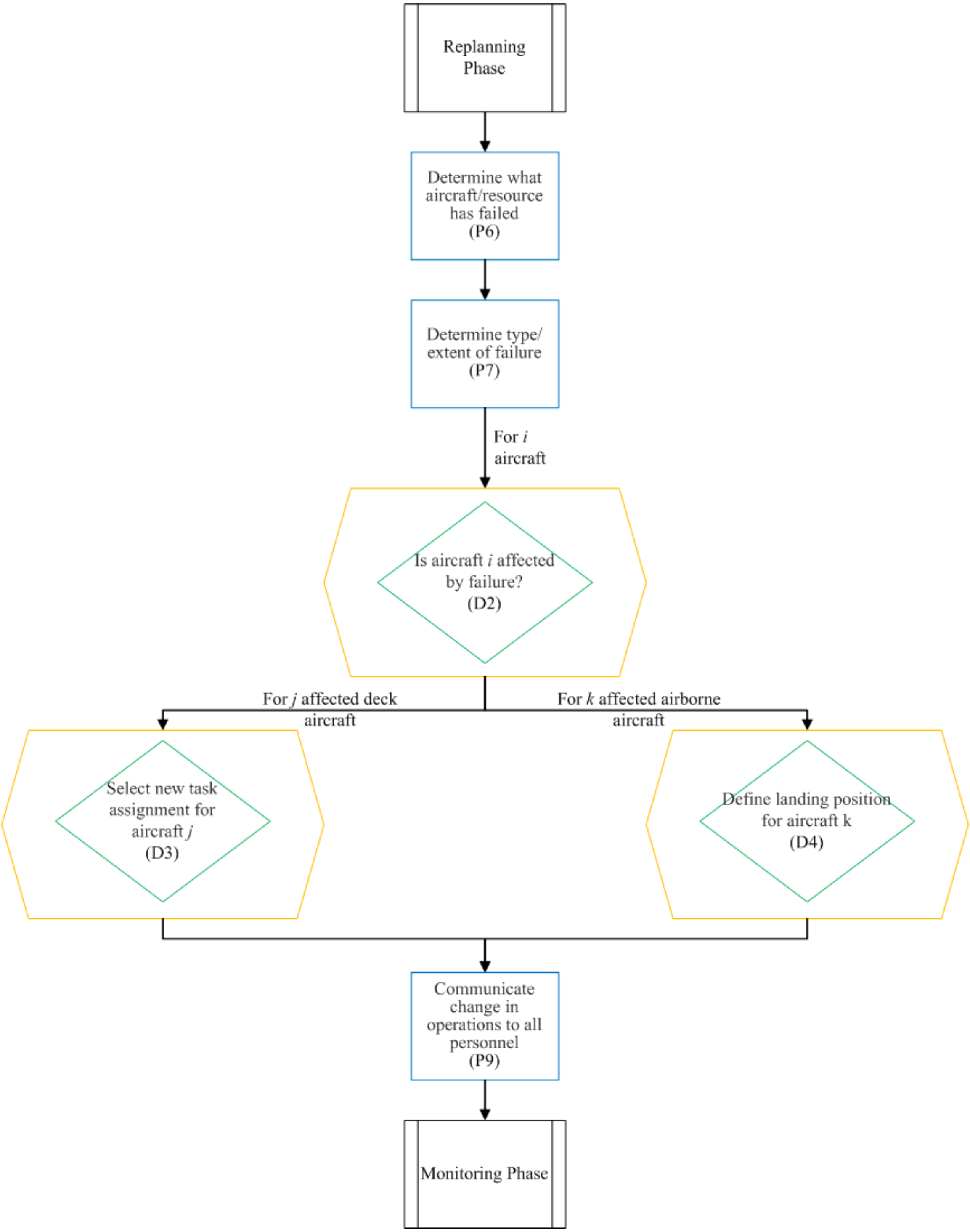

Replanning Phase
Determine what aircraft/resource has failed (P6)
Determine type/ extent of failure (P7)
For *i* aircraft
Is aircraft *i* affected by failure? (D2)
For *j* affected deck aircraft
For *k* affected airborne aircraft
Select new task assignment for aircraft *j* (D3)
Define landing position for aircraft k (D4)
Communicate change in operations to all personnel (P9)
Monitoring Phase

Figure B. 3. Event Flow Diagram 2 - *Replanning*.

B.1.3 Decision Ladders

For each Decision element that appears in a given EFD, a corresponding Decision Ladder (DL) is used to further detail the knowledge and information processing tasks required for a human user to reach the decision [135]. Decision Ladders begin with some form of alert to the operator (be it endogenous or exogenous) and delineate the specific tasks and information requirements required to work through the decision-making process. In doing so, DLs move from skill-based to knowledge-based behavior [136]. By tracking the steps in the decision-making process, a set of informational requirements can be generated. Information requirements describe individual bits of information that must be presented within the system interface. For instance, the operator may need to replan a schedule due to the emergence of a failure. The operator then must know the type of failure and the afflicted aircraft or resource. These two items – failure type and affected resource – are two separate Information Requirements for the display. Later in the process, the operator may need to submit a new schedule to the system. A Functional Requirement would then be a control item that allows this interaction. A total of four DLs were created for the EFDs in Figure B. 2 and Figure B. 3. These DLs appear in Figure B. 4 through Figure B. 7.

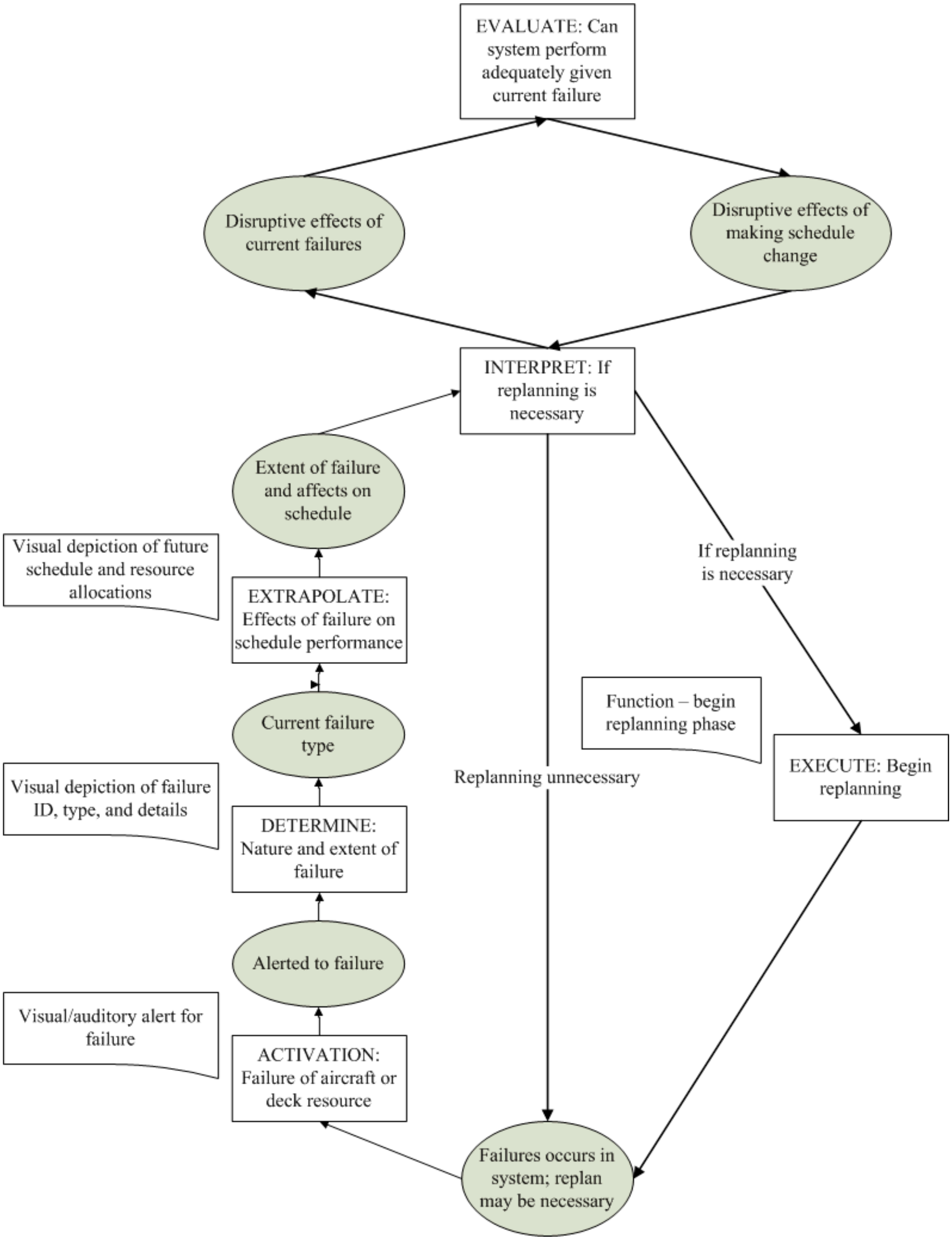

EVALUATE: Can system perform adequately given current failure
Disruptive effects of current failures
Disruptive effects of making schedule change
INTERPRET: If replanning is necessary
Extent of failure and affects on schedule
If replanning is necessary
Visual depiction of future schedule and resource allocations
EXTRAPOLATE: Effects of failure on schedule performance
Function – begin replanning phase
Current failure type
EXECUTE: Begin replanning
Replanning unnecessary
Visual depiction of failure ID, type, and details
DETERMINE: Nature and extent of failure
Alerted to failure
Visual/auditory alert for failure
ACTIVATION: Failure of aircraft or deck resource
Failures occurs in system; replan may be necessary

Figure B. 4. Decision Ladder 1 - Necessity of Replanning (from EFD 1).

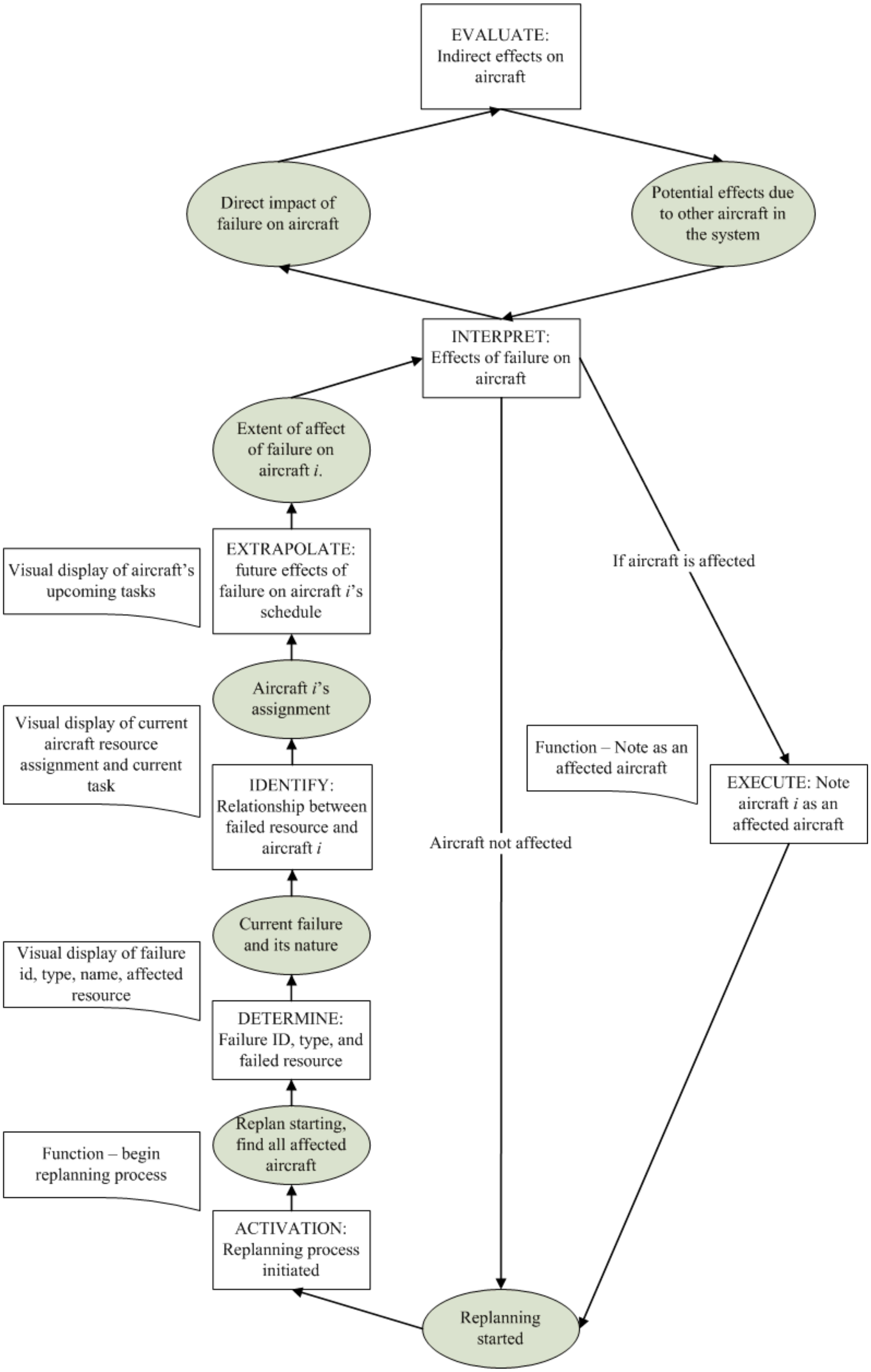

EVALUATE: Indirect effects on aircraft
Direct impact of failure on aircraft
Potential effects due to other aircraft in the system
INTERPRET: Effects of failure on aircraft
Extent of affect of failure on aircraft *i*.
EXTRAPOLATE: future effects of failure on aircraft *i*'s schedule
Visual display of aircraft's upcoming tasks
If aircraft is affected
Aircraft *i*'s assignment
Visual display of current aircraft resource assignment and current task
Function – Note as an affected aircraft
EXECUTE: Note aircraft *i* as an affected aircraft
IDENTIFY: Relationship between failed resource and aircraft *i*
Aircraft not affected
Current failure and its nature
Visual display of failure id, type, name, affected resource
DETERMINE: Failure ID, type, and failed resource
Replan starting, find all affected aircraft
Function – begin replanning process
ACTIVATION: Replanning process initiated
Replanning started

Figure B. 5. Decision Ladder 2 - Is aircraft affected by failure (from EFD 2).

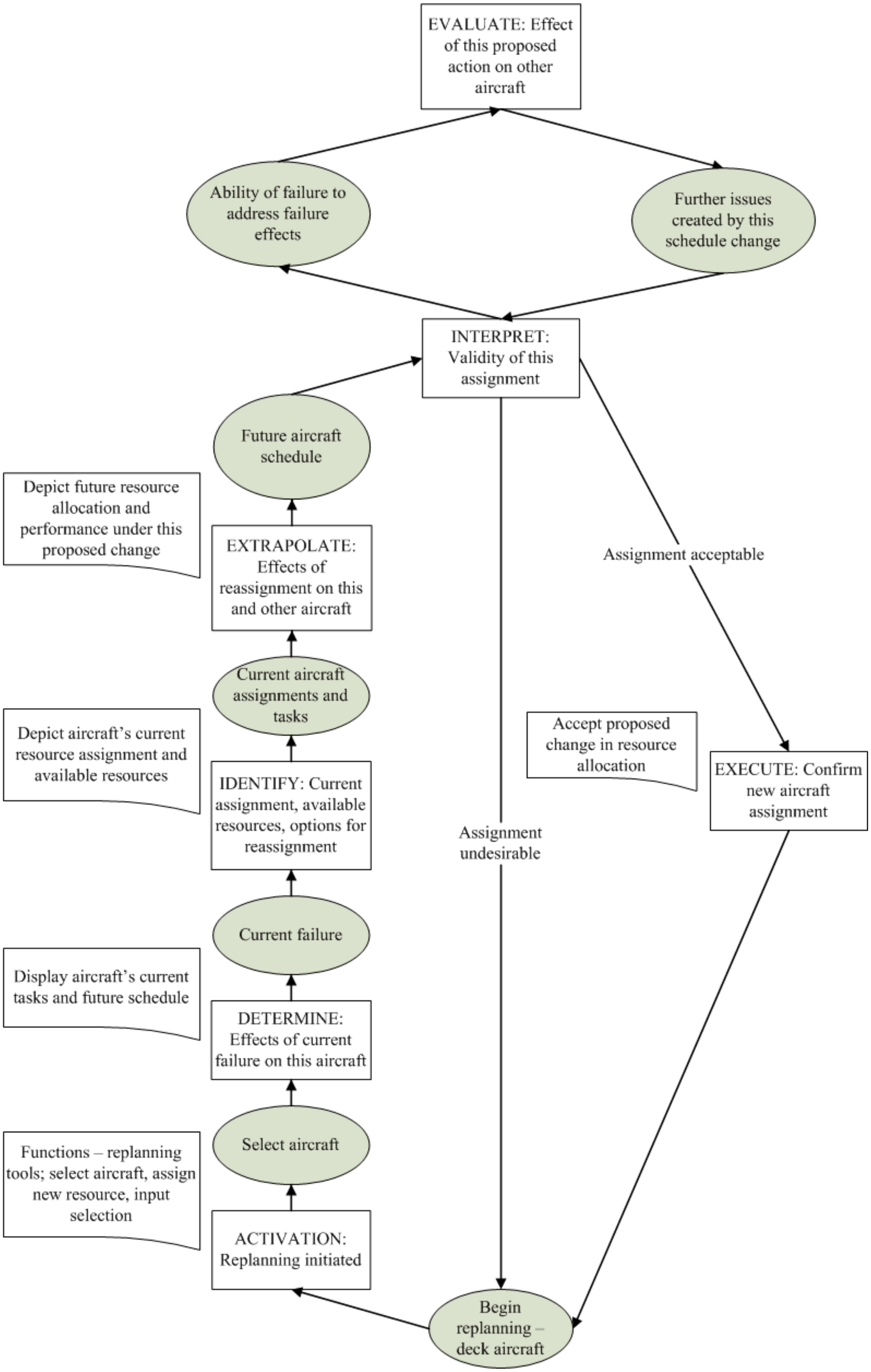

EVALUATE: Effect of this proposed action on other aircraft
Ability of failure to address failure effects
Further issues created by this schedule change
INTERPRET: Validity of this assignment
Future aircraft schedule
Depict future resource allocation and performance under this proposed change
EXTRAPOLATE: Effects of reassignment on this and other aircraft
Assignment acceptable
Current aircraft assignments and tasks
Depict aircraft's current resource assignment and available resources
Accept proposed change in resource allocation
EXECUTE: Confirm new aircraft assignment
IDENTIFY: Current assignment, available resources, options for reassignment
Assignment undesirable
Current failure
Display aircraft's current tasks and future schedule
DETERMINE: Effects of current failure on this aircraft
Select aircraft
Functions – replanning tools; select aircraft, assign new resource, input selection
ACTIVATION: Replanning initiated
Begin replanning – deck aircraft

Figure B. 6. Decision Ladder 3 - New task assignment for aircraft *j* (from EFD 2).

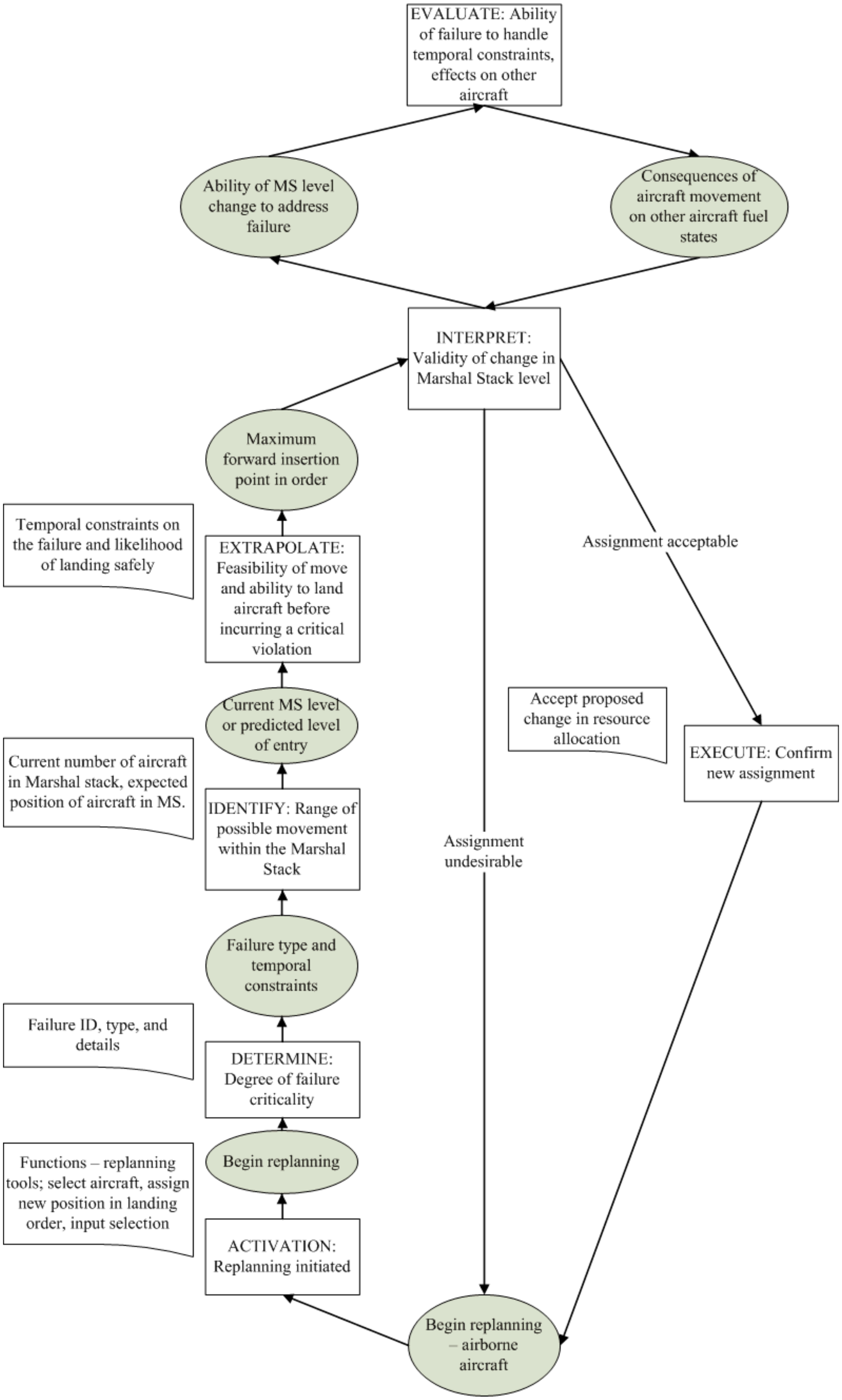

EVALUATE: Ability of failure to handle temporal constraints, effects on other aircraft
Ability of MS level change to address failure
Consequences of aircraft movement on other aircraft fuel states
INTERPRET: Validity of change in Marshal Stack level
Maximum forward insertion point in order
Temporal constraints on the failure and likelihood of landing safely
EXTRAPOLATE: Feasibility of move and ability to land aircraft before incurring a critical violation
Assignment acceptable
Current MS level or predicted level of entry
Accept proposed change in resource allocation
EXECUTE: Confirm new assignment
Current number of aircraft in Marshal stack, expected position of aircraft in MS.
IDENTIFY: Range of possible movement within the Marshal Stack
Assignment undesirable
Failure type and temporal constraints
Failure ID, type, and details
DETERMINE: Degree of failure criticality
Begin replanning
Functions – replanning tools; select aircraft, assign new position in landing order, input selection
ACTIVATION: Replanning initiated
Begin replanning – airborne aircraft

Figure B. 7. Decision Ladder 4 - Define landing position for aircraft *k* (from EFD 2).

In examining the construction of the STO and the resulting DLs, it was identified that a automated planning algorithm could support the efforts of the human operator in accumulating task data for all aircraft in the system, determining the effects of resource failures on these aircraft, and in judging the future performance and relative effectiveness of a replanned schedule. The inclusion of such a system would offload much of the decision loop processes in the *Replanning* EFD (Figure B. 3), but would fundamentally change how the operator interacts with the system. As such, a third STO Phase – *DCAP Replanning* – was created, with a subsequent EFD and set of DLs. These are detailed in the subsequent sections.

B.2 Secondary Analysis

The inclusion of an automated planning algorithm to offload operator replanning tasks changes the tasks and subtasks that an operator performs during the replanning process. Rather than creating specific schedules for aircraft, the operator instead manages inputs to and guides the performance of an automated algorithm. The DCAP system was designed to allow the operator with two levels of interaction, on both global and local levels. Prior research has shown that, while working on a local level provides better performance for an operator, many operators attempt to manage the global priorities of the system [137, 138]. The DCAP system utilizes both aspects, allowing users to rank a set of personnel group variables through a drag-and-drop interface [139] to apply global rankings, while allowing users to specify priority levels and suggest schedules for individual aircraft. This lead to the creation of additional STO phases, EFDs, and DLs.

B.2.1 Scenario Task Overview

An additional STO Phase – *DCAP Replanning* – was created and addresses how the operator would interact with the system in order to input these global and local planning constraints. Table B. 3 contains a list of the subtask elements and related EFD elements for this new phase.

Table B. 3. Scenario Task Overview – *DCAP Replanning* phase and subtasks

Phase	***Task Number***	***Task***	***Related EFD Symbol***
DCAP Replan	1	Determine what item has failed	P6
	2	Determine the type/extent of failure	P7
	3	Define personnel group priorities	P10
	4	Prioritize and suggest schedules for specific aircraft	L7
	5	Select aircraft	P11
	6	Determine priority status of aircraft *i*	D5
	7	Define aircraft as priority	P12
	8	Determine existence of desired schedule	D6
	9	Define suggested schedule	P13
	10	Determine acceptability of proposed schedule	D7

B.2.2 Event Flow Diagrams

An additional EFD was created to incorporate the subtasks contained with the new *DCAP Replanning* STO phase. This appears in Figure B. 8. In this case, only one major decision loop occurs, in which operators must examine all *i* aircraft in the system to determine if any aircraft specifically requires a new schedule suggestion.

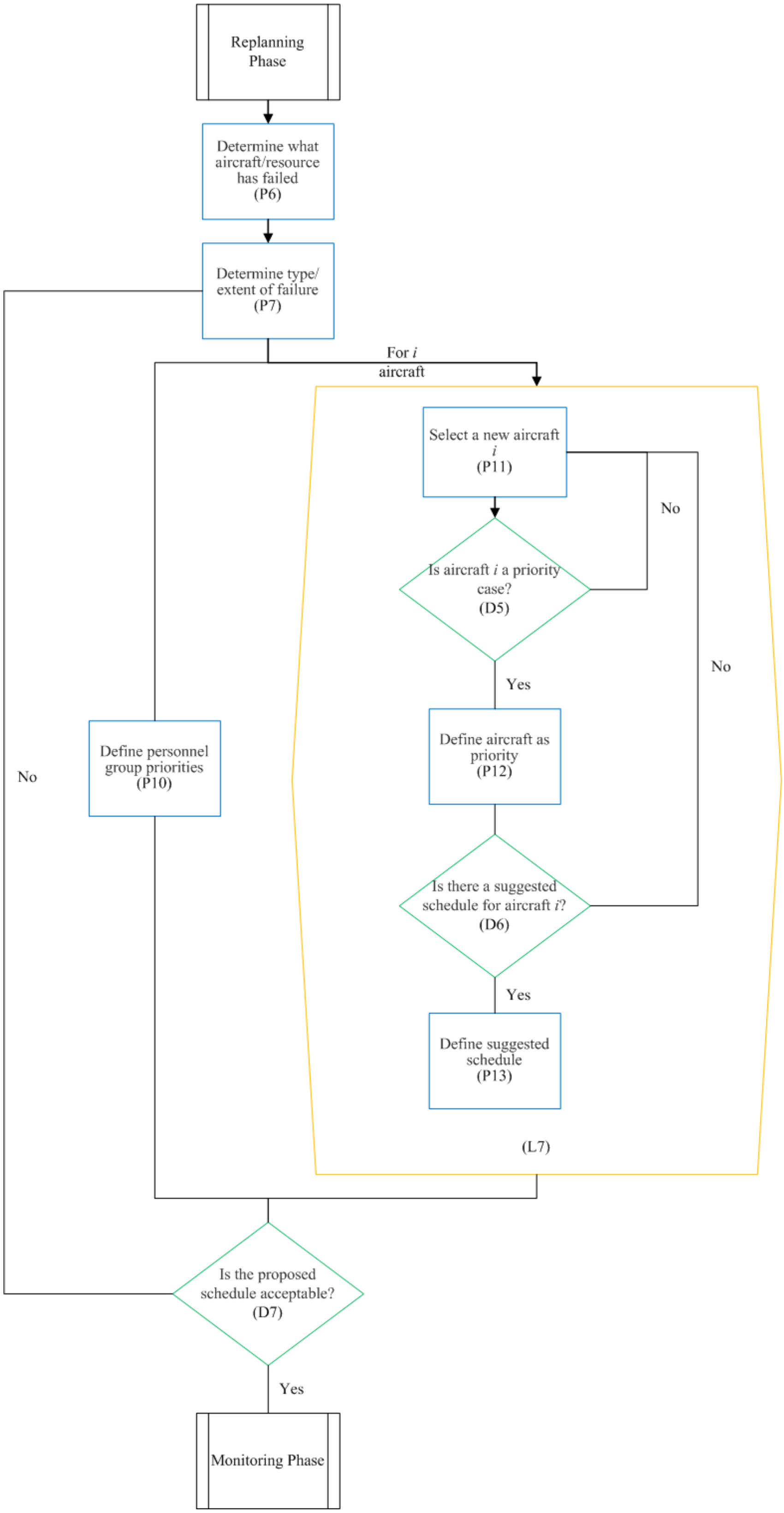

Replanning Phase
Determine what aircraft/resource has failed (P6)
Determine type/ extent of failure (P7)
For *i* aircraft
Select a new aircraft *i* (P11)
No
Is aircraft *i* a priority case? (D5)
No
Yes
Define aircraft as priority (P12)
Define personnel group priorities (P10)
No
Is there a suggested schedule for aircraft *i*? (D6)
Yes
Define suggested schedule (P13)
(L7)
Is the proposed schedule acceptable? (D7)
Yes
Monitoring Phase

Figure B. 8. Event Flow Diagram 3 – *DCAP Replanning*.

B.2.3 Secondary Decision Ladders

The *DCAP Replanning* EFD contains three new Decision elements, addressing how users determine the priority status of an aircraft, if the operator has a suggested schedule for the aircraft, and if the returned schedule proposed is acceptable. The DL for this element appears in Figure B. 9.

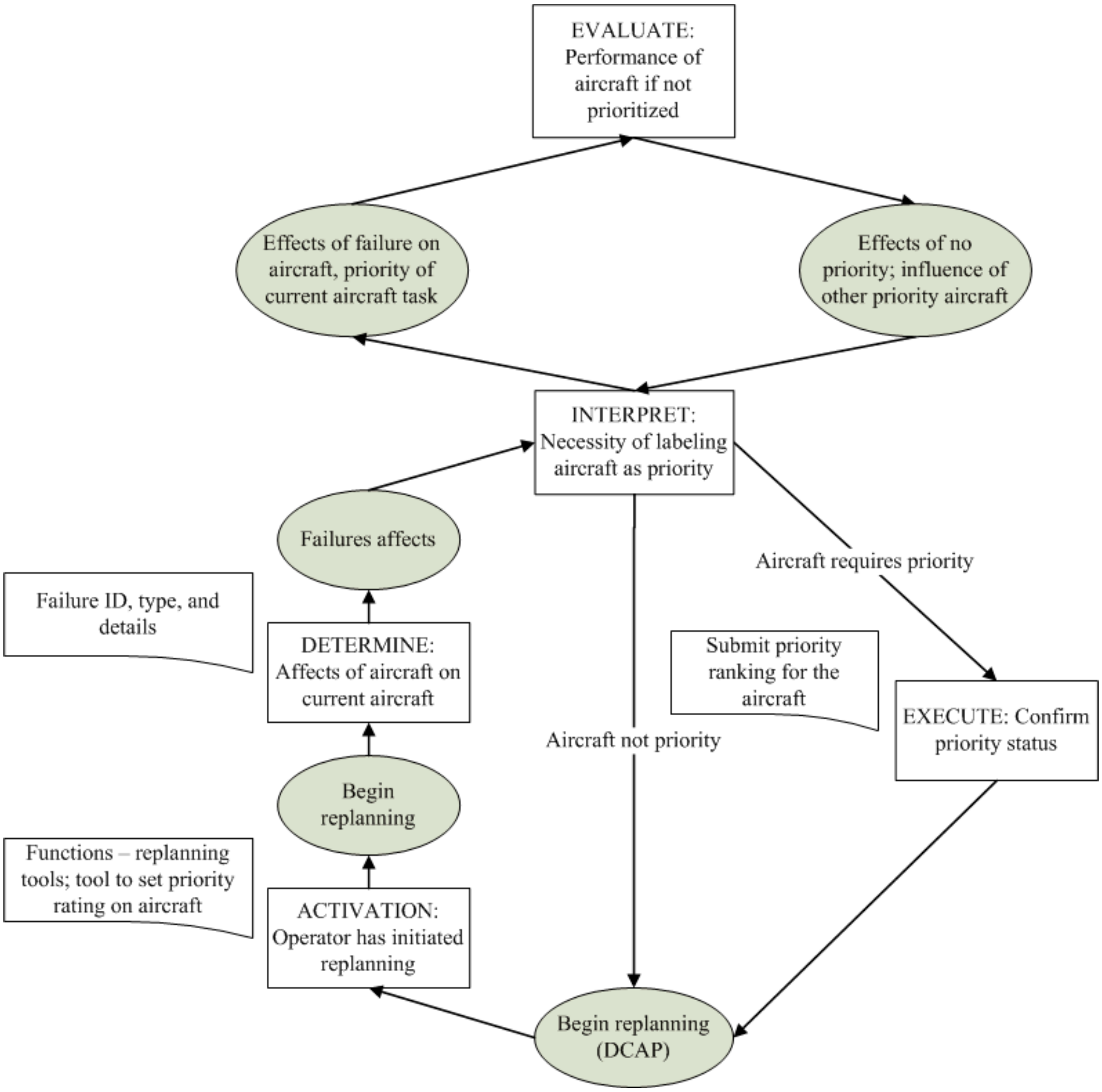

EVALUATE: Performance of aircraft if not prioritized
Effects of failure on aircraft, priority of current aircraft task
Effects of no priority; influence of other priority aircraft
INTERPRET: Necessity of labeling aircraft as priority
Failures affects
Aircraft requires priority
Failure ID, type, and details
DETERMINE: Affects of aircraft on current aircraft
Submit priority ranking for the aircraft
EXECUTE: Confirm priority status
Aircraft not priority
Begin replanning
Functions – replanning tools; tool to set priority rating on aircraft
ACTIVATION: Operator has initiated replanning
Begin replanning (DCAP)

Figure B. 9. Decision Ladder 5 - Is aircraft *i* a priority case (from EFD 3).

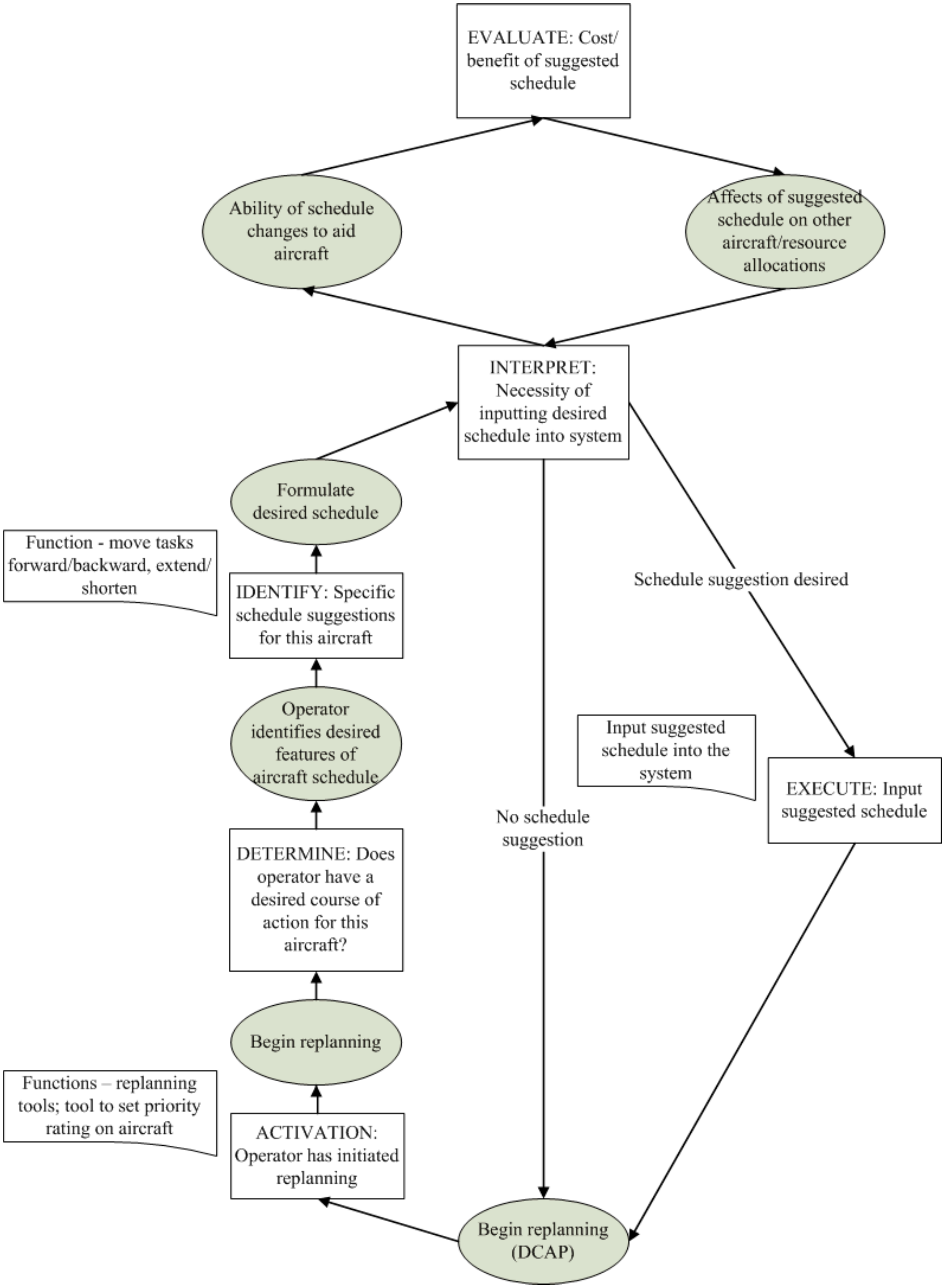

EVALUATE: Cost/ benefit of suggested schedule
Ability of schedule changes to aid aircraft
Affects of suggested schedule on other aircraft/resource allocations
INTERPRET: Necessity of inputting desired schedule into system
Formulate desired schedule
Function - move tasks forward/backward, extend/ shorten
IDENTIFY: Specific schedule suggestions for this aircraft
Schedule suggestion desired
Operator identifies desired features of aircraft schedule
Input suggested schedule into the system
EXECUTE: Input suggested schedule
No schedule suggestion
DETERMINE: Does operator have a desired course of action for this aircraft?
Begin replanning
Functions – replanning tools; tool to set priority rating on aircraft
ACTIVATION: Operator has initiated replanning
Begin replanning (DCAP)

Figure B. 10. Decision Ladder 6 - Suggested schedule for aircraft *i* (from EFD 3).

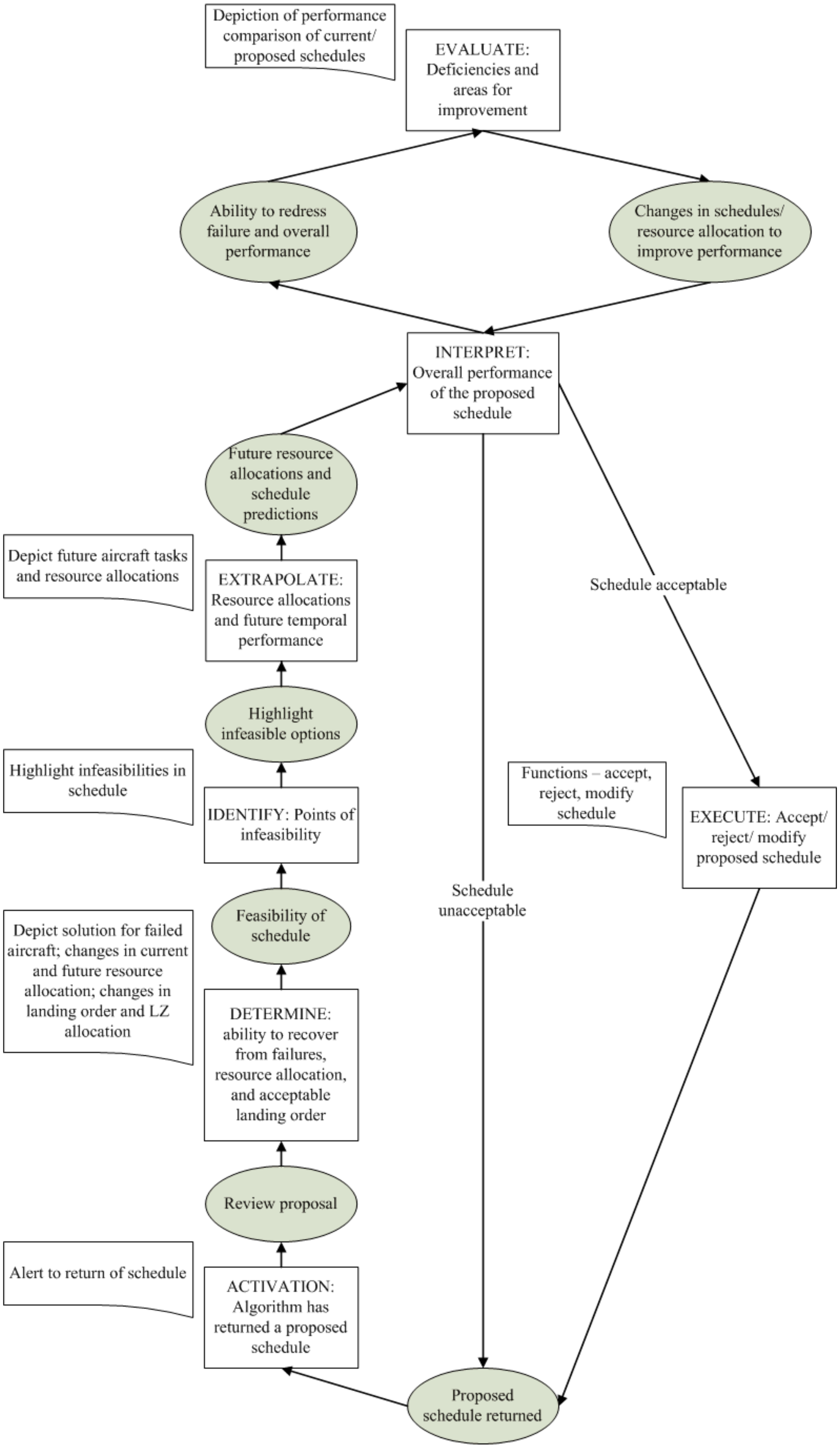

Depiction of performance comparison of current/ proposed schedules
EVALUATE: Deficiencies and areas for improvement
Ability to redress failure and overall performance
Changes in schedules/ resource allocation to improve performance
INTERPRET: Overall performance of the proposed schedule
Future resource allocations and schedule predictions
Depict future aircraft tasks and resource allocations
EXTRAPOLATE: Resource allocations and future temporal performance
Schedule acceptable
Highlight infeasible options
Highlight infeasibilities in schedule
IDENTIFY: Points of infeasibility
Functions – accept, reject, modify schedule
EXECUTE: Accept/ reject/ modify proposed schedule
Feasibility of schedule
Schedule unacceptable
Depict solution for failed aircraft; changes in current and future resource allocation; changes in landing order and LZ allocation
DETERMINE: ability to recover from failures, resource allocation, and acceptable landing order
Review proposal
Alert to return of schedule
ACTIVATION: Algorithm has returned a proposed schedule
Proposed schedule returned

Figure B. 11. Decision Ladder 7 - Is proposed schedule acceptable (from EFD 3).

B.2.4 Situational Awareness Requirements

From these EFDs, a set of Situational Awareness Requirements (SARs) can be generated, describing specific bits of information that must be displayed to the operator to ensure their awareness of the current operating state of the system. A total of 29 SARs were created for these two EFDs. Because the third EFD (*DCAP Replanning*) supersedes the second (*Replanning*), its SARs do not appear in this table.

Table B. 4. List of Situational Awareness Requirements (SARs).

	Level 1 (Perception)	Level 2 (Comprehension)	Level 3 (Projection)
SAR1	Visual depiction of crew on deck (L1)	Location, current action, and safety of crew members (L1)	Destination and future safety of crew members (L1)
SAR2	Visual depiction of aircraft on deck (L1)	Location, current action, and safety of deck aircraft (L1)	Destination and future safety of deck aircraft (L1)
SAR3	Visual depiction of deck vehicles (L1)	Location, current action, and safety of deck vehicles (L1)	Destination and future safety of deck vehicles (L1)
SAR4	Visual depiction of airborne aircraft (L2)	Location, current action, and safety of airborne aircraft (L1)	Destination and future safety of airborne aircraft (L1)
SAR5	Visual depiction of current landing order (L2)	Current landing order (L2)	-
SAR6	Visual depictions of current fuel states (L2)	Fuel levels of each aircraft (L2)	Likelihood of aircraft having insufficient fuel (L2)
SAR7	Visual depiction of catapults and status (L3)	Availability and operability of catapults (L2)	Flexibility of launch assignments (L2)
SAR8	Visual depiction of status of fuel stations (L3)	Availability and operability of fuel stations (L2)	-
SAR9	Visual depiction of status of elevators (L3)	Availability and operability of elevators (L2)	-
SAR10	Visual depiction of status of landing strip (L3)	Availability and operability of landing strip (L2)	Repercussions on airborne aircraft (L2)
SAR11	Visual depiction of current time of deck crew (L4)	Fatigue level of the crew (L4)	Time remaining before shift end (L4)
SAR12	Visual depiction of current shift time of pilots (L5)	Fatigue level of the pilot (L4)	-

Table B. 5. List of Situational Awareness Requirements (SARs), continued.

	Level 1 (Perception)	Level 2 (Comprehension)	Level 3 (Projection)
SAR13	Visual/auditory alert to new aircraft failures (L2)	Newly developed aircraft failure (L2)	Effects of failure on aircraft schedule, safety (L2)
SAR14	Visual/auditory alert to new resource failures (L3)	Newly developed resource failure (L3)	Effects of failure on aircraft schedule, safety (L2)
SAR15	Visual depiction of current resource allocations (L6)	Current allocation of aircraft to catapults/LZ (L6)	Bottlenecks in resource usage (L6)
SAR16	Visual depiction of current aircraft schedules (L6)	Current schedule of operations and level of delay (L6)	Effects of delays on overall operational time (L6)
SAR17	Visual depiction of current aircraft failures (P6)	Currently known aircraft failures (P6)	Aggregate effects of failures on schedule (P6)
SAR18	Visual depiction of current resource failures (P6)	Currently known resource failures (P6)	Aggregate effects of failures on schedule (P6)
SAR19	List of personnel group rankings (P10)	Previous rankings of personnel groups (P10)	-
SAR20	Currently selected aircraft (P11)	Aircraft currently being ranked (P11)	-
SAR21	Current priority status of all aircraft (P12)	Previous priority status of aircraft (P12)	-

B.3 Final Information and Functional Requirements

The final step in the hCTA process is the definition of a set of Information Requirements (IRs) for the resulting system display. These come jointly from the SARs and Decision Ladders developed from the Event Flow Diagrams. Table B. 6 lists the IRs for the DCAP interface; requirements are linked back to their corresponding Decision Ladder or EFD element. These information requirements support four main functions of the DCAP System – *Monitoring* the state of the world, *Alerting* the user to failures in the system, *Predicting* the future performance of the schedule, and *Supporting* the operator in replanning.

Table B. 6. Information Requirements for the DCAP Interface.

Information Requirements		
IR1	Location of all crew on deck	SAR1
IR2	Location of all deck aircraft	SAR2
IR3	Location of all airborne aircraft	SAR3
IR4	Location of all deck vehicles	SAR4
IR5	Current landing order	SAR5, DL4
IR6	Current fuel states	SAR6
IR7	Current catapult status (available and operational)	SAR7
IR8	Current fuel station status (available and operational)	SAR8
IR9	Current elevator status (available and operational)	SAR9
IR10	Current landing strip status (available and operational)	SAR10
IR11	Current work time of crew	SAR11
IR12	Current work time of pilots	SAR12
IR13	Alert for new aircraft failures	SAR13, DL1
IR14	Alert for new resource failures	SAR14, DL1
IR15	Current resource allocation	SAR15, DL5, DL6
IR16	Current schedules for all aircraft	SAR16, DL5, DL6
IR17	Currently existing aircraft failures	SAR17
IR18	Currently existing resource failures	SAR18
IR19	Current personnel group rankings	SAR19
IR20	Currently selected aircraft	SAR20
IR21	Current priority status of all aircraft	SAR21
IR22	Visual depiction of failure ID	DL1, DL5
IR23	Visual depiction of failure type	DL1, DL5
IR24	Visual depiction of failure details	DL1, DL5
IR25	Description of future schedule	DL1
IR26	Description of future resource allocations	DL1
IR27	Visual display of current aircraft task	DL5, DL6
IR28	Visual display of aircraft's upcoming tasks	DL5, DL6
IR29	Visual display of available resources	DL6
IR30	Visual display of future resource allocations	DL6
IR31	Visual depiction of performance under proposed assignment	DL6
IR32	Project aircraft position in Marshal Stack	DL5, DL6
IR33	Temporal constraints on aircraft failure	DL5, DL6
IR34	Visual/auditory alert to return of proposal	DL7
IR35	Visual depiction of solution for failed aircraft	DL7
IR36	Visual depiction of changes in resource allocation	DL7
IR37	Visual depiction of changes in aircraft schedules	DL7
IR38	Visual notation of points of infeasibility (non-adherence) for priority aircraft	DL7
IR39	Visual depiction of predicted aircraft schedules	DL7
IR40	Visual depiction of predicted resource allocation	DL7
IR41	Visual depiction relative schedule performance.	DL7

APPENDIX C – TUTORIAL FOR THE DCAP SIMULATION

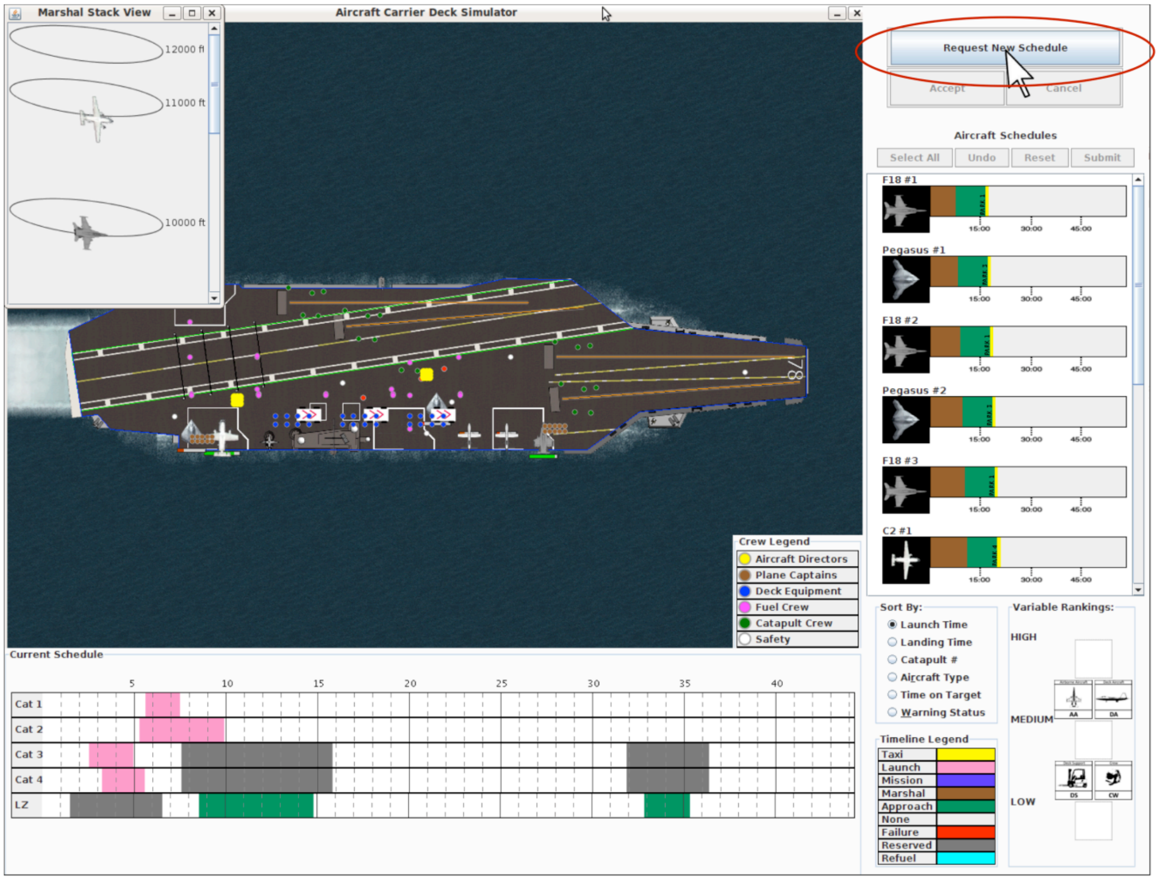
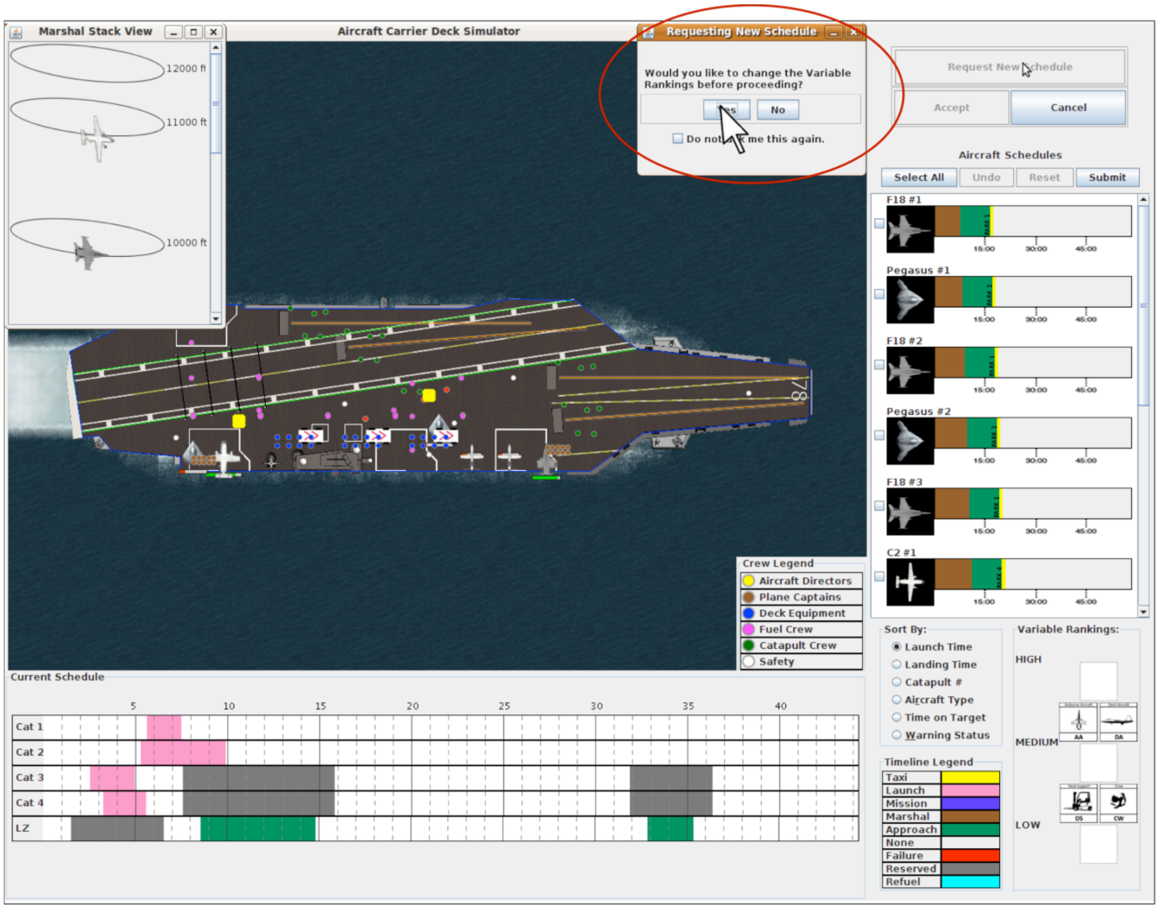


Figure C. 1. Tutorial Step 1 – requesting a schedule.

Step 1: Click the *Request Schedule* button in the upper right portion of the screen.



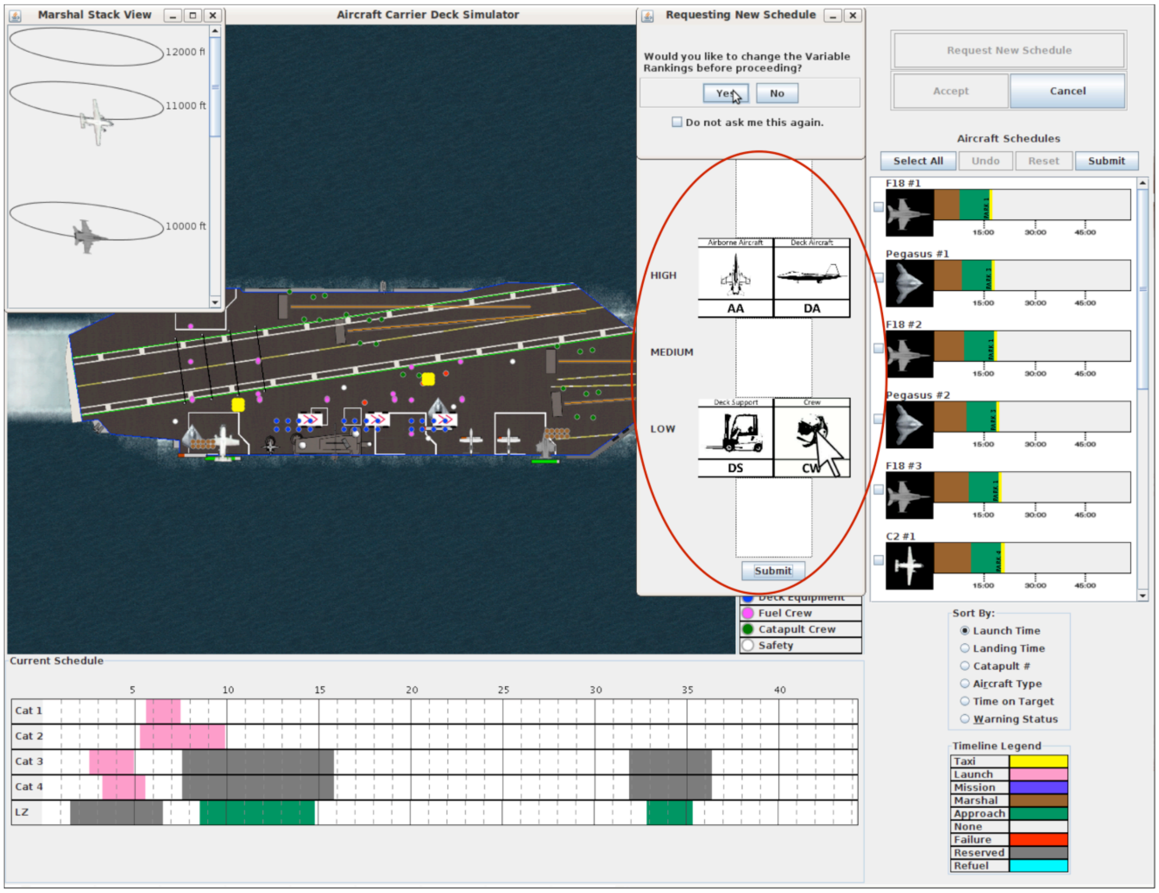
Would you like to change the Variable Rankings before proceeding?

☐ Do not ask me this again.

Sort By: ● Launch Time

Figure C. 2. Tutorial Step 2 – select whether to change variable rankings.

Step 2. The user is given the option of changing the rankings in the *Variable Ranking Tool* (VRT). Clicking "Yes" brings this window to the interior of the screen and makes it actionable. Clicking "No" skips the re-ranking step and leaves the variables with their current value. The user also has to option to permanently skip this step through a checkbox at the bottom of the frame.



Would you like to change the Variable Rankings before proceeding?

Figure C. 3. Tutorial Step 3 – changing variable rankings.

Step 3. If users decide to re-rank the four major personnel groups, they simply click and drag icons within the five levels in the VRT frame. Variables can be ranked in any manner within the five levels provided – all on one level, each on a separate level, and any combination in between. When users are satisfied with the rankings, they click the “Submit” button at the bottom of the screen to submit the rankings to the algorithm.

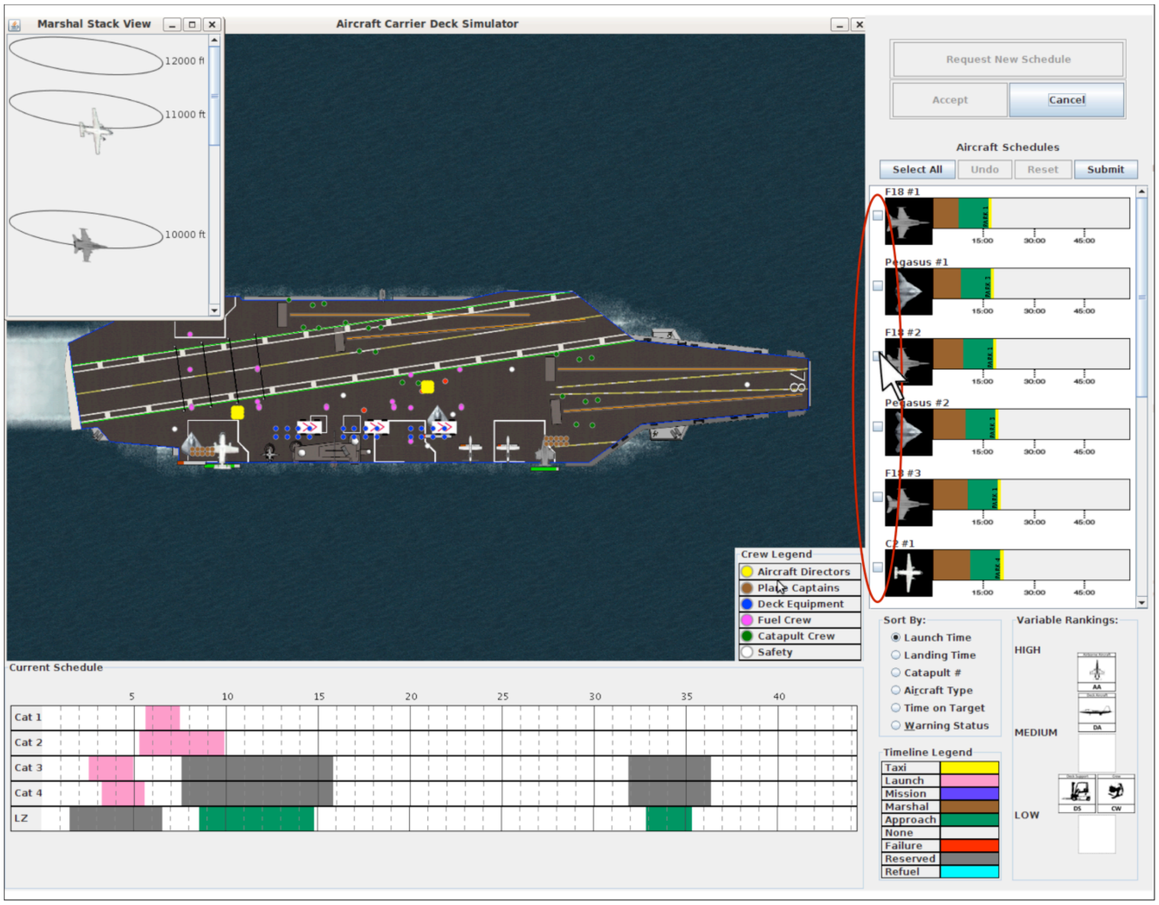


Figure C. 4. Tutorial Step 4– defining aircraft priorities.

Step 4. Clicking “Request Schedule” causes a set of checkboxes to appear next to aircraft in the *Aircraft Schedule Panel* (ASP). Checking one of these boxes designates the associated aircraft as “priority” to the algorithm. This is a binary condition – aircraft are either priority cases, or they are not. Users may assign priority to all aircraft or to none.

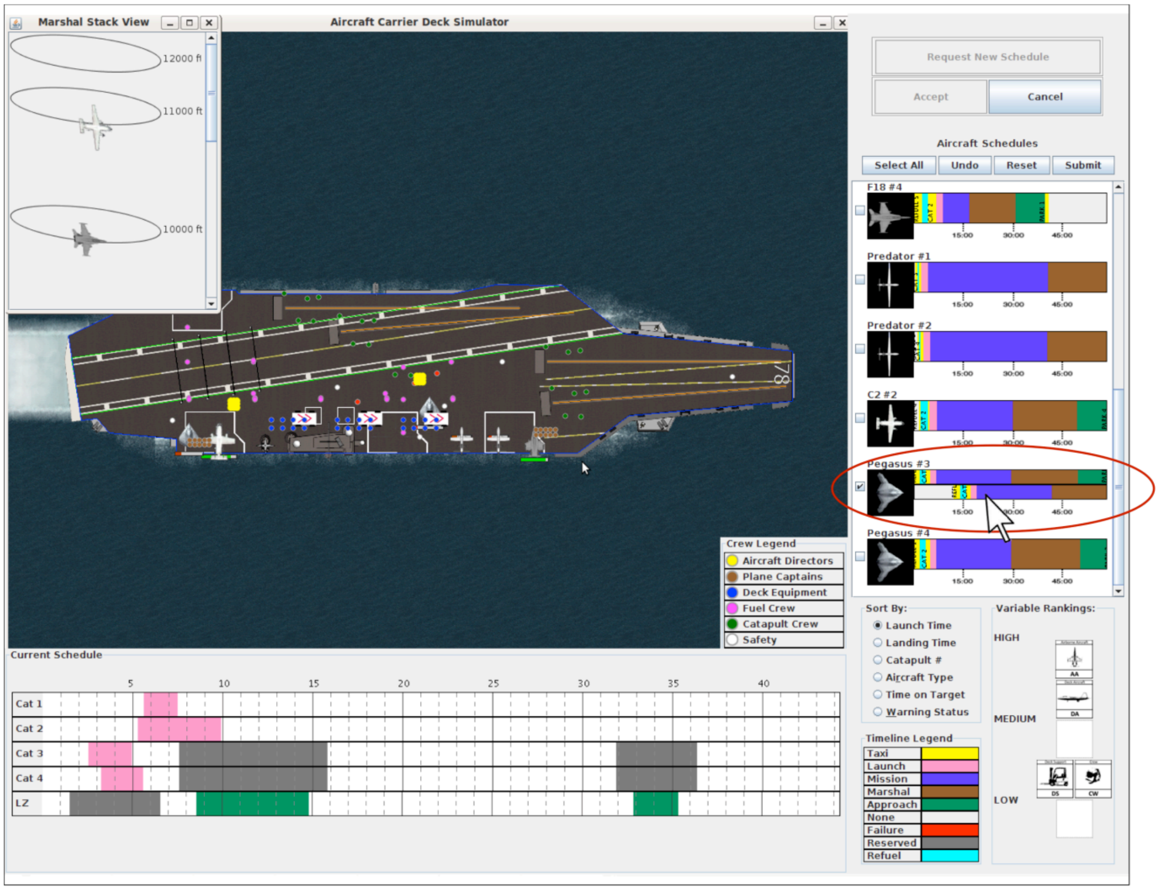


Figure C. 5. Tutorial Step 5 – suggesting aircraft schedules.

Step 5. Defining an aircraft as priority leads to an additional change in the ASP – the timeline for the associated aircraft splits horizontally into two segments. The upper segment depicts the current operating schedule of the aircraft. The bottom half of the bar will be used to submit the operator's desired schedule for this aircraft to the algorithm. When the timeline is split, the bottom bar becomes actionable and can be dragged left or right to accelerate or postpone the aircraft's schedule of tasks. In certain cases, individual tasks can be lengthened or shortened by changing the size of the associated color block. When users have completed their specification, they press the "Submit" button above the first aircraft in the list.

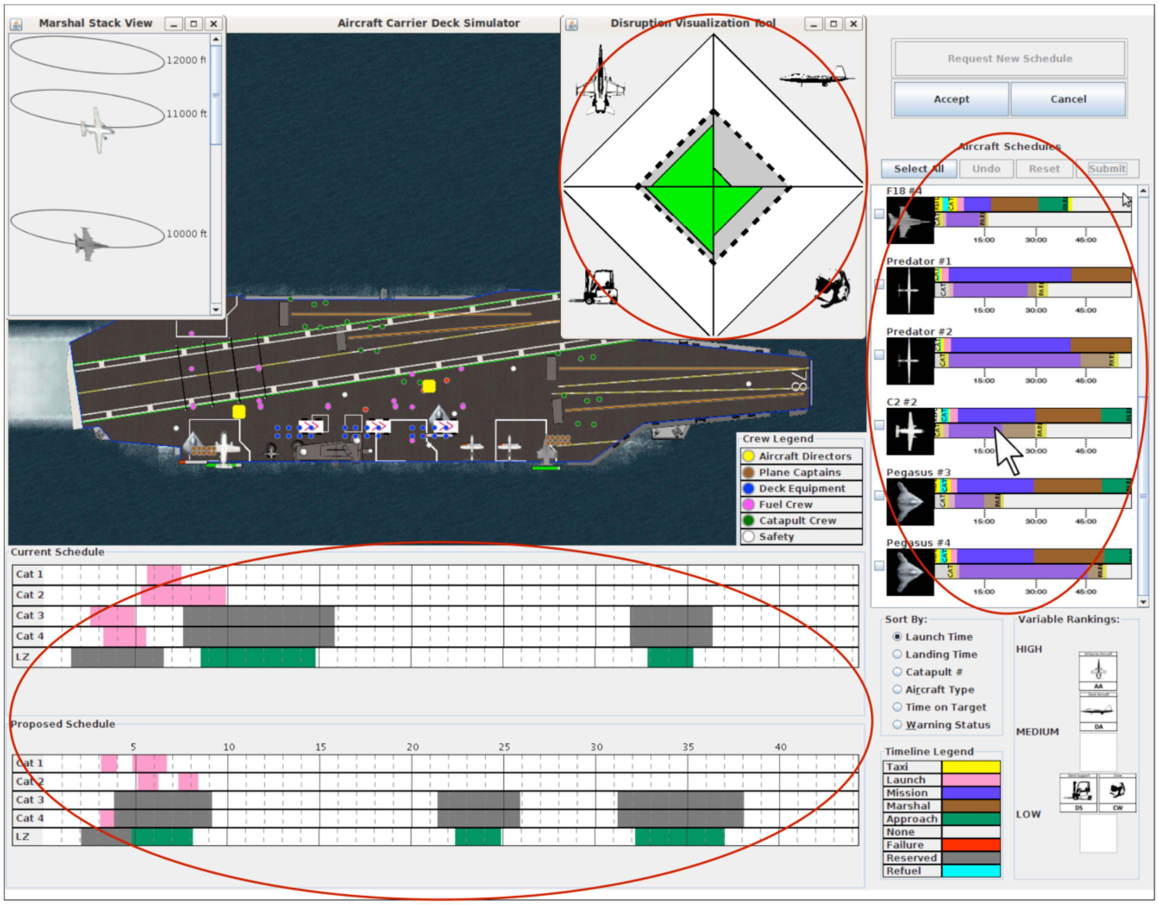


Figure C. 6. Tutorial Step 6 – review the proposed schedule.

Step 6. After pressing both "Submit" buttons (one each in the ASP and DVT), the algorithm processes the user inputs in light of the current state of operations on deck, creating a schedule proposal. This proposal is displayed to the operator through modifications of the ASP, the *Deck Resource Timeline* (DRT), and the appearance of the *Disruption Visualization Tool* (DVT). The ASP and DRT modifications display the current schedule on top and the proposed schedule on bottom (akin to the method of suggesting aircraft schedules in the previous step), allowing users to make a one-to-one comparison of changes in the schedules. The DVT shows the relative effectiveness of the schedule overall. Green triangles signify that all tasks for a variable group are completed in less operating time, while red triangles signify that the new schedule demands more operating time of this group. These states are reinforced by the size of the triangle as determined by its edge distance from the dashed black line. This line depicts no change between the schedules, such that green triangles appear within the dashed line, and red triangles appear outside it.

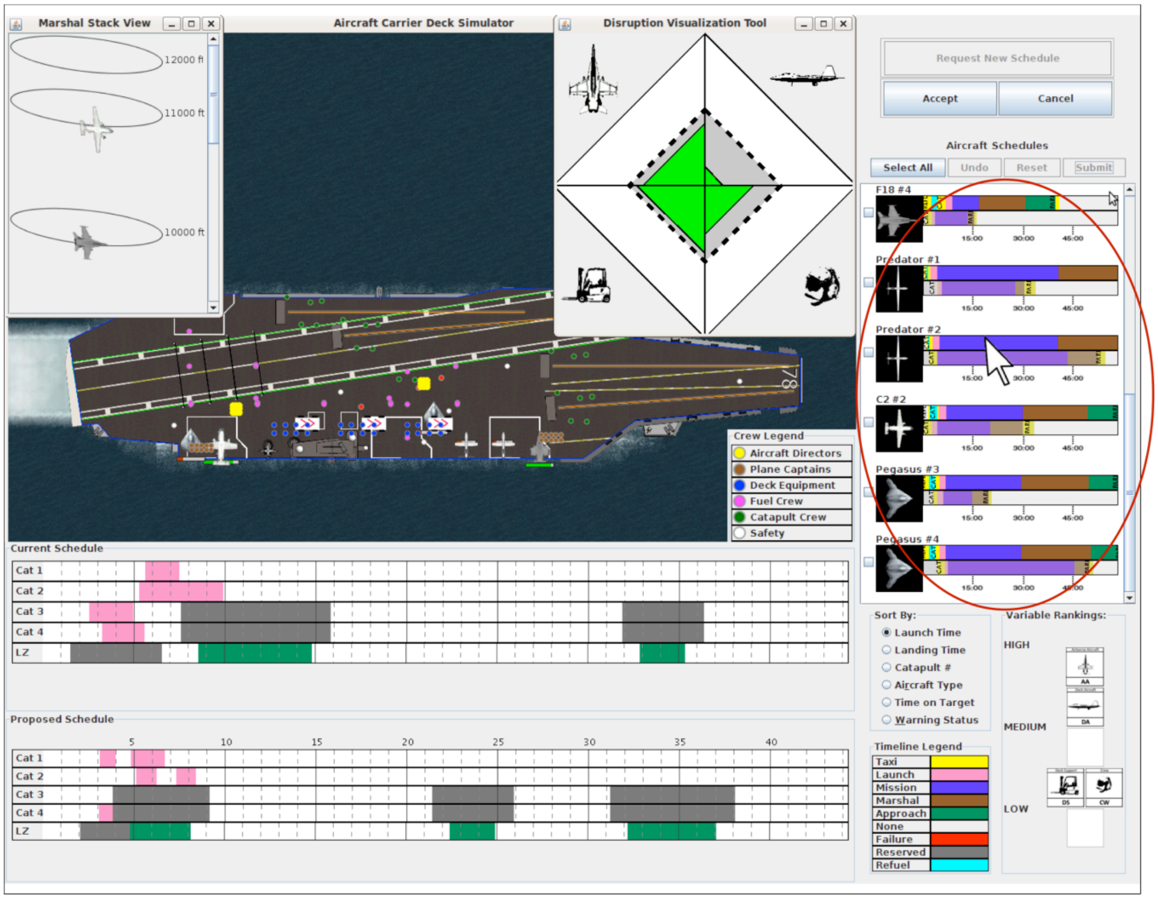


Figure C. 7. Tutorial Step 7 – adjust proposed schedule.

Step 7. The user retains the ability to make further suggestions to the scheduler at this juncture. The ASP timeline bars remain actionable, giving users the option to attempt to fine-tune the proposed schedule before implementation. Doing so, however, invalidates the current schedule and forces the algorithm to perform a new round of scheduling calculations. If changes are made, the user would again press “Submit” to initiate schedule generation.

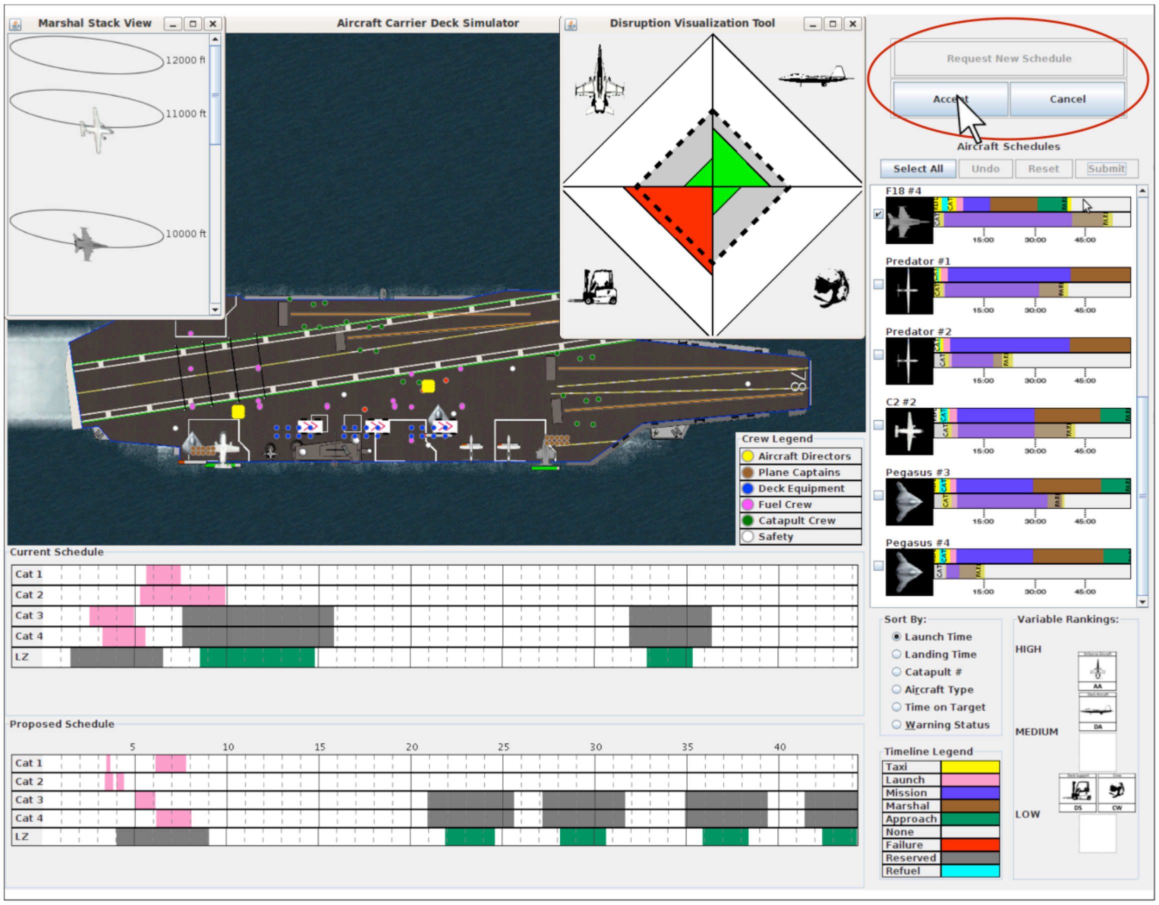
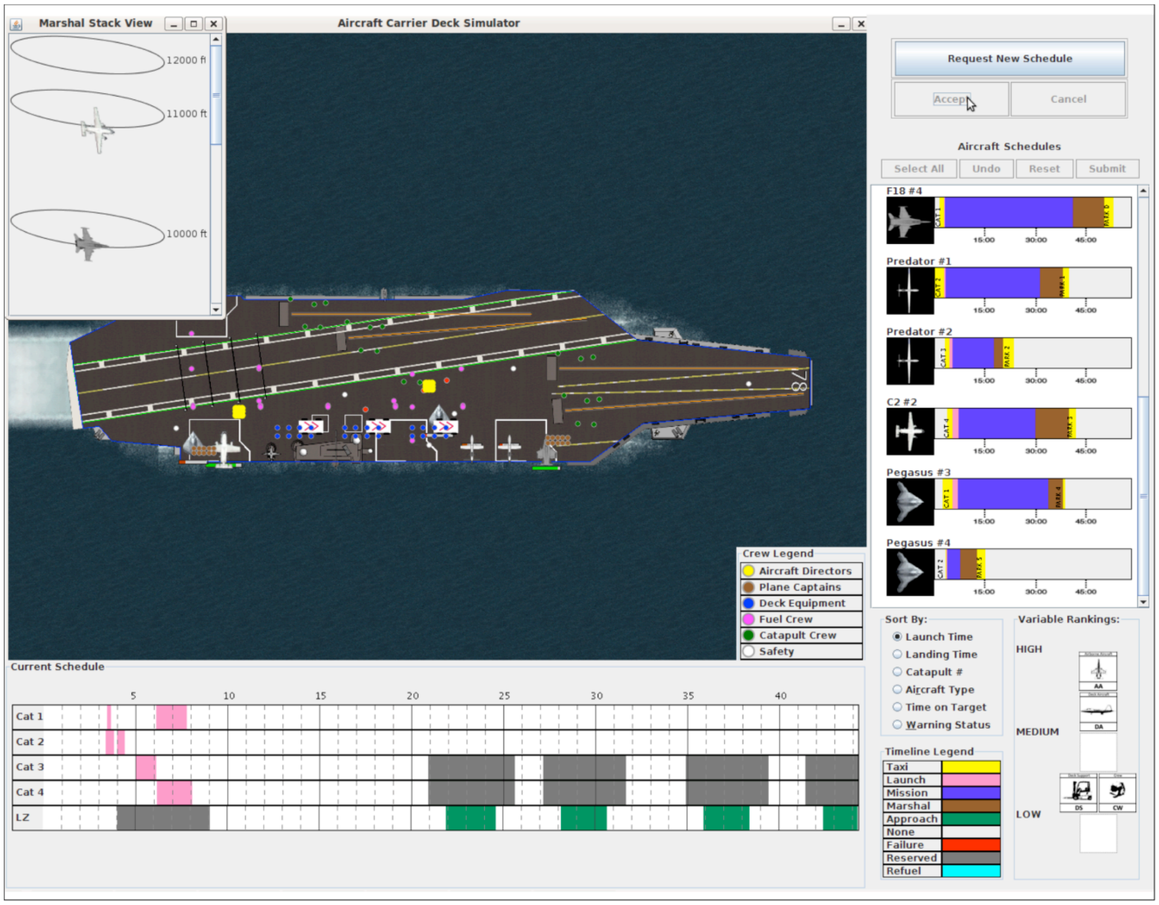


Figure C. 8. Tutorial Step 8 – accepting the proposed schedule.

Step 8. When users have a desirable schedule, they press "Accept" in the upper right corner of the interface to implement the schedule into the environment.



Sort By:

● Launch Time
○ Landing Time
○ Catapult #
○ Aircraft Type
○ Time on Target
○ Warning Status

Figure C. 9. Tutorial Step 1 – returning to a monitoring state.

Step 9. Pressing the “Accept” button then returns the operator to the monitoring state until a need to request a new schedule arises.

APPENDIX D – EXPERT USER REPLANNING ACTIONS

D.1 REPLANNING ACTIONS FOR THE EXPERT USER

As noted previously, in a further effort to reduce variability in the system so as to accurately depict the performance of the algorithm, the replanning actions of the Expert User were standardized according to the details in each scenario. The actions taken by the Expert User were dictated by the application of the expert user heuristics discussed in Chapter 4.1.1. The following sections describe the standard actions that were taken during replanning for each scenario.

D.1.1 PLANNING FOR THE SIMPLE SCENARIO

In performing manual planning for the Simple scenario, SME heuristics 1, 2, 4, and 6 were applied. This does not imply, however, that changes to other aircraft schedules did not occur. Table D.1 lists the aircraft whose new schedules resulted in catapult reassignments and the order in which replanning occurred. Table D.2 lists the new assignments list with the desired ordering of launches; asterisks (*) denote aircraft whose assignment was changed.

Table D.1. List of aircraft with catapult reassignments.

	Original Assignment	**New Assignment**
SMAC #2	3	2
FMAC #2	3	2
FMAC #5	3	2
FMAC #8	3	2
FMAC #11	3	4
FUAV #2	3	4
FUAV #4	2	4

Table D.2. Schedule assignment after manual replanning
asterisks (*) denote aircraft whose assignment was changed.

Catapult 1	Catapult 2	Catapult 3	Catapult 4
Disabled	-	SUAV #2	SUAV #1
	-	Disabled	SMAC #1
	SMAC #2*		FMAC #1
	FMAC #2*		FMAC #4
	FMAC #3		FMAC #7
	FMAC #5*		FMAC #10
	FMAC #6		FMAC #11*
	FMAC #8*		FUAV #1
	FMAC #9		FUAV #2*
	FMAC #12		FUAV #4*
	FUAV #3		-

In performing the manual, human-only replanning, the user clicked on the aircraft, pressed a trigger key (the *F1* key) on the computer keyboard, and then clicked on the destination catapult. After making the assignment to the new catapult, the aircraft immediately implemented the new schedule and began moving to its new destination. After completing all reassignments, the user pressed a second, different trigger key (*F10*) on the computer keyboard signaled the end of replanning. After this, no further user action was required and the scenario could run to completion.

In performing the automated planning for this Simple scenario, the user first clicked the "Request Schedule" button in the upper right hand corner of the screen. Given that this scenario is a launch configuration, the variable rankings were set to assign Deck Aircraft as the highest priority, Crew Working and Deck Support vehicles as medium priority, and Airborne Aircraft (none in the system) as the lowest priority. In the Aircraft Schedule Panel, the same aircraft that were given schedule changes in the manual replanning case (Table D.2) were given priority designations. However, no schedule changes were suggested; the schedules were left in their current state, signaling to the scheduler a desire to adhere to the current launch time schedule. In the returned schedule, shifts forward were acceptable, but movements backward in time would not be accepted. After receiving the new plan, the user reviewed the schedules for each of these aircraft, looking to see if the new schedule satisfied these criteria.

D.1.2 Planning for the Moderate Scenario

In the Moderate Scenario, only two aircraft required replanning. The first aircraft was a Fast Manned Aircraft that encountered a hydraulic leak, the second a Slow Manned Aircraft that encountered a fuel leak. In replanning for this scenario, five SME heuristics were applied (1, 2, 3, 7, and 9). The main emphasis is on Heuristic 9 and the differentiation between True and Urgent emergencies. Fuel leaks are considered a major, True emergency that must be handled immediately. The hydraulic leaks, as modeled in the system, develop more slowly and are considered to be less problematic. Furthermore, hydraulic failures increase the likelihood of accidents and debris at landing, which may further limit the ability of the deck to recover aircraft and create more failures in the system.

For the manual, human-only planning condition, this resulted in moving the fuel leak aircraft forward in the Marshal Stack, assigning it the most immediate landing slot, and moving the hydraulic failure backwards to the last *manned* position in the Marshal Stack. This preserves Heuristic 3 (Safety of Pilots and Crew). In executing these actions, the user again clicked the aircraft, pressed *F1*, and then clicked either the deck (to issue an immediate landing order) or an alternate area signifying a move to the back of the Marshal Stack. The user again pressed *F10* to signify the conclusion of the planning activities.

For the human-algorithm combined planning, group variables were set opposite to that of the Simple scenario. Crew Working and Deck Support vehicles still received a moderate rating, but the Airborne Aircraft were now of primary concern. Deck Aircraft were moved to the lowest setting, as there are none in the system. In the *Aircraft Schedule Panel*, the two failed aircraft were assigned priority ratings, but in this case, the Expert User provided suggestions for the schedules for these aircraft. The SMAC with the fuel failure is given a suggestion to move forward in time, while the FMAC with the hydraulic failure is suggested to delay operations. After receiving the proposed schedule, the user reviewed the aircraft schedules in order to ensure that the SMAC is moved forward in time, as this is the primary concern. The backward movement of the FMAC is a secondary concern, as the suggested schedule may actually induce a limit violation and be unacceptable for the planning algorithm.

D.1.3 Planning for the Complex Scenario

The Complex scenario includes aspects of both the Simple and Complex cases, and thus replanning included all of the above Heuristics (1, 2, 3, 4, 6, 7, and 9). For the manual case, application of these heuristics resulted in moving the SMAC with fuel leak forward in the Marshal Stack, inserting it into the first available landing configuration. However, planning for the aircraft on deck required ensuring that these aircraft were not sent to Catapults 3 and 4. If one of these aircraft were to become incapacitated in the Landing Strip, the SMAC would be placed in serious danger. As such, both of there aircraft were sent to Catapult 2, ensuring that no conflicts were created with the landing strip. The physical actions to do this are identical to those used for the deck and airborne aircraft in the previous two scenarios.

For the human-algorithm case, the strategy was similar to the Moderate case. Variable rankings were slightly adjusted due to the lower fuel levels of aircraft in this scenario; while Airborne Aircraft retained the highest priority, all other variables were placed in the lowest priority bin. This placed even greater emphasis on the critical nature of the fuel states of these aircraft. The failed aircraft was given a priority designation in the ASP with suggestion to accelerate the schedule. When reviewing the proposed plan, the user examined individual aircraft schedules to ensure that the failed SMAC was given a schedule that moved it forward in time.

APPENDIX E – METRICS WITH LOW EXTERNAL VALIDITY

Class - Subclass	*Measure*	*Abbreviation*	*Reason*
Mission Efficiency - Error	Hydraulic fluid violations	HFV	Never occurred in testing
	Total violations	TV	Since HFV did not occur, this metric is also invalid
Mission Efficiency – Time	Total UGV Active Time	TUAT	Lack of external validity; interviews with Subject Matter Experts revealed little concern for the deck support vehicles
	Wait Time in Queue - Catapults	WTQC	Not a concern for SMEs; a better measure is Total Aircraft Taxi time
	Wait Time in Queue - Crew	WTQCrew	Not a concern for SMEs; a better measure is Total Aircraft Taxi time
Mission Efficiency – Coverage	Catapult 1 Launches	C1L	Catapult 1 always disabled and value is always zero
	Catapult 2 Launches	C2L	Discrete counts with no variations; statistical tests are not required.
	Catapult 3 Launches	C3L	Discrete counts with no variations; statistical tests are not required.
	Catapult 4 Launches	C4L	Discrete counts with no variations; statistical tests are not required.
Algorithm Behavior Efficiency	Wait Time due to Processing	WTP	Algorithm processing always << 1 sec; measure provides no diagnostic help.
Human Behavior Efficiency	Wait Time due to Operator	WTO	Not a concern for SMEs

APPENDIX F – PRINCIPAL COMPONENTS ANALYSIS RESULTS

Table F. 1. PCA Results for the Simple scenario, Baseline planning condition.

Correlations Transformed Variables

	TATT	TAAT	TCAT	TUAT	TAT	WTC	WTCrew	PHV	PHV_D	SHV	SHV_D	MD	C2LR	C3LR	C4LR	TCLR	LZFT
TATT	1.000	.635	-.089	.059	.652	.788	.170	.343	.562	.480	.566	.770	-.570	-.782	-.782	-.723	.750
TAAT	.635	1.000	.266	.401	.939	.499	.470	.070	.591	.142	.310	.740	-.723	-.605	-.605	-.484	.741
TCAT	-.089	.266	1.000	.330	.219	-.212	.342	-.488	.257	-.587	-.006	.079	-.116	.141	.141	.196	.041
TUAT	.059	.401	.330	1.000	.496	-.317	.899	-.275	.567	-.314	-.010	.046	-.126	.144	.144	.213	.012
TAT	.652	.939	.219	.496	1.000	.534	.514	.131	.696	.181	.373	.766	-.752	-.628	-.628	-.526	.777
WTC	.788	.499	-.212	-.317	.534	1.000	-.281	.394	.349	.571	.496	.847	-.669	-.959	-.959	-.959	.871
WTCrew	.170	.470	.342	.899	.514	-.281	1.000	-.305	.495	-.323	-.011	.069	-.087	.102	.102	.203	.018
PHV^a	.343	.070	-.488	-.275	.131	.394	-.305	1.000	-.027	.827	.156	.151	-.227	-.296	-.296	-.286	.274
PHV_D	.562	.591	.257	.567	.696	.349	.495	-.027	1.000	.114	.587	.572	-.576	-.436	-.436	-.381	.554
SHV	.480	.142	-.587	-.314	.181	.571	-.323	.827	.114	1.000	.383	.353	-.340	-.503	-.503	-.523	.420
SHV_D	.566	.310	-.006	-.010	.373	.496	-.011	.156	.587	.383	1.000	.530	-.406	-.466	-.466	-.483	.506
MD	.770	.740	.079	.046	.766	.847	.069	.151	.572	.353	.530	1.000	-.845	-.901	-.901	-.850	.971
C2LR	-.570	-.723	-.116	-.126	-.752	-.669	-.087	-.227	-.576	-.340	-.406	-.845	1.000	.743	.743	.694	-.907
C3LR	-.782	-.605	.141	.144	-.628	-.959	.102	-.296	-.436	-.503	-.466	-.901	.743	1.000	1.000	.972	-.921
C4LR	-.782	-.605	.141	.144	-.628	-.959	.102	-.296	-.436	-.503	-.466	-.901	.743	1.000	1.000	.972	-.921
TCLR	-.723	-.484	.196	.213	-.526	-.959	.203	-.286	-.381	-.523	-.483	-.850	.694	.972	.972	1.000	-.871
LZFT	.750	.741	.041	.012	.777	.871	.018	.274	.554	.420	.506	.971	-.907	-.921	-.921	-.871	1.000
Dimension	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Eigenvalue	9.025	3.679	1.403	.911	.638	.421	.291	.168	.148	.122	.074	.055	.038	.016	.006	.005	.000

Table F. 2. PCA Results for the Simple scenario, Human-Only planning condition.

Correlations Transformed Variables

	TATT	TAAT	TCAT	TUAT	TAT	WTC	WTCrew	WTI	WTO	PHV	PHV_D	SHV	SHV_D	MD	UIC	UCD	UIT	UU	C2LR	C4LR	TCLR	LZFT
TATT	1.000	.933	.076	.626	.961	.981	.519	-.196	.011	.215	.646	.233	.422	.949	.008	-.263	-.278	-.387	-.951	-.949	-.917	.959
TAAT	.933	1.000	-.024	.651	.969	.939	.555	-.160	.055	.243	.602	.272	.422	.933	-.028	-.303	-.246	-.357	-.932	-.933	-.906	.977
TCAT	.076	-.024	1.000	-.030	.077	.047	.009	-.265	-.139	.288	-.037	-.037	-.193	-.004	-.377	.103	-.220	-.221	.002	.004	.058	-.006
TUAT	.626	.651	-.030	1.000	.702	.594	.956	.136	.327	-.287	.455	-.223	.339	.597	.340	-.174	.098	-.010	-.600	-.597	-.519	.631
TAT	.961	.969	.077	.702	1.000	.949	.613	-.183	.027	.186	.631	.178	.403	.948	-.020	-.314	-.273	-.378	-.950	-.948	-.915	.961
WTC	.981	.939	.047	.594	.949	1.000	.483	-.148	.047	.290	.695	.322	.497	.944	-.008	-.219	-.228	-.339	-.944	-.944	-.908	.972
WTCrew	.519	.555	.009	.956	.613	.483	1.000	.122	.345	-.287	.350	-.281	.245	.504	.313	-.187	.093	-.008	-.507	-.504	-.424	.521
WTI	-.196	-.160	-.265	.136	-.183	-.148	.122	1.000	.847	-.518	.026	-.193	.220	-.226	.741	.401	.970	.953	.232	.226	.273	-.154
WTO	.011	.055	-.139	.327	.027	.047	.345	.847	1.000	-.415	.036	-.134	.251	-.005	.782	.389	.861	.789	.010	.005	.076	.061
PHV	.215	.243	.288	-.287	.186	.290	-.287	-.518	-.415	1.000	.195	.795	.117	.230	-.663	.005	-.525	-.536	-.227	-.230	-.228	.260
PHV_D	.646	.602	-.037	.455	.631	.695	.350	.026	.036	.195	1.000	.281	.828	.612	.097	-.061	-.065	-.128	-.608	-.612	-.538	.656
SHV	.233	.272	-.037	-.223	.178	.322	-.281	-.193	-.134	.795	.281	1.000	.304	.247	-.253	-.033	-.217	-.244	-.240	-.247	-.209	.318
SHV_D	.422	.422	-.193	.339	.403	.497	.245	.220	.251	.117	.828	.304	1.000	.391	.339	.206	.203	.152	-.383	-.391	-.306	.464
MD	.949	.933	-.004	.597	.948	.944	.504	-.226	-.005	.230	.612	.247	.391	1.000	-.014	-.282	-.298	-.405	-1.000	-1.000	-.971	.959
UIC	.008	-.028	-.377	.340	-.020	-.008	.313	.741	.782	-.663	.097	-.253	.339	-.014	1.000	.270	.772	.728	.018	.014	.060	.010
UCD	-.263	-.303	.103	-.174	-.314	-.219	-.187	.401	.389	.005	-.061	-.033	.206	-.282	.270	1.000	.540	.544	.284	.282	.323	-.252
UIT	-.278	-.246	-.220	.098	-.273	-.228	.093	.970	.861	-.525	-.065	-.217	.203	-.298	.772	.540	1.000	.987	.304	.298	.357	-.229
UU	-.387	-.357	-.221	-.010	-.378	-.339	-.008	.953	.789	-.536	-.128	-.244	.152	-.405	.728	.544	.987	1.000	.411	.405	.455	-.342
C2LR	-.951	-.932	.002	-.600	-.950	-.944	-.507	.232	.010	-.227	-.608	-.240	-.383	-1.000	.018	.284	.304	.411	1.000	1.000	.972	-.958
C4LR	-.949	-.933	.004	-.597	-.948	-.944	-.504	.226	.005	-.230	-.612	-.247	-.391	-1.000	.014	.282	.298	.405	1.000	1.000	.971	-.959
TCLR	-.917	-.906	.058	-.519	-.915	-.908	-.424	.273	.076	-.228	-.538	-.209	-.306	-.971	.060	.323	.357	.455	.972	.971	1.000	-.924
LZFT	.959	.977	-.006	.631	.961	.972	.521	-.154	.061	.260	.656	.318	.464	.959	.010	-.252	-.229	-.342	-.958	-.959	-.924	1.000
Dimension	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
Eigenvalue	10.546	5.265	2.128	1.188	.938	.637	.480	.287	.133	.129	.085	.056	.043	.026	.023	.016	.011	.005	.003	.001	.000	.000

Table F. 3. PCA Results for the Simple scenario, Human-Algorithm planning condition.

Correlations Transformed Variables

	TATT	TAAT	TCAT	TUAT	TAT	WTQMS	WTC	WTCrew	WTI	PHV	PHV_D	SHV	SHV_D	MD	UCD	UIT	UU	C2L	C4L	C2LR	C4LR	TCLR	LZFT
TATT^{a}	1.000	.959	.735	-.026	.955	.806	.998	.159	-.498	.693	.208	.682	.221	.959	-.150	-.498	-.623	-.694	.813	-.806	.460	-.917	.788
TAAT^{a}	.959	1.000	.677	.151	.998	.765	.952	.288	-.495	.633	.055	.605	.072	1.000	-.187	-.495	-.600	-.662	.766	-.759	.414	-.964	.755
TCAT	.735	.677	1.000	-.050	.677	.685	.740	.172	-.287	.598	.122	.554	.107	.677	-.192	-.287	-.392	-.904	.694	-.690	.173	-.580	.627
TUAT	-.026	.151	-.050	1.000	.167	-.045	-.025	.924	.096	-.091	-.295	-.170	-.368	.151	-.438	.096	.099	-.084	-.053	.054	-.149	-.133	-.061
TAT^{a}	.955	.998	.677	.167	1.000	.765	.948	.303	-.478	.636	.051	.604	.068	.998	-.198	-.478	-.599	-.659	.766	-.759	.416	-.962	.757
WTQMS	.806	.765	.685	-.045	.765	1.000	.801	.140	-.555	.976	.212	.958	.223	.765	-.161	-.555	-.888	-.557	.999	-.999	.790	-.731	.990
WTC^{a}	.998	.952	.740	-.025	.948	.801	1.000	.169	-.481	.685	.192	.675	.202	.952	-.150	-.481	-.612	-.695	.809	-.802	.456	-.910	.779
WTCrew	.159	.288	.172	.924	.303	.140	.169	1.000	.025	.086	-.254	.006	-.319	.288	-.506	.025	-.038	-.238	.140	-.139	-.017	-.239	.104
WTI	-.498	-.495	-.287	.096	-.478	-.555	-.481	.025	1.000	-.511	-.220	-.540	-.248	-.495	.246	1.000	.750	.242	-.545	.545	-.459	.461	-.584
PHV	.693	.633	.598	-.091	.636	.976	.685	.086	-.511	1.000	.271	.990	.279	.633	-.165	-.511	-.896	-.442	.975	-.977	.849	-.611	.976
PHV_D	.208	.055	.122	-.295	.051	.212	.192	-.254	-.220	.271	1.000	.332	.959	.055	.086	-.220	-.224	-.106	.223	-.223	.218	-.051	.269
SHV	.682	.605	.554	-.170	.604	.958	.675	.006	-.540	.990	.332	1.000	.333	.605	-.136	-.540	-.908	-.377	.957	-.959	.877	-.590	.965
SHV_D	.221	.072	.107	-.368	.068	.223	.202	-.319	-.248	.279	.959	.333	1.000	.072	.119	-.248	-.241	-.111	.234	-.234	.228	-.075	.283
MD^{a}	.959	1.000	.677	.151	.998	.765	.952	.288	-.495	.633	.055	.605	.072	1.000	-.187	-.495	-.600	-.662	.766	-.759	.414	-.964	.755
UCD	-.150	-.187	-.192	-.438	-.198	-.161	-.150	-.506	.246	-.165	.086	-.136	.119	-.187	1.000	.246	.200	.139	-.167	.166	-.081	.140	-.149
UIT	-.498	-.495	-.287	.096	-.478	-.555	-.481	.025	1.000	-.511	-.220	-.540	-.248	-.495	.246	1.000	.750	.242	-.545	.545	-.459	.461	-.584
UU	-.623	-.600	-.392	.099	-.599	-.888	-.612	-.038	.750	-.896	-.224	-.908	-.241	-.600	.200	.750	1.000	.221	-.883	.885	-.893	.604	-.919
C2L^{a}	-.694	-.662	-.904	-.084	-.659	-.557	-.695	-.238	.242	-.442	-.106	-.377	-.111	-.662	.139	.242	.221	1.000	-.562	.557	.060	.549	-.489
C4L	.813	.766	.694	-.053	.766	.999	.809	.140	-.545	.975	.223	.957	.234	.766	-.167	-.545	-.883	-.562	1.000	-1.000	.790	-.731	.989
C2LR	-.806	-.759	-.690	.054	-.759	-.999	-.802	-.139	.545	-.977	-.223	-.959	-.234	-.759	.166	.545	.885	.557	-1.000	1.000	-.793	.724	-.989
C4LR^{a}	.460	.414	.173	-.149	.416	.790	.456	-.017	-.459	.849	.218	.877	.228	.414	-.081	-.459	-.893	.060	.790	-.793	1.000	-.454	.826
TCLR^{a}	-.917	-.964	-.580	-.133	-.962	-.731	-.910	-.239	.461	-.611	-.051	-.590	-.075	-.964	.140	.461	.604	.549	-.731	.724	-.454	1.000	-.735
LZFT	.788	.755	.627	-.061	.757	.990	.779	.104	-.584	.976	.269	.965	.283	.755	-.149	-.584	-.919	-.489	.989	-.989	.826	-.735	1.000
Dimension	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Eigenvalue	13.201	3.291	2.025	1.428	1.301	.918	.517	.111	.073	.045	.034	.024	.014	.009	.003	.002	.001	.001	.000	.000	.000	.000	.000

Table F. 4. Cross-Correlations for the Simple Scenario.

Simple Scenario																
	TATT	TAAT	TUAT	TAT	WTC	WTCrew	PHV	SHV	MD	C2LR	C4LR	TCLR	LZFT	WTI	UIT	UU
TATT	X															
TAAT		X														
TUAT			X													
TAT		X		X												
WTC	X				X											
WTCrew			X			X										
PHV[a]							X									
SHV							X	X								
MD	X	X		X	X				X							
C2LR		X		X					X	X						
C4LR										X	X					
TCLR	X				X				X	X		X				
LZFT	X	X		X	X				X	X	X	X	X			
WTI														X		
UIT														X	X	
UU														X	X	X

In this table, X’s denote that these metrics were highly correlated for all the planning conditions (Table F. 1 – Table F. 3). For example, TATT and MD were found to be highly correlated with the B, HO, and HA planning conditions within the Simple scenario. TATT and TAAT, however, were not highly correlated in at least one of these three planning conditions.

Table F. 5. PCA Results for the Moderate scenario, Baseline planning condition.

Correlations Transformed Variables

	TATT	TAAT	TCAT	TUAT	TAT	WTQMS	WTCrew	PHV	PHV_D	SHV	SHV_D	MD	LZFT	FMAC_6_EART	FMAC_6_HER	FMAC_6_AAT	SMAC_2_EART	SMAC_2_EFR	SMAC_2_AAT
TATT	1.000	.921	.640	.357	.963	.408	.734	.478	.595	.009	.552	.260	.938	.220	-.353	.567	.311	-.461	.487
TAAT	.921	1.000	.603	.577	.945	.634	.583	.396	.634	-.150	.542	.157	.892	.463	-.311	.513	.528	-.360	.349
TCAT	.640	.603	1.000	.269	.696	.324	.629	.259	.258	.041	.289	.246	.555	.310	-.097	.105	.222	-.412	.473
TUAT	.357	.577	.269	1.000	.487	.634	.154	.126	.478	-.324	.418	-.141	.337	.587	-.053	.172	.567	-.012	-.009
TAT	.963	.945	.696	.487	1.000	.516	.731	.429	.593	-.085	.533	.219	.907	.342	-.337	.521	.408	-.449	.485
WTQMS	.408	.634	.324	.634	.516	1.000	.085	-.114	.435	-.513	.174	-.066	.318	.894	-.104	.219	.938	-.134	.043
WTCrew	.734	.583	.629	.154	.731	.085	1.000	.238	.215	.184	.356	.522	.616	-.096	-.140	.307	-.099	-.351	.445
PHV	.478	.396	.259	.126	.429	-.114	.238	1.000	.414	.132	.494	-.096	.600	-.135	-.241	.215	-.069	-.285	.302
PHV_D	.595	.634	.258	.478	.593	.435	.215	.414	1.000	-.356	.761	-.263	.635	.318	-.434	.464	.348	-.212	.144
SHV	.009	-.150	.041	-.324	-.085	-.513	.184	.132	-.356	1.000	.025	.347	.048	-.433	.079	-.095	-.478	-.113	.186
SHV_D	.552	.542	.289	.418	.533	.174	.356	.494	.761	.025	1.000	-.162	.603	.088	-.256	.292	.106	-.277	.318
MD	.260	.157	.246	-.141	.219	-.066	.522	-.096	-.263	.347	-.162	1.000	.105	-.067	.351	-.116	-.168	-.139	.193
LZFT	.938	.892	.555	.337	.907	.318	.616	.600	.635	.048	.603	.105	1.000	.138	-.369	.551	.235	-.447	.450
FMAC_6_EART	.220	.463	.310	.587	.342	.894	-.096	-.135	.318	-.433	.088	-.067	.138	1.000	.027	.091	.884	-.083	.011
FMAC_6_HER	-.353	-.311	-.097	-.053	-.337	-.104	-.140	-.241	-.434	.079	-.256	.351	-.369	.027	1.000	-.832	-.108	.121	-.193
FMAC_6_AAT	.567	.513	.105	.172	.521	.219	.307	.215	.464	-.095	.292	-.116	.551	.091	-.832	1.000	.213	-.104	.192
SMAC_2_EART	.311	.528	.222	.567	.408	.938	-.099	-.069	.348	-.478	.106	-.168	.235	.884	-.108	.213	1.000	-.035	-.032
SMAC_2_EFR	-.461	-.360	-.412	-.012	-.449	-.134	-.351	-.285	-.212	-.113	-.277	-.139	-.447	-.083	.121	-.104	-.035	1.000	-.910
SMAC_2_AAT	.487	.349	.473	-.009	.485	.043	.445	.302	.144	.186	.318	.193	.450	.011	-.193	.192	-.032	-.910	1.000
Dimension	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
Eigenvalue	7.617	3.548	2.187	1.301	1.249	.662	.600	.541	.400	.289	.189	.137	.100	.061	.047	.030	.020	.011	.010

Table F. 6. PCA Results for the Moderate scenario, Human-Only planning condition.

Correlations Transformed Variables

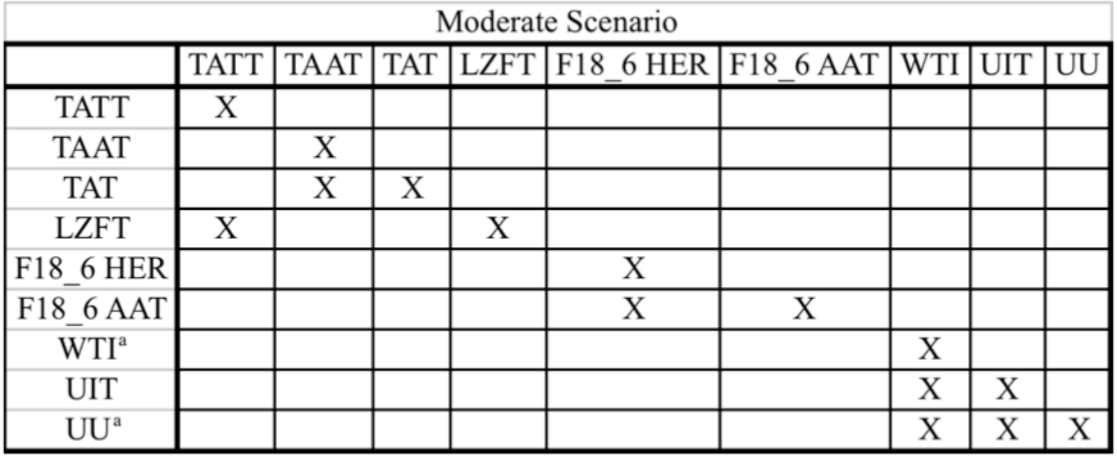
	TATT	TAAT	TCAT	TUAT	TAT	WTQMS	WTCrew	WTI	WTO	PHV	PHV_D	SHV	SHV_D	MD	UIC	UCD	UIT	UU	LZFT	FMAC_6_EART	FMAC_6_HER	FMAC_6_AAT	SMAC_2_EFR	SMAC_2_AAT
TATT	1.000	.202	.704	-.116	.202	.201	.965	-.201	-.465	-.129	.559	.255	.487	.233	.022	.534	-.096	-.334	.954	.247	-.629	.550	-.302	.302
TAAT	.202	1.000	.408	.403	1.000	.999	.302	-.376	-.143	.100	-.152	-.250	-.292	.998	-.201	-.118	-.366	-.454	.218	.997	-.608	.762	-.161	.158
TCAT	.704	.408	1.000	-.085	.408	.409	.746	-.420	-.534	.071	-.010	.029	.143	.424	.268	.165	-.287	-.531	.724	.460	-.853	.789	-.237	.243
TUAT	-.116	.403	-.085	1.000	.403	.404	.047	-.081	.161	.067	-.103	-.218	-.375	.400	-.255	-.142	-.136	-.047	-.152	.373	-.054	.165	.060	-.074
TAT	.202	1.000	.408	.403	1.000	.999	.302	-.376	-.143	.100	-.152	-.250	-.292	.998	-.201	-.118	-.366	-.454	.218	.997	-.608	.762	-.161	.158
WTQMS	.201	.999	.409	.404	.999	1.000	.303	-.372	-.137	.103	-.152	-.255	-.291	.998	-.196	-.124	-.357	-.450	.217	.997	-.607	.761	-.167	.163
WTCrew	.965	.302	.746	.047	.302	.303	1.000	-.236	-.393	-.034	.478	.204	.447	.336	.016	.507	-.112	-.338	.914	.345	-.693	.645	-.257	.253
WTI[a]	-.201	-.376	-.420	-.081	-.376	-.372	-.236	1.000	.599	-.184	.224	-.202	.113	-.375	-.194	-.150	.663	.866	-.249	-.393	.449	-.445	.184	-.186
WTO[a]	-.465	-.143	-.534	.161	-.143	-.137	-.393	.599	1.000	.092	-.239	-.285	-.032	-.139	-.189	-.353	.656	.796	-.489	-.165	.477	-.278	.276	-.294
PHV[a]	-.129	.100	.071	.067	.100	.103	-.034	-.184	.092	1.000	-.457	.102	-.039	.112	-.267	.069	-.016	-.119	-.127	.101	-.161	.154	.170	-.188
PHV_D[a]	.559	-.152	-.010	-.103	-.152	-.152	.478	.224	-.239	-.457	1.000	.156	.463	-.138	-.186	.313	-.005	.049	.416	-.135	-.055	-.028	-.241	.254
SHV[a]	.255	-.250	.029	-.218	-.250	-.255	.204	-.202	-.285	.102	.156	1.000	.608	-.243	-.129	.529	-.045	-.141	.303	-.252	.025	-.113	.184	-.181
SHV_D	.487	-.292	.143	-.375	-.292	-.291	.447	.113	-.032	-.039	.463	.608	1.000	-.263	-.124	.474	.287	.103	.481	-.272	-.042	-.003	-.055	.062
MD	.233	.998	.424	.400	.998	.998	.336	-.375	-.139	.112	-.138	-.243	-.263	1.000	-.198	-.088	-.359	-.456	.246	.996	-.616	.771	-.174	.171
UIC	.022	-.201	.268	-.255	-.201	-.196	.016	-.194	-.189	-.267	-.186	-.129	-.124	-.198	1.000	-.051	-.227	-.168	.073	-.169	-.131	-.006	.032	-.042
UCD	.534	-.118	.165	-.142	-.118	-.124	.507	-.150	-.353	.069	.313	.529	.474	-.088	-.051	1.000	-.080	-.209	.570	-.117	-.132	-.054	-.080	.089
UIT	-.096	-.366	-.287	-.136	-.366	-.357	-.112	.663	.656	-.016	-.005	-.045	.287	-.359	-.227	-.080	1.000	.779	-.137	-.370	.404	-.307	.064	-.073
UU[a]	-.334	-.454	-.531	-.047	-.454	-.450	-.338	.866	.796	-.119	.049	-.141	.103	-.456	-.168	-.209	.779	1.000	-.379	-.474	.559	-.512	.237	-.241
LZFT	.954	.218	.724	-.152	.218	.217	.914	-.249	-.489	-.127	.416	.303	.481	.246	.073	.570	-.137	-.379	1.000	.259	-.644	.546	-.304	.306
FMAC_6_EART	.247	.997	.460	.373	.997	.997	.345	-.393	-.165	.101	-.135	-.252	-.272	.996	-.169	-.117	-.370	-.474	.259	1.000	-.654	.802	-.172	.168
FMAC_6_HER	-.629	-.608	-.853	-.054	-.608	-.607	-.693	.449	.477	-.161	-.055	.025	-.042	-.616	-.131	-.132	.404	.559	-.644	-.654	1.000	-.913	.282	-.285
FMAC_6_AAT	.550	.762	.789	.165	.762	.761	.645	-.445	-.278	.154	-.028	-.113	-.003	.771	-.006	-.054	-.307	-.512	.546	.802	-.913	1.000	-.219	.214
SMAC_2_EFR	-.302	-.161	-.237	.060	-.161	-.167	-.257	.184	.276	.170	-.241	.184	-.055	-.174	.032	-.080	.064	.237	-.304	-.172	.282	-.219	1.000	-.996
SMAC_2_AAT	.302	.158	.243	-.074	.158	.163	.253	-.186	-.294	-.188	.254	-.181	.062	.171	-.042	.089	-.073	-.241	.306	.168	-.285	.214	-.996	1.000
Dimension	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
Eigenvalue	8.658	4.786	2.503	2.086	1.500	1.257	.794	.625	.525	.358	.285	.178	.153	.136	.072	.050	.017	.010	.005	.003	.001	.001	.000	.000

Table F. 7. PCA Results for the Moderate scenario, Human-Algorithm planning condition.

Correlations Transformed Variables

	TATT	TAAT	TCAT	TUAT	TAT	WTQMS	WTCrew	WTI	PHV	PHV_D	SHV	SHV_D	MD	UCD	UIT	UU	LZFT	FMAC_6_EART	FMAC_6_HER	FMAC_6_AAT	SMAC_2_EART	SMAC_2_EFR	SMAC_2_AAT
TATT	1.000	.097	.734	-.079	.102	.098	.577	-.302	.265	.103	.419	.347	.300	-.232	-.302	-.332	.961	.309	-.322	.345	.311	-.325	-.084
TAAT^a	.097	1.000	.092	.069	1.000	1.000	-.058	-.055	.088	-.551	.154	-.294	.677	-.158	-.055	-.074	-.037	-.034	.031	-.021	-.033	.039	-.998
TCAT	.734	.092	1.000	-.276	.098	.092	.564	-.266	.022	-.268	.306	.142	.304	-.193	-.266	-.343	.701	.307	-.313	.339	.314	-.317	-.090
TUAT	-.079	.069	-.276	1.000	.067	.070	-.404	.136	-.005	.007	-.097	-.067	-.518	-.180	.136	.163	-.021	-.675	.678	-.685	-.667	.664	-.064
TAT^a	.102	1.000	.098	.067	1.000	1.000	-.055	-.056	.086	-.552	.154	-.297	.679	-.152	-.056	-.079	-.031	-.033	.030	-.020	-.032	.038	-.998
WTQMS^a	.098	1.000	.092	.070	1.000	1.000	-.057	-.055	.088	-.551	.155	-.293	.677	-.158	-.055	-.074	-.037	-.034	.031	-.021	-.033	.039	-.998
WTCrew	.577	-.058	.564	-.404	-.055	-.057	1.000	-.188	.303	.123	.280	.232	.503	-.080	-.188	-.181	.506	.676	-.678	.676	.687	-.708	.075
WTI	-.302	-.055	-.266	.136	-.056	-.055	-.188	1.000	-.114	-.037	-.163	.261	-.151	-.096	1.000	.845	-.260	-.098	.101	-.087	-.103	.099	.060
PHV	.265	.088	.022	-.005	.086	.088	.303	-.114	1.000	.349	-.032	.349	.242	-.160	-.114	.014	.223	.157	-.149	.176	.159	-.166	-.082
PHV_D	.103	-.551	-.268	.007	-.552	-.551	.123	-.037	.349	1.000	.039	.425	-.320	.076	-.037	.095	.097	.067	-.060	.054	.072	-.078	.562
SHV	.419	.154	.306	-.097	.154	.155	.280	-.163	-.032	.039	1.000	.426	.180	-.157	-.163	-.147	.324	.163	-.190	.181	.170	-.166	-.152
SHV_D	.347	-.294	.142	-.067	-.297	-.293	.232	.261	.349	.425	.426	1.000	-.125	-.057	.261	.323	.377	.119	-.122	.143	.124	-.132	.311
MD	.300	.677	.304	-.518	.679	.677	.503	-.151	.242	-.320	.180	-.125	1.000	-.028	-.151	-.157	.122	.677	-.679	.687	.678	-.677	-.667
UCD	-.232	-.158	-.193	-.180	-.152	-.158	-.080	-.096	-.160	.076	-.157	-.057	-.028	1.000	-.096	.086	-.177	.094	-.064	.036	.097	-.099	.164
UIT	-.302	-.055	-.266	.136	-.056	-.055	-.188	1.000	-.114	-.037	-.163	.261	-.151	-.096	1.000	.845	-.260	-.098	.101	-.087	-.103	.099	.060
UU	-.332	-.074	-.343	.163	-.079	-.074	-.181	.845	.014	.095	-.147	.323	-.157	.086	.845	1.000	-.330	-.074	.086	-.082	-.078	.070	.078
LZFT	.961	-.037	.701	-.021	-.031	-.037	.506	-.260	.223	.097	.324	.377	.122	-.177	-.260	-.330	1.000	.178	-.190	.214	.179	-.196	.050
FMAC_6_EART^a	.309	-.034	.307	-.675	-.033	-.034	.676	-.098	.157	.067	.163	.119	.677	.094	-.098	-.074	.178	1.000	-.998	.995	.999	-.998	.046
FMAC_6_HER^a	-.322	.031	-.313	.678	.030	.031	-.678	.101	-.149	-.060	-.190	-.122	-.679	-.064	.101	.086	-.190	-.998	1.000	-.997	-.996	.995	-.042
FMAC_6_AAT^a	.345	-.021	.339	-.685	-.020	-.021	.676	-.087	.176	.054	.181	.143	.687	.036	-.087	-.082	.214	.995	-.997	1.000	.992	-.991	.031
SMAC_2_EART^a	.311	-.033	.314	-.667	-.032	-.033	.687	-.103	.159	.072	.170	.124	.678	.097	-.103	-.078	.179	.999	-.996	.992	1.000	-.998	.046
SMAC_2_EFR^a	-.325	.039	-.317	.664	.038	.039	-.708	.099	-.166	-.078	-.166	-.132	-.677	-.099	.099	.070	-.196	-.998	.995	-.991	-.998	1.000	-.053
SMAC_2_AAT^a	-.084	-.998	-.090	-.064	-.998	-.998	.075	.060	-.082	.562	-.152	.311	-.667	.164	.060	.078	.050	.046	-.042	.031	.046	-.053	1.000
Dimension	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Eigenvalue	7.305	5.033	3.248	2.447	1.377	1.003	.843	.544	.390	.323	.188	.158	.118	.011	.005	.005	.002	.001	.001	.000	.000	.000	.000

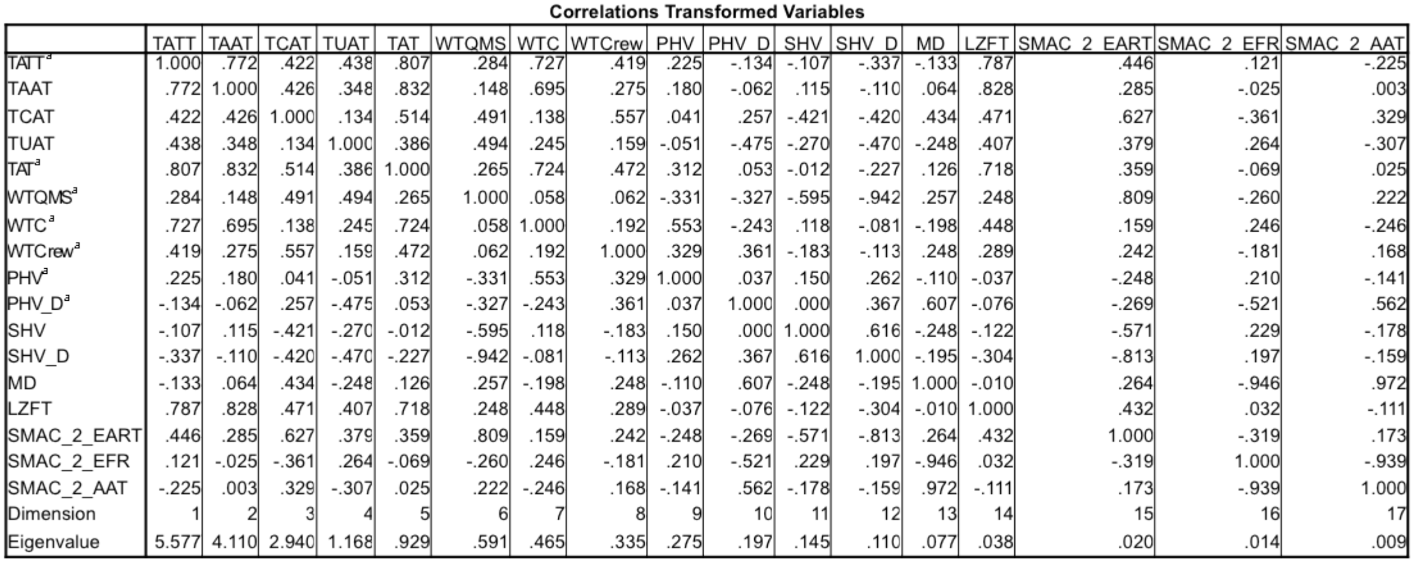
Table F. 8. Cross-Correlations for the Moderate Scenario.



Moderate Scenario									
	TATT	TAAT	TAT	LZFT	F18_6 HER	F18_6 AAT	WTI	UIT	UU
TATT	X								
TAAT		X							
TAT		X	X						
LZFT	X			X					
F18_6 HER					X				
F18_6 AAT					X	X			
WTI[a]							X		
UIT							X	X	
UU[a]							X	X	X

In this table, X’s denote that these metrics were highly correlated for all the planning conditions (Table F. 5 – Table F. 7). For example, TATT and LZFT were shown to be highly correlated for the B, HO, and HA planning conditions within the Simple scenario. TATT and TAAT, however, were not highly correlated in at least one of these three planning conditions.

Table F. 9. PCA Results for the Complex scenario, Baseline planning condition.



Correlations Transformed Variables

	TATT	TAAT	TCAT	TUAT	TAT	WTQMS	WTC	WTCrew	PHV	PHV_D	SHV	SHV_D	MD	LZFT	SMAC_2_EART	SMAC_2_EFR	SMAC_2_AAT
TATT[a]	1.000	.772	.422	.438	.807	.284	.727	.419	.225	-.134	-.107	-.337	-.133	.787	.446	.121	-.225
TAAT	.772	1.000	.426	.348	.832	.148	.695	.275	.180	-.062	.115	-.110	.064	.828	.285	-.025	.003
TCAT	.422	.426	1.000	.134	.514	.491	.138	.557	.041	.257	-.421	-.420	.434	.471	.627	-.361	.329
TUAT	.438	.348	.134	1.000	.386	.494	.245	.159	-.051	-.475	-.270	-.470	-.248	.407	.379	.264	-.307
TAT[a]	.807	.832	.514	.386	1.000	.265	.724	.472	.312	.053	-.012	-.227	.126	.718	.359	-.069	.025
WTQMS[a]	.284	.148	.491	.494	.265	1.000	.058	.062	-.331	-.327	-.595	-.942	.257	.248	.809	-.260	.222
WTC[a]	.727	.695	.138	.245	.724	.058	1.000	.192	.553	-.243	.118	-.081	-.198	.448	.159	.246	-.246
WTCrew[a]	.419	.275	.557	.159	.472	.062	.192	1.000	.329	.361	-.183	-.113	.248	.289	.242	-.181	.168
PHV[a]	.225	.180	.041	-.051	.312	-.331	.553	.329	1.000	.037	.150	.262	-.110	-.037	-.248	.210	-.141
PHV_D[a]	-.134	-.062	.257	-.475	.053	-.327	-.243	.361	.037	1.000	.000	.367	.607	-.076	-.269	-.521	.562
SHV	-.107	.115	-.421	-.270	-.012	-.595	.118	-.183	.150	.000	1.000	.616	-.248	-.122	-.571	.229	-.178
SHV_D	-.337	-.110	-.420	-.470	-.227	-.942	-.081	-.113	.262	.367	.616	1.000	-.195	-.304	-.813	.197	-.159
MD	-.133	.064	.434	-.248	.126	.257	-.198	.248	-.110	.607	-.248	-.195	1.000	-.010	.264	-.946	.972
LZFT	.787	.828	.471	.407	.718	.248	.448	.289	-.037	-.076	-.122	-.304	-.010	1.000	.432	.032	-.111
SMAC_2_EART	.446	.285	.627	.379	.359	.809	.159	.242	-.248	-.269	-.571	-.813	.264	.432	1.000	-.319	.173
SMAC_2_EFR	.121	-.025	-.361	.264	-.069	-.260	.246	-.181	.210	-.521	.229	.197	-.946	.032	-.319	1.000	-.939
SMAC_2_AAT	-.225	.003	.329	-.307	.025	.222	-.246	.168	-.141	.562	-.178	-.159	.972	-.111	.173	-.939	1.000
Dimension	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Eigenvalue	5.577	4.110	2.940	1.168	.929	.591	.465	.335	.275	.197	.145	.110	.077	.038	.020	.014	.009

Table F. 10. PCA Results for the Complex scenario, Human-Only planning condition.

Correlations Transformed Variables

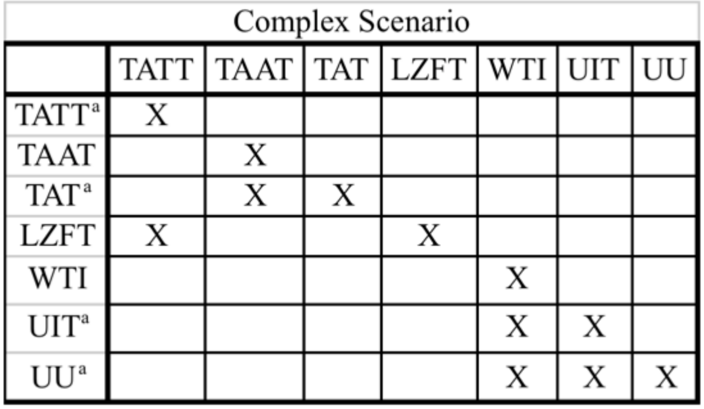
	TATT	TAAT	TCAT	TUAT	TAT	WTQMS	WTC	WTCrew	WTI	WTO	PHV	PHV_D	SHV	SHV_D	MD	UIC	UCD	UIT	UU	LZFT	SMAC_2_EART	SMAC_2_EFR	SMAC_2_AAT
TATT	1.000	.885	.361	.455	.932	.149	.430	.400	-.209	.272	.046	.006	.193	.026	.402	-.081	-.230	-.250	-.335	.892	-.157	-.138	.079
TAAT[a]	.885	1.000	.215	.451	.961	.100	.391	.197	-.316	.095	.046	-.106	.253	.000	.379	-.102	-.134	-.335	-.399	.779	-.150	.024	-.031
TCAT	.361	.215	1.000	.155	.335	.353	.163	.522	.245	.205	.340	.178	-.030	-.195	-.003	.058	-.392	.316	.171	.264	.088	-.057	.082
TUAT	.455	.451	.155	1.000	.402	-.142	.365	.181	-.206	.045	.007	-.040	.076	.341	.392	.331	-.281	-.229	-.159	.538	-.259	.125	-.099
TAT[a]	.932	.961	.335	.402	1.000	.128	.453	.301	-.295	.184	.060	-.067	.249	-.042	.391	-.149	-.213	-.331	-.418	.839	-.086	-.069	.046
WTQMS[a]	.149	.100	.353	-.142	.128	1.000	.064	.184	.456	-.285	-.068	.355	-.150	-.473	.220	-.349	-.107	.494	.469	.031	.348	-.301	.265
WTC	.430	.391	.163	.365	.453	.064	1.000	-.036	-.195	.079	.161	-.314	.152	-.028	.167	.096	-.189	-.288	-.256	.614	.118	-.057	.129
WTCrew	.400	.197	.522	.181	.301	.184	-.036	1.000	.081	.427	.008	.349	-.149	.112	.200	-.162	-.335	.069	-.004	.279	.196	-.236	.182
WTI	-.209	-.316	.245	-.206	-.295	.456	-.195	.081	1.000	-.296	-.097	.443	-.197	-.260	.130	.061	-.343	.932	.920	-.198	.231	-.260	.361
WTO[a]	.272	.095	.205	.045	.184	-.285	.079	.427	-.296	1.000	.218	-.167	.044	.169	-.221	.023	-.217	-.336	-.339	.192	.267	-.135	.113
PHV[a]	.046	.046	.340	.007	.060	-.068	.161	.008	-.097	.218	1.000	-.121	.249	-.253	-.124	.203	-.221	.025	-.030	-.038	.009	-.001	.026
PHV_D[a]	.006	-.106	.178	-.040	-.067	.355	-.314	.349	.443	-.167	-.121	1.000	-.423	.015	.502	-.040	-.114	.466	.444	.009	.100	-.314	.177
SHV	.193	.253	-.030	.076	.249	-.150	.152	-.149	-.197	.044	.249	-.423	1.000	-.078	-.170	-.210	-.013	-.107	-.076	.144	-.053	.218	-.145
SHV_D	.026	.000	-.195	.341	-.042	-.473	-.028	.112	-.260	.169	-.253	.015	-.078	1.000	.105	.248	.137	-.322	-.270	.081	-.149	.291	-.264
MD	.402	.379	-.003	.392	.391	.220	.167	.200	.130	-.221	-.124	.502	-.170	.105	1.000	.073	-.227	.047	.052	.418	-.043	-.169	.074
UIC	-.081	-.102	.058	.331	-.149	-.349	.096	-.162	.061	.023	.203	-.040	-.210	.248	.073	1.000	-.123	.038	.014	-.025	-.396	.441	-.426
UCD[a]	-.230	-.134	-.392	-.281	-.213	-.107	-.189	-.335	-.343	-.217	-.221	-.114	-.013	.137	-.227	-.123	1.000	-.273	-.276	-.293	-.325	.101	-.124
UIT[a]	-.250	-.335	.316	-.229	-.331	.494	-.288	.069	.932	-.336	.025	.466	-.107	-.322	.047	.038	-.273	1.000	.962	-.317	.188	-.220	.286
UU[a]	-.335	-.399	.171	-.159	-.418	.469	-.256	-.004	.920	-.339	-.030	.444	-.076	-.270	.052	.014	-.276	.962	1.000	-.337	.249	-.227	.304
LZFT	.892	.779	.264	.538	.839	.031	.614	.279	-.198	.192	-.038	.009	.144	.081	.418	-.025	-.293	-.317	-.337	1.000	-.123	-.176	.126
SMAC_2_EART[a]	-.157	-.150	.088	-.259	-.086	.348	.118	.196	.231	.267	.009	.100	-.053	-.149	-.043	-.396	-.325	.188	.249	-.123	1.000	-.152	.259
SMAC_2_EFR	-.138	.024	-.057	.125	-.069	-.301	-.057	-.236	-.260	-.135	-.001	-.314	.218	.291	-.169	.441	.101	-.220	-.227	-.176	-.152	1.000	-.905
SMAC_2_AAT	.079	-.031	.082	-.099	.046	.265	.129	.182	.361	.113	.026	.177	-.145	-.264	.074	-.426	-.124	.286	.304	.126	.259	-.905	1.000
Dimension	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Eigenvalue	5.491	4.441	2.319	2.000	1.821	1.289	1.118	.956	.785	.608	.507	.366	.333	.274	.207	.197	.102	.074	.054	.031	.018	.007	.004

Table F. 11. PCA Results for the Complex scenario, Human-Algorithm planning condition.

Correlations Transformed Variables

	TATT	TAAT	TCAT	TUAT	TAT	WTQMS	WTC	WTCrew	WTI	WTO	PHV	PHV_D	SHV	SHV_D	MD	UIC	UCD	UIT	UU	LZFT	SMAC_2_EART	SMAC_2_EFR	SMAC_2_AAT
TATT	1.000	.028	.672	.197	.204	-.123	.097	.683	-.139	-.139	-.064	.601	.490	.334	.002	-.044	-.039	-.139	-.153	.724	-.143	.118	-.168
TAAT	.028	1.000	-.126	-.202	.813	.849	.089	-.164	.319	.319	.126	.045	.394	.317	.716	.410	.412	.319	.098	.256	.761	-.720	.364
TCAT	.672	-.126	1.000	.057	.118	-.190	.003	.548	-.020	-.020	.194	.565	.347	.414	-.021	-.310	-.253	-.020	-.073	.444	-.162	.124	-.095
TUAT	.197	-.202	.057	1.000	-.028	-.138	-.189	.404	-.270	-.270	.151	.058	.155	-.080	.035	.098	.130	-.270	-.301	.249	-.115	.162	-.057
TAT	.204	.813	.118	-.028	1.000	.697	.239	-.059	.429	.429	.179	.084	.422	.415	.691	.291	.301	.429	.183	.366	.786	-.784	.220
WTQMS	-.123	.849	-.190	-.138	.697	1.000	-.013	-.278	.305	.305	.111	-.091	.277	.044	.832	.431	.456	.305	.123	.214	.648	-.629	.654
WTC^a	.097	.089	.003	-.189	.239	-.013	1.000	-.077	.191	.191	-.293	-.374	-.017	.078	.093	-.144	-.225	.191	.282	.078	.223	-.290	.040
WTCrew	.683	-.164	.548	.404	-.059	-.278	-.077	1.000	-.370	-.370	.105	.426	.485	.212	-.235	-.070	.004	-.370	-.377	.526	-.323	.341	-.282
WTI^a	-.139	.319	-.020	-.270	.429	.305	.191	-.370	1.000	1.000	-.200	-.173	-.254	.083	.275	.091	.090	1.000	.860	-.191	.495	-.535	.016
WTO^a	-.139	.319	-.020	-.270	.429	.305	.191	-.370	1.000	1.000	-.200	-.173	-.254	.083	.275	.091	.090	1.000	.860	-.191	.495	-.535	.016
PHV	-.064	.126	.194	.151	.179	.111	-.293	.105	-.200	-.200	1.000	.502	.185	.199	.097	.325	.333	-.200	-.281	.092	.246	-.226	.094
PHV_D^a	.601	.045	.565	.058	.084	-.091	-.374	.426	-.173	-.173	.502	1.000	.272	.345	-.038	.237	.250	-.173	-.252	.426	.011	.004	-.038
SHV	.490	.394	.347	.155	.422	.277	-.017	.485	-.254	-.254	.185	.272	1.000	.445	.200	.107	.148	-.254	-.445	.662	.096	-.075	-.046
SHV_D	.334	.317	.414	-.080	.415	.044	.078	.212	.083	.083	.199	.345	.445	1.000	.069	.154	.155	.083	.036	.216	.289	-.244	-.132
MD	.002	.716	-.021	.035	.691	.832	.093	-.235	.275	.275	.097	-.038	.200	.069	1.000	.338	.368	.275	.040	.201	.693	-.661	.688
UIC	-.044	.410	-.310	.098	.291	.431	-.144	-.070	.091	.091	.325	.237	.107	.154	.338	1.000	.969	.091	.011	.170	.280	-.208	.219
UCD	-.039	.412	-.253	.130	.301	.456	-.225	.004	.090	.090	.333	.250	.148	.155	.368	.969	1.000	.090	-.037	.157	.275	-.198	.245
UIT^a	-.139	.319	-.020	-.270	.429	.305	.191	-.370	1.000	1.000	-.200	-.173	-.254	.083	.275	.091	.090	1.000	.860	-.191	.495	-.535	.016
UU^a	-.153	.098	-.073	-.301	.183	.123	.282	-.377	.860	.860	-.281	-.252	-.445	.036	.040	.011	-.037	.860	1.000	-.239	.274	-.338	.053
LZFT	.724	.256	.444	.249	.366	.214	.078	.526	-.191	-.191	.092	.426	.662	.216	.201	.170	.157	-.191	-.239	1.000	-.028	.013	.082
SMAC_2_EART^a	-.143	.761	-.162	-.115	.786	.648	.223	-.323	.495	.495	.246	.011	.096	.289	.693	.280	.275	.495	.274	-.028	1.000	-.979	.332
SMAC_2_EFR^a	.118	-.720	.124	.162	-.784	-.629	-.290	.341	-.535	-.535	-.226	.004	-.075	-.244	-.661	-.208	-.198	-.535	-.338	.013	-.979	1.000	-.362
SMAC_2_AAT^a	-.168	.364	-.095	-.057	.220	.654	.040	-.282	.016	.016	.094	-.038	-.046	-.132	.688	.219	.245	.016	.053	.082	.332	-.362	1.000
Dimension	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Eigenvalue	6.819	4.774	2.966	1.886	1.348	1.155	.987	.808	.563	.465	.338	.283	.198	.110	.095	.079	.046	.037	.027	.012	.003	.000	.000

Table F. 12. Cross-correlations for the Complex scenario.



Complex Scenario							
	TATT	TAAT	TAT	LZFT	WTI	UIT	UU
TATT[a]	X						
TAAT		X					
TAT[a]		X	X				
LZFT	X			X			
WTI					X		
UIT[a]					X	X	
UU[a]					X	X	X

In this table, X's denote that these metrics were highly correlated for all the planning conditions (Table F. 9 – Table F. 11). For example, TATT and LZFT were shown to be highly correlated for the B, HO, and HA planning conditions within the Simple scenario. TATT and TAAT, however, were not highly correlated in at least one of these three planning conditions.

Appendix G – Normality Tests

Table G. 1. Results of Kolmogorov-Smirnov normality tests for the Simple scenario (asterisks * denote that data was non-normal).

Metric	Treatment	Statistic	df	Sig.
LZFT	B	0.115	30	0.200
	HO	0.152	30	0.073
	HA*	0.385	28	0.000
TATT	B	0.112	30	0.200
	HO*	0.200	30	0.004
	HA*	0.312	28	0.000
TAAT	B	0.137	30	0.159
	HO	0.104	30	0.200
	HA*	0.176	28	0.026
TCAT	B	0.151	30	0.080
	HO	0.145	30	0.107
	HA	0.092	28	0.200
MD	B	0.121	30	0.200
	HO*	0.193	30	0.006
	HA	0.153	28	0.091
C2LR	B	0.127	30	0.200
	HO*	0.182	30	0.013
	HA*	0.528	28	0.000

Metric	Treatment	Statistic	df	Sig.
C3LR	B	0.127	30	0.200
	HO*	0.182	30	0.013
	HA*	0.278	28	0.000
C4LR	B	0.127	30	0.200
	HO*	0.182	30	0.013
	HA*	0.358	28	0.000
TCLR	B	0.127	30	0.200
	HO*	0.182	30	0.013
	HA*	0.278	28	0.000
HV	B	0.123	30	0.200
	HO	0.099	30	0.200
	HA*	0.180	28	0.020
HV-D	B*	0.269	30	0.000
	HO	0.109	30	0.200
	HA*	0.292	28	0.000
UIT	B	-	-	-
	HO*	0.167	30	0.032
	HA*	0.196	28	0.007

Table G. 2. Results of Kolmogorov-Smirnov normality tests for the Moderate scenario (asterisks * denote that data was non-normal).

Metric	Treatment	Statistic	df	Sig.
LZFT	B	0.113	30	0.200
	HO	0.156	29	0.070
	HA	0.137	28	0.192
FMAC_6_HFR	B*	0.457	30	0.000
	HO*	0.369	29	0.000
	HA*	0.536	28	0.000
FMAC_6_EART	B*	0.342	30	0.000
	HO*	0.304	29	0.000
	HA*	0.223	28	0.001
SMAC_2_EART	B*	0.256	30	0.000
	HO	-	-	-
	HA	0.134	28	0.200
SMAC_2_EFR	B	0.137	30	0.155
	HO	0.130	29	0.200
	HA	0.120	28	0.200
SMAC_2_AAT	B	0.102	30	0.200
	HO	0.121	29	0.200
	HA	0.092	28	0.200
TATT	B	0.098	30	0.200
	HO	0.133	29	0.200
	HA	0.107	28	0.2.

Metric	Treatment	Statistic	df	Sig.
TAAT	B	0.085	30	0.200
	HO*	0.275	29	0.000
	HA	0.089	28	0.200
TCAT	B	0.090	30	0.200
	HO	0.089	29	0.200
	HA	0.103	28	0.200
WTQMS	B*	0.167	30	0.032
	HO*	0.367	29	0.000
	HA	0.088	28	0.200
MD	B	0.063	30	0.200
	HO*	0.183	29	0.014
	HA	0.105	28	0.200
HV	B*	0.195	30	0.005
	HO	0.106	29	0.200
	HA	0.096	28	0.200
HV-D	B*	0.163	30	0.040
	HO	0.144	29	0.131
	HA	0.127	28	0.200
UIT	B	-	-	-
	HO*	0.222	29	0.001
	HA	0.153	28	0.090

Table G. 3. Results of Kolmogorov-Smirnov normality tests for the Complex scenario (asterisks * denote that data was non-normal).

Metric	Treatment	Statistic	df	Sig.
LZFT	B	0.130	27	0.200
	HO	0.084	25	0.200
	HA	0.147	27	0.142
SMAC_2_AAT	B	0.164	27	0.059
	HO	0.154	25	0.127
	HA	0.161	27	0.070
SMAC_2_EFR	B	0.145	27	0.152
	HO	0.134	25	0.200
	HA	0.158	27	0.084
SMAC_2_EART	B*	0.303	27	0.000
	HO*	0.347	25	0.000
	HA*	0.170	27	0.044
TATT	B	0.099	27	0.200
	HO	0.120	25	0.200
	HA	0.119	27	0.200
TAAT	B	0.120	27	0.200
	HO	0.079	25	0.200
	HA*	0.343	27	0.000
TCAT	B	0.110	27	0.200
	HO	0.161	25	0.095
	HA	0.103	27	0.200
WTQMS	B*	0.187	27	0.016
	HO	0.156	25	0.117
	HA*	0.324	27	0.000

Metric	Treatment	Statistic	df	Sig.
MD	B	0.134	27	0.200
	HO	0.111	25	0.200
	HA*	0.175	27	0.032
C2LR	B	0.134	27	0.200
	HO	0.111	25	0.200
	HA	-	-	-
C3LR	B	-	-	-
	HO	-	-	-
	HA*	0.172	27	0.040
C4LR	B	-	-	-
	HO	-	-	-
	HA*	0.172	27	0.040
TCLR	B	0.134	27	0.200
	HO	0.111	25	0.200
	HA*	0.172	27	0.040
HV	B	0.141	27	0.177
	HO*	0.255	25	0.000
	HA	0.145	27	0.153
HV-D	B*	0.313	27	0.000
	HO*	0.401	25	0.000
	HA*	0.210	27	0.004
UIT	B	-	-	-
	HO*	0.286	25	0.000
	HA*	0.197	27	0.009

APPENDIX H – TESTS FOR HETEROSKEDASTICITY

Table H. 1. Results of Heteroskedasticity tests for pairwise comparisons in the Simple scenario.

			Levene Statistic	df1	df2	Sig.
B vs. HO	LZFT	Based on Mean	10.410	1	58	.002
	TATT	Based on Mean	15.481	1	58	.000
	TAAT	Based on Mean	15.975	1	58	.000
	TCAT	Based on Mean	27.719	1	58	.000
	MD	Based on Mean	6.501	1	58	.013
	C2LR	Based on Mean	10.124	1	58	.002
	C3LR	Based on Mean	17.286	1	58	.000
	C4LR	Based on Mean	9.361	1	58	.003
	TCLR	Based on Mean	4.021	1	58	.050
	HV	Based on Mean	12.746	1	58	.001
	HV-D	Based on Mean	.777	1	58	.382
B vs. HA	LZFT	Based on Mean	2.426	1	56	.125
	TATT	Based on Mean	2.660	1	56	.109
	TAAT	Based on Mean	11.738	1	56	.001
	TCAT	Based on Mean	52.938	1	56	.000
	MD	Based on Mean	10.275	1	56	.002
	C2LR	Based on Mean	15.030	1	58	.000
	C3LR	Based on Mean	6.470	1	58	.014
	C4LR	Based on Mean	13.328	1	58	.001
	TCLR	Based on Mean	7.140	1	58	.010
	HV	Based on Mean	5.884	1	56	.019
	HV-D	Based on Mean	65.170	1	56	.000
HO vs. HA	LZFT	Based on Mean	.211	1	56	.647
	TATT	Based on Mean	.238	1	56	.627
	TAAT	Based on Mean	3.786	1	56	.057
	TCAT	Based on Mean	4.948	1	56	.030
	MD	Based on Mean	3.091	1	56	.084
	C2LR	Based on Mean	13.100	1	58	.001
	C3LR	Based on Mean	3.064	1	58	.085
	C4LR	Based on Mean	10.508	1	58	.002
	TCLR	Based on Mean	3.064	1	58	.085
	HV	Based on Mean	.270	1	56	.605
	HV-D	Based on Mean	81.640	1	56	.000
	UIT	Based on Mean	6.148	1	56	.016

Table H. 2. Results of Heteroskedasticity tests for pairwise comparisons in the Moderate scenario.

			Levene Statistic	df1	df2	Sig.
B vs. HO	LZFT	Based on Mean	.139	1	57	.711
	FMAC_6_HFR	Based on Mean	6.228	1	57	.015
	FMAC_6_EART	Based on Mean	21.729	1	57	.000
	SMAC_2_EART	Based on Mean	-	-	-	-
	SMAC_2_EFR	Based on Mean	.029	1	57	.864
	SMAC_2_AAT	Based on Mean	.007	1	57	.933
	TATT	Based on Mean	2.198	1	57	.144
	TAAT	Based on Mean	13.907	1	57	.000
	TCAT	Based on Mean	5.446	1	57	.023
	WTQMS	Based on Mean	21.585	1	57	.000
	MD	Based on Mean	17.336	1	57	.000
	HV	Based on Mean	.861	1	57	.357
	HV-D	Based on Mean	27.354	1	57	.000
B vs. HA	LZFT	Based on Mean	1.193	1	56	.279
	FMAC_6_HFR	Based on Mean	20.506	1	56	.000
	FMAC_6_EART	Based on Mean	40.396	1	56	.000
	SMAC_2_EART	Based on Mean	63.484	1	56	.000
	SMAC_2_EFR	Based on Mean	4.496	1	56	.038
	SMAC_2_AAT	Based on Mean	1.569	1	56	.216
	TATT	Based on Mean	.232	1	56	.632
	TAAT	Based on Mean	6.177	1	56	.016
	TCAT	Based on Mean	1.346	1	56	.251
	WTQMS	Based on Mean	37.234	1	56	.000
	MD	Based on Mean	16.566	1	56	.000
	HV	Based on Mean	15.082	1	56	.000
	HV-D	Based on Mean	34.533	1	56	.000
HO vs. HA	LZFT	Based on Mean	1.967	1	55	.166
	FMAC_6_HFR	Based on Mean	47.875	1	55	.000
	FMAC_6_EART	Based on Mean	10.785	1	55	.002
	SMAC_2_EART	Based on Mean	-	-	-	-
	SMAC_2_EFR	Based on Mean	4.909	1	55	.031
	SMAC_2_AAT	Based on Mean	1.350	1	55	.250
	TATT	Based on Mean	3.121	1	55	.083
	TAAT	Based on Mean	5.395	1	55	.024
	TCAT	Based on Mean	.913	1	55	.343
	WTQMS	Based on Mean	9.653	1	55	.003
	MD	Based on Mean	.921	1	55	.341
	HV	Based on Mean	18.416	1	55	.000
	HV-D	Based on Mean	.973	1	55	.328
	UIT	Based on Mean	27.408	1	55	.000

Table H. 3. Results of Heteroskedasticity tests for pairwise comparisons in the Complex scenario.

			Levene Statistic	df1	df2	Sig.
B vs. HO	LZFT	Based on Mean	2.292	1	50	.136
	SMAC_2_AAT	Based on Mean	.287	1	50	.595
	SMAC_2_EFR	Based on Mean	.607	1	50	.439
	SMAC_2_EART	Based on Mean	3.382	1	50	.072
	TATT	Based on Mean	2.096	1	50	.154
	TAAT	Based on Mean	1.238	1	50	.271
	TCAT	Based on Mean	.032	1	50	.860
	WTQMS	Based on Mean	.436	1	50	.512
	MD	Based on Mean	1.722	1	50	.195
	C2LR	Based on Mean	1.681	1	50	.201
	TCLR	Based on Mean	1.681	1	50	.201
	HV	Based on Mean	.832	1	50	.366
	HV-D	Based on Mean	30.286	1	50	.000
B vs. HA	LZFT	Based on Mean	12.088	1	52	.001
	SMAC_2_AAT	Based on Mean	.982	1	52	.326
	SMAC_2_EFR	Based on Mean	1.029	1	52	.315
	SMAC_2_EART	Based on Mean	15.733	1	52	.000
	TATT	Based on Mean	.002	1	52	.961
	TAAT	Based on Mean	23.367	1	52	.000
	TCAT	Based on Mean	.369	1	52	.546
	WTQMS	Based on Mean	31.547	1	52	.000
	MD	Based on Mean	12.346	1	52	.001
	C2LR	Based on Mean	-	-	-	-
	C3LR	Based on Mean	-	-	-	-
	C4LR	Based on Mean	-	-	-	-
	TCLR	Based on Mean	8.217	1	52	.006
	HV	Based on Mean	27.620	1	52	.000
	HV-D	Based on Mean	101.324	1	52	.000
HO vs. HA	LZFT	Based on Mean	2.789	1	50	.101
	SMAC_2_AAT	Based on Mean	1.581	1	50	.214
	SMAC_2_EFR	Based on Mean	1.870	1	50	.178
	SMAC_2_EART	Based on Mean	15.679	1	50	.000
	TATT	Based on Mean	1.726	1	50	.195
	TAAT	Based on Mean	20.810	1	50	.000
	TCAT	Based on Mean	.711	1	50	.403
	WTQMS	Based on Mean	29.464	1	50	.000
	MD	Based on Mean	14.526	1	50	.000
	C2LR	Based on Mean	-	-	-	-
	C3LR	Based on Mean	-	-	-	-
	C4LR	Based on Mean	-	-	-	-
	TCLR	Based on Mean	10.908	1	50	.002
	HV	Based on Mean	11.456	1	50	.001
	HV-D	Based on Mean	286.921	1	50	.000
	UIT	Based on Mean	39.199	1	50	.000

APPENDIX I – BOXPLOTS FOR THE SIMPLE SCENARIO

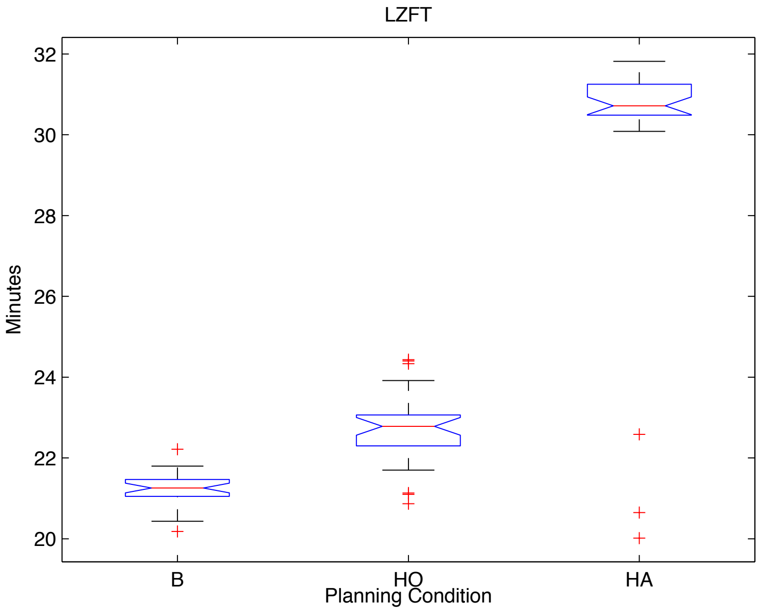


Figure I. 1. Landing Zone Foul Time (LZFT).

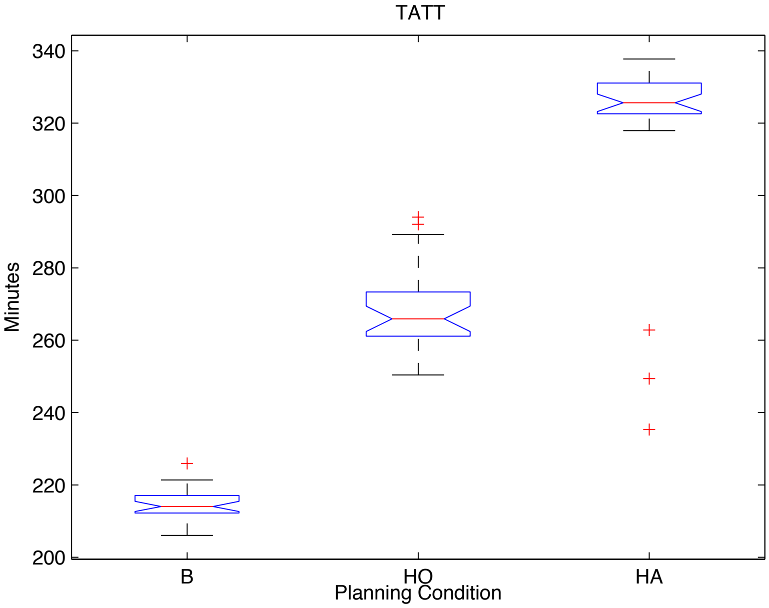


Figure I. 2. Total Aircraft Taxi Time (TATT).

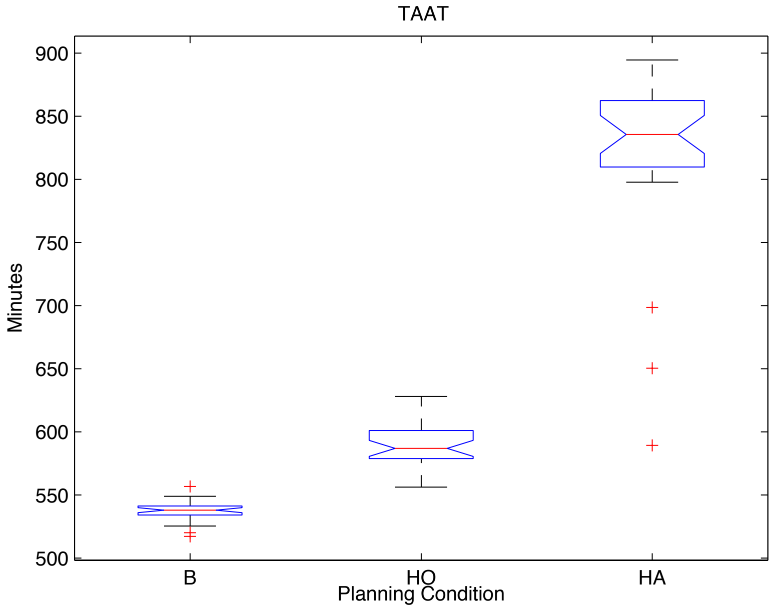


Figure I. 3. Total Aircraft Active Time (TAAT).

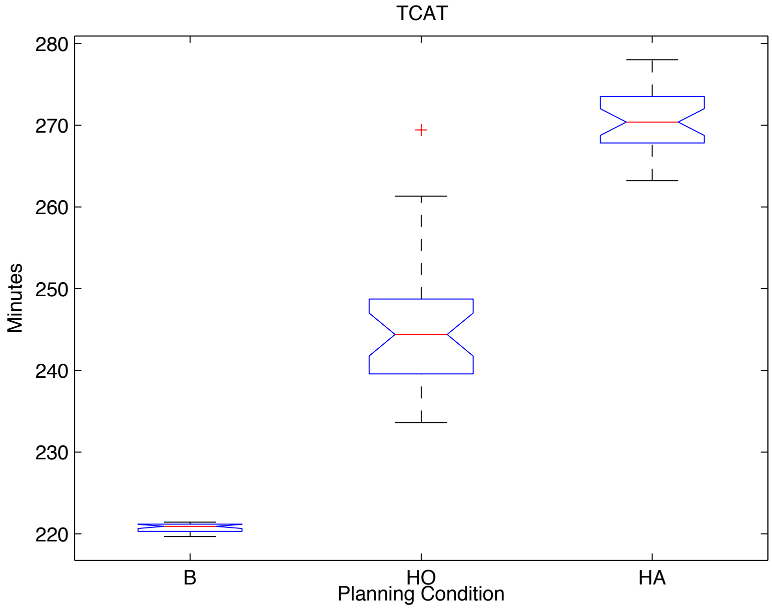


Figure I. 4. Total Crew Active Time (TCAT).

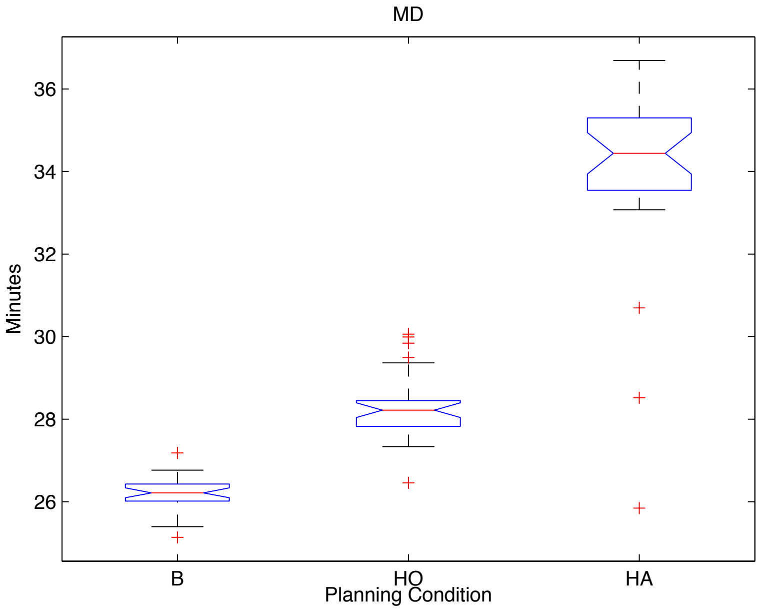


Figure I. 5. Mission Duration (MD).

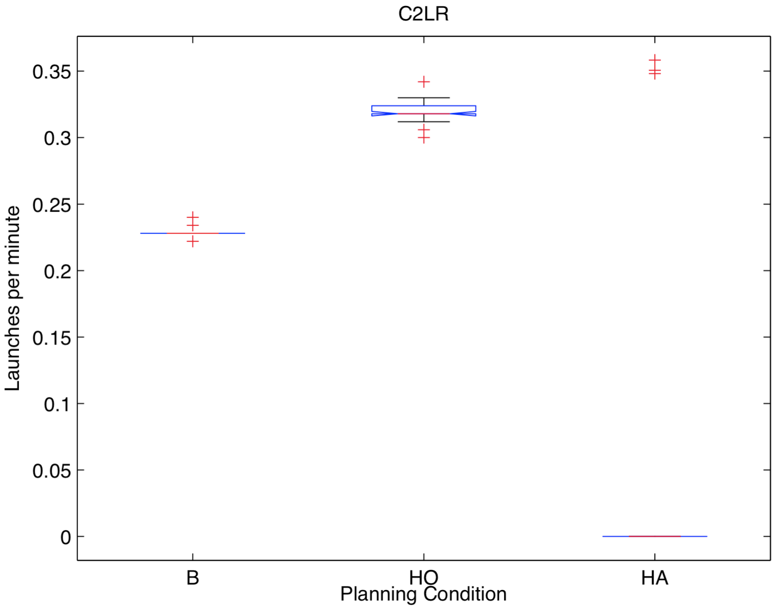


Figure I. 6. Catapult 2 Launch Rate (C2LR).

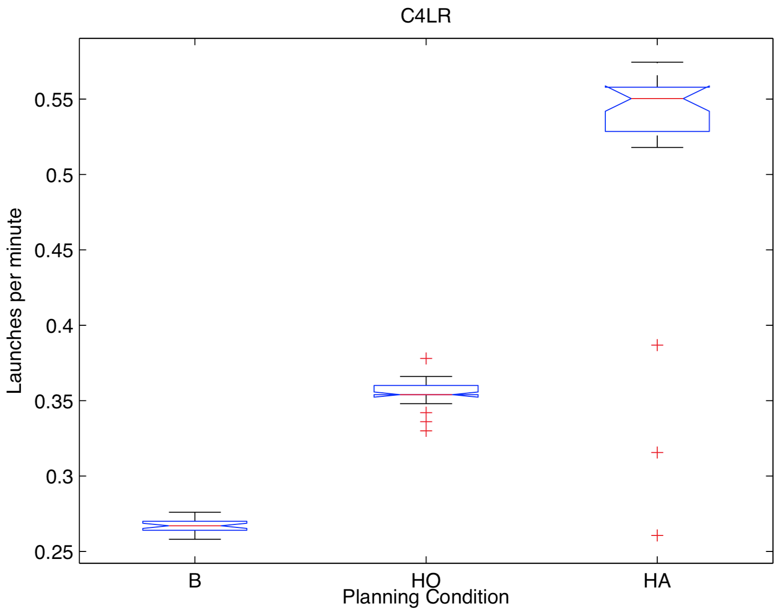


Figure I. 7. Catapult 4 Launch Rate (C4LR).

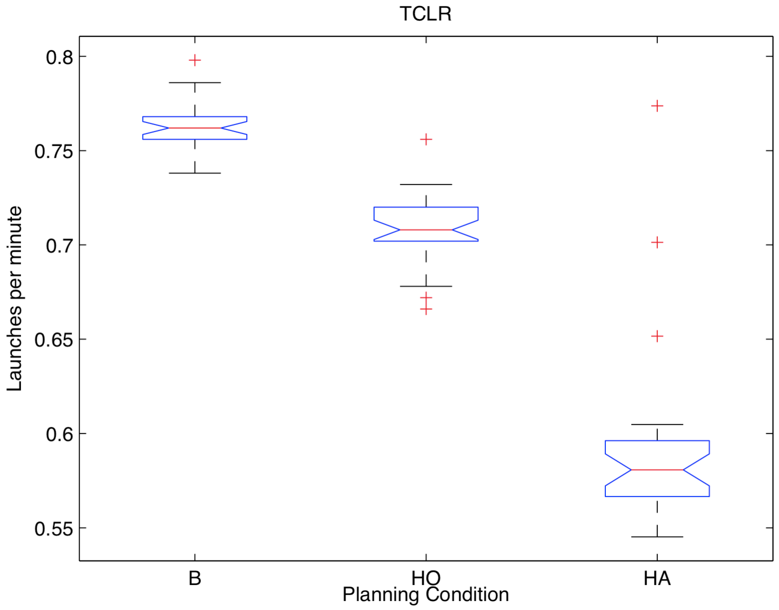


Figure I. 8. Total Catapult Launch Rate (TCLR).

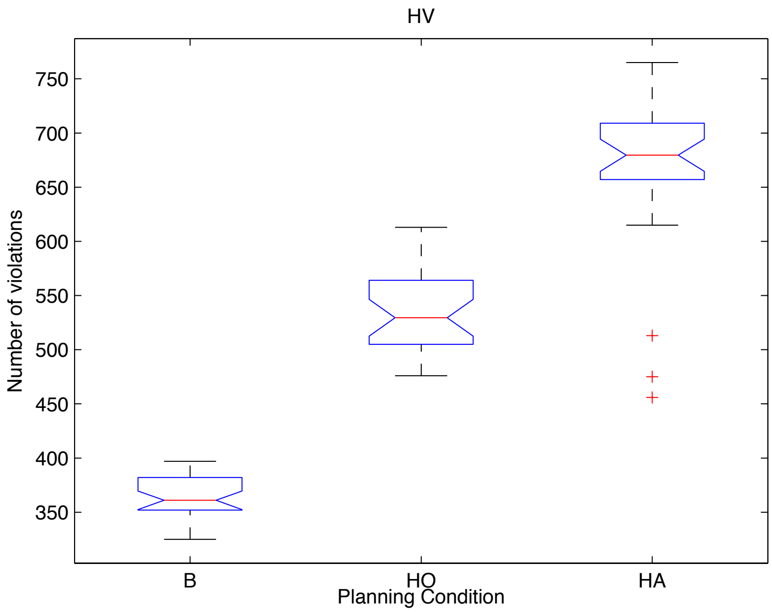


Figure I. 9. Halo Violations (HV).

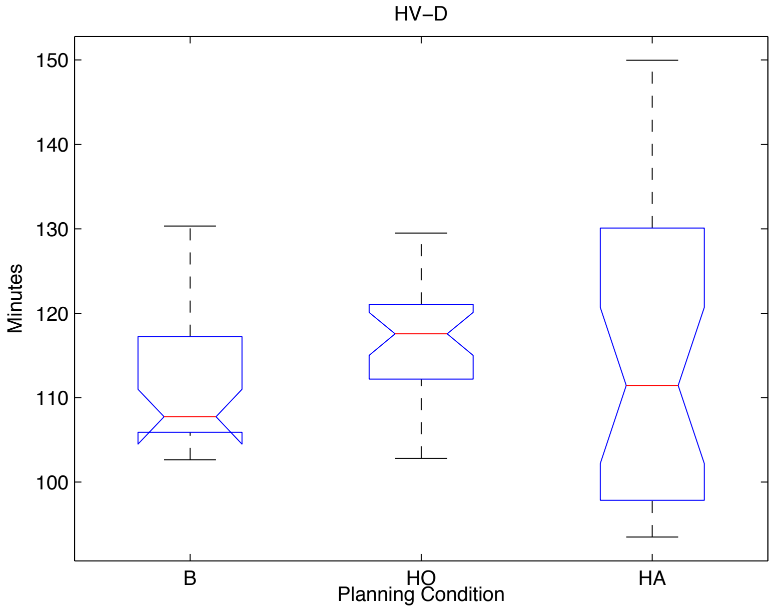


Figure I. 10. Halo Violation Durations (HV-D). Spikes in plot imply 95% confidence interval notches extend past included data.

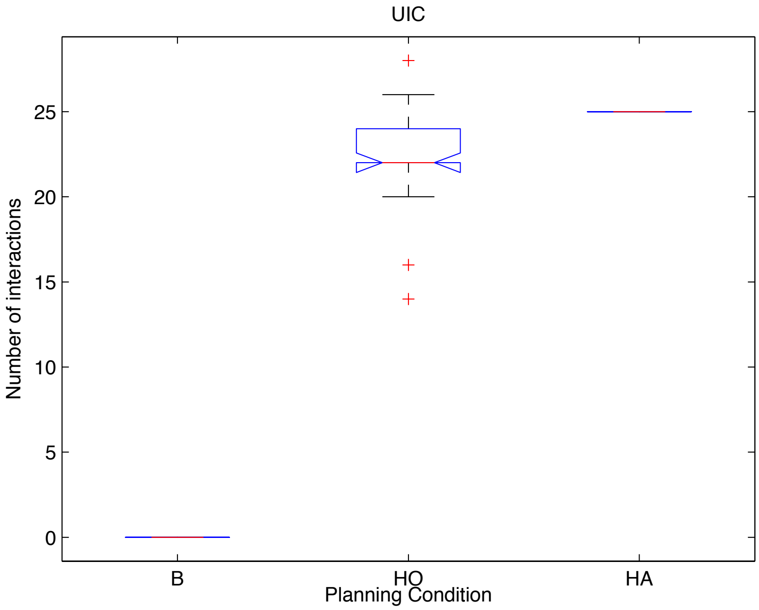


Figure I. 11. User Interaction Count (UIC). Spikes in plot imply 95% confidence interval notches extend past included data.

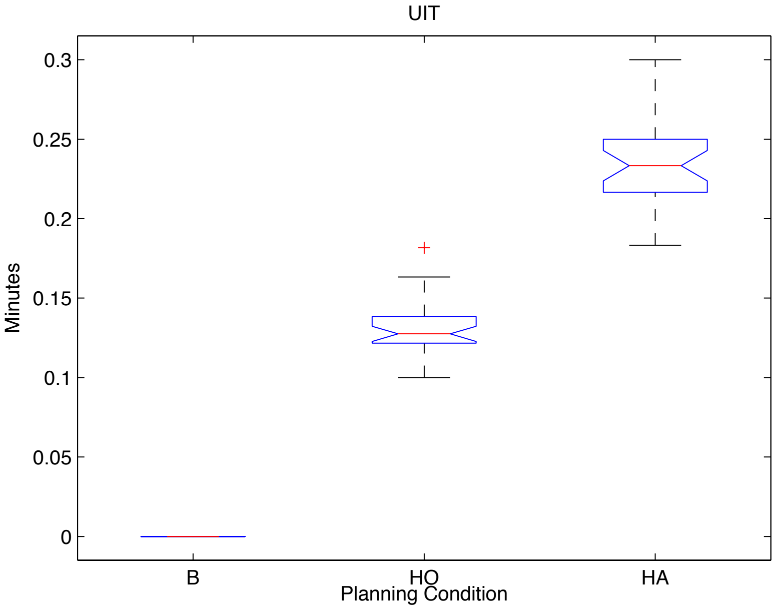


Figure I. 12. User Interaction Time (UIT).

APPENDIX J – BOXPLOTS FOR THE MODERATE SCENARIO

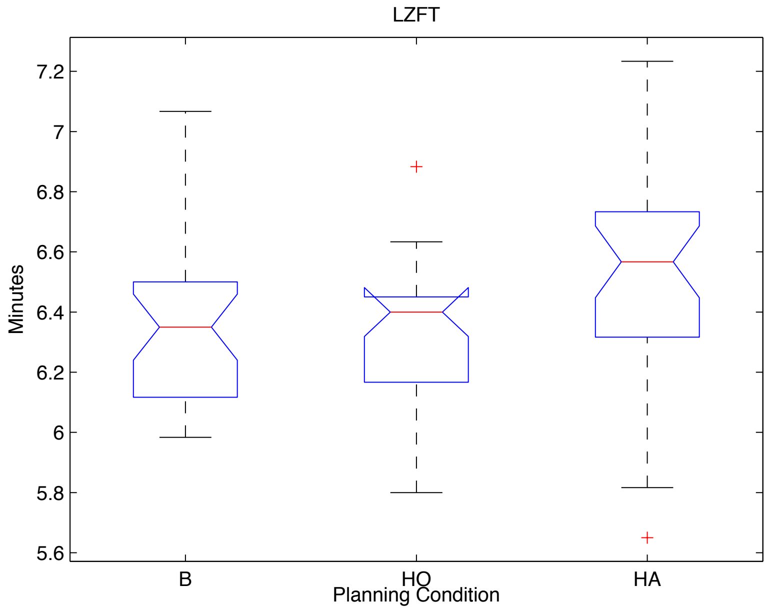


Figure J. 1. Landing Zone Foul Time (LZFT). Spikes in plot imply 95% confidence interval notches extend past included data.

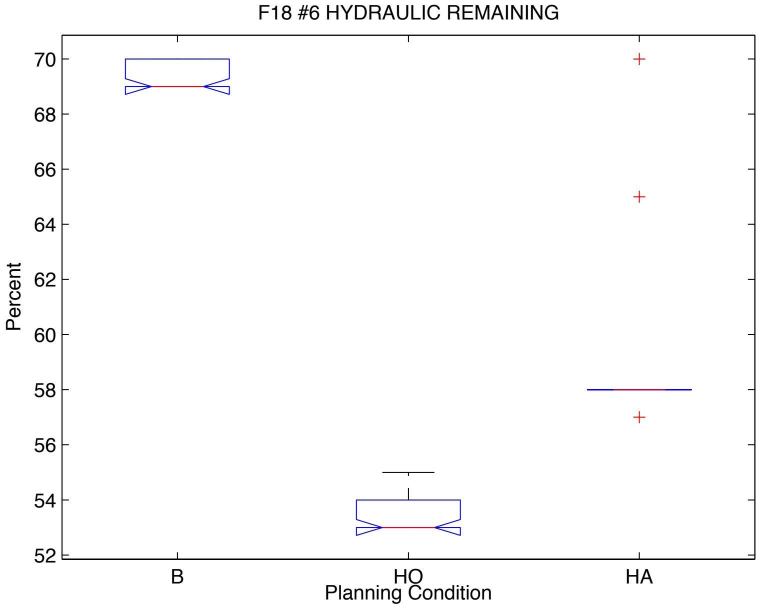


Figure J. 2. FMAC #6 Hydraulic Fluid Remaining (FMAC 6 HFR). Spikes in plot imply 95% confidence interval notches extend past included data.

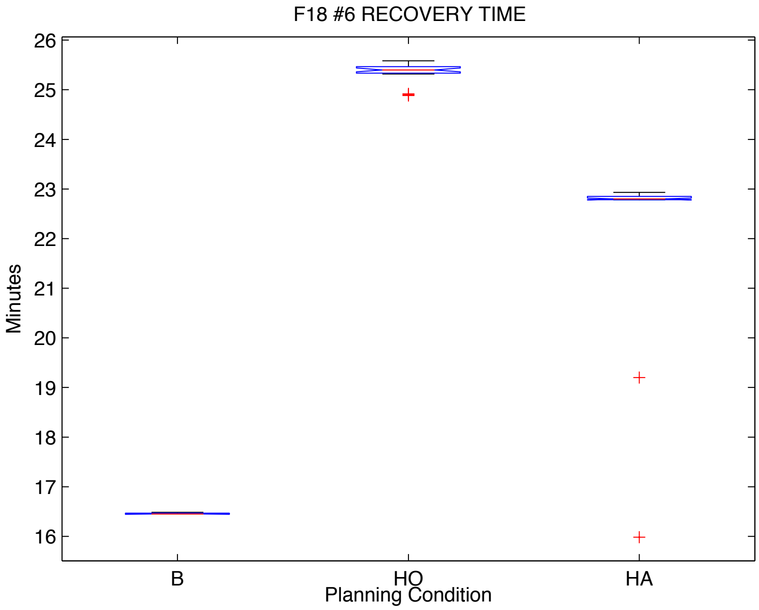


Figure J. 3. FMAC #6 Emergency Aircraft Recovery Time (FMAC 6 EART).

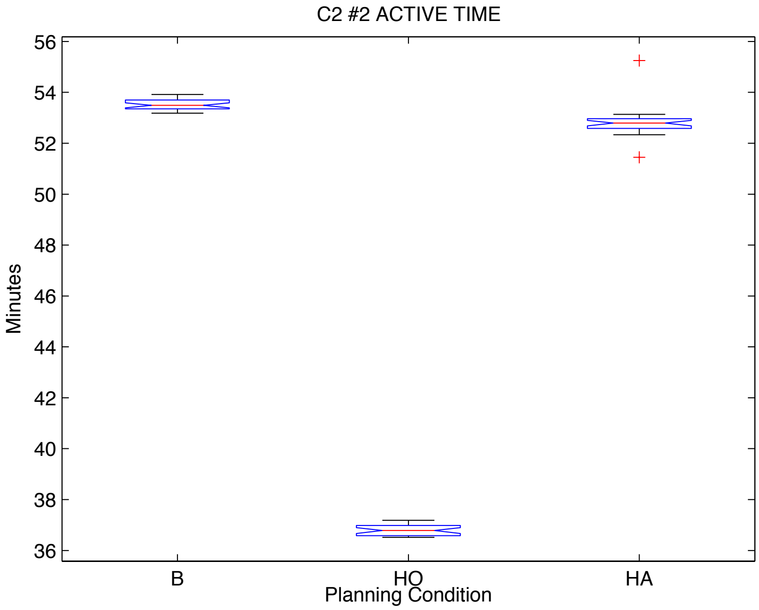


Figure J. 4. SMAC #2 Aircraft Active Time (SMAC 2 AAT).

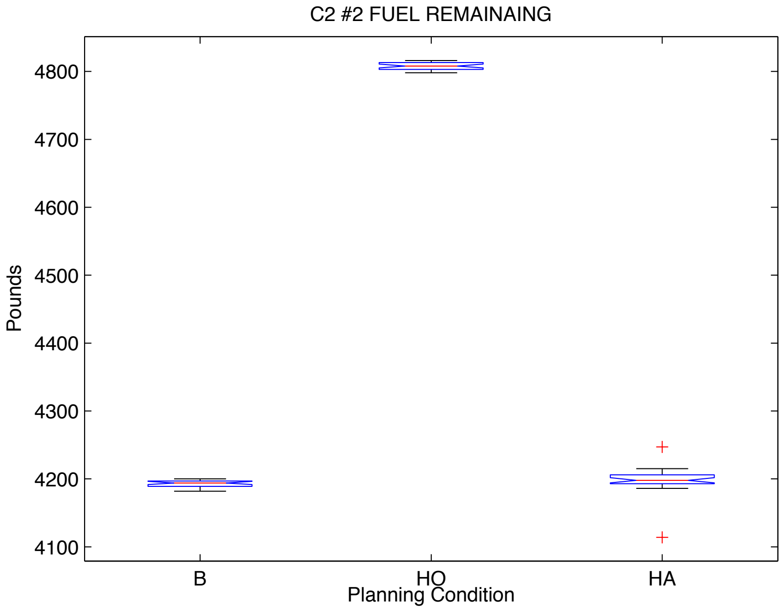


Figure J. 5. SMAC #2 Emergency Fuel Remaining (SMAC 2 EFR).

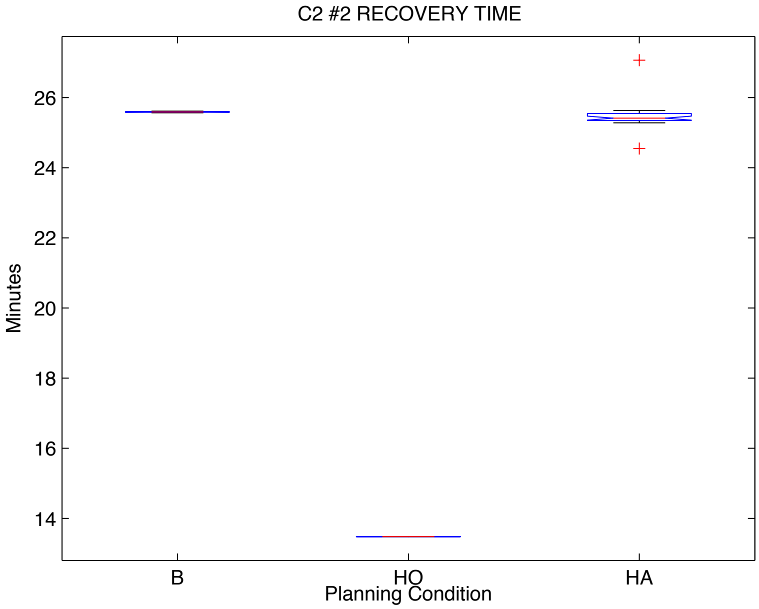


Figure J. 6. SMAC #2 Emergency Aircraft Recovery Time (SMAC 2 EART).

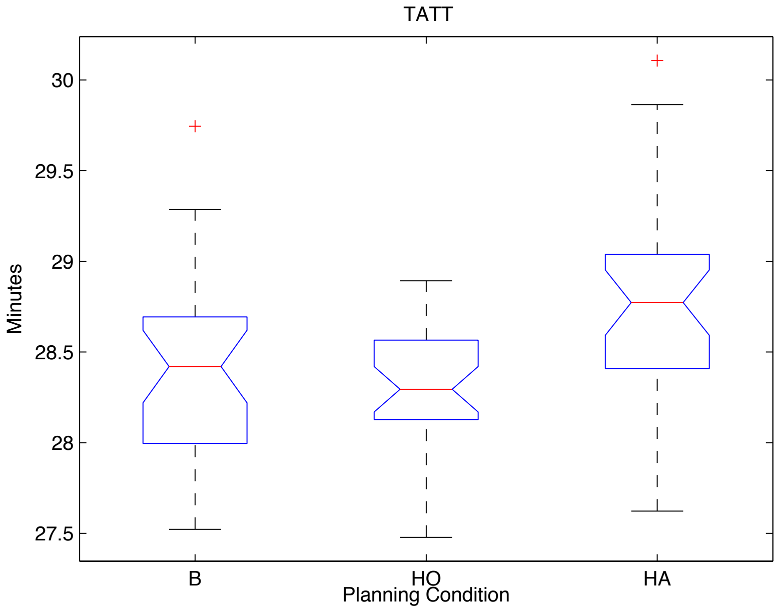


Figure J. 7. Total Aircraft Taxi Time (TATT).

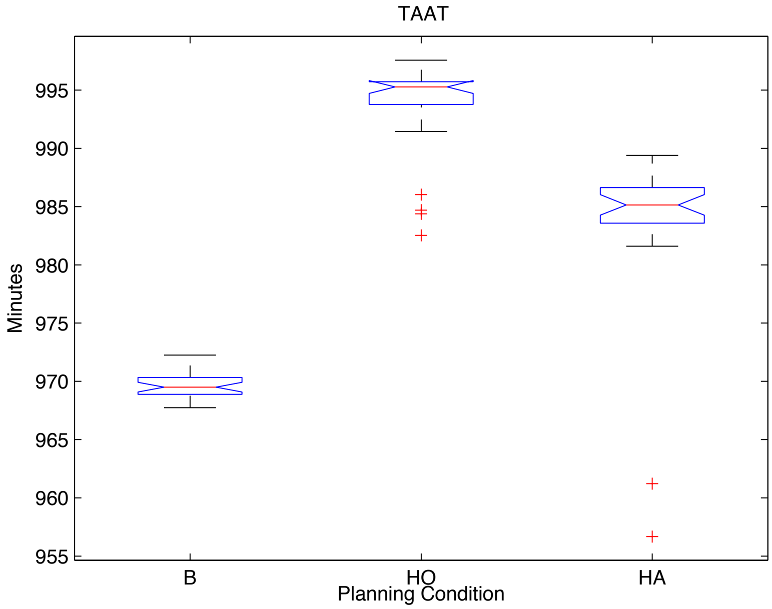


Figure J. 8. Total Aircraft Active Time (TAAT).

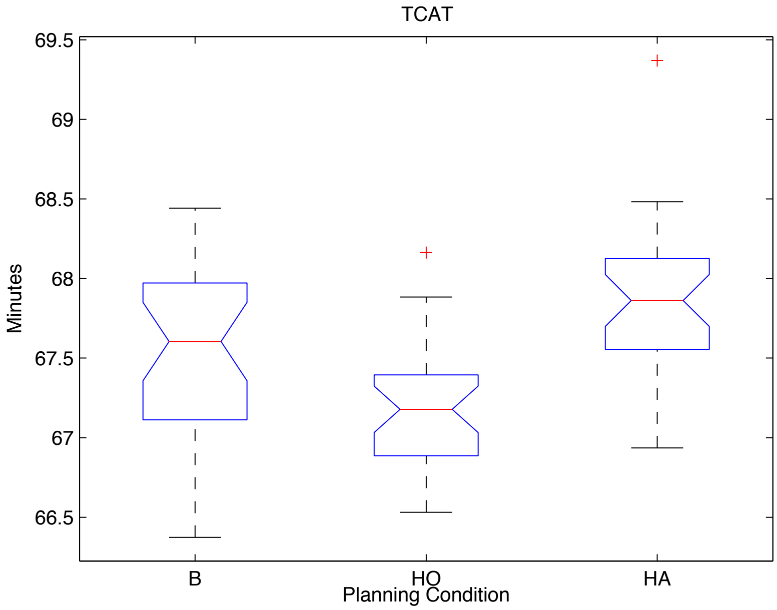


Figure J. 9. Total Crew Active Time (TCAT).

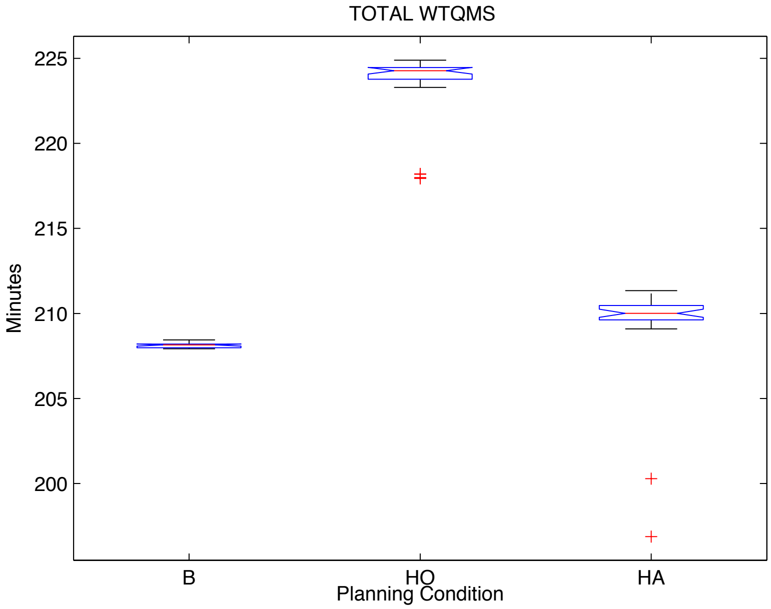


Figure J. 10. Wait Time in Queue in Marshal Stack (WTQMS).

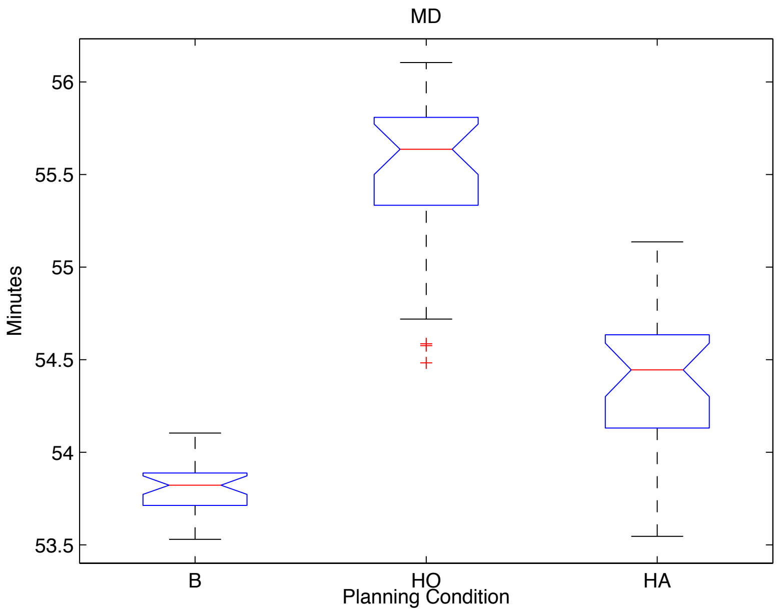


Figure J. 11. Mission Duration (MD).

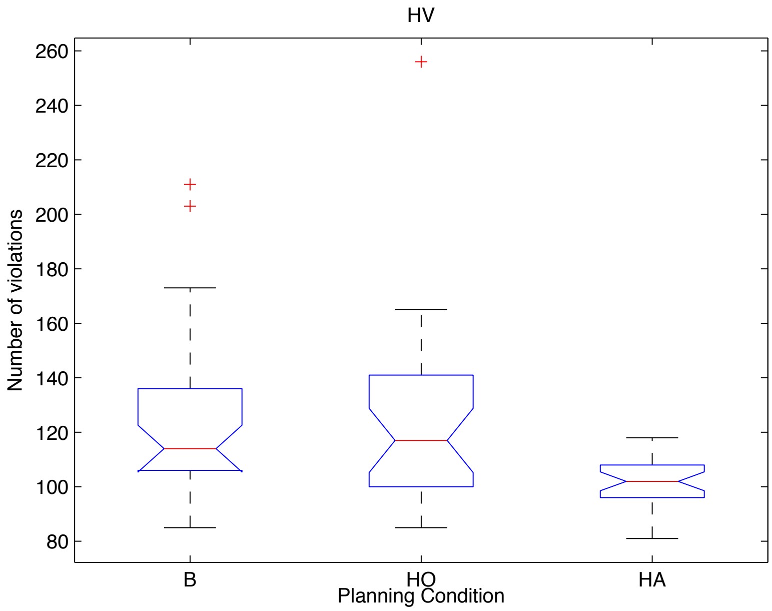


Figure J. 12. Halo Violations (HV).

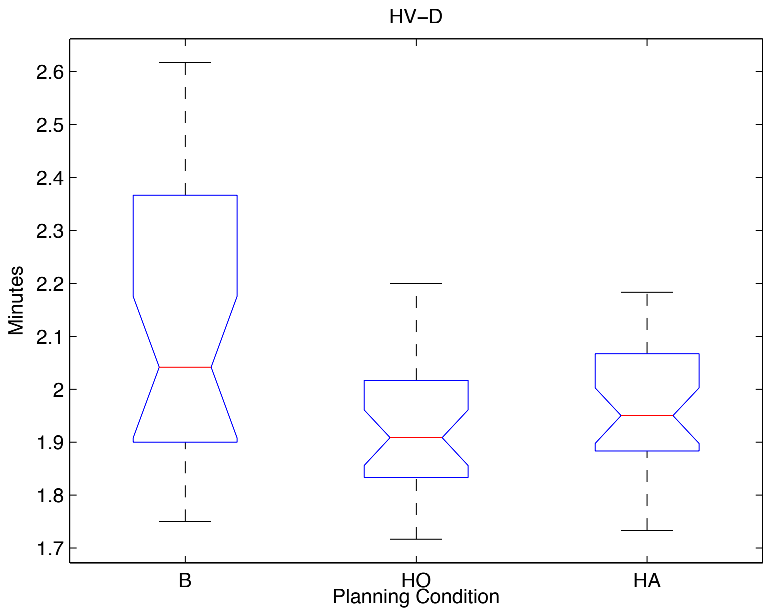


Figure J. 13. Halo Violation Durations (HV-D).

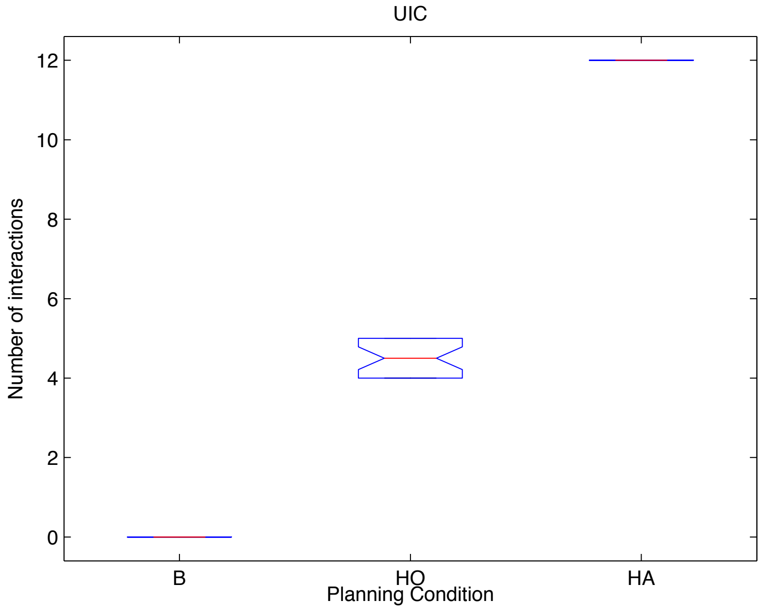


Figure J. 14. User Interaction Count (UIC).

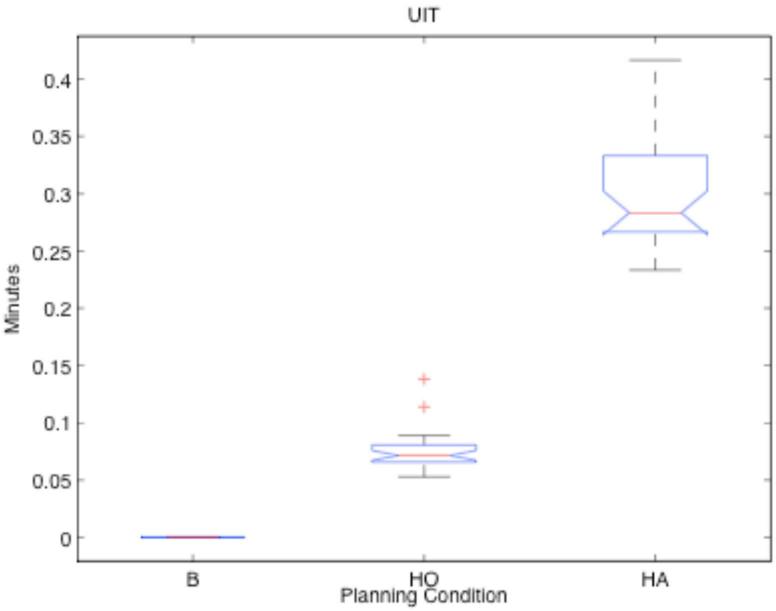


Figure J. 15. User Interaction Time (UIT).

APPENDIX K – BOXPLOTS FOR THE COMPLEX SCENARIO

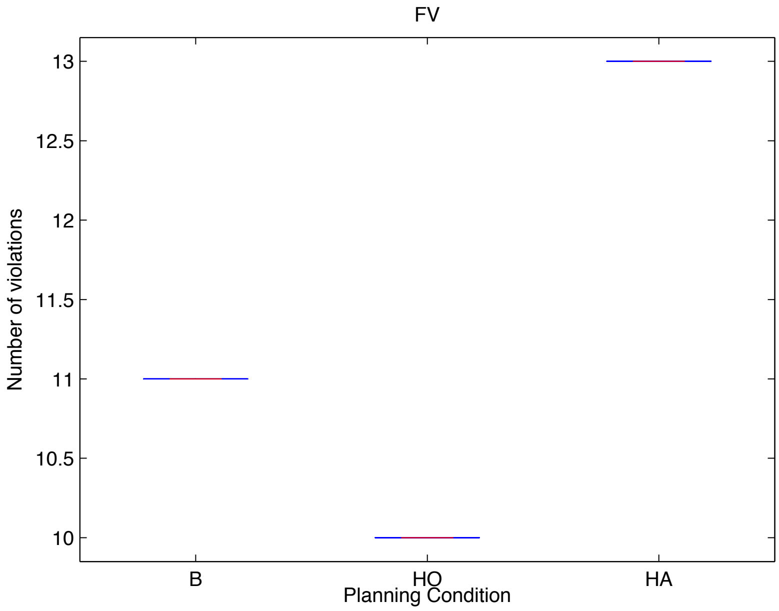


Figure K. 1. Fuel Violation (FV)

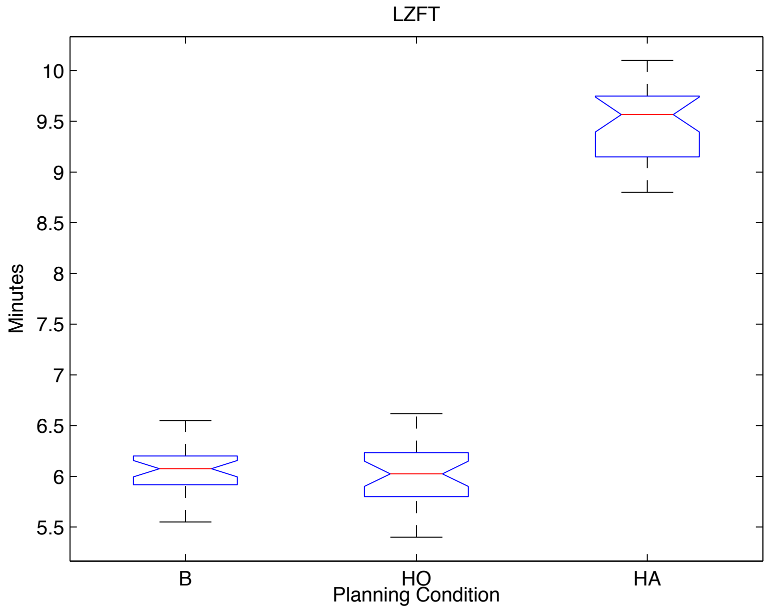


Figure K. 2. Landing Zone Foul Time (LZFT).

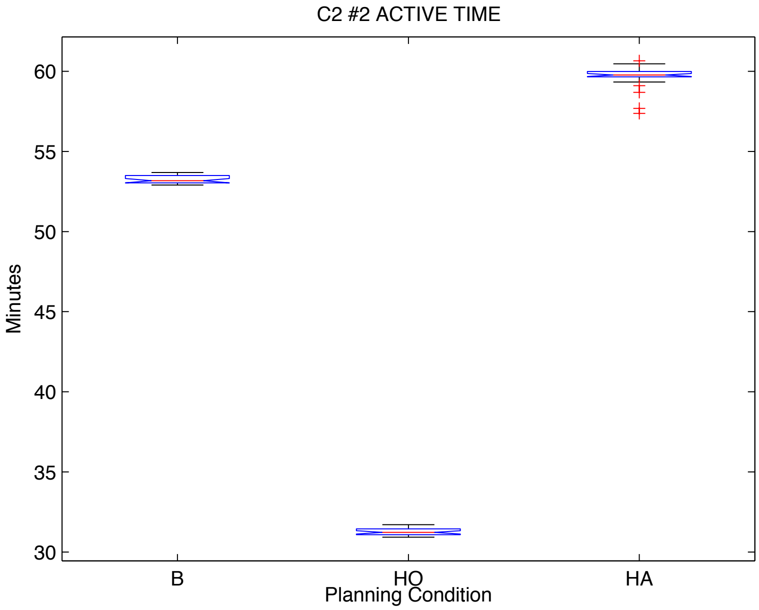


Figure K. 3. SMAC #2 Aircraft Active Time (SMAC 2 AAT).

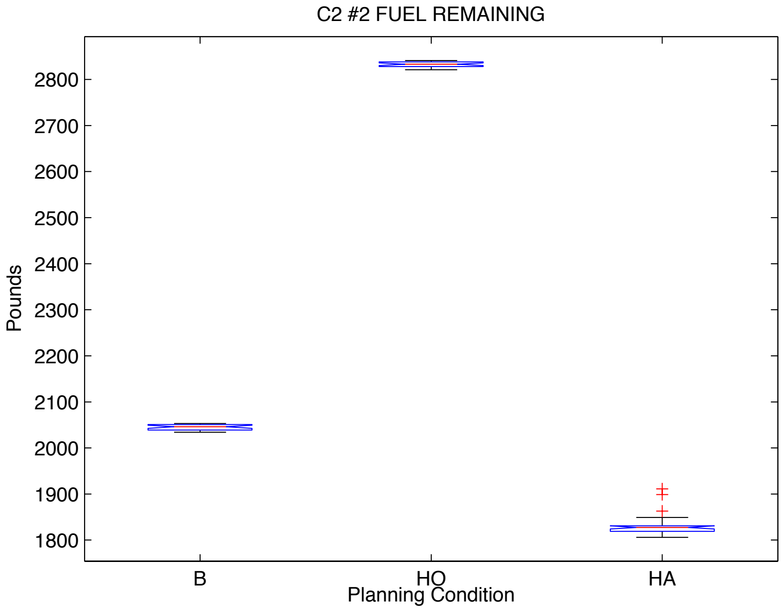


Figure K. 4. SMAC #2 Emergency Fuel Remaining (SMAC 2 EFR).

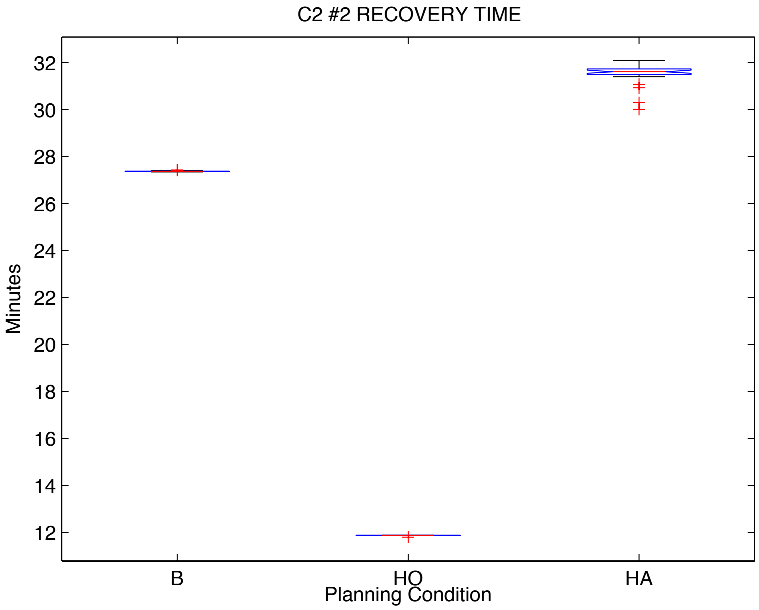


Figure K. 5. SMAC #2 EART (SMAC 2 EART).

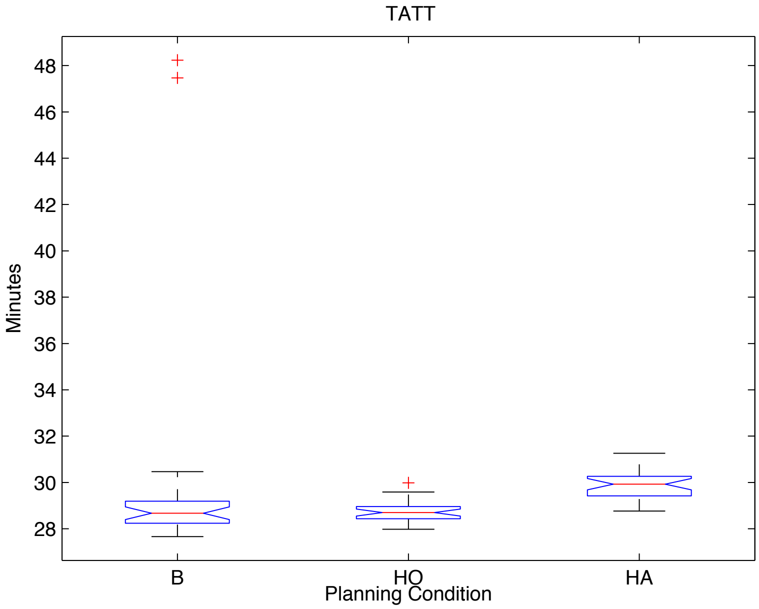


Figure K. 6. Total Aircraft Taxi Time (TATT).

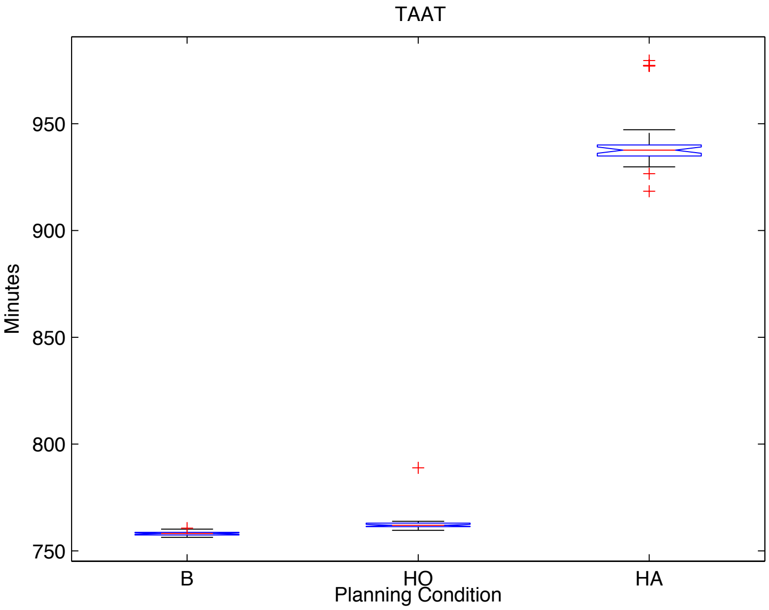


Figure K. 7. Total Aircraft Active Time (TAAT).

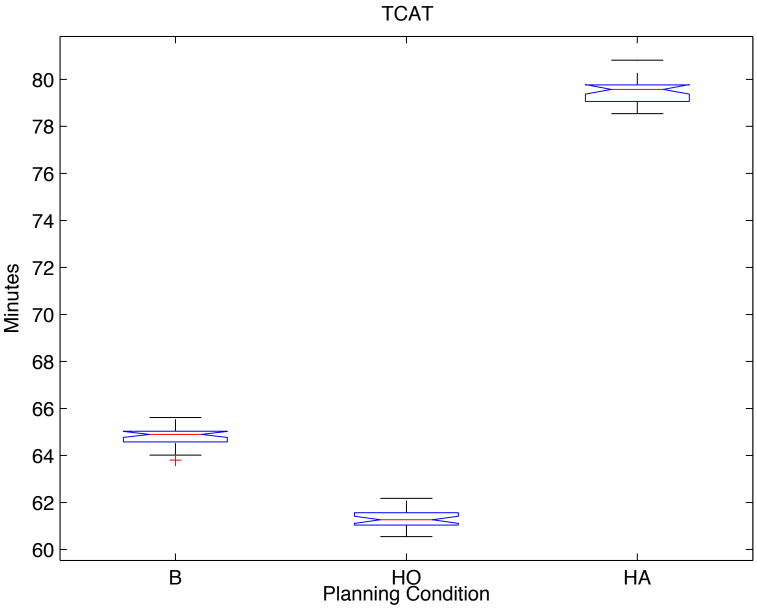


Figure K. 8. Total Crew Active Time (TCAT).

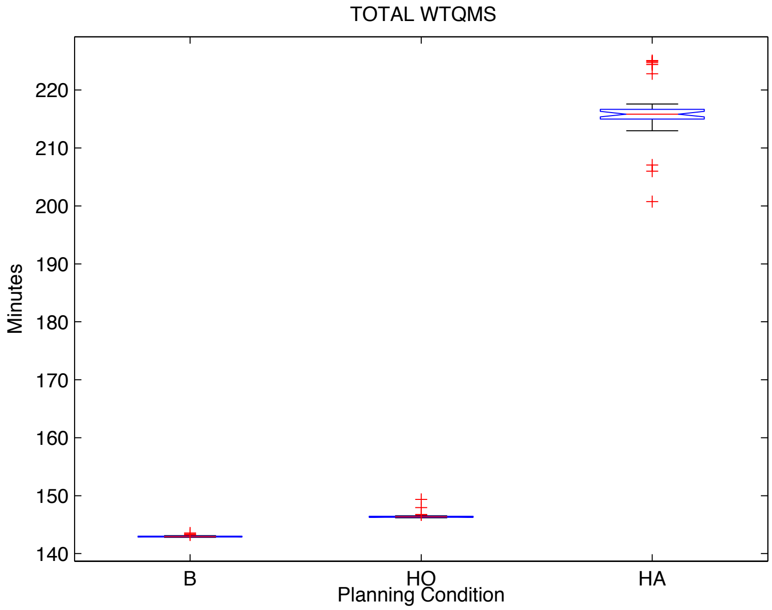


Figure K. 9. Wait Time in Queue in Marshal Stack (WTQMS).

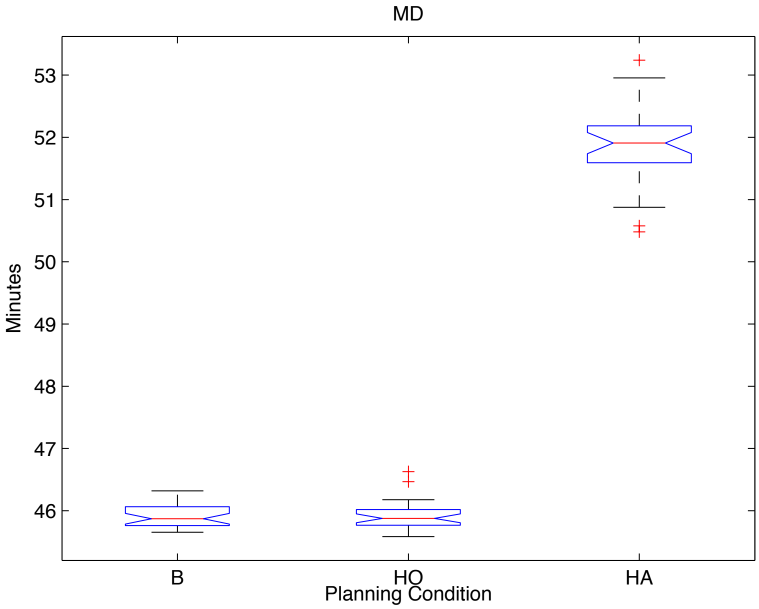


Figure K. 10. Mission Duration (MD).

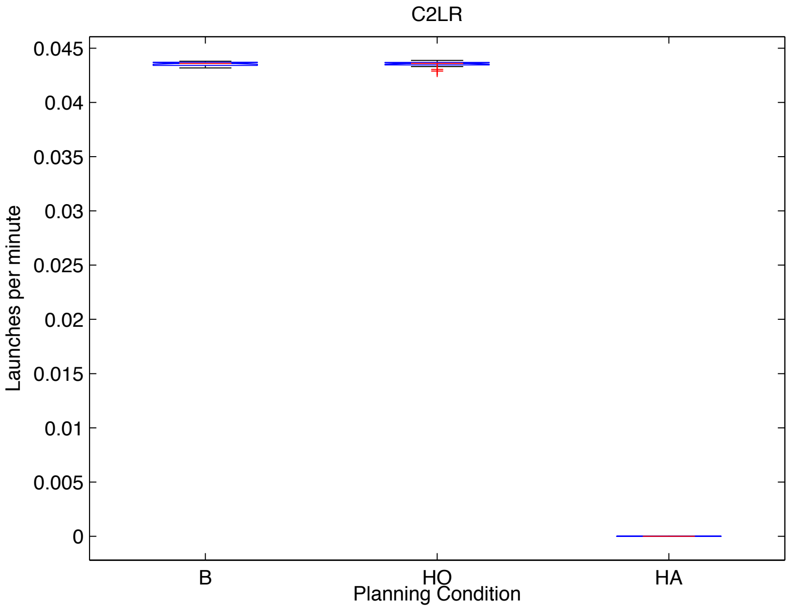


Figure K. 11. Catapult 2 Launch Rate (C2LR).

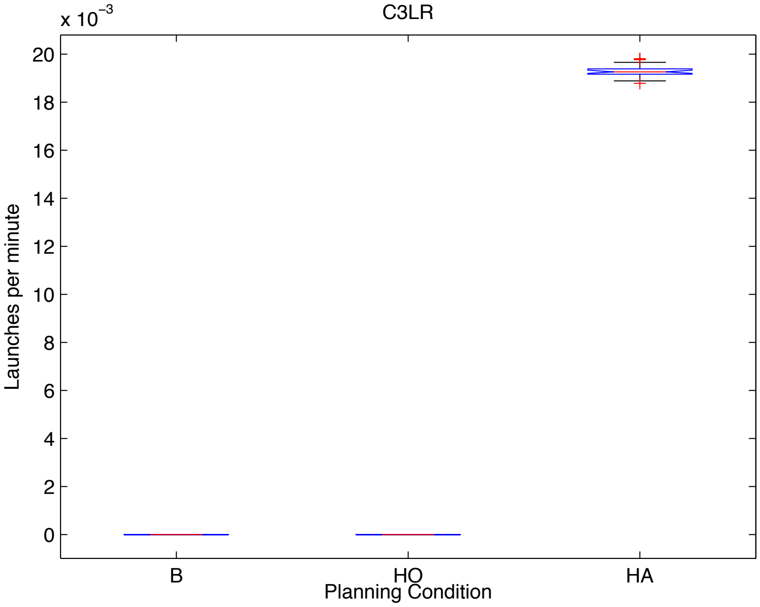


Figure K. 12. Catapult 3 Launch Rate (C3LR).

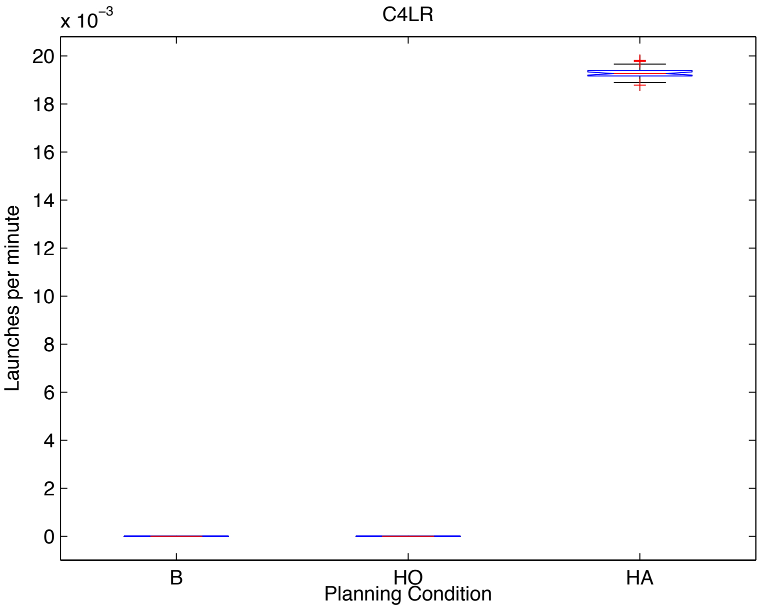


Figure K. 13. Catapult 4 Launch Rate (C4LR).

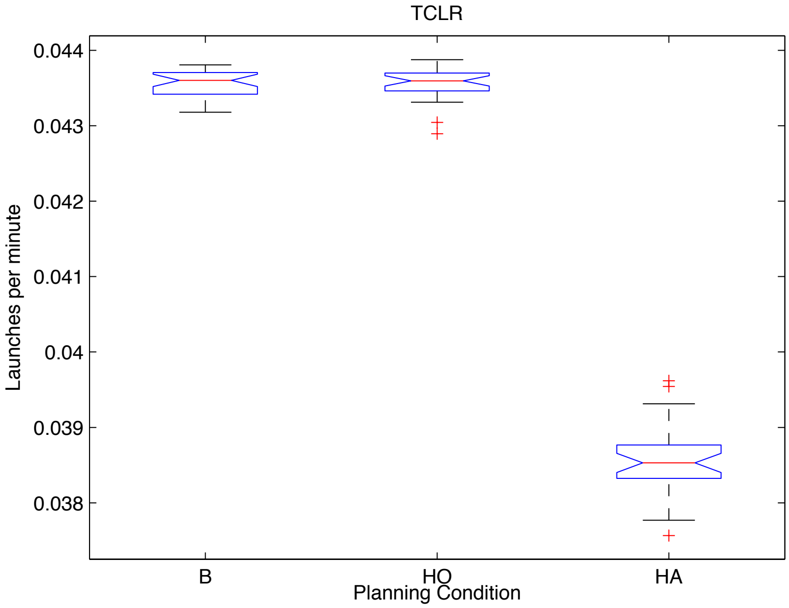


Figure K. 14. Total Catapult Launch Rate (TCLR).

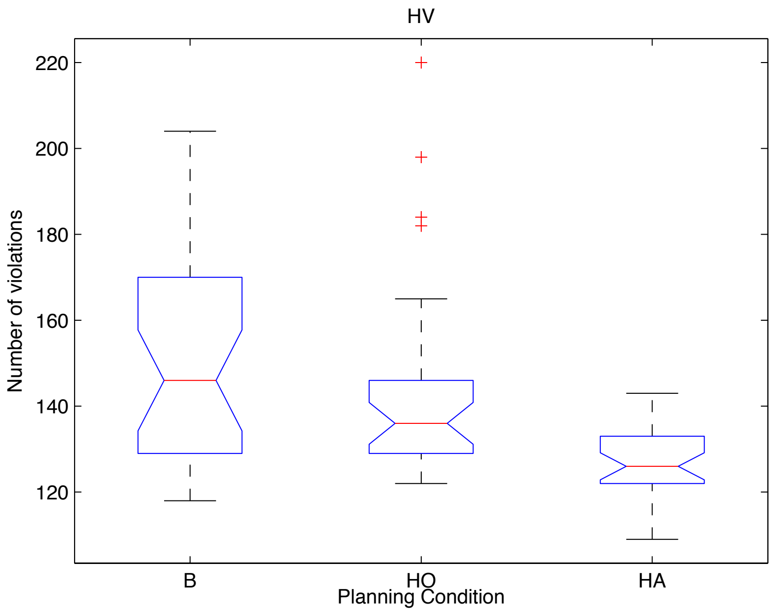


Figure K. 15. Halo Violations (HV).

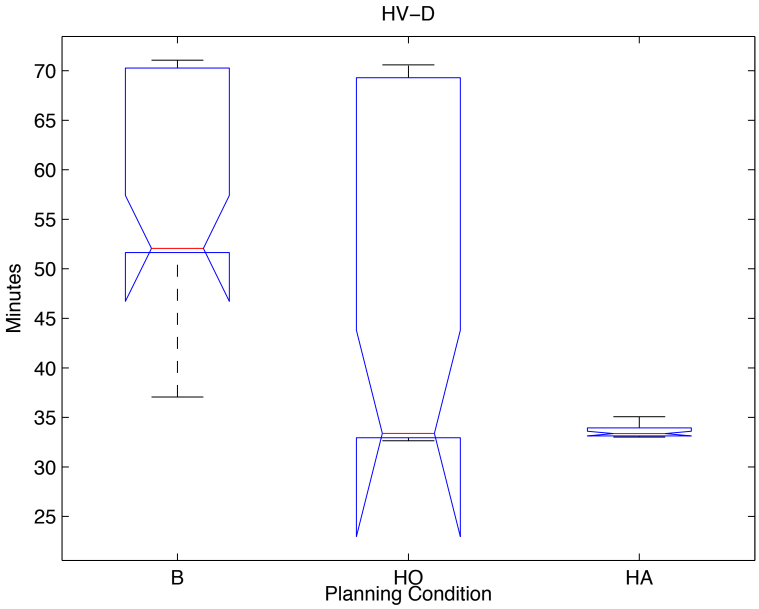


Figure K. 16. Halo Violation Durations (HV-D). Spikes in plot imply 95% confidence interval notches extend past included data.

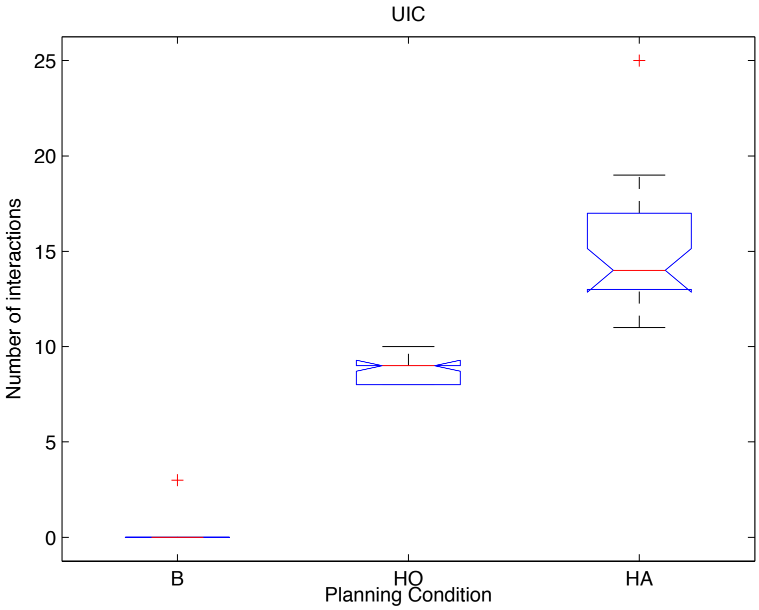


Figure K. 17. User Interaction Count (UIC). Spikes in plot imply 95% confidence interval notches extend past included data.

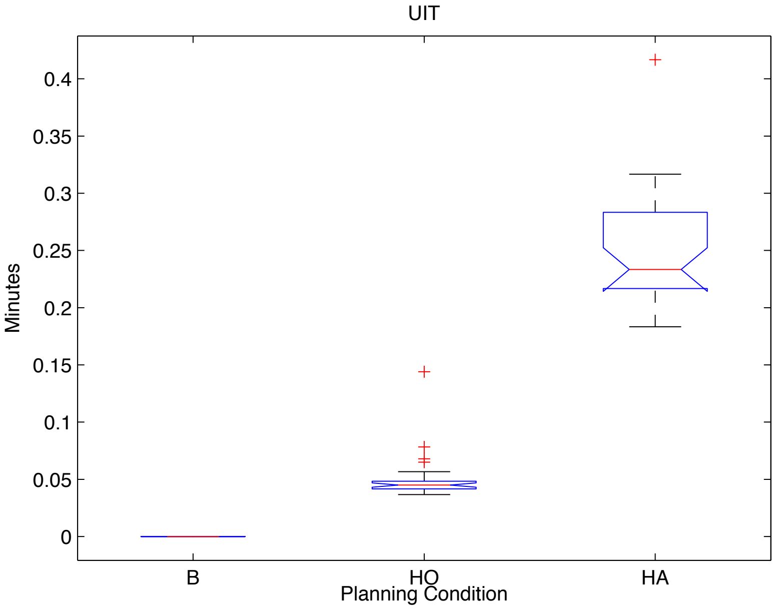


Figure K. 18. User Interaction Time (UIT).

REFERENCES

1. Sheridan, T.B., *Telerobotics, Automation and Human Supervisory Control*. 1992, Cambridge, MA: The MIT Press.
2. Roth, E., Hanson, M.L., Hopkins, C., Mancuso, V. and Zacharias, G.L., *Human in the Loop Evaluations of a Mixed-initiative System for Planning and Control of Multiple UAV Teams.* Proceedings of the HFES 48th Annual Meeting, 2004, New Orleans, LA.
3. Clare, A.S., *Dynamic Human-Computer Collaboration in Real-time Unmanned Vehicle Scheduling*. 2010, S.M. Thesis, Massachusetts Institute of Technology, Cambridge, MA.
4. Johnson, K., Ren, L., Kuchar, J.K. and Oman, C.M., *Interaction of Automation and Time Pressure in a Route Replanning Task*. Proceedings of the International Conference on Human-Computer Interaction in Aeronautics (HCI-Aero), 2002, Cambridge, MA.
5. Marquez, J.J., Cummings, M.L., Roy, N., Kunda, M. and Newman, D.J., *Collaborative Human-Computer Decision Support for Planetary Surface Traversal*. Proceedings of the AIAA Infotech@Aerospace Conference, 2005, Arlington, VA.
6. Goodrich, M.A., McLain, T.W., Anderson, J., Sun, J. and Crandall, J.W., *Managing Autonomy in Robot Teams: Observations from Four Experiments*. Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction (HRI '07), 2007, Washington D.C.
7. Naval Studies Board, *Autonomous Vehicles in Support of Naval Operations*. National Research Council, Washington D.C., 2005.
8. Kramer, L.A. and Smith, S.F., *Optimizing for Change: Mixed-Initiative Resource Allocation with the AMC Barrel Allocator*. Proceedings of the 3rd International NASA Workshop on Planning and Scheduling for Space, 2002, Houston, TX.
9. Smith, P., McCoy, E. and Layton, C., "Brittleness in the Design of Cooperative Problem-Solving Systems: The Effects on User Performance." *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 1997. **27**: p. 360-371.
10. Fitts, P.M., ed. *Human Engineering for an Effective Air Navigation and Traffic Control System*. 1951, National Research Council: Washington, DC.
11. Hollnagel, E. and Bye, A., "Principles for Modelling Function Allocation." *International Journal of Human-Computer Studies*, 2000. **52**: p. 253-265.
12. Gigerenzer, G. and Goldstein, D.G., "Reasoning the Fast and Frugal Way: Models of Bounded Rationality." *Psychological Review*, 1996. **103**: p. 650 - 669.
13. Gigerenzer, G. and Todd, P.M., *Simple Heuristics That Make Us Smart*, ed. S. Stich. 1999, Oxford, NY: Oxford University Press, Inc. 416.
14. Cummings, M.L. and Thornburg, K.T., "Paying Attention to the Man Behind the Curtain." *IEEE Pervasive Computing*, 2011. **10**(1): p. 58-62.
15. Malasky, J., Forest, L.M., Khan, A.C. and Key, J.R., *Experimental Evaluation of Human-Machine Collaborative Algorithms in Planning for Multiple UAVs*. IEEE Conference on Systems, Man, and Cybernetics 2005.
16. Schlager, K.J., "Systems Engineering - Key to Modern Development." *IRE Transactions*, 1956. **EM-3**: p. 64-66.
17. National Aeronautics and Space Administration, *Mars Climate Orbiter Mishap Investigation Board Phase I Report*. 1999.
18. Royce, W.W., "Managing the Development of Large Software Systems: Concepts and Techniques." *WESCON Technical Papers*, 1970. **14**: p. A1-1 - A1-9.
19. Federal Highway Administration, *Clarus: Concept of Operations*. Washington D. C., 2005.
20. Boehm, B., "A Spiral Model of Software Development and Enhancement." *Computer*, 1988: p. 61-72.
21. Boehm, B., *Spiral Development: Experience, Principles, and Refinements*. Proceedings of the Spiral Development Workshop, 2000, University of Southern California: Software Engineering Institute.
22. *Handbook of Human Factors and Ergonomics Methods*, ed. N. Stanton, et al. 2005: CRC Press.
23. *Guide to Human Performance Measurements*. 2001, American Institute of Aeronautics and Astronautics: Reston, VA.

24. Jackson, H.F., Boggs, P.T., Nash, S.G. and Powell, S., "Guidelines for Reporting Results of Computational Experiments: Report of the Ad Hoc Committee." *Mathematical Programming*, 1991. **49**: p. 413-425.

25. Barr, R.S., Golden, B.L., Kelly, J.P., Resende, M.G.C. and Stewart, W.R., "Designing and Reporting on Computational Experiments with Heuristic Methods." *Journal of Heuristics*, 1995. **1**: p. 9-32.

26. Pina, P.E., Cummings, M.L., Crandall, J.W. and Della Penna, M., *Identifying Generalizable Metric Classes to Evaluate Human-Robot Teams*. Proceedings of the Metrics for Human-Robot Interaction Workshop at the 3rd Annual Conference on Human-Robot Interaction, 2008, Amsterdam, The Netherlands.

27. Pina, P.E., Donmez, B. and Cummings, M.L., *Selecting Metrics to Evaluate Human Supervisory Control Applications.* MIT Humans and Automation Laboratory, Cambridge, MA, 2008

28. Olsen, R.O. and Goodrich, M.A., *Metrics for Evaluating Human-Robot Interactions*. Proceedings of the NIST Performance Metrics for Intelligent Systems Workshop, 2003.

29. Steinfeld, A., Fong, T., Kaber, D., Lewis, M., Scholtz, J., Schultz, A. and Goodrich, M.A., *Common Metrics for Human-Robot Interaction*. 1st Annual IEEE/ACM Conference on Human Robot Interaction, 2006, Salt Lake City, UT.

30. Crandall, J.W. and Cummings, M.L., *Developing Performance Metrics for the Supervisory Control of Multiple Robots*. Proceedings of the 2nd ACM/IEEE International Conference on Human-Robot Interaction, 2007, Arlington, VA.

31. Scholtz, J., *Evaluation Methods for Human-System Performance of Intelligent Systems*. Proceedings of the 2002 Performance Metrics for Intelligent Systems Workshop, 2002, Gaithersburg, MD.

32. Judd, K.B., *Trajectory Planning Strategies for Unmanned Aerial Vehicles*. 2001, M. S. Thesis, Brigham Young University, Provo, UT.

33. McGrew, J.S., How, J.P., Bush, L.A., Williams, B. and Roy, N., *Air Combat Strategy using Approximate Dynamic Programming*. AIAA Guidance, Navigation, and Control Conference, 2008, Honolulu, HI.

34. Kaber, D.B. and Endlsey, M.R., "The Effects of Level of Automation and Adaptive Automation on Human Performance, Situation Awareness and Workload in a Dynamic Control Task." *Theoretical Issues in Ergonomics Science*, 2004. **5**(2): p. 113-153.

35. Cummings, M.L. and Mitchell, P.J., "Predicting Controller Capacity in Supervisory Control of Multiple UAVs." *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 2008. **38**(2): p. 451-460.

36. Eun, Y. and Bang, H., "Cooperative Control of Multiple Unmanned Aerial Vehicles Using the Potential Field Theory." *Journal of Aircraft*, 2006. **43**(6): p. 1805-1813.

37. Caves, A., *Human-Automation Collaborative RRT for UAV Mission Path Planning*. 2010, M. Eng. Thesis, Massachusetts Institute of Technology, Cambridge, MA.

38. Shima, T. and Schumacher, C., *Assignment of Cooperating UAVs to Simultaneous Tasks Using Genetic Algorithms*. Proceedings of the AIAA Guidance, Navigation and Control Conference and Exhibit, 2005, San Francisco, CA.

39. Squire, P., Trafton, G. and Parasuraman, R., *Human Control of Multiple Unmanned Vehicles: Effects of Interface Type on Execution and Task Switching Times*. Proceedings of the Conference on Human-Robot Interaction, 2006, Salt Lake City, UT.

40. Marquez, J.J., *Human-Automation Collaboration: Decision Support for Lunar and Planetary Exploration*. 2007, Ph. D. Thesis, Massachusetts Institute of Technology, Cambridge, MA.

41. Ganapathy, S., *Human-Centered Time-Pressured Decision Making in Dynamic Complex Systems*. 2006, Ph. D. Thesis, Wright State University, Dayton, OH.

42. Ruff, H.A., Narayanan, S. and Draper, M.H., "Human Interaction with Levels of Automation and Decision-Aid Fidelity in the Supervisory Control of Multiple Simulated Unmanned Air Vehicles." *Presence*, 2002. **11**(4): p. 335-351.

43. Patek, S.D., Logan, D.A. and Castanon, D.A., "Approximate Dynamic Programming for the Solution of Multiplatform Path Planning Problems." *IEEE International Conference on Systems, Man, and Cybernetics*, 1999. **1**: p. 1061-1066.

44. Weinstein, A.L. and Schumacher, C., *UAV Scheduling via the Vehicle Routing Problem with Time Windows*. Proceedings of the AIAA Infotech@Aerospace Conference, 2007, Rohnert Park, CA.

45. Banerjee, A.G., Ono, M., Roy, N. and Williams, B., *Regression-based LP Solver for Chance Constrained Finite Horizon Optimal Control with Nonconvex Constraints*. Proceedings of the American Control Conference, 2011, San Francisco, CA (in review).

46. Russell, S. and Norvig, P., *Artificial Intelligence: A Modern Approach*. 2003, Upper Saddle River, NJ: Prentice Hall.

47. LaValle, S.M., *Planning Algorithms*. 2006: Cambridge University Press.

48. Van Den Berg, J., Ferguson, D. and Kuffner, J., *Anytime Path Planning and Replanning in Dynamic Environments*. Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), 2006, Orlando, FL.

49. How, J.P., Fraser, C., Kulling, K.C., Bertuccelli, L.F., Toupet, O., Brunet, L., Bachrach, A. and Roy, N., "Increasing Autonomy of UAVs." *IEEE Robotics & Automation Magazine*, 2009. **16**(2): p. 43-51.

50. Bitton, E. and Goldberg, K., *Hydra: A Framework and Algorithms for Mixed-Initiative UAV-Assisted Search and Rescue*. IEEE International Conference on Automation Science, 2008, Washington, D.C.

51. Michini, B. and How, J.P., *A Human-Interactive Course of Action Planner for Aircraft Carrier Deck Operations*. AIAA Infotech@Aerospace Conference, 2011, St. Louis, MO.

52. Klau, G.W., Lesh, N., Marks, J. and Mitzenmacher, M., *Human-Guided Search: Survey and Recent Results.* Mitsubishi Electric Research Laboratories, Cambridge, MA, 2003

53. Klau, G.W., Lesh, N., Marks, J., Mitzenmacher, M. and Schafer, G.T., *The HuGS platform: a Toolkit for Interactive Optimization*. Proceedings of the Advanced Visual Interfaces Conference, 2002, Trento, Italy.

54. Klau, G.W., Lesh, N., Marks, J.W. and Mitzenmacher, M., *Human-Guided Tabu Search*. Proceedings of the 18th National Conference on Artificial Intelligence, 2002, Edmonton, Alberta, Canada.

55. Howe, A.E., Whitley, L.D., Barbulescu, L. and Watson, J.P., *Mixed Initiative Scheduling for the Air Force Satellite Control Network*. 2nd International NASA Workshop on Planning and Scheduling for Space, 2000, San Francisco, CA.

56. Stentz, A., *Optimal and Efficient Path Planning for Partially-known Environments*. Proceedings of the IEEE International Conference on Robotics and Automation, 1994, San Diego, CA.

57. Zilberstein, S., "Resource-bounded Sensing and Planning in Autonomous Systems." *Autonomous Robots*, 1996. **3**: p. 31-48.

58. Zilberstein, S., Charpillet, F. and Chassaing, P., *Real-time Problem-solving with Contract Algorithms.* Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence, 1999, Stockholm, Sweden.

59. Joslin, D.E. and Clements, D.P., "'Squeaky Wheel' Optimization." *Journal of Artificial Intelligence Research*, 1999. **10**: p. 353-373.

60. Pearl, J., *Heuristics: Intelligent Search Strategies for Computer Problem Solving.* 1984: Addison-Wesley.

61. Anderson, D., Anderson, E., Lesh, N., Marks, J., Mirtich, B., Ratajczak, D. and Ryall, K., *Human-Guided Simple Search*. Proceedings of the AAAI 2000 Conference, 2000, Austin, TX.

62. Anderson, D., Anderson, E., Lesh, N., Marks, J., Perlin, K., Ratajczak, D. and Ryall, K., *Human-Guided Simple Search: Combining Information Visualization and Heuristic Search*. Proceedings of the Workshop on New Paradigms in Information Visualization and Manipulation, 1999, Kansas City, MO.

63. Simon, H.A., "Rational Choice and the Structure of the Environment." *Psychological Review*, 1956. **63**(2): p. 129-138.

64. Sheridan, T.B. and Verplank, W., *Human and Computer Control of Undersea Teleoperators.* Man-Machine Systems Laboratory, Department of Mechanical Engineering, MIT, Cambridge, MA, 1978

65. Karaman, S. and Frazzoli, E., *Sampling-based Motion Planning with Deterministic μ-calculus Specifications*. Proceedings of the IEEE Conference on Decision and Control, 2009, Shanghai, China.

66. Bertuccelli, L.F., Choi, H., Cho, P. and How, J.P., *Real-time Multi-UAV Task Assignment in Dynamic and Uncertain Environments*. AIAA Guidance, Navigation, and Control Conference, 2009, Chicago, IL.

67. van Veldhuizen, D.A. and Lamont, G.B., *On Measuring Multiobjective Evolutionary Algorithm Performance*, in *Congress on Evolutionary Computation (CEC 2000)*, A. Zalzala and R. Eberhart, Editors. 2000, IEEE Press: Piscataway, NJ. p. 204-211.

68. Mahadevan, S. and Narayanan, S., *Handling Real-Time Scheduling Exceptions Using Decision Support Systems*. IEEE International Conference on Systems, Man, and Cybernetics, 2003, Washington D.C.

69. Arslan, O. and Inalhan, G., *An Event Driven Decision Support Algorithm for Command and Control of UAV Fleets*. American Control Conference, 2009, St. Louis, MO.

70. Xue, F., Sanderson, A.C. and Graves, R.J., "Multiobjective Evolutionary Decision Support for Design-Supplier-Manufacturing Planning." *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 2009. **39**(2): p. 309-320.

71. Bailey, B.P., Konstan, J.A. and Carlis, J.V., *The Effects of Interruptions on Task Performance, Annoyance, and Anxiety in the User Interface*. Proceedings of the IFIP International Conference on Human Computer Interaction, 2001, Tokyo, Japan.

72. Wharton, C., Rieman, J., Lewis, C. and Polson, P., *The Cognitive Walkthrough Method: A Practitioner's Guide*, in *Usability Inspection Methods*, J. Nielsen and R. Mack, Editors. 1994, John Wiley & Sons: New York.

73. Ericsson, K.A. and Simon, H.A., *Protocol Analysis: Verbal Reports as Data*. 1985, Cambridge, MA: MIT Press.

74. Drury, J.L., Scholtz, J. and Yanco, H.A., *Applying CSCW and HCI Techniques to Human-Robot Interaction*. Proceedings of the CHI 2004 Workshop on Shaping Human-Robot Interaction, 2004, Vienna, Austria.

75. Scholtz, J., Young, J., Drury, J.L. and Yanco, H.A., *Evaluation of Human-Robot Interaction Awareness in Search and Rescue*. Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), 2004, New Orleans, LA.

76. Yanco, H.A., Drury, J.L. and Scholtz, J., "Beyond Usability Evaluation: Analysis of Human-Robot Interaction at a Major Robotics Competition." *Journal of Human-Computer Interaction*, 2004. **19**(1 & 2): p. 117-149.

77. Hilbert, D.M. and Redmiles, D.F., "Extracting Usability Information from User Interface Events." *ACM Computing Surveys*, 2000. **32**: p. 384-421.

78. Goodrich, M.A. and Olsen, D.R., *Seven Principles of Efficient Human Robot Interaction*. Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, 2003, Washington, D.C.

79. Crandall, J.W. and Cummings, M.L., "Identifying Predictive Metrics for Supervisory Control of Multiple Robots." *IEEE Transactions on Robotics - Special Issue on Human-Robot Interaction*, 2007. **23**(5): p. 942-951.

80. Crandall, J.W. and Cummings, M.L., *Attention Allocation Efficiency in Human-UV Teams*. AIAA Infotech@Aerospace Conference, 2007, Rohnert Park, CA.

81. Vovk, V., Gammerman, A. and Shafer, G., *Algorithmic Learning in a Random World*. 2005: Springer-Verlag.

82. Tsuda, K., Rätsch, G. and Warmuth, M.K., "Matrix Exponentiated Gradient Updates for On-Line Learning and Bragman Projection." *Journal of Machine Learning Research*, 2005. **6**: p. 995-1018.

83. Bottou, L. and Le Cun, Y., "On-line Learning for Very Large Data Sets." *Research Articles, Applied Stochastic Models in Business and Industry*, 2005. **21**(2): p. 137-151.

84. Baah, G.K., Gray, A. and Harrold, M.J., *On-line Anomaly Detection of Deployed Software: A Statistical Machine Learning Approach*. Third International Workshop on Software Quality Assurance, 2006, Portland, OR.

85. Talluer, D.A. and Wickens, C.D., *The Effect of Pilot Visual Scanning Strategies on Traffic Detection Accuracy and Aircraft Control*. Proceedings of the Proceedings of the 12th International Symposium on Aviation Pscyhology, 2003, Dayton, OH.

86. Marshall, S.P., "Identifying Cognitive State from Eye Metrics." *Aviation, Space, and Environmental Medicine*, 2007. **78**(5, Supplement): p. B165-175.

87. Boren, M.T. and Ramey, J., "Thinking Aloud: Reconciling Theory and Practice." *IEEE Transactions on Professional Communication*, 2000. **43**(3): p. 261-278.

88. Purtee, M.D., Gluck, K.A., Krusmark, M.A. and Kotte, S.A., *Verbal Protocol Analysis for Validation of UAV Operator Model*. Proceedings of the Interservice/Industry Training, Simulation, and Education Conference, 2003, Orlando, FL: National Defense Industrial Association.

89. Schmidt, D.K., "A Queuing Analysis of the Air Traffic Controller's Workload." *IEEE Transactions on Systems, Man, and Cybernetics*, 1978. **8**(6): p. 492-498.

90. Cummings, M.L. and Nehme, C., *Modeling the Impact of Workload in Network Centric Supervisory Control Settings*, in *Neurocognitive and Physiological Factors During High-Tempo Operations*, S. Kornguth, Editor. 2010, Ashgate Publishing. p. 23-40.

91. Farley, T.C., Hansman, R.J., Endsley, M.R., Amonlirdviman, K. and Vigeant-Langlois, L., *The Effect of Shared Information on Pilot/Controller Situation Awareness and Re-Route Negotiation*. 2nd USA/Europe Air Traffic Management Research and Development Seminar, 1998, Orlando, FL.

92. Cummings, M.L. and Brzezinski, A.S., "Global vs. Local Decision Support for Multiple Independent UAV Schedule Management." *International Journal of Applied Decision Sciences*, 2010. **3**(3): p. 188-205.

93. Metzger, U. and Parasuraman, R., "Automation in Future Air Traffic Management: Effects of Decision Aid Reliability on Controller Performance and Mental Workload." *Human Factors*, 2005. **47**(1).

94. St. John, M., Smallman, H.S., Manes, D.I., Feher, B.A. and Morrison, J.G., "Heuristic Automation for Decluttering Tactical Displays." *Human Factors*, 2005. **47**(3): p. 509-525.

95. Cummings, M.L. and Guerlain, S., "Developing Operator Capacity Estimates for Supervisory Control of Autonomous Vehicles." *Human Factors*, 2007. **49**(1): p. 1-15.

96. Lee, J.D. and See, K.A., "Trust in Automation: Designing for Appropriate Reliance." *Human Factors*, 2003. **46**(1): p. 50-80.

97. Lee, J.D., *Trust, Self-Confidence, and Adaptation to Automation*. 1992, Ph. D. Thesis, University of Illinois, Urbana, IL.

98. Muir, B.M., "Trust in Automation. Part I. Theoretical Issues in the Study of Trust and Human Intervention." *Ergonomics*, 1994. **37**(11): p. 1905-1922.

99. Muir, B.M. and Moray, N., "Trust in Automation. Part II. Experimental Studies of Trust and Human Intervention in a Process Control Simulation." *Ergonomics*, 1996. **39**(3): p. 429-460.

100. Riley, V., *A General Model of Mixed-Initiative Human-Machine Systems*. Proceedings of the Human Factors Society 33rd Annual Meeting, 1989, Denver, CO.

101. Wickens, C.D. and Xu, X., *Automation Trust, Reliability, and Attention* HMI 02 03, AHFD-02-14/MAAD-02-2, AHDF Tehcnical Report, Savoy, IL, 2002

102. Muir, B.M., *Operators' Trust in and Percentage of Time Spent Using the Automatic Controllers in a Supervisory Control Task*. 1989, Ph. D. Thesis, University of Toronto, Toronto, Ontario, Canada.

103. Lee, J.D. and Moray, N., "Trust, Self-Confidence, and Operators' Adaptation to Automation." *International Journal of Human-Computer Studies*, 1994. **40**(1): p. 153-184.

104. Lee, J.D. and Moray, N., "Trust, Control Strategies and Allocation of Functions in Human-Machine Systems." *Ergonomics*, 1992. **35**(10): p. 1234-1270.

105. Rouse, W.B. and Morris, N.M., "On Looking into the Black Box: Prospects and Limits in the Search for Mental Models." *Psychological Bulletin*, 1986. **100**: p. 349-363.

106. Besnard, D., Greathead, D. and Baxter, G., "When Mental Models Go Wrong: Co-occurrences in Dynamic, Critical Systems." *International Journal of Human-Computer Studies*, 2004. **60**: p. 117-128.

107. Moray, N., *A Taxonomy and Theory of Mental Models*. Proceedings of the Human Factors and Ergonomics Society 40th Annual Meeting, 1996, Philadelphia, PA.

108. Bolstad, C.A. and Endsley, M.R., *The Effect of Task Load and Shared Displays on Team Situation Awareness*. Proceedings of the 14th Triennial Congress of the International Ergonomics Association and the 44th Annual Meeting of the Human Factors and Ergonomics Society, 2000, San Diego, CA: HFES.

109. Parasuraman, R. and Riley, V., "Humans and Automation: Use, Misuse, Disuse, Abuse." *Human Factors*, 1997. **39**(2): p. 230-253.

110. Lee, J. and Moray, N., "Trust, Control Strategies and Allocation of Function in Human-Machine Systems." *Ergonomics*, 1992. **35**(10): p. 1243-1270.

111. Bruni, S., Marquez, J.J., Brzezinski, A., Nehme, C. and Boussemart, Y., *Introducing a Human-Automation Collaboration Taxonomy (HACT) in Command and Control Decision-Support Systems* Proceedings of the International Command and Control Research and Technology Symposium, 2007, Newport, RI.

112. Cummings, M.L. and Bruni, S., *Collaborative Human-Automation Decision Making*, in *Springer Handbook of Automation*, S.Y. Nof, Editor. 2009, Spring Berlin Heidelberg: Berlin, Germany. p. 437-447.

113. Nehme, C.E., Scott, S.D., Cummings, M.L. and Furusho, C.Y., *Generating Requirements for Futuristic Heterogeneous Unmanned Systems*. Proceedings of the Human Factors and Ergonomics Society, 2006, San Fransisco, CA.

114. Bennett, K.B. and Flach, J.M., "Graphical Displays: Implications for Divided Attention, Focused Attention, and Problem Solving." *Human Factors*, 1992. **34**(5): p. 513-533.

115. Wickens, C.D. and Carswell, C.M., "The Proximity Compatibility Principle: Its Psychological Foundation and Relevance to Display Design." *Human Factors*, 1995. **37**(3): p. 473-494.

116. Gibson, J.J., *The Ecological Approach to Visual Perception*. 1979, Boston, MA: Houghton Mifflin.

117. Rochlin, G.I., La Porte, T.R. and Roberts, K.H., "The Self-Designing High-Reliability Organization: Aircraft Carrier Flight Operations at Sea." *Naval War College Review*, 1987(42): p. 76-90.

118. Tversky, A. and Kahneman, D., "Judgment under Uncertainty: Heuristics and Biases." *Science*, 1974. **185**(4157): p. 1124-1131.

119. Simon, H.A., Dantzig, G.B., Hogarth, R., Plott, C.R., Raiffa, H., Schelling, T.C., Shepsle, K.A., Thaler, R., Tversky, A. and Winter, S., "Decision Making and Problem Solving." *Interfaces*, 1987. **17**(5): p. 11-31.

120. Spradley, J.P., *The Ethnographic Interview*. 1979, New York, NY: Holt, Reinhart, and Winston.

121. Schlindwein, S.L. and Ison, R., "Human Knowing and Perceived Complexity: Implications for Systems Practice." *Emergence: Complexity and Organization*, 2004. **6**(3): p. 27-32.

122. Cilliers, P., *Complexity and Postmodernism: Understanding Complex Systems*. 1998, London and New York: Routledge.

123. Rescher, N., *Complexity: A Philosophical Overview*. 1998, New Brunswick, NJ: Transaction Publishers.

124. Lemoigne, J.L., *La Modelisation des Systems Complexes [The Modeling of Complex Systems]*. 1995, Paris, France: Dunod.

125. Casti, J.L., *Complexification*. 1995, New York, New York: HarperCollins.

126. Martinez, M.E., "The Process of Knowing: A Biocognitive Epistemology." *The Journal of Mind and Behavior*, 2001. **22**(4): p. 407-426.

127. Bondi, A.B., *Characteristics of Scalability and Their Impact on Performance*. Proceedings of the 2nd International Workshop on Software and Performance, 2000, Ottawa, Ontario, Canada.

128. Donmez, B., Nehme, C. and Cummings, M.L., "Modeling Workload Impact in Multiple Unmanned Vehicle Supervisory Control." *IEEE Transactions on Systems, Man, and Cybernetics*, 2010. **40**(6): p. 1180-1190.

129. Rouse, W.B., *Systems Engineering Models of Human-Machine Interaction*. 1983, New York: North Holland. 152.

130. Keppel, G., *Design and Analysis: A Researcher's Handbook*. 1982, Englewood Cliffs, NJ: Prentice Hall.

131. Maxwell, S.E. and Delaney, H.D., *Designing Experiments and Analyzing Data: A Model Comparison Perspective*. 2004, Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

132. Scott, S.D. and Cummings, M.L., *Cognitive Task Analysis for the LCS Operator.* MIT Humans and Automation Laboratory Cambridge, MA, 2006

133. Carrigan, G., *The Design of an Intelligent Decision Support Tool for Submarine Commanders*. 2009, S. M. Thesis, Massachusetts Institute of Technology, Cambridge, MA.

134. Tappan, J.M., Pitman, D.J., Cummings, M.L. and Miglianico, D., *Display Requirements for an Interactive Rail Scheduling Display*. HCI International Conference, 2011, Orlando, FL.

135. Rasmussen, J., Pejtersen, A. and Goodstein, L., *Cognitive Systems Engineering*. 1994, New York, NY: John Wiley & Sons, Inc.

136. Rasmussen, J., "Skills, Rules, and Knowledge: Signals, Signs, and Symbols, and Other Distinctions in Human Performance Models." *IEEE Transactions on Systems, Man, and Cybernetics*, 1983. **13**(3): p. 257-266.

137. Cummings, M.L. and Mitchell, P.M., *Managing Multiple UAVs through a Timeline Display*. AIAA InfoTech@Aerospace, 2005, Arlington, VA.

138. Brzezinski, A.S., *StarVis: A Configural Decision Support Tool for Schedule Management for Multiple Unmanned Aerial Vehicles*. 2008, S. M. Thesis, Massachusetts Institute of Technology, Cambridge, MA.

139. Graham, H.D., Coppin, G. and Cummings, M.L., *The PRIM: Extracting Expert Knowledge for Aiding in C2 Sense and Decision Making*. Proceedings of the 12th International Command and Control Research and Technology Symposium, 2007, Newport, RI.