



**A STUDY OF HUMAN RELIANCE ON
IMPERFECT AUTOMATION**

THESIS

Christopher J. Garnick, 2d Lt, USAF
AFIT-ENG-MS-17-M-030

**DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY**

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

DISTRIBUTION STATEMENT A
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

The views expressed in this document are those of the author and do not reflect the official policy or position of the United States Air Force, the United States Department of Defense or the United States Government. This material is declared a work of the U.S. Government and is not subject to copyright protection in the United States.

AFIT-ENG-MS-17-M-030

A STUDY OF HUMAN RELIANCE
ON IMPERFECT AUTOMATION

THESIS

Presented to the Faculty
Department of Electrical and Computer Engineering
Graduate School of Engineering and Management
Air Force Institute of Technology
Air University
Air Education and Training Command
in Partial Fulfillment of the Requirements for the
Degree of Masters of Science in Computer Engineering

Christopher J. Garnick, B.S.C.E.

2d Lt, USAF

March 2017

DISTRIBUTION STATEMENT A
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

AFIT-ENG-MS-17-M-030

A STUDY OF HUMAN RELIANCE
ON IMPERFECT AUTOMATION

THESIS

Christopher J. Garnick, B.S.C.E.
2d Lt, USAF

Committee Membership:

Maj J. M. Bindewald, PhD
Chair

Dr B. J. Borghetti, PhD
Member

Maj C. F. Rusnock, PhD
Member

Abstract

Appropriate trust in human-machine teams is critical in ensuring proper utilization of automation. Without appropriate trust, human operators may disuse or misuse automation resulting in loss of capability, efficiency, and safety. Designing automation that ensures appropriate trust remains a difficult problem, as it requires a means to measure and calibrate trust. This research investigates trust calibration by examining the relationship between human reliance on automation as automation reliability varies.

In order to accomplish this goal, a human-machine system was developed, where automation performs a reliance-based task well enough that it is logical for human teammates to rely upon it. The human-machine team was then utilized in human-subject experimentation, which provided trust-based data gathered from subjective surveys and objective observations of human behavior.

Results from the experiment show that after experiencing reduced reliability rates, operators become better calibrated to the automation's capabilities, trust the automation more, and rely upon it more appropriately. In conclusion, these results suggest that exposing human-machine team operators to reduced reliability rates in training allow them to calibrate to the automation's capability and limitations, thus positively influencing trust calibration and reducing the likelihood of automation disuse or misuse.

Table of Contents

	Page
Abstract	iv
List of Figures	viii
List of Tables	x
I. Introduction	1
1.1 Motivation	1
1.2 Factors of Trust Calibration	3
1.3 Human Behaviors of Trust	4
1.4 Appropriate Trust	5
1.5 Research Objectives	6
1.6 Thesis Overview	8
II. Background	9
2.1 Application Environment	9
2.1.1 Space Navigator Tasks	10
2.1.2 Space Navigator Agents	10
2.2 Agent Design	11
2.2.1 Agent Structure: An Artificial Intelligence Perspective	11
2.2.2 Function Allocation	13
2.3 Collision Detection and Avoidance	19
2.3.1 Components of a Real-World Collision Avoidance System	19
2.3.2 Collision Detection in a Simulated Environment	20
2.4 Trust in Human Machine Teams	23
2.4.1 Appropriate Trust	23
2.4.2 Factors Influencing Trust	25
2.4.3 Measures of Trust	26
2.5 Chapter Summary	29
III. Methodology	30
3.1 Agent Design	30
3.2 Collision Detection	32
3.2.1 Detection Performance	34
3.3 Collision Avoidance	36
3.3.1 The Step-Back Route	37
3.3.2 The Least Congested Route	38
3.3.3 The Pseudo-Random Route	41

	Page
3.4 Evaluation of Collision Avoidance Agents	43
3.4.1 Performance Without Collision Avoidance Agent	44
3.4.2 Performance With Collision Avoidance Agent	44
3.4.3 Performance Overall	45
3.5 Reduced Agent Reliability	47
3.5.1 Reduced Reliability Process	47
3.5.2 Reduced Reliability Evaluation	48
3.6 Human-Subject Experiment	49
3.6.1 Experiment Data	49
3.6.2 Experiment Design	54
3.7 Chapter Summary	57
IV. Results	58
4.1 Subjective Reliability vs. Programmed Reliability	58
4.2 Subjective Trust vs. Programmed Reliability	63
4.3 Reliance Rate vs. Programmed Reliability	65
4.4 Team Performance	67
4.5 Correlations and Analysis of Variance	68
4.5.1 Subjective Reliability	69
4.5.2 Z-Scores of Subjective Reliability	71
4.5.3 Subjective Trust	72
4.5.4 Reliance Rate	74
4.6 Chapter Summary	77
V. Conclusion	78
5.1 Research Question	78
5.2 Research Accomplishments	79
5.3 Research Summary	80
5.4 Future Work	81
Appendix A. Pre-Experiment Questionnaire	83
1.1 Demographics Questionnaire	83
1.2 General Trust Questionnaires	83
1.2.1 Propensity to Trust Automation	84
1.2.2 Propensity to Trust Humans	84
Appendix B. Post-Experiment Questionnaire	86
2.1 Experience with <i>Space Navigator</i> Questionnaire	86
2.2 Background Questionnaires	87
2.2.1 Electronic Background Questionnaire	87
2.2.2 Gaming Background Questionnaire	88

	Page
Appendix C. Objective In-Game Data Collected	89
Appendix D. Informed Consent Document	90
Appendix E. Training Information	94
5.1 Training Slides	94
5.2 Training Script	113
Bibliography	121

List of Figures

Figure		Page
1	Calibration of trust vs. system reliability [16]	2
2	A screen capture of <i>Space Navigator</i> showing all of its components.	10
3	Performance dependence chart showing the relationship between human demand and system performance [5].	15
4	Left: Example of spatio-temporal interaction (top) and swept volume interface (bottom) collision detection methods, where two moving objects O_1 and O_2 collide at time t_c . Right: Example of spatio-temporal interaction (top) and swept volume interface (bottom) collision detection methods, where two moving objects O_1 and O_2 do not collide [25].	22
5	The relationship between trust and capability [27].	24
6	A behavior diagram [42] outlining the functionality of the CA agent.	32
7	Collision table used by the collision avoidance agent to organize predicted collisions, where $x \leq y \leq z$	34
8	Space Navigator represented in a 5X10 grid.	38
9	Average number of collisions with 95% confidence interval across 108 five-minute games for each examined set of players.	46
10	Average score with 95% confidence interval across 108 five-minute games for each examined set of players.	46
11	The resulting mean reliability rates from 48 games of 5 minutes at each programmed reliability rate.	49
12	A screen-shot of the reliability question within <i>Space Navigator</i>	51
13	A screen-shot of the trust question within <i>Space Navigator</i>	51

Figure		Page
14	Actual reliability rate and subjective reliability ratings across each programmed reliability rate.	60
15	Z-Scores of subjective reliability ratings across each programmed reliability rate.....	62
16	Human-subject experiment results of the subjective trust across each programmed reliability rate.	64
17	Human-subject experiment results of the reliance rate across each programmed reliability rate.	66
18	Average scores across each programmed reliability rate.	68

List of Tables

Table		Page
1	The human behaviors of reliance and compliance with their associated actions that exhibit trust or distrust in automation.	5
2	The original Fitts list outlining the aptitude of humans and machines [13].	14
3	Levels of automation for decision and action selection [34].	17
4	A Likert-type scale to capture a human’s subjective trust in automation.	27
5	The confusion matrix of collision detection in 60 minutes of game-play.	35
6	Collision detection performance metrics from 60 minutes of game-play.	36
7	Scenarios providing performance metrics analyzed in the evaluation experiment.	43
8	Average baseline performance results across 108 five-minute games with the human and/or line agent.	44
9	Average collision avoidance performance results across 108 five-minute games with the line agent and each of the collision avoidance methods.	45
10	The resulting mean reliability rates and confidence intervals from 48 games of 5 minutes at each programmed reliability rate.	48
11	Each user’s reliability rate order for blocks 2-5.	56
12	Human-subject experiment results of the actual and subjective reliability across each programmed reliability rate.	59
13	Results from the one-way ANOVA of subjective reliability by participant. (* denotes significance at the 0.01 level)	60

Table		Page
14	Human-subject experiment results of the subjective reliability and corresponding z-score across each programmed reliability rate.....	62
15	Human-subject experiment results of the subjective trust across each programmed reliability rate.	64
16	Human-subject experiment results of the reliance rate across each programmed reliability rate.	66
17	Human-subject experiment results of the average scores across each programmed reliability rate.	67
18	List of significant correlations with subjective reliability ratings. (** denotes significance at the 0.01 level, * denotes significance at the 0.05 level)	70
19	ANOVA results for subjective reliability. (* denotes significance at the 0.05 level)	70
20	Effect tests from the subjective reliability ANOVA. (* denotes significance at the 0.05 level)	71
21	List of significant correlations with z-scores for subjective reliability ratings. (** denotes significance at the 0.01 level, * denotes significance at the 0.05 level)	72
22	ANOVA results for Z-Scores of subjective reliability ratings. (* denotes significance at the 0.05 level)	72
23	List of significant correlations with subjective trust ratings. (** denotes significance at the 0.01 level, * denotes significance at the 0.05 level)	73
24	ANOVA results for subjective trust. (* denotes significance at the 0.05 level)	74
25	Effect tests from the subjective trust ANOVA. (* denotes significance at the 0.05 level)	74
26	List of significant correlations with reliance rates. (** denotes significance at the 0.01 level, * denotes significance at the 0.05 level)	75

Table		Page
27	ANOVA results for reliance rate. (* denotes significance at the 0.05 level)	76
28	Effect tests from the reliance rate ANOVA. (* denotes significance at the 0.05 level)	76
29	Demographics questionnaire filled out prior to the experiment for each participant.	83
30	General trust questionnaire used to gather a player's propensity to trust in automation.	84
31	General trust questionnaire used to gather a player's propensity to trust humans.	84
32	Space Navigator experience questionnaire capturing the participant's feelings toward the game and automation.	86
33	Electronic experience questionnaire capturing the participant's average usage of electronics.	87
34	Gaming experience questionnaire capturing the participant's experience with various types of games.	88
35	All objective events recorded during a game of <i>Space Navigator</i>	89

A STUDY OF HUMAN RELIANCE ON IMPERFECT AUTOMATION

I. Introduction

1.1 Motivation

In June 2015, the United States Air Force released *Autonomous Horizons* [16], a road-map for future autonomous systems. Their plan is to take advantage of the major potential of automation such as the reduction of costs and human error, and the increase in capability and reliability. These advantages apply in unmanned systems; intelligence, reconnaissance, and surveillance (ISR); and space and cyber operations allow the future Air Force to maintain Global Vigilance, Global Reach, and Global Power. Autonomy handling warfare applications alone brings about many concerns, such as:

- How should automation handle unforeseen situations [32]?
- Should we allow fully autonomous weapon systems to target humans [37]?

One Air Force solution to these questions is to place airmen in the loop, working harmoniously with automation to create a synergistic human-machine team. In April 2016, the Deputy Secretary of Defense Robert Work alluded to human-machine teaming in his description of the “Third Offset”, conveying the importance of this concept [46]. Human-machine teams aim to combine the strengths of automation and airmen to reduce each party’s weaknesses. However, developing automation for the desired team is easier said than done. Using four decades of experience and research

in the field [16], *Autonomous Horizons* lays out five major challenges that need to be solved in order to achieve the envisioned human-machine team. One of five major challenges outlined by *Autonomous Horizons* involves “developing a level of trust that is appropriately calibrated to the reliability and functionality of the system in various circumstances” [16].

In human-machine teams, trust directly influences the level that a human relies on a machine [27, 35]. In order to obtain the benefits of human-machine teams, the human’s level of trust must appropriately match the machine’s level of reliability as seen in Figure 1. When humans perceive the machine to be unreliable, they display low levels of trust. Improper under-trusting may lead to failures from ignoring or turning off automation (disuse). A real-world example of under-trusting occurred with the initial low-acceptance rate of the Air Force’s Automatic Ground Collision Avoidance System [33], which proved its effectiveness by saving four pilot lives to date [26].

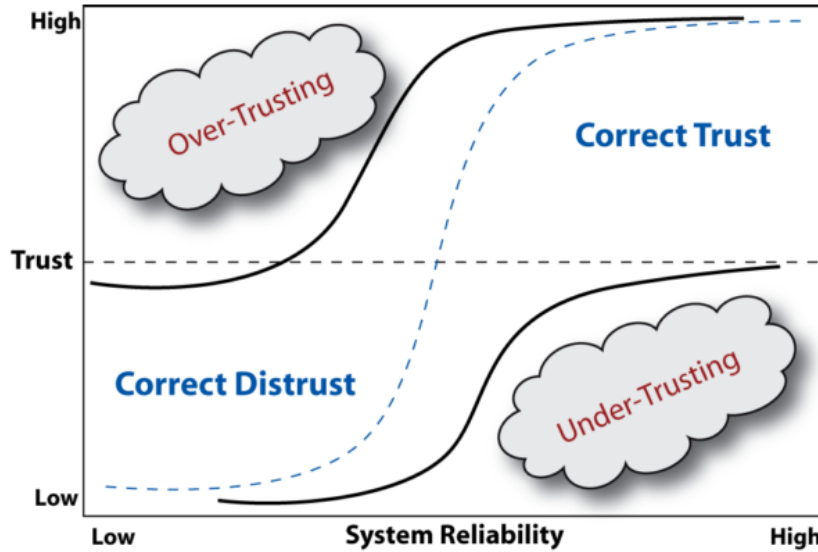


Figure 1. Calibration of trust vs. system reliability [16]

On the opposite end, when humans perceive the machine to be highly reliable,

they show high levels of trust. Improper over-trusting in the machine may result in failures from overestimating automation capabilities (misuse). The Tesla S crash of 2016 [32] exemplifies a recent case of improper over-trusting in automation. In this human-machine team, the operator over-relied on the autopilot’s capability, which did not recognize the tractor trailer turning in front of it at an intersection. As a result, the recognition failure led to a devastating crash and the operator’s death. In areas such as vulnerability detection in critical infrastructure [7] and operation of semi-autonomous aircraft [30], improper trust in automation may lead to a weakened national security posture. These situations illustrate the Department of Defense and commercial-driven need for human trust to be calibrated to an automation’s true capabilities [27].

1.2 Factors of Trust Calibration

Identifying the causes of under- and over-trusting, lies within factors that influence trust calibration. *Autonomous Horizons* classifies these factors into three separate categories: system, individual, and situational [16].

- *System Factors* involve the role and performance of the automation. This group contains features such as machine understandability, predictability, overall reliability, the operator’s perception of machine reliability, and situational reliability.
- *Individual Factors* relate directly to the operator. These factors include the operator’s ability to perform the task, propensity to trust, and other operator-based characteristics.
- *Situational Factors* depend on the current situation of the human-machine team. Specifically, they consist of the amount of workload, effort, and attention required to complete system tasks.

Unlike individual and situational factors, which vary between operators and situations, system factors influencing trust originate from a constant source: the machine. Machine reliability plays a major role in this category. Thus, studying the relationship between machine reliability and operator trust will aid in developing automation designs that positively influence trust calibration in human-machine teams.

1.3 Human Behaviors of Trust

In order to study the relationship between automation reliability and operator trust, there must be a means to measure trust. Subjective trust measurements can be retrieved from Likert-type questionnaires that consist of self-reported trust in automation. However, subjective measures cannot be relied upon alone, as outside factors such as self-confidence often influence these results [28]. For this reason, objective trust measures could prove useful. Boubin *et al.* [4] propose a methodology to gather objective measures related to trust by breaking it down into two components of human behavior: reliance and compliance. Compliance refers to the human's acceptance of automation action, and reliance refers to the human's acceptance of automation non-action. A lack of trust is shown when the human does not find the automation's action or non-action acceptable.

Table 1 breaks down reliance and compliance into the human actions which exhibit trust or distrust. In reliance-based automation, a human displays trust when they remain passive, letting the automation perform its own task while they focus on other system tasks. Meanwhile, distrust is shown when the human preempts automation, performing the automations task on their own. In compliance-based automation, a human displays trust when they concur with or obey the automations action. Distrust is shown when the human overrides or ignores the automations action. Reliance and compliance rates provide an objective measure that relates to human-operators' trust

in the system. Operators with high reliance/compliance rates display high trust in the automation, while operators with low reliance/compliance rates display low trust in the automation.

Table 1. The human behaviors of reliance and compliance with their associated actions that exhibit trust or distrust in automation.

Human Behavior	Human Action	
	Trust	Distrust
Reliance	Passive	Preempt
Compliance	Concur/Obey	Override/Ignore

Reliance and compliance rates provide an objective measure that relates to human-operators’ trust in the system. Operators with high reliance/compliance rates display high trust in the automation, while operators with low reliance/compliance rates display low trust in the automation. Human behavior-based measurement systems enable the recording of these ratings. Such a system consists of a human-machine team, where the machine effectively performs a reliance or compliance-based action. Thus, a reliance-based measurement system involves a human-machine team where automation performs a reliance-based task well enough such that it is logical for the human to rely upon it.

1.4 Appropriate Trust

The goal behind trust calibration is to influence human operators to develop appropriate trust in automation, thus mitigating occurrences of misuse and disuse. According to Lee and See [27], appropriate trust occurs when a human’s level of trust matches the true capabilities of the autonomous system. In subjective measurements, this happens when a human’s perceived automation reliability matches the automation’s actual reliability. When looking at human behavior’s related to trust, researchers often assume that appropriate reliance or compliance occurs when a hu-

man’s reliance or compliance rate matches the automation’s actual reliability [45]. However, this is not always the case. Take for example a diagnostic-based automation with 90% reliability. If the human relies on the automation’s diagnosis 90 times out of 100, then the team correctly diagnoses 81 situations ($0.9 * 90 = 81$). In the remaining 10 cases, the human assumes the automation is incorrect and disagrees, leading to 1 correct diagnosis ($0.1 * 10 = 1$). As result, the team correctly diagnoses a total of 82 out of 100 situations. Now if the human were to rely upon the automation 100% of the time, then the team correctly diagnoses 90 of 100 situations ($0.9 * 100 = 90$). Since relying 100% of the time on automation with 90% reliability yields a better performance than relying 90% of the time, attaining appropriate trust via reliability matching does not apply to the behavioral measurements of reliance and compliance. For the purpose of this research, a novel definition of appropriate reliance/compliance is proposed to mean the level of reliance/compliance that results in the best performance.

1.5 Research Objectives

Research Problem

With the use of automation expanding in military and civilian applications, there exists a need to calibrate for appropriate trust in automation. An automation’s reliability plays a large role in trust calibration. Prior research involves studying trust-based metrics with varying reliability in alarm [14, 31, 8] and compliance-based automation [36, 8]. However, research to analyze the reliability-trust relationship in automation that performs a reliance-based action is lacking. Therefore, the primary objective of this research is to answer the following question: “Does the level of automation reliability affect reliance-based trust metrics in the human members of human-machine teams?” This characterization research aims to examine the reliability-trust rela-

tionship to gain a better understanding of how to calibrate for appropriate trust in automation.

Research Hypothesis

As a human operator gains experience with reduced reliability, he or she will become better calibrated to the automation's true capability and limitations. As a result, the operator will display calibration for appropriate trust in the automation.

Investigative Questions

It is important to look at both subjective and objective measurements associated with trust when working with a reliance-based agent. In order to better answer the research question, the following investigative questions and their hypotheses are explored:

- Investigative Question: "What is the relationship between the human's perceived automation reliability and the actual automation reliability?"

Hypothesis: After gaining experience with reduced reliability, participants will become better calibrated to the automation's true capability.

- Investigative Question: "What is the relationship between the human's perceived trust in automation and the automation reliability?"

Hypothesis: After gaining experience with reduced reliability, participants will show trust appropriately correlated to automation reliability.

- Investigative Question: "What is the relationship between the human's reliance rate in automation and the automation reliability?"

Hypothesis: After gaining experience with reduced reliability, participants will show reliance appropriately correlated to automation reliability.

Secondary Objectives

In order to confirm these hypotheses, three secondary objectives must be accomplished:

- Objective One: Development and justification of a reliance-based measurement system. Specifically, this objective deals with confirming that it is logical for a human to rely on a reliance-based agent in order to increase overall team performance within a system.
- Objective Two: Design and implementation of a human-subject study, where participants work alongside a reliance-based agent with varying reliability. This objective includes collection of subjective data via pre-, post-, and in-game questionnaires and objective data from in-game events.
- Objective Three: Perform statistical and machine learning analysis on the human-subject data. Conclusions drawn from this objective provide characterization metrics to answer the investigative and research questions.

1.6 Thesis Overview

Chapter II introduces the application environment, further explains concepts of trust in automation, and provides sufficient background on major components of the collision avoidance reliance-based agent to create a reliance-based measurement system and human-subject study. This includes topics such as agent design, collision detection, and collision avoidance. Chapter III covers the design and efficacy of the reliance-based agent and outlines the human-subject study. Statistical and machine learning analysis on the data gathered in the human-subject study allows investigation of the relationship between human reliance and automation in Chapter IV. Lastly, Chapter V outlines the major trends and relationships gathered from the data to summarize the findings of this research. This chapter concludes with advice for future work enabled by this research.

II. Background

This chapter first introduces the application environment, *Space Navigator*. It then examines types of automated agents, methods for automation design, and approaches of collision detection and avoidance needed to create a reliance-based collision avoidance agent. Finally, this chapter covers concepts of trust in human-machine teams. This information provides the background required to create the reliance-based measurement system and human-subject study, which allow data capture and analysis for the relationship between the human-behavior of reliance and automation with varying levels of reliability.

2.1 Application Environment

The *Space Navigator* environment is a tablet-based computer game specifically designed for the evaluation of human-machine teams [2]. The game provides a suitable framework to study human behaviors of trust, as it allows access to source code to further develop reliance or compliance-based autonomous agents and has the capability to capture in-game human and agent driven events. Seen in Figure 2, *Space Navigator* has four components: ships, planets, no-fly zones (NFZs), and bonuses. Every two seconds, a ship spawns at a random location off-screen, moving in a random fixed direction across the screen at a constant velocity. Each ship is colored to match one of four possible destination planets. Human-players can guide each ship to any location by pressing down on the tablet’s touch screen and drawing a trajectory.

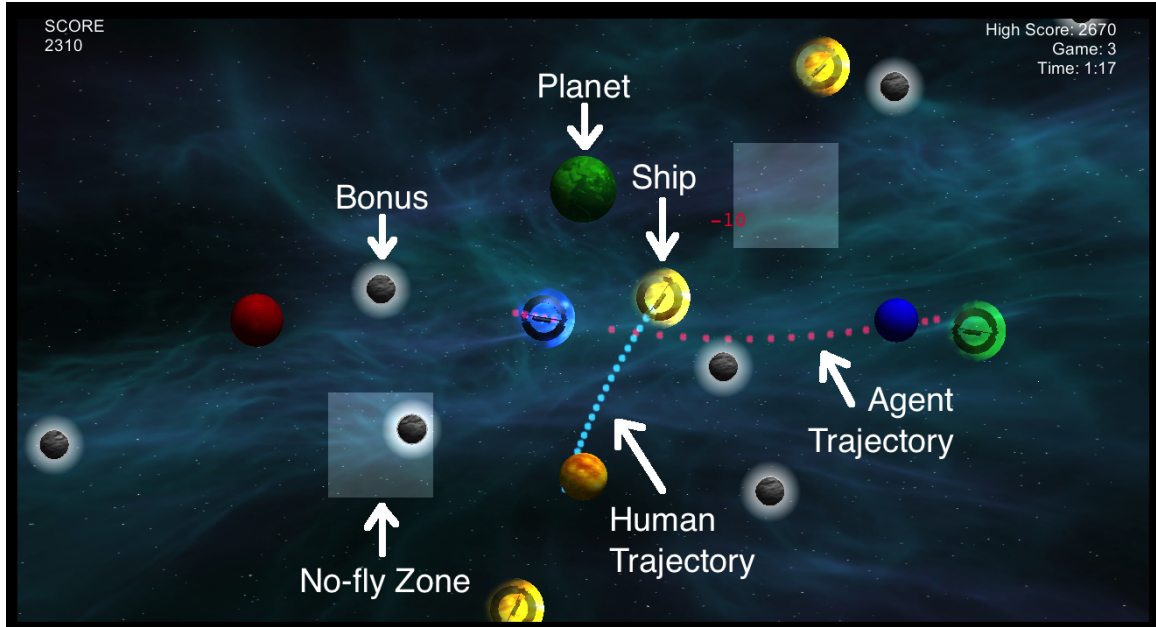


Figure 2. A screen capture of *Space Navigator* showing all of its components.

2.1.1 Space Navigator Tasks.

Space Navigator tasks the human player to achieve the highest score within 5 minutes. Points are acquired in one of two ways: +100 points for safely guiding a ship to its home planet and +50 points for guiding a ship through a bonus. Points are lost in one of two ways: -100 points per ship for colliding ships and -10 points per second for each ship flying through one of two stationary NFZs. Looking at this game from the perspective of human-machine teaming, there exist four distinct score-related tasks for either the human-player or automation to perform: routing ships to planets, routing ships through bonuses, avoiding no-fly zones, and avoiding collisions.

2.1.2 Space Navigator Agents.

Previous human-machine studies in the *Space Navigator* environment [3, 4] involved playing with a simple-reflex agent tasked to route ships to planets. This agent, known as the “line agent,” performs a compliance-based task of generating

a straight-line trajectory if a ship has not been routed within a set amount of time after its creation. Other variations of the line agent have been designed, such as one that generates similar trajectories as the human player [3]. However, no reliance-based agent exists in *Space Navigator*. In order to measure human reliance, *Space Navigator* requires an agent that executes a reliance-based task. The task of avoiding collisions is a good candidate. If players trust the autonomous “collision avoidance agent,” they rely on the agent to re-route collision-imminent ships. On the other hand, if players do not trust the agent, they re-route collision-imminent ships themselves.

2.2 Agent Design

Before beginning the development of the collision avoidance agent, research of automated agents in human machine teaming must occur. The notion of human-machine teams aims to integrate the strengths of each party, such as human decision making skills with machine computational bandwidth, in order to increase the overall performance and cost-effectiveness of a functional system [39]. Human-machine teams vary based on agent structure and functions allocated to each party.

2.2.1 Agent Structure: An Artificial Intelligence Perspective.

According to Russell and Norvig [38], an agent is “anything that can be viewed as perceiving its environment through sensors and acting upon that environment through actuators.” This broad definition applies to a variety of entities such as humans; however for the sake of clarity, this section will only use the term to identify software-based artificial intelligence (AI). In order to categorize agents, Russell and Norvig define four basic structures: simple reflex agent, model-based reflex agent, goal-based agent, and utility-based agent.

2.2.1.1 Simple Reflex Agent.

This first agent makes decisions based on a simple set of condition-action rules, which only take into account current percepts. These rules can be represented as If-Then statements. The simple reflex agent holds no information concerning previous states. As result, it is possible for the vacuum-robot to repeatedly run into a wall, move backward, and then move forward into the wall again. The next type of agent resolves this issue.

2.2.1.2 Model-Based Reflex Agent.

The model-based reflex agent builds off of the simple reflex agent by adding two capabilities. The first capability allows the agent to hold an internal state that provides information on its history or the results of previous actions. The second capability of a model-based agent involves the ability to hold some knowledge about “how the world works” via a world-model. The world-model, combined with the internal state, allows the agent to predict the result of actions given a set of percepts.

2.2.1.3 Goal-Based Agent.

The goal-based agent utilizes an internal state with the world-model in order to obtain a set of goals. This agent differs from the previous two in that it does not use a set of condition-action rules to determine an action. Instead, the goal-based agent employs search and planning algorithms to decide upon a sequence of actions which achieve the goals. Compared to simple reflex and model-based agents, the goal-based agent appears less efficient as performing reasoning algorithms take more time than choosing from a set of condition-action rules. On the other hand, the ability to quickly change the agent’s goals makes it more flexible.

2.2.1.4 Utility-Based Agent.

The utility-based agent builds off of the goal-based agent by adding a measure of “happiness” or utility to a goal’s sequence of actions. There often exists more than one solution to achieve goals; however, some solutions may be more desirable than others. In order to perform desirable actions, the utility-based agent chooses the solution which maximizes their utility.

Although the goal-based and utility-based agents often perform better than the simple reflex and model-based agents in planning-based tasks, they may not in other applications. Therefore, it is important to analyze system requirements beforehand to decide which type of agent best suits a given application.

2.2.2 Function Allocation.

Initially, automation served as a low-cost opportunity to perform functions faster, more accurately, and more reliably than humans [35]. For this reason, designers tended to automate everything possible and leave the remaining complex functions to be handled by the human operator [34]. As a result, these human-machine teams were defined by the functions automation could and could not perform. Since then, experience [5] has shown that this notion typically does not result in optimization of the whole system due to a number of unforeseen issues in human factors such as reduced situation awareness, unbalanced workload, and inappropriate trust [12]. In order to mitigate such issues, researchers developed various methodologies for function allocation in the design of automated agents. The following sections cover some of these methodologies.

2.2.2.1 The Fitts List.

In 1951, Paul Fitts released a report covering the different roles men and machines should play in future air navigation and air traffic control systems [13]. The report held a simple list, later known as the Fitts list, which still applies to function allocation in human-machine teams over 60 years later. Table 2 shows the complete, original list. In summary, the Fitts list holds 11 statements outlining strong and weak abilities of both humans and machines. The list serves as a guide in function allocation, such that a job requiring specific abilities is assigned to the party that performs it best. Although this methodology has been criticized as being outdated in recent years [13], the generalizability of the list serves as a useful starting point in the design of automated agents.

Table 2. The original Fitts list outlining the aptitude of humans and machines [13].

Humans appear to surpass present-day machines in respect to the following:	Present-day machines appear to surpass humans in respect to the following:
1. Ability to detect a small amount of visual or acoustic energy	1. Ability to respond quickly to control signals and to apply great force smoothly and precisely
2. Ability to perceive patterns of light or sound	2. Ability to perform repetitive, routine tasks
3. Ability to improvise and use flexible procedures	3. Ability to store information briefly and then to erase it completely
4. Ability to store very large amounts of information for long periods and to recall relevant facts at the appropriate time	4. Ability to reason deductively, including computational ability
5. Ability to reason inductively	5. Ability to handle highly complex operations, i.e. to do many different things at once
6. Ability to exercise judgment	

2.2.2.2 The Complementarity Principle.

While Fitts list provides an adequate outline for functional allocation, it assumes that functional responsibility remains static [5]. This quality appears insufficient today, as many human-machine teams involve varying levels in automation and require the human operator to handle situations of machine failure. The static environment proposed in Fitts list results in unwanted qualities such as an operator’s lack of skill

level or situation awareness [5]. The *complementarity principle* [21] uses a combination of human factors, cognitive systems engineering, and control theory to prevent these concerns by keeping the human operator constantly informed. The idea is to maintain an optimal socio-technical system by focusing on balancing the workload between human and machine.

The complementarity principle identifies that success in many human-machine teams depends entirely upon human performance. Figure 3 shows the performance dependence trend that occurs in a human-machine team. The left vertical axis displays the human-machine team performance, while the right vertical axis displays the human performance demands or workload. The horizontal axis covers the continuum of situations, which can be broken down into three states: data limited state, resource matching state, and resource limited state.

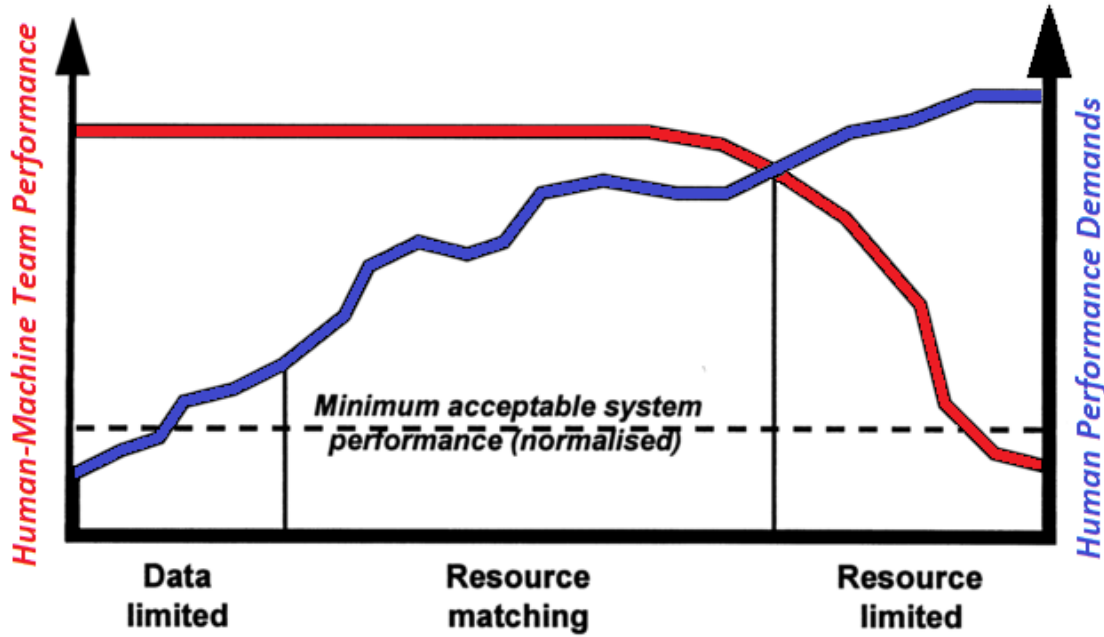


Figure 3. Performance dependence chart showing the relationship between human demand and system performance [5].

In the *data limited state*, automation performs the entirety of the task and requires little assistance from the operator. During this state, the human is engaged in the

task of overseeing the system. The joint system performance is maximized; however, the data limited state can lead to harmful situations of human skill-degradation and the lack of situation awareness due to low human workload.

The *resource limited state* covers the opposite situation where demands on the human operator exceed their capability. This state can occur when the operator sustains an increased workload over a long period or in the case of a sudden task transition from data limited state. In the resource limited state, the joint system performance falters greatly.

The *resource matching state* is ideal in the complementarity principle. In this state, the human experiences a moderate amount of workload in their operation. Their workload can range from maintaining situation awareness with appropriate machine feedback to preventing skill-degradation by making control decisions. By working in the resource matching state, humans can properly handle task transitions or situations when the machine fails.

2.2.2.3 Human-Centered Automation Design.

The next allocation methodology relates to the complementarity principle in that its automation design focuses on human factors. However, the *human-centered methodology* provides more detail in deciding the level of automation in the process of agent design. In order to describe their methodology, Parasuraman *et al.* [34] gives us four classes of autonomous application: Information Acquisition, Information Analysis, Decision and Action Selection, and Action Implementation.

The *information acquisition* class consists of sensing and recording input data. It can include functions required to gather data, such as positioning a camera for visual input; and operations of organizing information, such as preprocessing, sorting, and prioritizing.

- The *information analysis* class involves using working memory to make inferences on data. It can include functions such as analyzing trends to predict future information and integrating many observations into a single interpretation. This stage also contains transforming data into other representations, such as a visual display, in order to aid human understanding.
- The *decision and action selection* class utilizes data or trends in order to decide upon an action or suggest any number of options to the human.
- The *action implementation* class implements the chosen action. It may involve constraints such as waiting a specific amount of time to allow the human to interfere.

The first two classes involve autonomous input, while the second two relate to autonomous output. All of these can vary from no autonomy to full autonomy. For example, Table 2 shows Sheridan and Verplank’s 10 levels of automation [34], which holds the various automation levels related to autonomous output.

Table 3. Levels of automation for decision and action selection [34].

HIGH	10. The computer decides everything, acts autonomously, ignoring the human.
	9. informs the human only if the computer decides to
	8. informs the human only if asked,
	7. executes automatically, then necessarily informs the human, and
	6 allows the human a restricted time to veto before automatic execution,
	5. executes that suggestion if the human approves,
	4. suggests on alternative
	3. narrows the selection down to a few,
	2. The computer offers a complete set of decision/action alternatives,
	1. The computer offer no assistance: human must make all decisions and actions.
LOW	

Parasuraman *et al.* [34] utilize these four classes in the Human-Centered Design function allocation method. The process consists of iterating through multiple steps to identify the proper automation levels for each of the four classes of automation

application. Each iteration analyzes two evaluative criteria. The primary criterion evaluates human performance and relates to the complementarity principle. These performance metrics include mental workload, situation awareness, complacency, and skill degradation. The secondary criterion contains automation reliability and the cost of decision/action outcomes. This phase analyzes the probability to failure and situational unreliability, as well as their effect on human trust. It also takes into account the cost of failure, which can vary from no consequence to endangerment of human life.

2.2.2.4 Agent-Oriented Design.

Agent-oriented design [42] aims to create automated agents based on goals and roles within a human-machine team. In this context, an agent includes any “entity that performs a specific activity in an environment which it is aware and that can respond to changes”, while a goal describes an intended state of the human-machine environment. Goals may involve achieving, maintaining, ceasing, or avoiding a specific state in the environment. Meanwhile, roles characterize the functions that an agent must perform in order to attain the goals. One agent-oriented design, combines the Role-Oriented Analysis and Design for Multiagent Programming (ROADMAP) and the Radical Agent-Oriented Process/Agent-Object-Relationship (RAP/AOR) methodologies [42] in order to generate one methodology that converts a human-machine concept to implementation. The ROADMAP and RAP/AOR agent-oriented design involves a ten stage process. Stages I and II define the goals and roles of the system. Stages III and IV identify interactions between roles and environment knowledge required in roles. Stages V and VI assign agents to the roles and outlines knowledge required by each agent. Stages VII and VIII model interactions between agents and outlines their behaviors in specific scenarios. Finally, stages IX and X in-

volve designing the automation-based agents based on their behaviors via UML-based diagrams.

2.3 Collision Detection and Avoidance

In order to capture the human behavior of reliance, *Space Navigator* requires an agent that performs the task of collision avoidance. Collision detection and avoidance remain prevalent in realms such as efficient air traffic management[44], the design of Unmanned Aerial Vehicle systems[41], and applications in computer graphics [25]. The following sections cover components of a real-world collision avoidance system and methods for collision detection in simulated environments.

2.3.1 Components of a Real-World Collision Avoidance System.

Functions of collision avoidance can be separated into five components [1]: sensing, detection, conflict awareness, escape trajectory estimation, and maneuver realization.

Sensing refers to the ability to record environment data. This data is comprised of but not limited to position, velocity, and heading via on-board sensors, similar information of nearby aircraft or obstacles from radar or cameras, and cooperating aircraft details provided by communication links.

Detection involves processing the data in order to create a useful representation of the state-space. This function provides the ability to organize and manage collision risks involving nearby aircraft.

Conflict Awareness projects each collision risk into future state spaces to determine the likelihood of a future collision. If a collision is detected, this function calculates collision-based metrics such as time and point of collision. It may also include the prioritization of collisions.

Escape Trajectory Estimation deals with exploring and choosing a new collision-

avoiding trajectory. Trajectory selection methods may be predefined, protocol based, or algorithmically determined.

Maneuver Realization performs the selected trajectory with aircraft maneuvers. This function may need to consider real-world forces and limitations on the aircraft.

These functions provide the basic implementation of a collision avoidance system. However, other important factors such as how to deal with multiple conflicts and algorithmic-based time complexities may need to be addressed during the design process. Although all of these concepts may not be utilized in the following section, a background of real world systems provide important concepts required to create a collision avoidance agent.

2.3.2 Collision Detection in a Simulated Environment.

The field of computer graphics in video game design requires precise simulations of real-world motions. Imagine a simulated game of billiards. In order to accurately portray the results of a cue ball striking a group of balls, the game designer needs to determine the time, velocity, position, and angle of the collision. Collision detection becomes ever important as developers strive to make more realistic video games. The following three sections discuss commonly used methods for collision detection in simulated environments, while the final section introduces two generic methods of collision avoidance in the *Space Navigator* environment.

2.3.2.1 Multiple Interference Detection.

The first and simplest collision detection methodology involves sampling object trajectories and repeatedly performing a static collision test [25]. Take for example, two circular objects O_1 and O_2 with radii r_1 and r_2 and trajectories that can be represented as a piece-wise function of position over time or $P_1(t)$ and $P_2(t)$, respectfully.

Then the distance between these two objects at any time t can be calculated with $D(t) = ||(P_1(t) - P_2(t))||$. If at any real time $D(t) \leq r_1 + r_2$, then there exists a future collision. With multiple inference detection, this point in time is found by repeatedly sampling across the time domain until the distance between two objects is less than the sum of their radii or no collision is found.

The complexity of multiple inference detection lies within the sampling period of the algorithm. A sampling with too coarse of a period remains computationally efficient; however, it may miss a collision. Meanwhile, a sampling with too fine of a period will likely find all collisions, but may be computationally expensive. Some sampling strategies such as that seen in Gilbert *et al.* [18] utilize adaptive sampling in order to prevent misses and avoid computationally expensive runtimes.

2.3.2.2 Swept Volume Interface.

The second collision detection methodology requires computing the *swept volume* [25]. With two-dimensional objects, the swept volume contains the x and y space that the object moves across during a period of time. If no two object's swept volumes intersect during a period of time, then no collision exists during that period. On the other hand, if two object's swept volumes intersect, then a collision may exist. The swept volume interface provides an efficient method to determine that no collisions occur; however, it cannot confirm that a collision occurs if two swept volumes intersect since it does not account for the time domain. The next collision detection method solves this problem.

2.3.2.3 Spatio-Temporal Intersection.

The third collision detection methodology computes the *extruded volume* of an object [6]. Similar to the swept volume, an extruded volume of a two dimensional

object is created by generating the volume of the object as it moves across space. However unlike swept volume, the extruded volume projects the object into a three dimensional space to include time. Figure 4 shows two examples of swept volume interface and spatio-temporal intersection. On the left side of Figure 4, two collision-imminent objects O_1 and O_2 generate the extruded volumes and swept volumes seen on the top and bottom, respectively. The intersection in the extruded volume confirms that a collision will occur at some point t_c in the time dimension, while the intersection of the swept volume cannot confirm or deny the presence of a future collision. The right side of Figure 4 shows the resulting extruded and swept volumes of two non-colliding objects O_1 and O_2 . In this image, the non-intersection of the extruded volume confirms that a collision does not occur, while the intersection of the swept volume cannot confirm or deny the presence of a future collision.

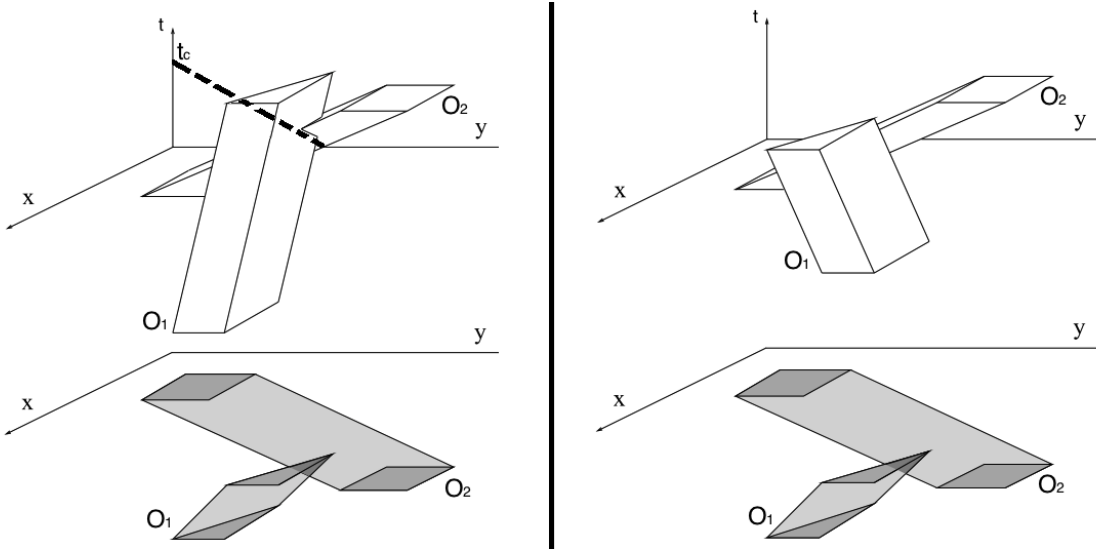


Figure 4. Left: Example of spatio-temporal interaction (top) and swept volume interface (bottom) collision detection methods, where two moving objects O_1 and O_2 collide at time t_c . Right: Example of spatio-temporal interaction (top) and swept volume interface (bottom) collision detection methods, where two moving objects O_1 and O_2 do not collide [25].

Figure 4 provides an example of how spatio-temporal intersection performs collision detection better than swept volume interface; however, generating three di-

mensional volumes for each object requires a much higher runtime cost than the two dimensional counterpart [25].

2.4 Trust in Human Machine Teams

Failures within human-machine teams result from many different sources, including poor system designs; software and hardware failures; and operator abuse, misuse, and disuse [23]. Misuse and disuse of automation often result from inappropriate trust in automation. For this reason, Lee and See [27] identify trust in human-machine teams as a critical factor in team performance. Human operators must mitigate misuse and disuse through the pursuit of appropriate trust, or a level of trust that matches the capabilities of the automation [15].

2.4.1 Appropriate Trust.

In order to achieve appropriate trust [27], Lee and See proposed three areas that examine the relationship of human trust with automation capabilities: calibration, resolution, and specificity.

Calibration refers to the relationship between a person’s level of trust in automation and the capabilities of the automation. Overtrust occurs when human trust surpasses the actual capabilities of the automation. Distrust occurs when the actual capabilities of the automation exceeds human trust. These cases are referred to as low calibration. In the ideal case, a human’s trust matches the actual capabilities of automation. This situation describes high calibration. Figure 5 displays an example of low and high calibration.

Resolution defines how different components of an automation’s capability influence human trust. If a large amount of capability only influences a small amount of trust, then great changes in capabilities may result in little change of trust. When

this occurs, humans display the quality of low resolution. Ideally, a range of capabilities should map to an equivalent range of trust. In this case, humans show a high resolution. Figure 5 shows an example of low and high resolution.

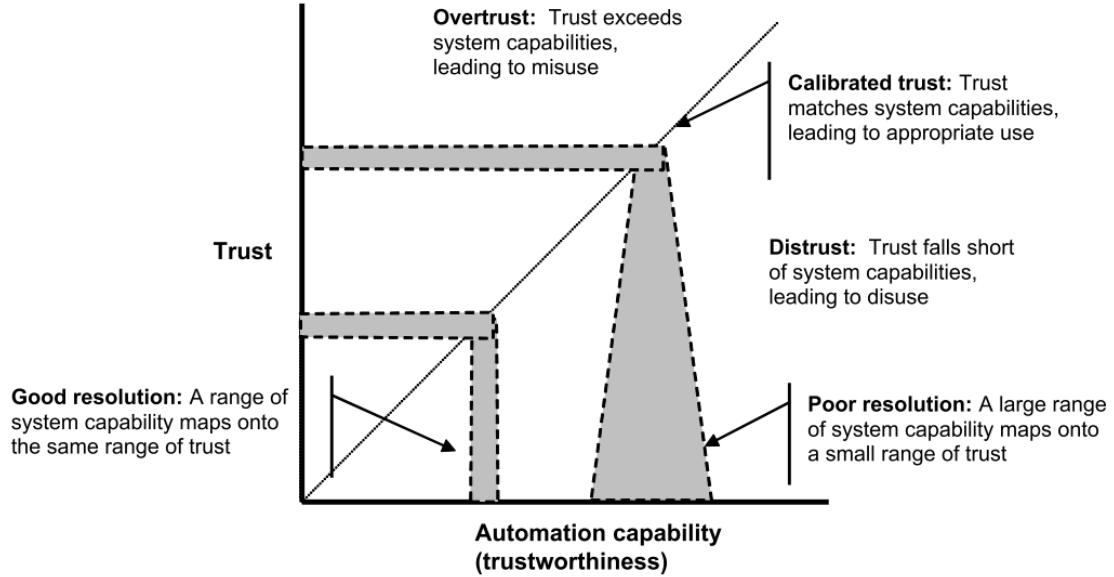


Figure 5. The relationship between trust and capability [27].

Specificity breaks down into two factors: functional specificity and temporal specificity. Functional specificity refers to the amount of trust appropriated to a specific function, sub-function, and various levels of automation. Low functional specificity occurs when a portion of human trust relates to the overall system or larger components. High functional specificity arises when human trust appropriately relates to individual sub-functions and modes of operation. Temporal specificity describes the level of trust provided in given situations or over time. Low temporal specificity occurs when an operator delivers the same level of trust over long periods of time, where high temporal specificity changes frequently on a moment-to-moment basis.

An appropriate level of trust results when the human operator has high calibration, high resolution, and high functional and temporal specificity.

2.4.2 Factors Influencing Trust.

The levels of calibration, resolution, and specificity are directly influenced by the information a human receives about automation. Lee and Moray [27] abstracted this information into three different factors: purpose, process, and performance.

Purpose outlines the designer’s intent for the automation [15]. This factor focuses on the function of the automation and the objectives that the automation aims to perform. It allows the human operator to understand how the automation fits into the overall human-machine team. Purpose helps the human to calibrate their trust to the system’s capability.

Process describes the internal specifics of the automation [15]. This factor includes the system’s methodology such as the algorithm used and the resources required to function appropriately. Understanding the process improves the human’s resolution and functional specificity.

Performance refers to previous and current measures of how well the automation achieves its purpose [27]. It includes information that describes automation reliability and predictability on overall and task-specific functions. This factor influences the human’s temporal specificity.

Duez *et al.* [15] designed a process that utilizes these factors to develop appropriate trust in a human-machine team. This process does not replace methods of agent design, but instead complements them. As result, it can be applied to a wide range of previously developed automation. The process applies the constraint-based, formative analysis tool Work Domain Analysis (WDA). This tool uses Abstraction Hierarchy to describe automation goals at different levels of abstraction, where overall system goals reside on the top level and are broken down into smaller goals at lower levels. The bottom level of the hierarchy contains resources required to achieve higher-level goals. By analyzing the hierarchy, WDA identifies the purpose, process,

and performance information of a system. This information can then be communicated to a human operator by means of documentation, logs, or feedback. Using the knowledge of a system’s purpose, process, and performance helps to aid the human operator in attaining high calibration, high resolution, and high functional and temporal specificity.

2.4.3 Measures of Trust.

When analyzing trust in human-machine teams, one needs a methodology to measure trust subjectively and objectively. Subjective measures contain information on how the human perceived the situation, while objective measures involve physical recordings. Obtaining these methodologies can be difficult in this complicated domain, as trust can depend on many human factors [17] such as biases, inexperience, and even emotions. The remaining sections delve deeper into the topics of subjective and objective trust measurements.

2.4.3.1 Subjective Measures.

Subjective measurements of trust make up the majority of trust measuring methodologies today, because they are more generalizable than the objective counterpart. Subjective measures of trust in human-machine teams can be broken down into three types: propensity to trust, trustworthiness, and trust itself [9].

Propensity to Trust

A person’s propensity to trust depends on the individual’s inherent likelihood to trust the system and is suspect to change throughout the course of interaction [40]. Propensity to trust often results from the individual’s personality and upbringing. Since the propensity to trust is formed before experiencing any automation, it provides a baseline measure of trust in general [24]. The general trust questionnaire seen in

Tables 30 and 31 of Appendix A provides a suitable means for measuring a person’s subjective propensity to trust in humans and automation. This questionnaire was derived from the work of Li *et al.* [29].

Trustworthiness

In the scope of interpersonal trust, trustworthiness reflects the ability, benevolence, and integrity of a person [9]. However in the human-machine teaming domain, trustworthiness defines a person’s perceived capability or reliability of the automation [27]. In *Space Navigator*, trustworthiness can be represented as the perceived reliability of the collision avoidance agent. Since the agent’s job is to avoid the next imminent collision, a subjective measure of trustworthiness can be recorded by asking the human-player the following question: “What percentage (between 0% and 100%) of imminent collisions do you believe were prevented by the automation?”

Trust Itself

A person’s propensity to trust and perceived trustworthiness help inspire their trust in automation. In this case, trust itself refers to the human’s internal opinion or judging of the automation [9]. Subjective measurements to capture trust in automation can be recorded using a Likert-type scale, which asks the person to rate the automation on a scale from *High Trust* to *High Distrust*, as seen in Table 4.

Table 4. A Likert-type scale to capture a human’s subjective trust in automation.

Based on your experience, please indicate your level of trust in the automation.	
1	I highly distrusted the automation.
2	I somewhat distrusted the automation.
3	I neither trusted nor distrusted the automation.
4	I somewhat trusted the automation.
5	I highly trusted the automation.

2.4.3.2 Objective Measures.

Trust is an ambiguous feeling, making it difficult to quantify. Objective measures of trust in human-machine teams exist in prior research [40, 17]. However, these measures only pertain to unique tasks of the system. In order to create a more generic measure of trust, Boubin *et al.* [4] proposed a methodology to capture trust-related measurements through human behaviors of compliance and reliance. Initially, Lee and See [27] coined the terms compliance and reliance. Their research discusses trust in alarm-based autonomy, so compliance referred to the operator’s response when the alarm remains silent, while reliance referred to the operator’s response when the alarm sounds. Boubin *et al.* [4] extended upon these definitions to include all forms of autonomy. Thus, adapting compliance to convey the human’s acceptance of an automation’s action, and reliance to indicate the human’s acceptance of an automation’s non-action.

Since the proposed *Space Navigator* agent performs the reliance-based task of collision avoidance, it involves the human’s acceptance of automation’s non-action. An act of acceptance includes any occasion where the player lets the agent prevent an imminent collision, while an act of non-acceptance occurs whenever the player prevents a collision him or herself. By recording acts of acceptance and non-acceptance during a session of gameplay, an overall rate of reliance can be measured. Players with high reliance rates show trust in the agent, since they trust the agent to prevent collisions while they focus on other tasks. On the other hand, players with low reliance rates show a lack of trust in the agent, as they continue to perform the agent’s task him or herself.

2.5 Chapter Summary

In summary this chapter contains the background required to create the human-reliance measurement system and the human-subject experiment for recording trust-based metrics. The human-reliance measurement system consists of the tablet video game *Space Navigator* with an agent that performs the reliance-based task of collision avoidance. Topics on automation types, function allocation, and collision detection and avoidance provide the information required for the design and development of the collision avoidance agent. Information on trust in human-machine teaming supplies methods for capturing subjective and objective measurements of reliance-based trust. Chapter III utilizes this subject matter to develop the collision avoidance agent and human-subject experiment required to analyze the relationship between human reliance and automation with varying reliability.

III. Methodology

This chapter expands on concepts from Chapter II for development of the reliance-based measurement system and the human-subject experiment. Success of the reliance-based measurement system depends on the proposed collision avoidance agent. For this reason, the following chapter covers the chosen methodology for agent design and implementation of collision detection and collision avoidance. After developing multiple collision avoidance methods, evaluation is performed in order to confirm that is logical for humans to rely on the agent. Justification of the reliance-based measurement system enables a human-subject experiment to evaluate the relationship between human reliance and automation with varying levels of reliability. The remainder of the chapter covers development of a human-subject experiment that allows for varying automation reliability and the collection of objective and subjective trust-based metrics.

3.1 Agent Design

Both human and agent-oriented methodologies provide beneficial concepts for the design of autonomous agents. Since the agent-oriented methodology of ROADMAP and RAP/AOR [42] focuses more on agent roles and their actions, this design process was determined better suited for the implementation of the human reliance system compared to human-centered automation design. Thus, the remainder of this research accepts Sterling *et. al.*'s definition of an agent to include any entity that is aware of its environment, performs a specific function within it, and can respond to perceived changes.

When applied to *Space Navigator*, the process helps determine agent roles, knowledge, and actions for a human player and the line and collision avoidance agents

defined in Section 2.1.2. The line agent focuses on the function of routing ships to planets, while the collision avoidance agent supports avoiding ship-to-ship collisions. If no automation is available, the player assumes responsibility for four functions: routing ships to planets, routing ships to bonuses, avoiding no-fly zones, and avoiding collisions. However, if either the line or collision avoidance agent is engaged, the player assumes a supervisory role for the respective agent’s function, while concentrating on the remaining tasks.

Focusing on the collision avoidance agent, ROADMAP and RAP/AOR resulted in the behavior diagram shown in Figure 6. The diagram flows from top to bottom and consists of rules and actions. Rules, represented by a circle, start and end actions, represented as rounded rectangles. Vertical arrows show control flow and horizontal arrows denote a condition. This behavior diagram enforces the collision avoidance agent’s overarching goal: to reduce the number of collisions. The goal breaks down into two sub-goals of collision detection and collision avoidance.

The functionality of the collision avoidance agent initiates at Rule 1, or the start of any *Space Navigator* gameplay session in which the agent is involved. Rule 1 transitions into the first sub-goal, seen within the top dotted-line rectangle, which begins the task of continuously monitoring the environment. Monitoring needs to take into account the agent’s knowledge on trajectories and velocities of ships in order to predict collisions. The agent also requires some protocol to handle the detection of multiple collisions at once. Once a collision is found at Rule 2, the agent’s functionality switches to the collision avoidance sub-goal, seen within the bottom dotted-line rectangle. This goal begins by deciding which collision-bound ship to act upon. Once this is completed, Rule 3 transitions into the final task of how to reroute the selected ship’s trajectory. Both tasks in the collision avoidance sub-goal may need to account for agent knowledge such as other ship positions and trajectories,

location of the destination planet, and possibility of creating future collisions.

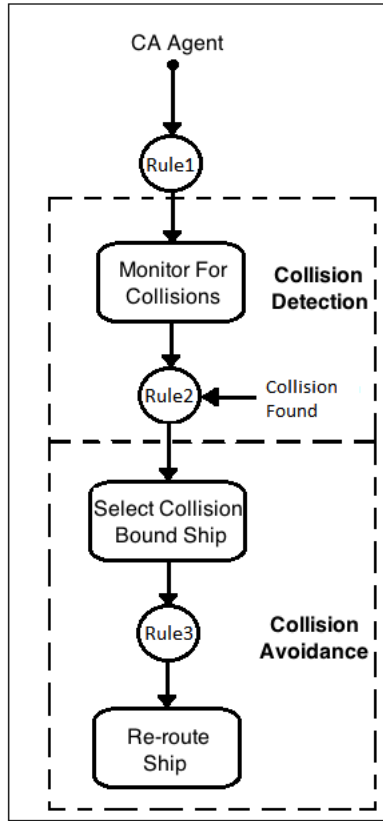


Figure 6. A behavior diagram [42] outlining the functionality of the CA agent.

3.2 Collision Detection

In the *Space Navigator* environment, collision detection involves projecting each ship’s location into the future in order to decide whether or not any two collide. Multiple inference detection [25] applies best in the *Space Navigator* environment, since its deterministic process provides sufficient detection reliability for the collision avoidance agent. In order to detect every possible collision, this process requires comparing every pair of ships in a set range of time.

Algorithm 1 shows the collision detection process, taking as input the full array of ships S in the environment, the number of seconds since game start t_{start} , and a

variable t_{LA} representing the number of seconds to look ahead for a future collision. The algorithm checks each ship's location (lines 6-9) x_{step} times every second for t_{LA} seconds into the future (lines 3-5,14). If a collision is found involving a pair of ships not in the collision table n (line 10), the algorithm adds the ship information and time of collision to the collision table (line 11). After the process looks t_{LA} seconds into the future the algorithm returns a final collision table C , ordered by time of collision (line 16). If the S contains N ships then the algorithm makes at most $x_{step} * t_{LA} * N * (N - 1)$ comparisons. This $O(x_{step} * t_{LA} * N^2)$ runtime becomes computationally expensive when looking far into the future with many ships on the screen.

Algorithm 1: Collision detection algorithm that finds future collisions within the next t_{LA} seconds.

```

1 function Collision Detection ( $S, t_{start}, x_{step}, t_{LA}$ )
  Input : Array of ships  $S$ , seconds since game start  $t_{start}$ , number of times to
           check per second  $x_{step}$ , and the seconds to look-ahead  $t_{LA}$ 
  Output: Collision table  $C$ , holding all detected collisions in order from the
           next  $t_{LA}$  seconds
2 initialize collision table,  $C$ 
3  $t_{test} = t_{start}$ 
4  $t_{end} = t_{start} + t_{LA}$ 
5 while  $t_{test} < t_{end}$  do
6   foreach  $(s_i, s_j) \in S$  do
7      $d = distance(position(s_i), position(s_j))$ 
8      $r_i = radius(s_i)$ 
9      $r_j = radius(s_j)$ 
10    if  $d \leq (r_i + r_j)$  &&  $(s_i, s_j) \notin C$  then
11       $append(s_i, s_j, t_{test})$  to  $C$ 
12    end
13  end
14   $t_{test} = t_{test} + \frac{1}{x_{step}}$ 
15 end
16 return( $C$ )

```

After monitoring the environment t_{LA} seconds into the future, the collision avoidance agent needs some method to store the detected collisions. A table data structure provides the functionality needed, where each entry holds the information for one de-

tected collision. Figure 7 displays the format of this collision table. The first index always holds the next-imminent collision, which the collision avoidance portion acts upon.

Index	Ship1	Ship2	Time Until Collision
1.	ShipID Location	ShipID Location	x seconds
2.	ShipID Location	ShipID Location	y seconds
⋮	⋮	⋮	⋮
n.	ShipID Location	ShipID Location	z seconds

Figure 7. Collision table used by the collision avoidance agent to organize predicted collisions, where $x \leq y \leq z$.

3.2.1 Detection Performance.

Before moving on to collision avoidance, the collision detection must be evaluated to ensure proper functionality. Collision detection can be represented as a binary classification problem, categorizing ships as either “collisions” or “non-collisions.” Collision detection metrics were gathered in a 60 minute session of *Space Navigator* with no agent intervention. During this time, the game recorded the collision table at each frame, about every 0.3 seconds. It also recorded ship creation and destruction events.

The recorded data were processed to create the confusion matrix in Table 5. The *Actual* condition holds the number of ships that collided in the *Yes* column and the number of ships that either ran off screen or reached its destination in the *No* column. The *Predicted* condition contains results from the collision detection’s collision table. A ship may appear in the collision table multiple times with different ships. However, only the next-imminent collision involving a ship will definitely occur. For this reason,

the *Predicted* condition’s *Yes* row only accounts for collision predictions such that it was the next-imminent collision for both ships involved. Meanwhile, the *No* row accounts for ships that were never predicted to be in a next-imminent collision. The game ran for 60 minutes, spawning a ship every 2 seconds. Thus, a total of 1800 ships were spawned. However as the game time ran out, 10 ships remained on screen. Since these 10 ships cannot definitively be categorized as an actual collision or not, they were ignored in the final analysis, resulting in a total sample size of 1790.

Table 5. The confusion matrix of collision detection in 60 minutes of game-play.

n = 1790		Actual		
Predicted		Yes	No	Total
	Yes	TP = 1106	FP = 0	1106
	No	FN = 36	TN = 648	684
	Total	1142	648	1790

Table 5 displays the number of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). These results show that the collision detection correctly classified 1754 ships, while misclassifying 36. All of the misclassifications occurred as false negatives, meaning collision detection did not predict 36 ships involved in a collision. Examining the false negatives found that 30 of the 36 ships involved a collision where one ship spawned less than 1 second prior to collision. The remaining 6 false negatives either occurred in 3-ship collisions or collisions with ships that reached their destination. Table 6 holds performance metrics derived from the confusion matrix in Table 5. According to the true positive and false positive rates, the collision detection correctly identifies collision-bound ships $\frac{1106}{1142} = 96.9\%$ of the time, while never mispredicting ships which do not collide. Thus, every ship predicted to collide will do so as shown by the 100% precision. Lastly, the accuracy shows that the collision detection correctly identifies $\frac{1106+648}{1790} = 98.0\%$ ships as collision-bound or not collision-bound. Although not perfect, these results confirm the efficacy of the agent’s collision detection.

Table 6. Collision detection performance metrics from 60 minutes of game-play.

Performance Metric	Formula	Result
True Positive Rate (Sensitivity)	$\frac{TP}{TP+FN}$	96.9%
False Positive Rate (Fall-Out)	$\frac{FP}{FP+TN}$	0%
Precision (Positive Predicted Value)	$\frac{TP}{TP+FP}$	100%
Accuracy (ACC)	$\frac{TP+TN}{N}$	98.0%

3.3 Collision Avoidance

With the knowledge of the next predicted collision being the first element in the collision table, the collision avoidance agent has the responsibility to try to prevent it. Collision avoidance consists of two stages: ship selection and trajectory rerouting.

The decision of which ship to operate on may take into account many different factors, including: airspace congestion, ship lifespan, and distance from screen edge. The method chosen here is to operate on the collision-bound ship furthest from its destination. This scheme increases agent predictability since ship distance from destination is easier for a human operator to perceive than factors such as ship lifespan or congestion. An increase in predictability will allow players to better anticipate the chosen ship, which may positively influence trust calibration [16]. Moreover, this scheme decreases airspace congestion. By not operating on the ship closest to the destination planet, it is more likely that this ship will reach its destination sooner, leaving fewer ships on the screen.

With the ship selection scheme determined, all that remains is the methodology for re-routing a ship’s trajectory. After considering many possible re-route methods, three potential re-routing techniques were chosen for testing: the step-back route, the least congested route, and the pseudo-random route. In *Space Navigator* trajectories are composed of a series of points. In order to preserve a player’s intended ship

destination and direction, all three re-routing techniques retain the last two points of the ship’s original trajectory.

3.3.1 The Step-Back Route.

The notion behind the first re-routing scheme is to have the ship “take a step back” before moving forward with its original trajectory. Since the predicted collision occurs later in the trajectory, a *step-back point* is placed at the front of the current trajectory. Moving the ship backward to the *step-back point*, before continuing on with its original trajectory temporarily delays the ship’s progress by 2 seconds. This delay attempts to prevent the originally detected collision, but may result in a new future collision.

Algorithm 2: The Step-Back Route algorithm generates a new trajectory by appending a *step-back point* to the front of the original trajectory.

```

1 function Step-Back Route ( $T_{orig}$ )
   Input : The original trajectory to be modified  $T_{orig}$ 
   Output: A new trajectory including the step-back motion  $T_{new}$ 
2  $firstPoint$  = first point in  $T_{orig}$ 
3  $secondPoint$  = second point in  $T_{orig}$ 
4  $stepBackPoint$  =  $getStepBackPoint(secondPoint, firstPoint)$ 
5 initialize new trajectory,  $T_{new} = T_{orig}$ 
6 append  $stepBackPoint$  to front of  $T_{new}$ 
7 return( $T_{new}$ )

```

The Step-Back Route method, outlined in Algorithm 2, takes as input the original ship’s trajectory. The process begins by extracting the next two points in the original trajectory (lines 2-3). These points are utilized in the *getStepBackPoint* function, which returns the previously defined *step-back point* (line 4). Adding this point to the front of the original trajectory queue creates a new *step-back trajectory* (lines 5-7). Since the *Step-Back Route* function involves no loops, it results in a small, constant runtime.

3.3.2 The Least Congested Route.

The next re-routing methodology involves representing *Space Navigator* as an $m \times n$ grid graph. Figure 8 shows an example of this representation, where a vertex is the center of a grid square and edges connect adjacent and diagonally touching grid squares.

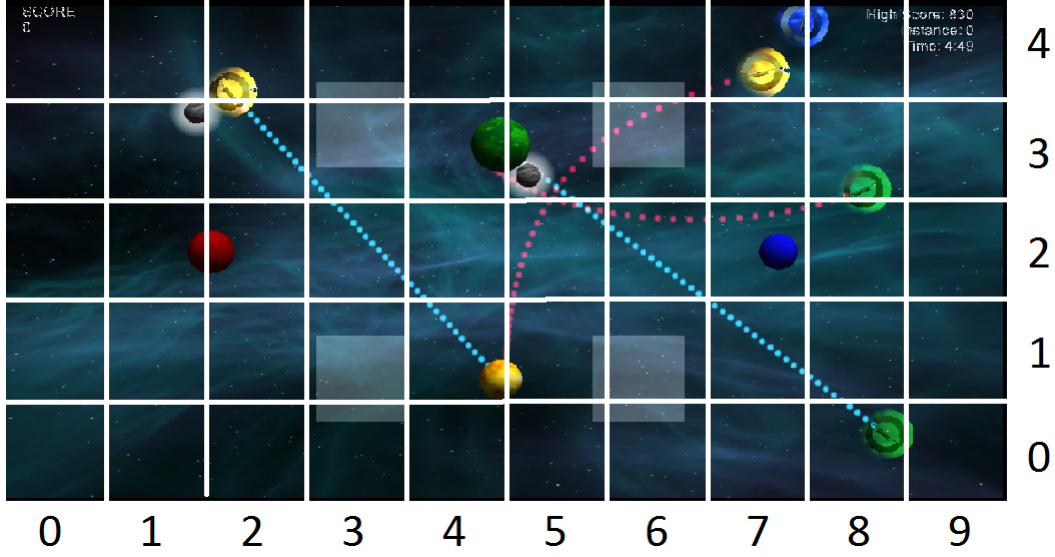


Figure 8. Space Navigator represented in a 5X10 grid.

The least congested method, considers all ship trajectories t_{LA} seconds into the future in order to find a least congested route. In order to do so, the *getCongestion* function was developed, taking in as input each ship's time-based trajectory points. It maps each point to a square, adding weight to the grid graph's corresponding vertex. Weights are determined by the following equation:

$$weight = t_{LA} - t_{Encounter} + 1$$

Where t_{LA} holds a constant value representing the number of seconds looked into the future and $t_{Encounter}$ holds the number of seconds until a ship reaches its trajectory point. For example if the function looks $t_{LA} = 15$ seconds into the future, a point to

be encountered in 1 second holds a weight of 15 and a point to be encountered in 15 seconds holds a weight of 1.

The least congested method, shown in Algorithm 3 takes in as input the trajectory to be modified, an array holding all time-based trajectory points, and the $m \times n$ grid graph. The process begins by updating the graph's vertex weights given the set of time-based trajectory points (line 2). It then extracts the next point and last two points of the current trajectory (lines 3-5). These points and the vertex weights are utilized in a modified version of Dijkstra's shortest path algorithm (lines 6-31) [10], where path costs include vertex weights (line 25). When the destination point is reached (line 16), the algorithm creates the least congested path (lines 17-22). Adding the final point to this path generates the new trajectory (lines 32-34).

Dijkstra's shortest path has a runtime of $O(|E| + |V| * |\log V|)$, where V is the number of vertices and E is the edges. Since every node connects to its adjacent neighbors, there exists 3 connection for each corner node, 5 connections for each edge node, and 8 connections for each interior node. With both m and n larger than 2, the grid graph consists of 4 corner nodes, $2(m - 2) + 2(n - 2)$ edge nodes, and $(m - 2) * (n - 2)$ interior nodes. For example, a 3x3 graph has 4 corner nodes, 4 edge nodes, and 1 interior node. Thus, an $m \times n$ grid results in Equations 1 and 2 for V and E , respectively.

$$V = m * n \tag{1}$$

and

$$\begin{aligned} E &= 3 * 4 + 5[2(m - 2) + 2(n - 2)] + 8[(m - 2) * (n - 2)] \\ &= 8m * n - 6m - 6n + 4 \end{aligned} \tag{2}$$

Therefore, the Least Congested Route algorithm for an $m \times n$ grid results in a time complexity of $O(|8m * n - 6m - 6n + 4| + |m * n| * |\log(m * n)|)$. In order for *Space*

Algorithm 3: The Least Congested Route algorithm generates a new trajectory that takes the least congested path to its original destination.

```

1 function Least Congested Route ( $T_{orig}, P, G$ )
  Input : The original, size  $n$  trajectory to be modified  $T_{orig}$ , an array holding
           all time-based trajectory points  $P$ , and the grid graph  $G$ 
  Output: A new least congested trajectory  $T_{new}$ 
2  $G.vertexWeights = getCongestion(P)$ 
3  $startPos = \text{first point in } T_{orig}$ 
4  $destPos = \text{second last point in } T_{orig}$ 
5  $finalPos = \text{last point in } T_{orig}$ 
6 initialize Dijkstra components:  $dist, prev, nodes, path$ 
7 foreach vertex  $v \in G$  do
8    $dist[v] = \infty$ 
9    $prev[v] = null$ 
10   $nodes.add(v)$ 
11 end
12  $dist[startPos] = 0$ 
13 while  $nodes \neq \emptyset$  do
14    $u = \text{vertex in nodes with smallest } dist$ 
15    $nodes.remove(u)$ 
16   if  $u \text{ equals } destPos$  then
17      $path[] = null$ 
18     while  $previous.Contains(u)$  do
19        $path.enqueue(u)$ 
20        $u = prev[u]$ 
21     end
22      $break$ 
23   end
24   foreach neighbor  $v$  of  $u$  do
25      $alt = dist[u] + distBetween(u, v) + vertexWeights(v)$ 
26     if  $alt < dist[v]$  then
27        $dist[v] = alt$ 
28        $prev[v] = u$ 
29     end
30   end
31 end
32 initialize new trajectory,  $T_{new} = path$ 
33 append  $lastPos$  to end of  $T_{new}$ 
34 return( $T_{new}$ )

```

Navigator to run smoothly, m and n must be small enough for Dijkstra’s algorithm to complete during one frame. However if m and n are too small, the Least Congested Route algorithm has less grids to work with and cannot perform as well. After testing with different variables, the values $m = 10$ and $n = 20$ were chosen as they allowed 200 nodes with no visual lagging.

3.3.3 The Pseudo-Random Route.

The final reroute generator creates a new pseudo-random route. This method serves as a control group to analyze the other collision avoidance methodologies. If the step-back and least congested routing methods perform significantly better than the pseudo-random method, then one can conclude that those two methods perform well and not just any routing method will suffice.

The pseudo-random method, shown in Algorithm 4 takes in as input the trajectory to be modified. The process begins by extracting three points from the original trajectory, the next point and the last two points (lines 2-4). The next point in the trajectory provides a starting reference, while the last two points are used to maintain the original trajectory destination. The function then iterates through a point generation-loop, which continues until the new point lies within a set distance from the destination (line 7). This loop utilizes two random values between 0-1 to create a new trajectory point (lines 8-9).

The first random value, P_{Dir} , determines whether or not the new point moves toward or away from the destination with the following process: The *getVelocity* function is used to determine the velocity vector from the current point to the destination (line 10). The sign of the x and y components provide the direction toward the destination (lines 11-12); for example, a negative x and a positive y means that the destination is to the left and above the current point. P_{Dir} then decides whether

Algorithm 4: The Pseudo-Random Route algorithm generates a pseudo-random path to its original destination.

```

1 function Pseudo-Random Route ( $T_{orig}$ )
  Input : The original trajectory to be modified  $T_{orig}$  of size  $n$ 
  Output: A new pseudo-random trajectory  $T_{new}$ 
2  $startPos = T_{orig}[0]$ 
3  $destPos = T_{orig}[n - 2]$ 
4  $finalPos = T_{orig}[n - 1]$ 
5 initialize new trajectory,  $T_{new}$ 
6  $newPoint = startPos$ 
7 while  $distance(newPoint, destPos) > 3$  do
8    $P_{Dir} = generateRandom(0, 1)$ 
9    $P_{\theta} = generateRandom(0, 1)$ 
10   $(vel.x, vel.y) = getVelocity(newPoint, destPos)$ 
11   $x_{Dir} = Sign(vel.x)$ 
12   $y_{Dir} = Sign(vel.y)$ 
13  if  $P_{Dir} > 0.95$  then
14     $x_{Dir} = -1 * x_{Dir}$ 
15     $y_{Dir} = -1 * y_{Dir}$ 
16  end
17   $x_{Offset} = 2.5 * P_{\theta}$ 
18   $y_{Offset} = \sqrt{2.5^2 - x_{Offset}^2}$ 
19   $newPoint.x = newPoint.x + x_{Dir} * x_{Offset}$ 
20   $newPoint.y = newPoint.y + y_{Dir} * y_{Offset}$ 
21  add  $newPoint$  to  $T_{new}$ 
22  if  $size\ of\ T_{new} > 300$  then
23    break
24  end
25 end
26 append  $destPos$  and  $finalPos$  to end of  $T_{new}$ 
27 return( $T_{new}$ )

```

or not the new point should move toward the destination, by not flipping the sign (95% chance), or away from the destination, by flipping the sign (5% chance) (lines 13-15).

The second value P_{θ} determines the magnitude of the new point's x and y offset (lines 17-18). The offsets range from 0-2.5 in order to create a vector with magnitude 2.5, equivalent to the ship's speed. The direction and offsets are then combined with

the current point to generate a new “random” point (lines 19-21). In order to prevent the unlikely case of an infinite loop, the trajectory generation is capped at 300 points (lines 22-24). Again, to maintain ship destination and direction, the last two points are added to the new trajectory (lines 26-27). Since the Pseudo-Random Route algorithm maxes out at 300 points or 300 iterations, the algorithm has a constant time complexity, $O(1)$.

3.4 Evaluation of Collision Avoidance Agents

In order for the collision avoidance agent within *Space Navigator* to be an effective reliance measurement system, it must avoid enough collisions that it is logical for a player to rely on it. This effectiveness is confirmed by comparing game-play data with and without the collision avoidance agent.

Table 7. Scenarios providing performance metrics analyzed in the evaluation experiment.

Scenario	Number of Games	Source
Line Agent	108	Repeated Study
Human	108	Previous Study [3]
Human & Line Agent	108	Previous Study [3]
Line Agent & Step-Back CA Agent	108	Repeated Study
Line Agent & Least Congested CA Agent	108	Repeated Study
Line Agent & Pseudo-Random CA Agent	108	Repeated Study

A previous *Space Navigator* human-subject study [3] provides score and collision measurements across 36 participants, resulting in human-subject data for 108 games with no autonomous aid and 108 games with the line agent. The following experiment repeats the previous human-subject study [3] with the line agent alone and paired with each collision avoidance method to gather performance metrics of these agent only teams. Performance metrics from the scenarios seen in Table 7 are analyzed in

the following sections to determine whether or not human-subjects should rely upon the collision avoidance agent.

3.4.1 Performance Without Collision Avoidance Agent.

Since the purpose of this experiment is to determine whether or not it is logical for a human to rely upon the collision avoidance agent, game-play without the collision avoidance agent provides a baseline of how well the line agent and human player perform in *Space Navigator*. Table 8 shows the mean number of ships involved in a collision, mean score, and their associated 95% confidence intervals across 108 five-minute games. This baseline confirms that human-players perform statistically better than the line agent in both collision avoidance and score metrics. However once the line agent and human-player work together, they significantly outperform both cases of playing alone.

Table 8. Average baseline performance results across 108 five-minute games with the human and/or line agent.

Scenario	Collisions		Score	
	Mean	95% CI	Mean	95% CI
Line Agent	41.3	± 1.39	4172	± 279
Human	30.5	± 2.32	5027	± 445
Human & Line Agent	23.2	± 1.65	7117	± 306

3.4.2 Performance With Collision Avoidance Agent.

The second set of testing involves the collision avoidance agent aiding the line agent in game-play. The results, seen in Table 9, hold the mean number of ships involved in a collision, mean score, and their associated 95% confidence intervals across 108 five-minute games. These results show that the step-back and least congested methods significantly outperform the pseudo-random routing scheme. This is expected as the

pseudo-random method does not necessarily try to avoid the detected collision, but instead simply creates a new “random” trajectory. Comparing the step-back and least congested methods, the results show relatively similar performances. However the two methods’ 95% confidence intervals do not overlap in either area. Thus confirming that when used with the line agent, the step-back method performs statistically best in collision avoidance and score metrics.

Table 9. Average collision avoidance performance results across 108 five-minute games with the line agent and each of the collision avoidance methods.

Scenario	Collisions		Score	
	Mean	95% CI	Mean	95% CI
Line Agent & Step-Back CA Agent	8.3	± 0.84	9911	± 179
Line Agent & Least Congested CA Agent	18.1	± 0.97	8096	± 204
Line Agent & Pseudo-Random CA Agent	37.0	± 1.21	4743	± 249

3.4.3 Performance Overall.

Results from Tables 8 and 9 allow for side-by-side comparisons of the collision and score metrics across all scenarios. These comparisons, seen in Figures 9 and 10, display performance metrics for the line agent, human, human/line agent team; and the line agent teamed with step-back, least congested (LC), and pseudo-random (Random) collision avoidance methods.

Figures 9 and 10 show two major findings: the collision avoidance agent using the step-back method teamed with the line agent performs collision avoidance better than the human/line team, resulting in a hypothesis that a human/step-back team will perform as well as any other human/line or human/collision avoidance team. These results confirm that it is logical for the human to rely upon the collision avoidance agent with the step-back methodology for the task of avoiding collisions. Therefore, a

Space Navigator system using this agent provides a useful tool for measuring human-reliance with automation in a human-machine team.

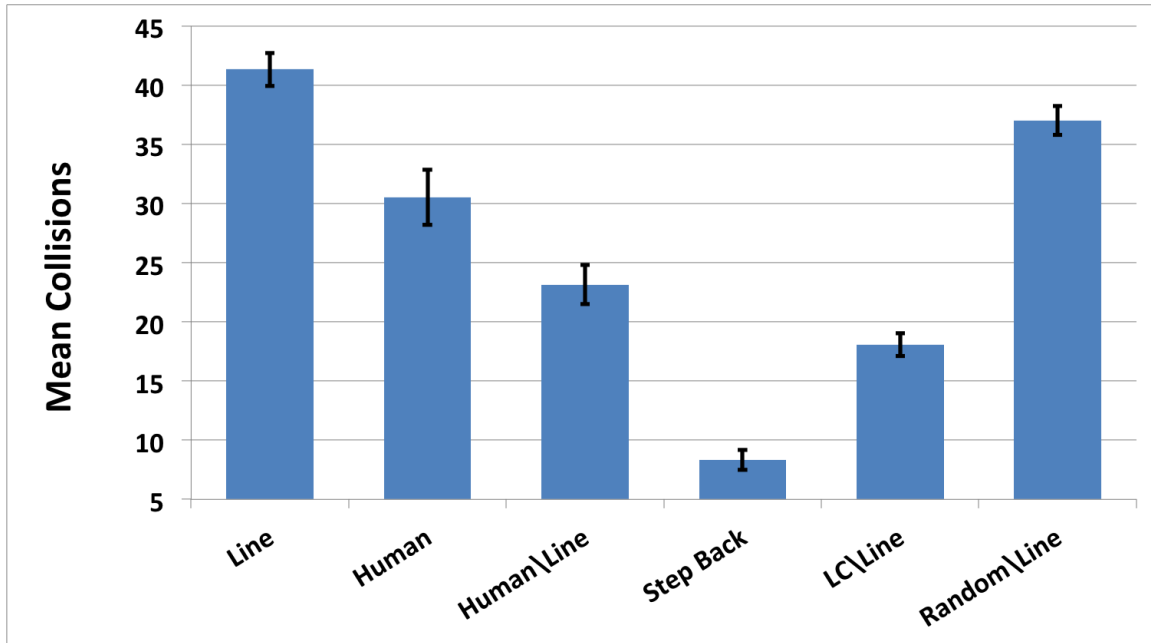


Figure 9. Average number of collisions with 95% confidence interval across 108 five-minute games for each examined set of players.

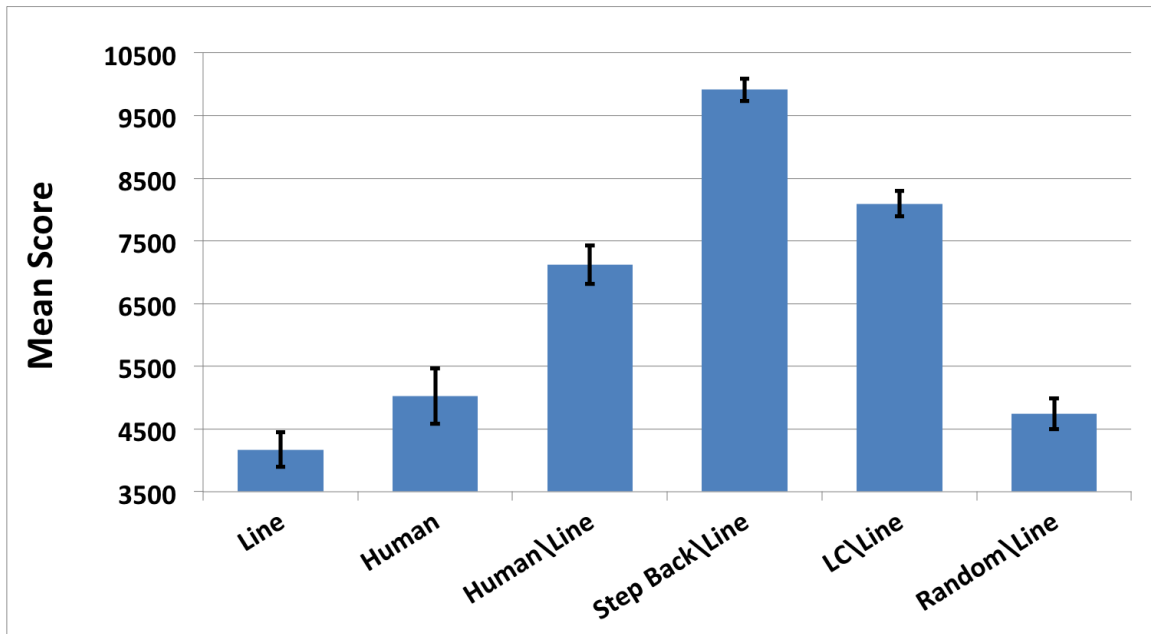


Figure 10. Average score with 95% confidence interval across 108 five-minute games for each examined set of players.

3.5 Reduced Agent Reliability

The research question aims to examine the relationship between human reliance and the level of automation reliability in a human-machine team. The reliance-based collision avoidance agent presents a means to engender human reliance; however it still requires the ability to operate at various levels of reliability. These various levels are defined by a reliability rate seen in Equation 3 which represents the percentage of detected collisions that are acted upon by the agent.

$$\text{Reliability Rate} = \frac{\text{Number of Agent Re-Routes}}{\text{Number of Agent Re-Routes} + \text{Number of Detected Collisions Ignored by Agent}} \quad (3)$$

Since the collision avoidance agent acted upon every collision detected in the evaluation experiment of Section 3.4.3, it operated at 100% reliability. A 100% reliability rate does not mean that no collisions will occur, but that every detected collision will be acted upon by the agent. The remainder of this section covers the process of generating reduced reliability rates and an evaluation experiment to ensure efficacy.

3.5.1 Reduced Reliability Process.

Achieving an exact reduced reliability rate during a game-play session of *Space Navigator* cannot always be accomplished, as the number of collisions vary from game to game. For this reason, the agent requires some means to determine which collisions to act upon and which to ignore in order to achieve a reliability rating of acted upon collisions close to the desired percentage.

The chosen reduced reliability method, known as shuffle bags [11], involves filling an imaginary “bag” with a set of booleans. The set holds a distribution of *true* and *false* values which represent the desired percentage. For example a 95% reliability

rating would include a set with 19 *true* values and 1 *false* value. Each time the agent detects a collision, it randomly samples a boolean from the bag without replacement. If it receives a *true*, the agent acts upon the collision. If it receives a *false*, the agent ignores the collision. The agent continues this process until the bag empties, at which point it refills the bag with the original set of booleans. The shuffle bags process provides the best of both worlds: unpredictability and consistent reliability rates close to the desired percentage. For this reason, the collision avoidance agent utilizes shuffle bags in order to provide reduced reliability ratings.

3.5.2 Reduced Reliability Evaluation.

Although the shuffle bags method seems effective, it must be evaluated with the collision avoidance agent before use in human-subject testing to show that it performs appropriately. The evaluation consisted of running 48 five-minute games in *Space Navigator* at the following reliability rates: 95%, 90%, 80%, and 70%. Table 10 contains the observed reliability rates from the experiment, which hold roughly the same value as their desired programmed rates. Analysis of Table 10’s results in Figure 11 shows that the programmed reliability rates fall within the confidence intervals of the experiment’s resulting rates, confirming the effectiveness of applying shuffle bags to achieve consistent reduced reliability rates.

Table 10. The resulting mean reliability rates and confidence intervals from 48 games of 5 minutes at each programmed reliability rate.

Programmed Reliability	Observed Reliability	
	Mean	95% CI
95%	95.04%	± 0.232
90%	89.87%	± 0.282
80%	80.16%	± 0.494
70%	69.93%	± 0.590

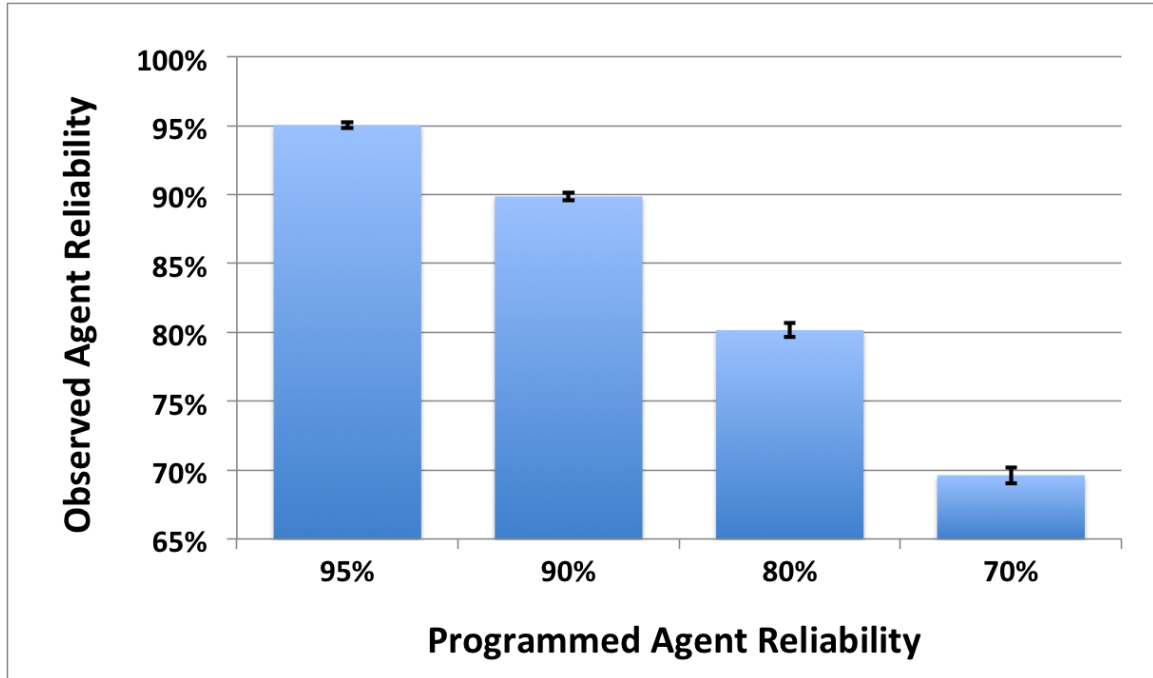


Figure 11. The resulting mean reliability rates from 48 games of 5 minutes at each programmed reliability rate.

3.6 Human-Subject Experiment

In order to determine how the level of reliability of automation affects the reliance of human subjects, a human-subject experiment utilizing the newly developed reliance-based measurement system must be performed. The main goal of the human-subject experiment is to gather human-subject data in the form of subjective and objective measures of human trust and trust-related behaviors. This data can be used to analyze how the human-players' trust-based metrics are affected by degraded automation reliability. The remainder of this section outlines data collected during the experiment, as well as the experiment design.

3.6.1 Experiment Data.

The experiment involves collecting subjective measurements through pre- and post-experiment questionnaires, as well as between-game questions. It also captures

objective measurements in the form of in-game events. All of the data collected is not utilized in this research; however, it may be of use in future research. The remainder of this section covers all of the data collected.

3.6.1.1 Subjective Measurements of Trust.

The subjective measurements in this experiment observe two important factors of human reliance in automation: the participant’s propensity to trust and his or her perceived trust in the automated agent.

A participant’s propensity to trust [24] provides a baseline of general trust that explains variance across different participant data. Furthermore, this baseline influences the rate at which a participant accepts or ignores aid from the collision avoidance agent. In order to create such a baseline, the Likert-scale questionnaires seen in Tables 30 and 31 of Appendix A are filled out by each participant prior to any game-play. Subsequent to capturing the propensity to trust, participants provide the remaining subjective measurements, which collect perceived trust and reliability of the collision avoidance agent. These measurements result from a short in-game questionnaire tasked to participants after each session with the collision avoidance agent at varying levels of reliability. Figures 12 and 13 show screen-shots of the questionnaires in *Space Navigator*.

The question in Figure 12 measures a feeling of trustworthiness toward the automation. In other words, it contains the participant’s perceived capability of the collision avoidance agent. Meanwhile, the question in Figure 13 records the participant’s overarching feeling of trust in the automated agent. These questionnaires record data required to answer the investigative questions.

Based on your experience playing the previous block, please indicate how reliable you believe the automation to be from 0 to 100.

What percentage of imminent collisions do you believe were prevented by the automation in the past block?

Invalid input.

☐ This box must be checked and input must be valid before answers will save.

Save Answers

Figure 12. A screen-shot of the reliability question within *Space Navigator*.

Based on your experience playing the last block, please indicate the extent to which you trusted the automated agent.

1) I highly trusted the automated agent.

2) I somewhat trusted the automated agent.

3) I neither trusted nor distrusted the automated agent.

4) I somewhat distrusted the automated agent.

5) I highly distrusted the automated agent.

1 = High Trust 5 = Low Trust

☐ This box must be checked before answers will save.

Save Answers

Figure 13. A screen-shot of the trust question within *Space Navigator*.

3.6.1.2 Remaining Subjective Measurements.

Further subjective measurements provide general information on the participants and their workload while playing *Space Navigator*. Prior to any game-play, the par-

participants fill out a pre-experiment questionnaire to capture demographic information seen in Table 29 of Appendix A. After the experiment, participants fill out a post-experiment questionnaire seen in Appendix B. This questionnaire queries the participants on their overall experience with *Space Navigator*; their weekly usage of laptops, tablets, smart phones, desktop computers, and gaming consoles; and their background with various types of games. During the experiment, the participants answer the Instantaneous Self-Assessment (ISA) [43] and NASA-Task Load Index (TLX) [22] workload questionnaires. These questionnaires aim to capture immediate subjective ratings of the participants performance and workload after each session of *Space Navigator*.

3.6.1.3 Objective Measurements of Reliance.

Objective measurements come directly from events recorded in the human-reliance measurement system. An act of reliance occurs in two scenarios: when the collision avoidance agent re-routes a collision-bound ship and when it ignores an imminent collision. The first scenario captures events where the human relies upon the agent, and the agent performs its job reliably. Meanwhile, the second scenario captures events where the human relies upon the agent, but the agent fails to perform its job. Both scenarios require the assumption that the participant saw the imminent collision and relied upon the agent to prevent it. Without this assumption, there would be no means to measure the human behavior of reliance.

The human-reliance measurement system also allows recordings of *unreliance*. This term refers to occasions when the participant performs the collision avoidance before the collision avoidance agent. In particular, unreliance occurs every time the participant draws a trajectory for a collision-bound ship. The unreliance measurement requires the assumption that the participant saw the imminent collision and

acted it before the agent could. Again without this assumption, there would be no means to measure the human behavior of reliance.

Recording acts of reliance and unreliance during a game of *Space Navigator* provides data required to calculate a participant’s reliance rate. The reliance rate, seen in Equation 4, holds the overall percentage that a participant relied upon the collision avoidance agent and provides an objective measurement related to trust.

$$\begin{aligned} \text{Reliance Rate} &= \frac{\text{Acts of Reliance}}{\text{Acts of Reliance} + \text{Acts of Unreliance}} \\ &= \frac{\text{Number of Agent Re-Routes} + \text{Number of Collisions Ignored}}{\text{Number of Agent Re-Routes} + \text{Number of Collisions Ignored} + \text{Acts of Unreliance}} \end{aligned} \tag{4}$$

3.6.1.4 Remaining Objective Measurements.

Further objective measurements consist of game-specific data and the Implicit Association Test [20]. The recorded game-specific data falls within one or more of the following categories: overall run, action, ship-life, and point-based.

- *Overall run data* contains identifying information and the resulting score of a specific game of *Space Navigator*. For the most part, run data provides a label to identify the exact game other in-game events occur in. This category includes user and run identifiers, a run’s score, and the agent reliability.
- *Action data* contains recordings of human and agent actions. This category includes trajectories, and acts of reliance and unreliance.
- *Ship-life data* encompass all ship creation and destruction events. This category includes when ships spawn, reach a destination, collide, and exit the screen.
- *Point-based data* holds all events that contribute to or subtract from the participant’s score. This category includes destination, collision, bonus, and NFZ metrics.

Table 35 of Appendix C lists all objective data, their description, and the category they fall within.

3.6.2 Experiment Design.

The human-subjects experiment was approved by an Institutional Review Board on 6 October 2016 with the approval number FWR20160150H. Participants were provided the informed consent document seen in Appendix D prior to the experiment and received *Space Navigator* training via the experiment slides and script in Appendix E. The goal of the experiment is to gather the subjective and objective human-reliance measurements of the collision avoidance agent with varying reliability. The recorded data enables analysis of the relationship between reliance-based trust metrics and varying automation reliability. The experiment consists of a within-subjects block-based design, playing *Space Navigator* with two blocks of 100% reliability and four blocks of different degraded reliability rates. The blocking scheme aims to reduce variability from extraneous factors such as the order of reduced reliability rates the participant experiences and aims to remain consistent with other ongoing related research at the Air Force Institute of Technology to facilitate further analysis on trust calibration. The experiment takes approximately two hours and consists of the following three phases:

Phase 1: Baseline

Purpose: To set a baseline of game-play used to both introduce the player to the Space Navigator game and to establish a model of reliance based on the inherent reliability of automation.

Method: Each participant is required to complete the pre-experiment questionnaire seen in Appendix A to gather demographic and propensity to trust humans and automation information and an Implicit Association Test [20] to capture implicit

feelings of trust toward humans and automation. The participant performs each task on a tablet computer with the *Space Navigator* game loaded. They are asked to complete four warm-up games in their entirety. The first warm-up game consists of a two-minute introduction to *Space Navigator* with no autonomous aid. The remaining three warm-up games consist of three three-minute games with automation, to gain experience with the collision avoidance agent. A complete game must be uninterrupted. Participants are advised on proper use conditions. Upon completion of the task, the participant moves on to the next experiment phase.

Phase 2: Varying Automation Reliability

Purpose: To gather significant game-play information surrounding a participants interaction with a varying reliability automated agent that aids them in playing Space Navigator.

Method: Each participant is required to complete six blocks of two five-minute games of Space Navigator. Each block contains the collision avoidance agent with one of the following reliability rates: 100% (twice), 95%, 90%, 80%, and 70%. Prior to each game, the participants is not aware of the reliability rates. The sequence of reliability rates begins and ends with 100% reliability in blocks 1 and 6. The four blocks in between contain user-based ordering of the reduced reliability rates as seen in Table 11.

Over the course of 24 participants, all of the 24 possible block organizations are used. After each block, participants rate the agent’s level of reliability/trust with the questions seen in Figures 12 and 13, and are given the ISA [43] and NASA-TLX [22] battery questions. Upon completion of the task, the tablet computer is collected and recorded data are uploaded for processing. Participant data is de-identified by assigning a participant number.

Table 11. Each user’s reliability rate order for blocks 2-5.

User	Block 2	Block 3	Block 4	Block 5
1	70	80	90	95
2	70	80	95	90
3	70	90	80	95
4	70	90	95	80
5	70	95	80	90
6	70	95	90	80
7	80	70	90	95
8	80	70	95	90
9	80	90	70	95
10	80	90	95	70
11	80	95	70	90
12	80	95	90	70
13	90	70	80	95
14	90	70	95	80
15	90	80	70	95
16	90	80	95	70
17	90	95	70	80
18	90	95	80	70
19	95	70	80	90
20	95	70	90	80
21	95	80	70	90
22	95	80	90	70
23	95	90	70	80
24	95	90	80	70

Phase 3: Post Experiment Interview

Purpose: To gather data based on the entirety of the experiment in order to assess the persons experiences with the adaptive automation system.

Method: Each participant will be required to complete the post-experiment questionnaire seen in Appendix B. The interview will assess the participants subjective views about the experiment, their performance in it, and any experience with electronics and games.

3.7 Chapter Summary

In summary, this chapter covers the methodology for the design of a collision detection and avoidance agent and a human-subject experiment. After confirming the efficacy of a collision detection methodology, three different collision avoidance techniques were analyzed: step-back, least congested, and pseudo-random. The step-back methodology combined with the line agent, performed significantly better than any other prior human-agent/agent-agent team [2]. This confirmed that it is logical for a human player to rely upon the collision avoidance agent in future experiments. Since the agent is used to record human reliance in automation with varying reliability, a method for enabling reduced reliability was developed. The collision avoidance agent with varying reliability is utilized in the outlined human-subject experiment in order to capture reliance-based trust metrics required for this research. Chapter IV covers the results of the human-subject experiment, providing analysis on human reliance in automation with varying levels of reliability.

IV. Results

This chapter reports the results and analysis from the human-subject experiment described in Section 3.6 of Chapter III. The first three sections present analysis on the investigative questions, comparing the trust-related factors of subjective reliability, subjective trust, and the reliance rate to each programmed reliability rate of the agent. The remaining sections look into correlations and analysis of variance (ANOVA) of these trust-related factors. These results provide characteristics and trends in order to examine the relationship between human reliance and automation with varying levels of reliability.

4.1 Subjective Reliability vs. Programmed Reliability

A participant’s subjective reliability ratings represent their perception of an autonomous agent’s capability. In the trust domain, this value represents a participant’s feeling of trustworthiness [27] toward the collision avoidance agent. Thus, comparing subjective reliability ratings against the programmed reliability of an agent allows for analysis on participant’s calibration to the agent’s capability. In order to capture a participant’s subjective view of the collision avoidance agent’s reliability, they were asked the question “What percentage of imminent collisions (from 0-100%) did the automation prevent?” after each block, as seen in Figure 12 of Section 3.6.1.1. Since the agent does not always prevent every collision, as seen in Section 3.4.3, the reliability question does not directly reflect the programmed reliability. Thus, the actual agent reliability rate in each block needs to be computed via Equation 5.

$$\text{Actual Reliability Rate} = \frac{\text{Number of Collisions Prevented by Agent}}{\text{Number of Collisions Prevented by Agent} + \text{Number of Collisions}} \quad (5)$$

The actual reliability rate holds the percentage of collisions prevented by the collision avoidance agent. When the collision avoidance agent re-routes a ship, that ship moves backward for 1 second and then continues on with its original trajectory or path. A collision is deemed “prevented” if the collision-imminent ship does not collide 1.5 seconds after the agent re-routes it. In other words, if a ship completes its step-back motion and continues forward without colliding, then the agent prevented the imminent collision. After performing the human-subject experiment, there existed 144 actual and subjective reliability ratings. Table 12 holds the average actual and subjective reliability rates and their 95% confidence intervals from 24 participants across 6 blocks of various programmed reliability rates. The two sessions of 100% programmed reliability rates are identified by block, where B1 and B6 represent blocks 1 and 6, respectively.

Table 12. Human-subject experiment results of the actual and subjective reliability across each programmed reliability rate.

Programmed Reliability	Actual Reliability		Subjective Reliability	
	Mean	95% CI	Mean	95% CI
100% B1	95.3%	± 0.75	76.4%	± 8.74
95%	91.4%	± 0.80	70.6%	± 7.26
90%	87.5%	± 0.96	60.8%	± 6.64
80%	81.0%	± 1.43	54.4%	± 5.54
70%	73.3%	± 1.20	47.3%	± 5.76
100% B6	94.8%	± 0.78	77.9%	± 7.53

Figure 14 displays the results from Table 12. At first glance, the figure shows that none of the actual and subjective reliability rating’s confidence intervals overlap. Thus, participants poorly perceive the reliability of the collision avoidance agent. This conclusion suggests that participants notice failures of automation more than successful operation. Another insight drawn from Figure 14 is that participants more accurately rate the reliability of higher programmed reliability agents. This is shown in the differences between the confidence intervals at each programmed reliability rate.

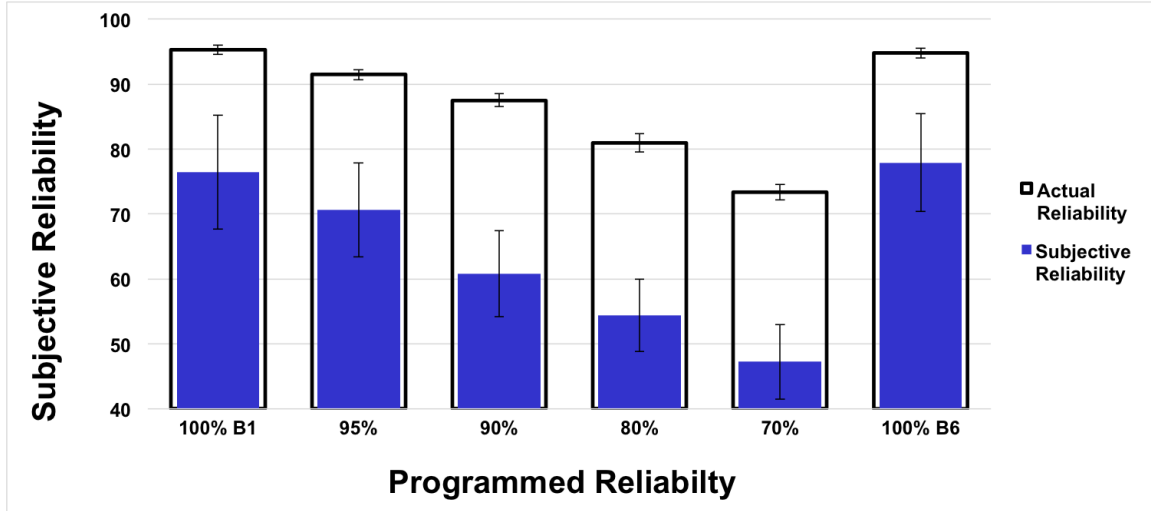


Figure 14. Actual reliability rate and subjective reliability ratings across each programmed reliability rate.

There exists a confidence interval difference of about 10% for higher reliability rates ($\geq 95\%$) and a difference of about 20% for lower reliability rates ($< 95\%$). Specifically, the confidence interval differences for the 100% block 1, 100% block 6, 95%, 90%, 80%, and 70% are 9.4%, 8.6%, 12.8%, 19.1%, 19.6%, and 19.1%, respectively.

The large confidence intervals of the subjective reliability suggests a significant difference among the participant's ratings of reliability. A significant difference in means can be confirmed using one-way analysis of variance (ANOVA) with the subjective reliability as the dependent variable and the participant number as the independent variable. Table 13 holds the results of the one-way ANOVA.

Table 13. Results from the one-way ANOVA of subjective reliability by participant. (* denotes significance at the 0.01 level)

Source	Df	Sum of Squares	Mean Square	F-Statistic	Pr(>F)
User	23	23298.639	1012.98	3.2453	<0.0001*
Error	120	37456.667	312.14		
C. Total	143	60755.306			

In one-way ANOVA, the null hypothesis is that all group means are equal. The F-statistic holds the ratio of variance between subjects over variance within subjects.

An F-statistic approximately equal to 1 results in a true null hypothesis. The null hypothesis can be rejected when the F-statistic is far from 1, or when the p-value is less than the significance value of 1% or 0.01. Table 13 shows an F-statistic equal to 3.2453 and a p-value less than 0.0001. Thus, the null hypothesis is rejected. These results confirm that the subjective reliability ratings have a high between-subject variance.

The results in Table 12 show how well participants predict the collision avoidance agent’s reliability; however due to the high between-subject variance, it does not accurately depict how well participants distinguish between different reliability rates. Assuming a normal distribution for each participant, individual z-score analysis provides a normalization method that enables the examination of how well participants differentiate between reliability rates. A z-score indicates how many standard deviations an observation is from the mean. Performing z-score analysis by individual produces a normalized scale across participants, which allows better analysis on ratings between programmed reliability rates. For example, if one participant provided the ratings: 100%, 90%, and 80%, while another participant provided: 70%, 50%, and 30%, then both participants would result in the same z-scores of 1, 0, -1, despite the high between-subjects variability.

The z-score of a subjective reliability rating X is calculated using the formula seen in Equation 6, where μ and σ represent the mean and standard deviation of the population, respectively.

$$\text{Z-Score} = \frac{X - \mu}{\sigma} \quad (6)$$

Table 14 holds the average subjective reliability rating, corresponding z-scores, and their 95% confidence intervals from 24 participants across 6 blocks of various programmed reliability rates. The table also contains the z-scores of the programmed

reliability rates, which provide expected values for the subjective z-scores.

Table 14. Human-subject experiment results of the subjective reliability and corresponding z-score across each programmed reliability rate.

Programmed Reliability	Programmed Reliability (Z-Score)	Subjective Reliability		Subjective Reliability (Z-Score)	
		Mean	95% CI	Mean	95% CI
100% B1	0.90	76.4%	± 8.74	0.63	± 0.296
95%	0.49	70.6%	± 7.26	0.31	± 0.259
90%	0.07	60.8%	± 6.64	-0.18	± 0.218
80%	-0.76	54.4%	± 5.54	-0.55	± 0.249
70%	-1.60	47.3%	± 5.76	-1.00	± 0.233
100% B6	0.90	78.0%	± 7.53	0.78	± 0.331

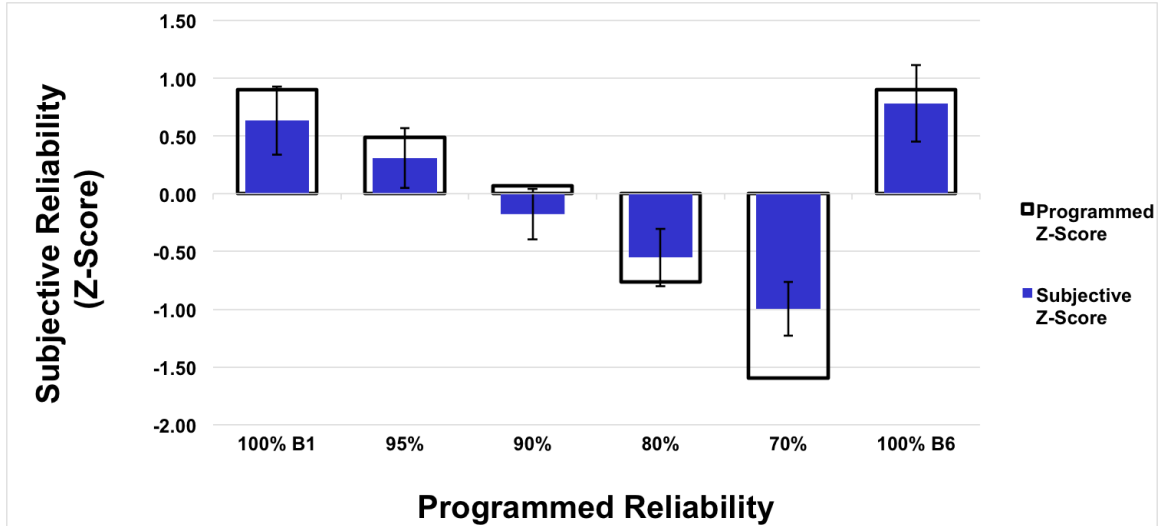


Figure 15. Z-Scores of subjective reliability ratings across each programmed reliability rate.

Figure 15 displays the results from Table 14. The results conclude that participants better distinguish the reliability of the agent at higher reliability rates. This is shown by looking at areas where confidence intervals do not overlap. Specifically, the z-scores of 0.63 ± 0.296 and 0.78 ± 0.331 for the 100% programmed reliability rates of block 1 and 6 do not overlap with the -0.18 ± 0.218 z-score for the 90% reliability rate. The lack of overlap confirms that participants significantly differentiate a 10%

difference from the highest reliability agent. Meanwhile at lower reliability rates, participants cannot significantly separate the 90% and 80%, or the 80% and 70% reliability rates. However, the z-scores of -0.18 ± 0.218 and -1.00 ± 0.331 for the 90% and 70% rates do not overlap. This lack of overlap confirms that participants significantly differentiate a 20% difference from the lowest reliability agent.

Figure 15 also shows calibration in effect when looking at the difference between the 100% reliability ratings in block 1 and block 6. Although not significant, the z-score of 0.78 ± 0.331 in block 6 shows a large overlap with the expected z-score of 0.90, while the z-score of 0.63 ± 0.296 shows little overlap. The increased overlap in block 6 suggests that participants become better calibrated to the agent’s capabilities after experiencing degraded reliability rates.

4.2 Subjective Trust vs. Programmed Reliability

A participant’s subjective trust holds their internal opinion or judgment of an autonomous agent [9]. In order to capture a participant’s subjective trust in the collision avoidance agent, they were asked to indicate their level of trust in the automated agent after each block using the Likert-type scale in Figure 13 of Section 3.6.1.1. The original scale held a value of 1 for “high trust” and 5 for “high distrust.” However, these values were reversed in analysis to facilitate comprehension such that 5 corresponds to “high trust.” Since this scale only ranges from 1-5, standard score analysis is not necessary. After performing the human-subject experiment, there existed 144 recorded subjective trust ratings. Table 15 holds the average subjective trust rating and their 95% confidence intervals from 24 participants across 6 blocks of various programmed reliability rates.

Figure 16 displays the results from Table 15. Looking at confidence intervals, the participant’s trust rating from block 1 (4.29 ± 0.363) does not overlap with the

Table 15. Human-subject experiment results of the subjective trust across each programmed reliability rate.

Programmed Reliability	Subjective Trust	
	Mean	95% CI
100% B1	4.29	± 0.363
95%	3.79	± 0.441
90%	3.63	± 0.388
80%	3.25	± 0.396
70%	2.83	± 0.385
100% B6	4.38	± 0.330

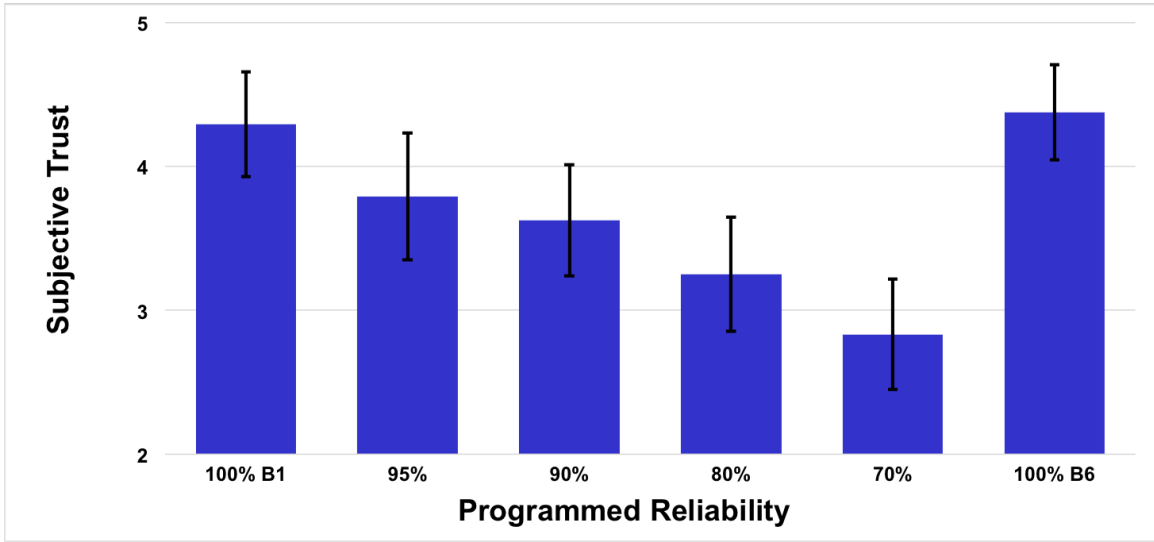


Figure 16. Human-subject experiment results of the subjective trust across each programmed reliability rate.

80% (3.25 ± 0.396) and 70% (2.83 ± 0.385) programmed reliability ratings, but overlaps with the 90% (3.63 ± 0.388) agent. Thus, participants initially trust the 100% agent significantly more than the 80% and 70% agents, but not the 90% agent. However when they experience the 100% agent again in block 6, the participants trust the 100% (4.38 ± 0.330) reliability agent significantly more than the 90% (3.63 ± 0.388) reliability agent, since their confidence intervals do not overlap. This demonstrates that as participants gain experience with reduced reliability, they trust the agent more at a 100% reliability. Combining this discovery with the finding in Section

4.1 that participants become better calibrated to agent capability, suggests that the participants began to develop appropriate trust in the agent after gaining experience with reduced reliability.

4.3 Reliance Rate vs. Programmed Reliability

A participant’s reliance rate shows how often he or she accepts the automation’s non-action. If participants have a high reliance rate, they display trust that the collision avoidance agent will do its job. If participants have a low reliance rate, they display a lack of trust and perform the agent’s job him or herself. In order to capture a participant’s reliance in the collision avoidance agent, three in-game events were captured: agents re-routing a collision-bound ship, agents ignoring an imminent collision, and participant’s re-routing a collision bound ship. The first two in-game events exhibit participants’ acts of reliance in the agent, while the third shows acts of unreliance. These measurements involve two assumptions. The first assumption originates from a finding in Goodman *et al.* [19], which also involved a study in *Space Navigator*, that a participant requires an average of 2.6 seconds after a ship spawns before they are able to operate on it. This finding leads to the assumption that participants require a ship to exist for at least 2.6 seconds in order to show an act of reliance or unreliance. Therefore, all acts of reliance or unreliance that included a ship with a lifespan less than 2.6 seconds were removed. The second assumption involves whether acts of reliance or unreliance by a participant are intentional or accidental. Since there exists no definite method to decipher between the two, all acts are considered intentional. Without this assumption, there would be no means to measure a participant’s reliance rate.

After removing data according to the assumptions, there remained 20,881 acts of reliance and 11,720 acts of unreliance across 144 total blocks. Table 16 holds the

average reliance rates and their 95% confidence intervals from 24 participants across 6 blocks of various programmed reliability rates.

Table 16. Human-subject experiment results of the reliance rate across each programmed reliability rate.

Programmed Reliability	Reliance Rate	
	Mean	95% CI
100% B1	68.1%	± 2.91
95%	63.6%	± 3.78
90%	61.6%	± 4.00
80%	58.3%	± 3.26
70%	57.4%	± 3.74
100% B6	64.0%	± 3.60

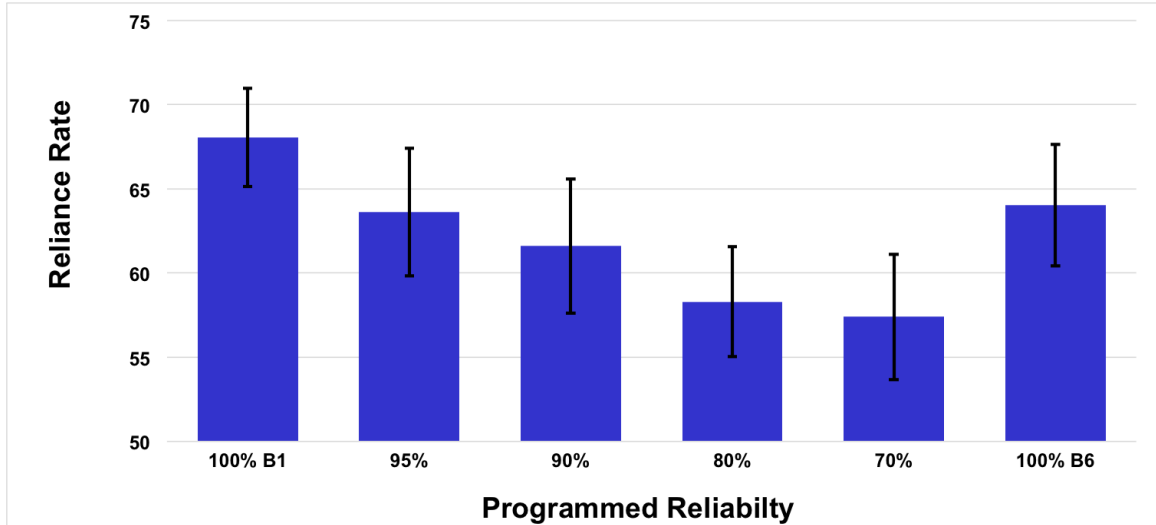


Figure 17. Human-subject experiment results of the reliance rate across each programmed reliability rate.

Figure 17 displays the results from Table 16. Looking at the reliance rates in block 1, participants rely upon the 100% reliability agent significantly more than the 80% and 70% agents, since the confidence interval for block 1 ($68.1\% \pm 2.91\%$) does not overlap with the 80% ($58.3\% \pm 3.26\%$) or 70% ($57.4\% \pm 3.74\%$) confidence intervals. However in block 6, participants do not rely upon the 100% ($64.0\% \pm 3.60\%$) agent significantly more than any reduced reliability agent, as its confidence interval overlaps

with all others. The decrease in reliance on the 100% reliability agent reveals that participants rely upon the agent less after experiencing reduced reliability rates.

4.4 Team Performance

Prior to beginning the experiment, participants were given the task to obtain the highest score possible within *Space Navigator*. Therefore, the average score from each game represents an overall metric of the human-machine team’s performance at a given programmed reliability rate. After performing the human-subject experiment, there existed 288 recorded game scores. Table 17 holds the average scores and their 95% confidence intervals from 24 participants across 6 blocks of various programmed reliability rates.

Table 17. Human-subject experiment results of the average scores across each programmed reliability rate.

Programmed Reliability	Average Score	
	Mean	95% CI
100% B1	10944.0	± 530.87
95%	10446.9	± 568.05
90%	9706.7	± 789.81
80%	9256.5	± 584.33
70%	8102.3	± 689.41
100% B6	11428.5	± 546.47

Figure 18 displays the results from Table 17. Looking at the average scores in block 1, participants only perform significantly better with the 100% agent than the 80% and 70% agents, since the confidence interval for block 1 (10944.0 ± 530.87) does not overlap with the 80% (9256.5 ± 584.33) or 70% (8102.3 ± 689.41) confidence intervals. In block 6, participants paired with the 100% agent continue to outperform the 80% and 70% agents; however, they also significantly outperform the 90% agent, as the confidence interval for block 6 (11428.5 ± 546.47) does not overlap with the 90% (9706.7 ± 789.81) confidence interval. This discrepancy between performance in blocks

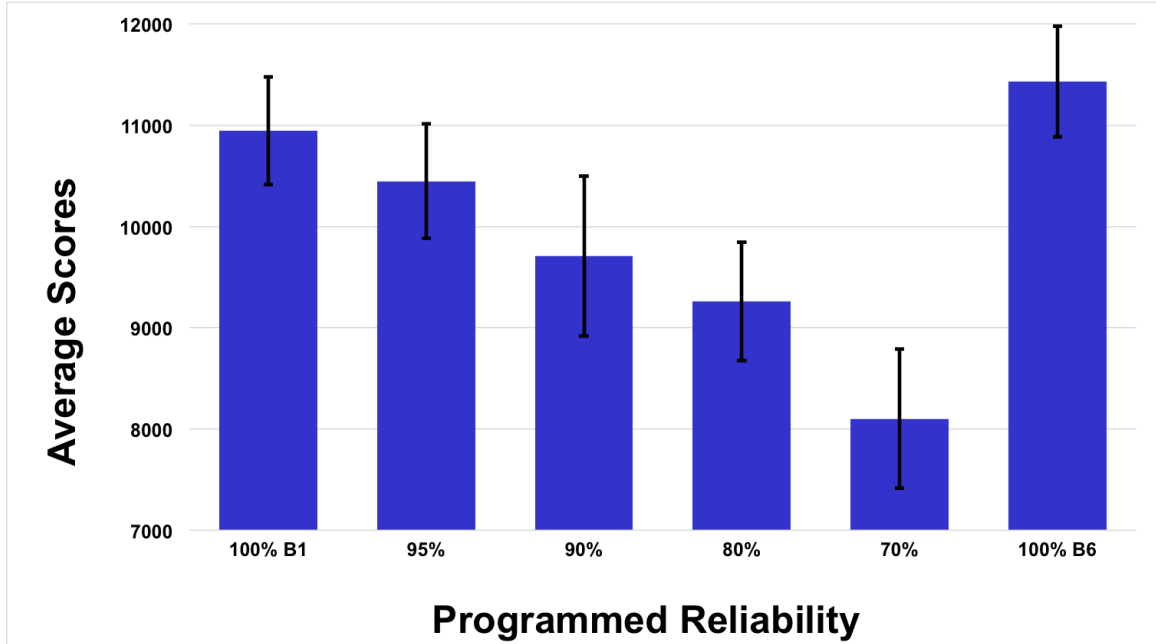


Figure 18. Average scores across each programmed reliability rate.

1 and 6 demonstrates that as participants gain experience with reduced reliability, the human-machine team displays an increase in performance. Combining this discovery with the finding in Section 4.3 that participants rely less upon the agent, suggests that the participants began to develop appropriate reliance in the agent after gaining experience with reduced reliability.

4.5 Correlations and Analysis of Variance

The previous sections analyzed relationships between perceived reliability, perceived trust, and reliance rates against automation with varying reliability. However, they do not explain factors that influence these trust-based metrics. In order to look into this, the remainder of the chapter uses Pearson’s correlations and analysis of variance (ANOVA). Pearson’s correlation measures the linear relationship between variables. Utilizing the Pearson’s correlation function within R software with the trust-based metrics, provides information on whether they are significantly related to

other factors, and if they have a positive or negative relationship. Any factors that are not correlated to the trust-based metrics can be utilized in multiple linear regression to obtain ANOVA results. Utilizing ANOVA within JMP statistics software reports significant factors which cause variance in the trust-based metrics.

The following sections look at correlations and ANOVA results of these trust-based metrics between themselves and 14 potential predictors of trust. These predictors include the average score; ISA and NASA-TLX ratings; number of player and automation routes drawn from each block; participant age, gender, education level, and average propensity to trust in automation and humans from the pre-experiment questionnaire in Appendix A; and use of laptops, tablets, desktop computers, and gaming consoles from the post-experiment questionnaire in Appendix B. The observations of smart phones usage were left out due to lack of variability.

4.5.1 Subjective Reliability.

Table 18 shows all factors that significantly correlate with subjective reliability rating. Subjective reliability unsurprisingly shows a positive correlation with programmed reliability (0.541), average score (0.486), and subjective trust (0.701), while displaying a negative relationship with the ISA (-0.429) and NASA-TLX (-0.510) workload ratings. It also makes intuitive sense that participant’s perceived reliability decreases when they have to draw many routes (-0.166) and increases when the collision avoidance agent draws many routes (0.185). The positive correlation between subjective reliability ratings and daily desktop computer use (0.332) provides an unexpected result. The reason for this is unknown, although may be due to the lack of variability, as 19 of the 24 participants rated that they use a desktop computer daily.

ANOVA requires predictors to be non-collinear. Therefore, only one correlated factor may be used in the analysis. Since the effects of programmed reliability were

Table 18. List of significant correlations with subjective reliability ratings.
(** denotes significance at the 0.01 level, * denotes significance at the 0.05 level)

	Prog. Reliability	Average Score	Subj. Trust	ISA	NASA -TLX	Desktop Use	Number of Player Routes	Number of Agent Routes
Pearson's Coefficient (r)	0.541**	0.486**	0.701**	-0.429**	-0.510**	0.332**	-0.166*	0.185*
Sig. (p)	0.000	0.000	0.000	0.000	0.000	0.000	0.046	0.026

analyzed throughout this research, it was chosen as the correlated ANOVA factor. Predictors that did not correlate to subjective reliability include participant age, gender, education level, propensity to trust in automation and humans, and use of laptops, tablets and gaming consoles. These predictors, along with the programmed reliability were used in forward step-wise regression to create a predictive model in order to generate the ANOVA results seen in 19. The regression model resulted in the following predictors: programmed reliability, propensity to trust in humans, education level, use of gaming consoles, and age. The p-value of <0.0001 provides the rejection of the null hypothesis, showing that at least one of the predictors has a significant effect in the model.

Table 19. ANOVA results for subjective reliability. (* denotes significance at the 0.05 level)

	Df	Sum of Squares	Mean Square	F-Statistic	Pr(>F)	R^2	Adjusted R^2
Model	5	26114.17	5222.83	20.81	<0.0001*	0.43	0.41
Error	138	34641.13	251.02				
C. Total	143	60755.31					

Table 20 provides significance effects for each of the model's predictors. Since each of the p-values lie under the significance level of 0.05, the null hypothesis for each predictor is rejected. Therefore, these predictors provide significant explanation to the variation of subjective reliability. In fact, the adjusted R^2 value of 0.41 in Table

19 confirms that 41% of the variation is explained by these values. The results from Tables 19 and 20 suggest that future work should look into these predictors when analyzing subjective reliability ratings of the collision avoidance agent.

Table 20. Effect tests from the subjective reliability ANOVA. (* denotes significance at the 0.05 level)

	Df	Sum of Squares	F-Statistic	Pr(>F)
Programmed Reliability	1	17769.81	70.79	<0.0001*
Propensity to Trust Humans	1	6473.59	25.79	<0.0001*
Education	1	3863.75	15.39	0.0001*
Use of Gaming Consoles	1	2521.14	10.04	0.0019*
Age	1	1259.07	5.02	0.0267*

4.5.2 Z-Scores of Subjective Reliability.

As previously mentioned, the raw subjective reliability ratings examine how well participants perceive the agent’s reliability, but not how well they distinguish between different levels of reliability. In order to examine this, z-scores of subjective reliability were calculated. Table 21 holds the significant correlations with the z-score of subjective reliability ratings. The z-score ratings show similar results as the raw subjective reliability with positive correlations in programmed reliability (0.685), subjective trust (0.685), average score (0.375), and number of agent routes, as well as negative correlations in the workload ratings of ISA (-0.364) and NASA-TLX (-0.437). However, a correlation with the reliance rate did not show up in the raw subjective reliability ratings. This correlation suggests that participants rely upon the agent more at higher perceived levels of reliability.

Predictors that did not correlate to the z-scores include participant age, gender, education level, propensity to trust in automation and humans, the average number

Table 21. List of significant correlations with z-scores for subjective reliability ratings. (** denotes significance at the 0.01 level, * denotes significance at the 0.05 level)

	Programmed Reliability	Reliance Rate	Subjective Trust	Average Score	ISA	NASA -TLX	Number of Agent Routes
Pearson's Coefficient (r)	0.685**	0.240**	0.685**	0.375**	-0.364**	-0.437**	0.364*
Sig. (p)	0.000	0.000	0.000	0.000	0.000	0.000	0.022

of player draws, and use of laptops, tablets, desktops, and gaming consoles. These predictors, along with the programmed reliability were used in forward stepwise regression to create a predictive model. The regression model, seen in Table 22 resulted in only one predictor: programmed reliability. The p-value of <0.0001 provides the rejection of the null hypothesis, showing that the programmed reliability has significant effects on the z-scores. These results suggest that none of the other predictors explain the variability in z-score values. Although the regression only resulted in one predictor, programmed reliability still provides explanation for 47% of the variance, as seen in the adjusted R^2 value.

Table 22. ANOVA results for Z-Scores of subjective reliability ratings. (* denotes significance at the 0.05 level)

	Df	Sum of Squares	Mean Square	F-Statistic	Pr(>F)	R^2	Adjusted R^2
Model	1	56.35	56.35	125.72	$<0.0001^*$	0.47	0.47
Error	142	63.65	0.45				
C. Total	143	120.00					

4.5.3 Subjective Trust.

Moving on to the next trust-based metric, Table 23 holds the significant correlations with the subjective trust ratings. As expected, there exists a positive correlation between subjective trust and programmed reliability (0.448), raw subjective reliabil-

ity ratings (0.701), z-scores of subjective reliability ratings (0.685), average score (0.348), and the number of agent draws (0.190), while it has a negative correlation with the ISA (-0.376) and NASA-TLX (-0.449) workload ratings. There also exists an interesting negative correlation between subjective trust ratings and age (-0.186). Generally, younger people have more experience with video games such as *Space Navigator*, which may result in higher trust ratings. If this were the case, it would be expected that a significant correlation between subjective trust and use of gaming console exists, unless participants play most of their games on smart phones. Either way, the reason for this correlation is unknown and may be analyzed further in future work.

Table 23. List of significant correlations with subjective trust ratings.
(** denotes significance at the 0.01 level, * denotes significance at the 0.05 level)

	Prog. Reliability	Subj. Reliability	Subj. Reliability (Z-Score)	Average Score	ISA	NASA -TLX	Age	Number of Agent Routes
Pearson's Coefficient (r)	0.488**	0.701**	0.685**	0.348**	-0.376**	-0.449**	-0.186*	0.190*
Sig. (p)	0.000	0.000	0.000	0.000	0.000	0.000	0.026	0.022

Predictors that did not correlate to subjective trust include participant gender, education level, propensity to trust in automation and humans, the average number of player draws, and use of laptops, tablets, desktop computers, and gaming consoles. These predictors, along with the programmed reliability were used in forward stepwise regression to create a predictive model. The regression model, seen in Table 24 contains the following predictors: programmed reliability, education level, use of desktop computers, and gender. The p-value of <0.0001 provides the rejection of the null hypothesis, showing that at least one of the predictors has a significant effect in the model. The model's predictors explain 32% of the variation in subjective trust, as seen in the adjusted R^2 value.

Table 24. ANOVA results for subjective trust. (* denotes significance at the 0.05 level)

	Df	Sum of Squares	Mean Square	F-Statistic	Pr(>F)	R^2	Adjusted R^2
Model	4	57.36	14.34	17.61	<0.0001*	0.34	0.32
Error	139	113.19	0.81				
C. Total	143	170.56					

Table 25 provides significance effects for each of the model’s predictors. The p-values for programmed reliability (<0.0001) and education level (0.0003) reject the null hypothesis, while use of desktop computers (0.0874) and gender (0.1684) do not. Therefore, only programmed reliability and education level provide significant explanation to the variation of subjective trust. While use of desktop computers and gender do not provide significant evidence in this case, they do remain predictors that should be analyzed in future work.

Table 25. Effect tests from the subjective trust ANOVA. (* denotes significance at the 0.05 level)

	Df	Sum of Squares	F-Statistic	Pr(>F)
Programmed Reliability	1	40.62	49.80	<0.0001*
Education	1	11.30	13.88	0.0003*
Use of Desktop Computers	1	2.41	2.96	0.0874
Gender	1	1.56	1.92	0.1684

4.5.4 Reliance Rate.

Moving on to the final trust-based metric, Table 26 holds the significant correlations with the reliance rates. To no surprise, there exists a positive correlation with programmed reliability (0.352) and z-scores of subjective reliability ratings (0.240). However, the remaining correlations were unexpected.

To start, reliance rate showed a negative correlation with the average score (-

Table 26. List of significant correlations with reliance rates.
(** denotes significance at the 0.01 level, * denotes significance at the 0.05 level)

	Programmed Reliability	Subjective Reliability (Z-Score)	Average Score	Age	Gender	Propensity to Trust Automation	Use of Desktop Computers
Pearson's Coefficient (r)	0.352**	0.240**	-0.377**	0.389**	0.195*	-0.179*	-0.198*
Sig. (p)	0.000	0.000	0.000	0.000	0.019	0.032	0.017

0.377). Intuitively, it is expected that players would have a higher score if they relied upon the agent more; however the conclusion in Section 4.4 found that participants perform better as they rely upon the agent less, which is reinforced in the correlation.

The reliance rate shows a negative correlation with the propensity to trust automation (-0.179). Again, this correlation is the opposite of what is expected. The propensity to trust questionnaires were derived from Lin *et. al.* [30], but were not statistically validated. The lack of validation may be the reason for this negative correlation. However, it may be that the participants did not accurately reflect their internal feelings toward automation while taking the questionnaires. For this reason, this relationship should be analyzed in future work.

The reliance rate shows a positive correlation with age (0.389) and gender (0.195). Similar to the subjective trust results, younger people and men often have more experience with video games such as *Space Navigator*, which may be causing these correlations. There also exists a negative correlation with desktop use (-0.198), suggesting that participants rely less upon the agent the more they use desktop computers. This correlation may also be related to gaming experience if participants play on their desktop computers. Regardless, the exact reasons for these correlations are unknown and may be analyzed further in future work.

Predictors that did not correlate to reliance rate include participant education level, propensity to trust in humans, and use of laptops, tablets, desktop comput-

ers, and gaming consoles. Other predictors that did not correlate such as ISA and NASA-TLX were left out as they were influenced by team performance. The chosen predictors, along with the programmed reliability were used in forward stepwise regression to create a predictive model. The regression model, seen in Table 27 contains the following predictors: programmed reliability, education level, propensity to trust in humans, and use of desktop computers, tablets, and gaming consoles. The p-value of <0.0001 provides the rejection of the null hypothesis, showing that at least one of the predictors has a significant effect in the model. The model's predictors explain 29% of the variation in reliance rates, as seen in the adjusted R^2 value.

Table 27. ANOVA results for reliance rate. (* denotes significance at the 0.05 level)

	Df	Sum of Squares	Mean Square	F-Statistic	Pr(>F)	R^2	Adjusted R^2
Model	6	4115.79	685.96	10.76	$<0.0001^*$	0.32	0.29
Error	137	8732.49	63.74				
C. Total	143	12848.27					

Table 28. Effect tests from the reliance rate ANOVA. (* denotes significance at the 0.05 level)

	Df	Sum of Squares	F-Statistic	Pr(>F)
Programmed Reliability	1	1595.95	25.04	$<0.0001^*$
Education	1	1367.52	21.45	$<0.0001^*$
Use of Desktop Computers	1	510.42	8.01	0.0054^*
Propensity to Trust Humans	1	484.82	7.61	0.0066^*
Use of Tablet Computers	1	415.80	6.52	0.0117^*
Use of Gaming Consoles	1	165.58	2.60	0.1093

Table 28 provides significance effects for each of the model's predictors. All of the p-values except for the use of gaming consoles (0.1093) reject the null hypothesis.

Therefore, programmed reliability, level of education, the use of desktop and tablet computers, and propensity to trust in humans all provide significant explanation of the variation in reliance rates. These factors should be analyzed more in future work.

4.6 Chapter Summary

In summary this chapter covered analysis on the relationships in the investigative questions, as well as determining factors that correlate with and explain variance in the trust-based metrics. Analysis on subjective reliability showed that players poorly perceive the collision avoidance agents capability. Z-score analysis on this data explains that participants better distinguish between higher reliability rates than lower and that participants become better calibrated to the agent's capabilities after experiencing reduced reliability rates. Examination on subjective trust confirms that participants trust the 100% agent more after experiencing reduced reliability rates. Combined with the finding that participants become better calibrated to the agent's capabilities, suggests that participants begin to develop appropriate trust after experiencing reduced reliability. Looking into reliance rates and performance, the results found that participants rely upon the 100 100% agent less, but perform better after experiencing reduced reliability rates. These conclusions suggest that participants begin to develop appropriate reliance after experiencing reduced reliability. Finally the outcomes from correlations and ANOVA provide characteristics and trends that explain the trust-based metrics. Chapter V concludes the research by providing a summary of the accomplished goals, key results, and recommendations for future work.

V. Conclusion

This chapter summarizes the research and how its results address the relationship between human reliance and automation with varying levels of reliability. The chapter concludes with recommendations for future research with the collision avoidance agent in order to better understand trust calibration in human-machine teaming.

5.1 Research Question

The purpose of this research was to answer the question “Does the level of automation reliability affect reliance-based trust metrics in the human members of human-machine teams?” Chapter I hypothesized that as a human operator gains experience with reduced reliability, they become better calibrated to the true automation’s capability and limitations; and as a result, the operator will display calibration for appropriate trust in the automation. In the human-subject experiment, reliance-based trust was measured via three different methods: subjective reliability of the agent, subjective trust in the agent, and the participant’s reliance rate in the agent. In order to confirm the hypothesis, three investigative questions were asked:

- “What is the relationship between the human’s perceived automation reliability and the actual automation reliability?”
- “What is the relationship between the human’s perceived trust in automation and the automation reliability?”
- “What is the relationship between the human’s reliance rate in automation and the automation reliability?”

5.2 Research Accomplishments

The first investigative question was answered by comparing average subjective reliability ratings from the 24 participants across the 6 blocks of agent reliability. These results found that participants poorly perceive the actual reliability of the collision avoidance agent; however, findings suggest that they perceive the rates more accurately at higher reliability rates. Since there existed a high between-subjects variance, z-scores of subjective ratings were derived to analyze how well participants distinguish between different reliability rates. These findings showed that participants better distinguish the reliability of the automation at higher reliability rates, and that they show trends of calibration to the automation's true capability after experiencing reduced reliability rates.

The second investigative question was answered by looking at the average subjective trust ratings from the 24 participants across the 6 blocks of agent reliability. The relationship confirmed that participants trust the automation more after experiencing reduced reliability rates. These findings, combined with the conclusion from the z-score analysis, suggest that participants begin to develop appropriate trust in the automation after gaining experience with reduced reliability.

The third investigative question was answered by comparing the average reliance rates from the 24 participants across the 6 blocks of agent reliability. These results demonstrated that participants rely less upon the true automation after experiencing reduced reliability. Analysis on average scores showed that the human-machine team increased performance after experiencing reduced reliability. Combining these findings suggest that participants begin to rely upon the automation appropriately after experiencing reduced reliability rates.

Since participants showed calibration to automation capability and development of appropriate trust and reliance in the automation, the research hypothesis is confirmed.

Although the hypothesis is only confirmed for human-machine teams within *Space Navigator*, the results may have real-world impacts. For example, *Space Navigator* closely resembles the type of task performed by an air traffic control human-machine team. Thus, the results of this research suggest that exposing air traffic controllers to reduced reliability in training may positively influence trust calibration and reduce events of misuse and disuse.

5.3 Research Summary

As stated in Chapter I, the United States Air Force plans on increasing the use and ability of human-machine teaming [16]. However, attaining such a team requires Airmen to be able to develop appropriate trust calibrated to the automation’s capabilities. Many factors influence the trust calibration, including the automation’s reliability. In order to understand how reliability affects trust, there needs to be a way to measure trust. This is achieved via subjective questionnaires of trust and objective measurements of human behaviors related to trust, such as reliance and compliance.

In order to record human reliance in automation, the research required development of a human-reliance measurement system. The chosen system consisted of the tablet-based video game *Space Navigator* with an agent that performed the reliance-based task of collision avoidance. Many collision avoidance methods were analyzed, but the best performing agent was chosen to be a part of the human-reliance measurement system. By comparing performance results to previous human-subject data [2], it was deemed logical for humans to rely upon the newly developed collision avoidance agent. The system was then given varying levels of reliability, which allowed recordings of human reliance against different levels of reliability.

A human-subject experiment was then designed and performed to record human-subject reliance data. The data consisted of subjective and objective trust-based

metrics from 24 participants across 6 blocks of gameplay with varying agent reliability. Subjective measurements included the participant’s perceived reliability and overall feeling of trust in the collision avoidance agent, while objective measurements consisted of the participant’s reliance rate. The final recordings were then analyzed to answer the research question, as outlined above.

5.4 Future Work

The development of the reliance-based measurement system opens the door for further research in trust calibration. The following are recommendations for such research:

- Gather more participants: Since the human-subject experiment only involved 24 participants, many of the results could not provide significant conclusions. Obtaining more data in the experiment may shrink the confidence intervals in the relationship between trust-based metrics and reliability, and could also provide improved correlation and ANOVA results.
- Refine collision detection: The collision detection methodology performs at 98% accuracy. In order to create a better collision avoidance agent, issues with this methodology should be examined and fixed.
- Refine reliance rate: The current definition for reliance rate is rather rudimentary. Other metrics, such as position in the collision list and time until collision, could be factored into the formula to create a better measure of human reliance in the collision avoidance agent.
- Future human-subject studies: Prior to this research, there existed only the line agent in *Space Navigator*. The development of the collision avoidance agent

provides more future human-subject possibilities. These possibilities include research with different collision avoidance methods, with only collision detection, or with both the line and collision avoidance agents.

- Repeat with different human-machine team environments: The hypothesis is only confirmed within the *Space Navigator* environment. However, these results suggest that experience in reduced reliability positively influences trust calibration in true automation. In order to confirm this within other domains, the human-subject experiment should be repeated with other human-machine teams.

Appendix A. Pre-Experiment Questionnaire

The pre-experiment questionnaire is performed by the participants prior to the experiment. It consists of the following questions.

1.1 Demographics Questionnaire

Table 29. Demographics questionnaire filled out prior to the experiment for each participant.

Please specify the hand you primarily use for the following activities.			
1. Writing:	Left	Right	Both
2. Throwing a ball:	Left	Right	Both
3. Using a computer mouse:	Left	Right	Both
What is your gender?			
	Male	Female	Prefer not to respond
What is your highest education level?			
a) Lower than highschool			
b) Graduated from highschool			
c) Some college			
d) Graduated from a vocational or technical school			
e) Graduated from a 4-year college			
f) Some post-graduate education			
g) Master's degree			
h) PhD degree			
i) Professional degree			

1.2 General Trust Questionnaires

The questionnaires in Tables 30 and 31 ask participants to circle the number that best reflects their belief using the following scale:

1 = The statement does not apply to me at all

2 = The statement usually does not apply to me

3 = Most often, the statement does not apply

- 4 = I am unsure about whether or not the statement applies to me,
or it applies to me about half the time
- 5 = The statement applies more often than not
- 6 = The statement usually applies to me
- 7 = The statement always applies to me

1.2.1 Propensity to Trust Automation.

Table 30. General trust questionnaire used to gather a player’s propensity to trust in automation.

1.	I feel that most automated systems would be employed for the participant’s best interest						
	1	2	3	4	5	6	7
2.	Most automated systems are helpful						
	1	2	3	4	5	6	7
3.	Most automated systems are employed for participant well-being						
	1	2	3	4	5	6	7
4.	In general, automated systems are competently programmed						
	1	2	3	4	5	6	7
5.	Most automated systems are capable of meeting participant needs						
	1	2	3	4	5	6	7
6.	I feel that most automated systems can meet the requirements for which they were designed						
	1	2	3	4	5	6	7
7.	I am comfortable relying on information from automated systems						
	1	2	3	4	5	6	7
8.	I feel fine using automated systems since they are generally reliable and accurate						
	1	2	3	4	5	6	7
9.	I always feel confident that I can rely on automated systems to perform as specified						
	1	2	3	4	5	6	7

1.2.2 Propensity to Trust Humans.

Table 31. General trust questionnaire used to gather a player’s propensity to trust humans.

1.	In general, People really do care about the wellbeing of others						
	1	2	3	4	5	6	7

- | | | | | | | | |
|----|---|---|---|---|---|---|---|
| 2. | The typical person is sincerely concerned about the problems of others | | | | | | |
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 3. | Most of the time, people care enough to try to be helpful, | | | | | | |
| | rather than just looking out for themselves | | | | | | |
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 4. | I believe that most professional people do a good job at work | | | | | | |
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 5. | Most professionals are very knowledgeable in their chosen field | | | | | | |
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 6. | A large majority of professional people is competent in their area of expertise | | | | | | |
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 7. | In general, most people keep their promises | | | | | | |
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 8. | I think people generally try to back up their words with their actions | | | | | | |
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 9. | Most people are honest in their dealings with others | | | | | | |
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 |

Appendix B. Post-Experiment Questionnaire

The post-experiment questionnaire is performed by the participants after completion of the experiment. It consists of the following questions:

2.1 Experience with *Space Navigator* Questionnaire

The questionnaire in Table 32 asks participants to circle the number that best reflects their belief using the following scale:

1 = The statement does not apply to me at all

2 = The statement usually does not apply to me

3 = Most often, the statement does not apply

4 = I am unsure about whether or not the statement applies to me,
or it applies to me about half the time

5 = The statement applies more often than not

6 = The statement usually applies to me

7 = The statement always applies to me

Table 32. Space Navigator experience questionnaire capturing the participant’s feelings toward the game and automation.

1.	I feel that Space Navigator’s automation is reliable						
	1	2	3	4	5	6	7
2.	Space Navigator’s automation improved my performance						
	1	2	3	4	5	6	7
3.	Space Navigator’s automation made the game less stressful						
	1	2	3	4	5	6	7
4.	Space Navigator is a difficult game						
	1	2	3	4	5	6	7
5.	Space Navigator’s automation made the game less difficult						
	1	2	3	4	5	6	7
6.	I was frustrated playing Space Navigator without automation						
	1	2	3	4	5	6	7
7.	I was frustrated playing Space Navigator with automation						

	1	2	3	4	5	6	7
8.	Space Navigator's automation made the game less frustrating						
	1	2	3	4	5	6	7

2.2 Background Questionnaires

The questionnaires in Tables 33 and 34 ask participants to circle the number that best reflects their average usage with the following scale:

1 = Daily

2 = 4-6 times a week

3 = 2-3 times a week

4 = Once a week

5 = Less than once a week

6 = Never

2.2.1 Electronic Background Questionnaire.

Using the scale above, participants are asked "In a given year, on average how often do you use the following types of devices?"

Table 33. Electronic experience questionnaire capturing the participant's average usage of electronics.

1.	Laptop computer					
	1	2	3	4	5	6
2.	Tablet computer					
	1	2	3	4	5	6
3.	Smart phone					
	1	2	3	4	5	6
4.	Desktop computer					
	1	2	3	4	5	6
5.	Gaming consoles					
	1	2	3	4	5	6

2.2.2 Gaming Background Questionnaire.

Using the scale above, participants are asked “In a given year, on average how often do you play the following types of games?”

Table 34. Gaming experience questionnaire capturing the participant’s experience with various types of games.

1.	Simulation (Sim City)					
	1	2	3	4	5	6
2.	Role-Playing (WoW)					
	1	2	3	4	5	6
3.	Action (Mario, Donky Kong)					
	1	2	3	4	5	6
4.	First Person Shooter (Halo)					
	1	2	3	4	5	6
5.	Strategy (Civilization)					
	1	2	3	4	5	6
6.	Puzzle (Tetris, Candy Crush)					
	1	2	3	4	5	6
7.	Casual (Angry Birds)					
	1	2	3	4	5	6
8.	Music (Guitar Hero)					
	1	2	3	4	5	6
9.	Sport (Madden Football)					
	1	2	3	4	5	6
10.	Board (Monopoly)					
	1	2	3	4	5	6
11.	Card (Poker, Pinochle)					
	1	2	3	4	5	6
12.	All types of video games					
	1	2	3	4	5	6

Appendix C. Objective In-Game Data Collected

Table 35. All objective events recorded during a game of *Space Navigator*.

Name	Description	Category
User Number	A number identifying the participant the data belongs to.	Run
Run Number	A number identifying the game to which the data belongs.	Run
Highscore	The final score at the end of a game.	Run
Agent Reliability	The reliability rating of the agent during a game.	Run
Human Trajectories	A trajectory drawn by the participant. Includes ship ID/location/destination/age, timestamp, and trajectory points.	Action
Agent Trajectories	A trajectory drawn by the agent. Includes ship ID/location/destination/age, timestamp, and trajectory points.	Action
Acts of Unreliance	When the participant draws a trajectory for a collision-bound ship. Includes ship information for both collision-bound ships, timestamp, time of predicted collision, and place in the collision table.	Action
Ignores	When the agent ignores a predicted collision. Includes ship location for both collision-bound ships, and timestamp	Action
Spawns	When a ship spawns into the game. Includes ship ID and timestamp.	Ship-Life
Screens	When a ship exits the side of the screen. Includes ship ID and timestamp.	Ship-Life
Destinations	When a ship reaches its destination planet. Includes ship ID and timestamp.	Ship-Life/ Point-Based
Collisions	When a ship collides with another ship. Includes ship ID and timestamp.	Ship-Life/ Point-Based
Bonus Captures	When a ship runs into a bonus. Includes ship ID and timestamp.	Point-Based
Enter NFZ	When a ship enters a No-Fly Zone. Includes ship ID and timestamp.	Point-Based
In NFZ	Capture of each second a ship remains in a No-Fly Zone. Includes ship ID and timestamp.	Point-Based
Exit NFZ	When a ship exits a No-Fly Zone. Includes ship ID and timestamp.	Point-Based

Appendix D. Informed Consent Document

INFORMATION PROTECTED BY THE PRIVACY ACT OF 1974

**Informed Consent Document
For
Automated Agent Reliability and its Effects on Reliance**

AFIT/ENG, WPAFB

Principal Investigator: Capt Jason M. Bindewald, Ph.D., 937-255-3636, AFIT/ENG
jason.bindewald@afit.edu

Associate Investigators: Dr. Michael E. Miller, Civ, Professor, AFIT/ENV, 937-255-3636,
michael.miller@afit.edu
Maj Christina F. Rusnock, Assistant Professor, AFIT/ENV, 937-
255-3636x4611, christina.rusnock@afit.edu
2Lt Jake M. Spuller, Student, AFIT/ENG, 937-255-3636,
jake.spuller@afit.edu
2Lt Christopher J. Garnick, Student, AFIT/ENG, 937-255-3636,
christopher.garnick@afit.edu
2Lt Anthony Hillesheim, Masters Student, AFIT/ENV,
anthony.hillesheim@afit.edu, 937-255-3636
Jayson Boubin, Intern, AFIT/ENV, 513-406-0144,
boubinjc@miamioh.edu, ORISE

1. **Nature and purpose:** You have been offered the opportunity to participate in the “Automated Agent Reliability and its Effects on Reliance” research study. Your participation will occur at the Wright-Patterson AFB AFIT/ENV HSI lab.

Specifically, this study aims to answer the question, “How does the inherent reliability of automated agents affect the reliance of participants on the automation?” This will be performed in the context of an automated agent that provides collision avoidance. The research is expected to be published in a peer-reviewed academic outlet

The time requirement for each volunteer subject is anticipated to be a total of one session of approximately 2.5 hours. A total of approximately 24-48 subjects will be enrolled in this study. Subjects must be able to draw lines on a tablet computer and must not be color-blind.

2. **Experimental procedures:** Each session will include short questionnaires, a computer route-drawing task in the *Space Navigator* environment, and verbal interview questions.

The study will take place in a single lab session. You will complete a questionnaire before and after a 1.5 hour gaming session. You will play a tablet-based route creation game in the following phases: five five-minute training sessions to become familiar with the game followed by 50 10-second baseline *Space Navigator* instances, followed by 12 experimental

game play sessions, and then the 50 baseline instances will be repeated. After each session you will be asked to provide information about your thoughts and experience.

3. **Discomfort and risks:** Those typical of normal office/computer work of eye, wrist and hand movement. You may opt out of the study at any time. Additionally, you will be given an opportunity to review information provided and make a reasoned judgment of the risks of divulging such information.
4. **Precautions for female subjects or subjects who are or may become pregnant during the course of this study:** There are no special precautions needed.
5. **Benefits:** You are not expected to benefit directly from participation in this research study. This study's findings will enable the creation of better computational agents to interact in human-machine teams. This directly addresses the DoD and Air Force priorities of the Third Offset to enable better human-machine teaming by learning about how people trust machine task partners.
6. **Compensation:** If you are active duty military you will receive your normal active duty pay. Otherwise, you will receive no other compensation directly from participation in this study.
7. **Alternatives:** Your alternative is to choose not to participate in this study. Refusal to participate will involve no penalty or loss of benefits to which you are otherwise entitled. You may discontinue participation at any time without penalty or loss of benefits to which you are otherwise entitled. Notify one of the investigators of this study to discontinue.
8. **Entitlements and confidentiality:**
 - a. Records of your participation in this study may only be disclosed according to federal law, including the Federal Privacy Act, 5 U.S.C. 552a, and its implementing regulations and the Health Insurance Portability and Accountability Act (HIPAA), and its implementing regulations, when applicable, and the Freedom of Information Act, 5 U.S.C. Sec 522, and its implementing regulations when applicable. The only personal information obtained during this study is your name/signature on this informed consent form and audio recording during the research, which will all be stored in a locked cabinet in an office that is locked when not occupied or in electronic files which will be password protected and stored only on a secure server. It is intended that the only people having access to your information will be the researchers named above and this study's Medical Monitor or Consultant, the AFRL Wright Site IRB, the Air Force Surgeon General's Research Compliance office, the Director of Defense Research and Engineering office or any other IRB involved in the review and approval of this protocol. After 3 years, or when no longer needed for research purposes, the data collected in this study will be destroyed in a secure manner (shredding). Complete confidentiality cannot be promised, in particular for military personnel, whose health or fitness for duty information may be required to be reported to appropriate medical or command authorities. If such information is to be reported, you will be informed of what is being reported and the reason for the report.

- b. The decision to participate in this research is completely voluntary on your part. No one may coerce or intimidate you into participating in this program. You are participating because you want to. Capt Jason M. Bindewald, or an associate, has adequately answered any and all questions you have about this study, your participation, and the procedures involved. Capt Jason M. Bindewald can be reached at 937-255-3636. Capt Jason M. Bindewald, or an associate will be available to answer any questions concerning procedures throughout this study. If significant new findings develop during the course of this research, which may relate to your decision to continue participation, you will be informed. If you have any questions or concerns about your participation in this study or your rights as a research subject, please contact AFRL IRB personnel at (937) 904-8100 or AFRL.IR.ProtocolManagement@us.af.mil.

STUDY PARTICIPATION AGREEMENT/CONSENT

Taking part in this research study is completely voluntary. Your signature below shows that:

- **You agree to be in this study**
- **The researcher has explained the study to you and you have read and understand the information you have been given**
- **You were given the opportunity to ask questions about the study and all of your questions have been answered to your satisfaction**
- **You understand that signing this consent does not take away any of your legal rights**

You will be given a copy of this signed consent form for your records

Volunteer Signature _____ **Date** _____

Volunteer Name (printed) _____

Advising Investigator Signature _____ **Date** _____

Investigator Name (printed) _____

Witness Signature _____ **Date** _____

Witness Name (printed) _____

Privacy Act Statement

Authority: We are requesting disclosure of personal information.. Researchers are authorized to collect personal information on research subjects under The Privacy Act-5 USC 552a, 10 USC 55, 10 USC 8013, 32 CFR 219, 45 CFR Part 46, and EO 9397, November 1943.

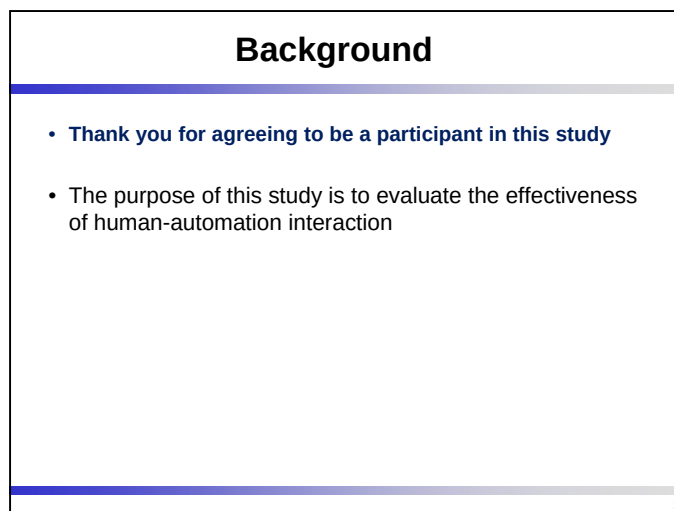
Purpose: It is possible that latent risks or injuries inherent in this experiment will not be discovered until some time in the future. The purpose of collecting this information is to aid researchers in locating you at a future date if further disclosures are appropriate.

Routine Uses: Information may be furnished to Federal, State and local agencies for any uses published by the Air Force in the Federal Register, 52 FR 16431, to include, furtherance of the research involved with this study and to provide medical care.

Disclosure: Disclosure of the requested information is voluntary. No adverse action whatsoever will be taken against you, and no privilege will be denied you based on the fact you do not disclose this information. However, your participation in this study may be impacted by a refusal to provide this information.

Appendix E. Training Information

5.1 Training Slides



Study Overview 1

- You will play Space Navigator, an air traffic control-style game, seeking to **earn the highest score possible**
- Each game consists of navigating a space ship to the planet of the corresponding color
- With the exception of the first warm-up game, all games will have an automated aid designed to assist you in avoiding the **next imminent collision**
- After each block of two games, you will be required to answer a set of questions

3

Study Overview 2

- First Implicit Association
- Space Navigator
 - Warm-up Sessions
 - One manual game followed by intro to TLX, ISA
 - Three games with automated assistance followed by intro to reliability/trust questions
 - Experimental Sessions
 - Six blocks (each 2 games) with automated assistance
- Second Implicit Association Task
- Final Questionnaire

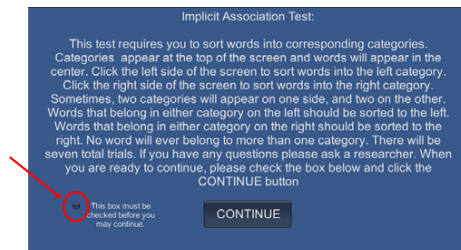
4

Implicit Association Test

- The Implicit Association Test examines your attitudes and preferences
- In this portion of the experiment, we will ask you to answer some questions about your thoughts on automated agents and systems
 - Trials 1-3 and 5-6 are for practice
 - Trials 4 and 7 record actual scores
- Instructions are available on the tablet which explain the specific aspects of the test
- Be sure to keep the tablet on the desk when completing the IAT

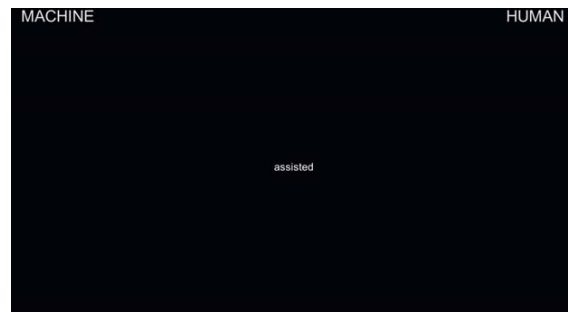
5

Implicit Association Test



6

Implicit Association Test



use both hands to press sides faster

7

Go To Tablet

- *Do you have any questions?*
- *Please return to the tablet to complete IAT*



8

Space Navigator

9

Scoring

Points Gained

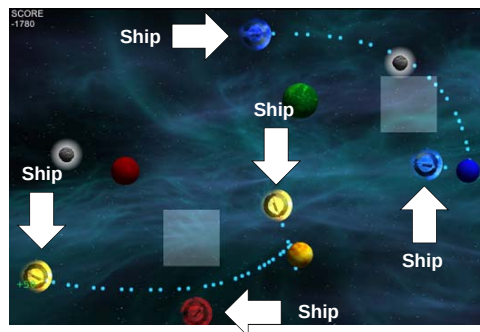
- Ship reaches planet
 - +100
- Run into bonus
 - +50

Points Lost

- Ships collide
 - -100 /ship (-200/crash)
- Ships travel through no-fly zone
 - -10/second

10

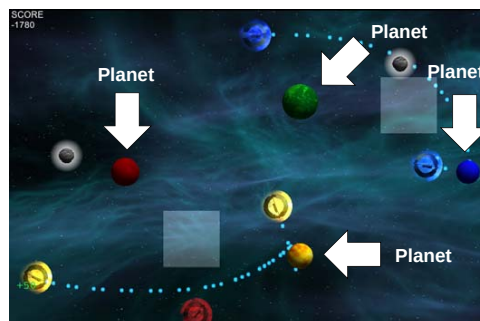
How to Play: Ships



Ships will spawn onto the screen and your goal is to touch and hold the ship then move it to the planet of the same color.

11

How to Play: Planets

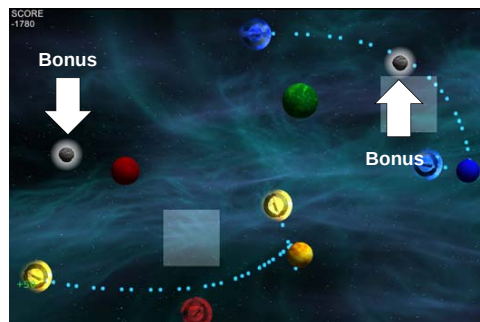


The ship will move along the trajectory that you drew

Make sure to avoid collisions between ships; otherwise, points will be subtracted from your score

12

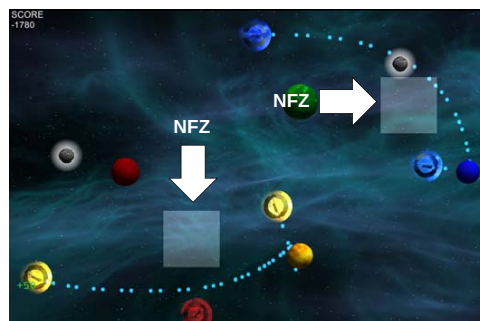
How to Play: Bonuses



Pick up bonuses (stationary) by creating a route that passes through them

13

How to Play: No-Fly Zones (NFZs)



Ships flying through no-fly zones will result in points being deducted

14

Scoring

Points Gained

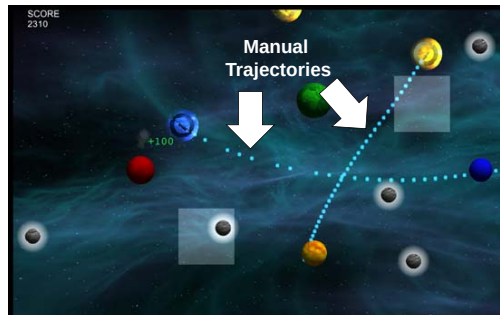
- Ship reaches planet
 - +100
- Run into bonus
 - +50

Points Lost

- Ships collide
 - -100 per ship
 - Ex: Two ships crash = -200
 - Ex: Three ships crash = -300
- Ships travel through no-fly zone
 - -10 per second per ship
 - Ex: Three ships in one no-fly zone = -30 per second

15

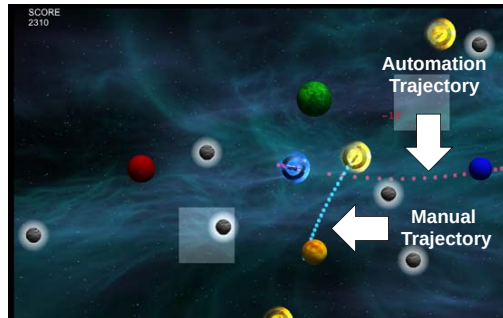
How to Play: Manual Trajectory



Trajectories
drawn by you
will be blue

16

Automation Trajectory



Trajectories drawn by the automated aid will be red

17

Any Questions?

- Do you have any questions on how to play the game?
- *Start the first 2 minute warm-up game, which has no automation*
- Return to slides as soon as game is complete, to review ISA and NASA-TLX



18

Between Game Questionnaire

- ISA Rating: Select ONE rating that best indicates your workload for the just-completed level:
 1. **Under-Utilized:** Nothing to do. Rather boring.
 2. **Relaxed:** More than enough time for all tasks. Active on the task less than 50% of the time.
 3. **Comfortably Busy Pace:** All tasks well in hand. Busy but stimulating pace. Could keep going continuously at this level.
 4. **High:** Non-essential tasks suffering. Could not work at this level very long.
 5. **Excessive:** Behind on tasks. Losing track of the full picture.

19

ISA

ISA Rating:
Select ONE rating that best indicates your workload for the just-completed level

1. Under-Utilized: Nothing to do. Rather boring.
2. Relaxed: More than enough time for all tasks. Active on the task less than 50% of the time.
3. Comfortably Busy Pace: All tasks well in hand. Busy but stimulating pace. Could keep going continuously at this level.
4. High: Non-essential tasks suffering. Could not work at this level very long.
5. Excessive: Behind on tasks. Losing track of the full picture.

 This box must be checked before answers will save.

Save Answers

20

ISA

- *Please return to the tablet to complete ISA*



21

NASA-TLX Questionnaire

- NASA-TLX is a subjective workload measure
- It is very important to review the following information carefully

22

NASA-TLX

NASA TLX Questionnaire: Rate each category from 0(Low) to 100(High) that indicates your experience with the just-completed level

1. Mental Demand: How much mental and perceptual activity was required? Was the task easy or demanding, simple or complex? Rate with value between 0-100 Invalid input.	4. Frustration Level: How irritated, stressed, and annoyed versus content, relaxed, and complacent did you feel during the task? 100 Input Valid: 100
2. Physical Demand: How much physical activity was required? Was the task easy or demanding, slack or strenuous? 50 Input Valid: 50	5. Effort: How hard did you have to work (mentally and physically) to accomplish your level of performance? 50 Invalid input.
3. Temporal Demand: How much time pressure did you feel due to the pace at which the tasks or task elements occurred? Was the pace slow or rapid? 0 Input Valid: 0	6. Performance: How successful were you in performing the task? How satisfied were you with your performance? 100 Invalid input.

☒ This box must be checked and inputs must be valid before answers will save.

Save Answers

23

TLX – Definitions and Directions

- To complete the TLX:
 - Fill in each category with a valid input (number 0-100)

1. Mental Demand: How much mental and perceptual activity was required? Was the task easy or demanding, simple or complex? Rate with value between 0-100 Invalid input.	Mental Demand: How much mental and perceptual activity was required (e.g. thinking, deciding, calculating, remembering, looking, searching, etc)? Was the task easy or demanding, simple or complex, exacting or forgiving?
1. Mental Demand: How much mental and perceptual activity was required? Was the task easy or demanding, simple or complex? 50 Input Valid: 50	Physical Demand: How much physical activity was required (e.g. pushing, pulling, turning, controlling, activating, etc)? Was the task easy or demanding, slow or brisk, slack or strenuous, restful or laborious?
50 Invalid Input. ❌	Temporal Demand: How much time pressure did you feel due to the rate of pace at which the tasks or task elements occurred? Was the pace slow and leisurely or rapid and frantic?
	Frustration Level: How insecure, discouraged, irritated, stressed and annoyed versus secure, gratified, content, relaxed and complacent did you feel during the task?

24

TLX – Definitions and Directions

- To complete the TLX:
 - Once every category has a valid input, check box and save

Effort:
How hard did you have to work (mentally and physically) to accomplish your level of performance?

Performance:
How successful do you think you were in accomplishing the goals of the task set by the experimenter (or yourself)? How satisfied were you with your performance in accomplishing these goals?

3. Temporal Demand: How much time pressure did you feel due to the pace at which the tasks or task elements occurred? Was the pace slow or rapid?

6. Performance: How successful were you in performing the task? How satisfied were you with your performance?

50

Input Valid: 50

50

Input Valid: 50

- This box must be checked and inputs must be valid before answers will save.

Save Answers

25

TLX – Important Note

- The **Performance** category of the TLX has a “**Good**” value when **high**, and a “**Poor**” value when **low**
- However, the **remaining categories** have a “**Good**” value when **low**, and a “**Poor**” value when **high**
 - **Mental/physical/temporal demands, effort, and frustration**
- Take a moment to reflect on why this switch makes sense
 - **High** performance should be associated with **low** workload ratings in the other categories

26

NASA-TLX

- Do you have any questions on how to complete the NASA-TLX?
- *Please return to the tablet to complete the NASA-TLX*



27

Warm-up w/ Automation

- Now you will begin playing with the autonomous aid
- The automation's job is to avoid the next imminent collision
- Selects a collision-bound ship and makes it take a "step-back" before continuing on its path/trajectory
- *Please return to the tablet to complete the next three 3-minute warm-up games*



28

Reliability and Trust

- After each block in the main experiment, you will be tasked to answer two questions on reliability and trust
- You will be completing these on the tablet
- Again, only consider the block you've most recently completed

29

Reliability Question

Based on your experience playing the previous block, please indicate how reliable you believe the automation to be from 0 to 100.

What percentage of imminent collisions do you believe were prevented by the automation in the past block?

Invalid input.

 This box must be checked and input must be valid before answers will save

Save Answers

30

Trust Question

Based on your experience playing the last block, please indicate the extent to which you trusted the automated agent.

- 1) I highly trusted the automated agent.
- 2) I somewhat trusted the automated agent.
- 3) I neither trusted nor distrusted the automated agent.
- 4) I somewhat distrusted the automated agent.
- 5) I highly distrusted the automated agent.

1 = High Trust

5 = Low Trust

☐ This box must be checked
before answers will save.

Save Answers

31

Any Questions?

- Do you have any questions on how to complete the reliability or trust ratings?
- *Perform the reliability and trust questions*
- *Return to slides when questions are complete*



32

Main Experiment

- You will complete six blocks with automated assistance each consisting of two 5-minute games
- The reliability of the automation **might** be reduced
 - A collision-bound ship may not take a “step-back”
- After each block, you will complete reliability and trust ratings, ISA and NASA-TLX
 - These should only reflect your experience in the **block you just completed**

33

Any Questions?

- Do you have any remaining questions?
- *Perform the 6 blocks of the experimental session*
- *Perform the Second Implicit Association Test*
- *Return to slides when complete*



34

TLX – Comparisons

- To complete the comparisons component of the TLX:
 - Click the factor that was the more significant driver of workload
 - **Think, while formulating your opinion of workload on playing the game, which factor was more relevant?**
 - *For example, mental or physical demand?*
 - Was the level of mental demand you exerted influencing your overall workload more, or were you more focused on the physical demands of the task?

Mental Demand

or

Physical Demand

How much mental and perceptual activity was required (e.g. thinking, deciding, calculating, remembering, looking, searching, etc)? Was the task easy or demanding, simple or complex, exacting or forgiving?

How much physical activity was required (e.g. pushing, pulling, turning, controlling, activating, etc)? Was the task easy or demanding, slow or brisk, slack or strenuous, restful or laborious?

35

Final Tablet Tasks

- Do you have any questions on how to complete the NASA-TLX pairings?
- *Return to the tablet to complete the NASA-TLX Pairings*



36

Post-Experiment Questionnaire

- Return to the questionnaire to answer a few questions about the game.
- Once you complete the questionnaire, you are free to go.



37

5.2 Training Script

INTRO/OVERVIEW

Thank you for participating in our research today!

Before we begin, I want to let you know that we will be in this room for approximately 2 hours. Bathroom breaks may be taken in between trials but are discouraged. So if you think you may need to use the restroom, please do so now.

I would also like to ask you to turn off or silence all electronic devices that may cause an interruption and place them as well as any watches **(in the designated area)**. Thank you.

This study will help improve our understanding of how people play a video game with an imperfect automated agent. First, you will take a color blindness test then complete a task. You will be asked to answer some questions about yourself, such as your gender, age, education, preferences, and characteristics. The study should take approximately 2 hours to complete. Before beginning, you will read the study consent form and decide whether or not to participate.

Color Test

Since this game involves guiding colored ships to their matching colored planet, we require a color blindness test before beginning the experiment. The Ishihara Color blindness test involves looking at a pattern composed of many colored dots to determine what number is shown. Are you ready to begin?

Answers: 12, 8, 5, 29, 74, 7, 45, 2

Pass:

Great, you passed the test. Let us move on with the experiment.

Fail:

I'm sorry, but you did not pass this test. This is not a diagnosis of colorblindness; however, we cannot continue on with the experiment. Thank you for coming out today.

Pre-Study Questionnaire

Okay, if you turn your attention to the tablet in front of you, we will move on to the pre-study questionnaire. Please fill out the questionnaire and let me know when you are finished. So you know, it is touchscreen so feel free to use that capability to aid you.

Slide 2 (Background)

Again, thank you for your participation.

The purpose of this study is to evaluate the effectiveness of human-automation interaction. In particular, when automation is not always perfect.

Slide 3 (Study Overview 1)

In overview, you will play Space Navigator, an air-traffic control-like tablet video game, where your goal is to earn the highest score possible. The basic concept of the game is to safely navigate space ships to their home-planet or planet with a corresponding color.

In all but the first game, an automated agent will be aiding you. This agent's job is to avoid the next imminent ship-to-ship collision. After each block of two games, you will answer questions based on your experience in those past two games.

Slide 4 (Study Overview 2)

Here is a complete overview of what you will be doing today. The slides will explain each component in further detail.

Slide 5 (IAT)

The first task you will take part in is the Implicit Association Test. This test examines your attitudes and preferences. During this portion, you will complete a total of 7 trials. Trials 1-3 and 5-6 are for practice. And trials 4 and 7 record actual scores.

Slide 6 (IAT)

Complete instructions are available on the tablet. Be sure to keep it on the desk while completing this component. These are the instructions you will see when you begin.

Slide 7 (IAT)

To reiterate what they are saying: a word will appear in the middle of the screen such as “assisted” seen here. The word will always be associated with a specific category located on the left or right side of the screen. You are tasked to use both hands and press on the side of the tablet that corresponds to the correct answer. If you believe the answer is the left-category, use your left hand to press on the left side of the screen. And if you believe the answer is the right-category, use your right hand to press on the right side. If you choose correctly, a new word will appear in the center. If you choose incorrectly, the word will turn red. You can move on to the next word by pressing on the correct side.

Slide 8 (Go To Tablet)

Do you have any questions?

Go ahead and perform the IAT.

TIP: You do not have to click on the word on the left/right side, just the general side of the screen.

Slide 9 (Space Navigator)

Space Navigator is the tablet-based video game that you will be working with today. Your task is to achieve the highest possible score.

Slide 10 (Scoring)

Points are gained and lost as shown on the screen. We will now walk through each of these components and then review these points again.

Slide 11 (Ships)

Ships will spawn onto the screen and your goal is to touch and hold the ship to move it to the planet of the same color. Any questions?

Slide 12 (Planets)

The ship will move along the trajectory that you drew. If it reaches the stationary destination planet then you will gain 100 points.

Make sure to avoid collisions between ships; otherwise, 100 points per ship will be subtracted from your score. To clarify if two ships collide, you will lose 200 points. If three collide, you will lose 300 points. Any questions?

Slide 13 (Bonuses)

You can pick up stationary bonuses by creating a route that passes through them. If a ship runs into the bonus, you will gain 50 points. Any questions?

Slide 14 (NFZs)

The last score-based component is a no-fly zone. For each ship flying through one of two no-fly zones, there will be a loss of 10 points per second. So if you have one ship flying through a no-fly zone, and two ships flying through the other no-fly zone, then you will lose 30 points per second while they are in there. Any questions?

Slide 15 (Scoring)

Here are the scores that you previously saw. Do you have any questions?

Slide 16 (Human Trajectories)

Trajectories that are drawn by you will be blue, while trajectories that are drawn by the automation will be red.

Slide 17 (Automation Trajectories)

The automated agent's job is to avoid the **next** imminent collision. It does so by selecting a collision-bound ship, and making it take a "step-back" before continuing on with its original path or trajectory. These slides show an example of the automation's capability.

Slide 18 (Questions)

Do you have any questions?

You will now play one 2-minute game of Space Navigator. This game does not have any autonomous aid. Go ahead and start your first warm-up game when you are ready.

Slide 19 (Between Game Questionnaires)

After each block of games in the main experiment and after your next block of warm-up games with automation, you will be tasked to perform the ISA and NASA-TLX questionnaires.

The ISA asks you to select a workload rating from 1 to 5 where the numbers correspond to the statements on the slide. So choosing a 1 means you felt like you had a very little workload, while 5 reflects a very high workload.

Slide 20 (ISA)

On the tablet, it will look like this. In order to move on, select the rating you feel fits best, check the box in the lower left corner, and save your answers. It is important that this rating only reflects your most recent block of games.

Slide 21 (ISA)

Any questions?

Go ahead and perform the ISA.

Slide 22 (NASA-TLX Questionnaire)

The NASA-TLX also asks you to rate various factors of workload.

Slide 23 (NASA-TLX)

On the tablet it looks like this.

Slide 24 (TLX Definitions and Directions)

To complete the task, you must provide a rating from 0-100 for each category.

4 of the 6 categories are listed on the slide. Take a moment to read these and let me know if you have any questions before we move on to the remaining 2.

Slide 25 (TLX Definitions and Directions)

The remaining two categories are on the slide. Take a moment to read these and let me know if you have any questions. In order to move on, provide a valid input for each category, check the box in the lower left corner, and save your answers. It is important that this rating only reflects your most recent block of games.

Any questions?

Slide 26 (TLX Important Note)

It is also important to note that the performance category corresponds to “good” when rated high and “bad” when rated low, while the remaining categories hold the opposite truth. Take a moment to reflect on why this makes sense and let me know if you have any questions.

Slide 27 (NASA-TLX)

Do you have any questions on how to complete the NASA-TLX?
Please complete the questionnaire.

Slide 28 (Warm-up w/ Automation)

Now you will begin playing games with automation. The automation’s job is to avoid the next imminent collision. It does so by selecting a collision-bound ship, and making it take a “step-back” before continuing on with its original path or trajectory. Do you have any questions on how the autonomous agent will aid you?

Go ahead and return to your tablet to complete three 3-minute warm-up games with automation.

Slide 29 (Reliability & Trust)

After each block in the main experiment, you will be tasked to answer two questions on reliability and trust.

Similarly to the ISA and NASA-TLX ratings, you will only consider the most recent block in these ratings.

Slide 30 (Reliability)

The reliability question is seen on the slide. Please take a moment to read it.

Note 0% means the automation prevented 0 imminent collisions, while 100% reflects the automation preventing all imminent collisions.

Do you have any questions?

Slide 31 (Trust)

The trust question is seen on the slide. Please take a moment to read it.

Note selecting a 1 reflects you highly trust the automation, while a 5 reflects high distrust. Do you have any questions?

Slide 32 (Any Questions?)

Do you have any overall questions on the reliability and trust questions?

Go ahead and answer these questions.

BREAK IF NEEDED

Slide 33 (Main Experiment)

Your next task is the main portion of this experiment. This portion involves playing six blocks consisting of two 5-minute games each (or a total of 12 games). Each block will contain the same autonomous agent. Just to reiterate, the job of this agent is to assist you in avoiding the next imminent collision. The agent aiding you in each block may have a reduced reliability, meaning it may not act upon every collision detected. After each block, you will complete the reliability and trust ratings, the ISA, and NASA-TLX. Again, I want to remind you that these ratings should only reflect the most recent block.

Slide 34 (Questions)

Do you have any questions?

Go ahead and perform the main experiment.

Once finished, you will complete the second IAT.

Slide 35 (TLX Comparisons)

Your final task is to complete the TLX comparisons. This portion involves comparing two factors of workload, and deciding which of the two are more relevant in the game. For example, if you are comparing Mental and Physical Demand, you can ask yourself:

Was the level of mental demand you exerted influencing your overall workload more, or were you more focused on the physical demands of the task?

Then choose the factor that influenced the overall workload more.

Slide 36 (Questions)

Do you have any questions?

Slide 37 (Post-Exp)

All that remains is a final post-experiment questionnaire. Once completed you are free to go.

If you would like to review any questionnaire data or want to know how we will be using your in-game data, feel free to ask after this questionnaire.

I request that you please keep experiment information to yourself in order to prevent disqualifying any future participants.

Thank you for your participation.

Bibliography

1. B. M. Albaker and N. A. Rahim. A survey of collision avoidance approaches for unmanned aerial vehicles. *International Conference for Technical Postgraduates 2009, TECHPOS 2009*, 2009.
2. Jason M. Bindewald, Michael E. Miller, and Gilbert L. Peterson. A function-to-task process model for adaptive automation system design. *International Journal of Human Computer Studies*, 72(12):822–834, 2014.
3. Jason M. Bindewald, Gilbert L. Peterson, and Michael E. Miller. Clustering-Based Online Player Modeling. In Tristan Cazenave, Stefan Edelkamp, and Mark Winands, editors, *International Joint Conference on Artificial Intelligence (IJ-CAI) - Computer Games Workshop*, New York, New York, USA, 2016.
4. J.G Boubin, C.F. Rusnock, J.M. Bindewald, and M.E. Miller. Quantifying Compliance and Reliance Trust Behaviors to Influence Trust in Human-Automation Teams. Unpublished, 2016.
5. Andreas Bye, Erik Hollnagel, and Tor Steinar Brendeford. Human-machine function allocation: a functional modelling approach. *Reliability Engineering & System Safety*, 1999.
6. Stephen Cameron. Collision Detection by Four-Dimensional Intersection Testing. *IEEE Transactions on Robotics and Automation*, 6(3):291–302, 1990.
7. Feng Cao and Saadat Malik. Vulnerability analysis and best practices for adopting IP telephony in critical infrastructure sectors. *IEEE Communications Magazine*, 44(4):138–145, 2006.
8. Alain Chavaillaz, David Wastell, and J?rgen Sauer. System reliability, performance and trust in adaptable automation. *Applied Ergonomics*, 52:333–342, 2016.
9. Jason a Colquitt, Brent a Scott, and Jeffery a LePine. Trust, trustworthiness, and trust propensity: a meta-analytic test of their unique relationships with risk taking and job performance. *The Journal of applied psychology*, 92(4):909–927, 2007.
10. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms, Third Edition*. The MIT Press, 3rd edition, 2009.
11. John Davis. Shuffle Bags: Making Random() Feel More Random, 2012. <https://gamedevelopment.tutsplus.com/tutorials/shuffle-bags-making-random-feel-more-random--gamedev-1249> Accessed: 2016-09-26.

12. E. J. de Visser, M. LeGoullon, a. Freedy, E. Freedy, G. Weltman, and R. Parasuraman. Designing an Adaptive Automation System for Human Supervision of Unmanned Vehicles: A Bridge from Theory to Practice. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 52(4):221–225, 2008.
13. J. C F de Winter and D. Dodou. Why the Fitts list has persisted throughout the history of function allocation. *Cognition, Technology and Work*, 2014.
14. Stephen R Dixon and Christopher D Wickens. Automation reliability in unmanned aerial vehicle control: a reliance-compliance model of automation dependence in high workload. *Human factors*, 48(3):474–486, 2006.
15. Pierre P Duez, Michael J Zuliani, and Greg a Jamieson. Trust by Design : Information Requirements for Appropriate Trust in Automation. *Proceedings of the 16th IBM Centre for Advanced Studies’ Annual International Conference on Computer Science and Software Engineering.*, 2006.
16. M.R. Endsley. Autonomous Horizons: System Autonomy in the Air Force - A Path to the Future. 1: Human-A:27, 2015.
17. Amos Freedy, Ewart DeVisser, Gershon Weltman, and Nicole Coeyman. Measurement of trust in human-robot collaboration. *Proceedings of the 2007 International Symposium on Collaborative Technologies and Systems, CTS*, 2007.
18. Elmer G. Gilbert, Daniel W. Johnson, and S. Sathiya Keerthi. A Fast Procedure for Computing the Distance Between Complex Objects in Three-Dimensional Space. *IEEE Journal on Robotics and Automation*, 4(2):193–203, 1988.
19. Tyler Goodman, Michael E. Miller, Christina F. Rusnock, and Jason M. Binde-wald. Timing Within Human-Agent Interaction and its Effects on Team Performance and Human Behavior. In *2016 IEEE International Multi-Disciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support (CogSIMA)*, pages 35–41. IEEE, 2016.
20. Anthony G. Greenwald, Brian A. Nosek, and Mahzarin R. Banaji. "Understanding and using the Implicit Association Test: I. An improved scoring algorithm": Correction to Greenwald et al. (2003). *Journal of Personality and Social Psychology*, 85(3):481–481, 2003.
21. G. Grote, S. Weik, T. Wäfler, and M. Zölch. Criteria for the complementary allocation of functions in automated work systems and their use in simultaneous engineering projects. *International Journal of Industrial Ergonomics*, 1995.
22. G. Hart, Sandra. NASA-task load index (NASA-TLX); 20 years later. *Human Factors and Ergonomics Society Annual Meeting*, pages 904–908, 2006.

23. Kevin Anthony Hoff and Masooda Bashir. Trust in Automation: Integrating Empirical Evidence on Factors That Influence Trust . *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 57(3):407–434, 2015.
24. Jiun-yin Jian, Ann M Bisantz, Colin G Drury, and James Llinas. Foundations for an Empirically Determined Scale of Trust in Automated Systems. 1998.
25. P. Jiménez, F. Thomas, and C. Torras. 3D collision detection: A survey. *Computers and Graphics (Pergamon)*, 25(2):269–285, 2001.
26. Matt Kamlet. NASA-Supported Collision Avoidance System Saves Unconscious F-16 Pilot In Fourth Confirmed Rescue, 2016. https://www.nasa.gov/centers/armstrong/features/auto-gcas_performs_fourth_confirmed_save.html Accessed: 2016-12-05.
27. J. D. Lee and K. A. See. Trust in Automation: Designing for Appropriate Reliance. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 2004.
28. John D.; Lee and Neville Moray. Trust, self-confidence, and operators’ adaptation to automation. *International Journal of Human-Computer Studies*, pages 230–253, 1994.
29. Xin Li, Traci J. Hess, and Joseph S. Valacich. Why do we trust new technology? A study of initial trust formation with organizational information systems. *The Journal of Strategic Information Systems*, 17:39–71, 2008.
30. Patrick Lin, George Bekey, and Keith Abney. Autonomous Military Robotics: Risk, Ethics, and Design. *California Polytechnic State University*, page 108, 2008.
31. Joachim Meyer, Rebecca Wiczorek, and Torsten Günzler. Measures of Reliance and Compliance in Aided Visual Scanning. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 56(5):840–849, 2014.
32. Peter G. Neumann. Risks of Automation: A Cautionary Total-System Perspective of Our Cyberfuture. *Communications of the ACM*, 59(10):26–30, 2016.
33. David J Niedober, Nhut T Ho, Gina Masequesmay, Kolina Koltai, Mark Skoog, Artemio Cacanindin, Walter Johnson, and Joseph B Lyons. Influence of Cultural, Organizational and Automation Factors on Human-Automation Trust: A Case Study of Auto-GCAS Engineers and Developmental History. *Human-Computer Interaction: Applications and Services, Pt Iii*, 8512:473–484, 2014.
34. R Parasuraman, T B Sheridan, and C D Wickens. A model for types and levels of human interaction with automation. *IEEE transactions on systems, man, and cybernetics. Part A, Systems and humans : a publication of the IEEE Systems, Man, and Cybernetics Society*, 2000.

35. Raja Parasuraman and Victor Riley. Humans and Automation: Use, Misuse, Disuse, Abuse. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 39(2):230–253, 1997.
36. J. M. Ross, J. L. Szalma, P. a. Hancock, J. S. Barnett, and G. Taylor. The Effect of Automation Reliability on User Automation Trust and Reliance in a Search-and-Rescue Scenario. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 52(19):1340–1344, 2008.
37. Stuart Russell. Take a stand on AI weapons. *Nature*, 521(7553):415–416, 2015.
38. Stuart Russell and Pete Norvig. *Artificial Intelligence: A Modern Approach*. 3 edition, 2009.
39. Mark W Scerbo. Theoretical Perspectives on Adaptive Automation. pages 37–63, 1996.
40. Kristin E. Schaefer. The perception and measurment of human-robot trust. 2013.
41. David Šišlák, Michal Pěchouček, Pemysl Volf, Dušan Pavlíček, Jií Samek, Vladimír Mařík, and Paul Losiewicz. AGENTFLY: Towards multi-agent technology in free flight air traffic control. *Defence Industry Applications of Autonomous Agents and Multi-Agent Systems*, pages 73–96, 2008.
42. Leon S. Sterling and Kuldar Taveter. The Art of Agent-Oriented Modeling. *The MIT Press*, 2009.
43. Andrew J. Tattersall and Penelope S. Foord. An experimental evaluation of instantaneous self-assessment as a measure of workload. *Ergonomics*, 39(5):740–748, 1996.
44. C Tomlin, G J Pappas, and S Sastry. Conflict resolution for air traffic management: a study in multiagent hybrid systems. *IEEE Transactions on Automatic Control*, 43:509–521, 1998.
45. Douglas A Wiegmann. Agreeing with automated diagnostic aids: A study of users’ concurrence strategies. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 44(1):44–50, 2002.
46. Bob Work. Deputy Secretary of Defense Speech, 2016. <http://www.defense.gov/News/Speeches/Speech-View/Article/753482/remarks-by-deputy-secretary-work-on-third-offset-strategy> Accessed: 2016-12-08.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. **PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.**

1. REPORT DATE (DD-MM-YYYY) 23-03-2017			2. REPORT TYPE Master's Thesis		3. DATES COVERED (From — To) Jun 2015 — Mar 2017	
4. TITLE AND SUBTITLE A Study of Human Reliance on Imperfect Automation					5a. CONTRACT NUMBER	
					5b. GRANT NUMBER	
					5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S) Garnick, Christopher J, 2d Lt, USAF					5d. PROJECT NUMBER	
					5e. TASK NUMBER	
					5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Air Force Institute of Technology Graduate School of Engineering and Management (AFIT/EN) 2950 Hobson Way WPAFB OH 45433-7765					8. PERFORMING ORGANIZATION REPORT NUMBER AFIT-ENG-MS-17-M-030	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) Air Force Office of Scientific Research 875 N. Randolph, Ste.325 Arlington VA 22203					10. SPONSOR/MONITOR'S ACRONYM(S) AFOSR	
					11. SPONSOR/MONITOR'S REPORT NUMBER(S) N/A	
12. DISTRIBUTION / AVAILABILITY STATEMENT DISTRIBUTION STATEMENT A: APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.						
13. SUPPLEMENTARY NOTES This work is declared a work of the U.S. Government and is not subject to copyright protection in the United States.						
14. ABSTRACT Appropriate trust in human-machine teams is critical in ensuring proper utilization of automation. Without appropriate trust, human operators may disuse or misuse automation resulting in loss of capability, efficiency, and safety. Designing automation that ensures appropriate trust proves to be a difficult problem, as it requires a means to measure and calibrate trust. The goal of this research intends to investigate trust calibration by examining the relationship between the human reliance in automation and automation with varying reliability.						
15. SUBJECT TERMS Human-Machine Teams, Trust in Automation, Human Behavior of Reliance, Trust Calibration						
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON	
a. REPORT	b. ABSTRACT	c. THIS PAGE			Maj Jason M. Bindewald, AFIT/ENG	
U	U	U	UU	138	19b. TELEPHONE NUMBER (include area code) (937) 255-6565; jason.bindewald@afit.edu	