3-23-2017

Analysis of Human and Agent Characteristics on Human-Agent Team Performance and Trust

Anthony J. Hillesheim

ANALYSIS OF HUMAN AND AGENT CHARACTERISTICS ON HUMAN-AGENT TEAM PERFORMANCE AND TRUST

Anthony J. Hillesheim, 2nd Lieutenant, USAF

AFIT-ENV-MS-17-M-194

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the United States Air Force, 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-ENV-MS-17-M-194

ANALYSIS OF HUMAN AND AGENT CHARACTERISTICS ON HUMAN-AGENT TEAM PERFORMANCE AND TRUST

THESIS

Presented to the Faculty

Department of Systems Engineering and Management

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 Master of Science in Systems Engineering

Anthony J. Hillesheim, BS

2nd Lieutenant, USAF

March 2017

AFIT-ENV-MS-17-M-194

ANALYSIS OF HUMAN AND AGENT CHARACTERISTICS ON HUMAN-AGENT TEAM PERFORMANCE AND TRUST

Anthony J. Hillesheim, BS

2nd Lieutenant, USAF

Committee Membership:

Maj Christina F. Rusnock, PhD
Chair

Dr. Michael E. Miller
Member

Maj Jason M. Bindewald, PhD
Member

AFIT-ENV-MS-17-M-194

Abstract

Recent Department of Defense strategy documents have outlined the need for further research into how human users interact with automated agents and the impact of human interaction with automated agents on overall human-agent team performance. The human-agent team represents a new construct in how the United States Department of Defense is orchestrating mission planning and mission accomplishment. In order for mission planning and accomplishment to be successful, several requirements must be met: a firm understanding of human trust in automated agents, how human and automated agent characteristics influence human-agent team performance, and how humans behave in human-agent teaming environments. This thesis applies a combination of modeling techniques and human experimentation to understand the aforementioned concepts. The modeling techniques used included static modeling in SysML activity diagrams and dynamic modeling of both human and agent behavior in IMPRINT. Additionally, this research included human experimentation in a dynamic, event-driven, teaming environment known as *Space Navigator*. Both the modeling and the human-in-the-loop experiment depict that the agent's reliability has a significant effect upon human-agent team performance. Additionally, this research found that the age, gender, and education level of the human user has a relationship with the perceived trust the user has in the agent. Finally, it was found that patterns of compliant human behavior,which are known as archetypes, can be created to classify human users.

Acknowledgments

I would like to extend a special thank you to my advisor, Maj Christina Rusnock, for all of her time, patience, and dedication in helping me throughout the course of this thesis effort. I would also like to thank my thesis committee, Dr. Michael Miller and Maj Jason Bindewald, for the help they have provided me throughout this process. This thesis effort would not have been possible without the support and encouragement of my classmates, family, and friends. Finally, I would like to express my sincere appreciation to my wife for her endless love and support.

Anthony J. Hillesheim

Table of Contents

List of Figures

Page

List of Tables

TRUST IN AUTOMATION

I. Introduction

Background

Automation is becoming an ever-increasing aspect of modern warfare. Automation applications can be seen across a wide spectrum of Department of Defense applications. Automation can be found in unmanned aerial vehicles (UAVS) in the United States Air Force, unmanned ground vehicles (UGVS) in the United States Army and Marine Corps, and unmanned maritime vehicles (UMVS) in the United States Navy (Endsley 2015). Automation is defined as the assignment of the execution of functions previously carried out by a human to a computer (Parasuraman & Riley 1997). One of the factors that affects human performance when interacting with automation is the automation's reliability (Wickens & Dixon, 2007).

The majority of research addressing the performance and user trust in the system has examined the use of automation reliability when the automation acts as an aid to the user (de Visser & Parasuraman, 2011; Dixon et al., 2006; Sheridan, 1984). However, there is a lack of depth in the literature addressing the effects of reduced reliability of an autonomous agent which acts as a teammate rather than an aid. An agent is an automation situated in some environment, and that is capable of autonomous action in this environment in order to meet its design objectives (Russell & Norvig, 2010), whereas autonomy is defined as a capability (or set of capabilities) that enables a particular action of a system to be automatic or, within programmed boundaries, 'self-governing' " (Kaminski, 2012).   Human-agent teaming is increasing across all branches of the Department of Defense (DoD), because increasing autonomy enables the DoD to achieve increased efficiency and effectiveness with fewer human resources.

Human-agent teaming in modern warfare does not simply replace the human, but rather augments the human's capabilities (Kaminski, 2012). However, the risks of increasing agent autonomy include an increase in complexity, greater potential for failures, and increased areas of concern for security vulnerabilities. Hoff and Bashir (2015) explain the purpose of automation and state the purpose of automation is to perform complex, repetitive tasks in an effective manner so as to let the human operator focus his or her attention where it is needed.

The United States Air Force has outlined four key advantages of increased levels of system autonomy. The first advantage is a reduction in unnecessary manual labor and lower system manning costs. The second advantage is the increase in range of operations and extension of manned capabilities. The third advantage is the reduction in time required to conduct time-critical operation. Finally, increased levels of system autonomy promise to provide operational reliability, persistence, and resilience (Endsley, 2015).

While there are numerous advantages to autonomy, increasing the level of human-machine teaming and automation is not without significant challenges. The DoD Autonomy Task Force and the Chief Scientist of the Air Force have described and outlined many of the challenges of implementing increased levels of system autonomy. Both the Task Force and the Chief Scientist suggest that trust in automation is a key underpinning to successful adoption of increased levels of system autonomy (Endsley, 2015; Kaminski, 2012). Specifically, the DoD Autonomy Task Force has called upon the Under Secretary of Defense for Acquisition, Technology, and Logistics to create developmental and operational test and evaluation techniques that focus explicitly on building trust in autonomous systems (Kaminski, 2012). Additionally, the Chief Scientist stated that building trust in autonomous systems is a hurdle that today's airman faces. She went on to state that the ability of people to effectively use imperfect automation has been strained by difficulties in determining appropriate levels of trust (Endsley, 2015). The levels in automation trust are determined by the relationship between the user's trust and the system's reliability, and can be classified into four categories: correct trust, correct distrust,over-trust, and under-trust seen in Figure 1. System Reliability and Trust

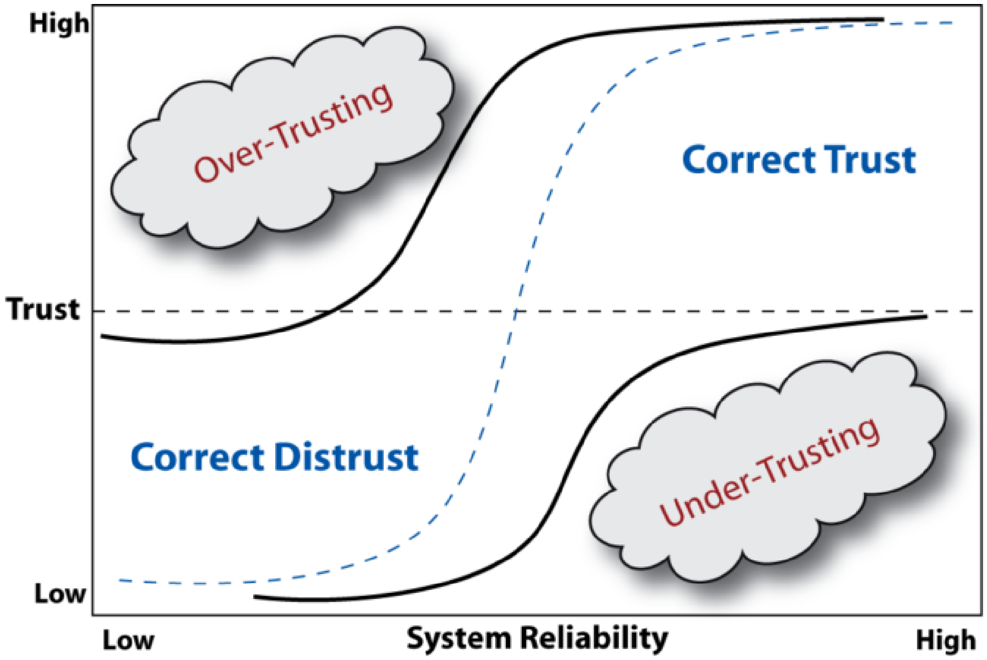


Figure 1. System Reliability and Trust (adapted from Endsley, 2015)

Correct trust is when the operator places the appropriate amount of trust in the automated system. Correct distrust is when the operator places the appropriate amount of distrust in a system that demonstrates low reliability. Over-trusting an automated system occurs when the user's trust level is high even though the system's reliability is low. Over-trust can result in the human operator misusing the automated system (Lee et al., 2004). Under-trust occurs when the user's trust is low even though the system's reliability is high, which can instigate disuse of the automated systems (Merritt, Heimbaugh, LaChapell, & Lee, 2013). There is an operational need for exploring what characteristics of both the operator and the autonomous system will affect the level of trust an operator has in an autonomous system (Endsley, 2015; Kaminski, 2012). For further information about trust, reliability, and automation reference Appendix 1.

Relevant Literature

The current literature reflects a large amount of work focused on user trust in automation and how it relates to reliance, compliance, reliability, misuse, and disuse of automation. Trust, as it pertains to automation, has been defined as, “One’s confidence in an automated system to help them achieve their goals in a situation characterized by uncertainty and vulnerability” (Lee and See, 2004). However, trust has also been defined as socially-learned expectations about a system, whereas others define it as the beliefs held by an individual about a system (Hoff & Bashir, 2015). At the most basic levels, trust is defined as a disposition toward the world (Hoff & Bashir, 2015).

If a human operator trusts a system’s automation that has a lower reliability than manual operation, or if a human operator distrusts a system’s automation that has a higher reliability than manual operation, poorly calibrated trust will likely result (Dzindolet, Peterson, Pomranky, Pierce, & Beck, 2003). Calibrating an operator’s trust in an automated system is necessary to prevent over-trust in the automation or distrust in the automation.

Misuse is a rather broad term when applied to automation and encompasses several key components of trust in automation. Misuse is classically defined as “overreliance on automation” (Dzindolet et al., 2003; Parasuraman & Miller, 2004). It has also been described as a mental state that consists of a low level of questioning (Dzindolet et al., 2003). Misuse stems from several variables. One of the most influential variables that drive misuse is operator knowledge about the system (Hoff & Bashir, 2015). If the human operator has an understanding of the purpose of the system or how it functions, it is likely that the human operator will have greater success in accurately aligning his or her trust to the system automation. Conversely, if a human operator has little to no knowledge about the purpose of the system or how the system functions, it is unlikely that he or she will be able to accurately align his or her trust to the automated system’s reliability. This is especially true when the formation of trust between the human operator and the automated system depends on the automated system’s performance that varies in separate contexts and differing temporal phases (Hoff & Bashir, 2015).

Under-trust can instigate disuse of the automated systems. Disuse is defined as the situation in which users fail to rely on automation when doing so would improve performance (Merritt et al., 2013). Several studies have demonstrated that disuse originates from a lack of appropriate instruction. If an automated system malfunctions in a way that the human operator is unable to explain or understand, disuse will likely occur (Dzindolet, Peterson, Pomranky, Pierce, & Beck, 2003). In the experiments conducted by Dzindolet et. al., they found that disuse of the automated system was more prevalent than misuse.

Problem Statement

Military environments are inherently high-stakes operational contexts in which military members are putting their lives and the lives of others on the line. The military has seen a large influx in the application of automation throughout all facets of operations. For both the members and the automation to be successful, there must be a relationship of trust between the member and the automation. This relationship is not fully understood, which in turn has cost the lives of many service members (Endsley 2015). The current body of research does not have the answers to how user trust, user demographic data, and human-agent team performance are related.

Research Question

The Chief Scientist of the Air Force and the task report compiled by the DoD call for further research into how agent reliability affects human-agent team overall performance (Endsley, 2015; Kaminski, 2012). Therefore, the research question this thesis poses is: how do human and autonomous agent characteristics affect human-agent team performance and user trust in an automated agent?

Investigative Questions

The thesis answers the research question by answering the following investigative questions:

1. How does automation reliability affect human-agent team performance?
2. Do archetypes of human trust behaviors affect human-agent team performance?

3. What human demographic factors predict perceived trust and trust behaviors?

The first question focuses on the agent characteristic of reliability and how reliability impacts overall team performance. It is expected that low levels of agent reliability will yield lower levels of team performance. It is hypothesized that at high levels of agent reliability, the human-agent team will suffer from only slightly degraded performance. However, it is also hypothesized that at decreasing levels of agent reliability, the human agent team will suffer from drastically reduced performance. The second question focuses on human archetypes or behavioral patterns that affect human-agent team performance. It is hypothesized that different archetypes of human user behavior will exhibit different levels of human-agent team performance. Finally, the third question looks to identify what human demographic factors predict trust. It is hypothesized that factors such as age, gender, education, and experience with technology will influence perceived trust and patterns of trust behavior.

Methodology

The environment in which this research is conducted is the tablet computer game, Space Navigator (Bindewald 2014). The game is a simple, open-source air-traffic management game similar to Harbormaster and Flight Control (Bindewald 2014). The game consists of activities that are completed by the human, the automated agent, or both. The goal of the game is to ensure that spaceships safely arrive at their correct destination while avoiding collisions with other ships or traversing no-fly zones. The automated agent is capable of performing the route-generation task, but does not perform collision or no-fly zone avoidance. Thus, to successfully complete the mission, the human and agent must work as a team.

The research consists of three stages. The first two stages address the first investigative question and the final stage addresses the last two investigative questions. The first stage consists of modeling the gameplay of a user and automation working collaboratively to answer the first research question posed earlier in this chapter. The gameplay is modeled using activity diagrams, and then the task networks are transferred to the Improved

Performance Research Integrated Tool modeling environment, which captures the flow of actions and decision logic. The models are then updated to examine reduced reliability rates. The model outputs are analyzed to examine the effects of reduced automation reliability rates on human-agent team performance. The next stage consists of a human-in-the-loop experiment, where the subjects worked as a human-agent team when the automation experienced varying rates of reduced reliability, which answers the second research question posed earlier in this chapter. The final stage of research consists of analyzing human compliance behavior through the utilization of archetypes and cluster analysis.

Limitations

The largest obstacle to this research is that the human subjects used in the human-in-the-loop experiment do not perfectly represent the habits and actions of military operators in a combat environment. Military environments are high-stakes environments where an inappropriate decision result in the inadvertent loss of life. Space Navigator cannot replicate the high-stakes environment. Space Navigator will not be able to completely simulate the relationship of an operator with automation when lives are at stake. The research is also limited in sample size and population. The sample size for the human-in-the-loop experiment is 48 participants which can lead to low R-values and an inability to fully explain the variance seen in the models.Correlations and regressions are subject to random error, and larger sample reduces that error and makes the estimate move closer to the true value. Finally, the human-in-the-loop experiment is inherently limited as the participant pool was comprised of college age students at a civilian institute, which does not represent military operators in a combat environment. The military population does not have the same demographic features as a population comprised of college age students at a civilian institute. This difference in demographics may affect the patterns of behavior (and thus, the archetypes identified in this research) of the participants; thus, further research is needed.

Preview

The Air Force has called for further research into what drives trust in automation and human-agent teaming. This thesis explores how agent reliability and human characteristics such as bias, implicit trust, and demographics affect human-agent team performance. This thesis consists of three articles that address different aspects of trust, automation, and reliability. The articles are independent of each other, but are all related. Chapter 2 contains the first article, which is “Predicting the Effects of Automation Reliability Rates on Human-Automation Team Performance” (Hillesheim & Rusnock, 2016), which examines how reduced reliability rates in a simulated environment affect human-agent team performance. Chapter 3 includes the article “The Effect of Automated Agent Reliability on Human-Agent Team Performance”, which uses a human-in-the-loop experiment to analyze how automated agent reliability effects human-agent team performance, this article looks to validate the findings from Chapter 2. Chapter 4 includes the article “Relationships between Human User Demographics and User Trust in an Autonomous Agent”, which helps understand the effect of user demographics such as age, gender, and education as well as agent characteristics such as reliability on user perceived trust in an automated agent. Chapter 5 includes the article “Analysis of Archetypes on Human-Agent Performance,” which creates user archetypes based on human behavior to cluster users. These clusters are then compared to user demographic data to explore the relationship between trust behavior and demographic data. Chapter 6 includes the conclusion, research objective, answers the investigative questions, and provides a synopsis of the work completed.

II. Predicting the Effects of Automation Reliability Rates on Human-Automation Team Performance

Abstract[1]

This study investigates the effects of reduced automation reliability rates on human-automation team performance. Specifically, System Modeling Language (SysML) activity diagrams and Improved Performance Research Integrated Tool (IMPRINT) models are developed for a tablet-based game which includes an automated teammate. The baseline model uses previously collected data from human-in-the-loop experiments where the automated teammate performs with 100% reliability. It is expected that team performance and user trust in automation will be affected if the automation is less reliable. The baseline model is modified to create alternate models that incorporate degraded automation reliability rates from 50% to 90%. This study finds that when automation reliability was 100%, the automation was an effective teammate and enabled the human-automation team to achieve statistically improved performance over human-only scenarios. However, at reliability rates of 90% and less, the presence of the automated agent degraded system performance to levels less than achieved in human-only scenarios.

Introduction

Automation is prevalent in nearly every facet of today's military, transportation, industrial, and medical fields. These fields deal with many complex, repetitive tasks, which are well-suited to automation, thus enabling the human operator to focus his or her attention where it is needed (Hoff & Bashir, 2015). The primary focus of many users' attention is on making decisions; automation consists of the technology that can select data, transform information, and make decisions or control processes (Lee and See2004). In order for the human-automation team to be effective or perform well, the cumulative effect of the team has to be greater than that of the human or

[1] This chapter was originally published and presented at the Winter Simulation Conference 2016

automation individually. One of the factors that affects human-automation teaming performance is automation reliability (Wickens and Dixon 2007). The majority of research supports the notion that human-automation teaming performance is generally quite good when the automation is perfect (Wickens and Dixon 2007; Dixon and Wickens 2006; Rovira, McGarry, and Parasuraman 2007). For example, Dixon and Wickens used a perfectly reliable auditory alert system to help pilots detect system failures during military reconnaissance missions, and they discovered that the automation improved performance (Dixon and Wickens 2006).

Unfortunately, most users of automation have experienced unreliable automation at some point. When conducting a complex and/or difficult task, performance will be limited and significantly reduced when automation is degraded; unreliable automation could actually harm performance relative to the human working without automation (Sheridan, 1984). Highly reliable but imperfect automation has shown to be the cause of states of over-trust (Parasuraman and Miller 2004), over compliance (Parasuraman and Manzey 2010) and reliance issues (Dixon, Wickens, and Mccarley 2006). While studies have examined the use of automation reliability as an aid to the user (Dixon and Wickens 2006; Wickens and Dixon 2007; Parasuraman and Manzey 2010) there is a lack of depth in the literature that addresses when an automated agent, who acts like a teammate rather than an aid, suffers from reduced reliability. Previous work found that systems using diagnostic automation experience positive performance when the automation had a reliability greater than 80%, neutral performance from 70% to 80% and negative performance when the automation's reliability was less than 70% (Dixon and Wickens 2006; Wickens and Dixon 2007; Maltz and Shinar 2003; Parasuraman and Manzey 2010). This paper extends this line of research by exploring the performance effects on human-automation teams when the automated agent is not perfectly reliable.

Two common forms of automation reliability errors are false alarms and misses (Dixon, Wickens, and Mccarley 2006). False alarms and misses directly affect both user reliance and compliance. Reliance pertains to the human operator's state when an alert or alarm is silent, meaning everything is "ok" (Dixon and Wickens 2006). Over-reliance on automation can create "automation-induced complacency," in which the automation is operating at a high level of reliability (but not perfectly), enabling the human operator to be lulled into a false sense of security, thus resulting in the human not detecting occasional automation failures (Singh, Tiwari, and Singh 2009). Inversely, compliance addresses the operator's response when the alarm sounds--whether true or false (Dixon and Wickens 2006). Wickens and Dixon examined false alarms and misses in terms of automation reliability as they pertain to unmanned aerial vehicles (UAVs). As previously discussed, perfect automation reliability had a beneficial effect on human-automation system performance (Dixon and Wickens 2006). When using an automation reliability rate of 67%, human-automation system performance was severely reduced and in some instances, the performance was worse than a human performing the tasks without the automation (Dixon and Wickens 2006).

In an effort to further understand the impact on performance, the effects on user compliance and reliance were also examined. It was found that automation false alarms decreased system failure detection rates and increased system failure detection times compared to when the human performed the tasks without the automation (Dixon and Wickens 2006). Additionally, when exploring degraded automation reliability and reliance; it was found that an increased miss rate negatively affected user reliance and users became less trusting of the automation (Dixon and Wickens 2006). In a similar study, Roivra, McGarry, and Parasuraman conducted an experiment examining several levels of automation reliability and how performance was affected. They found that lower levels of automation reliability led to greater cost in decision-making accuracy and decreased performance (Rovira, McGarry, and Parasuraman 2007). Additionally, they found that as automation reliability increased, complacency increased (Rovira, McGarry, and Parasuraman 2007).

The focus of this paper is on how reduced reliabilities affect human-automation team performance, where the automation is not just a decision aid but an "equal" teammate. This research hypothesizes that because team members fill a unique role on the team, interdependence, reliance, and expectations are higher for team members than for decision aids. This relationship amongst team members will demand high reliability from each teammate, with reduced reliability affecting both the automation's actions and the team interactions. To ensure a teaming scenario, this paper focuses on pairing an autonomous agent with the human to create a synergistic effect in which the human-automation team out performs either the automation or the human alone.

The work conducted in this paper leverages the advantages provided by the Improved Performance Research Integrated Tool (IMPRINT) and simulation in general. Simulation provided a means with which to examine how reduced automation reliability rates have the potential to affect human-automation team performance. Additionally, the simulations were conducted at no cost and in a low-risk environment. Simulation was also able to provide practical and timely results for analysis. This allowed for an increase in efficiency when exploring multiple alternative models.

Purpose

The purpose of the research is to explore how human-automation team performance is affected by varying levels of automation reliability. Varying levels of automation reliability have been studied in previous experiments and research; however, little research has been conducted using an automated agent that works together with the human as a teammate, rather than working as an aid. By working as a teammate, the automation is able to complete tasks and make decisions without human supervision. It is hypothesized that when the automated teammate has reduced levels of reliability; overall team performance will suffer. However, it is expected that just as imperfect human teammates can still be an asset to a team, imperfect (but highly reliable) automated teammates will also make a positive contribution to the human-automation team.

Application Environment

To explore the effect of automation reliability rates on team performance, it was necessary to select an application environment in which the human and the automation interact as a team, rather than the automation operating independently, as an aid, or under supervisory control. Thus, an environment in which tasks are highly integrated and there is a high level of human-automation interaction was necessary. The system selected for this research was the tablet computer game Space Navigator, a custom route-creation game similar to Harbormaster and Flight Control. The game consists of activities that are completed by the human, the automated agent, or both. The game contains four stationary planets present on the screen. Each planet is one of four colors: red, green, blue, or yellow. Spaceships are randomly generated on the sides of the screen at an interval of one spaceship every two seconds. Spaceships continue to appear until an allotted time of five minutes is over. Each spaceship is red, green, blue, or yellow. The player must direct each spaceship to the destination planet of corresponding color (e.g. red spaceship to red planet) by drawing a trajectory line on the game touch screen using his/her finger. The spaceship then follows this drawn trajectory route at a constant rate. If desired, trajectories may be re-drawn; this is often done to avoid a collision or account for dynamic changes in the environment. Points are earned when a ship successfully reaches its destination planet or traverses any of a number of small bonuses that appear throughout the play area. Upon reaching its destination planet, a spaceship disappears from the screen. When spaceships collide, points are lost and each spaceship involved in the collision is lost. Additionally, small bonuses appear in random locations throughout execution. If the path of a given spaceship crosses over one of these bonuses, it is `picked up' and a point bonus is given. The player loses points for allowing spaceships to traverse `no-fly zones’ that move to different random locations on the screen at a set time interval. In the human-in-the-loop experiment, the game features 100% reliable straight-line automation, which draws straight lines from the spaceships to the planets with a trigger rate of 2 seconds (meaning that the automation draws the route if the spaceship has been on-screen for 2 seconds without the human creating a route).

The environment allows the automation’s and the human’s actions to affect each other. The environment also creates the opportunity for human-agent team performance to be better than that of the human or automation alone. It is important to note that the automation does not feature collision avoidance, thus enabling the automation to serve as a team member with limited capability and not a perfect solution capable of performance superior to human performance. Figure 2. Space Navigator Environment depicts the game environment with an annotated screen capture from Space Navigator, which illustrates the elements of the game described herein.

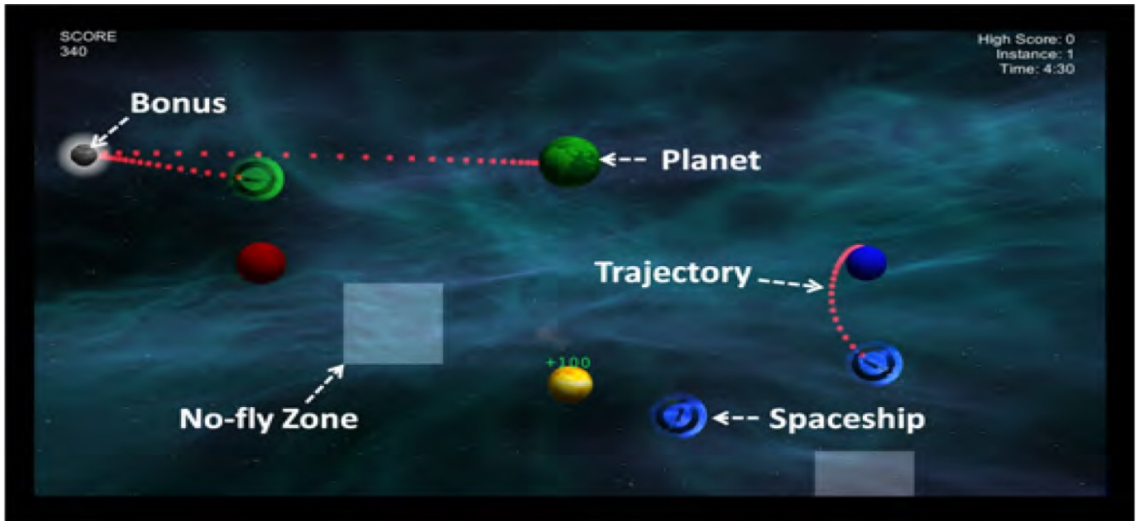


Figure 2. Space Navigator Environment

Method

The first step in the procedure was conducting a human-in-the-loop experiment. This experiment was previously conducted and the results were used for this paper (Bindewald, Miller, & Peterson, 2014; Bindewald, Peterson, & Miller, 2015, 2016). The second step consisted of modeling the application environment as a Systems Modeling Language (SysML) activity diagram. The activity diagram allowed for the conceptual model to be transferred to a diagram that consists of activities and functions with decision logic and flow. The activities and functions with decision logic and flow on the activity diagram were then transferred to an IMPRINT simulation model. IMPRINT is a discrete-event simulation software tool that allows the modeling of human performance and analysis of human performance with a graphical user interface. The software allows for the use of task-network models that provide a visual representation of tasks performed by human operators (Mitchell, 2009). This established a baseline model which captured all of the tasks that both the user and the automation completed during game play. The baseline model was validated using data previously collected from 36 particpants that played in the application domain with 100%

reliable automation. Once the baseline model was validated, an alternate model was created. The alternate model addressed the reduction in automation reliability.An initial investigation was conducted in which IMPRINT simulated operator performance under automation reliability rates of 50-100% in increments of 10%. Following the initial investigation, the analysis was refined to investigate performance under the additional automation reliability rates between 90-100% in 1% increments.The results of the simulation were then compared against each other and the baseline model. Figure 3 depicts the methodology.

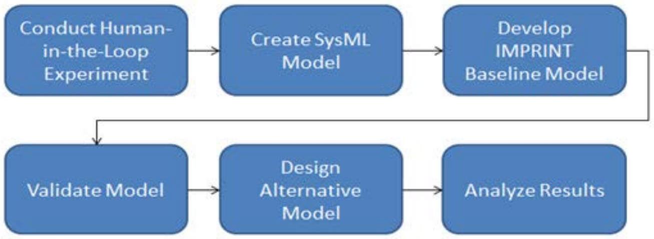


Figure 3. Methodology Approach

Human-in-the-Loop Experiment

The experiment involved 36 volunteers with an average age of 32.5 years and a range of 22 to 39 years. A total of 30 males and 6 females participated. The experimental procedure consisted of a within subjects design in which each participant completed 17 five-minute games of Space Navigator. The first five games contained no interaction from an automated agent and were used as participant training sessions. Following the training, participants completed three experimental sessions. Experimental sessions included 4 five-minute games: one manual and then three with three differing automated agents, each with its own route-generation strategy. While three different automated agents were utilized in the experiment, only the agent with the straight-line strategy was of interest for this reliability simulation study. The straight-line strategy draws straight-line routes (i.e. shortest, most direct path) from the ship to the corresponding planet and was 100% reliable (i.e., drew 100% of the lines to the planet with a color corresponding to the color of the ship). Since only this agent is analyzed in this simulation study, the participant data from the three games (3 games x 36 participants = 108 games) that contained straight-line automation were used to populate and validate the model (Bindewald, Peterson, & Miller, 2016).

SysML Model Development

After the human-in-the-loop experiment was conducted, an Activity Diagram was created using SysML. SysML is a general-purpose graphical modeling language that is useful for analysis, specification, design, verification and validation of complex systems (Steiner, Moore and Friedenthal 2014). The usability of the language extends to modeling human, automated, human-automation, and data centric systems (Steiner et al 2014). SysML facilitates the application of model-based systems engineering that provides several benefits: flow-based behavior, constraints on physical and performance properties, as well as structural classification of systems (Steiner et al 2014).

Figure 4. shows the activity diagram which consists of multiple elements that provide clarity in understanding the activities and the flow of tasks. The elements include action nodes, control nodes, pins, and flows. The action nodes are the “transformers” of the process. The action nodes take inputs and transform them into outputs. The inputs and outputs are denoted by activity pins. The flows connect the output of one action and connect it to the input of another action.

The activity diagram includes the actions of both the automation and the human operator. The diagram includes all of the appropriate actions and decision necessary to accurately depict the functions of the system. This diagram provides the basis for task networks within IMPRINT. For a complete description of all techniques and assumptions made for the SysML modeling, reference Appendix 2.

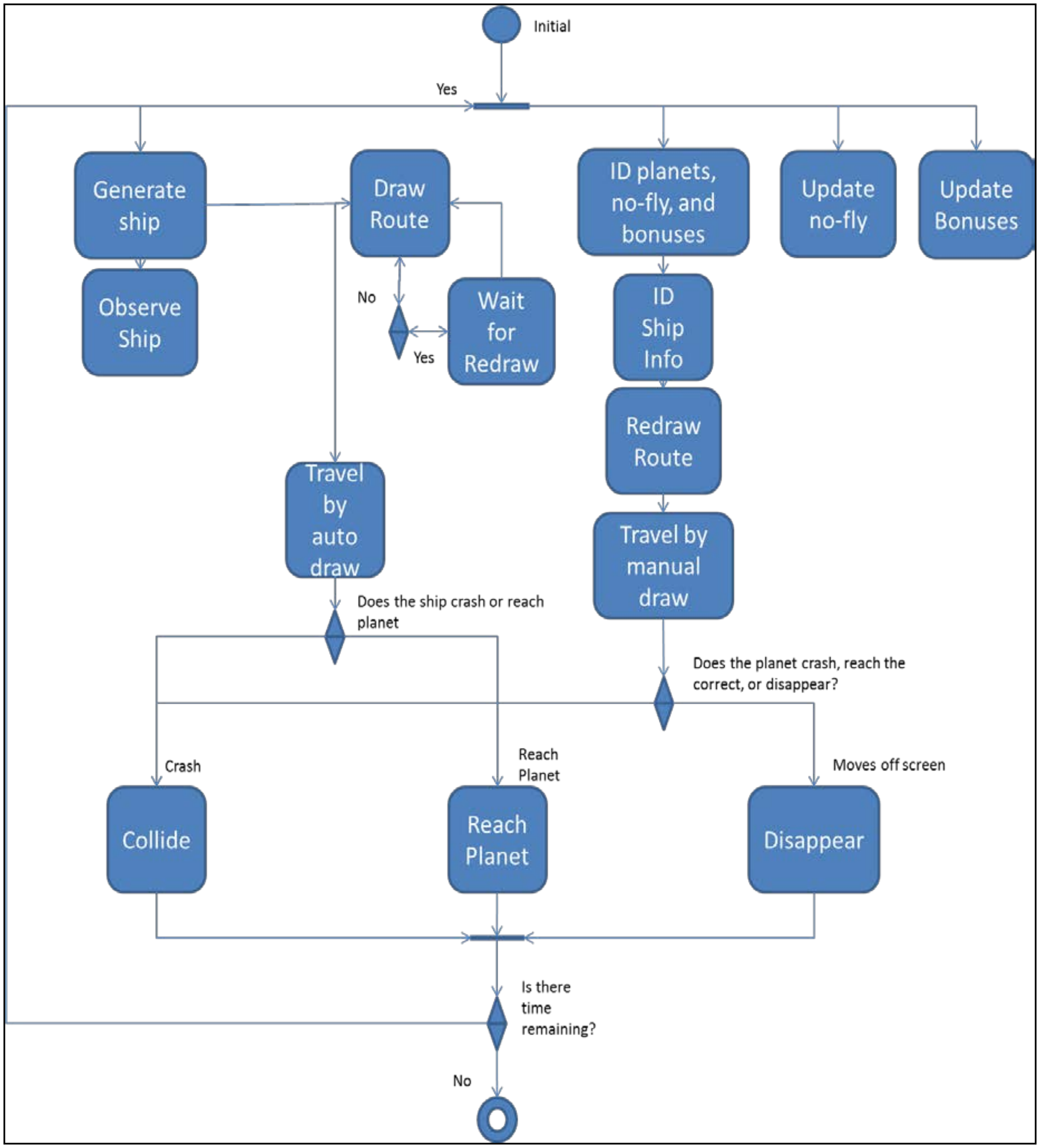


Figure 4. SysML Activity Diagram for Model

IMPRINT Baseline Model Development

As previously mentioned, IMPRINT provides an environment with which simulations can be performed to study human-system performance. The task networks developed in the activity diagram were transferred to this modeling environment, capturing the flow of actions and decision logic. The completion of the IMPRINT model required determining task time probability distributions as well as probability functions relating to the successful completion or failure of certain tasks. Using the data from the human-in-the-loop experiment, the baseline model was created.

As seen in Figure 5. the baseline model task network is composed of three different task types. The purple tasks denote tasks performed by the automated agent, the blue tasks denote human player tasks, and the green tasks denote game environment tasks. Starting at the top of Figure 4, the first task flow is generating ships. The task depicts the system creating ships throughout the entire game. The "Draw Route" task does not begin until there are ships on screen without routes and will continue to loop on itself while there are ships on screen without routes. The task also accounts for the human user redrawing and will wait to redraw routes. The "Wait for Redraw" task stems from the need for the automation to wait for the human player while he or she is given an opportunity to draw a route. The "Travel if by auto draw" task models a ship traveling on a route drawn by the automation. This task accounts for the ship picking up bonuses and/or flying through "no-fly zones". The model also accounts for the background activities that occur throughout the game. The "Update no-fly zones" task models the system updating the zones every thirty seconds. The "Update Bonus Locations" task ensures that every thirty seconds the game is populated with three new bonus orbs. The "Operate clock" task is used to control the length of each game. The games are five minute in length.

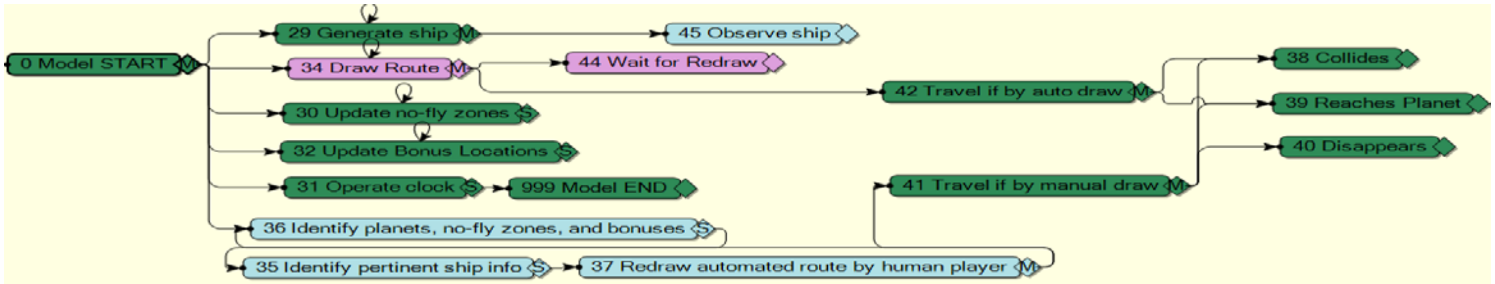

0 Model START
29 Generate ship
45 Observe ship
34 Draw Route
44 Wait for Redraw
42 Travel if by auto draw
38 Collides
39 Reaches Planet
40 Disappears
30 Update no-fly zones
32 Update Bonus Locations
31 Operate clock
999 Model END
41 Travel if by manual draw
36 Identify planets, no-fly zones, and bonuses
35 Identify pertinent ship info
37 Redraw automated route by human player

Figure 5. Baseline Model in IMPRINT

The bottom of Figure 5. depicts the tasks the human user accomplishes to identify the game background and redraw routes if he or she does not agree with the automated routes drawn. The “Identify planets, no-fly zones, and bonuses” task accounts for the user examining all of the background game information. Once the human user has accomplished this task, he or she will move on to identifying all pertinent ship information such as the automated route drawn, if the ship is projected to collect bonuses, pass through a no fly zone, or potentially collide with another ship. The task also ensures there are ships on screen before it passes to redrawing routes. Before the “Redraw automated route by human player” task begins, the release condition ensures that there are ships with automatically drawn routes and that there are ships on screen.

After traveling, a ship has two possible outcomes if it is traveling on an automated route: collide or reach the correct planet. The automation will not draw a ship off-screen. However, if the ship is traveling on a human-generated route, the ship can collide, reach the correct planet, or disappear (off-screen). If the ship goes to the “Collides” task, the task will account for the loss of 100 points. The “Reaches Planet” task increases score by 100 points. The final task encompasses when a ship disappears. The task “Disappears” accounts for when a ship with a manually draw route goes off-screen. It should be noted that this is an action taken by human players to reduce the number of ships on-screen to reduce the opportunity for collision, particularly in areas where ship density is high.

Model Validation

Before performing validation with the model predicted score data, we first confirm that the data meets the assumptions of normality. Figure 6. presents the normal probability plot to demonstrate that deviations are minimal and the data are normally distributed. The associated Shapiro-Wilk goodness-of-fit test yielded a p-value of 0.9914, thus the null hypothesis is not rejected and there is not significant evidence to state that the data are not normal.

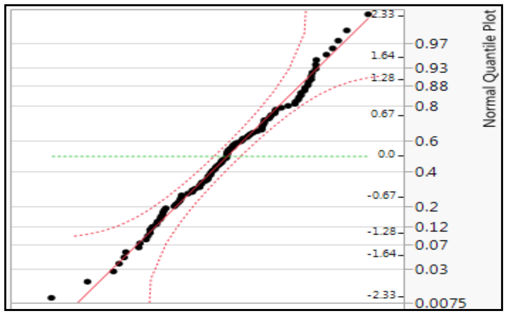
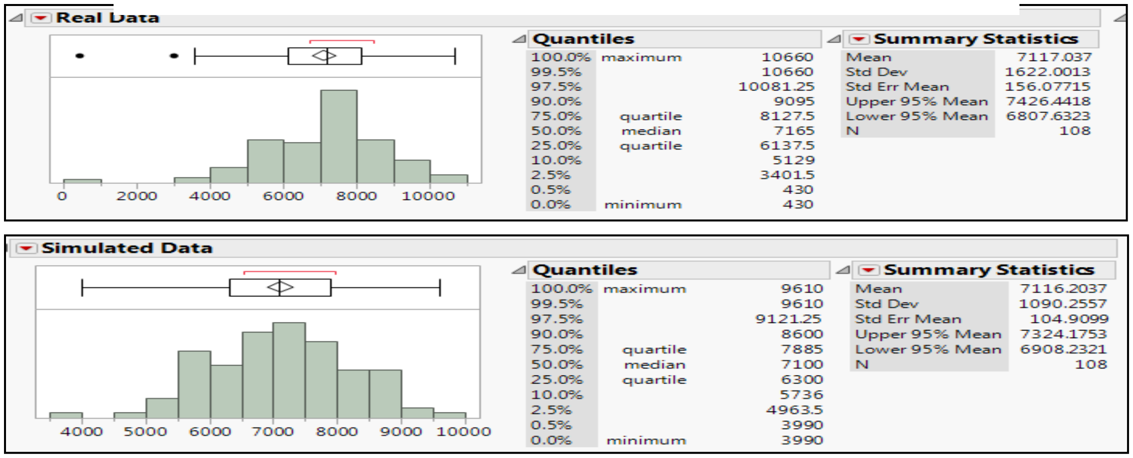


Figure 6. Normal Probability Plot for Simulated Data Results



Real Data

Quantiles

100.0%	maximum	10660
99.5%		10660
97.5%		10081.25
90.0%		9095
75.0%	quartile	8127.5
50.0%	median	7165
25.0%	quartile	6137.5
10.0%		5129
2.5%		3401.5
0.5%		430
0.0%	minimum	430

Summary Statistics

Mean	7117.037
Std Dev	1622.0013
Std Err Mean	156.07715
Upper 95% Mean	7426.4418
Lower 95% Mean	6807.6323
N	108

Simulated Data

Quantiles

100.0%	maximum	9610
99.5%		9610
97.5%		9121.25
90.0%		8600
75.0%	quartile	7885
50.0%	median	7100
25.0%	quartile	6300
10.0%		5736
2.5%		4963.5
0.5%		3990
0.0%	minimum	3990

Summary Statistics

Mean	7116.2037
Std Dev	1090.2557
Std Err Mean	104.9099
Upper 95% Mean	7324.1753
Lower 95% Mean	6908.2321
N	108

Figure 7. Comparasion of Real and Simulated Data

Figure 7. Real and Simulated Data depicts the simulated predicted score data against the human-in-the-loop score data and shows that the means over-lap. The results of the two-sample t-test comparing simulated and real score data provide a p-value of 0.9965 for the two-tail test, indicating there is not a statistically significant difference between the predicted and measured score. This means that there is insufficient evidence that the model produces results which differ from the real system; therefore, the baseline model is considered valid.

Experimental Design and Alternative Model Development

To capture differing reliability rates, the baseline model was modified in several ways. First, a new task node was created called "Travel to incorrect planet." This task accounts for a ship traveling along an incorrectly drawn route. The node assumes that the time a ship spends traveling along an incorrect route is distributed as a

Weibull distribution with the following parameters (5, 1). This task assumes that the amount of time it takes a ship to travel to an incorrect planet will be similar to the time it takes a ship to travel to the correct planet. The differing rates of reliability are captured in the “Travel if by auto draw” node. A random number is generated and compared against the following reliability rate setting. The model is run with fifteen reliability rate settings: 50%, 60%, 70%, 80%, 90%, 91%, 92%, 93%, 94%, 95%, 96%, 97%, 98%, 99%, and 100%. The ending effects account for either correctly drawing the route or sending the ship to the “Travel to incorrect planet” node. The “Travel to incorrect planet” node feeds into the existing “Redraw automated route by human player” node (or the “Collision” node in the event that a collision occurs when traveling along the incorrect route, prior to human correction). The “Redraw automated route by human player” node captures the human redrawing a route to correct the erroneous route due to automation failure. This node has the same task time distributions and probability outcomes as any other human redraw. Thus, it is assumed that the redraw behaviors are the same as when the ship is traveling on a correct route. Figure 8. shows the alternative model.

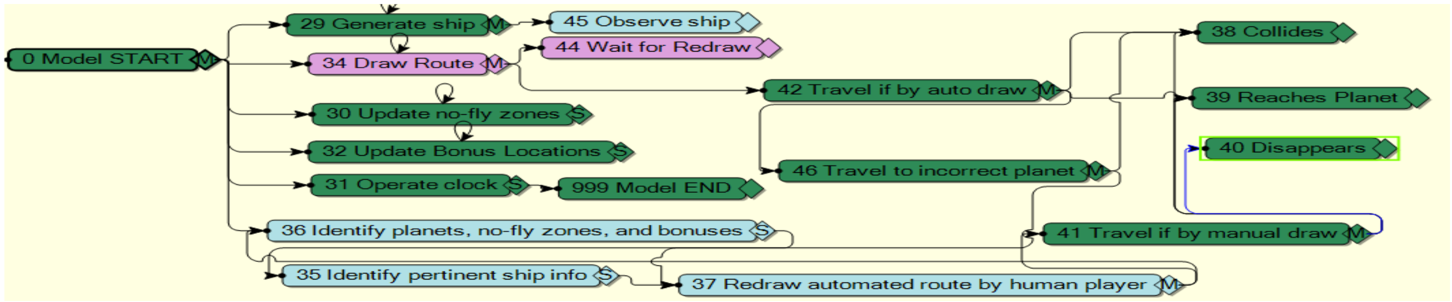

0 Model START
29 Generate ship
45 Observe ship
44 Wait for Redraw
34 Draw Route
30 Update no-fly zones
32 Update Bonus Locations
31 Operate clock
999 Model END
42 Travel if by auto draw
46 Travel to incorrect planet
38 Collides
39 Reaches Planet
40 Disappears
41 Travel if by manual draw
36 Identify planets, no-fly zones, and bonuses
35 Identify pertinent ship info
37 Redraw automated route by human player

Figure 8. Alternative Model in IMPRINT

Results and Discussion

A One-way ANOVA was performed to determine if there was a statistical difference between each of the score means of the automation reliability rates. The analysis of variances showed that the effect of automation reliability significantly influenced score, *F*(5, 251) = 996.9143, *MSE* = 1.4277e+9, *p* < .0001. Figure 9. and Table 1. provide the results of the Tukey’s HSD for the team score at each of the predetermined reliability rates: 100% (*M* = 7116.203, *SD* = 1090.26), 99% (M = 7110.8, SD = 1151.71), 98% (M = 6949.4, SD = 1132.31), 97% (M =

6822.1, SD = 1082.401), 96% (M = 6523.5, SD = 1071.66), 95% (M = 6067.3, SD = 1188.69), and 94% (M = 5896.9, SD = 1192.15), 93% (M = 5589.4, SD = 1101.86), 92% (M = 5309.8, SD = 1173.66), 91% (M = 4601.6, SD = 1214.21), 90% (*M* = 4119.6, *SD* = 1247.23), 80% (*M* = 2864.4, *SD* = 1246.55), 70% (*M* = 1394.9, *SD* = 1130.71), 60% (*M* = 311.0, *SD* = 1194.84), and 50% (*M* = -1250.5, *SD* = 1204.30). Based on a Tukey's value of *CD* = 2.8534, all groups differed statistically from all other groups. As expected, with decreasing levels of automation reliability, the overall performance decreases.

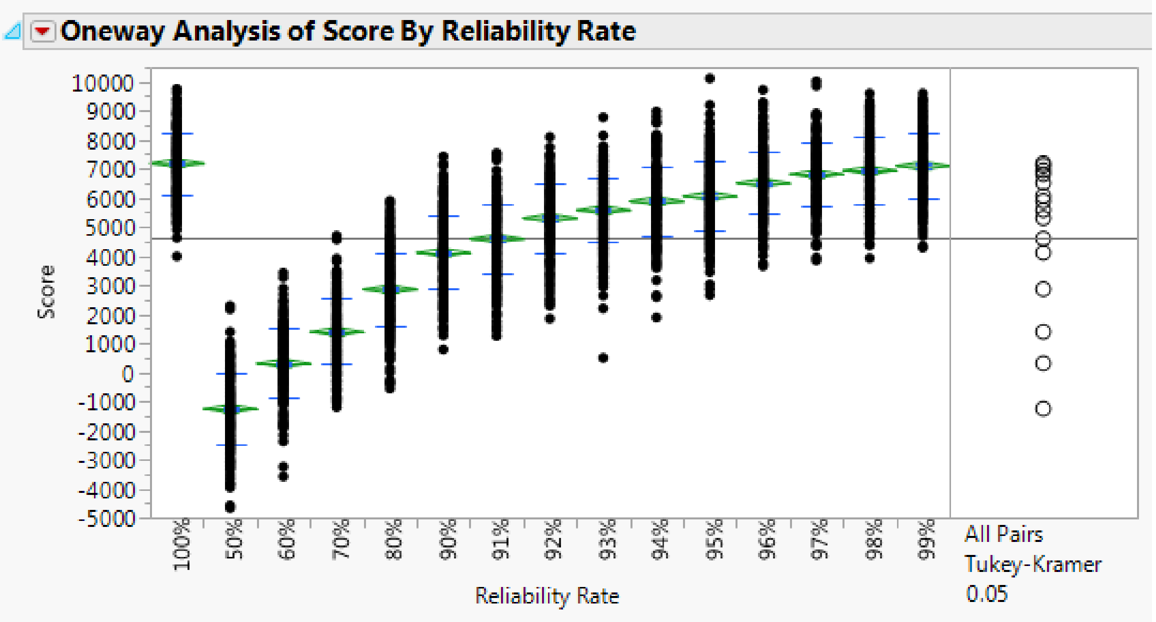


Figure 9. Means of Simulated Reduced Automation Reliability Rates

Table 1. Tukey Table for Varying Reliability Rates and Score

Reliability Rate		Score	
One-way ANOVA	Mean	[F(5, 251) = 996.9143, p = 0.000]	
		95% CI	Tukey Groupings
100%	7116.10	(6908,7324)	A
50%	-1250.50	(-1400,-1101)	B
60%	311.00	(162,459)	C
70%	1394.90	(1254,1536)	D
80%	2864.40	(2709,3019)	E
90%	4119.60	(3964,4275)	F
91%	4601.60	(4458,4745)	G
92%	5309.80	(5166,5454)	H
93%	5589.40	(5446,5733)	I
94%	5896.90	(5753,6041)	J
95%	6067.30	(5924,6211)	K
96%	6523.50	(6380,6667)	L
97%	6822.10	(6678,6966)	M
98%	6949.40	(6806,7093)	N
99%	7110.80	(6967,7255)	O

The purpose of the research is to explore how human-automation team performance is affected by the level of automation reliability. Due to the inter-connected nature of teaming, this research hypothesized that reliability would negatively impact team performance. However, it was expected that imperfect, but highly reliable (e.g. 90% reliable) automation would make a positive contribution to the human-automation team. Since each reliability rate produced statistically significant lower team performance, this indicates that the human-automation team performance is highly sensitive to the automation's reliability level. In the human-only trials, the players' mean score was 5027. Thus only highly reliable automation (>91% reliable) automation is effective in aiding human-automation team performance. All of the reduced reliability scenarios less than 91%, produced mean scores lower than the human-only scenarios, thus the automated teammate was hindering, rather than helping, the team's performance when the reliability rate was 90% or less.

The trend in the data aligns with Scerbo (1996) who found that with certain teaming tasks, the reliability of the automated teammate must be greater than 95% reliable (Scerbo, 1996) in order to benefit from the automation. Although other researchers, such as Wickens and Dixon, have found benefits in automation with reliability levels as low as 70% (Dixon & Wickens, 2006), these studies typically involve the use of an automated decision aid. The finding of this research supports the hypothesis that human-automation teaming may require higher levels of automation reliability than traditional warning or advisory automation aids.

Conclusions

This study sought to use simulation to understand the effects of automation reliability in a collaborative human-automation teaming scenario. The use of SysML modeling provided the framework for all tasks performed in the application environment. Implementing the SysML task network into IMPRINT enabled a simulation-based analysis of automation reliability. As anticipated, lower levels of automation reliability result in lower levels of human-automation team performance. While perfect automation enabled improved performance over human-only scenarios, reliability thresholds at 90% and below negated the value of the automation. The implication of this finding could affect autonomous system design, in that it may necessitate very high reliability requirements. If a system is being designed in which the automation acts as an independent teammate, it is likely that the automation will have to be designed and tested to have a reliability rate that is greater than 90%.

Future Research

The current IMPRINT model captures expected performance outcomes from degraded automation reliability, by modeling the human taking on the task of correcting routes which were incorrectly drawn by the automation. Thus the human is “re-doing” work that was erroneously performed by the automation. In addition, it would be quite likely that reduced automation reliability would reduce the human’s reliance on the automation, and thus instead of just “fixing” the automation’s errors, the human would pre-emptively perform some of the route-generation tasks currently performed by the automation. The current IMPRINT model does not include this additional operator-initiated trust behavior. Prior research demonstrates that reduced reliability results in reduced compliance and/or reliance, which in turn impacts human-automation performance (Dixon and Wickens 2006; Wright et al. 2013; Parasuraman and Miller 2004; Dzindolet et al. 2003; Hoffman, Lawson-Jenkins, and Blum 2006). Further iterations of this model could incorporate these trust-based behaviors.

In an effort to investigate the results presented in this analysis, a follow-on human-in-the-loop study that examines the relationship between reliability rates, performance, and user compliance is currently being conducted. The purpose of the follow-on human-in-the-loop study is to corroborate the findings of this paper, as well as highlight any shortcomings in the model created for reduced reliability. The information gathered from this experiment will help to understand the relationship between trust and reliability for human-automation teaming, and how this relationship differs when the automation is not just a decision aid, but an autonomous teammate.

III. The Effect of Automated Agent Reliability on Human-Agent Team Performance

Abstract

This research investigates the effects of reduced automation reliability rates on human-automation team performance. Specifically, System Modeling Language (SysML) activity diagrams and Improved Performance Research Integrated Tool (IMPRINT) models are developed for a tablet-based game which includes an automated agent teammate. The baseline model uses previously collected data from human-in-the-loop experiments where the automated teammate performs within a range of 50%- 100% reliability. The model found that when agent reliability was 100% the agent was an effective teammate and enabled the human-agent team to achieve statistically improved performance. The model predicted that human agent team performance would degrade when the agent suffered from reduced reliability, providing performance inferior to human only performance for reliability rates less than 92%. A subsequent human-in-the-loop experiment indicated that that the human is able to compensate for small reductions in agent reliability. At both 95% and 100% agent reliability, the human-agent team performed statistically the same However, the human-agent suffered degraded performance when the agent had reliabilities of 70%, 80%, and 90%.

Introduction

The utilization of automation is present throughout many aspects of life in modern society. Most of the objects a person interacts with on a daily basis have either been assembled by an automated system or feature some form of automation or autonomous agency. An example of an automated agent can be found in the pocket of many individuals. The iPhone™ has an automated agent known as Siri™, which can act as an aid to the user. The purpose of automation is to aid a human user in some form. Automation agents are particularly adept at dealing with complex, repetitive tasks (Hoff & Bashir, 2015). There are many definitions of automated agents. One of the more complete definitions of an automated agent comes from Wooldridge and Jennings (Wooldridge & Jennings, 1995).

Wooldridge and Jennings define an automated agent as a system that is software based and has several distinct autonomous properties. First, the agent works without direct intervention of humans and can take action independent of human control. Agents also have the ability to communicate with other agents. Agents can also perceive--and react to--their environment. Additionally, some agents demonstrate a pro-activeness attribute that allows them to anticipate future events (Wooldridge & Jennings, 1995).

One of the driving factors behind how humans interact with agents is the reliability of the agent. When discussing reliability, the parlance of the automation community refers to the failures of the automation (de Visser & Parasuraman, 2011; Dixon, Wickens, & Mccarley, 2006; Lee & See 2004; Maynard & Rantanen, 2005; Merritt, Heimbaugh, LaChapell, & Lee, 2013; Rovira, McGarry, & Parasuraman, 2007). The two common forms of errors made by unreliable automation are misses and false alarms. Misses occur when the automation fails to detect or complete a task. False alarms occur when the automation signals something is wrong or incorrect, but there is not an actual error (Dixon et al., 2006). Both misses and false alarms are most commonly found in diagnostic automation or agent systems that distinguish between states of safety and danger or correct and incorrect (Dixon & Wickens, 2006). The misses and false alarms stem from imperfect sensors, algorithms, noisy data, or probabilistic data in a complex, changing environment (Dixon & Wickens, 2006). However, the rate at which the automation is unreliable can vary greatly.

The effects of varying the reliability of automated aids has been studied for the past several years (Chen & Joyner, 2006; de Visser & Parasuraman, 2011; Maynard & Rantanen, 2005; Ross, Szalma, Hancock, Barnett, & Taylor, 2008; Rovira et al., 2007; Rovira & Parasuraman, 2010; Wickens & Dixon, 2007). The research conducted in this area supports a nearly universal conclusion that human-automation team performance and the willingness of the user to trust and use the automated aid is greater when the automated aid has greater reliability (de Visser & Parasuraman, 2011; Ross et al., 2008; Rovira et al., 2007; Rovira & Parasuraman, 2010; Wickens & Dixon, 2007). However, there is little to no research conducted in what happens to user trust and human-agent team performance when the automated agent acts like a teammate rather than just an aid or diagnostic tool, which means the agent does not have as much autonomy as an agent that acts as a teammate.

Research has shown that even with varying rates of automation reliability, human users are able to respond fairly accurately to the changes in automation reliability in terms of how they trust and rely on that automation (Ross et al., 2008). This innate ability of humans to perceive automation reliability allows for the human user to maximize overall performance across a range of automation reliability rates down to 70% , which has been shown to be the threshold of usefulness (Ross et al., 2008; Wickens & Dixon, 2007). However, most users struggle with identifying the first failure of automated systems (Rovira et al., 2007). It is important to note that even if an automated system operates at 100% reliability, the human operator can perceive the reliability to be less than 100%. This happens when the automated system does not behave as the user expects the system to, even when the automated system has been designed correctly (Ho, Kiff, Plocher, & Haigh, 2005). An example of this can be seen in automated medication dispensers such as the Honeywell Independent LifeStyle Assistant (I.L.S.A). One user commented on I.L.S.A. not recognizing a change in her usage pattern, a feature it was not designed to provide. This perception of unreliability caused the user to not trust I.L.S.A and not use the system (Ho et al., 2005).

One of the effects of unreliable automation surfaces in user compliance with the automation or agent. Compliance relates to the human operator’s response when an alarm or signal sounds, whether that alarm or signal is true or false (Chen & Barnes, 2014; Chen & Joyner, 2006; Dixon & Wickens, 2006; Dixon et al., 2006). A compliant human operator is an operator who rapidly switches his or her attention from concurrent activities to the alarm domain (Dixon & Wickens, 2006). The operator may then immediately initiate an alarm-appropriate response, such as hitting the snooze button on an alarm clock (Dixon & Wickens, 2006). Research has shown that as the rate of false alarms increases, the operators experience a reduction in compliance (Chen & Joyner, 2006; Dixon & Wickens, 2006; Dixon et al., 2006). This compliance reduction can result in longer response times to automation alerts or alarms (Dixon et al., 2006). However, as the false alarm rate increases to a certain threshold--

which is different for every operator--the operator will start to disregard the false alarms entirely. This phenomenon is known as the “cry wolf” effect or “alarm fatigue” (Dixon et al., 2006). Multiple studies support the conclusion that compliance is the driving factor in task time completion, as well as accuracy in completing the task. In all of the aforementioned studies, the researchers used an automated aid that provided some form of recommendation or alarm. This research needs to be expanded beyond automation alarms to address what happens to user compliance and the human-agent team when an agent that acts as a teammate has imperfect reliability.

Purpose

The purpose of this research is to explore and garner further understanding into how human-agent team performance is affected by varying levels of agent reliability. As previously discussed, agent reliability has been studied in previous experiments; however, the current body of literature does not provide a great deal of insight into environments in which the agent acts as an equal teammate to the human user rather than merely an aid to the user. The concept of a teammate consists of the notion of an agent that is able to sense the environment, make decisions, and complete the task, without the human user providing supervision. This paper hypothesizes that when the automated teammate has reduced levels of reliability; overall team performance will suffer. However, it is expected that just as less than100% reliable human teammates can still be an asset to a team, a less than100% reliable automated teammate will also make a positive contribution to the human-automation team.

Application Environment

The application environment was a tablet based game, *Space Navigator*. This environment features the ability for the human and the agent to interact and work together as a team rather than the human and agent working independently of each other. This game requires tasks similar to terminal approach in an air traffic control situation and is similar to the custom route-creation games Harbormaster and Flight Control. The user activity during the game consists of drawing trajectories from spaceships to planets. The activities in the game are completed by the human, agent, or both. The game environment contains four static planets on the screen. These planets are uniquely colored: red, green, blue, and yellow. Space ships are generated on the sides of the screen and proceed to move through the game environment. Each spaceship is red, green, blue, or yellow and is generated at a rate of one spaceship every two seconds. The spaceships are generated until the allotted time of four minutes is over. The objective of the game is to have each spaceship reach the corresponding colored planet (e.g. red spaceship to red planet). The human user drags her or his finger along the touch screen, starting at the spaceship to create a path for the spaceship to follow. The spaceship follows the trajectory at a constant rate until the planet is reach, a collision with another spaceship occurs, or the spaceship travels off screen. The trajectories may be redrawn. This is often done to avoid collisions or account for dynamic changes in the environment, such as the generation of bonus orbs (small orbs that are generated randomly throughout the game). Points are rewarded to the player when a spaceship successfully reaches the destination planet or collects a bonus orb. If the spaceship reaches the correct planet, the spaceship will disappear immediately. If the spaceship collides with another spaceship, points are lost and each spaceship involved in the collision immediately disappears. If the spaceship collects one of the bonus orbs, the user receives a point bonus. The player can also lose points for crossing 'no-fly zones' which are shaded boxes near the planets.

This game environment features an automated agent that draws straight-line trajectories from spaceships to the planets immediately after the ships spawn. The agent is not capable of collision avoidance, thus enabling the agent to serve as a team member but not a perfect solution capable of replacing the human player. Therefore, human involvement allows for better performance than if the agent works alone. Figure 10. depicts the game environment with an annotated screen capture from Space Navigator, illustrating the elements of the game interface.

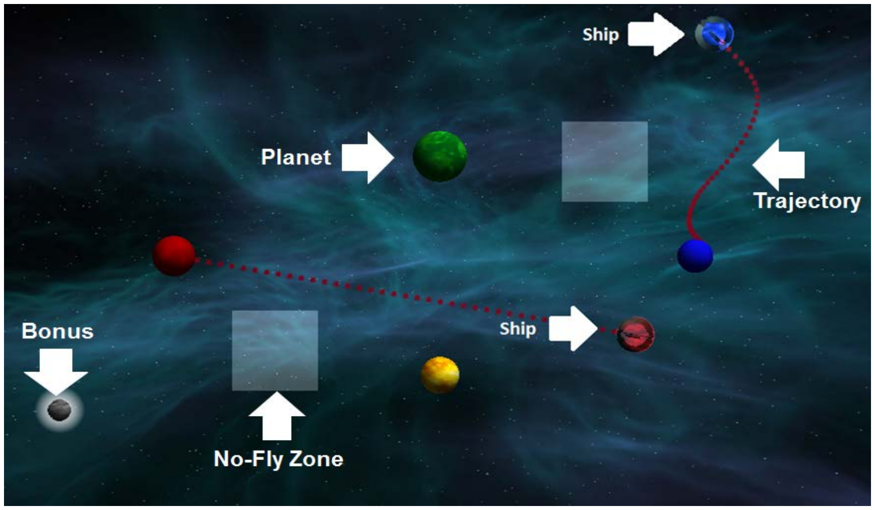


Figure 10. Space Navigator Environment

Method

As previously stated, this research explores the effect of agent relability on human-agent team performance. Specifically, this research aims to construct a model capable of predicting the change in score which occurs as a result of changes in agent reliability. The method includes three stages of experimentation and modeling, including: 1) building and validating a baseline model, 2) conducting a reduced reliability simulation, and 3) conducting a reduced reliability human in the loop experiment. The first stage consisted of performing a human-in-the-loop experiment (HITL) in which participants were exposed to the application environment with an automated agent performing with 100% reliability. The data from the HITL experiment was used to create a model and simulation to replicate the HITL. Finally, the model was validated by evaluating the results from the simulation against the data collected in the HITL experiment. The results from the first stage are presented in Chapter 2 and Appendix 2. Using the validated model from the first stage, updated reduced reliability automated agent simulations were created in the second stage. The third stage consisted of conducting a second HITL study in which the participants were subject to the same application environment, but with an automated agent that suffered from reduced reliability. Specifically the reduced reliability rates used in the stage two simulations. Only the results from the last two stages are presented in this Chapter. Figure 11. depicts this methodology graphically.

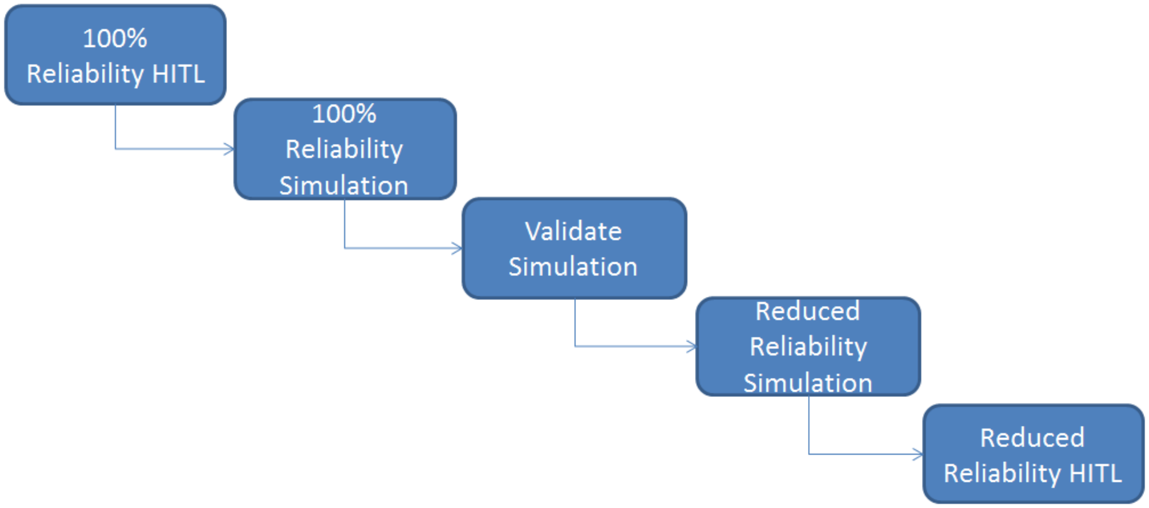


Figure 11. Overview of Method

Independent Variable

The independent variable of interest is the reliability of the automated agent. In the *Space Navigator* environment, an automation error is defined by the automated agent drawing a route from a ship to an incorrect planet (i.e., drawing a trajectory from a ship of one color to a planet of a different color). Collisions resulting along a trajectory drawn by an agent is not considered an error. As previously established, the reliability of an automated agent plays a vital role in the human-agent team (Parasuraman & Miller, 2004; Ross et al., 2008; Rovira & Parasuraman, 2010; Wickens & Dixon, 2007). The current state of literature reflects that when either member of the human-agent team suffers a reduction in reliability, the overall team performance decreases. (Antifakos, Kern, Schiele, & Schwaninger, 2005; Dixon & Wickens, 2006; Ross et al., 2008; Singh, Tiwari, & Singh, 2009; Wickens & Dixon, 2007). This research explores a range of reliabilities (100%, 95%, 90%, 80%, and 70%). However, the majority of the aforementioned literature focuses solely on agents that act as decision aids, rather than teammates. This research focuses on the impact of agent reliability on human-agent team performance when the agent behaves as a teammate rather than a decision aid. The researchers hypothesize that the relationship between agent reliability and human-agent team performance will be similar to that of the research previously conducted, that is, significant degradation in performance is expected to occur for reliability rates less than 90%.

Dependent Variable

The dependent variable of interest is the score of the human-agent team. Although there are several metrics with which to measure performance, the score of the human-agent team is one of the easiest to understand and study. In the *Space Navigator* environment, score is a direct reflection of how well the human-agent team performed the primary task of directing spaceships to planets. The human-agent team gains 100 points for every spaceship that reaches the correct planet and 50 points for every bonus collected. However, the team loses 100 points for every spaceship lost in a collision and 10 points per second each spaceship spends in a no-fly zone.

Stage 1: 100% Reliability Human-in-the-Loop Experiment (HITL)

The first stage in the procedure was conducting a 100% reliable agent human-in-the-loop experiment. This experiment was previously conducted and the results from this paper have been previously reported in Bindewald, Miller, and Peterson (2014). The experiment included 36 participants with a mean age of 32.5 years and a range of 22 to 39 years with a total of 30 males and 6 females. The experimental procedure used in this research was a within-subjects design. The design included each participant completing 17 five-minute games of *Space Navigator*. In an effort to address learning effects, the first five games contained no interaction for the automated agent. These games were consisted training games for the participant. After the training games, the participants completed three experimental sessions. Each experimental session contained 4 five-minute games. The first game without an automated agent and the following three with three different types of automated agents, each with a unique route-generation strategy. Although three different automated agents were used, only the data from the automated agent with the straight-line route-generation strategy is used in the current study. The straight-line automated agent was configured to draw straight-line routes, meaning the most direct path from the spaceship to the correct planet. This automated agent was 100% reliable, meaning that it always drew routes from space shipes to the planet of corresponding color. This human-in-the-loop experiment provided 108 data points which were used to create the 100% reliability simulation. For full results of this research, see Bindewald, Miller, and Peterson (2014). Using the data from the 100% reliability HITL experiment, several models were created. The first model that was created was an activity diagram. The activity diagram is depicted in Figure 12. . The diagram shows the activities of both the human and the agent as well as background functions that enable the game to be played such as updated bonuses and no-fly zones.

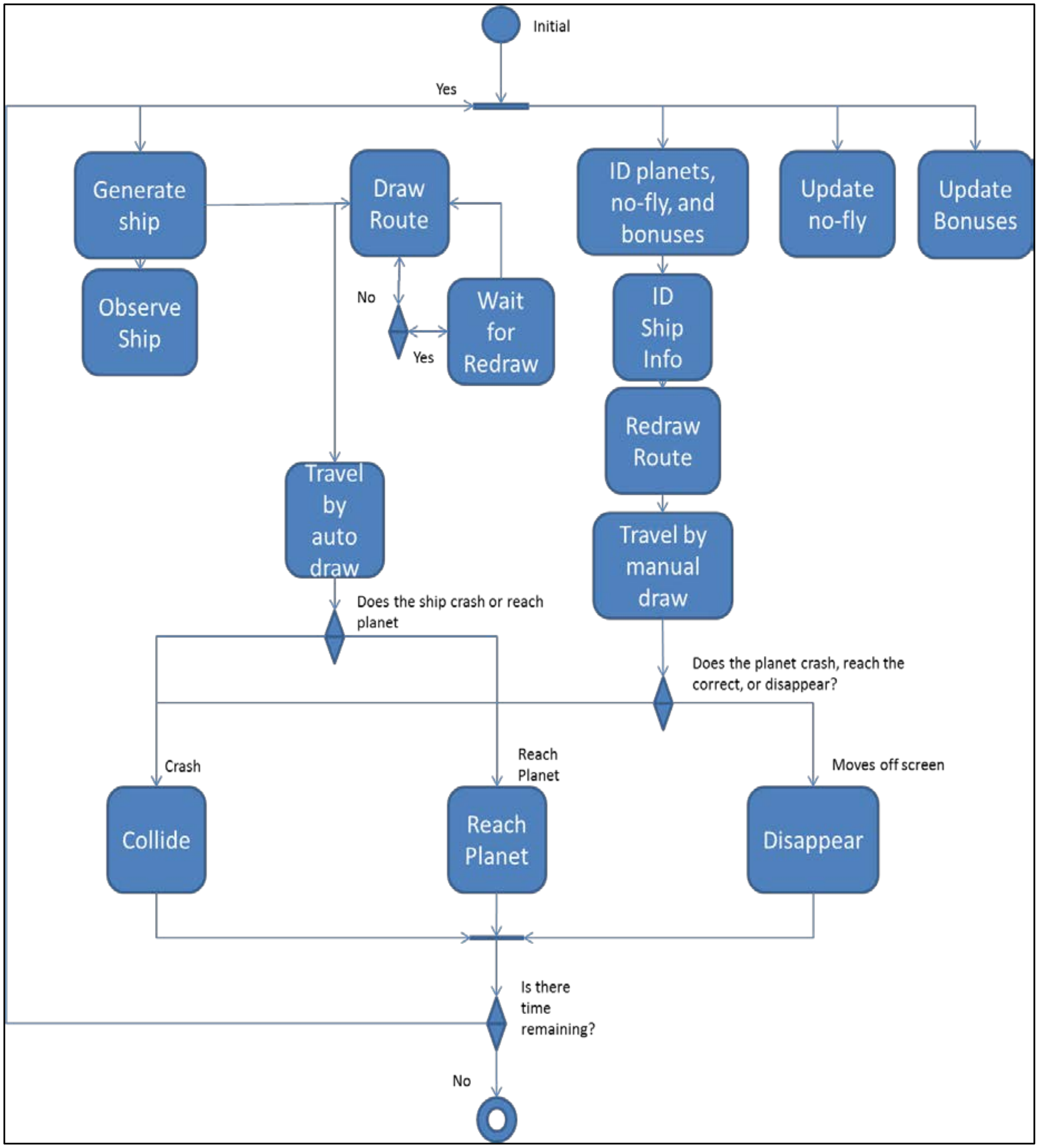


Figure 12. Activity Diagram for Space Navigator

The activity diagram was then translated into a discrete-event simulation using the Improved Performance Research Integrated Tool (IMPRINT). IMPRINT provided a simulation environment in which human-automation team performance could be analyzed. IMPRINT allowed for all of the game environment aspects to be captured

(Game clock, bonuses, no-fly zones, and ship generation). IMPRINT also allowed activities of both the automated agent and the human to be modeled and simulated. The model used the number of spaceships on screen as a function of the number of collisions the user would experience. This means that as the number of spaceships on screen increased, the probability of collisions increased. For more detailed information about IMPRINT, the model used, and results, Chapter 2 and Appendix 2. After the modeling and simulation in IMPRINT were completed, the results of the simulation were validated against the real data collected in the 100% reliability HITL experiment. Using several statistical measures such as the Shapiro-Wilk goodness-of-fit test and two-sample t-tests, it was determined that the simulation produced scores that were valid and consistent with the data collected from human participants. For more information about the validation, reference Chapter 2 and Appendix 2.

Stage 2: Reduced Reliability Simulation

Once the baseline model was validated, the baseline model was then updated to reflect changes in experimental conditions between the 100% Reliability HITL study and the upcoming Reduced Reliability HITL study. The clock time was reduced to four minutes based on a HITL pilot study which indicated 5 minute games would create a full study that would be too long. The reliability rates were changed to 100%, 95%, 90%, 80%, and 70% based on findings from the rates used in Chapter 2 as well as pilot study findings that the game became excessively frustrating to play with reliability rates less than 70%. The 95% reliability condition was added to garner further understanding into the behavior and performance of the human-agent team between 90% and 100%. In the reduced reliability human-in-the-loop experiment the user does not experience the reduced reliability until 40 seconds into each experimental session. These changes resulted in a Reduced Reliability baseline model and alternate models that reflect all of the experimental conditions in the reduced reliability human-in-the-loop experiment.

Stage 3: Reduced Reliability Human-in-the-Loop Experiment (HITL)

The final stage of the method was conducting a reduced reliability human-in-the-loop experiment, which examined five different reliability rates: 100%, 95%, 90%, 80%, and 70%. The human-in-the-loop experiment included 48 participants with a mean age of 22.6 years and a range of 18 to 38 years. A total of 18 males and 30 females participated. The experimental procedure consisted of a within-subjects design in which each participant completed 4 four-minute training sessions, followed by 12 four-minute games of *Space Navigator*. The training sessions familiarize the participants with playing the game and agent interaction. All training sessions featured a 100% reliability. Following the training, participants completed six experimental blocks. Each block consisted of two games of the same reliability level. The participants were unaware of the reliability rates presented in the experimental blocks. The first block consisted of game play with a 100% reliable agent. After the first block, the user experienced four blocks in a randomized order of each reduced reliability rate until all reliability rates were experienced. The user then experienced a final block with two 100% reliability games. Therefore, the usable data consist of 48 participants' data or 576 scores, each accumulated from a 4 minute game. The mean for each block was used; yielding 288 total data points (48 participants by 6 blocks). All of the instruments used for data collection to include the surveys, questionnaires, and overall experiment overview can be seen in Appendix 3.

Data Analysis

The software package: Statistical Package for the Social Sciences (SPSS) was used to conduct a linear regression on the data provided by the reduced reliability simulation and the reduced reliability human-in-the-loop experiment. The researchers hypothesized that performance of the human-agent team would decrease at a linear rate. The hypothesis is tested using a linear regression. A linear regression was selected due to its ability to predict outcomes using an input variable. The researchers aim to predict the human-agent team score based on the reliability of the automated agent. Before analysis can begin, the assumptions of linear regression must be met. The method for applying the regression is as follows: validate the assumptions of linear regression using the simulated score data, use the data to create a linear regression, and analyze the output. The linear regression is analyzing the relationship between agent reliability and human-agent team performance.

The first step in the method of applying the linear regression is validating the assumptions. Using SPSS, the assumptions for the linear regression were met. The score data used had a continuous dependent variable, linearity for all X and Y relationships, independence of error, homoscedasticity, and normality. Reference Appendix 4 for regression assumptions validation.

Results

The results of this research are decomposed into two sections—results for the reduced reliability simulation and results from the reduced reliability HITL experiment. The results from the reduced reliability simulation are shown to highlight the difference between the data that was produced from the simulation and the data from the HITL experiment.

Results for Reduced Reliability Simulation

Table 2. Descriptive Statisticsshows the descriptive statistics of the score variable. The mean score across all reliability conditions was 4703 with a standard deviation of 1625. This data had a range of 7870 with the minimum score being 580 and the maximum score being 8450. The distribution of the standardized residuals of the mean scores can be found in Figure 14. Distribution of Means. The figure shows that the distribution follows a normal distribution, which meets the assumptions of normality for linear regression.

Table 2. Descriptive Statistics for Reduced Reliability Simulation

	MEAN	Std. Dev	Range	Min	Max
Score	4703	1625.03	7870	580	8450

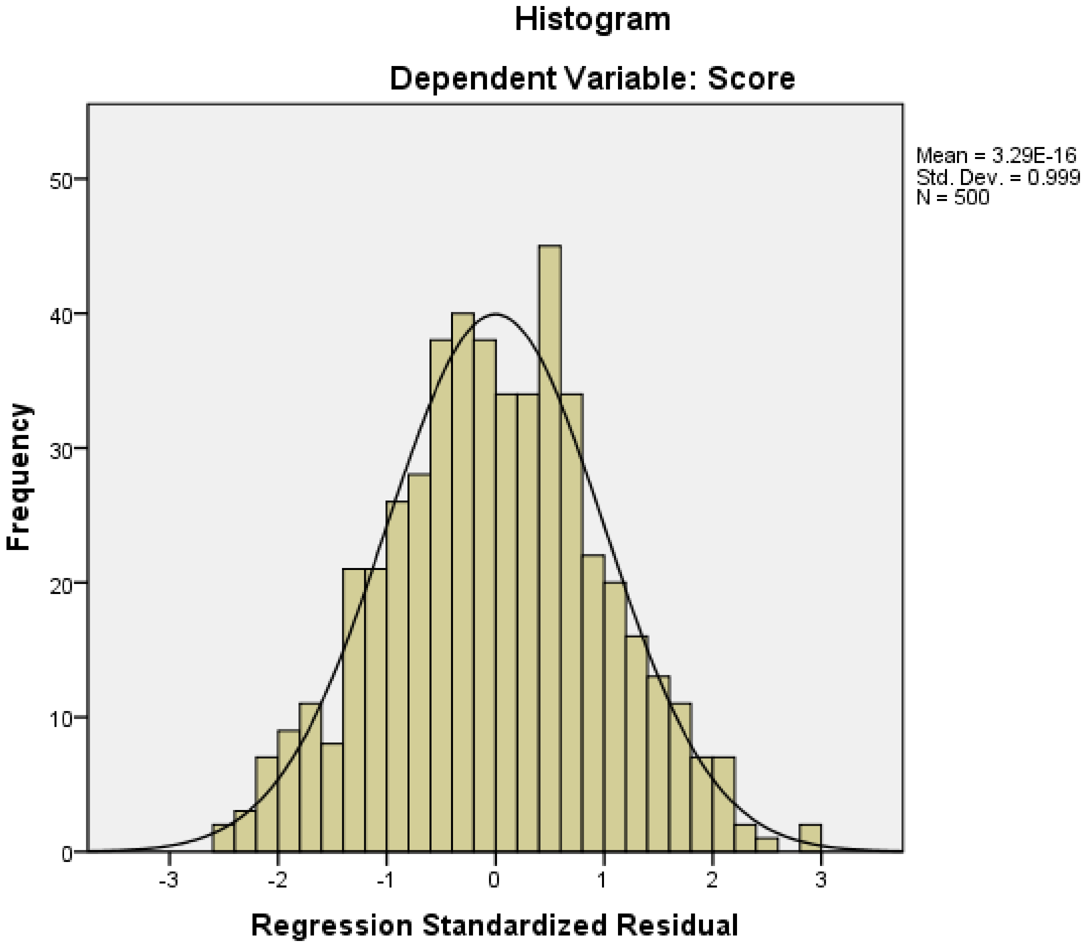


Figure 13. Distribution of Standardized Residuals for Means from Reduced Reliability Simulation

Table 3 depicts the mean scores at each of the reliability conditions of interest as well as the number of data points at each condition and the standard deviation. As expected, the highest mean score occurs at the 100% agent reliability condition. The mean score then significantly decreases as the agent reliability decreases. The minimum mean occurs at the 70% condition which is expected; however, it is still a positive score which means more ships made it to the correct planet than collided. It is acknowledged that this is a positive score, but a very low score.

Table 3 Simulated Mean Score at Each Reliability Condition

Reliability	Mean Score	Std. Dev
70%	2618	1038.05
80%	3907	1038.24
90%	5098	959.68
95%	5690	978.71
100%	6202	913.93

The linear equation created by SPSS can be seen in Table 4, which includes the coefficients for both the constant and the reliability. It is noted that both of these coefficients are significant with significance values of .000. Additionally, this significance demonstrates that the entire regression is significant. This model shows that for every one percentage point increase in reliability, the human-agent team performance will increase by 199.962 points. This model predicts that human-agent team performance will decrease at a linear rate as agent reliability decreases.

Table 4 Linear Regression for Simulated Data

Model		Unstandardized Coefficients	Standardized Coefficients	t	Sig.
1	(Constant)	-5733.775		-15.994	0.00
	Reliability	119.962	.796	29.334	0.00

The results of the simulation provided valuable insight into the sensitivity of the human-agent team's performance. As previously stated, for a one percentage point decrease in reliability, the human-agent team performance will decrease by 199.962 points. This finding indicates that the human users are unable to compensate for decreases in agent reliability. This finding supports the hypothesis that near 100% reliability conditions are required for human-agent team performance to be favorable.

Table 5 depicts the ANOVA Table for linear regression of the simulated data. Table 4 ANOVA for Simulated Data shows that the model created is significant with a p< .000. The p-value of less than .000 indicates the model is a viable model that can be used for analysis. Additionally, there were 499 degrees of freedom with an F value of 860.51.

Table 5 ANOVA for Reduced Reliability Simulated Data

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	8.347E8	1	8.347E8	860.51	0.00
	Residual	4.830E8	498	969977.30		
	Total	1.318E9	499			

In addition to the ANOVA table, a complete model summary was created in which the R Squared value can be seen. Table 6 shows the model summary. The R-squared value is 0.633 meaning that the linear regression model accounts for 63.3% of the variance in the data. The score data has a high degree of variability due to the many factors at play including workload on the user, user strategy, differences in users, and so forth. This model was able to explain 63.3% of the variability using only the reliability of the agent as a predictor. It is acknowledged that the R-squared value is not in the 90th percentile, but never-the-less is an adequate value. Additionally, the standard error for this model was 984.874.

Table 6 Model Summary for Reduced Reliability Simulated Data

Model	R	R-Squared	Std Error
1	0.796	.633	984.874

Results for Reduced Reliability HITL Experiment

Table 7 shows the descriptive statistics of the score variable. The mean score across all reliability conditions was 6102 with a standard deviation of 1406.20. This data had a range of 8200 with the minimum score being 2330 and the maximum score being 10530. The range and standard deviation are similar to that of the simulated data. However, at reliability conditions less than 100% differences between the simulation and HITL emerged. This is likely due to the users' ability to compensate for the shortcomings of the agent. The distribution of the standardized residuals of the mean scores can be found in Figure 15. Distribution of Means for Reduced Reliability HITL Experiment. The figure shows that the distribution follows a normal distribution, which meets the assumptions of normality for linear regression.

Table 7. Descriptive Statistics for Reduced Reliability HITL Experiment

	Mean	Std. Dev	Range	Min	Max
Score	6102	1406.20	8200	2330	10530

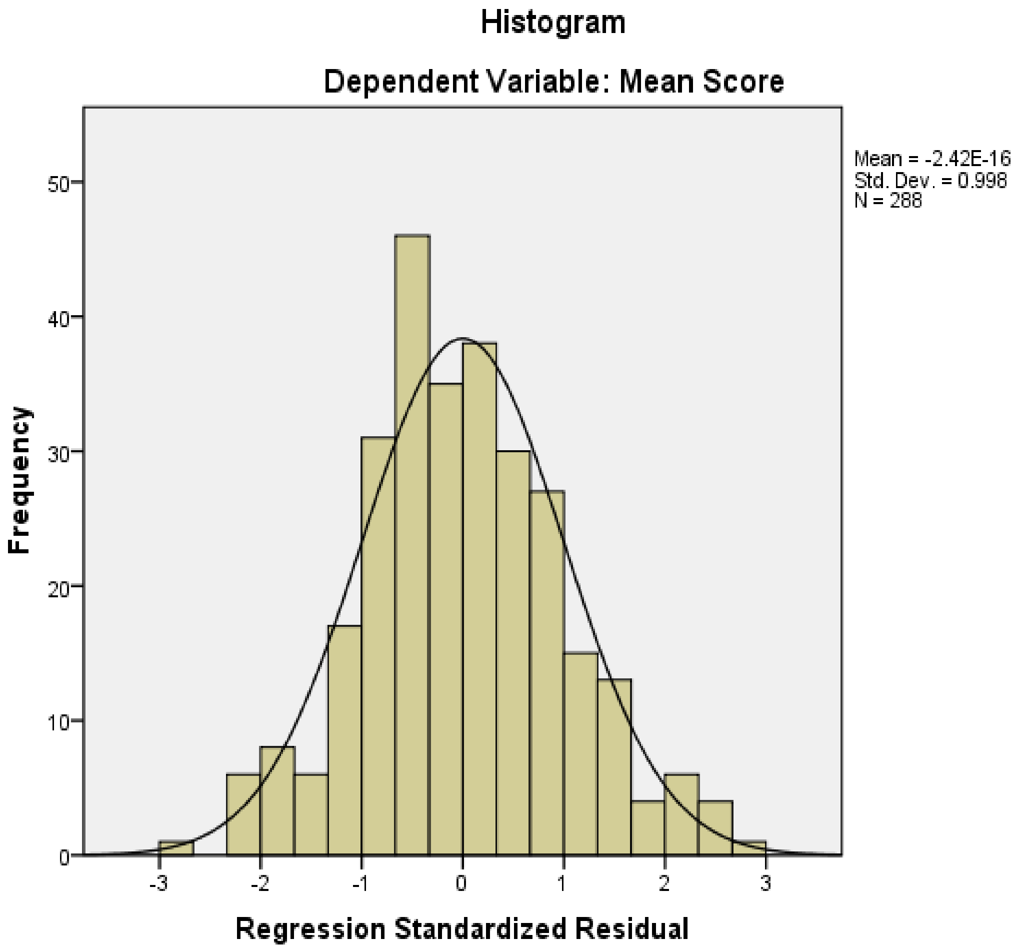


Figure 14. Distribution of Standardized Residuals for Means from Reduced Reliability HITL Experiment

Table 8 depicts the mean scores across the entire range of reliability conditions in the reduced reliability HITL experiment. It should be noted that the 95% and 100% reliability conditions were demonstrated to be statistically the same. The 95% reliability condition had a mean score of 6436 points with a standard deviation of 1459.71 whereas the 100% reliability condition had a mean score of 6435 with a standard deviation of 1214.62. This finding indicates that the human users were able to compensate for the reduction in agent reliability. This finding was not seen in the simulated data. This finding does not support the hypothesis that the human users would be very sensitive to changes in agent reliability, rather the human users are able to compensate for about a 5% reduction in agent reliability. Additionally, the mean score at the 70% reliability condition was significantly higher than the simulation had predicted. This is also likely due to the human users' ability to make up the agent's lack of reliability. However, the model predicts well at the 100% reliability condition.

Table 8 Mean Score for Reliability Condition in Reduced Reliability HITL Experiment

Reliability	Mean Score	Std. Dev
70%	5406	1444.38
80%	5717	1476.53
90%	6184	1459.71
95%	6436	1304.44
100%	6435	1214.62

Table 9. depicts the for Repeated Measures Reduced Reliability HITL Experiment Model. It can be seen that the model is significant with a p< 0.00. Additionally, the degrees of freedom for this model is 287 with an F value of 24.87. The primarly source of variation stems from the reliability conditions experienced by each participant in the experiment which is seen by the within groups statistics in the table. It is acknowledged that individual differences

between participants also contributed a sizeable amount of variation to the model which is seen in the sum of squares of the between groups category.

Table 9 ANOVA for Repeated Measures Reduced Reliability HITL Experiment Model

Source of Variation	*SS*	*df*	*MS*	*F*	*P-value*	*F crit*
Between Groups	27698233	5	5539647	2.893907	0.014475	2.246015
Within Groups	5.4E+08	282	1914245			
Total	5.68E+08	287				

Table 10 depicts the R-values for the linear regression model. This adjusted R-squared value is very low, 0.077. This means that the model is only able to explain 7.7% of the variance in the data. Although the model is statistically significant, the low adjusted R-squared value indicates that there are more factors required to accurately predict human-agent team performance. The simulated data had a much higher adjusted R-squared (63.3%) but this was likely due to the assumptions of the simulation. The researchers believe the primary cause of this difference is the ability of the human to compensate for reductions in agent reliability.

Table 10 R-values for Reduced Reliability HITL Experiment Model

Model	R	R-Squared	Adjusted R-Squared	Std Error
1	0.283	.080	.077	1351.15

Table 11 shows the linear regression model for the Reduced Reliability HITL experiment. The table includes the coefficients for both the constant and the reliability. Note that both of these coefficients are significant with p<

0.000. Additionally, this significance demonstrates that the entire regression is significant. This model shows that for every one percent increase in reliability, the human-agent team performance will increase by 36.221 points. This model predicts that human-agent team performance will decrease at a linear rate as agent reliability decreases. When compared against the simulated data model, this model is not as sensitive to changes in agent reliability.

Table 11 Linear Regression for Reduced Reliability HITL Experiment

Model		Unstandardized Coefficients	Standardized Coefficients	t	Sig.
1	(Constant)	2872.369		4.402	0.00
	Reliability	36.221	.283	4.987	0.00

Discussion

The purpose of this research is to explore and garner further understanding into how human-agent team performance is affected by varying levels of agent reliability in human-agent teaming environments. This research is focused specifically on environments where the automated agent acts as a teammate rather than a decision-aid. The two step methodology of this research provided many opportunities to understand the intricacies in how automated agent reliability plays a role in the human-agent team. The first step in this research modeled and simulated the effect of reduced agent reliability on the human-agent team performance. The simulation model predicted that performance would decrease at a nearly linear rate when agent reliability was degraded. However, the data provided from the reduced reliability human-in-the-loop experiment indicated the user was not as sensitive to changes in agent reliability as the simulation had predicted. Table 12 and Figure 16. shows the means of both the simulation and the reduced reliability HITL experiment.

Table 12 Simulated vs Actual Mean Scores

Reliability	Simulated Mean Score	Actual Mean Score
70%	2943.3	5405.7
80%	4048.6	5716.7
90%	5272.6	6184.0
95%	5842.1	6436.3
100%	6202.1	6435.0

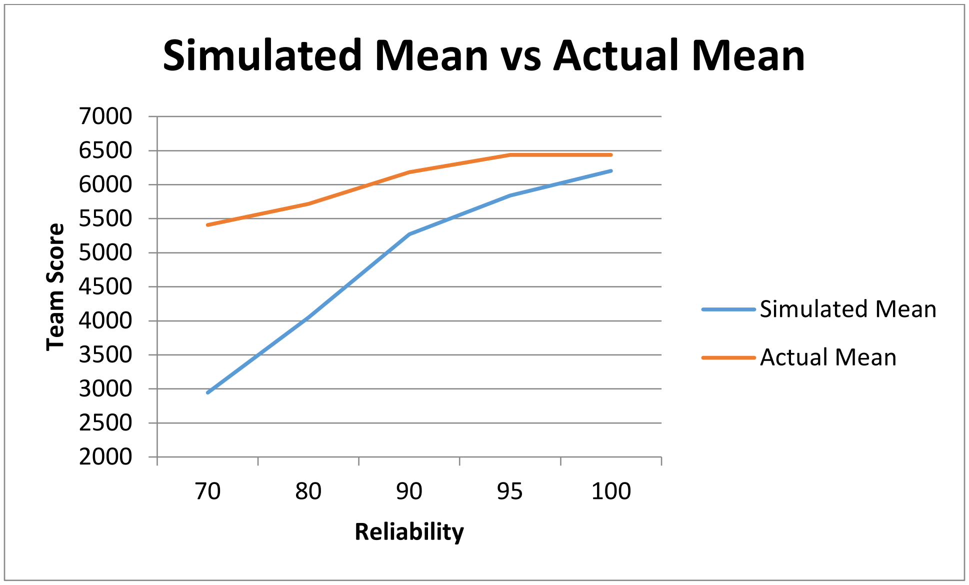


Figure 15. Simulated Mean vs Actual Mean

The reduced reliability human-in-the-loop experiment demonstrated that performance is degraded when the agent suffers from reduced reliability, but the degradation of performance follows more of a plateau into a linear rate. It is hypothesized that this shape of the performance curve is due to the human user being able to absorb a small part of the automated agent’s task load before overall performance is degraded. The human users were more capable of compensating for the changes in automated agent reliability than the model had predicted. As previously mentioned, the model used the number of spaceships on screen as a function of the number of collisions the user would experience which means that as the number of ships on screen increased, there would be a greater probability of collisions. However, this assumption did not address many other factors which are believed to contribute to the discrepancy in modeled and actual performance. One of the factors that the model struggled to capture was how human behavior changed in response to a change in a teammate's behavior. The human user was able to make up for the discrepancy in agent performance to a certain extent. This human behavior highlights the need to capture the dynamics of teaming environments. The model changed the automated agent's behavior but did not account for the overall change in the team's behavior. Feature research could examine revising the model to address the human ability to partially compensate for automated agent unreliability.

Conclusion

This research sought to use both simulation and experimentation to explore and understand the effects of automated agent reliability in a human-agent team environment. The research used a three-step methodology to address both simulation and experimentation. The first step examined data from a previously conducted 100% reliability human-in-the-loop experiment and created a model. The model used SysML task networks and IMPRINT to enable a simulation-based analysis of automation reliability. The results of the simulation-based analysis indicted that human-agent team performance decreased at a nearly linear rate. The results also suggested that in this environment high levels of automation (>95%) are necessary for desirable human-agent team performance.

The second step of this research used simulation to explore the relationship between automated agen reliability and human-agent performance. Several changes were made to the model used in the 100% reliability model. The simulation was then conducted. As expected, the highest mean score occurs at the 100% agent reliability condition. The mean score then significantly decreases as the agent reliability decreases. The minimum mean occurs at the 70% condition which is expected; however, it is still a positive score which means more ships made it to the correct planet than collided. It is acknowledged that this is a positive score, but a very low score.

The third step of this research consisted of experimentation. A reduced reliability human-in-the-loop experiment was conducted. The experimental data was analyzed and it was found that the results differed from the predicted results from the modeling and simulation. The analysis found that human-agent team performance followed a plateau into a linear rate when subject to reduced agent reliability. The human users were able to adapt to changes in agent reliability more effectively than the model suggested. The findings suggest that in the *Space Navigator* environment, the agent can suffer from reduced agent reliability and have favorable human-agent team performance. It was found that 95% agent reliability yielded statistically identical results to 100% agent reliability. The implications of these findings could affect many aspects of how users and designers interact and design automated agents. Agent reliability is only one aspect that drives human-agent team performance. Considerations must be made to address characteristics of the user such as demographic data.

Future Work/ Lessons Learned

This research examined the relationship between reliability and performance. It was found that there is a negative relationship between reliability and performance in human-agent team environments as expected from prior research (Wickens 2007). However, this research also suggests that there are more factors than just reliability that influence human-agent team performance. These additional factors provide many lessons learned that are applicable to both designers of automated agents and those that model and simulate human-agent teams. Special attention should be paid to characteristics of the human user and to the team dynamics. Users vary greatly in terms of performance, demographic characteristics, and their interactions with automation. In modeling situations it is imperative to account for changes in workload and user interaction with reduced reliability agents. This research focused on the agent’s reliability which in turn output data which did not accurately reflect the actual data provided by the experiment. It is for this reason that all aspects of the agent, human user, and team should be accounted for to the maximum extent possible.

IV. Relationships between Human User Demographics and User Trust in an Autonomous Agent

Abstract

Reliability of autonomous agents has been shown to play a pivotal role in the human-agent team. Research has shown that more reliable automation tends to increase human-machine team performance. However, performance is not strictly derived from the autonomous agent’s reliability. Performance may also be impacted by the user’s trust in automation. This research investigates the relationship between demographic factors and trust using the application environment Space Navigator. It was found using a stepwise multiple linear regression that workload (NASA-TLX), gender, education level, and the reliability of the autonomous agent impact the perceived reliability or user trust in the system. When the user experienced higher workload, the user had less trust in the autonomous agent. Females trusted the agent less and more educated users trusted the autonomous agent more. Finally, more reliable agents led to higher levels of trust by human users.

Introduction

One of the factors that affects human performance when interacting with automation is the automation’s reliability (Wickens & Dixon, 2007). Automation is defined as the assignment of the execution of functions previously carried out by a human to a computer (Parasuraman & Riley 1997). Previous research indicates that when automation has near perfect reliability, the human-automation team performs better than if the automation has degraded reliability (de Visser & Parasuraman, 2011; Dixon & Wickens, 2006; Dixon et al., 2006; Endsley, 2015; Wickens & Dixon, 2007). Unfortunately, not all automation is perfectly reliable. Most users of automation have experienced unreliable automation at some point. Human-automation team performance is limited and significantly reduced when automation is degraded; unreliable automation can actually harm performance relative to the human working without automation (Sheridan, 1984). However, the actual reliability of automation is only one of many aspects that affect performance.

The actual reliability of automation can differ from the perceived reliability of that automation. The perceived reliability of an automated system can have a larger effect on team performance than the actual reliability (Dzindolet, Peterson, Pomranky, Pierce, & Beck, 2003). When users encounter information or actions that are in dissonance with previous experiences or knowledge, the user is likely to remember the event. The inconsistencies between expectation and reality distort the user's memory to such an extent that the user may struggle to accurately gauge the system's actual reliability. Therefore, the perceived reliability of automation may be a better metric for understanding performance and user trust in a system than actual reliability (Dzindolet et al., 2003).

The majority of research addressing the performance and user trust in the system has examined the use of automation reliability when the automation acts as an aid to the user (de Visser & Parasuraman, 2011; Dixon et al., 2006; Sheridan, 1984). However, there is a lack of depth in the literature addressing the effects of reduced reliability of an autonomous agent which acts as a teammate rather than an aid. An agent is an automation situated in some environment, and that is capable of autonomous action in this environment in order to meet its design objectives (Russell & Norvig, 2010). The focus of this research is on human-agent teaming rather than human-automation teaming or teaming environments where the automation works as just a decision aid to the user.

For an autonomous agent to be considered a teammate, several conditions must be met. The agent must work together with the human to achieve desired goals, demonstrate predictability in its actions, maintain a common ground with the human user, and be directable (Klein et al., 2004). However, it is important to note that an agent is a teammate, and not a tool, when the agent is capable of performing the task at hand with complete autonomy.

One aspect of successful teams is trust between teammates. Trust in automation is defined as the human user's confidence that the automated system will help the user achieve his or her goals in environments of uncertainty and vulnerability (Ross et al., 2008). Research has shown that in most environments, performance is higher when the user trusts the automation to perform its role (Endsley, 2015; Ho et al., 2005; Wickens & Dixon, 2007).

User trust varies between individuals and can be affected by demographic factors; including culture, age, gender, and personality (Hoff & Bashir, 2015). Culture plays a pivotal role in how people trust. Trust varies across countries, races, religions, and socio-economic status. However, little research has been conducted on how culture affects trust in automation; further research is required to garner a full understanding of how culture plays a role (Wickens & Dixon, 2007). In terms of age, users above the age of 60 tend to trust and rely on automation and decision aids more than younger users (Ho et al., 2005; Wickens & Dixon, 2007). Gender is believed to play a role in how users trust, but there is a lack of investigation concentrated solely on trust with autonomous agents. Previous research has found that in human-human trust relationships, women are more trusting and are likely to regain trust after a trust violation than men (Bonein & Serra, 2009; Buchan, Croson, & Solnick, 2008; Chaudhuri, Paichayontvijit, & Shen, 2013; Haselhuhn, Kennedy, Kray, Van Zant, & Schweitzer, 2015). However, additional research has shown that in human-human trust relationships, the environment can play a role in determining which gender is more trusting. Researchers have found that in a gaming environment, male participants tended to be more trusting than women (Buchan et al., 2008). Personality and personality traits such as neuroticism can negatively impact trust in automation which can, in turn, impact performance (Wickens & Dixon, 2007). The implications of autonomous agent reliability and user trust in autonomous agents should be considered when designing new systems in order to achieve high levels of system performance. However, more research is needed to understand what drives user trust in autonomous agents, especially when the autonomous agent acts as a teammate.

Purpose

The purpose of this research is to identify how the relationship between demographic factors--such as age, gender, technology experience, gaming experience, propensity to trust in other humans, and propensity to trust automation-- relate to the trust a user has in an autonomous agent. As previously discussed, demographic factors and automation reliability play a role in how a human user trusts and interacts with automation. In a teaming environment, the interaction between humans and agents is likely to affect overall team performance. By identifying which factors predict human trust in autonomous agents, designers of autonomous agents will be able to create agents that accommodate many user populations to maintain high levels of performance. It is hypothesized that the factors previously discussed will play a role in the human-agent trust relationship. Finally, it should be noted that this research is exploratory in nature; thus, further research is warranted if the findings contained herein are promising.

Application Environment

The system selected for this research was the tablet computer game *Space Navigator*, a game similar to Harbormaster and Flight Control (Bindewald et al., 2014). Fig. 10 depicts the game environment with an annotated screen capture from *Space Navigator*, which illustrates the elements of the game. The game consists of activities that are completed by the human, the autonomous agent, or both. The game contains four stationary planets present on the screen. Each planet is one of four colors: red, green, blue, or yellow. Spaceships are randomly generated on the edges of the screen at an interval of one spaceship every two seconds. This provides a constant task load across all games. The spaceships move from the allotted generation point on a random trajectory until acted upon by the agent or user. Spaceships continue to appear until an allotted time of four minutes is over. Each spaceship is red, green, blue, or yellow. The player must direct each spaceship to the planet of corresponding color (e.g. red spaceship to the red planet) by drawing a trajectory line on the game touch screen using his or her finger. The spaceship then follows this drawn trajectory route at a constant rate. If desired, trajectories may be re-drawn; this is often done to avoid a collision or account for dynamic changes in the environment (i.e. the appearance of bonuses).

Points are earned when a ship successfully reaches its destination planet or traverses any of a number of small bonuses that appear at random locations throughout the play area. Upon reaching its destination planet, a spaceship disappears from the screen. If the path of a given spaceship crosses over one of these bonuses, it is `picked up' and a point bonus is given. Points are lost when spaceships collide or navigate “no-fly zones’. When spaceships collide, points are lost and each spaceship involved in the collision is lost. The player loses points for allowing spaceships to traverse `no-fly zones’, which were located in fixed locations for the current experiment. In the human-in-the-loop experiment, the game features 100% reliable straight-line automation as well as varying rates of reliability (95%, 90%, 80%, 70%). In this environment, reliability refers to the ability of the autonomous agent to draw trajectories from the spaceships to the corresponding planet (versus drawing to the incorrect planet). The straight-line automation draws straight lines from the spaceships to the planets immediately after the ships spawn. The environment allows for the automation’s and the human’s actions to affect each other. The environment also creates the opportunity for the human-agent team performance to be better than that of the human or automation alone. It is important to note that the automation does not feature collision avoidance, thus enabling the automation to serve as a team member rather than a perfect solution capable of replacing the human player. Because the automation does not feature collision avoidance, human involvement results in higher performance than if the autonomous agent were to operate on its own.

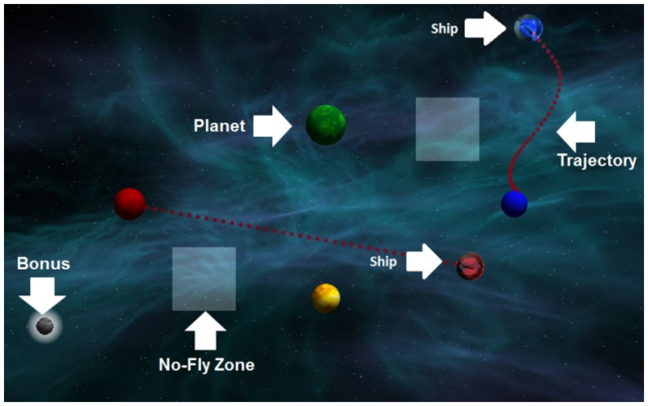


Figure 16. Space Navigator Environment

Method

Experimental Design

The human-in-the-loop experiment included 48 participants with a mean age of 22.6 years and a range of 18 to 38 years. A total of 18 males and 30 females participated. The experimental procedure consisted of a within-subjects design in which each participant completed 12 four-minute games of Space Navigator and 4 four-minute training sessions. The first four games were used as participant training sessions to familiarize the participants with playing the game and agent interaction and featured 100% reliability. Following the training, participants completed six experimental blocks. Each block consisted of two games of the same reliability level. The first block consisted of game play with a 100% reliable agent. After the first block, the user experienced four blocks in a randomized order of each reduced reliability rate until all reliability rates were experienced. The user then experienced a final block with two 100% reliability games. The five different reliability rates used in this experimental procedure were: 100%, 95%, 90%, 80%, and 70%.Therefore, the usable data consist of 48 participants' data or 576 data points. Due to the blocking scheme used in the data collection, the mean for each block was used; yielding 288 total data points (48 participants by 6 blocks). All of the instruments used for data collection to include the surveys, questionnaires, and overall experiment overview can be seen in Appendix 3.

Predictor Variables

Fifteen variables were selected as potential predictors of trust. The selection of these variables stem from previous research conducted on trust (Hoff & Bashir, 2015; McCarley, Wiegmann, Wickens, & Kramer, 2003). These variables include the reliability of the agent, the total weighted NASA-TLX workload score which consists of (Temporal demand, Mental demand, Physical demand, Effort, Frustration, and Performance), the average propensity to trust automation gathered from 9 questions about trust, the average propensity to trust humans gathered from 9 questions about trust, the reliability of the autonomous agent, experience with technology (laptop computers, tablets, and desktops), experience with video games, age, gender, and highest education level (high school graduate, some college, and college graduate). The experiment gathered additional data; however, a great deal of this data had little to no variability. For example, data were collected on the use of smartphones, however only one out of 48 participants did not use a smartphone daily. Other variables omitted due to a lack of variability include handedness, Post-Graduate education, and users that did not complete high school. Refer to Appendix 3 to see the General Trust Questionnaire that was used in the experiment. The descriptive statistics of the variables can be seen in Table 13. Descriptive Statistics All categorical variables were coded using dummy variables.

Table 13. Descriptive Statistics for Predictor Variables

Variable	Mean	Standard Deviation	Frequency	Variable Type
Perceived Trust (DV)	53.32	24.527		Mediating
Reliability of Agent	89.17	10.980		Continuous
Average Trust in Automation	4.99	.774		Continuous
Average Trust in Humans	4.97	.688		Continuous
Total Workload	58.77	17.781		Continuous
Age	22.60	4.711		Continuous
Experience with Video Games	4.67	1.562		Continuous
Experience with Laptops	1.38	.929		Continuous
Experience with Tablets	4.29	1.661		Continuous
Experience with Desktops	3.98	1.481		Continuous
Experience with Gaming Consoles	4.83	1.422		Continuous
Male			18	Categorical
Female			30	Categorical
High School Grad			4	Categorical
Some College			39	Categorical
College Grad			4	Categorical

Dependent Variable

The dependent variable of interest was the user-rated reliability of the autonomous agent. The user-rated reliability (perceived trust) is used as a proxy for user trust in the autonomous agent. It is acknowledged that perceived reliability is only one aspect of user trust (Hoff & Bashir, 2015; McCarley, Wiegmann, Wickens, & Kramer, 2003). However, using perceived reliability as a metric to quantify trust has been shown to be an effective means with which to measure how much a user will trust in an automated decision aid (Ross et al., 2008). This research uses this same metric with an autonomous agent rather than just an automated decision aid.

After the data were collected from the human-in-the-loop experiment, a multiple, stepwise, linear regression was applied to examine the data, using the Statistical Package for the Social Sciences (SPSS). All assumptions for the multiple linear regression were met: The data used had linearity for all X and Y relationships, independence of error, homoscedasticity, and normality as determined using the Breusch-Pagan Test, analysis of residuals, and Normal Q-Q plots. Finally, assessments were made to determine if the predictor variables had multicollinearity. Using SPSS to examine the correlations and collinearity statistics, it was determined that there were no correlation issues determined using a variance inflation factor with a tolerance of 0.20.

Results

While perceived reliability is an effective means with which to measure trust, the participants perceived the agent to be far less reliable than it actually was, as seen in Figure 17.

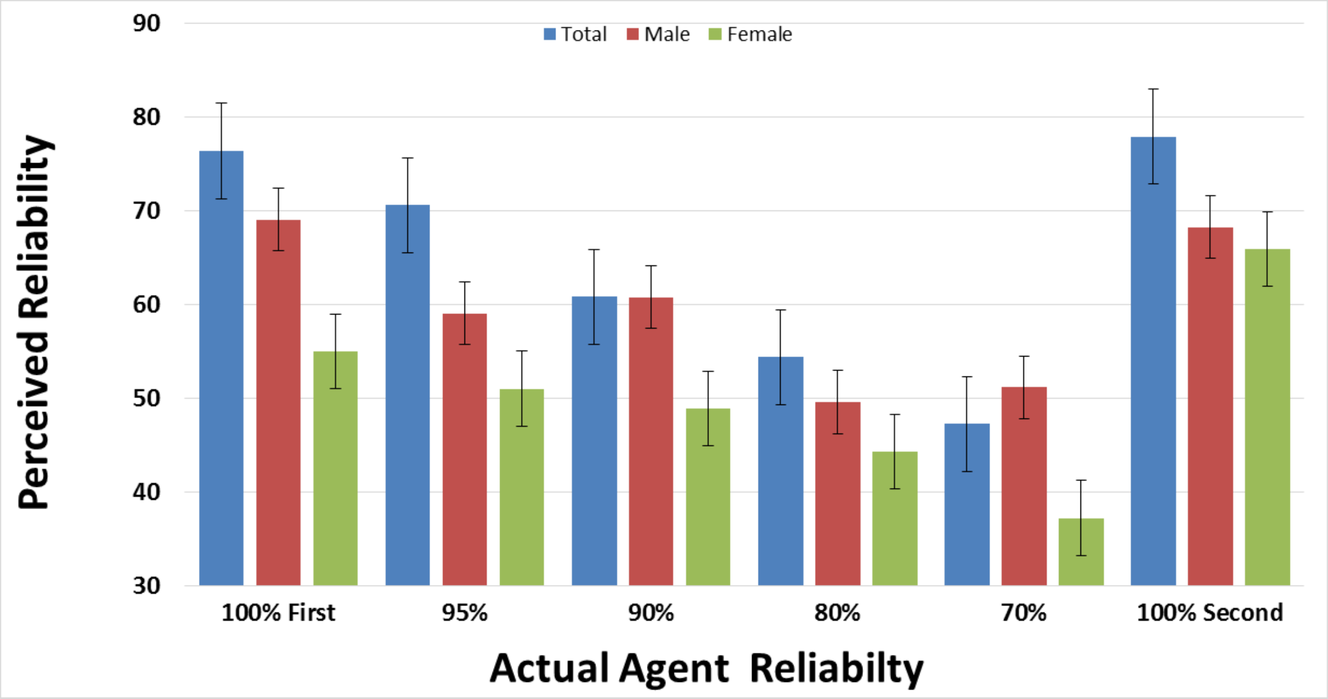


Figure 17. Perceived Reliability vs Actual Agent Reliability

The stepwise regression consisted of 4 models using the predictor variables previously described. Table 14 provides the model output and Table 15 provides coefficient data for each of the four factors used in the final model. This table depicts the coefficient value for all factors, standard error, t-value, as well as the significance of all the factors. The final model included 4 of the 15 predictor variables under examination. The 4 predictor variables were the Total Workload, Reliability of the Autonomous Agent, Gender of the User, and College Graduate. These variables provided a significant model.

The multiple linear regression was calculated to predict user reliability rating (trust) based on several predictor variables. A significant regression equation was found which is shown in equation (1) in which Ŷ is the perceived reliability (trust), W is Total Workload, R is Reliability of the Autonomous Agent, G is Gender, and E is Education Level College Graduate.

$$\hat{Y} = 41.292 - .500W + .534R - 11.59431G + 12.315E \tag{1}$$

This model was found to be significant: F(6, 283)=26.504, p<.000, R^2 of 0.273. A possible explanation for the R2 value comes from the lack of diversity in the participant pool. From this equation we seek, that the participants' perceived reliability rate decreased 0.500 percentage points for each point increase in Total Workload, the perceived reliability rate increased 0.534 percentage points for each percentage point increase in Autonomous Agent Reliability, females had an 11.59 percentage point lower perceived reliability than males, and perceived reliability rate increased 12.32 percentage points for users with College Graduate Education versus those with less than a College Graduate Education or the one participant that had post-graduate education. These results indicate that more is needed to further explore some of the variables that were not found to be significant. Previous research has indicated that individual biases and propensities toward automation play a role in how users trust automation (Parasuraman & Manzey, 2010). However, this research found that propensity toward trust did not play a significant role. This could be explained in part to a potential difference in population samples or the means with which this information was collected. Additionally, previous research has indicated that age plays a significant role in how users interact with decision aids (Ho et al., 2005). However for this research, the age of participants ranged from 18 to 39. The average age was 22.6 was a standard deviation of 4.75 years. However, 30 of the participants were between the ages of 19 and 21. Due to the majority of participants (30/48) being between the ages of 19-21 no statistical significance could be found between age and perceived trust

Table 14. Multiple Step-wise Linear Regression Model R-Values

Model Iteration	R	R-Squared	Adjusted R-Squared	Std Error
1	0.386	0.149	0.146	22.666
2	0.454	0.206	0.201	21.929
3	0.503	0.253	0.246	21.303
4	0.522	0.273	0.262	21.066

Table 15 Coefficient Data for Final Regression Model

Coefficient	Coefficient Value	Std. Error	t	Significance
Constant	41.292	12.079	3.419	.001
Total Workload	-.500	.072	-6.971	.000
Reliability of the Agent	.534	.116	4.624	.000
Female User	-11.594	2.576	-4.500	.000
College Graduate	12.315	4.524	2.722	.007

Discussion and Conclusion

The purpose of this research is to explore relationships between demographic factors such as age, gender, technology experience, gaming experience, propensity to trust in other humans, and propensity to trust in automation relate to and the trust a user has in an autonomous agent. Trust has been identified as a key component to human-team performance. This research provides insight into human characteristics which drive trust in human-automation teaming environments. This research hypothesized that all of these factors would play a role in the relationship with perceived reliability (trust). However, after running a multiple linear regression, it was found that only 4 of the 15 predictor variables were significant. Those variables are the Total Workload, education level, gender, and reliability of the autonomous agent.

In the context of Space Navigator, it was found that higher levels of subjective workload (NASA-TLX) had a negative impact on the trust relationship. It should be noted that this relationship is independent of the agent reliability as shown by the testing for multicollinearity. There was not a significant correlation between the actual agent reliability and workload. Furthermore, the effects of actual agent reliability are also captured separately in the model. At low levels of workload, the user's trust (perceived reliability of the agent) was very high. However, when the users reported high levels of workload, their trust (perceived reliability) was lower. For every one point increase in workload, the users had a 0.500 percentage point decrease in perceived reliability (trust). Although this value appears to be small, the range of workload values in the data is very large (12.98 to 98.00).

As previously discussed, agents with reduced reliability tend to decrease user trust—as seen in this experiment as well. For every 1 percentage point increase in reliability, the users reported 0.534 percentage points greater perceived reliability.

Additionally, these results demonstrated the relationship between gender and trust—a relationship that was previously only fully examined in human-human relationships. The findings reveal that female participants tended to have less trust in the agent than the male participants. As previously discussed, females tend to have higher levels of interpersonal trust, but trust is highly dependent on context, with little research specifically examining gender trust differences with an autonomous agent. However, these findings of males having higher trust in the agent are consistent with findings that males have higher trust in gaming contexts.

Finally, participants with College Graduate education had higher perceived reliability (trust). Although further research is needed to explore why more highly educated users have higher levels of perceived trust in automated agents, it is possible that throughout the course of a college education the users have been exposed to the benefits of autonomous agents through course material.

The four factors that were found significant in perceived reliability (trust) were also significantly correlated to the human-agent team performance (score). Table 16. shows the correlations between the four significant factors and the score of the human-agent team. The Pearson Correlation and 2-tailed significance test was used to find that each factor demonstrated a significant relationship with the human-agent score. The correlations found between the significant factors and score highlight the importance of considering more than just the agent's reliability. The workload, education level, and gender of the human play a significant role in the human-agent team. In the environment Space Navigator, being a female or being a college grade had a negative correlation with score. This finding suggests that changes should be made to the agent's design to address the characteristics of female and college grad users. Potential changes include training to allow female and college graduates to better calibrate their trust in the system. Additionally, alarms could be placed in the system design so as to alert the user to perform additional cross-checks to ensure the automated agent is performaing as it should. Additionally, there was a negative correlation between Total Workload of the user and the human-agent team performance. Previous research has found that high levels of workload decrease overall performance (Dixon et al., 2006; Maynard & Rantanen, 2005; Singh et al., 2009). Finally, as expected, higher levels of agent reliability were positively correlated with human-agent performance. Surprisingly, all of the predictor variables identified by the regression were nearly equal in terms of correlation to score of the human-agent team. It was originally believed that the agent reliability would have the strongest correlation to score, but this research has found that in the Space Navigator environment, agent reliability is only one of four nearly equal factors that play a role in the performance of the team.

Table 16. Correlation Table

Variable	Correlation with Score
Score	1
Reliability of Agent	0.283
Female	-0.269
College Grad	-.0263
Total Workload	-0.298

Future Work

Further research is required to understand what drives the aforementioned relationships. Future work could focus on the disparity in trust between female and male users as well as what design considerations should be made to accommodate various user populations. Additionally, further research is needed to expand the findings of this research to different application environments as well as further work to expand the understanding of the relationship between trust and human-agent team performance. Finally, further research using a much larger sample size with a more diverse population could potentially increase the R value associated with the regression.

V. Analysis of Archetypes on Human-Agent Performance

Abstract

Human behavioral patterns play a significant role in how humans interact in teaming environments. This research examines behavioral patterns of automation compliance and classifies them as archetypes. The research focuses on compliant behavior in humans when interacting with an automated agent that acts as a teammate, while performing an air traffic control style task. Based upon behavior, participants are classified into the archetypes of reduced compliance, high compliance, and flexible compliance using k-means clustering. The data were clustered using 488 data points and three clusters. It was found that 13 participants demonstrated reduced compliance, 27 high compliance, and 8 flexible compliance. This research than examined if demographic factors such as age, gender, education, and video game experience played a role in predicting the archetype of a user. It was found that none of the factors played a significant role in determining archetype.

Introduction

Classifying people into groups based on certain characteristics is not a new concept. More specifically, the idea of classifying people based on their behavior, has been part of human society for a great period of time. Many businesses and industries benefit from being able to group employees into categories which describe their performance or type of work. However, there is not a great deal of research that focuses on a reproducible means with which to classify people based on their interactions with automation or automated agents. The research that most closely resembles the work required to classify people according to their interactions with automation or automated agents comes from the world of video games. A large amount of research has surfaced in the last decade that focuses on creating profiles of video game players. These profiles of users are being used extensively by game developers to create state-of-the-art video games that are increasingly adaptive and dynamic (Bakkes, Spronck, & van Lankveld, 2012). The focus of these profiles is on creating artificial intelligence that is entertaining to the human player (Bakkes et al., 2012). The artificial intelligence is created by modeling human strategies, tactics, and actions so as to create an environment that is as realistic and dynamic as possible. Before the modeling can occur, a profile of the human players must be created first. One of the more common approaches to creating human player profiles is the stereotype approach (Yannakakis, Spronck, Loiacono, & André, 2013). The stereotyping approach creates subgroups with key characteristics of a population and then assigns players to subgroups according to the previously defined key characteristics (Yannakakis et al., 2013). In some instances, the key characteristics are actually empirical models based on the 'Big 5' aspects of personality (Cowley, 2009; Goldberg, 1993; Spronck, Balemans, & Lankveld, 2012; Yannakakis et al., 2013).

This research seeks to understand how behavior is impacted by more than the 'Big 5' aspects of personality. Rather than generating stereotypes of users, this research examines archetypes or patterns of behavior. Although a sizeable amount of research has been conducted examining player profiles, there has been little research conducted which examines how archetypes can be created to help understand how users interact with automated agents. Specifically, little to no research exists that examines how key demographic factors affect human interaction with automated agents in teaming environments. However, player profiling in video games is a growing field of research and will be used as a basis with which to explore the concept of archetypes. The concept of archetypes is similar to player profiling in that both player profiling and archetypes classify people based on certain characteristics. This research focuses on using human behavioral patterns to classify people into archetypes.

The current state of literature regarding player profiling is quite new. There is only a small amount of research on real-time game analysis and player profiling at the lowest level of abstraction. The literature also lacks depth in individual game elements and their interactions. (Cowley, 2009). Cowley states that many of the models used in player profiling are too simplistic in nature and fail to provide descriptive richness. Additionally, many of the profiles only fit expert players, not beginners (Cowley, 2009). Perhaps one of the greatest disadvantages player profiling experiences is player specificity. Each player profile that is created begins very specific to the player that was studied (Bakkes et al., 2012). Creating the profile for the player is also generally very time intensive and requires large amounts of player observation in order to achieve the profile with the greatest amount of accuracy (Bakkes et al., 2012). Bakkes et al. states that the best player profiling stems from well-designed models that examine both the player characteristics such as playing style (Bakkes et al., 2012).

Although player modeling suffers from several limitations, there are several key advantages to creating player profiles. The first advantage is the ability for game engineers to help create a game environment that is adjusted to an appropriate level of difficulty for each player profile (Cowley, 2009). This adjustment of difficulty creates a game experience that many different players can enjoy yet at the same time be challenged and interested in the game. Similarly, in the world of industry or the military, providing environments in which employees are both challenged and involved, is a challenge that may be overcome with archetype analysis. Another advantage lies in smoothing the learning curve (Cowley, 2009). With an understanding of player profiles, the game engineers can cater the game training to adjust for various player profiles. This will in turn reduce the amount of time any given type of player takes to learn how to play the game or become proficient in the games dynamics. The smoothing of the learning curve is a challenge that many new employees face regardless of job type. Both industry and military applications can benefit from being able to reduce the amount of time it takes to train new members. Finally, using player profiling, the game experience can be alter to enhance the experience for any given player, regardless of profile type (Cowley, 2009). This concept of improving the game experience is very closely related to the challenge employers face when providing the best possible work environment for their employees to succeed.

The variables of interest for this research were the reliability of the automated agent and the demographic data. In the *Space Navigator* environment, unreliability is defined as the automated agent drawing a route from a ship to an incorrect planet. Unreliability does not account for collisions that happen along the trajectory. The reliability of an automated agent, plays a vital role in the human-agent team (Parasuraman & Miller, 2004; Ross et al., 2008; Rovira & Parasuraman, 2010; Wickens & Dixon, 2007). The demographic data of interest for this research included gender, age, video game use, and education level of the users. These factors were selected due to previous research indicting their possible significance in impacting human behavior in video games (Bakkes et al., 2012; Yannakakis et al., 2013). For a complete listing of all of the demographic data collected, reference Appendix 3.

One of the effects of unreliable automation surfaces in the user compliance with the automation or agent. The dependent variable of interest is the number of redraws performed by the human. Redraws are defined as a change in an agent drawn trajectory made by the participant. This metric acted as a means with which to measure the compliance rate of each participant. Compliance relates to the human operator's response when an alarm or signal sounds, whether that alarm or signal is true or false (Chen & Barnes, 2014; Chen & Joyner, 2006; Dixon & Wickens, 2006; Dixon et al., 2006). A compliant human operator is an operator who rapidly switches his or her attention from concurrent activities to the alarm domain (Dixon & Wickens, 2006). The operator may then immediately initiate an alarm-appropriate response, such redrawing a trajectory (Dixon & Wickens, 2006). Research has shown that as the rate of false alarms increases, the operators experience a reduction in compliance (Chen & Joyner, 2006; Dixon & Wickens, 2006; Dixon et al., 2006). This compliance reduction can result in longer response times to automation alerts or alarms (Dixon et al., 2006).

Purpose

The purpose of this research is to classify humans working with an automated agent into archetypes. This research then seeks to identify how the archetypes affect human-agent team performance. Additionally, this research explores the relationships between archetypes and demographic factors such as age, gender, and experience with various forms of video games. As previously discussed, archetypes or profiling users based on behavior, has strong links to demographic factors. It is hypothesized that using an archetype approach will allow researchers to better understand how patterns of behavior are related to both performance and demographic factors.

Application Environment

The application environment that was utilized in this research was the tablet-based game, *Space Navigator*. This environment is unique because it features the ability for the human and the agent to interact and work together as a team rather than the human and agent working independently of each other. This game mimics an air traffic control situation and is similar to the custom route-creation games Harbormaster and Flight Control. The premise of the game consists of drawing trajectories from spaceships to planets. The activities in the game are completed by the human, agent, or both. The game environment contains four static planets on the screen. These planets are uniquely colored: red, green, blue, and yellow. Space ships are generated on the sides of the screen and proceed to move through the game environment. Each spaceship is red, green, blue, or yellow and is generated at a rate of one spaceship every two seconds. The spaceships are generated until the allotted time of four minutes is over. The objective of the game is to have each spaceship reach the corresponding colored planet (e.g. red spaceship to red planet) by drawing trajectories from the spaceship to the planet. The human user accomplishes this action by dragging his/her finger along the touch screen, starting at the spaceship and creating a path for the spaceship to follow. The spaceship will then follow the trajectory at a constant rate until the planet is reach, a collision with another spaceship occurs, or the spaceship travels off screen. The trajectories may be rerouted. This is often done to avoid collisions or account for dynamic changes in the environment such as the generation of bonus orbs. Points are rewarded to the player when a spaceship successfully reaches the destination planet or a bonus orb is collected. If the spaceship reaches the correct planet, the spaceship will disappear immediately. If the spaceship collides with another spaceship, points are lost and each spaceship that was involved in the collisions immediately disappears. If the spaceship collects one of the bonus orbs (small orbs that are generated randomly throughout the game) then the user receives a point bonus. The player can also lose points for crossing ‘no-fly zones’ which are shaded boxes when the planets.

This game features an agent that draws straight-line trajectories from spaceships to the planets. The straight-line automation draws straight lines from the spaceships to the planets immediately after the ships spawn. The agent does not feature a collision avoidance aspect, thus enabling the agent to serve as a team member and not a perfect solution capable of replacing the human player. Human involvement allows for better performance than if the agent works alone. Figure 18. Application Environment for Archetype Analysis depicts the game environment with an annotated screen capture from Space Navigator, which illustrates the elements of the game described herein.

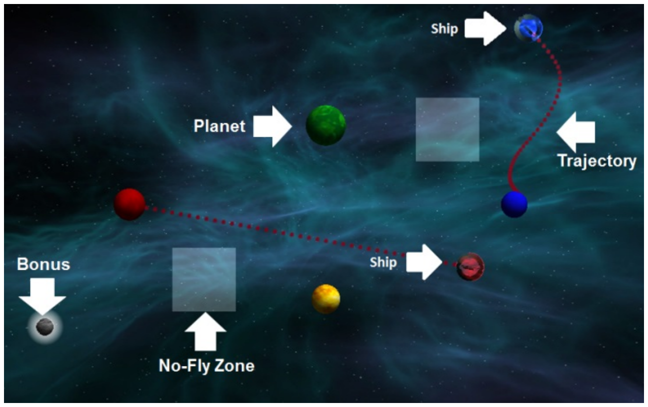


Figure 18. Application Environment for Archetype Analysis

Methodology

The methodology of this research consisted of collecting data using a human-in-the-loop experiment. The data collected was then analysed using a K-means clustering algorithm to detect the presence of clusters in the data. The identified clusters were then used to discover patterns in the identified clusters with the demographic data collected from the participant pool.

Human-in-the-Loop Experiment

The human-in-the-loop experiment included 48 participants with a mean age of 22.6 years and a range of 18 to 38 years. A total of 18 males and 30 females participated. The experimental procedure consisted of a within-subjects design in which each participant completed 12 four-minute games of *Space Navigator* and 4 four-minute training sessions. The first four games were used as participant training sessions to familiarize the participants with playing the game and agent interaction and featured 100% reliability. Following the training, participants completed six experimental blocks. Each block consisted of two games of the same reliability level. The first block consisted of game play with a 100% reliable agent. After the first block, the user experienced four blocks in a randomized order of each reduced reliability rate until all reliability rates were experienced. The user then experienced a final block with two 100% reliability games. Five different reliability rates were used in this experimental procedure were: 100%, 95%, 90%, 80%, and 70%. Therefore, the data consisted of performance and compliance data for 576 games (48 participants by 12 games). Due to the blocking scheme used in the data collection, the mean compliance rate for each block was used; yielding 288 total data points (48 participants by 6 blocks).   All of the instruments used for data collection to include the surveys, questionnaires, and overall experiment overview can be seen in Appendix 3.

Identifying Archetypes: K-Means Clustering

The purpose of this research is to create classifications of patterns of behavior. One of the metrics that was collected during participant testing was the number of redraws each participant performed during each game. As previously mentioned, redraws are defined as a change in an agent drawn trajectory made by the participant. This metric acted as a means with which to measure the compliance rate of each participant. Compliance is a pattern of behavior that can be applied to various applications; therefore, it is necessary to identity these patterns in a basic environment such as *Space Navigator*. In order to identify these patterns, the statistical package “R” was used in applying the K-Means clustering algorithm. “R” is a programming language and environment used for calculating statistics and associated statiscal graphs. The software is open-source and encompasses a wide variety of statistical and graphical techniques (R Foundation). “R” contains a K-Means clustering function. This research utilizes a 2-dimensional clustering approach. The dimensions which were used in this approach were the number of routes redrawn by each participant and the range in the number of redraws for each participant. The number of redraws (compliance) was averaged for each participant, creating 48 data points--one for each participant in the study. The range was found by subtracting the highest number of redraws from the lowest number of redraws for each participant across all reliability conditions. The K-Means clustering approach was applied with three clusters, high, flexible, and reduced compliance. The “Flexible Compliance” archetype contained participants that changed their compliance based on the reliability of the autonomous agent, and thus exihibeted larger ranges than the other other two clusters. In order to capture this range in compliance the range of redraws was used in the clustering analysis. It is believed that an individual with a large range of redraws is likely changing their behavior based on the reliability of the agent, hence the term, “Flexible Compliance.” The results section depicts the clustering of the 48 data points which is referred to as the consolidated data set and presents the fit and centers of each cluster.

Analysis of Demographic Factors on Archetype

The three clusters were then analyzed using several demographic factors such as age, education level, gender, and experience with video games. Age was measured continuously in years. Education level was blocked into the following categories: high school graduate, some college, college graduate, some post-graduate education. Gender was categorized as male and female. Experience with videos was blocked into the following groups: no video game use, using video games less than once a week, using video games once a week, using video games 2-3 times a week using video games 4-6 times a week, using video games daily. In order to address difference between the groups, Analysis of Variance (ANOVA) was used. Using ANOVA allows for statistical differences within the various groups to be detected. However, ANOVA does not allow for conclusive statistical proof of where the differences are between the groups. Therefore, a Chi-Squared Test was used to check for differences between groups. Additionally, descriptive statistics were used to address the types of demographic factors that composed each archetype.

Results: Identifying Archetypes

The clustering of data consisted of using a K-Means algorithm on 48 data points. The data points are the mean number of redraws (compliance) for each participant across 12 games. Three clusters were input to the K-Means algorithm. The clusters consisted of a high, reduced, and flexible archetypes. A visual depiction of the clustering results can be seen in Figure 19.

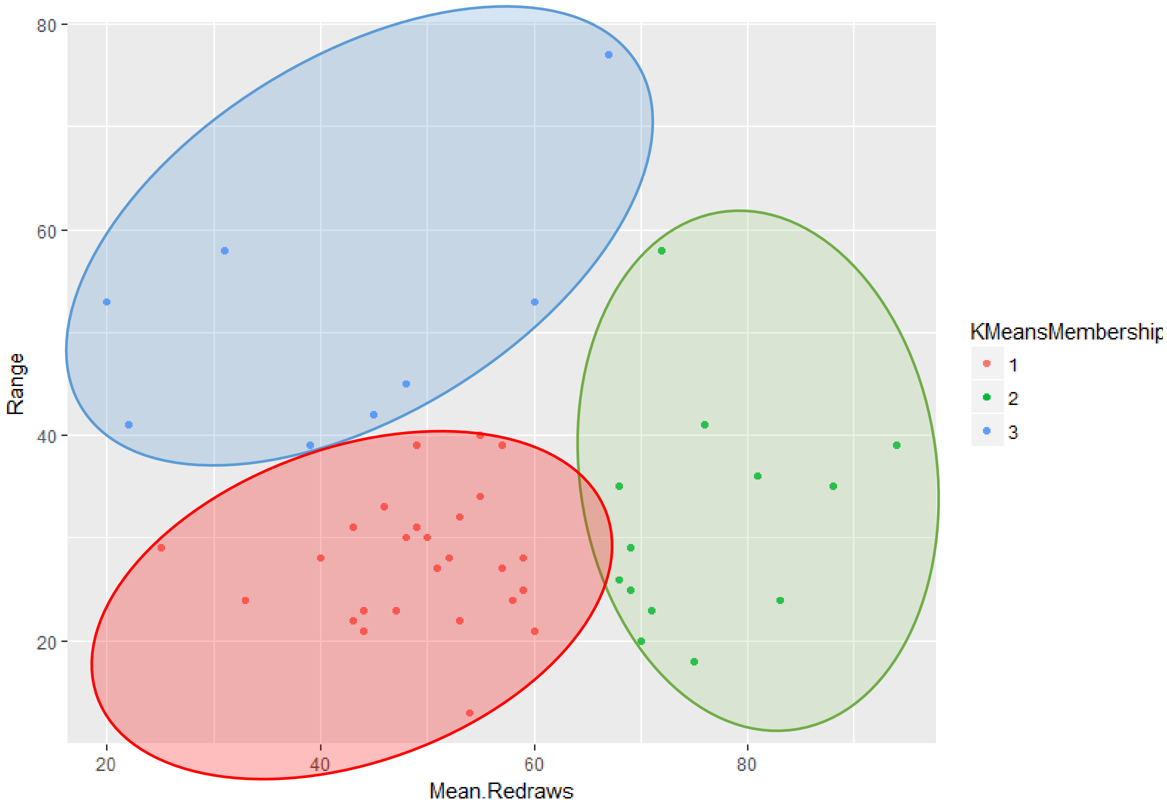


Figure 19. Clustering on Mean Redraws and Range

The results of the clustering can be seen in Table 17. The high compliance cluster center is at 75.69 mean redraws and a mean range of 31.46. The flexible compliance cluster had a cluster center at 41.50 mean redraws with a mean range of 51.00. The reduced compliance cluster had a cluster center at 49.70 mean redraws and a mean range of 27.70. Twenty-seven participants were classified as high compliant users, 8 participants as flexible compliant users and 13 participants as reduced compliant users. The highly compliant users redrew on average 26 more routes than the reduced compliant users and 34 more routes than the flexible users. Additionally, as expected, the flexible compliant users had a much larger mean range than the high and reduced compliant users. Additionally, the model had a fit of 57.8% meaning that the 3 cluster model was able to explain 57.8% of the variance that occurred in the data set. The betweenness is the weighted sum of squares between two means, to measure how well cluster centers are separated, which for this set of clusters was 11147.00 which indicates the cluster centers are significantly separated.

Table 17: Consolidated Data Set Clustering Statistics

3 Cluster Model	48 Data points
Reduced Compliance Cluster Mean Redraws	49.70
Reduced Compliance Cluster Mean Range	27.70
Flexible Compliance Cluster Mean Redraws	41.50
Flexible Compliance Cluster Mean Range	51.00
High Compliance Mean Redraws	75.69
High Compliance Mean Range	31.45
High Compliance Cluster Size	27
Flexible Compliance Cluster Size	8
Reduced Compliance Cluster Size	13
Fit (Withinness)	57.8%
Betweenness	11147.00

Results: Analysis of Demographic Factors on Archetype

This research also investigated the relationships between age, video game use, education level, and gender. Due to the limited variability in the age, video game usage, and education of the participant pool, no statistical significant differences could be found between these factors and archetype cluster.

Age. The age of participants ranged from 18 to 39. The average age was 22.6 was a standard deviation of 4.75 years. However, 30 of the participants were between the ages of 19 and 21. Due to the majority of participants (30/48) being between the ages of 19-21 no statistical difference could be found among archetype cluster.

Video Game Use. A large proportion of the participant pool did not contain video game users. Twenty participants reported no video game use, 12 participants reported using video games less than once a week, 6 participants reported using video games once a week, 3 participants reported using video games 2-3 times a week, 4 participants reported using video games 4-6 times a week, and 3 participants reporting using video games daily. It is believed that this sample represents college-aged students, thus no strong conclusions can be drawn connecting video game use and archetype.

Education. Over 80% (39 of 48) of the participants reported having some college education, with the remainder being divided among high-school graduates, college graduates, and some post graduate education. Due to the lack of variability no statistic difference could be found among archetype cluster.

Gender. Using a Chi-squared test to test for differences it was found that there was not a significant difference between male and female users. The Chi-square value for gender was .536. Reference Appendix 5 for the Chi-Squared test. Table 19: Gender Distribution for Archetypes depicts the distribution of males and females across the three patterns of behavior. Of all female participants, 56.67% were classified as high compliance users whereas 55.56% of males were classified as high compliance users. Additionally, of the high compliance users, 62.96% were female.

Table 18: Gender Distribution for Archetypes

Gender	Reduced Compliance	Flexible Compliance	High Compliance
Female	9	4	17
Male	4	4	10

Results: Analysis of Archetype on Performance

This research examined the relationship between archetype and performance. It was found using ANOVA that there was not a statisitical difference in terms of performance between the identified archetypes. The ANOVA performed used the Consolidated Data Set. The Consolidated Data Set featured three archetypes: reduced, flexible, and high. Although the reduced compliance archetype group was the highest performing archetype with 125.82 points more than the flexible compliance archetype, and 199.36 points more than the high compliant group, these differences were not statistically significant. The ANOVA can be seen in Appendix 6.

Discussion

Previous research has examined player profiles for individual players using models that specifically linked player game style to the generated player profile (Bakkes et al., 2012; Cowley, 2009). In contrast to the previous research, this research seeks to generate archetypes for participants based on demographic factors such as age, gender, and experience with various forms of video games. This research is unique in that it has not only created a “flexible” compliance archetype but also specifically addressed player archetypes by examining general demographic factors which can apply to many participants rather than just individual players. Before delving into the relationships between the archetypes and demographic data, the clusters must be discussed.

The clusters were identified using the K-Means algorithm with three cluster centers and 48 data points with two dimensions: mean range of redraws and mean redraws. This clustering provided insight into one pattern of behavior, compliance. The clustering identified three archetypes of compliance: reduced, flexible, and high compliance users. However, the majority of participants were clustered as being highly compliant (27 participants) with only 8 participants being clustered as flexible compliance users and 13 being clustered as reduced compliance users. Although the reduced compliance archetype group was the highest performing archetype with 125.82 points more than the flexible compliance archetype, and 199.36 points more than the high compliant group, these differences were not statistically significant. It is possible that with a more diverse participant pool and more participants, significant difference would become apparent, but further research is required.

It was found that there was no significant difference between male and female users. Although there was no significant statistical difference for gender, age, education, and video game use, this lack of statistical difference could be attributed to the lack of variability in the sample. It is acknowledged that true differences may exist in the population.

Conclusion

This research sought to further research into reproducible means with which to classify people based on their interactions with automation or automated agents. The means with which this research helped further this goal was through the use of archetype analysis. The research started with identifying archetypes of human compliance behavior and then clustered users into the defined archetypes. Using K-Means clustering, three archetypes were identified, reduced, flexible, and high compliance. It was found that majority (56.25%) of participants were classified as highly compliant users. Using a 2-dimensional clustering scheme with 3 archetypes allowed for a model that was able to fit users of the subject pool into three clear groups.

This research also wanted to identify how the identified archetypes affected human-agent team performance as well as explore the relationship between archetypes and demographic factors. After analysis, there were no significant connections demonstrated between age, education, or video game use, the only significant connection between archetypes and demographic factors was that of gender. It was found that female participants were more likely to be highly compliant users than their male counterparts. This finding indicates gender can be indicative of how a user will interact with an automated agent. Although there is not a significant difference between male and female users, previous research suggests there is a difference between how male and female users trust and interact with automated agents (Chaudhuri et al., 2013; Koustanai, Cavallo, Delhomme, & Mas, 2012; Verberne, Ham, & Midden, 2012).

Future Work

Throughout the course of this chapter, the amount of variability has been addressed several times. Due to the lack of variability in the participant pool, it is unclear whether demographic factors such as age, video game use, and education status have a significant role in archetype classification or human behavior. In order to address this concern, future work should extend this study to a wider range of participants. This includes participants across a larger range of ages, video game experience, and education status. Studies have shown that older users of automation tend to interact differently with automation than their younger counterparts (Chen & Barnes, 2014; McCarley et al., 2003). Finally, future work could include the creation of additional archetypes which would create additional archetypes for analysis.

VI. Conclusions and Recommendations

Chapter Overview

The chapter provides an overview of the research motivation concerning human-agent teaming to include the work done on both human-agent team trust and performance. It then reiterates the overall research objectives of this paper. Proceeding the objectives are the three sections which address the investigative questions posed in Chapter 1. The chapter concludes with the impact of this research and recommendations for future research in this field.

Research Motivation

Human-agent teaming plays a pivotal role in many facets of today's world. From home applications to military applications, human-agent teaming is nearly everywhere. Human-agent teaming is focused on a human-user and automated agent working collaboratively, rather than the agent replacing the human. The driving concept behind this teaming is to augment the human's capability (Kaminski, 2012). Automated agents present many benefits to the human user. Four of the major advantages include: reduction in unnecessary labor and manning costs, increased range of operations and capabilities, reduction of time to conduct operations, and increased operational reliability (Endsley, 2015).

However, due to ever-increasing application of automated agent in today's world, the risks of unintended outcomes is increasing. This risk of unintended outcomes stems from the increased complexity of system, greater potential for failure, and security vulnerabilities. The DoD has focused specifically on the impacts of user trust in automated agents and systems. The issue is so important to the DoD that the DoD Autonomy Task Force has called upon the Under Secretary of Defense for Acquisition, Technology, and Logistics to create developmental and operational test and evaluation techniques that focus explicitly on building trust in autonomous systems (Kaminski, 2012). The United States Air Force is also very interested in solving the issues surrounding user trust in automated agents and the resulting impact on human-agent team performance. The Chief Scientist of the Air Force has explicitly stated that building trust in autonomous system is a major hurdle that today’s airman faces (Endsley, 2015).

Research Objective

The focus of this research has been on exploring several aspects of human-agent teaming. First, this research examined how agent characteristics such as reliability impacted the human-agent team performance. This research simulated how performance differed across several levels of reduced agent reliability. The next step of the research consisted of performing a human-in-the-loop experiment in which the assumptions and results of the simulation were tested. The research then examined the relationships between human characteristics such as age, gender, and education, impacted the amount of trust a user had in an automated agent. Finally, the research examined how human behavior was related to human-agent performance as well as the relationship between human behavior (classified into archetypes) and human demographic data.

Summary of Research Gap

The research gap addressed in this research is focused on two topics: trust and user characteristics such as demographic data and behavioral patterns. A great deal of research has been conducting examining user trust in automated agents that function as decision aids. However, there is very little research pertaining to user trust in automated agents that act as teammates rather than just decision aids. The distinction lies within the ability of the agent itself. A decision aid agent merely provides recommendations to the user. These agents can be seen nearly everywhere, Siri is just one example of this type of agent that can found in the pockets of millions of users. An agent that acts as a teammate possesses the ability to carry out actions and make decisions. This type of agent can work independently of the human, if need be.

The current state of research lacks sufficient understanding as to how human characteristics impact trust and human-agent team performance. There is research that speculates the impact of demographic data on how a user trusts an automated agent, but there are few experiments that support the claims when dealing with an agent that works as a teammate rather than just a decision aid. Additionally, there is little research examining the impact of human behavior patterns on human-agent team performance. A large amount of research is emerging that examines human behavior in video game environments which may in turn spur on research directed toward understand human behavior with automated agents.

Research Questions and Answers

This thesis posed three investigative questions in Chapter 1. The three investigative questions are used to decompose the larger research question: how do human and autonomous agent characteristics affect human-agent team performance and user trust in an automated agent? The following section summarizes the findings to address the questions outlined in Chapter 1.

How does automated agent reliability affect human-agent team performance?

The first step in answering this question was creating a simulation on human-agent interaction. The work surrounding this simulation can be seen in Chapter 2 of this thesis. The simulation found that a highly reliable agent (>92% reliable) is required for the team to perform better than the human or agent alone. Much like human-human teams, both members of the team are required to perform adequately so as to not create additional task load on the other teammate, which can lead to decreased performance (Hoc & Lemoine, 1998; Mahfouf et al., 2007; Mitchell, 2009). However, in order to validate the results of the simulation, further research and experimentation was required. The researchers created a human-in-the-loop experiment in which human participants were required to work with an agent that experienced reduced reliability. The experiment found that human users were able to compensate for a portion of the agent's reduction in reliability. The study found that human-agent team performance with a 95% reliable agent was statistically the same as when the agent had 100% reliability. However, at values lower than 95% percent agent reliability, the human-agent team performance suffered. As stated in

Chapter 3, the implications of these findings could affect many aspects of how users and designers interact and design automated agents.

Do archetypes affect human-agent team performance?

In order to analyze the effect of archetypes or human behavior patterns, archetypes had to be identified. As stated in Chapter 5, one of the metrics that was collected during participant testing was the number of redraws each participant performed during each game. Redraws are defined as a change in an agent drawn trajectory made by the participant. This metric acted as a means with which to measure the compliance rate of each participant. The archetypes of high, reduced, and flexible compliance were identified. The participants of the study were then placed into these archetypes according to a cluster analysis. After examining the performance of each archetype, it was determined that the reduced compliance group performed the best with an average score of 6235.19, followed by the flexible group with a score of 6109.38, and the high compliance archetype performing the worst with a mean score of 6035.83. Although these performance scores are different, the variance of the scores was so large that the differences were not deemed statistically different. Therefore, in the *Space Navigator* environment, archetypes existed but did not have a statistical effect on the human-agent team performance. However, it is believed that in other environments, archetypes may play a role in human-agent team performance.

What demographic factors predict perceived trust and trust behaviors?

As seen in Chapter 4, it was found using a stepwise multiple linear regression that workload (NASA-TLX), gender, education level, and the reliability of the autonomous agent impact the perceived reliability or user perceived trust in the system. When the user experienced higher workload, the user had less trust in the autonomous agent. Females trusted the agent less and more educated users trusted the autonomous agent more. Finally, more reliable agents led to higher levels of trust in human users. These findings indicate that demographic factors as well as agent characteristics play a role in how humans interact with an automated agent.

Using the current body of literature regarding predictors for human behavior in video games, several metrics were considered in the analysis of determining which factors may predict archetypes of trust behavior. Of the collected demographic factors from the human-in-the-loop experiment, gender, age, video game use, and education level were selected for analysis due to their respective influence on human behavior in video games as discussed in Chapter 5. The factors were then analyzed to detect a relationship between the archetypes previously created. As stated in Chapter 5, due to the limited age range of the participant pool, no statistical significant differences could be found between age and archetype cluster. It was also found that education level and video game use did not play a significant role in the archetype clustering. All of which stem from a lack of variability in the sample. Finally, although there were no statistical difference found between male and female participants, previous research suggests there may be differences between male and female users (Chaudhuri et al., 2013; Koustanai, Cavallo, Delhomme, & Mas, 2012; Verberne, Ham, & Midden, 2012).

Recommendations for Future Research

As stated in Chapter 1, this research is conducted in a tablet based environment using an automated agent. Due to this construct, the risk to the user is very low, creating an environment that does not replicate the high-stakes pressures of a military operating environment. Further research would benefit by creating a test bed in which the users experience a high-stakes environment. This could be difficult to implement, but the results of such an experiment would likely be more indicative of the real world.

Another area in which future research could continue would be to expand the current human-in-the-loop experiment to include more participants, including non-college students. An increased number of participants and a broader sample would provide an increased ability to identify differences based on demographic data. The increase in participants will thereby increase the accuracy and legitimacy of the research. The research is currently very theoretical and identifies many relationships that need further study.

A final area of further research includes examining different, focused populations. The population used in this research consisted of primarily college aged females at the Ohio State University. This population is not indicative of the military's population (the population of interest for this research's applications). Future research using a military population will enhance interpretations of trust and archetype behaviors in a military context.

Significance of Research

The Air Force has called for further research into what drives trust in automation and what affects human-machine teaming. This thesis explores the characteristics of both autonomous systems and humans and how the characteristics affect trust and human-agent team performance. Through the use of both simulation and experimentation, this research demonstrates that automated agent reliability plays a critical role in the human-agent team. The automated agent's reliability has a measurable effect on the team's performance. This information can be used in further research and design of automated agents. The framework has now been established that solidifies how a reduction in automated agent reliability decreases the overall team performance. Additionally, it was found that the human user is able to compensate for a small degree of agent unreliability. This information should be taken into account when analyzing future systems with automated agents that work as teammates.

The next area in which the research has shown to be significant is in the classification of human behavior. Previously, archetypes have not been considered in the fields of human-machine teaming and human-agent interaction. This research opened the door to understanding the various types of compliant human behavior. A new concept of "flexible" compliance was generated which allowed for a more accurate description and classification of human users' interaction with automated agents. Finally, this research found that the age, gender, and education level of the human user has a relationship with the perceived trust the user has in the agent. Overall, this research has brought greater insight and understanding into what drives user trust and human-agent team performance.

Appendix 1: Extended Literature Review

Introduction

Human-agent teaming is prevalent in nearly every facet of today's military, transportation, industry, and medical fields. These fields deal with many complex, repetitive tasks, which are well-suited to human-agent teaming, thus enabling the human operator to focus his or her attention where it is needed (Hoff & Bashir, 2015). One aspect that plays a critical role in human-agent teaming is the user's trust in the automated agent. Human-agent teaming and trust in automation are two independent, yet highly intertwined topics that encompass many of the challenges today's users of automation face.

This literature review focuses on several topics that pertain to human-agent teaming including: automation, user trust in automation, reliability of automation/agent, compliance, reliance, and human-agent teaming. This examination of literature shows that there is a gap in research pertaining to human-agent team performance, when the agent acts like a teammate but has less than perfect reliability. This review shows that trust in the automated agent is composed of several subtopics such as user compliance and reliance as well as how misuse and disuse of automation may have dire consequences. Overall, this review of literature encompasses a wide spectrum of human-agent teaming benefits, challenges, and background research on the topic.

Automation

Automation is a rather difficult term to define. Many definitions and explanations of automation exist in the literature surrounding the field of human-agent teaming. The struggle in defining and explaining automation stems from a wide variety of system applications. For example, automation can include the most mundane tasks, such as the tasks performed by an alarm clock as well as some of the most intricate and complicated tasks such as the tasks performed by a driver-less car. Traditionally automation has been defined as a system that functions with little to no human operator input with specific actions(Endsley, 2015). However, this definition does not fully encompass all aspects of automation.

Automation also includes any sensing, detecting, information-processing, decision-making, or controlling action that is usually performed by a human, but is now performed by a machine (Moray, Inagaki, & Itoh, 2000). The actions of automation are not limited to cognitive tasks, but also many physical tasks (Ross et al., 2008). Traditionally automation implementation has focused on tasks that humans do not want to perform, cannot accurately perform, or cannot perform as reliably as an automated system (Parasuraman, Sheridan, & Wickens, 2000). The analysis of what automation is and what it is capable of doing is a monumental undertaking. The Department of Defense (DoD) assembled a task force in hopes of capturing what automation is capable of doing and the current limitation of automation.

The Task Force provided the definition of “automation” to be a capability or set of capabilities that acts as an enabler for a system to be automatic or self-governing within programmed boundaries (Kaminski, 2012). This definition is the one that will be used for the remainder of this thesis. However, the Task Force acknowledged that the word “autonomy” has a rather negative connotation in some military leaders’ minds.

Unfortunately, Hollywood has painted the image of autonomy making independent decisions and taking uncontrolled actions. This notion of independent decision making and uncontrolled actions is far from reality. It is imperative to understand that all autonomous systems are supervised by human operators at some level—whether that level is at the software and programming level or the operational level (Kaminski, 2012).

Parasuraman and Riley (1997) make the argument that some instances of automation become so ingrained in everyday activities that the human users forget the presence of the automation. Parasuraman and Riley (1997) stated that automation was the execution of a task by a machine agent that was previously performed by a human operator and, that when the reallocation of this task performance is complete and permanent, the automation is simply viewed as machine operation (Parasuraman & Riley, 1997). Examples of this permanent and complete reallocation of task performance include starter motors in cars, automatic elevators, automatic teller machines (ATMs), and cruise control in cars. A critical review of the previously-listed machines highlighted the notion of multiple levels of automated behavior.

Not all automation functions exactly the same. For example, a toaster requires a different set of inputs than an ATM. Automation is not an on-and-off switch; it is not an “all-or-nothing phenomenon” (Cosenzo, Parasuraman, Novak, & Barnes, 2006). Rather, automation lends itself to varying degrees in which a task or function may be performed. Automation can extend all the way to fully autonomous (in which the human is ignored) to no assistance at all or can maintain a middle ground known as autonomy (Endsley, 2015).

Several taxonomies have been created in hopes of creating a suitable range of automation levels.

Sheridan and Verplank (1978) created ten levels of automation model to complement the four-stage model of human information processing. The following list starts at the highest level of automation capability and decreases in automation involvement.

10. The automation decides everything, acts autonomously, ignores the human.

9. The automation informs the human only if it, the automation, decides to.

8. The automation informs the human only if asked.

7. The automation executes automatically, and then necessarily informs the human.

6. The automation allows the human a restricted time to veto before automatic execution.

5. The automation executes a suggestion if the human approves.

4. The automation suggests one alternative.

3. The automation narrows a selection of alternatives down to a few.

2. The computer offers a complete set of decisions/action alternatives.

1. The computer offers no assistance: humans must take all decisions and actions (Thomas B. Sheridan & Verplank, 1978).

Wickens and Dixon (2007) created a four-stage model-to-model automation that builds on Sheridan and Verplank's work. Stage One included diagnostic aiding, which was filtering or focusing attention on information deemed to be of interest. Stage Two included information integration and inference formulation. Stage Three included decision recommendations and Stage Four included action implementation. This model of automation levels is simplistic and can be widely applied; however, a ten-level approach has also been developed, which provides a far more comprehensive approach.

Automation is not restricted to only one level of automation. Automated systems can operate at specific levels within this continuum and switch according to user preference. An example of this is the conflict detection and resolution system utilized by air traffic controllers to notify aircraft of conflicting flight paths. At level four, the automation would suggest a resolution, but at level six or greater, the system would automatically execute its own resolution advisory (Parasuraman et al., 2000). The level of automation can also shift over time as necessary.

Endsley provided a time-based graph for the levels of automation.

Endsley used a six-level approach to the levels of automation in Figure 22. Levels of Autonomy, which closely resembled the ten-level approach previously described.

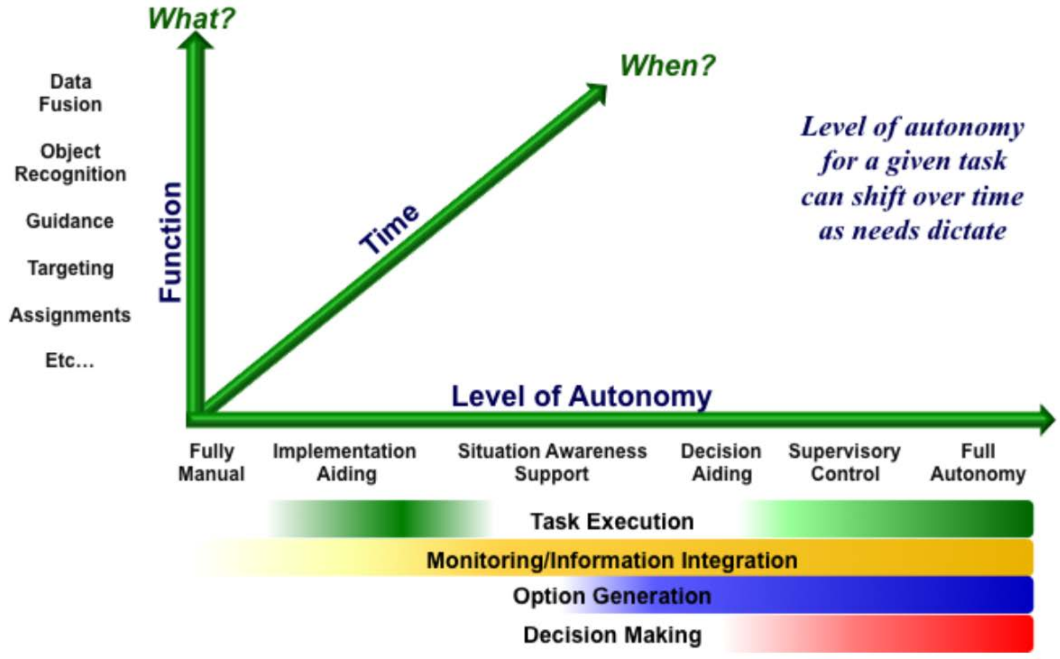


Figure 20. Levels of Autonomy (Endsley, 2015)

The four-stage view of human information processing presents a comprehensive, yet simple, approach to how humans acquire information and act on the acquired information. The first stage of human information processing includes the acquisition and registration of multiple information sources. This stage encompasses positioning and orienting all sensors related to processing and gathering information. The second stage includes the conscious perception and manipulation of all of the previously gathered information. The actions and cognitive actions performed in Stage Two all happen prior to the decision point. The third stage includes the cognitive processes where the human made the decision. The fourth and final stage includes the action consistent with the decision made in stage three (Parasuraman et al., 2000). Understanding the four stages of how a human processes information is imperative to understanding how a human interacts with automation, and what is the human’s role in automated systems.

Automation Benefits

For many years, the primary principles for adopting automation were technological feasibility and cost—all while disregarding the human's role in automation. Capability and cost are not comprehensive enough reasons for adopting automation that completely replaces operators. Humans are generally more flexible, adaptable, and creative than automation and are more capable in responding to changing or unforeseen conditions (Parasuraman & Riley, 1997). Given that the automation programmer cannot accurately predict all possibilities in a complex environment, humans are still required to exercise judgment while using automation. Automation is typically designed to automate everything that leads to an economic benefit, while allowing the human operator to supervise or manage the system (Parasuraman & Riley, 1997). The synergistic effect of humans and automated systems provides many benefits.

The benefits from automation utilization have been outlined by multiple literature sources. The United States Air Force outlined four high-level, key advantages of increased levels of automation. The first advantage was a reduction in unnecessary manual labor and lower system manning costs. The second advantage was the increase in range of operations and extension of manned capabilities. The third advantage was the time reduction required to conduct time-critical operations. Finally, increased levels of system autonomy promised to provide increased levels of operational reliability, persistence, and resilience (Endsley, 2015).

The Defense Science Board's Autonomy Task Force outlined several more benefits of utilizing automation. Automation has the potential to increase battle space awareness, as well as greater levels of situational awareness. Automation creates persistent visibility, while removing soldiers from dangerous situations (Kaminski, 2012).

However, the benefits of automation can also be seen on lower-levels. Automation may reduce human labor costs and expenses (Rovira et al., 2007; Rovira & Parasuraman, 2010; Wickens & Dixon, 2007). Automation can also increase the productivity and consistency of production as well as the quality and consistency of a product (Parasuraman & Riley, 1997). However, automation does have limitations and challenges.

Automation suffers from a wide range of issues. The Air Force has outlined three main concerns with automation:

- As hardware complexity grows, there will be more opportunity for failures.
- As software complexity grows there will be more opportunities for bugs and vulnerabilities.
- As these systems are injected into an adversarial environment, there will be opportunities for encountering situations that the original designers had never considered (Endsley, 2015)

Additionally, as automated systems are used in more complex environments, several more issues can potentially surface. These issues include reduced understanding of the system due to its complexity, reduced predictability in terms of how it will perform in any given situation, challenging the people who must interact with it, and greater vulnerabilities through the communications links created for human intervention used to offset the first two items (Endsley, 2015). Endsley argues that the ability to effectively use automation has been strained by multiple factors to include reduced situation awareness, increased workload, increased decision-making time, and difficulties in forming appropriate levels of trust in the automation (Endsley, 2015).

Any human operator of automation must demonstrate high levels of situation awareness to ensure the automated system performs in such a manner that is consistent with operational goals (Endsley, 2015). With greater levels of automation control, the human operator becomes less vigilant due to several factors, which include: " interfaces that do not provide needed information and often little feedback on system state; systems that require extensive human monitoring, a skill that people do not excel at due to decrements in vigilance that can occur after as little as 30 minutes; and a shift from active to passive processing of information" (Endsley, 2015). The issue of situation awareness has plagued several Air Force career fields and is frequently seen in the aviation career field. Aviation accidents have occurred due to a loss of pilot situation awareness in which the pilot was unable to respond to a critical event in a timely manner. However, the loss of situation awareness is not the only issue at hand. High levels of operator workload also play a critical role in optimizing the human-automation team.

One of the purposes of automation is to reduce workload on the operator; however, the automated system does not always successfully accomplish this purpose. The “irony of automation” is where the automation often increases workload during high workload phases and decreases during low workload phases (Bainbridge, 1983). This issue of shifting workload in inappropriate temporal phases is a critical issue of automation that needs to be addressed. Research needs to be completed to lay the groundwork for creating automated systems that can be trusted, easily understood, easy to work with, and work as they are supposed to work. Special attention needs to be paid to designing and implementing automation that does not leave the operators with an incoherent set of tasks that cannot be successfully automated (Endsley, 2015).

Automation Reliability

Agent reliability is one of the driving factors in how a human operator interacts with an automated system or an automated agent. When discussing reliability, the parlance of the automation community refers to the accuracy of the automation as opposed to automation failures (de Visser & Parasuraman, 2011; Dixon et al., 2006; Maynard & Rantanen, 2005; Merritt et al., 2013; Parasuraman & Miller, 2004; Ross et al., 2008; Rovira et al., 2007; Rovira & Parasuraman, 2010). The two common forms of errors made by unreliable automation are misses and false alarms. Misses occur when the automation fails to detect or complete a task. False alarms occur when the automation signals something is wrong or incorrect, but there is not an actual error (Dixon et al., 2006). Both misses and false alarms are most common due to diagnostic automation or agent systems that distinguish between states of safety and danger or correct and incorrect (Dixon & Wickens, 2006). The misses and false alarms stem from imperfect sensors, algorithms, noisy data, or probabilistic data in a complex, changing environment (Dixon & Wickens, 2006). However, the rate at which the automation is unreliable can vary greatly.

The effects of varying the reliability of automated aids is and has been studied for the past several years (Chen & Joyner, 2006; de Visser & Parasuraman, 2011; Maynard & Rantanen, 2005; Ross et al., 2008; Rovira et al., 2007; Rovira & Parasuraman, 2010; Wickens & Dixon, 2007). The research conducted in this area supports a nearly universal conclusion that human-automation team performance and the willingness of the user to trust and use the automated aid is greater when the automated aid has greater reliability (de Visser & Parasuraman, 2011; Ross et al., 2008; Rovira et al., 2007; Rovira & Parasuraman, 2010; Wickens & Dixon, 2007). However, there is little to no research conducted in what happens to user trust and human-agent team performance when the automated agent acts like a teammate rather than just an aid to the user or some form of diagnostic automation. When the automation functions as just an aid to the user, the threshold for usefulness has shown to be about 70% reliable depending on the task (Wickens & Dixon, 2007).

Research has shown that even with varying rates of automation reliability, human users are able to respond fairly accurately to the changes in automation reliability in terms of how they trust and rely on that automation (Ross et al., 2008). This innate ability of humans to perceive automation reliability allows for the human user to maximize overall performance across a range of automation reliability rates down to 70% (Ross et al., 2008). However, most users struggle with identifying the first failure of automated systems (Rovira et al., 2007). It is important to note that—even if an automated system operates at 100% reliability, the human operator can perceive the reliability as less than 100%. This happens when the automated system does not behave as the user expects the system to, even when the automated system has been designed correctly (Ho et al., 2005). An example of this can be seen in automated medication dispensers such as the Honeywell Independent LifeStyle Assistant (I.L.S.A). One user commented on I.L.S.A. not recognizing a change in her usage pattern, a feature it was not designed to provide. This perception of unreliability caused the user to not trust I.L.S.A and not use the system (Ho et al., 2005). Overall, unreliable automation can cause compliance issues in users.

Compliance

One of the effects of unreliable automation surfaces in the user compliance with the automation or agent. Compliance relates to the human operator’s response when an alarm or signal sounds, whether that alarm or signal is true or false (Chen & Barnes, 2014; Chen & Joyner, 2006; Dixon & Wickens, 2006; Dixon et al., 2006). A compliant human operator is an operator who rapidly switches his or her attention from concurrent activities to the alarm domain (Dixon & Wickens, 2006). The operator may then immediately initiate an alarm-appropriate response, such as hitting the snooze button on an alarm clock (Dixon & Wickens, 2006). Research has shown that as the rate of false alarms increases, the operators experience a reduction in compliance (Chen & Joyner, 2006; Dixon & Wickens, 2006; Dixon et al., 2006). This compliance reduction can result in longer response times to automation alerts or alarms (Dixon et al., 2006). However, as the false alarm rate increases to a certain threshold--which is different for every operator--the operator will start to disregard the false alarms entirely. This phenomenon is known as the "cry wolf" effect or "alarm fatigue" (Dixon et al., 2006). Multiple studies support the conclusion that compliance is the driving factor in task time completion, as well as accuracy in completing the task (Chen & Joyner, 2006; Dixon & Wickens, 2006; Dixon et al., 2006). However, there is a small amount of literature that addresses what happens to user compliance when the operator is working with an agent that works as a team member rather than just an aid. There have been few studies conducted in which the automation works at the highest level of automation capability. In all of the aforementioned studies, the researchers used an automated aid that provided some form of recommendation or alarm. This research needs to be expanded to address what happens to user compliance and the human-agent team when a fully autonomous agent, which ignores the human user, has imperfect reliability. This expansion of research will also need to address the issues of framing reliability information.

Framing the information regarding the reliability has been shown to affect the user's perception of the automation. Negative framing such as "the automation is 30% unreliable" can result in a greater negative perception of the automation rather than positive framing such as, " the automation is 70% reliable" (Lacson, Wiegmann, & Madhavan, 2005). The study also showed that compliance rates were affected by the framing and amount of information provided. Compliance rates were higher for the users that were informed about the reliability of the automated aid. However, it is important to note that regardless of the frame type, compliance rates were higher than when no framing was used.

Reliance

Reliance is, in some regards, the opposite of compliance. User reliance is described in several ways. The most common definition of user reliance a human user's state when an alert or alarm is silent, meaning everything is "ok" (Dixon & Wickens, 2006). Several researchers expanded this definition as follows: a discrete process of engaging or disengaging (Lee et al., 2004); what the human user does when the automation diagnoses noise in the environment (Dixon et al., 2006); and the tendency to employ automation to replace manual control (Ross et al., 2008). The level of user reliance can vary from over-reliance to under-reliance and everywhere in-between. Over-reliance occurs when a user employs automation even though the user is more capable or reliable than the automation, and thus the user would perform better without the automation (Cosenzo et al., 2006; Ho et al., 2005). Over-reliance can come from several factors, but one of the most prevalent factors is automation bias (Cosenzo et al., 2006). Each automation user is unique and has unique experiences, which have shaped how he or she interacts with automation. Just as experiences shape how a user may over-rely on automation, the experiences may also create under-reliance on automation. Under-reliance on automation is when the user should have relied on the automation rather than his or her own ability (Cosenzo et al., 2006; Ho et al., 2005). Over-reliance and under-reliance, as well as reliance in general, are strongly affected by the age of the user.

As humans age, they experience cognitive changes, which ultimately affect how they rely on automation (Ho et al., 2005). The primary cognitive changes that a user experiences as he or she is aging, includes deficits in: attention, memory, learning, decision-making, and reasoning (Ho et al., 2005). Although older users monitor and attend to automation more frequently, they still have greater rates of automation-induced errors. This finding points to several factors that lead to reliance issues (Ho et al., 2005). Older users experience deficits in working memory, which leads to greater reliance on automation. Older users are unable to recall an accurate mental representation of what the appropriate values should be, which makes the older users less able to determine whether the automation is making an appropriate decision (Ho et al., 2005). Older users also must devote more cognitive resources to determine if the automation is making a correct decision, which increases the workload of the user. Higher mental workloads have shown to increase user reliance on automation (Ho et al., 2005). However, higher mental workload is only one part of increasing reliance. The type of task also plays a role in user reliance on automation (Cosenzo et al., 2006). The exact role of how each task affects reliance is still being discovered. It is known that both reliance and compliance relate directly to a user's trust in the automation or agent.

Trust

Defining trust in automation is a challenge faced by the researchers in the automation field. There are a myriad of definitions for trust in automation. One of the more complete definitions is as follows:

> The willingness of a party to be vulnerable to the actions of another party based on the expectation that the other will perform a particular action important to the trustor, irrespective of the ability to monitor or control that other party (Verberne et al., 2012).

Ross et al. (2008) have defined trust as the operator's confidence in an automated system or agent to help the user achieve their goals in a situation characterized by uncertainty and vulnerability. Other research has explained trust as a user's expectation for the automation to be technically competent in its role performance (Lee et al., 2004). One source defines trust as an expectation that a service or commitment will be fulfilled (Hoffman, Lawson-Jenkins, & Blum, 2006). All of the definitions have a common theme of expectation, but how does the user form this expectation?

Forming an expectation or trust formation stems from both thinking and feeling (Hoff & Bashir, 2015). The emotions of the user are the primary drivers of trusting behavior (Hoff & Bashir, 2015; Lee et al., 2004). Unfortunately, feelings do not always follow rational patterns or logical progressions. Emotions have a critical role in human behavior, but they depend on thinking and human cognition (Lee et al., 2004). Analogic thought processes include using societal norms and opinions of others to determine trust (Hoff & Bashir, 2015). The current body of literature also points to several “bases” of trust which help determine how and why a person trusts (Li, Hess, & Valacich, 2008). Four bases exist for user trust: personality base, cognitive base, calculative base, and institutional base. The personality base is composed of two subcomponents: faith in humanity and trusting stance (Li et al., 2008; McKnight, Choudhury, & Kacmar, 2002). Faith in humanity is a person’s belief about general human nature and with greater faith in humanity, comes a greater amount of trust (Li et al., 2008; McKnight et al., 2002). The trusting stance refers to a person’s belief that better outcomes will come from treating trustees as though they are well-intentioned and reliable, regardless of the trustees’ real qualities (Li et al., 2008). A more positive trusting stance will lead to higher initial trust (Li et al., 2008). The cognitive trusting base refers to the use of reputation and stereotyping to classify trustees into groups of who can and cannot be trusted (Li et al., 2008). The third trusting base, calculative, refers to a person making trust decisions based on economic principles and what will benefit him or her (Li et al., 2008). The final trust base is institutional trusting base and refers to the impersonal structures that are inherent to specific environments which can lead to trust building (Li et al., 2008). In addition to trust bases, there were also empirical factors which influence trust.

Hoff and Bashir (2015) created a three-layered framework for conceptualizing trust variability. The three-layered framework included dispositional, situational, and learned trust (Hoff & Bashir, 2015). Dispositional trust included culture, age, gender, and personality. Culture played a pivotal role in how people trust. Trust varied across countries, races, religions, and socio-economic statuses; however, little research has been conducted on how culture affects trust in automation (Hoff & Bashir, 2015). As previously discussed, older users tended to trust and rely on automation and decision aids more than younger users (Ho et al., 2005; Hoff & Bashir, 2015). Gender is thought to play a role; however, little research has been conducted in analyzing the differences in trusting automation between the genders (Hoff & Bashir, 2015). The final component of dispositional trust is personality. Personality traits, such a neuroticism, negatively impact trust in automation (Hoff & Bashir, 2015).

The second layer of situational trust included external variability and internal variability. External variability is the factors outside the user's mind such as the type of system, system complexity, and framing of the task (Hoff & Bashir, 2015). Internal variability included of the internal factors such as self-confidence, mood, and attention span (Hoff & Bashir, 2015). The third layer was learned trust, which was when the user made a determination of trust based on previous experiences (Hoff & Bashir, 2015). All of the layers are necessary to understand and characterize user trust.

Characterizing and modeling trust in human-automation interactions is imperative for successful performance. If a human operator trusts a system's automation that has a lower reliability than manual operation, or if a human operator distrusts a system's automation that has a higher reliability than manual operation, poorly calibrated trust will likely result (Dzindolet et al., 2003). Calibrating an operator's trust in an automated system is necessary to prevent over-trust in the automation or distrust in the automation.

Over trusting an automated system can result in the human operator misusing the automated system (Lee et al., 2004). Misuse is a rather broad term when applied to automation and encompasses several key components of trust in automation. Misuse is classically defined as "overreliance on automation" (Dzindolet et al., 2003; Parasuraman & Miller, 2004). Misuse has also been described as a mental state that consists of a low-level of questioning (Dzindolet et al., 2003). Misuse stems from several variables. One of the most influential variables that drives misuse was operator knowledge about the system (Hoff & Bashir, 2015). If the human operator has an understanding of the system's purpose or how it functions, it is likely that the human operator will have greater success in accurately aligning his or her trust to the system automation. Conversely, if a human operator has little to no knowledge about the system's purpose or how the system functions, it is unlikely that he or she will be able to accurately align his or her trust to the automated system's reliability. This misalignment of trust is especially prevalent when the formation of trust between the human operator and the automated system depends on the automated system's performance. The system's performance varies in separate contexts and differing temporal phases (Hoff & Bashir, 2015). The effects of misuse can have tragic consequences. One example of misuse is the Costa Concordia cruise ship that crashed off the coast of Italy in January of 2012. The crash was thought to be the result of the ship's captain under-trusting the ship's navigation system. The investigations that were conducted after the incident point to the fact that the captain diverged from the ship's programmed route prior to hitting the shallow reef which caused the incident (Hoff & Bashir, 2015).

However, the effects of misuse can be mitigated through several means. One of the means with which to mitigate the prevalence of misuse from overtrust is training. Previous studies have indicated that by training operators about an automated system's actual reliability, it is feasible to alter trust and reliance patterns, increase task performance, and reduce complacency (Hoff & Bashir, 2015; Koustanai et al., 2012). Additionally, misuse from overtrust can also be decreased by providing reliable feedback to the human operator. A study conducted in 2005 examined the idea of providing real-time confidence levels to help users calibrate their trust effectively. The study showed that providing a confidence level to the human operator allowed for the human operator to rely more frequently on the automated system. This study suggested that the operator had an increase in system trust (Antifakos et al., 2005).

Just as overtrust is a prevalent issue for automated systems, undertrust also provides a challenge to automation designers. Undertrust can instigate disuse of the automated systems. Disuse is defined as the situation in which users fail to rely on automation, when doing so would improve performance (Merritt et al., 2013). Several studies have demonstrated that disuse originates from a lack of appropriate instruction. If an automated system malfunctions in a way that the human operator is unable to explain or understand, disuse will likely occur (Dzindolet et al., 2003). In the experiments conducted by Dzindolet et al. (2003), it was found that disuse of the automated system was more prevalent than misuse. (Dzindolet et al., 2003)

The study demonstrates that disuse is an important issue that needs to be addressed through further research. Additionally, the study showed that an operator can switch from misuse to disuse of an automated system with the smallest of reliability changes.

Human-Agent Teaming

Before delving into the world of human-agent teaming, one must step back and examine agency. The literature surrounding agency is constantly evolving—thus the definition of "agent" is a fleeting concept. Franklin and Graesser (2005) provided 11 different definitions of "agent." The most simplistic definition is that of the *Brustonloni Agent* which states that agents are, "systems capable of autonomous, purposeful action in the real world" (Bradshaw, Sierhuis, Acquisti, & Feltovich, 2003; Franklin & Graesser, 2005). On the other side of the spectrum, the *Wooldridge-Jennings Agent* is defined as a system that is software based and has several distinct autonomous properties. First, the agent works without direct intervention of humans and can control their actions. They also have the ability to communicate to other agents. The agents can also perceive and react to their environment. Finally, the agents demonstrate a pro-activeness attribute that allows them to anticipate future events.

However, Wooldridge and Jennings (1995) argued that the *Wooldridge-Jennings Agent* is a weak notion of agency. This weak notion of agency is commonly applied in a wide-range of research (Wooldridge & Jennings, 1995). The weak notion of agency is the definition that will be applied for the research conducted in this thesis. The stronger notion of agency implies concepts that are usually applied to humans, which include *mentalistic* notions such as knowledge, belief, intention, and obligation (Franklin & Graesser, 2005). In some instances, researchers have gone on to describe the agents as "emotional agents" (Franklin & Graesser, 2005; Nourbakhsh et al., 2005).

The term "agent" extends its definition from very simplistic application such as a "reflex agent," like a thermostat, to disembodied software agents that use algorithms to learn and adapt to changing environments (Chen & Barnes, 2014). Chen and Barnes (2014) outlined three different types of agents that were commonly seen. The three were teaming agents, hierarchical agents, and flexible agents. Each system had its unique strengthens and weaknesses. Teaming agents have several key strengths. The strengths of teaming agents included supporting optimal task-allocation planning and flexible plan-execution; allowing plan deviations, and incorporating the distribution of useful information (Chen & Barnes, 2014). However, teaming agents are more sensitive to changes in tasking environments than human-human teams (Chen & Barnes, 2014). Hierarchical agents excel at dividing task complexity between senior and specialized agents but struggle with communication between the agents (Chen & Barnes, 2014). Flexible agents are capable of effectively reducing workload and increasing situation awareness but struggle with sudden changes in task state and communication between the human and other agents (Chen & Barnes, 2014). Despite differing types of agents, the human still plays a role in interacting with the agents.

Human and agent roles are very different in terms of strengths and weaknesses. Agents excel at making calculations and using algorithms effectively. The means with which an agent processes a situation are based on algorithms rather than previous experiences like a human. The combination of both human and agents demonstrated effectiveness in open world environments(Chen & Barnes, 2014). An example of this type of situation is in combat environments. Combat environments are very dynamic and stochastic in nature, which does not allow for "preprogramming" an agent. The human is more capable of adapting to this dynamic environment and leveraging the strengths of the agent to create a synergistic team effect. The current construct of human-agent teaming generally places the agent as a subordinate team member to the human operator (Chen & Barnes, 2014). However, in emergency situations, such as collision-avoidance systems in cars, the agent is in control and does not alert or notify the human of its actions. The human role is slowly transitioning from the human acting as the manual operator to more of a supervisory and televisory controller (Urlings, Sioutis, Tweedale, Ichalkaranje, & Jain, 2006). As a supervisory controller, the human is functionally removed from the system; as a televisory controller, the human was physically removed from the system (Urlings et al., 2006). More specifically, supervisory control allows for the human to set initial conditions of the agent and periodically readjust and receive feedback (Urlings et al., 2006). A supervisory controller has the following responsibilities: plan the task execution, program the agent, monitor the agent, intervene when the agent experiences a failure mode, and learn from the agent to make better applications in the future (T.B. Sheridan, 1984; Urlings et al., 2006). Conversely, the agent's responsibilities include: maintain adequate knowledge base with regard to task, create alternative control strategies, detect or plan for failures, recall past data or performance, and provide a flexible human interface (T.B. Sheridan, 1984; Urlings et al., 2006). Once again special considerations need to be put in place to address both the strengths and weaknesses of humans and agents alike.

Televisory control is in many ways similar to supervisory control, with the exception of separating the human from the agent in a physical manner. Televisory control situations exist and are growing in the fields of space, undersea, construction, medicine, and military environments (T.B Sheridan, 1984; T. B. Sheridan & Verplank, 1978; Urlings et al., 2006). Televisory control in human-agent teaming revolutionized the manner in which the DoD conducts operations. The United States Air Force has dramatically increased the use of Unmanned Aerial Vehicles (UAVS) in the last decade, which has in turn increased the demand for greater understanding of the relationship between humans and agents (Endsley, 2015; Kaminski, 2012; Urlings et al., 2006). One component of the relationship between humans and agents that resembles human-human relationships is communication.

Communication between humans and agents in human-agent teams is critical for successful team performance (Chen & Barnes, 2014). Communication facilitates coordination and collaboration. Much like human-human teams, humans and agents communicate for the sake of task allocation. Research is finding that high levels of communication are necessary for human-agent team performance, when the team is subject to new and complex environments (Urlings et al., 2006). However, effective communication requires clear and effective syntax an semantics but is also practical to the task environment and speaker(Chen & Barnes, 2014). This effective communication can be difficult to ascertain due to the current level of technology. Current systems struggle with communication and collaboration due to the speed and efficiency of speech processing automation (Chen & Barnes, 2014; Klein et al., 2004).

Summary

Throughout the course of this review, it is apparent that human-agent teaming appears nearly everywhere in everyday life. The agent or automation can take the role of ten different levels of involvement in the human-agent team. The levels range from no assistance to acting without input from the human user. Using automation presents both benefits and challenges to the human user. The automation can help increase performance and reduce workload, but it can also decrease performance and increase workload if the automation is not implemented correctly. Incorrect implementation of automation, such as reduced automation reliability, leads to issues with compliance and reliance. Issues surrounding compliance, reliance, and reduced reliability present the notion of users failing to trust in automated agents. This review has shown that there is a gap in current research that addresses how reduced reliability agents (that act as teammates) affect human-agent team performance. Additionally, there is little to no research that examines how varying demographic variables, such as age, affect user compliance, reliance, and trust in unreliable agents.

Appendix 2: Detailed Model Descriptions

This appendix provides all of the details for the models and simulations used in Chapter 2 of this thesis.

Activity Modeling

Before the simulation could begin, a model was created. Using the construct of SysML, an activity diagram was created to model all of the tasks associated with the human player and 100% reliable straight line automation.

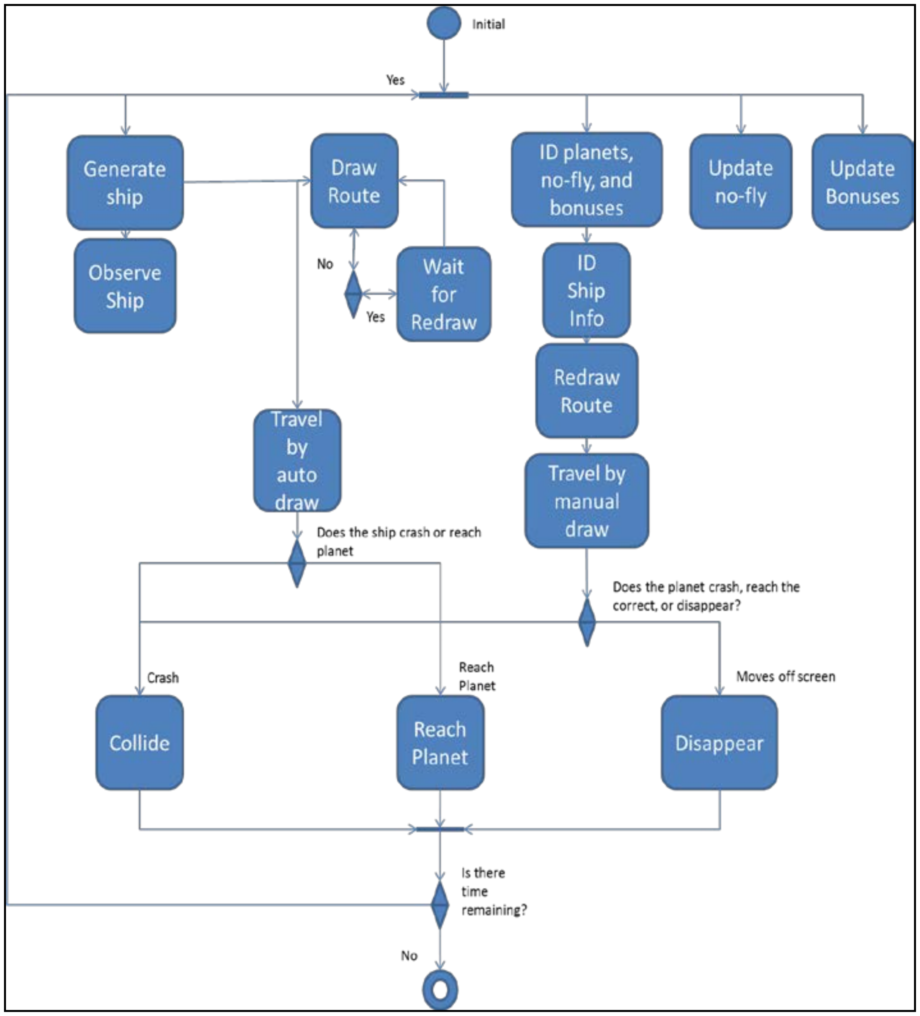


Figure 21. Activity Model

Model depicts the task network for the Space Navigator game with 100% reliability. Each of the activity nodes are described below.

- Player Observe Ship
  - Description : The human player will observe the generation of ships produced by the program
  - Who performs? : The human player
  - Resources/ Materials needed (inputs) : A tablet pc, operational copy of Space Navigator, a charger to make sure the tablet is fully charged
  - What is produced (outputs): The human player's situational awareness is updated
  - Decision Logic: None
- Wait for Redraw
  - Description : The automation waits to draw a route if the user is manipulating the ship of interest
  - Who performs? : The automation
  - Resources/ Materials needed (inputs) : Generated ships
  - What is produced (outputs): Waiting for a redraw
  - Decision Logic: The model proceeds back to drawing routes
- Identify planets, no-fly zones, and bonuses
  - Description: The user identifies all pertinent background information.
  - Who performs? : The human player
  - Resources/ Materials needed (inputs) : Generated planets, bonuses, and no-fly zones
  - What is produced (outputs): The human's situation awareness is updated
  - Decision Logic: None
- Identify pertinent ship info
  - Description: The user identifies all pertinent ship information such as trajectory, speed, possible collisions, and the need to redraw.
  - Who performs? : The human player
  - Resources/ Materials needed (inputs) : Generated planets, bonuses, no-fly zones, ship
  - What is produced (outputs): The human's situation awareness is updated
  - Decision Logic: None
- Update no-fly zones
  - Description: The program generates" no-fly zones" randomly on the game screen. The "no fly zones" are square, grey boxes that subtract ten points per second, per space ship that is in the "no fly zone".
  - Who performs? : The program
  - Resources/ Materials needed (inputs): The program requires the game background to place the "no fly zones" on.
  - What is produced (outputs): Square, grey boxes
  - Decision Logic: none
- Update Bonuses
  - Description: The program generates bonuses. Bonuses are small orbs that are generated randomly across the game screen and are stationary. When collected by the spaceship, the provide 50 points. The bonuses are continually spawned as long as there is time remaining

  - Who performs? : The program
  - Resources/ Materials needed (inputs): The program requires the game background to place the bonuses on. The program also requires the plants to be in location so as to not place the bonuses on top of the planets.
  - What is produced (outputs): Small orbs worth 50 points
  - Decision Logic: None
- Generate Ship
  - Description: The program generates spaceships on the edges of the screen. The spaceships are medium sized orbs that correspond to the respective planets. The ships float randomly on the game background until acted upon by the program or human player. The spaceships are spawned at a rate of one ship per every two seconds while there is game time remaining.
  - Who performs? : The program
  - Resources/ Materials needed (inputs): The program requires the game background to place the space ships on. The program needs the background dimensions in order to correctly place the spaceships on the edges
  - What is produced (outputs): Space ships on the periphery of the game screen
  - Decision Logic: None
- Draw Route
  - Description: Program provides automation that draws routes from the generated space ships to their respectively colored planets. The automation does not account for bonuses, no fly zones, or the position of other space ships.
  - Who performs? : The program
  - Resources/ Materials needed (inputs) : The space ships, planets
  - What is produced (outputs): Routes from spaceships to planets
  - Decision Logic: The model proceeds to a decision node
    - If the player is redrawing routes, then model will go back to drawing routes as long as there are ships without routes on screen.
    - If the player is drawing routes, then the model will proceed to the “Wait for Redraw” activity.
- Redraw automated route by human player
  - Description: The human player draws a new route for a spaceship. The human player drags his/her finger from the spaceship to a new location.
  - Who performs? : The human player
  - Resources/ Materials needed (inputs) : The spaceship must be generated with a route
  - What is produced (outputs): A new route
  - Decision Logic: none
- Travel if by auto draw
  - Description: The space follows the drawn route. This task accounts for collecting bonuses and going through no-fly zones.

    - Who performs? : The program
    - Resources/ Materials needed (inputs) : The spaceship and the drawn route by the automation
    - What is produced (outputs): Movement of the spaceship along the drawn route
    - Decision Logic: The model proceeds to a decision node
        - The spaceship will either crash or reach the correct planet
    - Decision Logic: none
- Travel if by manual draw
    - Description: The space follows the drawn route. This task accounts for collecting bonuses and going through no-fly zones.
    - Who performs? : The program
    - Resources/ Materials needed (inputs) : The spaceship and the drawn route by the human player
    - What is produced (outputs): Movement of the spaceship along the drawn route
    - Decision Logic: The model proceeds to a decision node
        - The spaceship will either crash, reach the correct planet or disappear off stage
- Collide
    - Description: The spaceship collides with another spaceship on the screen. Both of the spaceships are removed from the screen and the player loses 100 points for each spaceship that was lost in the accident.
    - Who performs? : The program is responsible for the deletion of the spaceships, but the route could have been drawn by either the automation or the human player
    - Resources/ Materials needed (inputs) : Spaceship, route, and another spaceship
    - What is produced (outputs): The loss of the involved spaceships and the loss of 100 points per spaceship
    - Decision Logic: None
- Reach Correct Planet
    - Description: The spaceship reaches the correct planet. The spaceship disappears and the human player receives 100 points
    - Who performs? : The program is responsible for deleting the spaceship and awarding 100 points to the human player
    - Resources/ Materials needed (inputs) : The planet and the spaceship
    - What is produced (outputs): The spaceship is removed and the human player receives 100 points
    - Decision Logic: None
- Disappear
    - Description : The spaceship disappears off screen
    - Who performs? : The program is responsible for deleting the spaceship
    - Resources/ Materials needed (inputs) : Spaceship
    - What is produced (outputs): The spaceship is removed
    - Decision Logic: None

Final Node-End Game

The game ends

IMPRINT Task Network Description

The following section further decomposes the IMPRINT model depicted in Figure 5 in Chapter II. The “Draw Route” task does not begin until there are ships on screen without routes and will continue to loop on itself while there are ships on screen without routes. The task also accounts for the human user redrawing and will wait to redraw routes. The task decrements the variable “ShipWithoutRoute”. Additionally, the task positively increments the variable, “ShipWithRoute”, which is the number of ships that have routes. The “Wait for Redraw” task stems from the need for the automation to wait for the human player while he or she redraws a route. The “Travel if by auto draw” task models a ship traveling on a route drawn by the automation. This task accounts for the ship picking up bonuses and/or flying through “no-fly zones”. The task positively increments the “PickUpBonus” variable which is the total number of bonuses picked up. The task also positively increments the “Score” variable by 50 points for each bonus picked up. At the same time, the task decrements the “BonusCount” variable, which is the number of bonuses on screen. The task also counts the number of no-fly zones passed through, which is captured in the “NoFlyZonesPassedThrough”. Finally, the task decrements the score by 40 points for each no fly zone passed through. It is assumed that the average amount of time spent is 4 seconds in each no fly zone passed through and the penalty is 10 points per second. The task then accounts for sending the ship to the correct planet or colliding with another ship.

The “Update no-fly zones” task models the system updating the zones every thirty seconds. The “Update Bonus Locations” task ensures that every thirty seconds the game background populated with three bonus orbs and is modeled using the “BonusCount” variable. The “Operate clock” task is used to control the length of each game. The games are five minute in length.

The “Identify planets, no-fly zones, and bonuses” task accounts for the user examining all of the background game information. Once the human user has accomplished this task, he or she will move on to identifying all pertinent ship information such as the automated route drawn, if the ship is projected to collect bonuses, pass through a no fly zone, or potentially collide with another ship. The task also ensures there are ships on screen before it passes to redrawing routes. Before the “Redraw automated route by human player” task begins, the release condition ensures that there are ships with automatically drawn routes and that there are ships on screen. The task increases the variable “ShipWithRoute” and decrease the “ShipWithoutRoute”. The model then progresses to the ship traveling task, “Travel if by manual draw”. This task accounts for the ship picking up bonuses and/or flying through “no-fly zones”. The task positively increments the “PickUpBonus” variable which is the total number of bonuses picked up. The task also positively increments the “Score” variable by 50 points for each bonus picked up. At the same time, the task decrements the “BonusCount” variable, which is the number of bonuses on screen. The task also counts the number of no-fly zones passed through, which is captured in the “NoFlyZonesPassedThrough”. Finally, the task decrements the score by 40 points for each no-fly zone passed through. It is assumed that the average amount of time spent is 4 seconds in each no-fly zone passed through and the penalty is 10 points per second. The task then accounts for sending the ship to the correct planet, colliding with another ship, or disappearing off screen.

After traveling, a ship has two possible outcomes if it is traveling on an automated route, collide or reach the correct planet. However, if the ship is traveling on a human user route, the ship can collide, reach the correct planet, or disappear. If the ship goes to the “Collides” task, the task will positively increment the “NumCrashes” variable, which tracks the number of crashes. The task will also decrement the variables “ShipWithRoute” and “ShipsOnScreen” because once a ship collides with another ship; the ship will disappear. Finally, the task decrements the “Score” variable by one-hundred points. The “Reaches Planet” task increments the “PlanetsReached” variable, which counts the total number of planet reached. The task will also decrement the variables “ShipWithRoute” and “ShipsOnScreen” because once a ship reaches the correct planet; the ship will disappear. Additionally, the task increases “Score” by one-hundred points. The final task encompasses when a ship disappears. The task “Disappears” accounts for when a ship with a manually draw route goes off screen. The task decrements both “ShipWithRoute” and “ShipsOnScreen”.

Data Description

Maj Bindewald’s involved 36 volunteers with an average age of 32.5 years and a range of 22 to 39 years. In an effort to provide clarity, the following section will be broken down into each of the tasks and will cover all of the associated probability distributions/probabilities/task times selected for each of the tasks.

Generate Ship

The task time is two seconds. The game generates ships every 2 seconds.

Observe Ship

It is assumed that it will take the average user 2 seconds to see that a ship has been created and acknowledge its location on screen. This comes from person experience playing the game.

Draw Route

The task time is set at .1 seconds. The game generates routes when ships are on screen every .1 seconds. Additionally, the task calculates if the model will need to wait to draw routes will the human player is redrawing a route. The probability function comes from Maj. Bindewald’s experiment three data and Jayson Boubin’s analysis. The probability function stems from the number of ships on screen.

Wait For Redraw

The about of time spent waiting for a redraw is distributed as a Weibull distribution with a beta value of 4.1535 and a lambda of .8564. This comes from Maj. Bindewald’s data. The function is capped at thirty seconds because they probability of the wait time being over thirty seconds is nearly zero.

Travel if by auto

The about of time the ship spends traveling is distributed as a Weibull distribution with a beta value of 13.1238 and a lambda of 1.9895. This comes from Maj. Bindewald's data. The function is capped at 45 seconds because they probability of the travel time being over 45 is nearly 0. This task also captures the probability of a ship collecting a bonus or flying through a no fly zone. The probability of a ship collecting a bonus, given there is a bonus on screen, is 22%. This comes from the number of bonuses collected in Maj. Bindewald's data. Additionally, the probability of flying through a no fly zone is 51%. This was also calculated from Maj. Bindewald's data. Finally, the task calculates the probability to either have the ship collide with another ship or reach a planet. The probability function is dependent on the number of ships on screen. The task only addresses colliding or reaching a planet because a ship with an automated route will not disappear off screen.

Update no-fly zones

The task time is set at 30 seconds because the game updates no-fly zones every 30 seconds.

Update bonus locations

The task time is set at 30 seconds because the game updates bonuses every 30 seconds.

Operate Clock

The task time is 5 minutes because each of the games lasts exactly 5 minutes.

Identify planets, no-fly zones, and bonuses.

It is assumed that it will take the user 1 second to effectively look at the screen and gather all of the necessary background information. This comes from personal experience playing the game.

Identify pertinent ship info.

The amount of time the user spends identifying pertinent ship info such as route and projected collisions is distributed as a Weibull distribution with a beta value of 2.6407 and a lambda of 1.7529. It was calculated by looking at the amount of time between user redraws. This comes from Maj. Bindewald's data. The function is capped at 10 seconds because they probability of the identify time being over 10 is nearly 0.

Redraw automated route by human player

The about of time the user spends redrawing a ship is distributed as a Weibull distribution with a beta value of .6463 and a lambda of 1.8158. It was calculated by looking at the amount of time the user had his or her finger on the screen when redrawing a route. This comes from Maj. Bindewald's data. The function is capped at 2 seconds because they probability of redrawing being over two is nearly 0.

Travel if by manual draw

The about of time the ship spends traveling is distributed as a Weibull distribution with a beta value of 11.0257 and a lambda of 1.7784. This comes from Maj. Bindewald's data. The function is capped at 45 seconds because they probability of the travel time being over 45 is nearly 0. This task also captures the probability of a ship collecting a bonus or flying through a no fly zone. The probability of a ship collecting a bonus, given there is a bonus on screen, is 22%. This comes from the number of bonuses collected in Maj. Bindewald's data. Additionally, the probability of flying through a no fly zone is 51%. This was also calculated from Maj. Bindewald's data. Finally, the task calculates the probability to have the ship collide with another ship, reach a planet, or disappear. The probability function is dependent on the number of ships on screen. This task addresses when the ship collides, reaches the correct planet or disappears.

Collides, Reaches Planet, Disappears

None of these tasks have task times or probabilities associated with them.

Assumptions

In order to model the system in IMPRINT, there were assumptions that had to be made either due to complications in accurately depicting the activity or because of the limitations of IMPRINT. All of the assumptions used in the creation of the IMPRINT model are described below.

- User is proficient with Space Navigator -The user data is only collected after the user has played 4 or more practice games
- Planets already generated-The model begins once the software is up and running
- Software is working correctly (no glitches) - The game will not have any lag which could affect user performance
- User does not learn more as game progresses-It is possible that the user will learn or change his/her strategy throughout the trials. However, this would be hard to capture so it is assumed it will not change
- Max of 20 ships onscreen at once-Data has shown that in almost all instances there will be no more than 20 ships on the screen at once
- The user will only redraw routes- it is assumed that if the user will only redraw routes that have already been drawing. It is assumed that the using will not be faster than the .1 second automation
- It takes the user 1 second to identify planets, no fly zones and bonuses-Data has shown that it takes on average 1 second to identify planets, no fly zones, and bonuses. For the sake of modeling, it is assumed this is a deterministic value because I have not found a distribution for this.
- Identifying pertinent ship info will take no longer than 10 seconds-Data has shown that identifying routes follows a Weibull distribution and takes no more than 10 seconds to identify the routes and collisions for each ship
- Travel time will take no more than 45 seconds regardless of the route or who drew the route-Previous data indicates it takes at most 45 seconds for a ship to follow a route and reach the destination if it does not crash. This also follows a Weibull distribution

- If the human player redraws a route, the automation will not attempt to redraw-Rather than trying to model a situation where the user and automation constantly trade-off drawing multiple routes, it is assumed once the user redraws a route, the automation won't try to draw again
- Generating no fly zones, bonuses, and generating ship happen at fixed times-No fly zones and bonuses spawn every thirty seconds. Ships spawn every 2 seconds until there are 13 ships on screen
- Time to draw is less than 2 seconds for the user-Previous data has shown it takes the user less than 2 seconds to draw a route and follows a Weibull distribution.
- Average time spent in no-fly zones is 4 seconds-Personal experience provides this average
- Time to identify a ship being created- Personal experience provides this average
- Jayson Boubin and Maj. Bindewald's data and analysis accurately reflect human players -Both Jayson and Maj. Bindewald appear to be SMEs with the Space Navigator game
- The travel time to an incorrect planet is Weibull distribution with values of (5, 1)-This comes from a user experience. It appears to take about 5 seconds to correct for an incorrect draw.
- The number of ships that collide after an incorrect draw follows the same probability function as the number of ships that collide after an automatically drawn route-I am assuming that the probability of collisions do not change regardless of the planet the ship is drawn to.
- The ability to pick up bonuses, or go through no-fly zones is not being accounted for when a ship is traveling to an incorrect planet. There is not data for the instances when the ship is traveling for a short period on an incorrect route; therefore adding this to the model would be a guess. It is acknowledged that this will have an impact on the score, but it is believed that the effect is not large enough to skew the output analysis.

Appendix 3: Reduced Reliabiltiy Human-in-the-Loop Experiment Instruments

----

Space Navigator Detailed Task Instructions

Task Performance Instructions:

Research Assistant will read the instructions below to the participant:

> "The research environment is a simple open-source route creation game similar to Harbor Master and Flight Control. Spaceships appear and the user must direct them (without causing collisions) to a designated planet by drawing lines on the game screen using his/her finger. Additionally, small bonuses appear in random locations throughout execution. If the path of a given spaceship crosses over one of these bonuses it is `picked up' and a point bonus is given. Points are taken away from the player for allowing spaceships to traverse `no-fly zones.' The document you are about to read is a comprehensive game-play description:"

Participants will read the following at their own pace:

-Begin document-

Space Navigator Game Goal:

The goal of the game is to obtain the most points over a period of five minutes.

Game Components:

1. Spaceships – Spaceships are entities that "fly" onto the screen, and must be controlled by the user. The goal of each spaceship is to land on its assigned destination planet. Spaceships appear at regular intervals. Each spaceship appears at a random location along the edge of the screen. In order to ensure against collisions, consecutive spaceships cannot come from overlapping locations (i.e. they must be more than a spaceship's width away from the location of the last ships appearance). A spaceship appears on the screen moving toward the screen edge opposite from which it emanated.
2. Planets – Planets are fixed objects on the screen. Each spaceship is assigned a destination planet (colors of the spaceship and its destination planet will match). When the edge of spaceship overlaps the edge of the destination planet, a spaceship lands on the planet and points are given to the player accordingly. Spaceships cannot collide with planets that are not their destination and will instead move through them.
3. Routes – A route represents the line of movement that a spaceship is assigned to follow from its current location to the destination planet. A valid route emanates from a spaceship and ends at the given spaceship's destination planet. A valid route is drawn by touching the finger on top of the given spaceship, dragging the pressed finger across the screen, and releasing at the ship's destination planet. The line traced by the finger is recorded as the ship's route, and is displayed on the screen as a series of red dots. Routes will be followed unconditionally unless redrawn by selecting a given spaceship and drawing a valid route. The ship automatically traverses its completed route until it lands on a planet or reaches the end of the route, in which case it will continue in the direction of the last heading
4. Bonuses – Bonuses are objects that can be picked up by spaceships to gain extra points. A bonus can appear randomly at any location on the screen. They appear at time intervals according to some probability distribution (in the standard case, this is a uniform distribution). A bonus is picked up by a ship if the ships edge comes into contact with the edge of the bonus. Thus, a ship can pick up a bonus by following a route that leads it into contact with the bonus. Any ship can pick up any bonus. Upon a ship picking up a bonus, the player is allotted the number of points the bonus is worth. Once it appears, a bonus does not move.

5. No-fly zones – No-fly zones are areas that randomly appear on the screen. Each no-fly zone is a small rectangle that covers a portion of the screen. A spaceship that enters a no-fly zone will lose points according to the amount of time it is in the no-fly zone, for each second in the no-fly zone the player loses ten points. A spaceship is in a given no-fly zone when any part of the spaceship overlaps any part of the no-fly zone. The no-fly zones will move at set intervals to random places on the screen.

6. Automated Agent – During games with automation, the automated agent generates routes automatically for each ship. The agents' goal is to route the spaceships to the corresponding planet. Routes are drawn as straight lines from the ship to the planet. Routes do not take into account bonuses, no fly zones, or other ships. Some agents may experience less than 100% reliability.

Points can be obtained or lost in four ways:

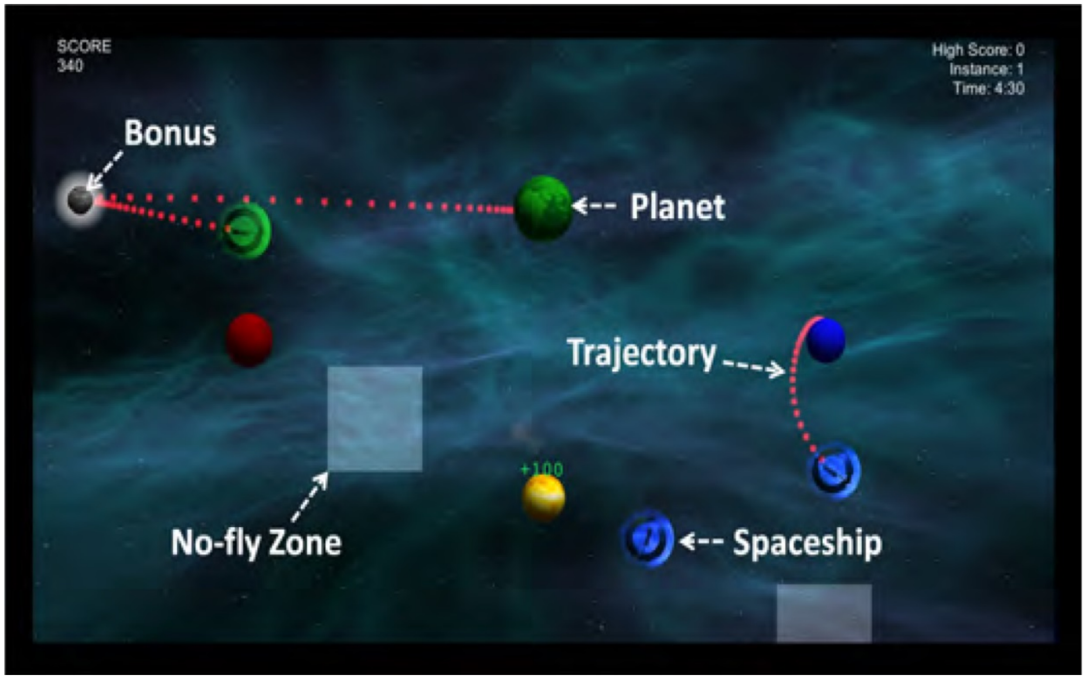
1. Points are gained for successfully landing a ship on a planet – Upon landing on a planet, a spaceship disappears and 100 points are given to the player.
2. Points are gained for picking up bonuses – 50 points are given for each bonus picked up.
3. Points are lost for traversing "no-fly zones" – 10 Points are lost for each second a spaceship is in a no-fly zone. Each spaceship loses points for each no-fly zone it is currently traversing.
4. Points are lost when two or more ships collide – When two ships overlap in any way on the screen, a collision occurs. Both ships, and any routes assigned them, are immediately removed from the screen, and 100 points are taken away from the player for each ship lost. A collision is signified by a graphic showing a small explosion.

Game end condition:

The game ends when a time limit of five minutes is reached. Upon the end of time, the game is immediately ended and no more points can be earned.

-End document-

Screen shot of Navigator



Contact Information Questionnaire for Extra Credit and C-REP

1. Participant Number (Assigned By Researcher):

2. Name:

3. Email Address:

Demographics Questionnaire

1. Participant Number: ______________
2. Age: ______________
3. Handedness:
   What hand are you? LEFT RIGHT BOTH

   a. For the following please list what hand you primarily use to:

   ___Write with ___Throw a ball ___Use a computer mouse

4. Are you male or female? Male___ Female____ Prefer not to answer ____

5. What's your highest education level?

   A. Lower than high school
   B. Graduated from high school
   C. Some college
   D. Graduated from college
   E. Some Post-graduate training after college
   F. Master's degree
   G. Ph.D. degree

6. How much experience do you have with the following:

   a. In a given year, on how many days do you interact with the following types of devices?

	Never	<1 per month	1-3 per month	1-2 per week	3-6 per week	Daily
Laptop computer	0	1	2	3	4	5
Tablet computer	0	1	2	3	4	5
Smart phone	0	1	2	3	4	5
Desktop computer	0	1	2	3	4	5
Gaming consoles	0	1	2	3	4	5

b. In a given year, on how many days do you play **video games** of the following type?

	Never	<1 per month	1-3 per month	1-2 per week	3-6 per week	Daily
Simulation (SimCity)	0	1	2	3	4	5
Role-Playing (WoW)	0	1	2	3	4	5

Action (Mario, Donky Kong)	0	1	2	3	4	5
First Person Shooter (Halo)	0	1	2	3	4	5
Strategy (Civilization)	0	1	2	3	4	5
Puzzle (Tetris, Candy Crush)	0	1	2	3	4	5
Casual (Angry Birds)	0	1	2	3	4	5
Music (Guitar Hero)	0	1	2	3	4	5
Sports (Madden Football)	0	1	2	3	4	5
Board (Monopoly)	0	1	2	3	4	5
Card (Poker, Pinochle)	0	1	2	3	4	5
All Video Games (TOTAL)	0	1	2	3	4	5

After completing the above questionnaire, participants will take an Implicit Association Test designed to discern their level of implicit trust or mistrust in automated systems. The features of this test are as follows:

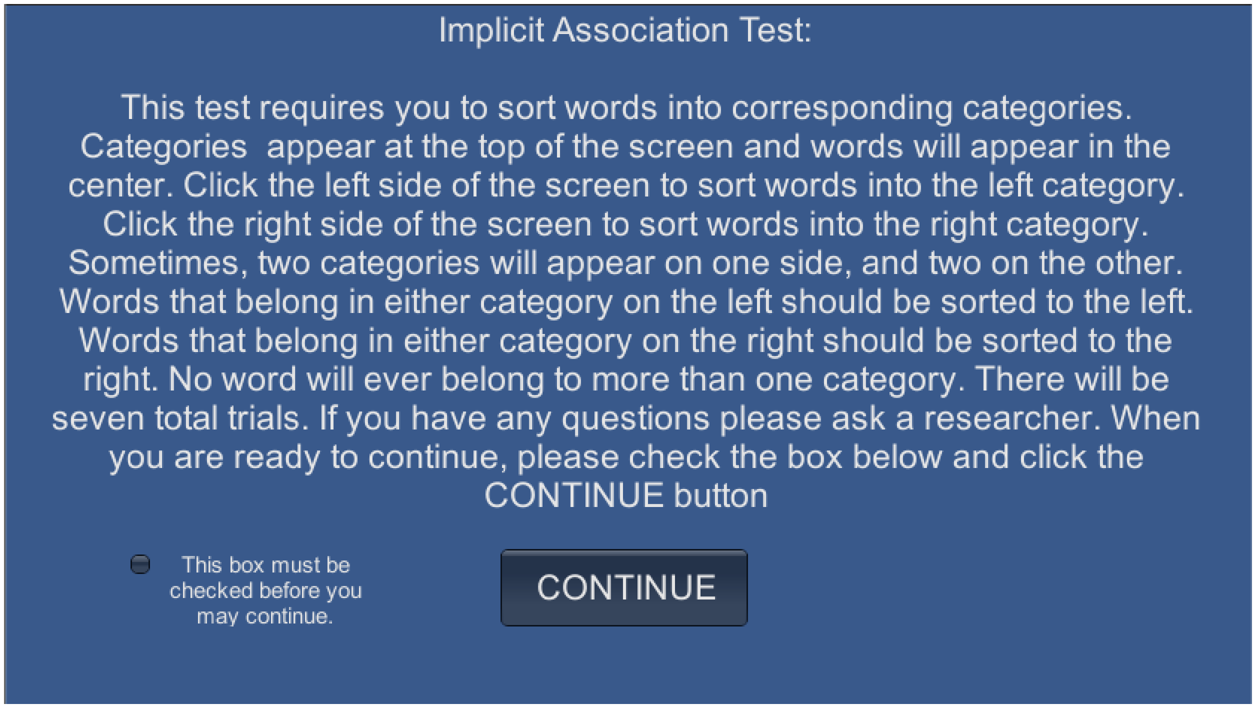
Categories for association: Trust, Mistrust, Machine, Human

Words by category:

- Trust: Kind, Trustworthy, Altruistic, Honest, Reliable, Truthful
- Mistrust: Scheming, Greedy, Corrupt, Doubtful, Dirty, Lying
- Machine: System, Programmed, Assisted, Robot, Electronic, Automation
- Human: Manual, Individual, Physical, Unaided, Personal, Natural

The test will be administered on the same machine that the participants used for the study and will only record the times required to associate words and categories as well as the participant’s assigned number. No personal information will be recorded.

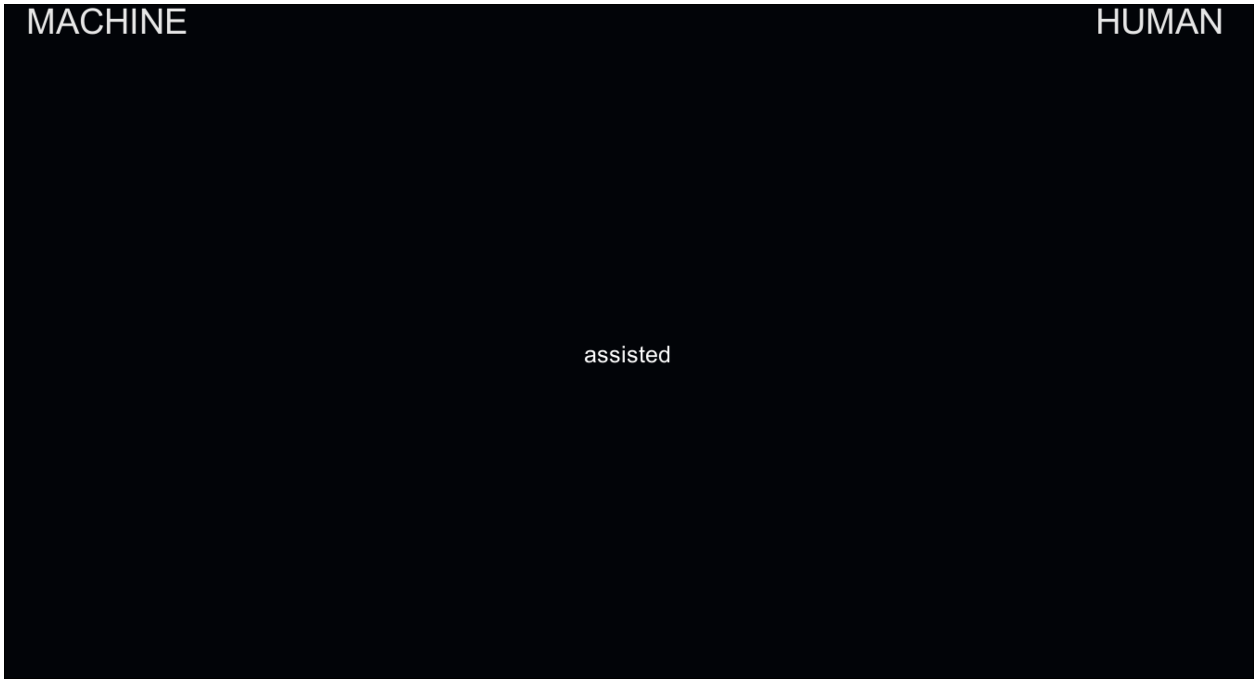
Below are sample images of the IAT.



Implicit Association Test:

This test requires you to sort words into corresponding categories. Categories appear at the top of the screen and words will appear in the center. Click the left side of the screen to sort words into the left category. Click the right side of the screen to sort words into the right category. Sometimes, two categories will appear on one side, and two on the other. Words that belong in either category on the left should be sorted to the left. Words that belong in either category on the right should be sorted to the right. No word will ever belong to more than one category. There will be seven total trials. If you have any questions please ask a researcher. When you are ready to continue, please check the box below and click the CONTINUE button

○ This box must be checked before you may continue.



General Trust Questionnaire

Below you will find a series of statements. Please read each statement carefully and respond to it by expressing the extent to which you believe the statement applies to you. For all items, a response from 1 to 7 is required. Circle the number that best reflects your 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. I feel that most automated systems would be employed for the user'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 user 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 user 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

10. In general, People really do care about the wellbeing of others

1 2 3 4 5 6 7

11. The typical person is sincerely concerned about the problems of others

1 2 3 4 5 6 7

12. 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

13. I believe that most professional people do a good job at work

1 2 3 4 5 6 7

14. Most professionals are very knowledgeable in their chosen field

1 2 3 4 5 6 7

15. A large majority of professional people is competent in their area of expertise

1 2 3 4 5 6 7

16. In general, most people keep their promises

1 2 3 4 5 6 7

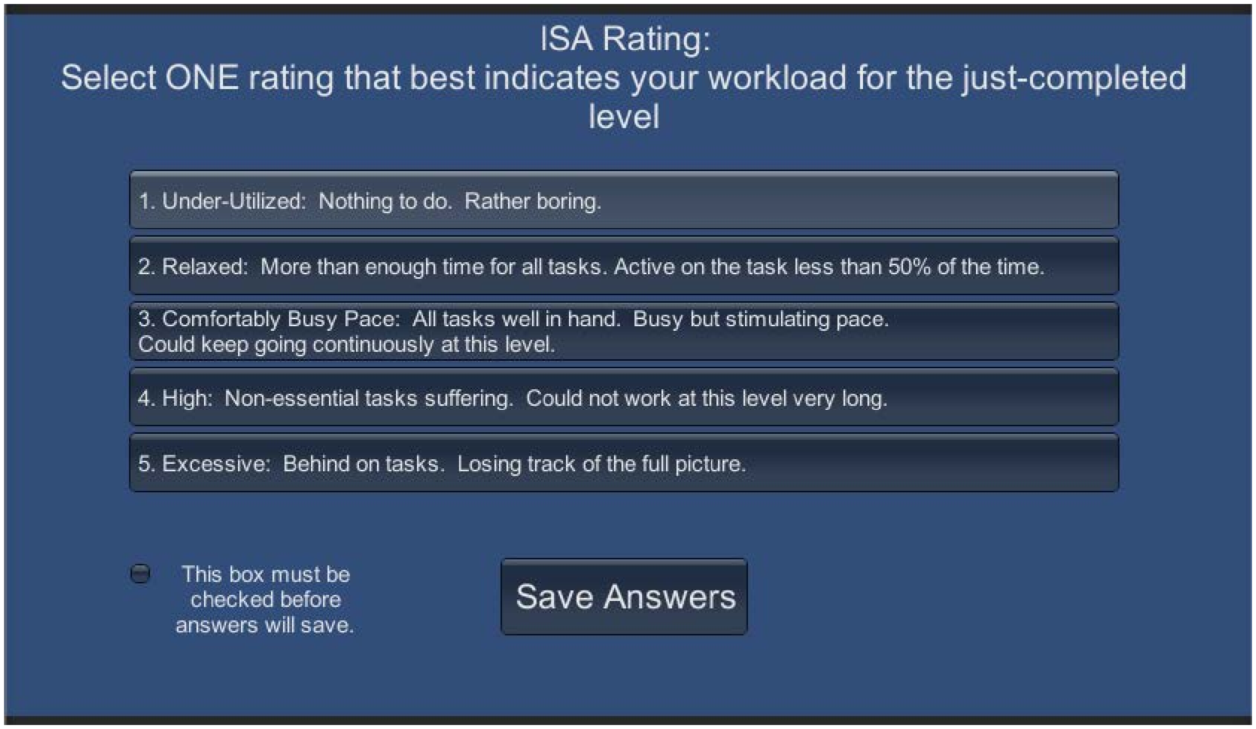
17. I think people generally try to back up their words with their actions

1 2 3 4 5 6 7

18. Most people are honest in their dealings with others

1 2 3 4 5 6 7

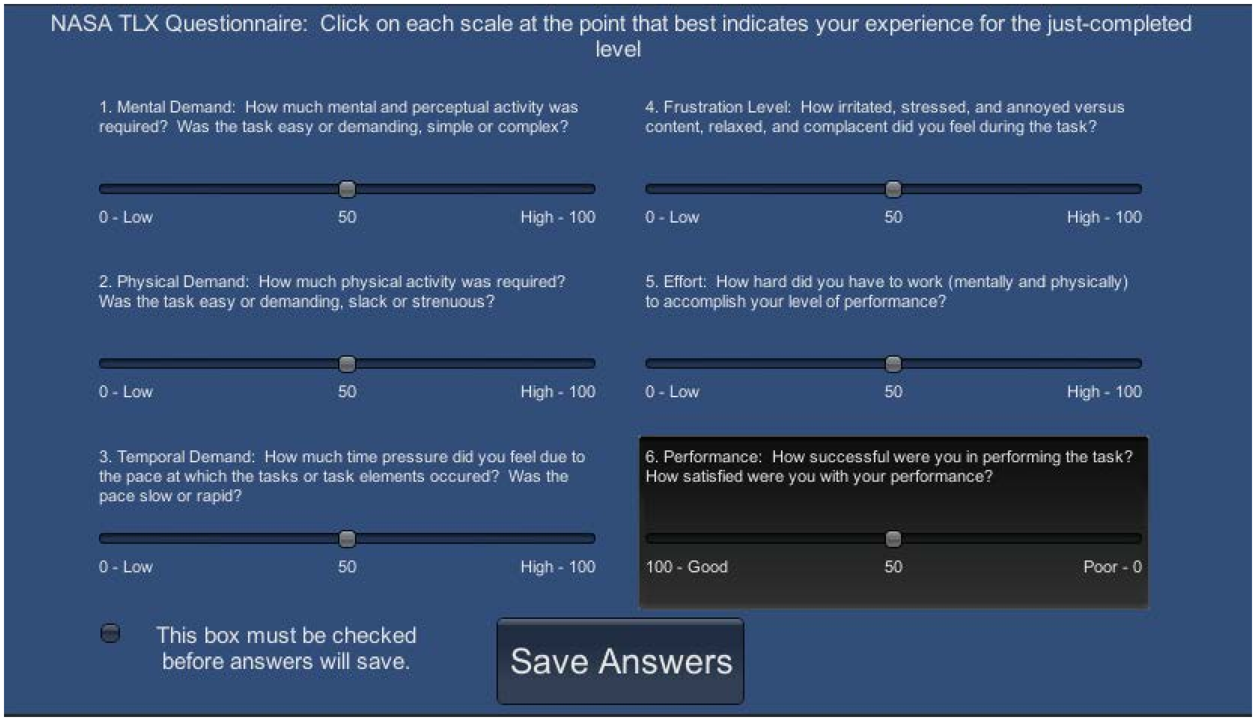
Instantaneous Self-Assessment



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.



NASA TLX Questionnaire: Click on each scale at the point that best indicates your experience for the just-completed level

1. Mental Demand: How much mental and perceptual activity was required? Was the task easy or demanding, simple or complex?

0 - Low 50 High - 100

2. Physical Demand: How much physical activity was required? Was the task easy or demanding, slack or strenuous?

0 - Low 50 High - 100

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

0 - Low 50 High - 100

4. Frustration Level: How irritated, stressed, and annoyed versus content, relaxed, and complacent did you feel during the task?

0 - Low 50 High - 100

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

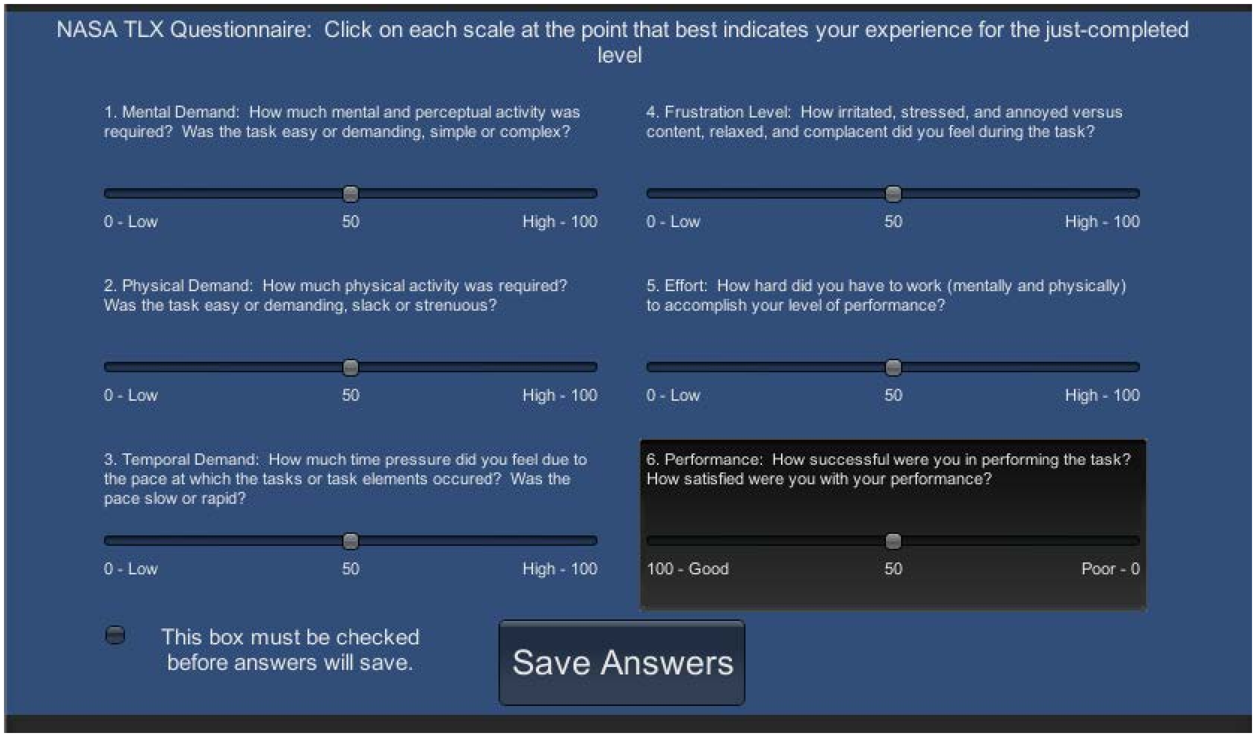
0 - Low 50 High - 100

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

100 - Good 50 Poor - 0

○ This box must be checked before answers will save.

NASA-TLX



NASA TLX Questionnaire: Click on each scale at the point that best indicates your experience for the just-completed level

1. Mental Demand: How much mental and perceptual activity was required? Was the task easy or demanding, simple or complex?

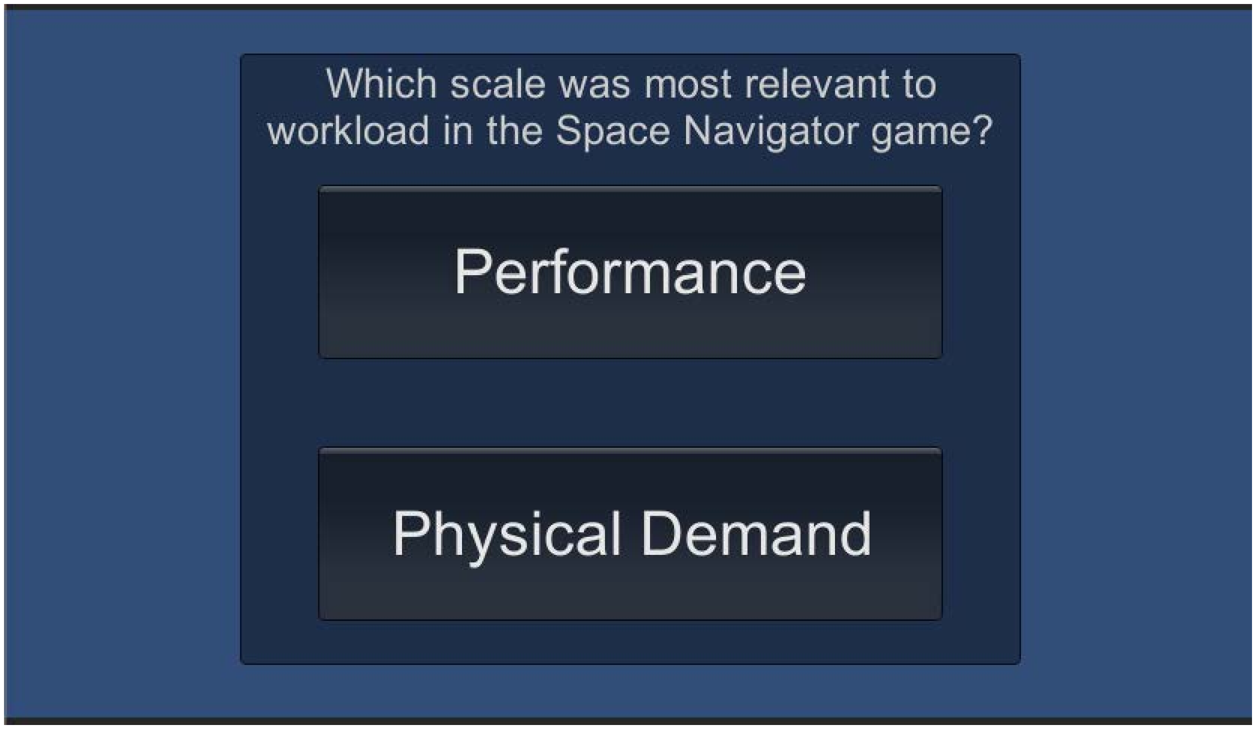
2. Physical Demand: How much physical activity was required? Was the task easy or demanding, slack or strenuous?

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

4. Frustration Level: How irritated, stressed, and annoyed versus content, relaxed, and complacent did you feel during the task?

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

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



Which scale was most relevant to workload in the Space Navigator game?

Reliability Rating Questionnaire

How reliable do you believe Space Navigator’s automation to be (in percent)? ___%

Communication Questionnaire

"Based on your experience playing the game today, please indicate the extent to which you agree with the following statement." Response options will be on a 7-point scale where 1=strongly disagree and 7=strongly agree.

1) I wanted to talk to the agent during the game.
2) I wanted the agent to talk to me during the game.
3) I wanted to know why the agent picked certain routes during the game.
4) I tried to figure out if the agent was employing a strategy during the game.

5) There was not enough time to talk with the agent during the game.

Space Navigator Trust Questionnaire

Below you will find a series of statements. Please read each statement carefully and respond to it by expressing the extent to which you believe the statement applies to you. For all items, a response from 1 to 7 is required. Adapted from Li 2008. Circle the number that best reflects your 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. 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

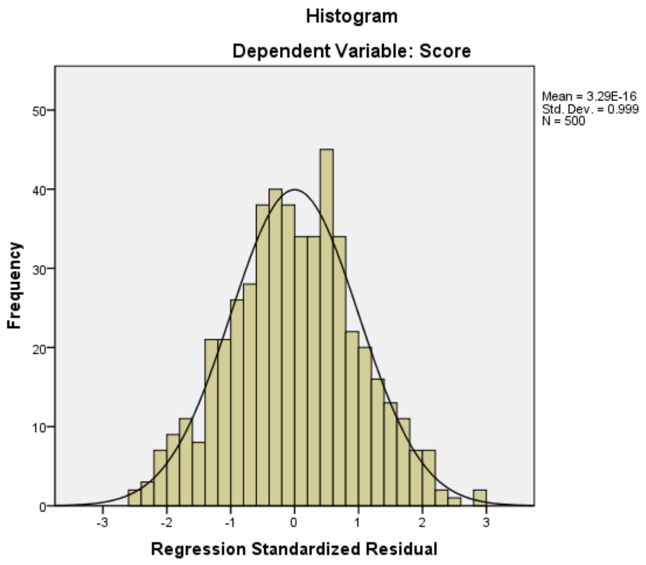
7. I was frustrated playing Space Navigator with automation (R)
   1 2 3 4 5 6 7

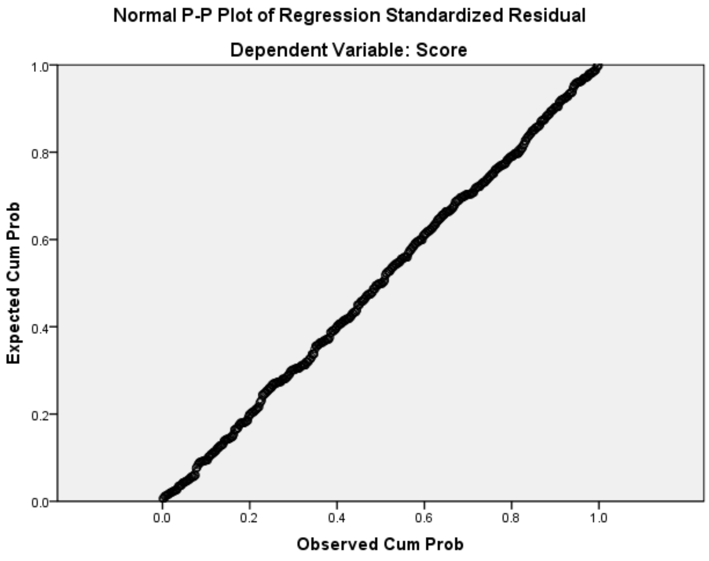
8. Space Navigator's automation made the game less frustrating
   1 2 3 4 5 6 7

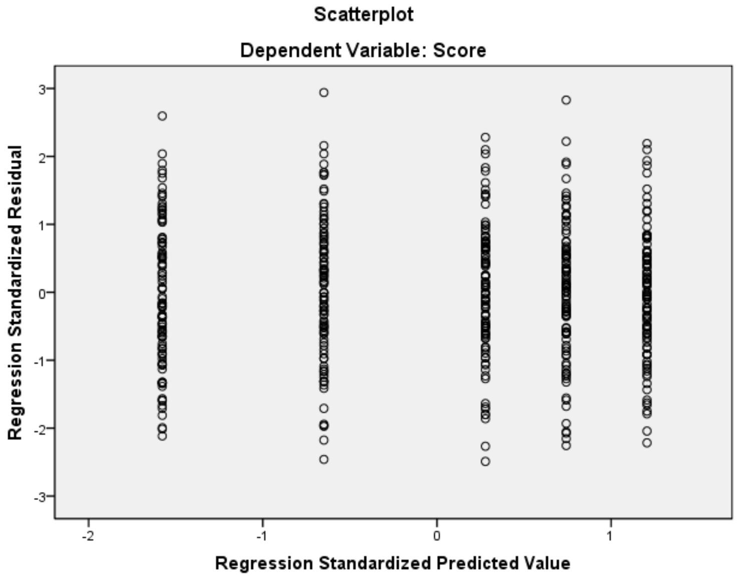
9. Space Navigator's automation made the game less frustrating
   1 2 3 4 5 6 7

Appendix 4: Validation of Linear Assumptions

Validation of Linear Assumptions for Chapter 3 Reduced Reliability Simulation

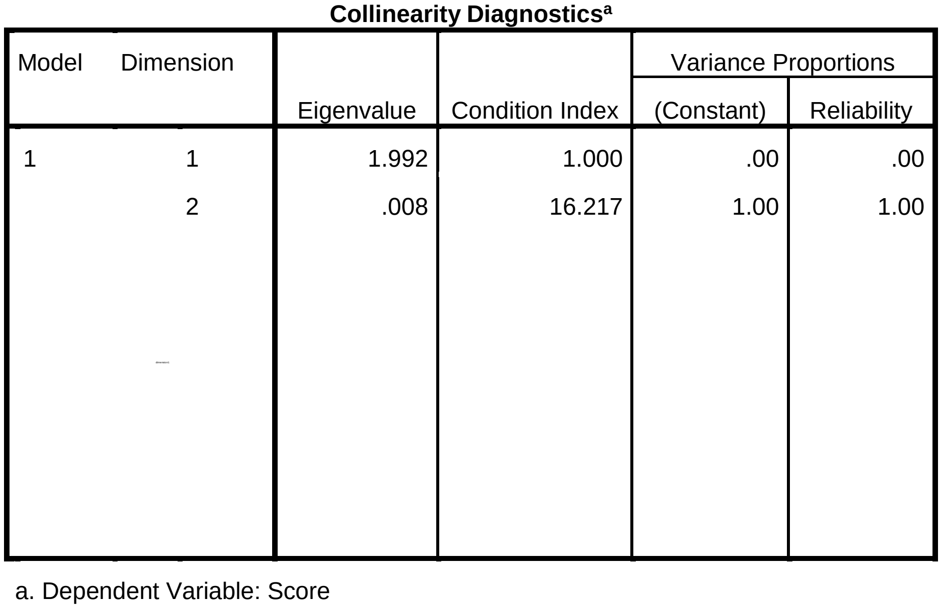






Correlations

		Score	Reliability
Pearson Correlation	Score	1.000	.796
	Reliability	.796	1.000
Sig. (1-tailed)	Score	.	.000
	Reliability	.000	.
N	Score	500	500
	Reliability	500	500

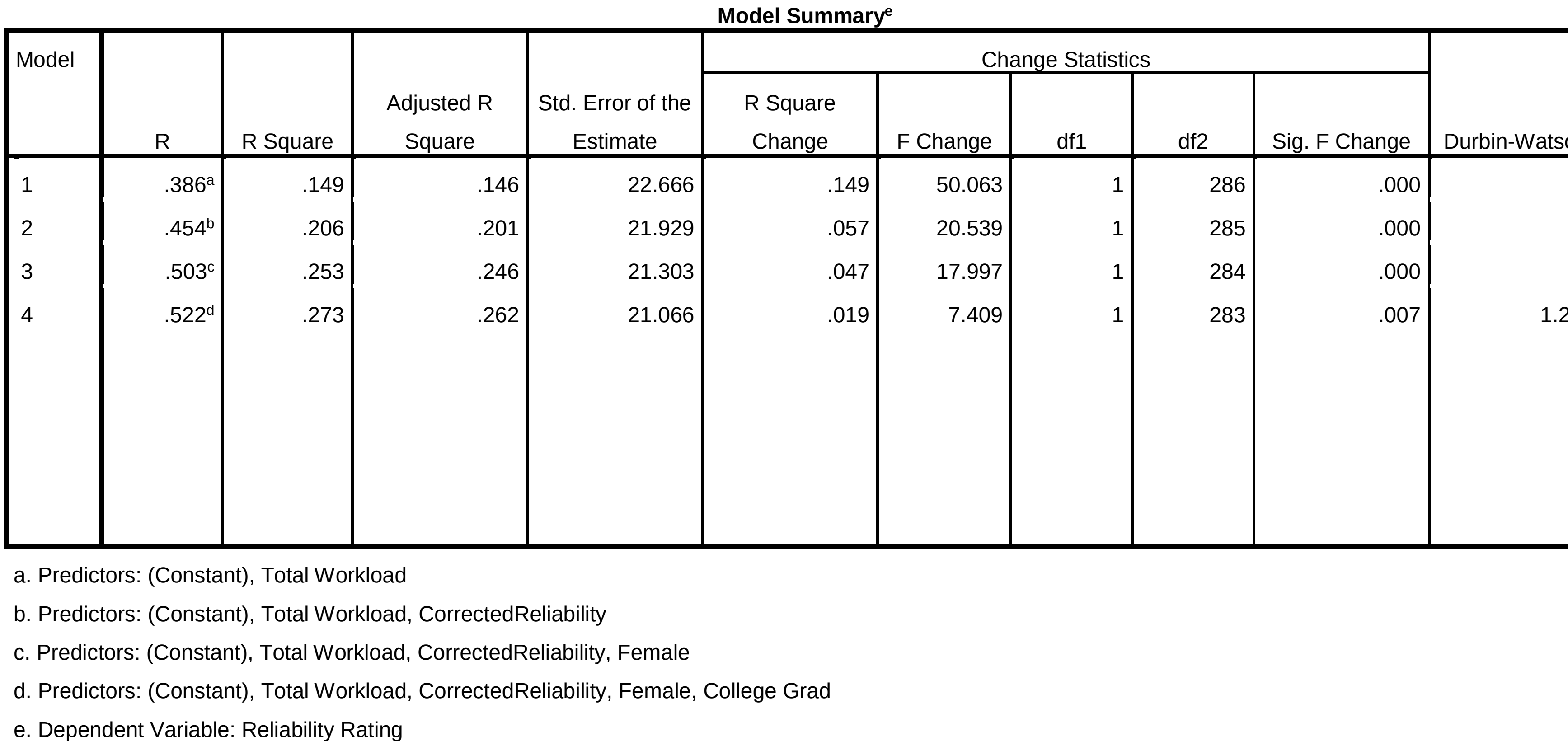


Collinearity Diagnostics[a]

Model	Dimension			Variance Proportions	
		Eigenvalue	Condition Index	(Constant)	Reliability
1	1	1.992	1.000	.00	.00
	2	.008	16.217	1.00	1.00

a. Dependent Variable: Score

Validation of Linear Assumptions for for Chapter 5 Reduced Reliability HITL Experiment



Model Summary[e]

Model					Change Statistics					
	R	R Square	Adjusted R Square	Std. Error of the Estimate	R Square Change	F Change	df1	df2	Sig. F Change	Durbin-Wats
1	.386[a]	.149	.146	22.666	.149	50.063	1	286	.000	
2	.454[b]	.206	.201	21.929	.057	20.539	1	285	.000	
3	.503[c]	.253	.246	21.303	.047	17.997	1	284	.000	
4	.522[d]	.273	.262	21.066	.019	7.409	1	283	.007	1.2

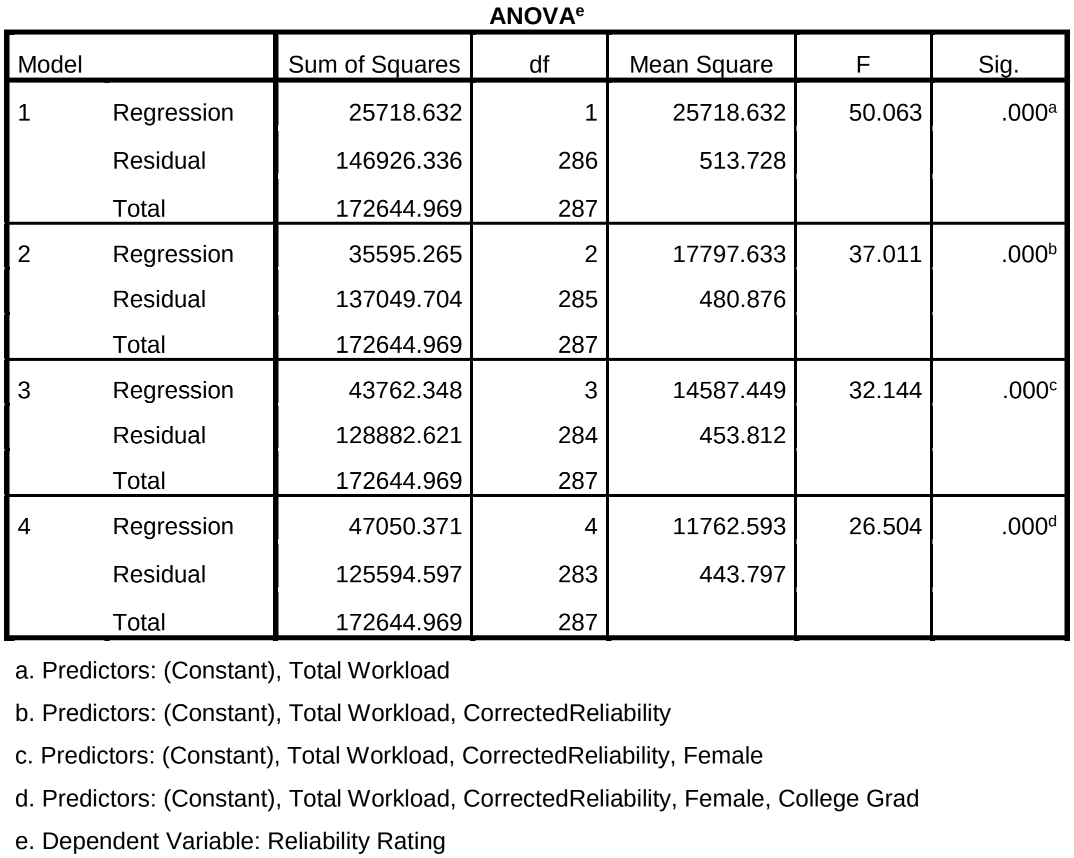
a. Predictors: (Constant), Total Workload

b. Predictors: (Constant), Total Workload, CorrectedReliability

c. Predictors: (Constant), Total Workload, CorrectedReliability, Female

d. Predictors: (Constant), Total Workload, CorrectedReliability, Female, College Grad

e. Dependent Variable: Reliability Rating



ANOVA[e]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	25718.632	1	25718.632	50.063	.000[a]
	Residual	146926.336	286	513.728		
	Total	172644.969	287			
2	Regression	35595.265	2	17797.633	37.011	.000[b]
	Residual	137049.704	285	480.876		
	Total	172644.969	287			
3	Regression	43762.348	3	14587.449	32.144	.000[c]
	Residual	128882.621	284	453.812		
	Total	172644.969	287			
4	Regression	47050.371	4	11762.593	26.504	.000[d]
	Residual	125594.597	283	443.797		
	Total	172644.969	287			

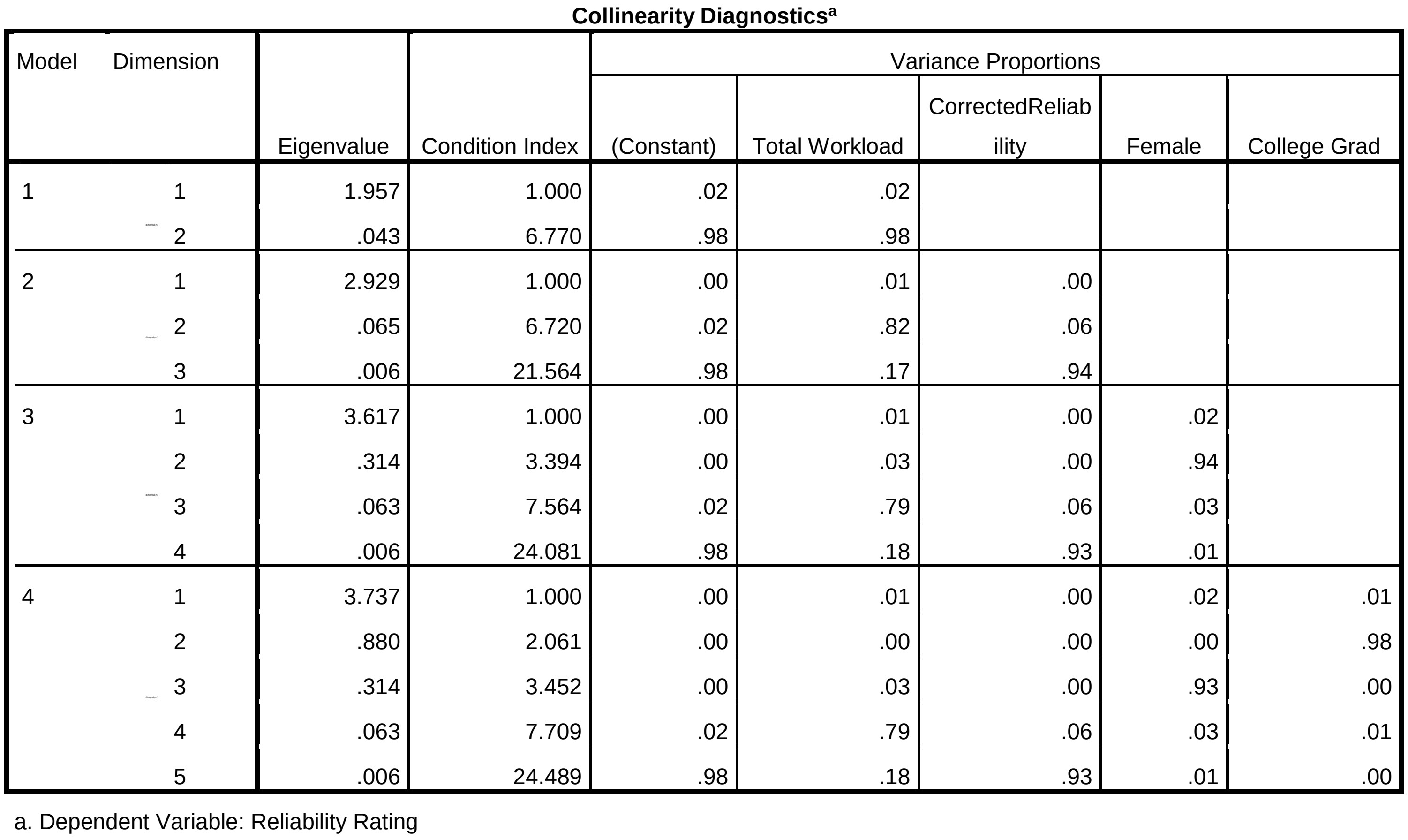
a. Predictors: (Constant), Total Workload

b. Predictors: (Constant), Total Workload, CorrectedReliability

c. Predictors: (Constant), Total Workload, CorrectedReliability, Female

d. Predictors: (Constant), Total Workload, CorrectedReliability, Female, College Grad

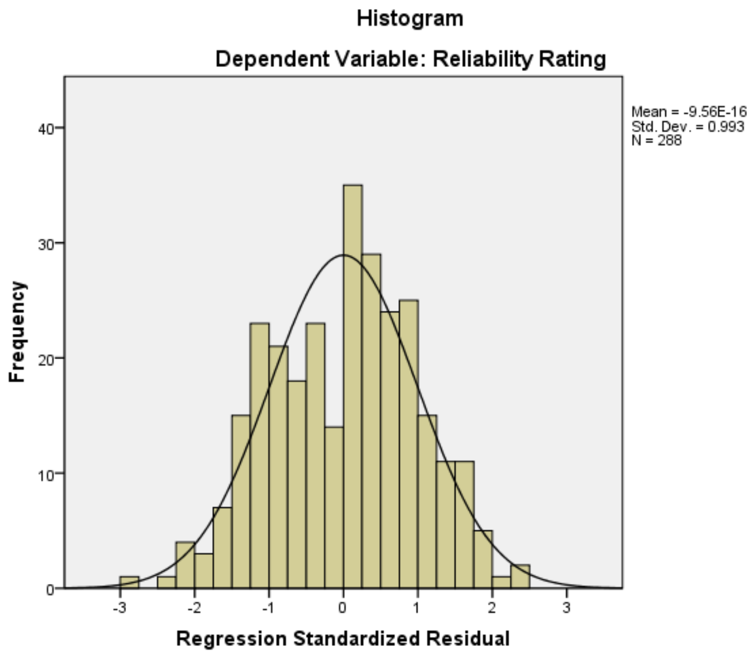
e. Dependent Variable: Reliability Rating

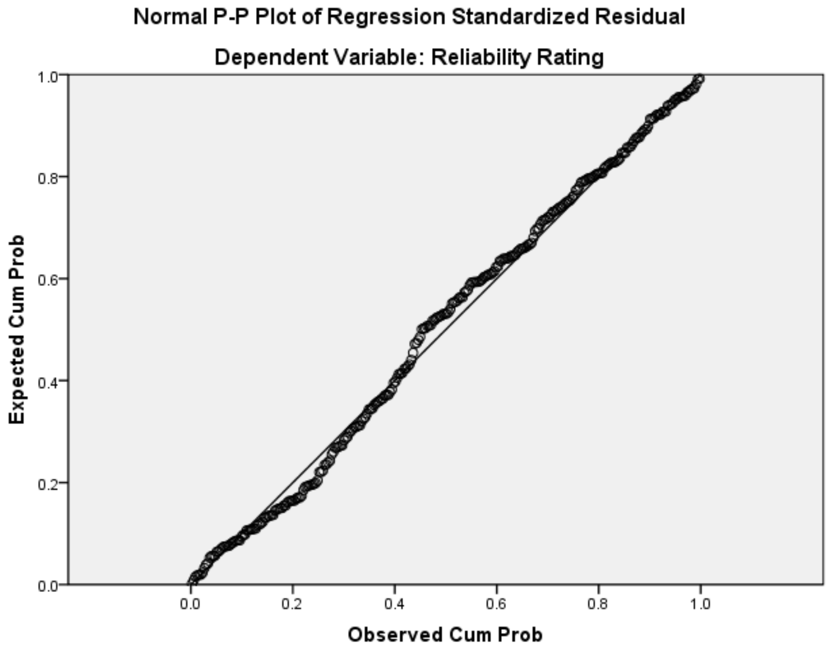


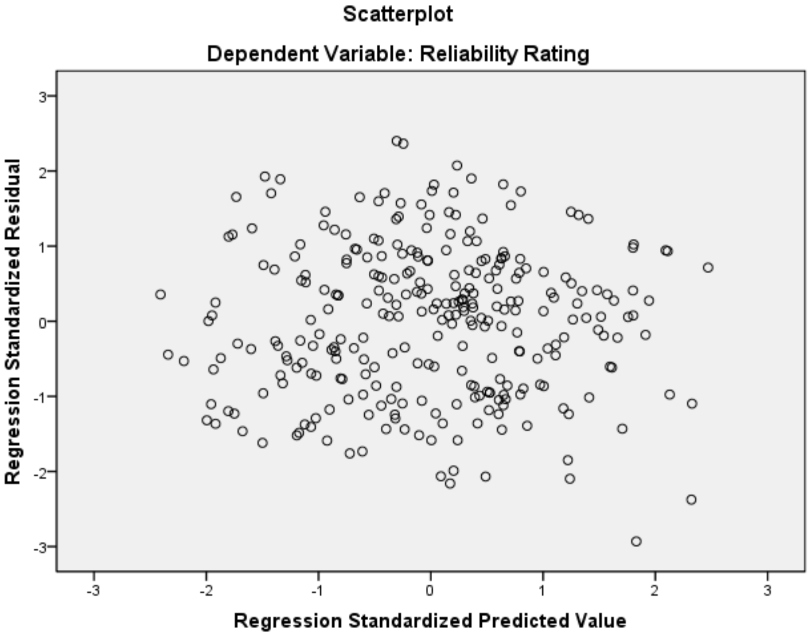
Collinearity Diagnostics[a]

Model	Dimension	Eigenvalue	Condition Index	Variance Proportions				
				(Constant)	Total Workload	CorrectedReliability	Female	College Grad
1	1	1.957	1.000	.02	.02			
	2	.043	6.770	.98	.98			
2	1	2.929	1.000	.00	.01	.00		
	2	.065	6.720	.02	.82	.06		
	3	.006	21.564	.98	.17	.94		
3	1	3.617	1.000	.00	.01	.00	.02	
	2	.314	3.394	.00	.03	.00	.94	
	3	.063	7.564	.02	.79	.06	.03	
	4	.006	24.081	.98	.18	.93	.01	
4	1	3.737	1.000	.00	.01	.00	.02	.01
	2	.880	2.061	.00	.00	.00	.00	.98
	3	.314	3.452	.00	.03	.00	.93	.00
	4	.063	7.709	.02	.79	.06	.03	.01
	5	.006	24.489	.98	.18	.93	.01	.00

a. Dependent Variable: Reliability Rating


Histogram
Dependent Variable: Reliability Rating
Mean = -9.56E-16
Std. Dev. = 0.993
N = 288
Frequency
0
10
20
30
40
-3
-2
-1
0
1
2
3
Regression Standardized Residual


Normal P-P Plot of Regression Standardized Residual
Dependent Variable: Reliability Rating
Expected Cum Prob
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.2
0.4
0.6
0.8
1.0
Observed Cum Prob


Scatterplot
Dependent Variable: Reliability Rating
Regression Standardized Residual
3
2
1
0
-1
-2
-3
-3
-2
-1
0
1
2
3
Regression Standardized Predicted Value

Appendix 5:Chi Squared Test for Gender Differences

Chi Squared Test for Gender Differences

	Male	Female	Total	Percentage	
Reduced	4	9	13	27%	
Flexible	4	4	8	17%	
High	10	17	27	56%	
Total	18	30	48		
Expected	2.70833	8.125			
	3	5			
	10.125	16.875			
	Chi Value	0.53632			

Appendix 6: Archetype ANOVA

Archetype ANOVA for Performance with Consolidated Data Set

ANOVA						
Source of Variation	*SS*	*df*	*MS*	*F*	*P-value*	*F crit*
Between Groups	349264	2	174632	0.146297	0.864311	3.204317
Within Groups	53715775	45	1193684			
Total	54065039	47				

Bibliography

Antifakos, S., Kern, N., Schiele, B., & Schwaninger, A. (2005). Towards improving trust in context-aware systems by displaying system confidence. *Proceedings of the 7th International Conference on Human Computer Interaction with Mobile Devices & Services - MobileHCI '05*, 9. http://doi.org/10.1145/1085777.1085780

Bainbridge, L. (1983). Ironies of automation. *Automatica*, *19*(6), 775–779. http://doi.org/10.1016/0005-1098(83)90046-8

Bakkes, S. C. J., Spronck, P. H. M., & van Lankveld, G. (2012). Player behavioural modelling for video games. *Entertainment Computing*, *3*(3), 71–79. http://doi.org/10.1016/j.entcom.2011.12.001

Bindewald, J. M., Miller, M. E., & Peterson, G. L. (2014). A function-to-task process model for adaptive automation system design. *International Journal of Human-Computer Studies*, *72*(12), 822–834. http://doi.org/10.1016/j.ijhcs.2014.07.004

Bindewald, J. M., Peterson, G. L., & Miller, M. E. (2015). Trajectory Generation with Player Modeling. In *Canadian Artificial Intelligence Conference*. Halifax, Nova Scotia, CA.

Bindewald, J. M., Peterson, G. L., & Miller, M. E. (2016). Clustering-Based Online Player Modeling. In T. Cazenave, S. Edelkamp, & M. Winands (Eds.), *International Joint Conference on Artificial Intelligence (IJCAI) - Computer Games Workshop*. New York, New York, USA.

Bonein, A., & Serra, D. (2009). Gender pairing bias in trustworthiness. *Journal of Socio-Economics*, *38*(5), 779–789. http://doi.org/10.1016/j.socec.2009.03.003

Bradshaw, J., Sierhuis, M., Acquisti, A., & Feltovich, P. (2003). Adjustable autonomy and human-agent teamwork in practice: An interim report on space applications. *Proceedings of the Seveth International Syposium on Artificial Intelligence Robotics and Automation in Space*, (October 2002), 7–9. http://doi.org/10.1007/978-1-4419-9198-0_11

Buchan, N. R., Croson, R. T. A., & Solnick, S. (2008). Trust and gender: An examination of behavior and beliefs in the Investment Game. *Journal of Economic Behavior and Organization*, *68*(3–4), 466–476. http://doi.org/10.1016/j.jebo.2007.10.006

Chaudhuri, A., Paichayontvijit, T., & Shen, L. (2013). Gender differences in trust and trustworthiness: Individuals, single sex and mixed sex groups. *Journal of Economic Psychology*, *34*, 181–194. http://doi.org/10.1016/j.joep.2012.09.013

Chen, J. Y. C., & Barnes, M. J. (2014). Human–Agent Teaming for Multirobot Control: A Review of Human Factors Issues. *IEEE Transactions on Human-Machine Systems*, *44*(1), 13–29. http://doi.org/10.1109/THMS.2013.2293535

Chen, J. Y. C., & Joyner, C. T. (2006). Individual Differences in Concurrent Performance of Gunner's and Robotic Operator's Tasks. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, *50*(8), 1759–1763. http://doi.org/10.1177/154193120805201922

Cosenzo, K., Parasuraman, R., Novak, A., & Barnes, M. (2006). *Implementation of automation for control of robotic systems*. Retrieved from http://oai.dtic.mil/oai/oai?verb=getRecord&metadataPrefix=html&identifier=ADA474882

Cowley, B. (2009). Player Profiling and Modelling in Computer and Video Games. *School of Computer and Information Engineering, Ph.D.*(October), 299.

de Visser, E. J., & Parasuraman, R. (2011). Adaptive Aiding of Human-Robot Teaming: Effects of Imperfect Automation on Performance, Trust, and Workload. *Journal of Cognitive Engineering and Decision Making*, *5*(2), 209–231.

Dixon, S. R., & Wickens, C. D. (2006). Automation reliability in unmanned aerial vehicle control: a reliance-compliance model of automation dependence in high workload. *Human Factors*, *48*(3), 474–486. http://doi.org/10.1518/001872006778606822

Dixon, S. R., Wickens, C. D., & Mccarley, J. S. (2006). On The Independence of Compliance and Reliance : Are

Automation False Alarms Worse Than Misses ?, *49*(March), 564–572. http://doi.org/10.1518/001872007X215656.Copyright

Dzindolet, M. T., Peterson, S. A., Pomranky, R. A., Pierce, L. G., & Beck, H. P. (2003). The role of trust in automation reliance. *International Journal of Human-Computer Studies*, *58*(6), 697–718. http://doi.org/10.1016/S1071-5819(03)00038-7

Endsley, M. R. (2015). Autonomous Horizons: System Autonomy in the Air Force - A Path to the Future, *1: Human-A*, 27.

Franklin, S., & Graesser, A. (2005). Is It an Agent , or Just a Program ? : A Taxonomy for Autonomous Agents.

Goldberg, L. R. (1993). The structure of phenotypic personality traits. *The American Psychologist*, *48*(1), 26–34. http://doi.org/10.1037/0003-066X.48.12.1302

Haselhuhn, M. P., Kennedy, J. A., Kray, L. J., Van Zant, A. B., & Schweitzer, M. E. (2015). Gender differences in trust dynamics: Women trust more than men following a trust violation. *Journal of Experimental Social Psychology*, *56*, 104–109. http://doi.org/10.1016/j.jesp.2014.09.007

Ho, G., Kiff, L. M., Plocher, T., & Haigh, K. Z. (2005). A Model of trust and reliance of automation technology for older users. *AAAI-2005 Fall Symposium:"Caring Machines: AI in Eldercare*.

Hoc, J.-M., & Lemoine, M.-P. (1998). Cognitive Evaluation of Human-Human and Human-Machine Cooperation Modes in Air Traffic Control. *International Journal of Aviation Psychology*, *8*(1), 1–32. Retrieved from http://search.ebscohost.com/login.aspx?direct=true&db=a9h&AN=7366343&site=ehost-live

Hoff, K. A., & Bashir, M. (2015). 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. http://doi.org/10.1177/0018720814547570

Hoffman, L. J., Lawson-Jenkins, K., & Blum, J. (2006). Trust beyond security: an expanded trust model. *Commun. ACM*, *49*(7), 94–101. http://doi.org/10.1145/1139922.1139924

Kaminski, P. (2012). Role of Autonomy in DoD Systems, (July), 125.

Klein, G., Woods, D. D., Bradshaw, J. M., Hoffman, R. R., & Feltovich, P. J. (2004). Ten challenges for making automation a “team player” in joint human-agent activity. *IEEE Intelligent Systems*, *19*(6), 91–95. http://doi.org/10.1109/MIS.2004.74

Koustanai, A., Cavallo, V., Delhomme, P., & Mas, A. (2012). Simulator Training With a Forward Collision Warning System: Effects on Driver-System Interactions and Driver Trust. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, *54*(5), 709–721. http://doi.org/10.1177/0018720812441796

Lacson, F. C., Wiegmann, D. A., & Madhavan, P. (2005). Effects of Attribute and Goal Framing on Automation Reliance and Compliance. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, *49*(3), 482–486. http://doi.org/10.1177/154193120504900357

Lee, J. D.& See, K. A. (2004). Trust in Automation : Designing for Appropriate Reliance, *46*(1), 50–80.

Li, X., Hess, T. J., & Valacich, J. S. (2008). 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. http://doi.org/10.1016/j.jsis.2008.01.001

Mahfouf, M., Zhang, J., Linkens, D. A., Nassef, A., Nickel, P., Hockey, G. R. J., & Roberts, A. C. (2007). Adaptive Fuzzy Approaches to Modelling Operator Functional States in a Human-Machine Process Control System. In *2007 IEEE International Fuzzy Systems Conference* (pp. 1–6). IEEE. http://doi.org/10.1109/FUZZY.2007.4295371

Maynard, P. W., & Rantanen, E. M. (2005). Pilot Dependance on Imperfect Diagnostic Automation in Simulated UAV Flights: An Attentional Visual Scanning Analysis. *13th International Symposium on Aviation Psychology, Daytona, OH.*, 1–6.

McCarley, J. S., Wiegmann, D. a., Wickens, C. D., & Kramer, a. F. (2003). Effects of Age on Utilization and Perceived Reliability of an Automated Decision-Making aid for Luggage Screening. *Proceedings of the*

Human Factors and Ergonomics Society Annual Meeting, *47*(3), 340–343. http://doi.org/10.1177/154193120304700319

McKnight, D. H., Choudhury, V., & Kacmar, C. (2002). Developing and validating trust measures for e-commerce: An integrative typology. *Information Systems Research*, *13*(3), 334–359. http://doi.org/10.1287/isre.13.3.334.81

Merritt, S. M., Heimbaugh, H., LaChapell, J., & Lee, D. (2013). I Trust It, but I Don't Know Why: Effects of Implicit Attitudes Toward Automation on Trust in an Automated System. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, *55*(3), 520–534. http://doi.org/10.1177/0018720812465081

Mitchell, D. K. (2009). Workload Analysis of the Crew of the Abrams V2 SEP : Phase I Baseline IMPRINT Model. *Engineering*, (September).

Moray, N., Inagaki, T., & Itoh, M. (2000). Adaptive automation, trust, and self-confidence in fault management of time-critical tasks.

Nourbakhsh, I. R., Sycara, K., Koes, M., Yong, M., Lewis, M., & Burion, S. (2005). Human-robot teaming for Search and Rescue. *IEEE Pervasive Computing*, *4*(1), 72–77. http://doi.org/10.1109/MPRV.2005.13

Parasuraman, R., & Miller, C. a. (2004). Trust and etiquette in high-criticality automated systems. *Communications of the ACM*, *47*(4), 51. http://doi.org/10.1145/975817.975844

Parasuraman, R., & Riley, V. (1997). Humans and Automation: Use, Misuse, Disuse, Abuse. *Human Factors: The Journal of the Human Factors and Ergonomics Society*. http://doi.org/10.1518/001872097778543886

Parasuraman, R., Sheridan, T. B., & Wickens, C. D. (2000). 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*, *30*(3), 286–97. http://doi.org/10.1109/3468.844354

Ross, J. M., Szalma, J. L., Hancock, P. a., Barnett, J. S., & Taylor, G. (2008). 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. http://doi.org/10.1177/154193120805201908

Rovira, E., McGarry, K., & Parasuraman, R. (2007). Effects of imperfect automation on decision making in a simulated command and control task. *Human Factors*, *49*(1), 76–87. http://doi.org/10.1518/001872007779598082

Rovira, E., & Parasuraman, R. (2010). Transitioning to future air traffic management: effects of imperfect automation on controller attention and performance. *Human Factors*, *52*(3), 411–425. http://doi.org/10.1177/0018720810375692

Rowe, A., Spriggs, S., & Hooper, D. (2015). Fusion: A Framework For Human Interaction With Flexible-Adaptive Automation Across Multiple Unmanned Systems. *International Symposium on Aircraft Psychology*, *18*(18th Annual), 464–470.

Scerbo, M. W. (1996). Theoretical Perspectives on Adaptive Automation. *Automation and Human Performance: Theory and Applications*.

Sheridan, T. B. (1984). Supervisory contol of remote manipulators, vehicles and dynamic process: Experiments in command and display aiding, 49–137.

Sheridan, T. B., & Verplank, W. L. (1978). Human and Computer Control of Undersea Teleoperators. *ManMachine Systems Lab Department of Mechanical Engineering MIT Grant N0001477C0256*.

Singh, A. L., Tiwari, T., & Singh, I. L. (2009). Effects of automation reliability and training on automation-induced complacency and perceived mental workload. *Journal of the Indian Academy of Applied Psychology*, *35*(spec iss), 9–22. Retrieved from http://ovidsp.ovid.com/ovidweb.cgi?T=JS&CSC=Y&NEWS=N&PAGE=fulltext&D=psyc6&AN=2011-19929-001%5Cnhttp://sirius.library.unsw.edu.au:9003/sfx_local?sid=OVID:psycdb&id=pmid:&id=doi:&issn=0019-

4247&isbn=&volume=35&issue=spec+iss&spage=9&pages=9-22&date=2009&titl

Spronck, P., Balemans, I., & Lankveld, G. Van. (2012). Player Profiling with Fallout 3. *Proceedings, The Eighth AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, 179–184.

Urlings, P., Sioutis, C., Tweedale, J., Ichalkaranje, N., & Jain, L. (2006). A future framework for interfacing BDI agents in a real-time teaming environment. *Journal of Network and Computer Applications*, *29*(2–3), 105–123. http://doi.org/10.1016/j.jnca.2004.10.005

Verberne, F. M. F., Ham, J., & Midden, C. J. H. (2012). Trust in Smart Systems: Sharing Driving Goals and Giving Information to Increase Trustworthiness and Acceptability of Smart Systems in Cars. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, *54*(5), 799–810. Retrieved from http://hfs.sagepub.com/content/54/5/799.abstract

Wickens, C. D., & Dixon, S. R. (2007). The benefits of imperfect diagnostic automation: a synthesis of the literature. *Theoretical Issues in Ergonomics Science*. http://doi.org/10.1080/14639220500370105

Wooldridge, M., & Jennings, N. R. (1995). Intelligent agents: theory and practice. *The Knowledge Engineering Review*. http://doi.org/10.1017/S0269888900008122

Wright, J. L., Chen, J. Y. C., Quinn, S. A., & Barnes, M. J. (2013). The Effects of Level of Autonomy on Human-Agent Teaming for Multi-Robot Control and Local Security Maintenance, (November).

Yannakakis, G. N., Spronck, P., Loiacono, D., & André, E. (2013). Player Modeling. *Dagstuhl Follow-Ups*, *6*, 59. http://doi.org/10.4230/DFU.Vol6.12191.45

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 074-0188

1. REPORT DATE *(DD-MM-YYYY)* 23-03-2017	**2. REPORT TYPE** Master’s Thesis	**3. DATES COVERED** *(From – To)* Aug 2015 – March 2017

TITLE AND SUBTITLE

Analysis of Human and Agent Characteristics on Human-Agent Team Performance and Trust

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Hillesheim, Anthony J., 2Lt, USAF

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S)

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way, Building 640
WPAFB OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT-ENV-MS-17-M-194

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Katie Wisecarver
katie.wisecarver@us.af.mil
703-426-9544
Air Force Office of Scientific Research
875 N. Randolph
Arlington Virginia, 22203

10. SPONSOR/MONITOR’S ACRONYM(S)

AFOSR

11. SPONSOR/MONITOR’S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

DISTRIBUTION STATEMENT A. APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES

This material is declared a work of the U.S. Government and is not subject to copyright protection in the United States.

14. ABSTRACT

The human-agent team represents a new construct in how the United States Department of Defense is orchestrating mission planning and mission accomplishment. In order for mission planning and accomplishment to be successful, several requirements must be met: a firm understanding of human trust in automated agents, how human and automated agent characteristics influence human-agent team performance, and how humans behave. This thesis applies a combination of modeling techniques and human experimentation to understand the concepts aforementioned. The modeling techniques used include static modeling in SysML activity diagrams and dynamic modeling of both human and agent behavior in IMPRINT. Additionally, this research consisted of human experimentation in a dynamic, event-driven, teaming environment known as Space Navigator. Both the modeling and the experimenting depict that the agent’s reliability has a significant effect upon the human-agent team performance. Additionally, this research found that the age, gender, and education level of the human user has a relationship with the perceived trust the user has in the agent. Finally, it was found that patterns of compliant human behavior, archetypes, can be created to classify human users.

15. SUBJECT TERMS

Trust, Automation, Agent, Human-Agent Teaming, Reliability, Performance

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON** Maj Christina Rusnock, AFIT/ENV
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			**19b. TELEPHONE NUMBER** *(Include area code)*
U	U	U	UU	158	(937) 255-3636, ext 4611 (christina.rusnock@afit.edu)