Effects of Changing Reliability on Trust of Robot Systems

Munjal Desai[1], Mikhail Medvedev[1], Marynel Vázquez[2], Sean McSheehy[1],
Sofia Gadea-Omelchenko[3], Christian Bruggeman[2], Aaron Steinfeld[2], and Holly Yanco[1]

[1]University of Massachusetts Lowell, Lowell, MA 01854
{mdesai, mmedvede, smcsheeh, holly}@cs.uml.edu

[2]Carnegie Mellon University, Pittsburgh, PA 15213
{marynel, steinfeld}@cmu.edu, cgbrugge@andrew.cmu.edu

[3]University of Pittsburgh, Pittsburgh, PA 15260
sog9@pitt.edu

ABSTRACT

Prior work in human-autonomy interaction has focused on plant systems that operate in highly structured environments. In contrast, many human-robot interaction (HRI) tasks are dynamic and unstructured, occurring in the open world. It is our belief that methods developed for the measurement and modeling of trust in traditional automation need alteration in order to be useful for HRI. Therefore, it is important to characterize the factors in HRI that influence trust. This study focused on the influence of changing autonomy reliability. Participants experienced a set of challenging robot handling scenarios that forced autonomy use and kept them focused on autonomy performance. The counterbalanced experiment included scenarios with different low reliability windows so that we could examine how drops in reliability altered trust and use of autonomy. Drops in reliability were shown to affect trust, the frequency and timing of autonomy mode switching, as well as participants' self-assessments of performance. A regression analysis on a number of robot, personal, and scenario factors revealed that participants tie trust more strongly to their own actions rather than robot performance.

Categories and Subject Descriptors

H.1.2 [**User/Machine Systems**]: Human factors

General Terms

Experimentation, performance

Keywords

Trust, automation, experiments

1. INTRODUCTION

One of the key factors for the acceptance and safe deployment of robots is the degree to which a user trusts the robot. If trust can be modeled, the model can be used to design robot interfaces and behaviors that foster appropriate levels of trust. Mobile robots, especially those not designed for social interaction, are particularly interesting since they are likely to be task-oriented and therefore used for time-dependent activities, capable of damaging objects and hurting people, and unable to express their intent to bystanders (e.g., [18]).

In their survey of trust and automation research, Lee and See define trust as "the attitude that an agent will help achieve an individual's goals in a situation characterized by uncertainty and vulnerability" [13]. A person's level of trust of an automated system is a key factor that influences their use of that system: an inappropriate level of trust may result in the misuse or disuse of automation [13].

The broad range of automation research provides a context for examining issues of trust with robots. However, there are a number of factors that limit how well this work generalizes to the robotics domain. For example, studies of automated systems have tended to utilize systems such as autopilots, flight management systems, vision systems for target or obstacle identification, and factory control systems [13]. Participants interact with a simulated system, which allows experimenters to inject errors and observe how participants' levels of trust of and use of the system change as a result. The systems used in these experiments generally do not have a physical embodiment and do not interact with the physical world. Furthermore, these automated systems tend to be designed for rigid tasks; that is, each system performs only one very specific type of task. In contrast, many HRI tasks are dynamic and unstructured, occurring in the open world. For example, an operator supervising a remote, autonomous mobile robot must contend with noisy sensing, incomplete perception, unpredictable environments, and bystanders.

In this work, we have investigated how changing the robot's reliability influences people's use of robot autonomy and their trust in the robot system through experiments with participants operating a real robot. The experiment, described in detail below, was designed to have a high workload so that the participants would need to use the autonomous capabilities of the robot in order to complete the task in time and to be able to complete the secondary task. We hypothesized that people would trust a robot system less when its reliability in autonomous mode decreased, switching to a manual mode. We wanted to determine how long it would take participants to switch back to autonomous mode after the robot's reliability recovered. We further hypothesized that the timing of the reliability decreases would influence trust in the robot's autonomy.

HRI'12, March 5–8, 2012, Boston, Massachusetts, USA.

2. PRIOR WORK

Parasuraman and Riley define automation as "the execution by a machine agent (usually a computer) of a function that was previously carried out by a human" [19]. Automation has traditionally been employed in systems that are complicated, tedious, or time critical, but it has also been used for economic reasons [19]. When automation was first introduced in the 1930's, its use was limited to large industries; however, at the present, automation can be found in many places, from home appliances to nuclear power plants.

Automation has always had weaknesses: namely, it has only been effective in well-structured and controlled environments and continues to remain so. To avoid catastrophic failures in safety critical systems due to either flaws or limitations of automation, an operator must be present at all times to take control of the system. Situations of this kind in which a human operator is working with an automated system are referred to as "human-in-the-loop control." While utilizing a human operator may be beneficial in certain situations, addressing the inadequacies of automation for the human-in-the-loop control creates a different set of problems. When an operator is added to the system, improving the overall system performance requires more than simply optimizing operator performance and, separately, optimizing automation performance. The interaction between the two needs to be considered as well.

Researchers have shown that trust influences operators' use of automation (e.g., [3, 4, 10, 12, 16, 21]): the more operators trust automation, the more they tend to use it. Moreover, if an operator trusts his own abilities more than those of the automated system, he tends to choose manual control. A user's trust in his own capabilities is most often referred to as "self-confidence."

For several decades, researchers in the human automation interaction field have examined the factors which influence people's trust of automated systems (see [13] for an overview) and how this level of trust, in turn, affects the way in which people use, misuse, or disuse automation. Specifically, Dzindolet et al. [4] demonstrated the impact of system performance on user trust. The results of their study indicated that while users initially placed trust in a decision aid and agreed with its suggestions, as users observed the system making errors they would distrust even a generally high-performing aid unless provided with reasons as to why the errors had occurred. Providing these reasons increased trust in the automated aid, even when the aid performed poorly.

Additional factors contributing to a user's trust of an automated system include the recency of errors made by the system [22], the user's prior knowledge about the system's performance [22], the user's knowledge of the capabilities of the system [22], the user's expectations of the system's performance [22], the usability of the interface [1], and situation awareness gained through using a system [14]. Atoyan et al. [1] found that interface design plays an important role in influencing users' trust in automation.

Some studies have reported a lag between changes in trust and self-confidence and an actual change in allocation strategy; this lag is referred to as "inertia" [12]. When the user changes the allocation strategy, the performance of the system inevitably changes. For the feedback loop to close, the user needs to observe this change. Depending on the system, there might be a significant time delay before the user observes the change in performance [4].

To date, there has been little work examining issues of trust with non-social robots, although some work has been conducted involving simulated robots. Dassonville et al. [2] conducted a study in which participants used a joystick to control a simulated PUMA arm. Errors were introduced into the simulation, and participants were asked to rate the reliability, performance, and predictability of the joystick's behavior (as well as how difficult it was to make such ratings). The results of the study were consistent with prior work in autonomous systems that suggest that the user's self-confidence is a significant factor which influences use of such systems.

Freedy et al. [6] examined trust in the context of mixed-initiative command and control systems using the MITPAS (Mixed-Initiative Team Performance Assessment System) Simulation Environment. The researchers constructed a quantitative measure of trust by assuming that people use a rational decision model such that "trust behavior is reflected by the expected value of the decisions whether to allocate control to the robots on the basis of past robot behavior and the risk associated with autonomous robot control" [6]. Participants assumed the role of a controller of an unmanned ground vehicle (UGV). The UGV autonomously targeted and fired, but participants were instructed to take control of the UGV if its behaviors would lead to a time delay or a failure. The experimenters varied the competency of the UGV's firing behavior and recorded participants' choices to override the UGV. The results suggested that if participants could gauge whether the UGV was very competent or incompetent, they adjusted their behavior accordingly. This adjustment implied that the participants trusted the system to continue to maintain the same level of competence. It was more difficult for users to adjust their behavior when the system showed indeterminate competence. More work needs to be done to determine how these results would generalize to physical robots in the real world.

While relatively little work has been done investigating trust in robots, there is a large body of research on trust in different types of technologies. For example, Song et al. [23] developed a neural network-based trust model for understanding users' acceptance of recommendations from a system of heterogeneous agents. Another agent-related trust model was developed by Rehak et al. [20], who used fuzzy numbers to represent trust in cooperating ubiquitous devices. McKnight et al. [15] developed a trust model to understand users' acceptance of a website offering legal advice. Because we are interested in developing a model of trust for human-robot interaction, we have restricted the scope to trust models that were developed for other technology domains.

Riley hypothesized a general model for trust in automation, including how different factors influence each other and ultimately the operator's reliance on automation [21]. We believe that trust models developed for traditional automation need alteration in order to be useful for HRI. Like most, Riley's model does not consider some factors that are relevant to robots such as interface usability, proximity to robot (co-located or remote-located), situation awareness, and dynamics of the operating environment. To advance the field, a systematic study of the factors that could influence trust in HRI is necessary to build trust models in this domain. To this effect we conducted a study with a real robot with dynamic workload, complex task, and variable reliability.

3. METHODOLOGY

To determine how changing reliability impacts a person's use of autonomy and trust in a robot system, we conducted the same set of experiments at University of Massachusetts Lowell (UML) and Carnegie Mellon University (CMU).[1] Twelve participants were recruited at both sites. The average age of the participants at UML was 23.6 years ($SD = 6.1$) and at CMU was 30.6 years ($SD = 14.2$). All 24 participants were classified as novice robot users as none of them had any prior experience controlling remote robots.

[1]Unless explicitly mentioned, all of the parameters were identical between the two sites.

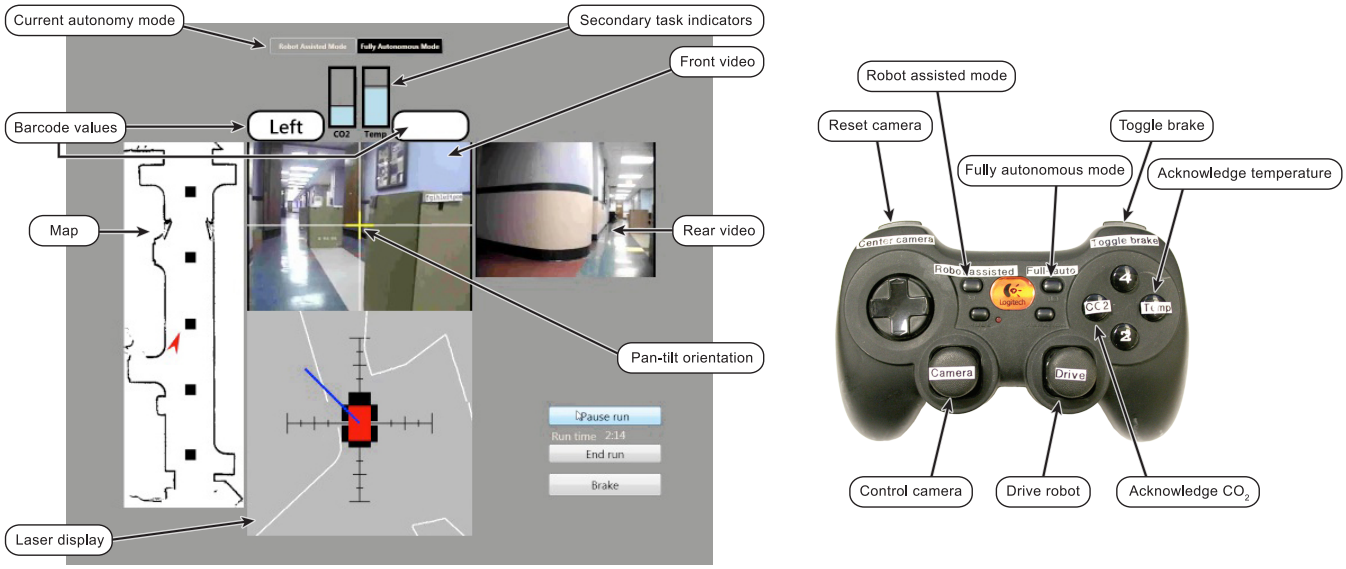

Figure 1: The interface and the gamepad used to control the robot.

3.1 Robot system

Two iRobot ATRV-JR robots were used for this experiment, one at UML and the other at CMU. There was a camera mounted on the front of the robots on a Directed Perception PTU-D46-17 pan-tilt unit and another camera was mounted on the rear. For distance sensing, a SICK LMS200 was used on the front and a Hokuyo URG-04LX laser was mounted on the back. The robots had computers with similar capabilities and ran the same code base.

Figure 1 shows the user interface (UI) used to control the robots. The video from the front camera was displayed in the middle, the video from the back camera was displayed on the top right (mirrored to simulate a rear view mirror in a car). The distance information from both lasers was displayed on the bottom around a graphic of the robot. The map of the course with the pose of the robot was displayed on the left. Using the gamepad shown in Figure 1, participants were able to drive the robot, control the pan tilt unit for the front camera, select the autonomy mode, turn the brakes on or off, recenter the camera, and acknowledge the secondary tasks.

3.2 Tasks

The participants were asked to drive the robot as quickly as they could along a specified path, searching for victims, not hitting objects in the course, and responding to the secondary tasks. To create additional workload, simulated sensors for CO_2 and temperature were used. The participants were not told that the sensors were not real. They were also informed that the robot's performance was not influenced in any way by changes in temperature and carbon dioxide. The values from the sensors were displayed on the UI (Figure 1), which the participants were asked to monitor. Participants were asked to acknowledge high CO_2 and temperature values by pressing the corresponding buttons on the gamepad. The values were considered high when their values were above the threshold lines on the secondary task indicators (Figure 1); values over the threshold were indicated by changing the color of the bars from light blue to red to assist the participants in recognizing the change. The level of workload was varied by changing the frequency with which the values crossed the threshold. The simulated sampling rate for the sensors was kept steady.

3.3 Test course

Figure 2 shows the course used at UML. The course at CMU had the same length, layout for the boxes, and driving clearances. Both courses were set in hallways with little foot traffic. The courses were approximately 18 meters (60 feet) long and had 5 obstacles (boxes) placed about 2.75 meters (9 feet) from each other. The width of the course at UML was 2.43 meters (8 feet), and the width of the course at CMU was 1.98 meters (6.5 feet). The discrepancy in the hallway widths was compensated by using 61 cm (24 inch) wide boxes at UML and 15.2 cm (6 inch) wide boxes at CMU. The clearance on either side of the boxes was 0.9 meters (3 feet), and the robots were 0.66 meters (26 inches) wide.

The start and end positions were the same for each run. For each run, the participants were asked to follow a set path. We designed five different paths based on the following criteria:

- The length of each path must be the same (∼61 meters (200 feet)).
- The number of u-turns in a path must be the same (3).
- The number of transitions from the left side of the course to the right and vice versa must be the same (3).

As the maps were similar in difficulty and length, we did not counterbalance paths for the participants. Instead, paths were selected based on a randomly generated sequence. A sample path is shown in green in Figure 2.

Text labels were placed on top of the boxes to indicate the path ahead. Since the boxes at UML were wide, similar labels were placed on both edges of the face as shown in Figure 2 to make it easy for the participants to read the labels as they went past the boxes. The labels indicated 'left,' 'right,' or 'uturn.' The directions were padded with additional characters to prevent the participants from recognizing the label without reading them. Figure 2 shows the two types of labels that were used. The labels with white background (referred to as white labels) were to be followed for the first half of the entire length and the labels with black background (referred to as black labels) for the second half. The transition from following the white labels to black labels was indicated to the participants via the UI. When the participants were supposed to follow the black

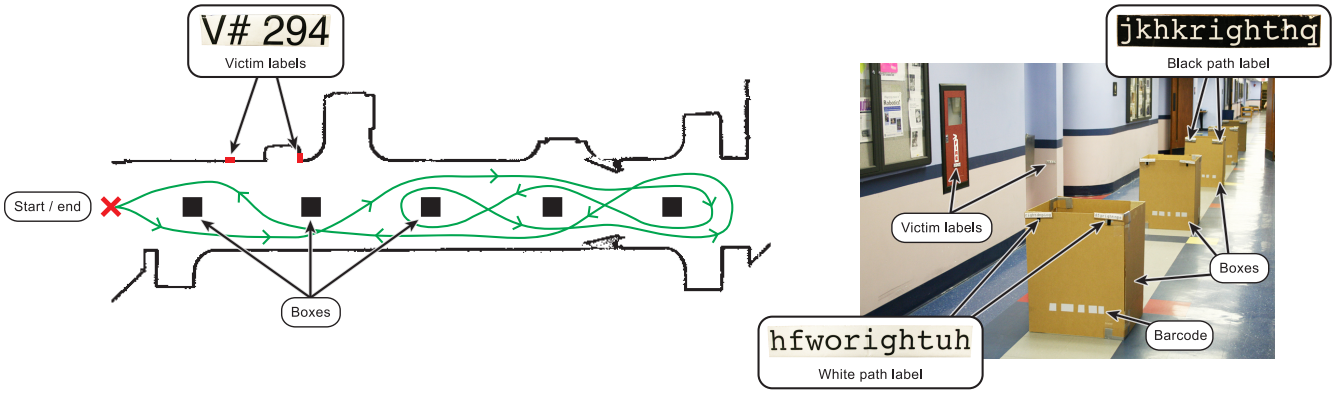

Figure 2: The top down view of the course used for the experiments (left) and the photo looking down the hallway (right).

labels the background for the barcode values (shown in Figure 1) would turn black. Two sets of labels were necessary to prevent the participants from driving in an infinite loop.

The boxes also had barcodes made from retro-reflective tapes that the robot could read (Figure 2). The robot would display the contents of the bar code on the UI. However, the paths for each run were hard coded because the barcodes could not be consistently read by the robot. While the barcodes were not used by the robot, the participants were told that the robot read the barcodes to determine the path ahead, just like they read the labels. Based on a constant video compression rate, sampling resolution, and the font size, the labels could be read from about 1 meter (3 feet) away. The robot was set to simulate reading the label from approximately the same distance. The participants were informed that, at times, the robot might make a mistake reading the barcodes and that they should ensure that the direction read by the robot is correct. The participants were also told that if the robot did make a mistake in reading the barcode, it would proceed to pass the next box on the incorrect side, resulting in the participant being charged with an error on their score (see below).

The course also had four simulated victims. These victims were represented using text labels like the one shown in Figure 2. The victim tags were placed on the walls of the course between 0.7 meters (2.5 feet) and 1.8 meters (6 feet) from the floor. The victim locations were paired with the paths and were never placed in the same location during the participant's five runs. While there was a number associated with each victim, the participants were told to ignore the number while reporting the victims. Whenever the participants found a new victim, they were told to inform the experimenter that they had found a victim. They were explicitly instructed to only report victims they have not reported before.

3.4 Autonomy modes

The participants could operate the robot in one of two autonomy modes: robot assisted mode or fully autonomous mode. The participants were instructed that they were free to select either mode and could switch as many times as they wanted. They were also informed that there were no benefits or penalties for selecting either mode. When each run started, no autonomy mode was selected by default, thereby requiring the participants to make a selection. The maximum speed at which the robot would move was the same in both modes (approximately 0.125 meters (0.41 feet) per second). These configurations ensured that both autonomy modes were similar from a performance standpoint.

In fully autonomous mode, the robot ignored the participant's input and followed the hard coded path. The obstacle avoidance algorithm ensured that the robot never hit any object on the course. In robot assisted mode, the participant had a significant portion of the control and could easily override the robot's movements. The robot would provide its desired velocity vector based on the path it was supposed to follow. The robot's desired vectors were calculated the same way in both autonomy modes and were displayed on the UI to show the participant the robot's desired direction.

3.5 Compensation

Using higher levels of automation reduces workload and hence is desirable, especially under heavy workload from other tasks. To prevent participants from using high levels of autonomy all the time, regardless of the autonomous system's performance, it is typical to introduce some amount of risk. Hence, in line with similar studies (e.g., [5, 11, 21]), the compensation was based in part on the overall performance. The participants at UML could select a gift card to a local restaurant or Amazon.com, and the participants at CMU received cash.[2] The maximum amount that the participants at both sites could earn was $30. Base compensation was $10. Another $10 was based on the average performance of 5 runs. The last $10 was based on the average time needed to compete the 5 runs, provided that the performance on those runs was high enough.

The performance for each run was based on multiple factors, with different weights for each of those factors determined before the experiments were run. The participants were told there was a significant penalty for passing a box on the incorrect side, regardless of the autonomy mode. If the participants passed a box on the wrong side, they were heavily penalized (20 points per box). In addition to the loss of score, participants were told that time would be added based on the the number of wrong turns they took, but the specific penalties were not revealed. For the first box passed on the wrong side, no additional time was added, to allow people to realize that the reliability of the system had dropped. For the second incorrect pass, 60 seconds were added, with an additional 120 seconds for the third and an additional 240 for the fourth, continuing with an exponential increase. Finding the victims was also an important task, so 10 points were lost for each victim missed. Equation 1 was used to calculate the score for each run.

$$\begin{aligned} Score = {} & 100 - 20 \times numIncorrectPasses \\ & - 10 \times numVictimsMissed - 5 \times numPushes \\ & - 2 \times numBumps - numScrapes - secondaryTaskScore/2 \end{aligned} \quad (1)$$

The scoring formula was not revealed to participants, although they were told the factors that would influence their score. The score for each run was bounded between 0 and 100. If the score was 50

[2]The means of compensation were institutional limitations.

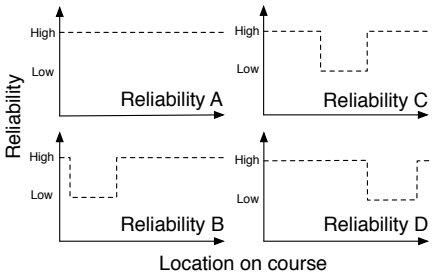

Figure 3: The different reliability configurations.

or more, the participants were eligible for a time bonus; if they had completed the runs in under 11:45 minutes on average, they would receive an additional $10. If they had a score of 50 or more and took between 11:45 and 15 minutes per run on average, they would receive a $5 bonus. Participants were told about this interdependence between score and time, which was designed to prevent participants from quickly running through the course, ignoring the tasks, while also providing a significant motivation to perform the task quickly.

At the end of each run, its score was calculated and the participants were informed about the amount of compensation that could be received based only on that run. At the end of five runs, the average compensation was calculated and given to the participant.

3.6 Questionnaires

There were three sets of questionnaires. The pre-experiment questionnaire was administered after the participants signed the consent form; it was focused on demographics (i.e., age, familiarity with technology similar to robot user interfaces, tendency towards risky behavior, etc). The post-run questionnaire was administered immediately after each run; participants were asked to rate their performance, the robot's performance, and the likelihood of not receiving their milestone payment. Participants were also asked to fill out previously validated trust surveys in their unaltered form, referred to in this document as Muir [17] and Jian [9], and a TLX questionnaire after each run. After the last post-run questionnaire, the post-experiment questionnaire was administered, which included questions about wanting to use the robot again and its performance.

3.7 Procedure

After participants signed the informed consent form, they were provided an overview of the robot system and the task to be performed. Then, participants were asked to drive the robot through the trial course in fully autonomous mode. The experimenter guided the participant during this process, by explaining the controls and helping with tasks if necessary. The trial course was half the length of the test course. Once participants finished, they were asked to drive the robot again through the same course in robot assisted mode. Since there are multiple tasks that participants need to perform, we decided to first show them the fully autonomous mode, as that would be a less overwhelming experience. Once the participants finished the second trial run, they were asked to fill out the post-run questionnaire. While the data from this questionnaire was not used, it allowed participants to familiarize themselves with it and also helped to reinforce some of the aspects of the run that they would need to remember. The participants were also told that they could take a break whenever they wanted.

After the two trial runs, the participants were asked to drive the robot for five more runs. In each run, a different map was used. During these runs the reliability of robot autonomy was either held high throughout the run or was changed. Figure 3 shows the four different reliability configurations. The changes in reliability were triggered when the robot passed specific points in the course.

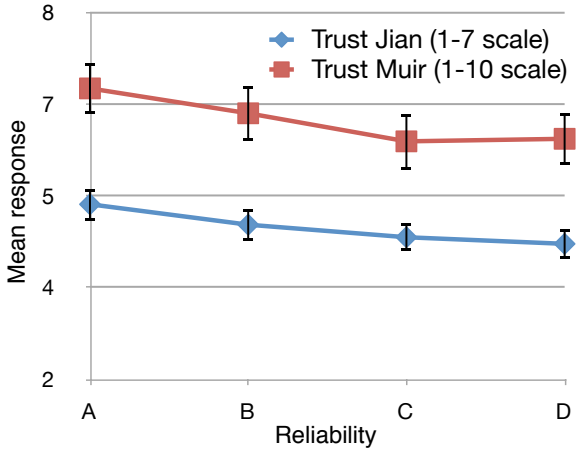

Figure 4: Impact of reliability on trust.

These locations were equal in length and there were no overlaps. For all four patterns, the robot always started with high reliability. The length of each low reliability span was about one third the length of the entire course. Using different dynamic patterns for reliability allowed us to investigate how participants responded to a drop in reliability at different stages and the changes' influence on control allocation. Every participant started with a baseline run under full reliability (Reliability A in Figure 3). Then, the four reliability profiles were counterbalanced for the remaining four runs. Instead of a single long run, we elected to use shorter duration runs because we wanted to directly compare failures at different periods. Failure during longer runs can be investigated in the future, once the important reliability patterns can be identified.

Our goal was to investigate how trust is initially formed; hence we recruited novice participants. We expect expert users would interact differently with the system, as they would already have a mental model of it and its performance.

4. RESULTS AND DISCUSSION

While 12 of the participants were run at CMU and 12 at UML, there were consistent behaviors across the sites related to reliability and autonomy switching, so this data is reported in aggregate. There were some site differences in terms of the trust scales used, which we discuss below.

Unless noted, data from the practice and baseline runs were not included in the analyses. We checked for practice effects (run order) and map effects and did not find any issues. This suggests the counterbalancing and map designs were adequate.

4.1 Positivity bias

We found that 13 participants started all four runs by switching into the fully autonomous mode and 17 participants started run 1 in the fully autonomous mode. Of the 96 total runs, participants initially opted to start in full autonomy for 65 of them. The breakdown for the individual runs was: $run_1 = 17$, $run_2 = 15$, $run_3 = 17$, and $run_4 = 16$, which is remarkably stable. The participants' willingness to initially trust the robot indicates the possibility of a positivity bias. These findings are consistent with the findings of Dzindolet et al. [4] where they found that given little experience or information about an automated decision aid, people were willing to trust it.

4.2 Effect of reliability changes on trust

The two trust survey methods (Muir, Jian) were highly correlated with each other ($r = 0.84$, $p < 0.0001$) suggesting either can be used

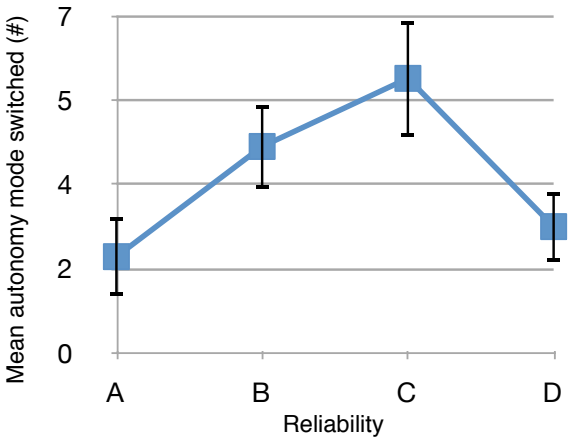

Figure 5: Impact of reliability on mode switching.

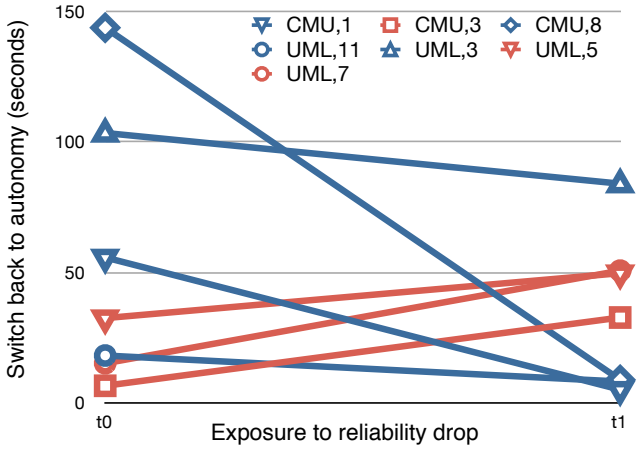

Figure 6: Autonomy return by exposure to low reliability. t0 represents the time when reliability drops and t1 represents the time when reliability increases.

for such experiments in the future. In our analysis in later sections, we have elected to standardize on Muir due to its shorter length.

The Muir and Jian post-run trust surveys were examined with REML (REstricted or REsidual Maximum Likelihood) [8] on the effects of Site (CMU, UML), Reliability (A, B, C, D), and Participants as a random effect and nested by Site. In both cases, there were significant differences for Site and Reliability, but not the interaction. UML trust responses were significantly higher than CMU for Muir, $F(3) = 9.7$ ($p < 0.01$), and Jian, $F(3) = 9.7$ ($p < 0.01$). Student's t post hoc tests of Reliability on Muir, $F(3) = 2.6$ ($p = 0.059$), and Jian, $F(3) = 3.0$ ($p < 0.05$), showed reliability A as being significantly higher than C and D for both metrics (Figure 4). These nearly identical results for Muir and Jian reinforce the earlier finding that using just one approach is appropriate in the future.

These results mean that trust is highest in high reliability runs (A); slightly reduced in runs with low reliability at the beginning of the run and high at the end (B); and more reduced for runs where reliability was low in the middle or end of the runs (C & D). This result means timing is important for trust – drops in reliability after a period of good performance are more harmful than early failures. Whether this is due to memory recency or a breakage in the participant's mental model of robot performance is uncertain.

The influence of Site on trust survey results is likely due to UML's population being slightly younger (mean of 7 years younger) and more predisposed towards risky behavior (0.66 higher on a set of 7-point self-rating scales). Significance tests for both demographic features were close, but not statistically significantly different. However, their combined effect may have produced this Site effect.

4.3 Changing autonomy levels

To obtain a high-level view, we performed a REML analysis of how many times participants switched the autonomy level within a run on the effects of Site (CMU, UML), Reliability (A, B, C, D), and Participants as a random effect and nested by Site. This analysis resulted in a significant difference only for Reliability, $F(3) = 4.7$ ($p < 0.01$), where a Student's t post hoc revealed participants switched considerably more within reliability C, as compared to A and D (Figure 5). Likewise, B was higher than A.

Of the 24 participants, five did not switch autonomy levels during any of their runs, regardless of the reliability profiles. Two of these participants stayed in robot assisted mode for all of their runs, two stayed in the autonomous mode, and one participant used robot assisted mode for all but one run. Participants were binned into three behavior groups: FullAuto, Mixed, and MostlyRobotAssisted. Sample sizes for these groups were imbalanced and too small for statistical analysis (2, 19, and 3, respectively), but there were several clear trends. The MostlyRobotAssisted group run times were noticeably slower, and the FullAuto rated their own performance low in comparison to the other two groups. There was a general trend of lower trust on the Muir questions as autonomy use increased across the three groups (see Familiarity bias below).

Of the 19 participants in the Mixed behavior category, nine did not change their autonomy level during the baseline run, which was held constant at high reliability (3 in robot assisted mode, 6 in autonomous). In the second run with high reliability, eleven did not change their autonomy level (1 in robot assisted, 10 in autonomous). Seven of these participants overlapped, meaning that during the high reliability runs, all but six participants did not change their autonomy mode in at least one of those runs.

In contrast, during the runs with changing reliability, all Mixed participants switched autonomy modes in at least one of the other three variable reliability conditions. Also, 14 of the 19 participants switched autonomy modes in all three of the variable reliability conditions (B, C, & D). This data indicates that participants recognized they were operating under dynamic reliability and adjusted their control allocation accordingly. It also indicates that participants recognized the risk of decreased compensation and tried to optimize the allocation strategy to obtain maximum compensation. To further investigate how the participants used autonomy, we analyzed the participants' behavior during periods of low reliability.

4.4 Use of autonomy during periods of unreliability

To examine behavior during low reliability, we focused on the scenario where participants entered a low reliability window during autonomy use. This window corresponded to the point at which reliability decreased (t0) to when it increased (t1). By definition, runs with reliability A were not included, as reliability did not decrease during those runs. For the 17 participants who switched during this window, the mean use of autonomy during low reliability was 30 percent.

A total of 15 participants switched from autonomy to robot assisted mode after t0 and from robot assisted mode to autonomy after t1. This behavior was constrained to reliability conditions B and C

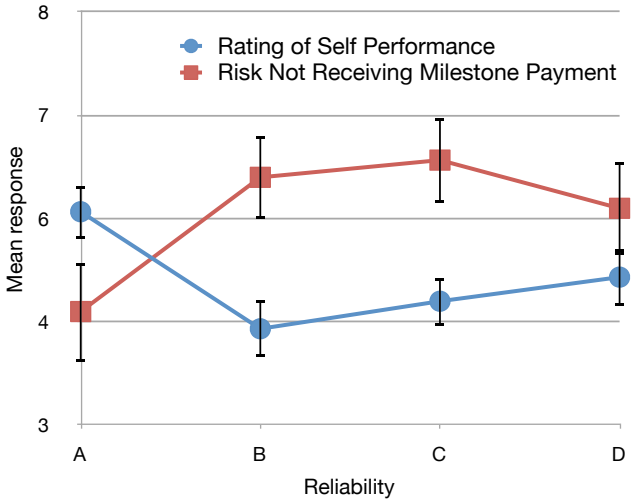

Figure 7: Impact of reliability on self assessment ratings.

(7 and 8 participants respectively); we conjecture that participants did not have enough time to recover from the reliability drop in D, where the drop occurred near the end of the run. Within this group, the mean switching time after the reliability drop at t0 was 16.6 seconds ($SD = 13.1$). The return to autonomy after reliability improved at t1 occurred a mean of 39.0 seconds later ($SD = 41.0$). A one tailed t-test, $t(14) = 2.04$ ($p < 0.05$) confirmed that participants waited longer to switch back to autonomy than switching away from autonomy. While only marginally statistically significant (one tailed, t-Ratio $t(13) = 1.73$ ($p < 0.1$), there were strong indications that participants switched away from autonomy at t0 twice as slowly for C than B (means 22 and 11 seconds respectively). However, there were no differences between C and B for switching back to autonomy at t1.

Of these 15 participants, seven returned to the fully autonomous mode at t1 for both the B and C conditions. Four of these seven switched back to autonomy faster on their second exposure to a reliability change, while the rest switched back more slowly (Figure 6). This results suggests that repeated exposure to changing reliability impacts the speed at which people switch back to autonomy, although we do not posses enough evidence to determine what causes this behavior.

4.5 Effect of self-assessment on trust

ANOVA analysis of participant ratings of robot performance across reliability levels were inconclusive, $F(3, 92) = 1.09$ ($p = 0.36$). However, participants did respond differently for ratings of their own performance, $F(3, 92) = 3.4$ ($p < 0.05$), and were marginally significant on the risk of not receiving a milestone payment $F(3, 92) = 2.2$ ($p < 0.1$). Student's t post hoc analyses showed a higher rating of self performance and better odds of receiving the milestone payment for reliability A, as compared to C and B, in both measures (Figure 7). Assessment of self performance was also sensitive when examining trust, with a significant correlation to Muir ($r = 0.43$, $p < 0.0001$).

As has been seen in some of our prior experiments, participants were pretty accurate in their assessment of milestone performance. Ratings of risk of not being paid the extra money were inversely correlated with actual payment ($r = -0.58, p < 0.01$).

4.6 Familiarity bias

The protocol was intentionally designed to promote use of autonomy. As expected, higher use of autonomy was correlated with better performance on finding more victims ($r = 0.30, p < 0.01$) and faster route completion time ($r = -0.51, p < 0.0001$). These results suggest that general use of autonomy had a perceptible, beneficial impact on the task.

As mentioned, prior work shows that increased use of autonomy with positive performance outcomes leads to higher trust (e.g., [11]). However, the Muir post-run trust ratings and the percentage of time spent in full autonomy were inversely correlated ($r = -0.20, p < 0.05$). This fact, combined with the results above, suggest that overall familiarity is less powerful than scenario factors.

Table 1: Backwards stepwise regression results for Muir trust ratings

Effect	Estimate	p
Cognitive load (TLX)	-0.33	< 0.01
Victims found	-1.58	< 0.01
Payment (performance)	-0.22	< 0.01
Tendencies towards risky behavior	0.65	< 0.01
Risk of not receiving milestone payment	0.28	< 0.05
Participant age	-0.05	< 0.1
Self performance rating	0.50	< 0.1
Robot performance rating	removed	x
Experience with robot-like UIs	removed	x
Autonomy switches	removed	x
Technology demographics	removed	x
Secondary task performance	removed	x
Percent autonomy	removed	x
Map time	removed	x

4.7 Predicting trust

An important question for human-robot interaction is whether trust can be predicted. To examine this question, Muir trust ratings were examined in the context of cognitive load (TLX), victims found, secondary task performance, payment (i.e., overall performance), number of switches between autonomy and robot assisted modes, a collection of demographic features, and the three post-run assessment ratings. A backwards stepwise regression on these independent measures accurately predicted Muir ratings ($R^2 = 0.84$). Significance results showed that higher trust was predicted by low cognitive load, poor victim performance, lower payment, lower expected payment, high ratings of self performance, younger age, and high risk tendencies (Table 1). While these factors strongly predict trust, it should not preclude other factors from being investigated in the future. Note that autonomy switching, secondary task performance, ratings of robot performance, and percentage of time using full autonomy do not predict trust. These results suggest that trust is heavily tied to factors with semantic association to risk and personal feelings about performance, rather than robot performance. This result is contrary to what Hancock et al. [7] found. We speculate that the difference is due to the fact that our methodology involved controlling a remote robot making it difficult to gauge the robot's actual performance, as was the case in the studies examined by Hancock et al. [7].

5. CONCLUSIONS

Our goal of creating an experience that was challenging, would force autonomy use, and keep participants focused on performance clearly succeded. This design increases the realism of the experience and is analogous to scenarios where robot users will be forced to

use autonomy deliberately, carefully, and with supervision (e.g., assistive robotics, bomb disposal).

We hypothesized that people would trust a robot system less when its reliability in autonomous mode decreased, as evidenced by switching to a manual mode. Trust was affected by drops in reliability. We further hypothesized that the timing of the reliability decreases would influence trust in the robot's autonomy. This hypothesis was true, especially when reliability drops occurred late (D) or in the middle (C) of runs. Reliability patterns also led to different mode switching behavior. Reliability drops in the middle of the run (C) led to sharp increases in the number of mode switches and switches away from autonomy were twice as slow as those seen for early drops in reliability (B).

We also wanted to determine how long it would take participants to switch back to autonomous mode after the robot's reliability increased. As is typical with trust, users switched away from autonomy during a reliability drop much faster than returning to autonomy after a reliability increase. However, the reliability pattern did not impact the speed at which users returned to autonomy when reliability improved. We did see mixed behavior when looking at the exposure to a reliability drop – about half the users returned to autonomy faster the second time, while the other half were slower. We are unclear on why this difference exists and the limited sample size (n=7) prevents any conclusive interpretations.

Team experience suggests that risk is important, which was our main motivation for the milestone payment and the inclusion of "victims" in the protocol. A regression analysis of various factors' influence on trust ratings was, in fact, dominated by features associated with risk. These were explicit factors (e.g., tendencies towards risky behavior, payment, risk of not receiving payment) and implicit (e.g., victims found, participant age). Personal factors associated with self assessment (TLX, self performance rating) were also important in the analysis. These risk and personal factors overwhelmed factors associated with robot performance and robot reliability. In short, operators tie trust to their own actions rather than robot performance. Other work by the team has revealed similar tendencies in expert robot operators.

6. ACKNOWLEDGMENTS

This work has been supported in part by the National Science Foundation (IIS-0905228 and IIS-0905148). Funding for Sofia Gadea-Omelchenko was provided through the Quality of Life Technology ERC REU program (EEEC-0540865). Thanks to Poornima Kaniarasu, Arne Suppe, and John Kozar for their assistance with robot and experiment logistics, Adam Norton for his assistance with figures, and to all of the experiment participants.

7. REFERENCES

[1] H. Atoyan, J. Duquet, and J. Robert. Trust in new decision aid systems. In *Proceedings of the 18th Int'l Conference of the Association Francophone d'Interaction Homme-Machine*, page 122. ACM, 2006.

[2] I. Dassonville, D. Jolly, and A. Desodt. Trust between man and machine in a teleoperation system. *Reliability Engineering & Systems Safety*, 53(3):319–325, 1996.

[3] P. deVries, C. Midden, and D. Bouwhuis. The effects of errors on system trust, self-confidence, and the allocation of control in route planning. *Int'l Journal of Human-Computer Studies*, 58(6):719–735, 2003.

[4] M. Dzindolet, S. Peterson, R. Pomranky, L. Pierce, and H. Beck. The role of trust in automation reliance. *Int'l Journal of Human-Computer Studies*, 58(6):697–718, 2003.

[5] M. Dzindolet, L. Pierce, H. Beck, and L. Dawe. The perceived utility of human and automated aids in a visual detection task. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 44(1):79, 2002.

[6] A. Freedy, E. DeVisser, and G. Weltman. Measurement of trust in human-robot collaboration. *Collaborative Technologies and Systems*, Jan 2007.

[7] P. Hancock, D. Billings, K. Schaefer, J. Chen, E. de Visser, and R. Parasuraman. A meta-analysis of factors affecting trust in human-robot interaction. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 53(5):517–527, 2011.

[8] D. Harville. Maximum likelihood approaches to variance component estimation and to related problems. *Journal of the American Statistical Association*, pages 320–338, 1977.

[9] J. Jian, A. Bisantz, and C. Drury. Foundations for an empirically determined scale of trust in automated systems. *Int'l Journal of Cognitive Ergonomics*, 4(1):53–71, 2000.

[10] J. Lee and N. Moray. Trust, self-confidence and supervisory control in a process control simulation. *IEEE Int'l Conference on Systems, Man, and Cybernetics*, pages 291–295, 1991.

[11] J. Lee and N. Moray. Trust, control strategies and allocation of function in human-machine systems. *Ergonomics*, 1992.

[12] J. Lee and N. Moray. Trust, self-confidence, and operators' adaptation to automation. *Int'l Journal of Human-Computer Studies*, 40(1):153–184, 1994.

[13] J. Lee and K. See. Trust in automation: designing for appropriate reliance. *Human Factors*, 46:50–80, 2004.

[14] C. Liu and S. Hwang. Evaluating the effects of situation awareness and trust with robust design in automation. *Int'l Journal of Cognitive Ergonomics*, 4(2):125–144, 2000.

[15] D. H. McKnight, V. Choudhury, and C. Kacmar. The impact of initial consumer trust on intentions to transact with a web site: a trust building model. *Journal of Strategic Information Systems*, 11(3–4):297–323, 2002.

[16] B. Muir. Trust between humans and machines, and the design of decision aids. *Int'l Journal of Man-Machine Studies*, 27(5-6):527–539, 1987.

[17] B. Muir. *Operators' trust in and use of automatic controllers in a supervisory process control task*. PhD thesis, University of Toronto, 1990.

[18] B. Mutlu and J. Forlizzi. Robots in organizations: the role of workflow, social, and environmental factors in human-robot interaction. In *HRI '08: Proceedings of the 3rd ACM/IEEE Int'l Conference on Human Robot Interaction*, pages 287–294, New York, NY, USA, 2008. ACM.

[19] R. Parasuraman and V. Riley. Humans and automation: Use, misuse, disuse, abuse. *Human Factors*, 39(2):230–253, 1997.

[20] M. Rehak, L. Foltyn, M. Pechoucek, and P. Benda. Trust model for open ubiquitous agent systems. In *IEEE/WIC/ACM Int'l Conference on Intelligent Agent Technology*, pages 536–542, 2005.

[21] V. Riley. Operator reliance on automation: Theory and data. *Automation and Human Performance: Theory and Applications*, pages 19–35, 1996.

[22] J. Sanchez. *Factors that affect trust and reliance on an automated aid*. PhD thesis, Georgia Institute of Technology, 2006.

[23] W. Song, V. Phoha, and X. Xu. An adaptive recommendation trust model in multiagent system. *Published by the IEEE Computer Society*, 2004.