Characterizing Efficiency of Human Robot Interaction: A Case Study of Shared-Control Teleoperation[1]

Jacob W. Crandall and Michael A. Goodrich

Computer Science Department, Brigham Young University

Abstract
Human-robot interaction is becoming an increasingly important research area. In this paper, we present a theoretical characterization of interaction efficiency with an eye towards designing a human-robot system with adjustable robot autonomy. In our approach, we analyze how modifying robot control schemes for a given autonomy mode can increase system performance and decrease workload demands on the human operator. We then perform a case study of the design of a shared-control teleoperation scheme and compare interaction efficiency against a traditional manual-control teleoperation scheme.

1. Introduction

In many applications, it is desirable to allow a human to interact with multiple robots. These applications include search-and-rescue, exploration, hazardous waste clean-up, and so on. Unfortunately, there is a limit to how many tasks a human can manage in a given time. This means that the number of independent robots in a human-robot team is limited. To understand how many robots a human can manage, it is necessary to understand how humans interact with individual robots under varying circumstances. The likely performance of a particular interaction scheme encodes this efficiency, but this performance degrades as human workload and environmental complexity increase. Thus, it is important to understand how the likely performance of an interaction scheme changes as a function of workload and complexity because such an understanding allows us to predict the performance of a team of many robots.

Formally, we define an *interaction scheme* as a triplet consisting of the *autonomy mode* of the robot, the *control element* used by the human to communicate information to the robot, and the *information element* used by the robot to communicate to the human. In this paper, we present a theoretical framework for understanding how the expected performance of a particular interaction scheme changes as robots are neglected and as world complexity increases. We then present results from a case study that compares the neglect and complexity tolerance of two autonomy modes under identical control and information elements. These results are a first step toward validating the theoretical framework. We conclude by discussing how the framework can be further validated and how the validated framework can be used to guide the design of human-robot sytems.

[1]This paper appeared in IROS'02.

2. Related Literature

While different levels of autonomy have been studied extensively, research in teleoperation is most mature [7]. Perhaps the most difficult obstacle to effective teleoperation occurs when there are communication delays between the human and the robot. The standard approach for dealing with these issues is to use supervisory control. Work on teleautonomy [3] and behavior-based teleoperation [8] are extensions to traditional supervisory control that are designed specifically to account for time delays. Of particular interest are approaches to behavior-based design of robots that can interact with humans. Arkin and Ali's work has been particularly relevant to our research [1]. In their work, they show how potential fields can be used for shared-control teleoperation. They present experimental results for hundreds of test subjects of a shared-control system that allows a human to interact with a team of simple behavior-based robots.

In measuring the effectiveness of human-machine interaction, much work has been done on operator workload. Of particular relevance is Boer's work relating workload and entropy [2]. In addition, Boer has used secondary tasks to help evaluate the cognitive workload placed on human operators.

3. Interaction Efficiency

As stated in the introduction, one purpose of this paper is to present a theoretical framework for characterizing the efficiency of human-robot interaction. This framework is built on the intuition that the likely performance of human-robot interaction degrades as the human neglects the robot to perform other tasks and as world complexity increases.

3.1 Framework

Consider the design of optimal controllers. The design of such controllers is the task of choosing a control law π that maps observations (states) of the environment s into actions a in such a way that performance is maximized (or cost is minimized). Formally and in our notation, the objective of an optimal controller can be stated as follows:

$$\text{Maximize :}$$

$$J(\pi) = E\left[\sum_k \Phi(s_{k+1)} + \Lambda(\pi(s_k))\right] \tag{1}$$

$$\text{Subject to :}$$

$$s_{k+1} = f(s_k, a_k) \tag{2}$$

where $\Phi(s_k)$ is the payoff of visiting state s_k on a path to a goal, $\Lambda(\pi(s_k))$ is the payoff for using control action $a_k = \pi(s_k)$, the sum indicates that performance is accumulated over time, $f(s_k, a_k)$ is a model that describes how action a at time k translates the state s_k into a new state s_{k+1}, and $E(\cdot)$ indicates an expectation. Expectation is included since the dynamics model may be probabilistic (e.g., as in Markov decision processes). An optimal control law π is the mapping from states to actions that maximizes the expected payoff subject to the constraint imposed by the way inputs change the state of the world.

In human-robot interaction, the behavior of the robot is produced by a control law that accepts human input. Thus, we generalize the notion of a control law to include the closed loop of human-robot interaction, and replace the term *control law* with the term *interaction scheme*. The interaction between a human and a robot is diagrammed in Figure 1 which illustrates the interface between human and robot as well as the autonomy

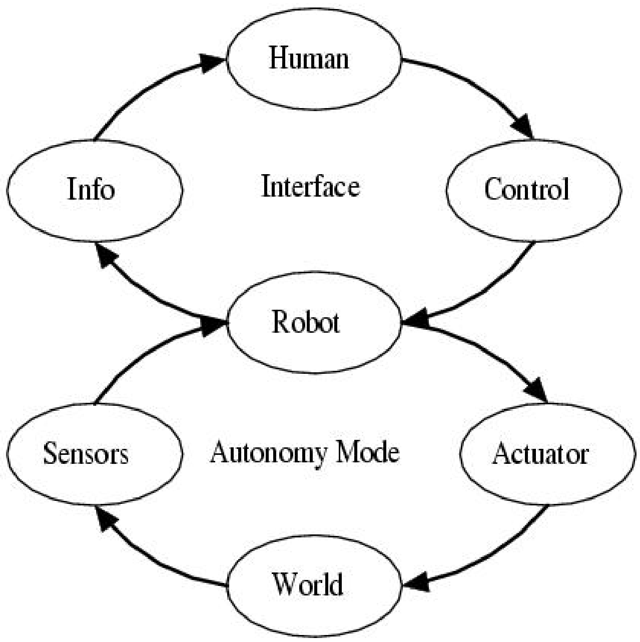

Figure 1: The interface loop and autonomy loop for human-robot interaction.

loop between robot and world. Recall from the introduction that an interaction scheme is defined as a triplet consisting of the *autonomy mode* of the robot, the *control element* used by the human to communicate information to the robot, and the *information element* used by the robot to communicate to the human. The autonomy mode refers to the closed loop behavior of the robot in the world, and the control and information elements refer to the closed loop behavior of the robot in the interface with the human.

In human-robot interaction, the action a_k is composed of both robot input and human input. Since human attention is switched between multiple tasks, the action a_k is not influenced by a human at every sample interval. The effective rate of interaction between the robot and the human is a random variable that strongly infulences the perforamnce J. Interaction schemes π that are designed for frequent human input will not produce high payoffs when humans interact less frequently. We generalize the notion of interaction rate to *neglect* and denote this by the random process N. The notion of neglect includes the possibility of variable interaction rates, multiple tasks, and switching costs.

In addition to the influence of N, the expected performance $J(\pi)$ of a particular interaction scheme π is also affected by how the world responds to robot actions. The manner in which the world responds is encoded in Equation (2) as the function $f(s_k, a_k)$. Since many of the worlds in which robots will benefit by human interaction are highly dynamic and complex, the environment function f is also a random process. We will restrict attention to fixed domains whence we assume that the qualitative characteriscs of f stay the same, but the complexity of the environment, denoted by C, can change. Interaction schemes that are designed for a particular level of environmental complexity may not perform well for other environment complexities.

In Equation (1), the expected payoff J for a particular interaction scheme π is a scalar value, but when the influence of neglect N and complexity C are taken into consideration J also becomes a random process. In general, as complexity or neglect increases, expecte pformance decreases as illustrated in Figure 2. The trend that performance decreases as neglect or complexity increase characterize both the interface and autonomy loops in Figure 1, respectively. The plot $J(\pi; N, C)$ characterizes how performance generated by a particular interaction scheme is influenced by interaction rate and complexity.

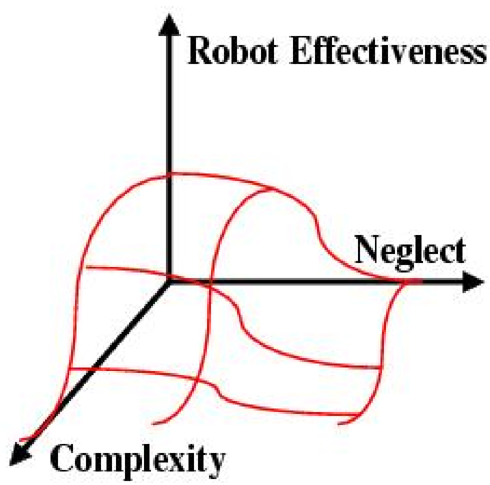

Figure 2: Performance $J(\pi; N, C)$ of interaction scheme π as a function of neglect and task/world complexity.

3.2 Performance Depends on Neglect

To enable a human to manage multiple tasks (including interacting with multiple robots), it is necessary to know how long a human can give attention to one robot before the performance of the other tasks deteriorate. The relationship between neglect and expected performance can be characterized using the neglect curve illustrated in Figure 3 for a human-robot system under various autonomy modes. The idea of the neglect curve is simple. Interaction scheme A's likely effectiveness, which measures how well the human-robot system accomplishes its assigned task and how compatible the current task is with the human's objective, decreases when the human turns attention from the task to a secondary task; when the task is neglected the interaction scheme becomes less effective.

The neglect curve can be used to determine how often we would expect interactions to occur to maintain a level of performance. To prevent the performance of an interaction scheme from dropping below an acceptable level, the robot can only be neglected for a

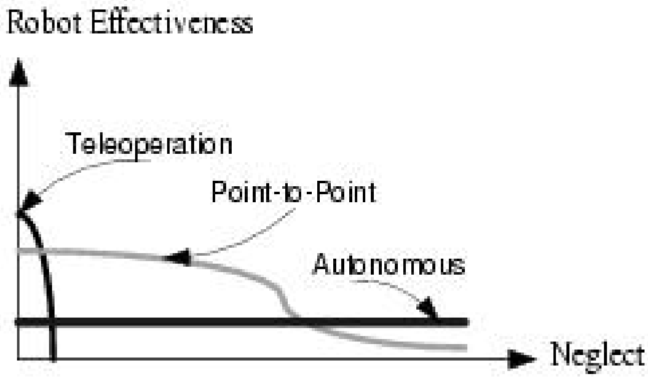

Figure 3: The neglect curve is a plot of $J(\pi; N)$ for constant complexity as a function of N. The nearly vertical curve represents an interaction scheme which includes the potential for great effectiveness but which fails if the operator neglects the robot. The horizontal line represents a fully autonomous robot which includes less potential for effectiveness but which maintains this level regardless of operator input. The sloping curve represents intermediate types of interaction for which effectiveness decreases as neglect increases.

certain period of time defined as the time spent off the task plus the time spent on the task bringing the performance back to a high level. The acceptable neglect time (time-off-task) includes both the time spent on other tasks as well as the time to switch attention.

3.3 Performance Depends on Complexity

To illustrate how world complexity can impact performance, consider how neglect tolerance depends on the number of branches and amount of clutter in an environment. If the world has minimal clutter and very few branches, then the robot can be neglected for an extended period of time. If, however, the world is cluttered and has many branches, then uncertainty will increase causing the robot to be less tolerant to neglect. Thus, performance decreases as complexity and neglect increase.

3.4 Performance Depends on Information and Control

Given the curves that describe the expected performance of interaction as a function of neglect and complexity, $J(\pi; N, C)$, it is appropriate to explore how presenting information affects this efficiency. An information system can increase neglect tolerance primarily by decreasing the amount of time required for the human to switch attention from another task and gain relevant situation awareness for the particular robot. The information presented by such systems performs three objectives: it triggers an attention switch from a secondary task to a relevant robot interaction task, it speeds up the time to switch between the secondary task and the interaction task by helping the human get "in the loop" faster, and it helps the human perform the task more quickly thereby decreasing time-on-task. Unfortunately, a poorly designed information system may cause the process of gathering information to become a task in and of itself. This effectually extends the time to switch from a secondary task by compelling the human to attend to the information task before attending to the primary decision task.

Similar to the way in which information can change the characteristics of interaction, the manner of giving information to the robot also changes these characteristics. For example, if a control scheme is very complex, the human may have difficulty forming an efficient mental model of the interaction. Without an efficient mental model, the process of presenting information to the robot may become a task in and of itself. This effectually extends the time to switch from a secondary task by compelling the human to attend to the control task after attending to the primary decision task.

4. Shared Control

The purpose of this section is to explain the shared-control teleoperation system that we have created. The development of this system was described in [4], but we will review this algorithm and present more complete experimental results in this paper. The system consists of a Nomad SuperScout robot and a remote computer. The remote computer and the robot are connected via an 11Mb/s wireless ethernet. A GUI displays video, sonar readings, and compass information from the robot. Through a Microsoft SinderWinder II Force Feedback Joystick, the human guides the robot.

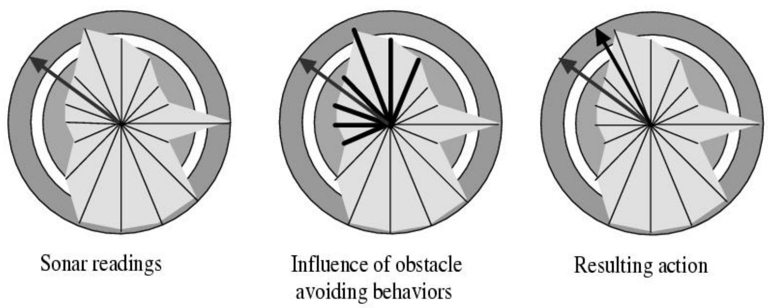

Figure 4: A graphical depiction of the algorithm for a robot positioned in a hallway with a door open on the robot's right. Raw sonar readings (left) are translated into relevant behaviors (middle) and combined with the human input to produce the actual robot action (right).

Our approach to shared-control teleoperation uses a variant of potential fields. In the algorithm, the angle of each sonar is associated with a behavior. Sonars that measure nearby obstacles return repelling behaviors, and sonars that measure open spaces return attracting behaviors. More specifically, sonar distances are classified into three categories: repelling, neutral, and attracting. If the sonar returns a distance greater than a pre-defined *safe distance* (65 inches in our experiments) then the corresponding behavior is categorized as an attracting behavior. If the sonar returns a distance less than a pre-defined *risk distance* (40 inches in our experiments) then the corresponding behavior is categorized as a repelling behavior. For other sonars, the corresponding behavior is categorized as a neutral behavior.

Given these categorizations, the attracting behaviors are assigned strengths according to how close their angles are to the human input. Angles that are nearby are given large strengths, and angles that are far away are given zero strength. Similarly, the repelling behaviors are weighted by how close their angles are to the human input. However, unlike attracting behaviors, the strength of each repelling behavior is also weighted by the distances they return; small distances indicate obstacles that are very close and are therefore given high strength. After the strength of each behavior is obtained, the behavior vectors are summed with the human input vector to produce the resulting direction that the robot will move. The strengths used in the experiments presented herein are given in [4].

This process is illustrated in Figure 4. In the figure, the human tells the robot to go forward and left (see the image on the left). Sonar readings that are relevant are identified (see the image in the middle). Those behaviors that would move the robot toward an opening (as indicated by the sonar reading terminating in the outer shaded circle) in the world pull the robot toward the opening, and those behaviors that would move the robot toward a nearby obstacle (as indicated by the sonar reading terminating in the inner shaded circle) push the robot away from the obstacle. These pulls and pushes are combined with the human input to specify the direction that the robot will go (see the image on the right); in the example, the robot will still go forward and left, but will not go as far to the left as suggested by the human.

Since vector summation in a potential fields algorithm allows for some obstacle-avoiding behaviors to cancel out, sometimes undesirable emergent behaviors occur. In our case, under certain circumstances, the robot can be directed into an obstacle. To avoid this, we include a safe-guarding [5, 6] behavior, which can veto the direction. Using all sixteen sonar readings we define a *safe* region by simply finding the points at which the sonars indicate that there are objects. Connecting these points yields a polygon with sixteen sides, which makes up the safe region. By predicting where the robot will be at some future time t, the robot can determine if it will leave this region anytime in the near future if it continues the course it has selected. If the robot thinks it will leave this safe region anytime in the near future, the direction is vetoed and the robot defaults to a behavior that causes the robot to rotate slowly in place towards the nearest perceived clear pathway.

5. Validation Experiment

In this section, we present an experiment to compare the shared-control teleoperation system described above with a direct-control teleoperation system. The two schemes both use a joystick as the control element and both use a video display and graphical depictions of sonar readings as the information element. The interaction schemes differ by the autonomy mode, shared control or direct control. First, we describe the experiment and then we explain the criteria and results.

5.1 Experiment Description

The primary task in the experiments is to guide a robot through a cluttered course with simple decision points. The course is illustrated in Figure 5. In experiments involving human cognitive load, experiment participants are sometimes asked to perform a secondary task (or tasks) as they perform a primary task [2]. In our experiment, subjects must solve two digit addition problems while performing the robot guidance task.

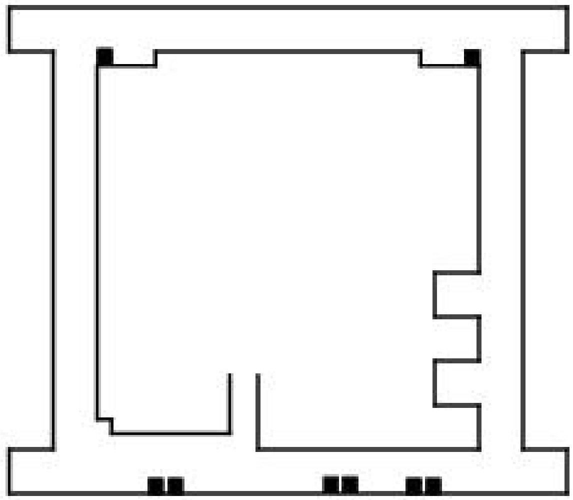

Figure 5: *The environment used to measure neglect. Notice the nominal amount of clutter.*

As a rule, experiment participants should not have experience driving the robot. This ensures that no biases are introduced due to past training. For each participant, the following steps are followed:

Step 1. The math proficiency level of the participant is determined. Two digit addition problems are displayed on the screen along with four multiple choice answers (only one being the correct answer). The participant is given 5 seconds to answer the question. A log of math proficiency is kept. After the participant answers the question, he or she may proceed to a new problem by clicking on a button. This proficiency test lasts for two minutes. If the participant cannot successfully complete 60% of the problems, the difficulty level is reduced to adding a two-digit number to a one-digit number.

Step 2. Next, the participant must be trained to guide the robot using a particular autonomy mode. Scheme S is the *S*hared-control teleoperation scheme, and Scheme D is a traditional *D*irect-control teleoperation scheme. In order to not bias results, some participants are trained and tested on Scheme A first, and others are trained and tested on Scheme B first. After completing initial training, the participant is asked to guide the robot through the course as quickly as possible. While doing so, he or she must look out for the safety of the robot. Training is complete when the subject has successfully guided the robot through the course one time.

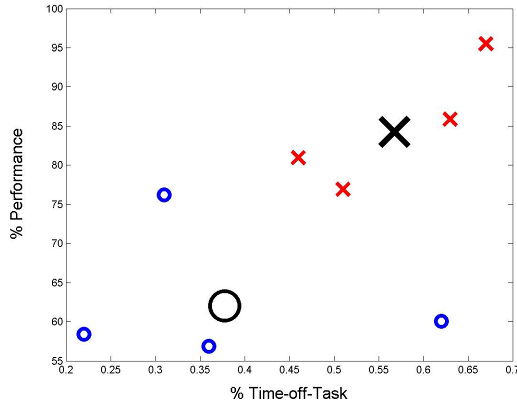

Figure 6: A plot of robot effectiveness verses neglect rate. The vertical axis represents robot performance (as a percentage to maximum effectiveness) and the horizontal axis represents neglect (in terms of percentage of time off task).

Step 3. The participant is asked to again guide the robot through the course. This time, the participant is asked to do math problems as he or she drives the robot. The participant is instructed to guide the robot through the course as quickly as possible, and to answer as many math problems in this time as he or she can, while making sure the robot is safe.

Steps 4–6. The participant repeats steps 2–3 using the other control scheme. That is, if the participant started with Scheme S, then he or she is next tested on Scheme D and vice versa.

5.2 Evaluation Criteria and Results

In this experiment, we fix the level of complexity and explore how interaction efficiency is affected by human neglect. The best interaction scheme for a given level of complexity is the system that can move the knee of the neglect curve as far to the right as possible. In general, a lower workload imposed by an interaction scheme means the operator is free to neglect the robot more. This, in turn, means that the knee of the curve will be moved right, assuming that performance level doesn't decrease. There are several ways that we show neglect and workload in our system, and these measurements and results are described in the following subsections. Figure 6 shows robot effectiveness verses neglect for the task performed in the experiment. It is interesting to note from this graph that the shared-control system (represented by the x's) dominates the direct-control system (represented by the circles) for each participant on the given task.

Neglect Rates Neglect time is the amount of time spent doing other tasks. Thus, neglect is the time spent solving arithmetic problems divided by the total time of the trial run. In the experiments, the four participants were able to neglect the robot an average of 50% more using shared control than direct control.

Joystick Steering Entropy We obtain the joystick steering entropy for each participant using the algorithm described in [2]. Slight adjustments are made to this algorithm, but they are small. Note, however, that entropy data from this paper should not be compared to entropy readings in [2]. Entropy ratings range between 0 and 1. A high entropy rating means that joystick movements are choppy and thus indicates that the operate is under a higher workload. Thus, lower entropy ratings indicate that the operator has a lower and more manageable workload.

In the experiments, joystick steering entropy was considerably higher on this task for the direct-control system. On average, entropy increased by just over 50% when the direct-control system was used. This indicates that the cognitive workload was higher for direct-control than shared-control. These results are consistent with the results on neglect rates, since workload and neglect rates should have a significant negative correlation.

Primary Task Effectiveness This is how well the participant did in driving the robot through the course. To keep things simple and objective, the judgement of how well a task was performed is established simply by how much time it takes to get the robot around the building. The distance the robot is required to travel and maximum robot speed dictates that it take at least 170 seconds to get through the course. We base performance off this number: $Performance = \frac{170}{TimeElapsed} \times 100$. In the experiments, performance levels for the shared-control system exceeded performance levels of the direct-control system by an average of about 35%.

Secondary Task Effectiveness This is a measurement of how well the participant performed on the arithmetic problems. Both the number of problems completed per minute and the problem proficiency are important. Since each participant's math abilities differ, only comparisons between how well a participant performed in different control schemes is relevant. It is theorized that participants should perform better on the secondary task when they have a lower workload imposed by the primary robot control task.

In the experiments, the secondary task results correlate with the results of all the other recorded data for this experiment. The average arithmetic proficiency on the shared-control system exceeded the average arithmetic proficiency on the direct-control system by 9%. Additionally, the average number of arithmetic problems attempted per minute increased from 7.3 problems per minute when participants used the direct-control system to 12.0 problems per minutes when participants used the shared-control system. That represents an increase of about 65%.

Subjective Rating Each participant is asked to tell which system was better. The judgement criteria of what is better should be based on a general perception of how the participant felt they did on each scheme. In the experiments, the participants of the experiment unanimously have indicated that the shared-control system is better than the manual-control system for the task tested in the experiment.

6. Summary and Future Work

We have presented a framework for evaluating the expected efficiency of an interaction scheme in terms of its sensitivity to neglect and complexity. We then performed a case study that evaluated the observed interaction efficiency of a shared control teleoperation algorithm and compared this efficiency to the efficiency of direct teleoperation. We showed that, for the level of complexity used in the experiments, the shared control scheme was more tolerant to neglect. These results correlated well with measures of human workload and ease of use, suggesting that the framework is valid in some cases.

Future work includes further validation of the framework by conducting experiments that control both neglect level as well as complexity level. Using these experiments, we can characterize the expected performance of various interaction schemes, and identify characteristics of efficient interaction. These experiments will allow us to identify design principles to create efficient teams of robots that interact with a human.

7. Acknowledgment

This work was partially funded by the DARPA Synergistic Cyber Forces project.

References

[1] Khaled S. Ali and Ronald C. Arkin. multiagent teleautonomous behavioral control. In *Machine Intelligence and Robotic Control, Vol 1, No. 2*, pages 3–10, 2000.

[2] Erwin R. Boer, Okihiko Nakayama, Tohru Futami, and Tomokazu Nakamura. Development of a steering entropy method for evaluating driver workload. In *International Congress and Exposition*, 1999.

[3] L. Conway, R. A. Volz, and M. W. Walker. Teleautonomous systems: Projecting and coordinating intelligent action at a distance. In *IEEE Transactions on Robotics and Autonomation 6(2)*, 1990.

[4] Jacob W. Crandall and Michael A. Goodrich. Experiments in adjustable autonomy. In *IEEE Systems, Man, and Cybernetics*, 2001.

[5] T. Fong, C. Thorpe, and C. Baur. A safeguarded teleoperation controller. In *IEEE International Conference on Advanced Robotics (ICAR)*, 2001.

[6] E. Krotkov, R. Simmons, F. Cozman, and S. Koenig. Safeguarded teleoperation for lunar rovers: from human factors to field trials. In *IEEE Planetary Rover Technology and Systems Workshop*, 1996.

[7] Thomas B. Sheridan. *Telerobotics, Autonomation, and Human Supervisory Control.* The MIT Press, 1992.

[8] M. R. Stein. *Behavior-Based Control for Time-Delayed Teleoperation.* Ph.D Dissertation, University of Pennsylvania, 1994.