Robot-Proxy Grounding

Kristen Stubbs

CMU-RI-TR-08-37

Submitted in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Robotics.

Robotics Institute
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213

August 2008

© 2008 by Kristen Stubbs. All rights reserved.

Abstract

Exploration robotics has traditionally utilized an encoder-decoder model of communication between users and a robot. This means that users construct a plan (sequence of actions) to be sent to the robot; the robot executes the plan and returns data to the users, who then construct another plan. The problem with this interaction paradigm is twofold: (1) users must develop a complex mental model of the robotic system in order to create intricate plans, and (2) the robot does not have the users' specialized domain knowledge, so the robot does not have any way to ensure that how it handles unexpected events in the field is consistent with the users' goals (what the users were trying to accomplish through the plan).

Because the cost of communication with the remote robot is extremely high, this work introduces the concept of a "robot proxy," a software system which models both the robot's capabilities and the user's goals. The robot proxy interacts with the user in real-time in place of the robot so as to promote common ground between the two; this process is referred to as Robot-Proxy Grounding. Robot-Proxy Grounding is a novel interaction model for exploration robotics derived from common ground theory, a model of human-human communication introduced and experimentally validated by Herbert Clark and his colleagues. Robot-Proxy Grounding is also based on detailed observations and analysis of the Life in the Atacama exploration robotics project, which indicated that a majority of the errors and miscommunications which occurred during the project resulted from a lack of common ground between participants even as the robot became more autonomous.

A proof of concept study was conducted which compared the effects of an encoder-decoder planning system and a prototype robot proxy; the study suggested that the use of a robot proxy was effective in improving task efficiency and fostering feelings of collaboration. A full implementation of a robot proxy-based planning system was constructed and evaluated. This user study demonstrated that participants who used the robot proxy were more efficient at the task, collected higher-quality data, and possessed more accurate information about the robot's internal state and its context than participants without a robot proxy. The results suggest that the implementation was successful at promoting common ground with the user, resulting in improved task performance.

Dedication

To all of the teachers who have inspired me and encouraged me
in my journey as a learner and as a scientist, especially
my elementary school teachers Leslie Hochsprung and Janice Gardner;
my high school math teachers Carol Keating and Tricia Lillygren;
my high school English teacher, Joyce Halsey;
and my high school debate coach, Rachel Russell.

Acknowledgments

While it has been said that "It takes a village to raise a child," at this point in my life I am convinced that it also takes a village to create a PhD. I am grateful to a large community of people who have provided guidance, encouragement, and support to me as I have progressed through graduate school.

First and foremost, I am indebted to my husband, Colin, for all of his loving support, including, but not limited to: providing feedback on my ideas, proofreading papers, helping me write algorithms, debugging my code, debugging other people's code, and spending hours pre-testing user studies. I am extremely grateful for all of the time and effort he has spent helping me to achieve my dream.

I would also like to express my sincere appreciation for and gratitude to my advisors, Illah Nourbakhsh and David Wettergreen. They have been absolutely fantastic to work with: encouraging, supportive, and patient. My thanks to Dave and Illah for letting me follow my interests from the very beginning and for believing in me and my research. Thank you, Illah, for inspiring me to think big; and thank you, Dave, for reminding me to keep my feet on the ground. I feel like I still have a lot to learn from both of you, and I will do my best to advise future students with the same thoughtfulness and care that you have shown me.

Robotics as a field still has relatively few women in it, which is why I am all the more grateful for the help of Pam Hinds and Sara Kiesler, who have been mentors and role models to me over the course of my thesis work. I would like to thank Sara for inviting me to become involved with the Project on People and Robots, for all of the constructive criticism she has provided me on my research over the years, and for her patient assistance with the data analysis for my pilot study. My thanks to Pam for teaching me how to conduct and analyze ethnographic studies, for introducing me to common ground theory, and for collaborating with me on the Life in the Atacama project. I appreciate her sharing her expertise with me and am honored to have had the opportunity to work so closely with her.

Thanks to the other faculty members who have provided me with feedback and with a supportive environment in which I could grow as a scientist, especially Brett Browning, Kevin Crowley, Jodi Forlizzi, Matt Mason, and Reid Simmons. Thanks also to the current and former members of the Project on People and Robots who shared their insights with me, especially Carl DiSalvo, Marek Michalowski, Bilge Mutlu, and Cristen Torrey. I would also like to thank everyone who participated in my research studies: the em-

ployees at the San Francisco Exploratorium, the NASA Ames Exploration Center, the National Science Center, and the Smithsonian Air and Space Museum who participated in my Personal Exploration Rover study; the graduate students who participated in my user interface studies; and, most of all, the entire Life in the Atacama team. There are too many people to name everyone, but I would especially like to thank science team members Nathalie Cabrol, Jim Dohm, Andy Hock, Jen Piatek, Kim Warren-Rhodes, and Shmuel Weinstein as well as engineers Dom Jonak, Trey Smith, Dave Thompson, and Mike Wagner. This thesis would not have been possible without all of your hard work (and without your letting Pam and I scrutinize every minute of it).

I consider myself extremely fortunate to have met so many colleagues who have also become dear friends. I would like to thank Debra Bernstein, a wonderful friend and research collaborator, for her help with my Personal Exploration Rover research as well as providing emotional support (and many trips to Trader Joe's). Thanks to Rachel Kirby, my longtime officemate and close friend, for all of the thoughtful, insightful feedback she has given to me on my papers and talks during my time at CMU. I also credit Rachel with introducing me to the fun of Stitch 'n Bitch and patiently teaching me how to crochet. I am grateful to the RoboGals who came before me and mentored me, especially Bambi Brewer, Maayan Roth, Caroline Pantofaru, and Nidhi Kalra. Thank you to all of the other people who have made my life in Pittsburgh so much fun, especially Alice Brumley (and Saba), Sonia Chernova and James Hays, Emily Hamner, Tom Lauwers, Liz Liao, and Doug Vail. Thanks to Cheryl Bernstein and Tina Kosko for helping me gain the strength I needed to complete my PhD.

Finally, I would like to express my gratitude to my family, who have supported me from the very beginning. Thanks to my sister, Kaci, for always being there for me and for making me laugh when I needed it the most, and thanks to my parents for talking with me and encouraging me every week. I am extremely proud to become a Doctor of Philosophy just like my father.

**For some reason, they want another [panorama],
so that's what I'll do now.**
Life in the Atacama engineer, on following the science team's plan
Excerpt from field notes of P. Hinds, Life in the Atacama Project, 2005

This is a case of us thinking in numbers instead of in reality.
Life in the Atacama scientist, on the challenges of creating robot plans
Excerpt from field notes of K. Stubbs, Life in the Atacama Project, 2005

Contents

1 Introduction 21
- 1.1 Objective . . . 22
- 1.2 Approach . . . 23
- 1.3 Contributions . . . 26
- 1.4 Overview of Thesis . . . 27

2 Background 29
- 2.1 Human-Robot Interaction . . . 29
- 2.2 Exploration Robotics . . . 30
- 2.3 Human-Robot Interaction in Exploration Robotics . . . 32
  - 2.3.1 Space Robotics . . . 32
  - 2.3.2 Urban Search and Rescue . . . 33
  - 2.3.3 Educational Robotics . . . 33
- 2.4 Common Ground Theory . . . 34
  - 2.4.1 Empirical Validation . . . 36
  - 2.4.2 Situation Awareness . . . 40
- 2.5 Applications of Common Ground Theory . . . 42
  - 2.5.1 Common Ground and Human-Computer Interaction . 42
  - 2.5.2 Common Ground and Human-Robot Interaction . . . 42
- 2.6 Summary . . . 44

3 Foundations of Robot-Proxy Grounding 47
- 3.1 Research Methodology . . . 47
  - 3.1.1 Observational Studies . . . 47
  - 3.1.2 Coding Schemes . . . 48
- 3.2 The Personal Exploration Rover . . . 49
  - 3.2.1 System Description . . . 49
  - 3.2.2 Long-Term Human-Robot Interaction . . . 51

3.2.3 Impact on Formation of Robot-Proxy Grounding . . . 54
3.3 Life in the Atacama 54
3.3.1 Project Description 54
3.3.2 Autonomy and Common Ground in Human-Robot Interaction 58
3.3.3 Impact on Formation of Robot-Proxy Grounding . . . 68
3.4 Summary 69

4 Characterization of Common Ground for Human-Robot Interaction 71
4.1 Common Ground for Human-Robot Interaction 72
4.2 Analysis of the Life in the Atacama Project 75
4.3 Analysis of the Personal Exploration Rover Project 77
4.4 Summary 80

5 Robot-Proxy Grounding 81
5.1 Characteristics of Grounding in Exploration Robotics 82
5.1.1 Constraints on Grounding 82
5.1.2 Costs to Participants 83
5.2 Presentation-Acceptance 85
5.2.1 Traditional Common Ground Presentation-Acceptance 87
5.2.2 Presentation-Acceptance with a Robot Proxy 88
5.3 Summary 90

6 Proof-of-Concept Study 91
6.1 Study Design and Method 91
6.1.1 Participants 92
6.1.2 Procedure 92
6.1.3 Simulation 95
6.1.4 Dependent Variables 97
6.2 Results 97
6.2.1 Task Performance 100
6.2.2 Mental Model Development 102
6.2.3 Self-Evaluation of Performance 103
6.3 Discussion 104

7 Robot Proxy Requirements 107
7.1 Robot Proxy Overview 107
7.2 Goal Representation 108
7.2.1 The Need for Goal Representation 108

7.2.2 Goal Representation Requirements 110
7.3 Goal Validation 114
7.3.1 The Need for Goal Validation 114
7.3.2 Goal Validation Requirements 116
7.4 Robot Model 118
7.5 Summary 118

8 Robot Proxy Design and Implementation 119
8.1 System Overview 119
8.2 Goal Representation Implementation 123
8.2.1 Action 124
8.2.2 Locale 124
8.2.3 Science Goals 125
8.3 Robot Model Implementation 125
8.3.1 Robot Manager 125
8.3.2 Configuration Files 126
8.4 Goal Validation Implementation 127
8.4.1 Check Action 127
8.4.2 Check Locale 127
8.4.3 Check Plan 129
8.4.4 Goal Validation Responses and Remedies 129
8.5 Summary 135

9 Evaluation Study Design 137
9.1 Study Design and Method 137
9.2 Participants 138
9.3 Procedure 138
9.4 Simulation 147
9.5 Dependent Variables 147

10 Analysis of Evaluation Results 151
10.1 Task Performance 151
10.1.1 Phase 1: Locale Placement 151
10.1.2 Phase 2: Examining Rocks for Signs of Life 152
10.2 Self-Evaluation of Performance 154
10.3 Establishing Common Ground 160
10.4 Constraints and Limitations 162
10.5 Support for Robot-Proxy Grounding 167

11 Conclusion **169**
11.1 Contributions 171
11.2 Generalization 172
11.2.1 Building Common Ground Using a Robot Proxy . . . 172
11.2.2 The Robot-Proxy Grounding Presentation-Acceptance Process 175
11.3 Future Work 176
11.4 Summary 177

A Glossary **181**

B 2005 Life in the Atacama Errors/Miscommunications **183**

C Excerpts from Life in the Atacama Field Notes **185**
C.1 2005 Regular Operations (Low Autonomy) 185
C.2 2005 Science Autonomy (High Autonomy) 186

List of Figures

1.1 Current exploration robotics systems utilize an encoder-decoder model of communication. 22

1.2 A common ground-based model for exploration robotics mission planning which supports a "conversation" between the user and robot. 24

1.3 A common ground-based model for exploration robotics mission planning in which the user builds common ground with a robot proxy system before the plan is sent to the robot for execution. 25

2.1 A simple example of the presentation-acceptance process. 35

3.1 The Personal Exploration Rover (PER) at a museum installation. 50

3.2 Collaboration diagram of all members of the PER system and their interactions [Stubbs et al., 2006a]. 51

3.3 Zoë, the "robotic astrobiologist" used in the Life in the Atacama project. 55

3.4 Science team members view data returned from the robot. . . 56

3.5 Members of the engineering team work to repair the robot. . 56

3.6 This study examined the grounding process (a) between the science team and the robot and (b) between the science team and the engineering team. 57

3.7 Number of problem instances related to a lack of common ground in 2004 and 2005. 58

3.8 Types and levels of autonomy in 2004, 2005, and with the science autonomy system. 60

3.9 Common ground issues for problems relating to autonomy. . . 67

6.1 The side and top-down views of the robot as presented to study participants. 93
6.2 For each fragment, participants were given (a) an overhead map and (b) a set of five possible plans. 95
6.3 An example of the type of feedback shown to participants in the Robot Proxy group. 96
6.4 A screenshot containing an image of the first fragment after it has been completely cleaned. 96
6.5 Mean number of cycles per trial for participants who successfully completed the task. 100
6.6 Mean number of incorrect plans sent to the robot per trial. . 101
6.7 Mean proportion of time spent reviewing data from the robot per trial. 102
6.8 Least squares mean of score on each question by condition. . 103
6.9 Least squares means of self-evaluation of effectiveness. 104
6.10 Least squares means of self-evaluation of collaboration with the system. 104

7.1 Part of a geologic map of Site F created by one of the members of the LITA science team. 112

8.1 System diagram 120
8.2 A sequence diagram illustrating the overall process of checking an action. 121
8.3 A sequence diagram illustrating the overall process of checking a locale. Checking a plan follows essentially the same sequence. 122
8.4 A Unified Modeling Language (UML) class diagram of the three major components of the goal representation [Booch et al., 1999]. 123
8.5 A Unified Modeling Language inheritance diagram illustrating the possible responses from the Goal Validator [Booch et al., 1999]. 133
8.6 Unified Modeling Language inheritance diagram illustrating the possible options to repair a failed goal validation check [Booch et al., 1999]. 134

9.1 Handout that was presented to study participants on the dimensions of the robot. 139

9.2 Handout that was presented to study participants on the commands which could be sent to the robot. 140
9.3 Sample images taken by the robot using (a) the fluorescence imager (FI) and (b) the spectrometer (SPEC). 142
9.4 The orbital map presented to study participants showing the location of the robot (square) and five areas of interest labeled A through E. 142
9.5 An example overhead map showing the location of the robot and the rock. 143
9.6 The main interface display for participants in (a) the No Proxy group and (b) the Robot Proxy group. 144
9.7 The interface used to add actions to a locale for participants in (a) the No Proxy group and (b) the Robot Proxy group. 145
9.8 The two screens of feedback provided after the "Check Locale" button is clicked: (a) a description of which actions do not overlap and (b) a graph which displays the starting position of the robot, ending position of the robot, and the fields of view of all instrument readings. 146

10.1 Mean and standard error of number of plans executed for both conditions across all three trials. 153
10.2 Mean and standard error of percentage of rock face visible for both conditions across all three trials. 155
10.3 Participants' ratings of how confident they were when sending (a) their first plan to the robot and (b) their last plan to the robot. . . 158
10.4 Participants' ratings of how much responsibility they assigned themselves for (a) successful plans and (b) failed plans. 161
10.5 Participants' (a) average percentage accuracy in predicting image overlap and (b) confidence in their predictions. 163
10.6 Participants' (a) average percentage accuracy in predicting how much of the rock would be visible in an image and (b) confidence in their predictions. 164
10.7 Participants' (a) average error in estimated distance in meters between the robot and the rock and (b) confidence in this estimate. . 165

List of Tables

2.1 Robot-Proxy Grounding contrasted with other related work. . 44

3.1 For each interview and content code, the value listed is equal to the ratio of the number of times that that content code was used out of the total number of lines coded. *Indicates a statistically significant change (one-way repeated-measures ANOVA). [Stubbs et al., 2005] 53

5.1 Summary of Grounding Costs to User and Robot 86
5.2 Impact of Robot Proxy on Grounding Costs to User 87

6.1 Dependent Variables 98
6.2 Multivariate Correlation 99

9.1 Dependent Variables 149

10.1 Multivariate Correlations (Self-Evaluation of Performance and Actual Performance) 156
10.2 Multivariate Correlations (Self-Evaluation of Performance) . . 159
10.3 Multivariate Correlations (Establishing Common Ground and Task Performance 166

B.1 Number of instances of each error type and miscommunication type from the LITA project in 2005. 184

Chapter 1

Introduction

Building better human-robot systems requires that we understand the complex interactions that occur within such systems. As human-robot interaction (HRI) develops, we are becoming more ambitious about the types of interactions we envision for our robots and their users. This thesis focuses on improving human-robot interaction for exploration robotics. Robotic exploration tasks are defined broadly as those in which a robot co-investigates an unknown environment with a remote human partner. Exploration is an important domain of study because of its applicability to a wide variety of problems, which range from searching for signs of life on other planets to investigating debris after a building collapse. In particular, this work centers on exploration which involves the deployment of autonomous robots that work in complex, real-world settings. In these situations, robot users are not likely to be experts in robotics, and they may possess inaccurate mental models of robotic technologies. At the same time, these users often possess sophisticated domain knowledge which the robot does not. In order to facilitate successful exploration, the goal of this thesis is to promote shared understanding between users and robots: that is, to increase users' understanding of robots and foster accurate mental models, and, at the same time, enhance robots' understandings of users and their goals in order to drive robots' decision-making processes.

Current exploration robotics systems follow a communication model similar to the encoder-decoder model of information processing [Krauss and Fussell, 1996] (see Figure 1.1). The user possesses goals which he/she would like to accomplish and uses an interface to encode those goals into machine-readable actions. These actions are sent to the robot, which utilizes a planner to decode and schedule the necessary low-level commands and an exec-

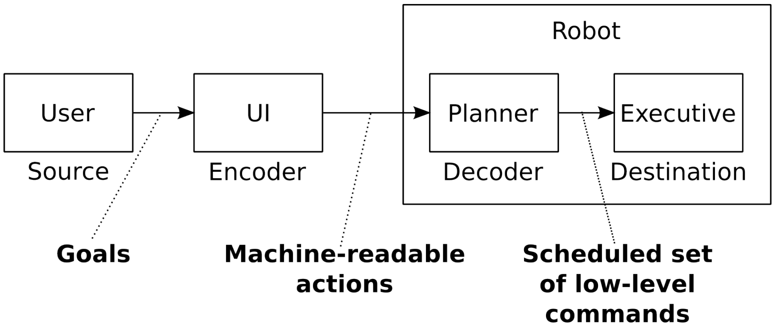


Figure 1.1: Current exploration robotics systems utilize an encoder-decoder model of communication.

utive process to direct the execution of the commands.

There are two significant drawbacks to the use of this communication model:

- The user receives feedback from the robot (in the form of data returned from the robot) only at the end of an execution cycle. Particularly in the case of space robotics missions, time, energy, and communications bandwidth are extremely valuable resources. If a plan results in poor-quality data, these resources may be used inefficiently.
- The robot does not have access to users' underlying goals; it can only access a set of specific actions. In the event of an error or unexpected event during the execution cycle, the robot has no additional information about the user's higher-level goals; thus, there is no way for the robot to ensure that any changes it makes to the plan will still result in valuable data being returned to the user.

1.1 Objective

Based on the weaknesses of the encoder-decoder communication model which is prevalent in exploration robotics system design, this thesis research seeks an answer to the question:

> **In a system consisting of a human and an autonomous, mobile robot engaged in an exploration task, can we improve *efficiency* by helping the user develop a more accurate *mental model* of the robot and ensuring that**

> **the actions that the robot executes are consistent with the user's *underlying goals*?**

Efficiency refers to a ratio of cost to performance. In particular, this thesis focuses on reducing costs to the user by reducing the number of planning and execution cycles required to complete exploration tasks and improving the quality of data returned to the user.

As the user develops plans for the robot to execute, she relies on a **mental model** of the robot's current environment, the robot's current internal state, how the robot's instruments function, how the robot navigates terrain, and so forth. In order for the user to make efficient use of the robot, the user must have accurate knowledge of the robot's capabilities.

Because of the challenges that non-expert users face in developing accurate mental models of complex robotic systems, this thesis also focuses on users' higher-level **goals**—what users are trying to accomplish through a particular sequence of actions. These goals are formulated in terms of users' domain-specific knowledge. In order for a mission to be successful, the actions which the robot executes must help meet these goals.

1.2 Approach

As an alternative to the encoder-decoder model, the focus of this thesis is to explicitly promote common ground between users and robots. As defined by Herbert Clark and colleagues, common ground between two participants in a joint activity is "the knowledge, beliefs, and suppositions they believe they share about the activity" [Clark, 1996, p. 38]. In order for two individuals to communicate and collaborate successfully, they must have common ground [Clark and Wilkes-Gibbs, 1986; Clark and Marshall, 1981]. This thesis considers the user's presentation of a set of goals to be a communicative act which is part of the grounding process (the process by which common ground is built between the user and robot). A common ground-based model of exploration planning is based on the premise that the set of actions which the robot will execute emerges from the process by which the user and the robot come to agree that the user's goals have been understood.

> **This dissertation advances the hypothesis that by utilizing a common ground-based model of exploration planning, it is possible to: improve task efficiency, help the user generate more accurate mental models of the robot,**

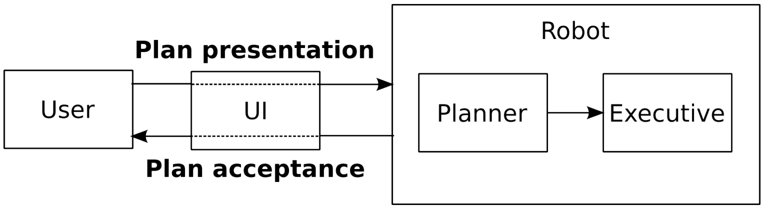


Figure 1.2: A common ground-based model for exploration robotics mission planning which supports a "conversation" between the user and robot.

> **and ensure that the actions which the robot executes meet the user's goals.**

According to this common ground-based model, the robot actively participates in a "conversation" with the user (see Figure 1.2). Initially, the user presents a plan to the robot (Plan presentation). The robot checks that the information which has been presented is consistent and free of errors, providing feedback to the user as needed (Plan acceptance). This process is repeated until the user and robot have reached a mutual understanding that the plan will meet the user's goals, at which point the user commands the robot to execute the plan.

In order to determine how to integrate a common ground-based approach into exploration robotics mission planning, it is important to consider the constraints under which the user and robot are collaborating. Communicating with the robot is extremely expensive in terms of energy and bandwidth; thus, the user has very limited opportunities to interact directly with the robot. Particularly in the case of space exploration, the user and robot may exchange data only once or twice per day. However, the user does not pay a penalty in terms of data return if the user spends time revising the plan (so long as the final plan is sent to the robot by the time it is needed). In order to promote a "conversation" with the robot under these circumstances, this thesis introduces the concept of a *robot proxy* so that the user can participate in the grounding process during plan creation **before the plan is sent to the robot** (Figure 1.3). That is, during plan creation, the user will interact in real-time with a proxy software system. This method is referred to as Robot-Proxy Grounding.

Because Robot-Proxy Grounding takes place during plan creation, it provides crucial feedback to the user and supports transparency without consuming time or resources during plan execution and without requiring additional communication with the robot. Robot-Proxy Grounding improves

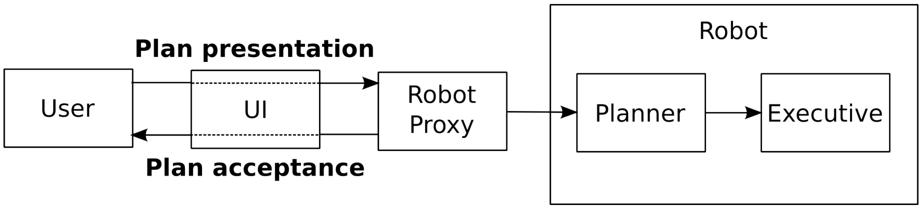


Figure 1.3: A common ground-based model for exploration robotics mission planning in which the user builds common ground with a robot proxy system before the plan is sent to the robot for execution.

upon conventional, conversation-based grounding between two people since that requires the availability of both parties. By utilizing a proxy, it is possible to recreate the behavior of the robot without the actual robot being a part of the conversation. This enables people who interact with a remote robot to understand how the robot would respond to their requests and provides the immediate feedback so critical to the grounding process.

In order to accomplish this, this work introduces a representation of user goals which integrates actions as well as relationships between actions. Given that an autonomous robot has a means to represent user goals, the robot can then check whether the user goals are physically possible and internally consistent; eventually, the results of this analysis could be used at execution time to improve task performance.

As the user interacts with the proxy according to the Robot-Proxy Grounding method, this interaction promotes common ground between the user and robot. This allows the user to develop a more accurate mental model of the robot's state and capabilities while providing the robot with information about the user's higher-level goals.

Uncertainty. It is important to note that there are a number of factors which create uncertainty as the user and robot work to complete their joint task:

- The user may not completely understand or integrate all of the information which is presented to him/her by the robot. This work assumes that the user understands the information presented to him/her, although this may not be the case. However, the fact that the grounding process is iterative increases the probability that the information will be received eventually. A probabilistic model representing the likelihood that user has obtained a particular piece of information would be more accurate although that is beyond the scope of this thesis.

- The robot's sensors or motors may fail. This thesis is primarily concerned with movement and sensor deployment which occurs after the robot has stopped at a location of scientific interest, known as a locale[1]. The information which the robot proxy acquires could be used to help mitigate the effects of these kinds of failures, although this is beyond the scope of this work.

1.3 Contributions

This thesis makes four key contributions to the field of robotics:

- **Detailed analyses of human-robot systems, adding to the body of knowledge of how these systems function and what problems still need to be addressed.** Careful examinations of the Personal Exploration Rover museum exhibit and the Life in the Atacama remote exploration robotics project have demonstrated how a lack of common ground can result in errors, miscommunications, and inefficiencies in human-robot tasks.

- **The application of common ground theory to exploration robotics through the development of the concept of Robot-Proxy Grounding.** This thesis introduces the concept of Robot-Proxy Grounding and explains the implementation of a proxy consisting of a goal representation, a goal validation system, and a robot model. A proof-of-concept study is used to demonstrate that the use of Robot-Proxy Grounding improves task efficiency in an example exploration robotics task.

- **A goal representation for a specific domain, which may later be generalized to other types of HRI problems.** This representation is primarily based on observations of the scientists working on the Life in the Atacama project. The goal representation focuses on the exploration strategy and constraints which were of interest to the Life in the Atacama science team.

- **An implementation of a robot proxy system which has been demonstrated to improve task performance for an exploration robotics task.** This thesis includes an implementation of a robot proxy and an interface to support Robot-Proxy Grounding for developing plans for exploration robotics missions. A user study demonstrates that the use of the robot proxy results in improved efficiency on an exploration task and higher-quality data as well as improving users' common ground with the system and engendering stronger feelings of collaboration with the system.

---

[1] See the Glossary for further information.

1.4 Overview of Thesis

Chapter 2 introduces the research domains of human-robot interaction (HRI), exploration robotics, and common ground theory and describes work at the intersections of these domains. Chapter 3 presents studies of the Personal Exploration Rover system and the Life in the Atacama project, in which tenets of common ground theory are utilized in order to better understand these exploration robotics systems. Based on this analysis, Chapter 4 proposes a division of common ground more specifically tailored to the issues which arise in human-robot collaboration. Chapter 5 includes an analysis of the key characteristics of the process by which a user and robot build common ground in the exploration robotics domain and introduces the Robot-Proxy Grounding process. Chapter 6 presents the results of a preliminary study designed to determine the impact of a robot proxy on a basic exploration robotics task, which indicated that participants who were provided with a robot proxy performed more efficiently on the task and felt a stronger sense of collaboration with the system than participants without a proxy. Chapter 7 presents the requirements which a robot proxy must meet in order to participate successfully in Robot-Proxy Grounding. Chapter 8 describes the design and implementation of each of the three major components of the robot proxy: the goal representation, the robot model, and the goal validation system. Chapter 9 describes an extension to the proof-of-concept study in which participants utilized a complete robot proxy implementation in order to conduct an exploration robotics task. The results of this user study are presented in Chapter 10. The most significant findings indicated that participants who utilized the robot proxy performed more efficiently, collected higher-quality data, and established greater common ground with the robot. This study validates Robot-Proxy Grounding as a means to improve efficiency in exploration robotics tasks. The thesis concludes with a review of key contributions, an examination of how Robot-Proxy Grounding may be utilized in other human-robot interaction problems, and a discussion of future work to further improve HRI for exploration robotics.

Chapter 2

Background

This chapter introduces the research domain of human-robot interaction (HRI), the problem domain of exploration robotics, and work in HRI which has focused on exploration robotics. This chapter also introduces common ground theory, the framework which this work utilizes in order improve human-robot interaction in exploration robotics. Finally, this chapter examines other projects which have utilized common ground theory, both in human-computer and human-robot interaction.

2.1 Human-Robot Interaction

Goodrich and Schultz define Human-Robot Interaction (HRI) as “a field of study dedicated to understanding, designing, and evaluating robotic systems for use by or with humans” [Goodrich and Schultz, 2007, p. 204]. They argue that in order for a human and robot to be interacting, they must be communicating with one another in some way. The human and robot may be collocated with one another, such as the robot *Kismet* [Breazeal, 2003], or separated by time and/or space, such as Urban Search and Rescue (USAR) robots which are sent inside of collapsed buildings ([Burke et al., 2004]).

Within these kinds of task-oriented problems, the goal of HRI research can be summarized as “to build better HRI and to better the performance of human-robot teams“ [Steinfeld et al., 2006, p. 33]. Steinfeld et al. propose a number of metrics which can be used to measure progress toward this goal. When considering an entire human-robot system, Steinfeld et al. propose the following common metrics across tasks [Steinfeld et al., 2006]:

- System performance (measures of quantitative performance, subjective ratings of performance, and appropriate regulation of human-robot autonomy)

- Operator (human) performance (measures of situation awareness, workload, and accuracy of mental models of device operation)

- Robot performance (measures of self-awareness, human awareness, and autonomy)

According to this paradigm, the quality of human-robot interaction of a system can be measured by considering the entire system, the human point of view, and the robot point of view. The goal of task-oriented HRI research is to improve performance with respect to all of these perspectives.

This is the paradigm of high-quality HRI that will be adopted for the purpose of this thesis. In particular, this work focuses on overall system performance through an emphasis on improving task efficiency, and it focuses on operator performance through an emphasis on improving the user's mental model of the robot.

2.2 Exploration Robotics

While there are many robots that have been designed specifically for use in human-robot interaction studies (see [Fong et al., 2002] for a survey), this thesis focuses primarily on the domain of exploration robotics; robotic exploration tasks are defined broadly as those in which a robot co-investigates an unknown environment with a remote human partner. Exploration is an important domain of study because of its applicability to a wide variety of problems, which range from searching for signs of life on other planets to investigating debris after a building collapse.

Exploration robotics tasks represent an interesting and challenging problem domain due to a number of constraints which must be considered in the design and deployment of such systems:

- **Energy.** Robots must be able to use energy resources wisely. This may involve planning paths to maximize the sunlight on solar panels or reducing the number of actions to be executed in order to conserve battery power.

- **Time.** Robots may be operating under time constraints, having only a limited amount of time available to meet their goals. For example, a solar-powered robot may need to reach a certain location before the sun sets. Urban search and rescue robots must locate disaster victims quickly enough to ensure that they receive proper medical attention.

- **Bandwidth.** The amount of bandwidth available for communication to and from the robot may be limited.
- **Remote operation.** Robots may need to work at a distance from their human operators; the operators may not be able to observe the robots directly as the robots perform their tasks.
- **Asynchrony.** In some exploration robotics tasks, the human and robot do not interact directly in real-time; instead, there may be a time delay between when a command is sent to the robot and when the robot receives and executes it. Depending on the task, this delay may be only a couple of seconds, or, in more extreme cases, hours or days.
- **Unknown environment.** Robots must be able to operate with little, if any, prior information about the locations of obstacles or other objects in the environment. Robots must be able to collect and interpret information about the changing world around them.
- **Cost.** Deploying robots to work autonomously in remote, harsh environments is very expensive.

In order to address these challenges, robots are becoming increasingly technologically capable and autonomous. However, it is ultimately people who must command robotic explorers and must interpret the data that these robots return: advances in robotic software and hardware have little benefit if they cannot be used effectively by humans. Understanding and improving the human-robot interactions between people and their robotic co-investigators is therefore crucial to the continued improvement of the field of exploration robotics as a whole.

This is particularly true given that robotic exploration systems are increasingly designed not to be operated by robotics experts alone, but by non-roboticists who have expertise in the domain under investigation. Either by directly controlling the robots (i.e. rescue workers who are trained to operate USAR robots) or by working in close collaboration with roboticists (i.e. scientists who command NASA Mars rovers in conjunction with engineers), non-roboticists are being given a larger role in exploration robotics missions. Robotic explorers must be able to collaborate effectively with and meet the needs of these domain experts in order to successfully complete their tasks.

The following section presents previous research on human-robot interaction for exploration robotics followed by an introduction to common ground theory, which forms the basis for the approach used in this work to improve HRI for exploration robotics.

2.3 Human-Robot Interaction in Exploration Robotics

This thesis focuses on exploration which involves the deployment of autonomous robots that work in complex, real-world settings. As described above, in these situations, system users are not likely to be experts in robotics, and they may possess inaccurate mental models of robotic technologies. At the same time, these users often do possess sophisticated domain knowledge which has not been programmed into the robot. In order to facilitate successful exploration, this thesis focuses on promoting shared understanding between users and robots: that is, the goal of this work is to increase users' understanding of robots and foster accurate mental models, and, at the same time, enhance robots' understandings of users and their goals in order to drive robots' decision-making processes. HRI work in exploration robotics domains such as space robotics, urban search and rescue, and educational robotics informs this thesis.

2.3.1 Space Robotics

Most HRI work in space robotics has tended to focus on robots that work in close proximity with astronauts. Burridge et al. conducted a study comparing the performance of geologists with and without a robot teammate and found the robot to have a detrimental effect [Burridge et al., 2003]; however, it is important to note that the robot used in the experiment had virtually no artificial intelligence or reasoning to allow it to assist the people working with it. In 2005, NASA conducted a demonstration involving a team of three humans and two heterogeneous robots interacting to complete a space construction task [Fong et al., 2006]. Two astronauts in suits worked with the two robots outside of a "habitation," in which a third astronaut helped coordinate the construction project. The Human-Robot Interaction Operating System developed to support the task focuses on natural language, task-oriented dialog. This thesis is more applicable to systems in which people may not be collocated with a robot, and the robot must execute tasks while receiving little or no input from people. In addition, none of these projects involves modeling common ground explicitly or specifically working to promote common ground, both contributions of this thesis. Focusing on the grounding process allows this thesis to support an ongoing "conversation" between a scientist and a robot without the use of real-time, natural language interaction.

2.3.2 Urban Search and Rescue

Another relevant domain is that of HRI in Urban Search and Rescue (USAR). In this domain, one or more operators control a robot which must be maneuvered to collect information and search for survivors in urban environments after a disaster. Murphy provides an overview of HRI issues relevant to USAR in [Murphy, 2004], although this focuses entirely on issues stemming from teleoperation. A number of studies have been conducted on USAR operations, from examinations of the Robocup Rescue competition [Yanco et al., 2004; Scholtz et al., 2004] to observational studies of USAR training exercises [Burke et al., 2004; Burke and Murphy, 2004]. While these studies provide insight into what information operators need and how it can best be presented, the robots involved are all teleoperated. This thesis focuses on allowing autonomous robots to actively promote grounding with users and obtain explicit representations of users' goals.

2.3.3 Educational Robotics

This thesis is also informed by previous work in educational robotics; of particular interest are the educational robotic platforms developed by Nourbakhsh et al. In [Nourbakhsh et al., 2003], the authors describe a study of the educational impact of a robot autonomy course for high schoolers; beyond learning the nuts-and-bolts of programming a robot, students also learned valuable lessons about teamwork and were able to better identify with technology. The Personal Exploration Rover [Nourbakhsh et al., 2004] is another platform which has been evaluated for its educational impact [Nourbakhsh et al., 2005]. These studies demonstrate the importance of understanding how people change as a result of interacting with robots. They also present techniques which were later used as part of the foundation of this thesis (see Section 3.1). However, understanding how people think about robots is only a partial solution: to establish common ground between a person and a robot, the robot must be able to represent some aspect of this knowledge.

This thesis supports the building of common ground by allowing the robot to represent user goals and provide feedback on them to the user.

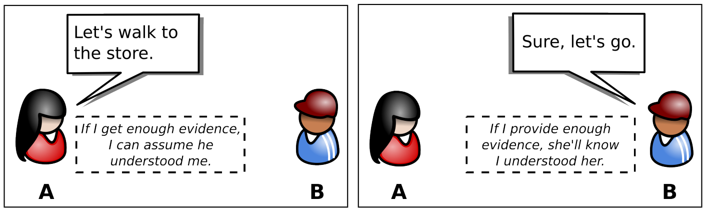
These applications represent the dimensions of research in human-robot interaction that have had the greatest impact on Robot-Proxy Grounding. While these studies are concerned with how well a user is able to perform an exploration-related task or what a user learns from such a system, Robot-Proxy Grounding contributes to the field of human-robot interaction by explicitly representing common ground and promoting the grounding process between a robot and a user.

2.4 Common Ground Theory

Robot-Proxy Grounding is primarily based on the theory of common ground, a communication theory developed by Herbert H. Clark and colleagues. Common ground theory was originally developed to describe verbal communication between people, but it has since been extended to a variety of different situations and communication modalities. A related theoretical framework, situation awareness, has also been applied to HRI problems. Because common ground theory and situation awareness overlap, it is important to highlight how promoting common ground is beneficial for users collaborating with remote, autonomous robots in order to complete exploration tasks.

As two individuals participate in a joint activity, they accumulate common ground, “the knowledge, beliefs, and suppositions they believe they share about the activity” [Clark, 1996, p. 38]. For example, the common ground between two rescue workers at a disaster site would include the knowledge of how to properly search a building for survivors, the knowledge of who is currently inside the building, and the knowledge of how many survivors have already been located.

Clark and his colleagues propose that common ground is required for successful collaboration—it helps collaborators to know what information is needed by their partners, how to present information so that it is understood, and whether or not the information has been interpreted correctly [Clark and Wilkes-Gibbs, 1986; Clark and Marshall, 1981]. At the start of an interaction, collaborators share a certain amount of common ground. For example, if they are members of the same discipline or work group, they likely have a common language and perspective that provides common ground and facilitates communication [Fussell and Krauss, 1992]. Common ground can be developed over time as collaborators share common experi-


Let's walk to the store.
If I get enough evidence, I can assume he understood me.
A
B
Sure, let's go.
If I provide enough evidence, she'll know I understood her.
A
B

(a) Presentation Phase: Contributor A conveys information to partner B under the assumption that if A receives a certain amount of evidence from B, A can believe that B understands what A means.

(b) Acceptance Phase: B accepts the information from A by giving evidence of B's understanding; B assumes that once A receives this evidence, A will recognize that B understands what A means.

Figure 2.1: A simple example of the presentation-acceptance process.

ences [Clark and Brennan, 1991], but it also can be disrupted by factors such as being located in and drawing information from different physical contexts [Cramton, 2001].

The interactive process by which common ground is established is referred to as "grounding." More specifically, Clark and Brennan describe grounding as follows [Clark and Brennan, 1991, p. 129]:

> The contributor and his or her partners mutually believe that the partners have understood what the contributor meant to a criterion sufficient for current purposes. This is called the *grounding criterion*. Technically, then, grounding is the collective process by which the participants try to reach this mutual belief.

Clark and Brennan argue that grounding takes place through an exchange called the presentation-acceptance process [Clark and Brennan, 1991]. This is a two-phase process by which a contributor (A) conveys information to his or her partner (B) (see Figure 2.1). First is the presentation phase: A presents some information to B. A does this under the assumption that if A receives a certain amount of evidence from B, A can believe that B understands what A means. This is followed by the acceptance phase, in which B accepts the information from A by giving evidence that B understands what A means. B assumes that once A receives this evidence, A will recognize that B understands what A means [Clark and Brennan, 1991].

2.4.1 Empirical Validation

The theory of common ground has been validated and extended through numerous psychological studies. The studies most related to this thesis are those that have focused on establishing the nature of the grounding process between pairs of people; this thesis also builds on studies which have examined distributed collaboration—how individuals build common ground when they are not collocated. A selection of papers on both of these topics is presented below.

The Grounding Process

Clark and Wilkes-Gibbs conducted an experiment in which two participants, a "director" and "matcher", were seated at a table separated by an opaque barrier [Clark and Wilkes-Gibbs, 1986]. Each participant was given a set of twelve cards depicting Tangram figures (images created using simple geometrical shapes). The director's cards were arranged in a specific order whereas the matcher's cards were arranged randomly. The director was instructed to describe the cards to the matcher, who would attempt to put his cards in the same order as the director's as quickly as possible. This process was repeated six times. Clark and Wilkes-Gibbs analyzed the conversations of the director-matcher pairs in order to determine how they referred to the figures and how those references changed over time. Since the Tangram figures were abstract, geometrical shapes, there was no obvious way to refer to any particular figure. The experimenters hypothesized that directors would use longer explanations the first time a figure was discussed, but these explanations would become shorter over time as more common ground was established. The data supported this hypothesis. In addition, the experimenters found that after a director initially presented a noun phrase (description) referring to a figure, either participant might repair, expand on, or replace the noun phrase so as to minimize the joint effort required in order for the noun phrase to be mutually understood. This thesis uses the mutual acceptance process as a basis for the interaction between the robot proxy and user during plan creation.

Issacs and Clark examined how experts and novices refer to objects in conversations [Issacs and Clark, 1987] and found that the expert and novice first assess each other's expertise. The expert then provides specialized terminology to make references more efficient, and the novice acquires and uses this new, specialized knowledge. For this study, the experimenters asked pairs of participants who were or were not experts about New York City landmarks to arrange pictures of such landmarks by talking about them. As participants began conversing, they quickly determined which partner had more expertise. This partner supplied terminology, such as the names of buildings, which both partners then used when referring to the buildings later. This study demonstrates "that people accommodate to each other quickly and automatically in the process of making themselves understood" [Issacs and Clark, 1987, p. 36].

Brennan and Clark also focused on how people refer to objects in [Brennan and Clark, 1996]; in particular, they studied how people's references to objects change over time. In three experiments, a "director" described cards with pictures to a "matcher," who had to place his cards in the same order as the director's as quickly as possible. Once the pairs had established a reference to a particular picture, they tended to use that same reference in later trials even when a shorter reference would have sufficed. This demonstrated the importance of conversation history in determining how people select references to use.

More recently, Clark and Krych examined how speakers monitor their listeners for evidence of understanding [Clark and Krych, 2004]. In this experiment, a "director" instructed a "builder" on how to construct a Lego model. In one condition, directors could see the builders' workspaces; in another, they could not; and in a third, directors recorded an audio tape which was later played to builders. When directors could see what the builders were doing, builders communicated using nonverbal behaviors such as pointing at blocks and nodding or shaking their heads; directors observed these behaviors and modified their instructions mid-sentence as a result. Builders were much faster and more accurate in completing the task when they could be observed by directors. This study illustrates the challenges faced by partners who are not able to communicate by nonverbal cues.

Distributed Collaboration

Common ground can be developed over time as collaborators share common experiences [Clark and Brennan, 1991]; however, prior work suggests that people may not establish common ground at all when they are geographically separated.

Cramton [Cramton, 2001] documented the challenges of distributed collaboration in her study of graduate students' working on a group project in 6-person teams distributed across three continents. Through an analysis of communications between team members, including e-mails, online chat logs, and analysis papers written by the students, Cramton identified five types of “failures of mutual knowledge”: “failure to communicate and retain contextual information, unevenly distributed information, difficulty communicating and understanding the salience of information, differences in speed of access to information, and difficulty interpreting the meaning of silence” [Cramton, 2001, p. 346].

This thesis focuses primarily on the problems of missing contextual information and the difficulty of interpreting the meaning of silence. According to Clark and Brennan, missing contextual information jeopardizes shared understanding because “the addressee has to imagine appropriate contexts for both the sender and the message” [Clark and Brennan, 1991, p. 143]. Cramton argued that difficulties in establishing mutual knowledge arise when team members have difficulty sharing and remembering information about the contexts in which remotely-located collaborators are working. Contextual information includes a wide variety of information that may not be directly related to the task at hand, including anything from national holidays to pressures from supervisors. Within the context of remote exploration robotics, important contextual information might include the weather at the robot’s location or the hypotheses that the science team is trying to verify. Previous studies have documented some of the problems which can result from missing contextual information in [Stubbs et al., 2006b].

Another common ground problem that Cramton identified which is relevant to this thesis was the difficulty of interpreting the meaning of silence. In distributed teams, the communication media most easily accessible to collaborators (e.g., email, instant messaging, etc.) do not generally support the subtle nuances that people rely on to resolve the meaning of silence. People tend to remain quiet rather than try to resolve problems using these technologies. As applied to robotic exploration, when scientists do not receive the data which they expect the robot to return, they struggle to interpret the meaning of this “silence” [Stubbs et al., 2006b]. Is the data missing because an instrument had broken, or because the robot could not reach the desired location, or for some other reason?

More recently, Kraut, Fussell, and Siegel have examined how visual information can be used to build common ground between participants who are collaborating on a physical task [Kraut et al., 2003]. In this work, the researchers describe experiments in which two participants collaborate to repair a bicycle. One person, the “worker”, was responsible for performing the physical labor needed to repair the bicycle; a second participant, the “helper”, provided instructions and assistance. Two experiments were conducted: one in which workers wore a head-mounted video system, which allowed helpers to see the workers’ hands and actions, which was compared against an audio-only condition; in the second experiment, the head-mounted video system was compared against a side-by-side condition in which the helpers and workers were collocated.

The results indicated that workers performed more efficiently when the helper was physically present with them, although a remotely-located helper was more beneficial than no helper at all. The fact that co-present collaborators can both see the objects relevant to the task, the area around the task, and the behavior of their partners is a major benefit. Analysis of participants' conversations indicated that they used this visual information to make conversational utterances more efficient. The researchers concluded that video-mediated dialogs were less efficient than side-by-side dialogs due to several factors: the video system did not capture important elements such as the workers' facial expressions; making use of the video system required participants to try and establish what objects were or were not part of the shared view; and helpers had no way to gesture to task objects. The study demonstrated that shared visual space helps to facilitate grounding in a number of ways, and that the effectiveness of a video configuration depends on the extent to which it captures key elements of the visual space.

Gergle, Kraut, and Fussell further examined how visual information improves the efficiency of conversations in [Gergle et al., 2004]. In these experiments, pairs of participants worked together to complete an online puzzle. "Helpers" were able to see the completed puzzle and provide instructions, but only "workers" were able to manipulate puzzle pieces. Because the communication was mediated by computers, the experimenters were able to manipulate what portion of the worker's workspace was displayed to the helper on the helper's screen. They also manipulated the rate at which the helper's screen was updated; that is, in some conditions the helpers had a delayed view of what workers were doing. The data indicated that workers changed what they said and did based on what the helper could or could not see; this is consistent with Clark and Brennan's work that people change their grounding strategies based on the costs and affordances of the communication technologies available to them [Clark and Brennan, 1991].

The goal of this thesis is to facilitate this grounding process between a user and a robot such that each can understand the other for the purpose of achieving the user's goals. One of the most significant contributions is the adaptation of the presentation-acceptance process for use in exploration robotics mission planning.

2.4.2 Situation Awareness

In addition to common ground theory, situation awareness is a theoretical framework which has been used to analyze how people work with robots or teammates. Endsley defines situation awareness (SA) as “knowing what is going on around you” [Endsley, 2000]. This includes perceiving elements in the environment, understanding their meaning, and being able to predict their status in the near future [Drury et al., 2003]. Situation awareness-based approaches differ from common ground theory in several ways, including SA’s primary focus on the human operator(s), conceptualization of human-robot interactions as non-symmetrical, and emphasis on real-time, synchronous interactions. By contrast, common ground theory focuses on the ongoing “conversation” between the robot and human over time and how mutual knowledge is created or disrupted.

Within the HRI domain, situation awareness has been examined particularly with urban search and rescue (USAR) robots [Burke et al., 2004; Drury et al., 2003; Yanco et al., 2004]. These studies tend to focus on the information needs of the human operator(s). For example, Burke et al. assessed the situation awareness of USAR operators in a training scenario by answering the following question:

> How well did the operator understand what he was seeing through the robot’s eye-view, what the robot’s state was at any given moment, and how the robot-supplied information related to other operator knowledge concerning the technical search operation? [Burke et al., 2004]

In addition, Casper and Murphy [Casper and Murphy, 2003] found in their study of human-robot teams responding to the World Trade Center disaster that operators’ lack of awareness regarding the state of the robot and how it was situated in the rubble affected the performance of the teams. Empirical work indicates that USAR operators spend significantly more time trying to gain SA—assessing the state of the robot and environment—than they do navigating the robot [Burke et al., 2004; Drury et al., 2003].

Burke and Murphy also note that the SA between two operators working with the same robot can, through communication, become a foundation for common ground; however, they do not test this relationship directly [Burke et al., 2004]. In more recent study, they found that video from the robot itself could also help operators build common ground with each other [Burke and Murphy, 2007]. As described above, these studies of HRI in the USAR domain examine how much SA operators have while using robots and what the process is by which operators obtain SA. Common ground theory emphasizes the information needs of both the human and the robot, the need to establish that they share this information, and the "conversation" through which this mutual knowledge is built.

A second difference between SA and common ground theory involves how the interaction between humans and robots is characterized. In a discussion of awareness in HRI, Drury addresses the "non-symmetrical nature of the human-robot collaboration" [Drury et al., 2003]. In particular, for each of Drury's five types of awareness in HRI (human-human, robot-human, robot-robot, and human overall), different types of information are required. For example, robots must have awareness of the commands which they have received from humans, but only humans must be aware of the overall goals of the joint activity. This thesis argues that errors and miscommunications can arise when humans and robots do not have mutual knowledge with respect to information such as higher-level goals. While the interaction between a human and a robot may never be (or should be) truly symmetrical, common ground theory demonstrates the importance of mutual knowledge for successful joint activities.

It is also important to note that Endsley emphasizes the importance of time in SA, such as an operator's awareness of "how much time is available until some event occurs or some action must be taken" [Endsley, 2000, p. 7]. This is very relevant to work on HRI for USAR, which tends to focus on "real time" interaction (with teleoperated robots); however, its applicability is less clear for HRI with robots that are remotely and asynchronously commanded. Previous common ground research has specifically considered the challenges faced by collaborators who are separated by space and/or time; these constraints on the grounding process and their applicability to remote exploration robotics are discussed in greater detail in Chapter 5.

In summary, previous work in situation awareness for HRI, particularly in the USAR domain, has tended to focus primarily on the information needs of human users as part of a non-symmetrical, real-time interaction with robots. By contrast, common ground theory emphasizes the "conversation" which must take place between a user and robot in order for them to build mutual knowledge while addressing the challenges of remote, asynchronous interaction.

2.5 Applications of Common Ground Theory

Common ground theory has been applied both to human-computer interaction and, more recently, to human-robot interaction problems; this section describes relevant work in these areas and contrasts it with the approach utilized in this thesis.

2.5.1 Common Ground and Human-Computer Interaction

Although the common ground framework was developed to understand conversation and collaboration among people, not between people and machines, recent work has extended the framework into the field of human-computer interaction [Brennan and Hulteen, 1995; Paek and Horvitz, 1999]. This research suggests that interfaces can be improved by thinking about the users' experience as a conversation in which shared meaning between the user and the interface must be developed. By ensuring that common ground can be constructed incrementally, users have more information about what has and has not been understood and can correct accordingly [Brennan and Hulteen, 1995]. Studies of the Life in the Atacama project have demonstrated how a lack of common ground contributed to problems experienced by the science team and the engineering team (Chapter 3). By encouraging the grounding process between a human and robot, the likelihood that their collaboration will be successful is increased. The use of Robot-Proxy Grounding is a first step towards remedying these types of problems.

2.5.2 Common Ground and Human-Robot Interaction

Kiesler has described experiments reporting more effective communication between people and robots when common ground is greater [Kiesler, 2005]. Other researchers have found that information exchange is more effective when a robot can adapt its dialog to fit a user's knowledge [Torrey et al., 2006].

Severinson-Eklundh and her colleagues used common ground theory as a basis for the dialog system used by *Cero*, a service robot designed to perform fetch-and-carry tasks in an office environment [Severinson-Eklundh et al., 2003]. *Cero* uses natural-language and gestures to communicate with users. The dialog system utilizes a "cautious grounding strategy aimed at assuring that the user is certain about what instructions the robot has received and is about to carry out" [Severinson-Eklundh et al., 2003, p. 11]. If the robot receives a command that is only partially understood by the dialog system, it will ask for clarification; if the user provides an acceptable clarification for the unclear part of the command, the rest of the command is assumed to be part of the common ground. *Cero* also provides feedback using speech and gestures corresponding to different depths of grounding according to the work of Brennan and Hulteen [Brennan and Hulteen, 1995]. For example, to indicate that *Cero* can hear the user (a relatively shallow level of grounding), the robot raises its head toward the user. To indicate that the robot is reporting on text execution (a relatively deep level of grounding), the robot makes a "walking" gesture and states that it is executing the task. This emphasis on building common ground through gestures and spoken, natural-language interaction is not necessarily applicable to the exploration robotics domain. Many exploration robotics systems do not incorporate natural-language dialog nor do these robots have the capability to gesture. Robot-Proxy Grounding adapts common ground theory to this domain by conceptualizing the planning process itself as a communication method and the use of presentation-acceptance phases to promote common ground as planning proceeds.

Burke and Murphy conducted a study during a USAR training scenario in which they gave team members video taken from the robot's point of view, arguing that "the robot can serve as a source of common ground for the distributed team" [Burke and Murphy, 2007]. They found that the use of RSVP (remote shared visual presence) predicted team performance, but its efficiency may vary based on the users' experience and team cohesion. This study demonstrates how a robot can provide information which helps improve users' mental models of a given situation. This thesis assumes that team members are collocated with each other and must build common ground with the robot in order to successfully accomplish exploration tasks.

Li et al. have introduced a dialog system for mobile robots that utilizes presentation and acceptance phases, but it primarily focuses on natural language conversation and face-to-face interaction [Li et al., 2006]. By contrast, Robot-Proxy Grounding extends common ground theory to situations involving remote collaboration which do not necessarily utilize natural language dialog. Robot-Proxy Grounding promotes the grounding process with respect to a user's goals through mission planning. By encouraging the grounding process between a human and a robot, the likelihood that their collaboration will be successful is increased.

A summary of how Robot-Proxy Grounding differs from the most relevant research discussed in this chapter is shown in Table 2.1. As illustrated in this table, most applications of common ground theory with respect to HRI have focused on social interactions in which the robot and human are

Related Work	Common Ground	Use of Robot	Remote Collaboration	No Natural Language	Exploration Task
Brennan and Hulteen [1995]	X				
Paek and Horvitz [1999]	X				
Burridge et al. [2003]		X		X	X
Severinson-Eklundh et al. [2003]	X	X			
Burke and Murphy [2004]		X	X	X	X
Murphy [2004]		X	X	X	X
Scholtz et al. [2004]		X	X	X	X
Siino and Hinds [2004]	X	X			
Yanco et al. [2004]		X	X	X	X
Kiesler [2005]	X	X			
Fong et al. [2006]		X			
Torrey et al. [2006]	X	X			
Li et al. [2006]	X	X			
Moshkina et al. [2006]		X	X	X	X
Robot-Proxy Grounding	**X**	**X**	**X**	**X**	**X**

Table 2.1: Robot-Proxy Grounding contrasted with other related work.

collocated and engage in natural language conversations. The Mission Repair Feature of the *MissionLab* planning tool is similar to the implementation of Robot-Proxy Grounding presented in this thesis [Moshkina et al., 2006]. This tool provides a graphical user interface designed to support the manual creation of missions for autonomous robots; example tasks include moving to a location or surveying a room. After the user has specified a mission, he can "play back" the plan and observe the robots' behavior. If the user observes erroneous behavior, he can use the Mission Repair Feature. This feature guides the user to identify and correct errors in the plan. Moshkina et al. demonstrate through user studies the effectiveness of this technique in reducing the number of unsuccessful missions and increasing ease of use of the software. However, this technique places the burden of identifying errors largely on the user. Robot-Proxy Grounding focuses on building shared information between the robot and user. This allows both parties to collaborate to ensure that plans meet the user's goals.

2.6 Summary

In order for two individuals to collaborate successfully on a joint task, they need common ground. As the individuals participate in the task, they gradually build common ground through the presentation-acceptance process. This thesis focuses on a human and an autonomous robot collaborating on an exploration task, a problem relevant to space robotics, urban search and rescue, and educational robotics. Common ground theory has been applied to human-computer interaction as well as human-robot interaction. Robot-Proxy Grounding builds upon common ground theory through an emphasis on promoting common ground between a person and a robot who are collaborating remotely without natural language-driven interaction.

Chapter 3

Foundations of Robot-Proxy Grounding

To date, I have conducted studies of two different human-robot systems: the Personal Exploration Rover museum exhibit, conducted in conjunction with Debra Bernstein, Kevin Crowley, and Illah Nourbakhsh, and the Life in the Atacama project, conducted in conjunction with Pamela Hinds and David Wettergreen. The results of these studies, in particular the observations of the Life in the Atacama project, serve as the foundation of this thesis.

This chapter provides a brief introduction to the methodology which was utilized in these studies, followed by a more detailed description of each project and a discussion of how the results form a foundation for Robot-Proxy Grounding.

3.1 Research Methodology

The Personal Exploration Rover and Life in the Atacama system studies which form the foundation of this thesis are mainly qualitative studies of human-robot systems. This section provides background information on the methodological techniques used in this thesis: observational studies and the coding methods used in social science work.

3.1.1 Observational Studies

Turkle documented the process by which children develop a relationship with computers through ethnographic observations and interviews [Turkle, 1984]. Turkle found that children first philosophize about the nature of computers, then desire to master them, and then construct identities with respect to them. Similarly, this thesis is based upon how people understand and use robots as documented through ethnographic observation (described in detail in Section 3.3).

Observational studies have been used in a wide variety of HRI domains. As described above, observational studies have been conducted on USAR operations, including studies of the Robocup Rescue competition [Yanco et al., 2004; Scholtz et al., 2004] and studies of USAR training exercises [Burke et al., 2004; Burke and Murphy, 2004]. More recently, Drury et al. have conducted an observational study of uninhabited aerial vehicle training [Drury et al., 2006]. These studies have demonstrated the value of careful observation of users outside of a research laboratory as they interact with robots to solve real-world problems.

Other ethnographic observations of human-robot systems include work by Jones and Hinds, who studied SWAT teams and used their findings to develop a system for coordinating distributed robots [Jones and Hinds, 2002]. Siino and Hinds conducted an ethnographic analysis of a community hospital before an autonomous mobile robot arrived to be used at the hospital and documented how different groups of hospital employees made sense of the new technology [Siino and Hinds, 2004]. These groups interpreted the robot and its use in a variety of ways; the authors argue that people become committed to these interpretations and that new organizational structures will form as a result. The methodology of these studies was utilized to collect data about a particular human-robot system (the LITA project); in particular, the analysis focused on how common ground is created or disrupted between people and robots. This data serves as a basis for the development of Robot-Proxy Grounding.

3.1.2 Coding Schemes

Once ethnographic observations or interviews have been conducted, the development of a coding scheme allows for chunks of data (such as lines of an interview transcript) to be categorized. For example, in work involving the Personal Exploration Rover, lines of interview transcripts were labeled with categories such as "Anthropomorphism" and "Role of Robot" according to the topic of each utterance (see Section 3.2 for more details). This technique has a long history of use in the behavioral sciences [Strauss and Corbin, 1998]. Within human-robot interaction, this technique has been used by Nourbakhsh et al. in their work with educational robotics [Nourbakhsh et al., 2003, 2004]. The development of a coding scheme for categorizing types of utterances with respect to learning and museums can be found in [Schauble et al., 2002] and [Crowley and Jacobs, 2002]. This type of coding process helps us to convert qualitative data to quantitative data, to identify patterns in qualitative data, and to interpret how people think about robotic systems.

These two different types of research methodologies represent the primary methods by which the data that forms the basis of Robot-Proxy Grounding was collected and analyzed. These research methods are not commonly used within the robotics community, at least in part because most roboticists have been trained in computer science or engineering as opposed to the social sciences. Because these methods emphasize users' experiences and knowledge, they are well-suited to examining problems relating to common ground and the grounding process. It is hoped that this thesis will further encourage the use of these methodologies for examining human-robot systems. The two projects which were analyzed for this thesis are the Personal Exploration Rover museum exhibit (Section 3.2) and the Life in the Atacama Project (Section 3.3).

3.2 The Personal Exploration Rover

The Personal Exploration Rover (PER) is the third rover designed and built as part of the Personal Rover Project [Nourbakhsh et al., 2004]. The goal of this project is to design and build interactive robots capable of educating and inspiring children. The PER was designed as a tool to educate the public about certain aspects of NASA's Mars Exploration Rover (MER) mission. The goals of the PER are to demonstrate to the public that rovers are tools used for doing science and to illustrate the value of on-board rover autonomy.

3.2.1 System Description

Physically, the PER is reminiscent of the MER in its overall mechanical design (Figure 3.1). The PER is a six-wheeled robot that uses a Rocker-Bogie suspension system similar to that used on the MERs. The PER is equipped with a camera and infrared range finder mounted on a pan-tilt head as well as an ultraviolet light for conducting simulated scientific testing.

The PER museum exhibit consists of a PER deployed inside a simulated Martian environment (the "Mars yard") complete with several large rocks as "science targets" and an interactive kiosk, equipped with a trackball and a single button. The premise of the exhibit is that visitors will use the robot

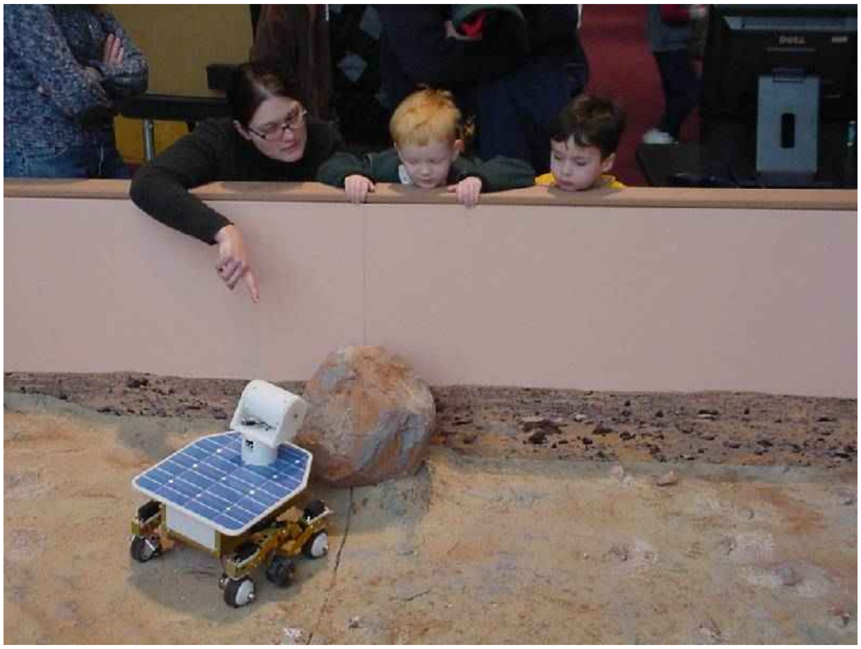


Figure 3.1: The Personal Exploration Rover (PER) at a museum installation.

to search for life within the Mars yard. The robot is able to test for signs of life using a simulated organofluorescence test, in which the robot shines a UV light on a rock. As the robot conducts the test, it sends a picture of the rock back to the kiosk, where visitors look for a "glow" indicating the presence of (simulated) organic material.

There are a three different groups of individuals who have had interactions with the PERs since the PER project began. These are the creators of the PERs at Carnegie Mellon University, museum employees at the PER installation sites, and the museum visitors who use the PER exhibit (Figure 3.2). Bernstein et al. [Bernstein, 2004] conducted a study of how visitors interact with and react to the PER exhibit, but these interactions rarely last more than several minutes. Museum employees, including administrators, explainers, and technical support people, were chosen to be the focus of this study due to their regular interactions with the PERs over a period of months. These interactions include setting up the PERs at the start of the day, changing their batteries, diagnosing and repairing problems, and talking about the PERs and their exhibit to museum visitors. In addition, museum employees together form a group of naïve initial users who will

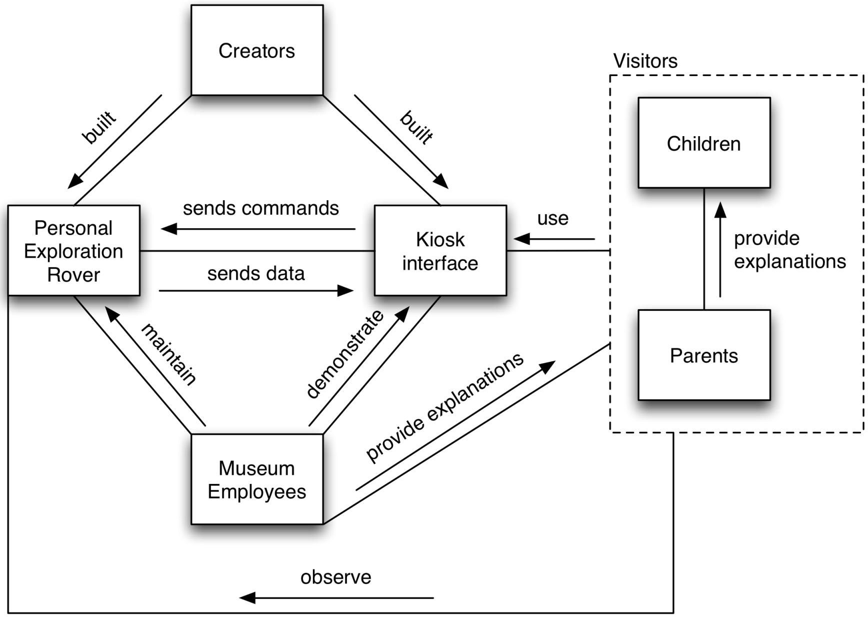

Creators
Visitors
built
built
Children
sends commands
use
Personal Exploration Rover
Kiosk interface
provide explanations
sends data
maintain
demonstrate
Parents
provide explanations
Museum Employees
observe

Figure 3.2: Collaboration diagram of all members of the PER system and their interactions [Stubbs et al., 2006a].

learn over time and develop cognitive models that they initially may not have had. These two characteristics make them a group well-suited for a study of long-term human-robot interaction.

3.2.2 Long-Term Human-Robot Interaction

Method

For this study, the goal was to develop a methodology that would enable me to answer the following types of questions about employees' cognitive models of the PER:

- How does the employee's conception of robot intelligence change over weeks of interaction?
- How do employees anthropomorphize the robot, if at all?
- As employees gain more experience working with the robot, how do their descriptions of its capabilities change?
- How do employees see the connection between the PER and the MER?

In order to answer these questions, museum employees were interviewed periodically from December 2003 through June 2004. These open-ended interviews were conducted once before the PER exhibit had been installed, one to two weeks after the exhibit had been installed, approximately one and a half months after installation, and approximately three and a half months after installation. The exact questions asked to employees at each interview varied slightly, but each employee had an equal opportunity to comment on all question topics. Eighteen museum employees at four PER installations were interviewed; of these, only eleven were able to complete the first three interviews. The code running on-board the PER and on the kiosks remained essentially unchanged throughout the course of the study.

After the interviews were transcribed, a coding scheme was designed to reflect the museum employees' thoughts about the robot (a complete description is available in [Stubbs et al., 2005]). This coding scheme was used to categorize what the employees were talking about (i.e., the Reliability code was used to label statements relating to the reliability of the PER, but the presence of this code does not necessarily indicate employees felt the PER was more reliable than other exhibits).

Results

All together, the forty-four interview transcripts contained 2,821 lines of text. Each line that was transcribed was assigned a particular content code. The data from the eleven employees who completed the first three interviews were used to compute matched-sample statistics. The distribution of content codes is shown in Figure 3.1.

The fact that there were many significant changes in employees' talk about the PERs between the first and second interviews suggests that regular interaction with a robot for even two weeks has a large impact on a person's cognitive model. However, the only content code that increased across all three interviews was Anthropomorphization. In addition, talk about anthropomorphization was significantly positively correlated with talk about visitors, reliability, and intelligence ($N = 44$, $p < 0.05$, $p < 0.01$, and $p < 0.01$, respectively).

Based on the changes that were observed in this study, some of the key factors that should be considered when constructing a cognitive model of how people understand robots include:

- A robot's actual failures and successes may be more important than its purported capabilities. In order to aid people in developing accurate

	Interview		
Code	*1*	*2*	*3*
Reliability	1.1%	*7.1%	6.1%
Anthropomorphization	1.1%	*10.1%	18.4%
Intelligence	1.7%	*6.4%	4.3%
Different POV	7.1%	4.0%	0.5%
MER mission	11.1%	8.5%	*4.3%
Role of robot	12.2%	4.1%	0.5%
Capabilities	14.5%	10.9%	13.5%
Failures	17.0%	17.3%	16.7%
Visitor description	34.1%	*31.5%	35.8%

Table 3.1: For each interview and content code, the value listed is equal to the ratio of the number of times that that content code was used out of the total number of lines coded. *Indicates a statistically significant change (one-way repeated-measures ANOVA). [Stubbs et al., 2005]

cognitive models, it is best to keep robot behavior transparent (it should be obvious to users what the robot is doing). Providing this transparency into the robot's successes and failures will allow users to develop the best possible cognitive model, one based on their own experiences rather than on extensive pretraining.

- Anthropomorphism is a broad concept, and there was a significant positive correlation between talk about anthropomorphism and a number of other concepts, such as reliability. While it is clear that anthropomorphization is an important part of a person's cognitive model of a robot, exactly what role anthropomorphism plays in that model remains an open question.

- Talk about higher-level concepts, such as the idea of robotic intelligence, declined over time but this decrease was matched by an increase in talk about anthropomorphism. This suggests that people may be thinking of the robot less as a machine and more as a collaborator. A quantitative model of long-term human-robot interaction will need to recognize this distinction between "interactive device as robot" and "interactive device as collaborator" as a person moves from one to the other.

3.2.3 Impact on Formation of Robot-Proxy Grounding

This kind of attention to understanding people and how they think about robots is crucial in order to develop technologies that will remain useful to people for long periods of time. In the case of exploration robotics, scientists, like museum employees, are long-term, naïve robot users. While the focus of this study was primarily on the knowledge and beliefs of these users without considering the information possessed by the robot, the results lead to some of the key concepts which were developed as part of Robot-Proxy Grounding:

- As demonstrated by the docent data, users may put much more weight on the actual failures and successes they experience when using the robot, regardless of what its stated capabilities are. This suggests that, when building common ground between the user and robot, it is not sufficient to provide feedback to users based solely on whether an action is theoretically possible for the robot to execute. Users also must build an awareness of how actions relate to each other as this will more accurately reflect what will happen during plan execution.

- These results also suggested that people may shift their thinking about a robot from “machine” to “collaborator”. This led to the hypothesis that interacting with a robot proxy might also lead to stronger feelings of collaboration; this hypothesis was confirmed this both in a proof-of-concept study (Chapter 6) and in the final system evaluation (Chapters 9 and 10).

3.3 Life in the Atacama

This thesis is largely centered around the exploration robotics domain and results from two years of observations of one particular human-robot system. An analysis of this data led to the development of Robot-Proxy Grounding and the three components of the robot proxy: a representation of science goals, a robot model, and a goal validation system.

3.3.1 Project Description

During the fall of 2004 and 2005, Pamela Hinds and I conducted an observational study of the Life in the Atacama (LITA) project. LITA is a multi-site, multi-disciplinary collaboration primarily funded by NASA. The goals of the LITA project are twofold: to use the Atacama desert of Chile as

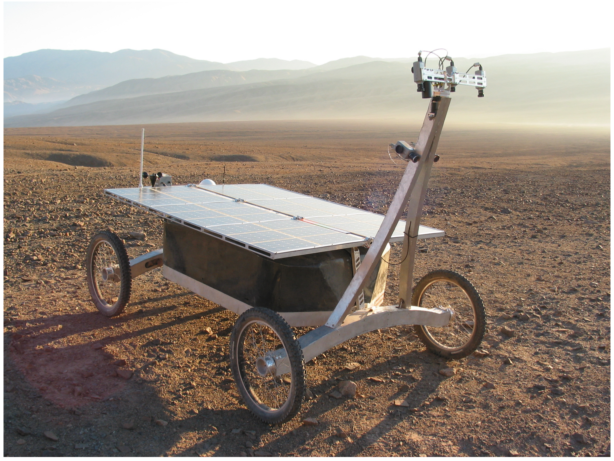


Figure 3.3: Zoë, the "robotic astrobiologist" used in the Life in the Atacama project.

a testing ground to develop technologies and methodologies that may someday be used in the robotic exploration of Mars and to generate new scientific knowledge about the Atacama desert itself. The focus of technology development has centered around a series of semi-autonomous mobile rovers and science instrument payloads. Zoë is the most recent rover and was in use during this study (Figure 3.3). Zoë is a four-wheeled, solar-powered rover equipped with a number of scientific instruments, including cameras for navigation and for acquiring panoramic images; an on-board near-infrared spectrometer; and an underbelly fluorescence imager used for organofluorescence testing which can detect the presence of biological molecules such as DNA.

This study focused on a particular part of the LITA field season known as remote science operations. Remote science operations involved the use of the robot by two different groups of people: the science team, located in Pittsburgh, and the engineering team, located in the Chilean desert with the robot. The science team was composed of biologists, geologists, and instrument specialists from around the United States and Europe (Figure 3.4). Their role was to use the robot to search for signs of life in the desert. The engineering team was composed primarily of roboticists and instrument specialists from Carnegie Mellon University; it also included other instrument specialists and technicians from universities in the United States and Chile. The role of the engineering team was to ensure that the robot was operating safely, to troubleshoot when problems arose, to collect data using instru-

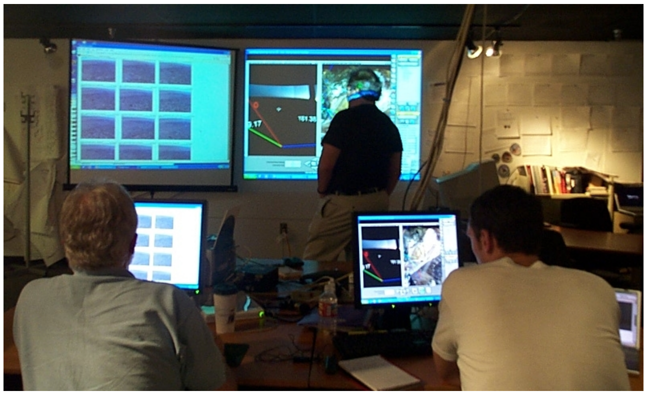
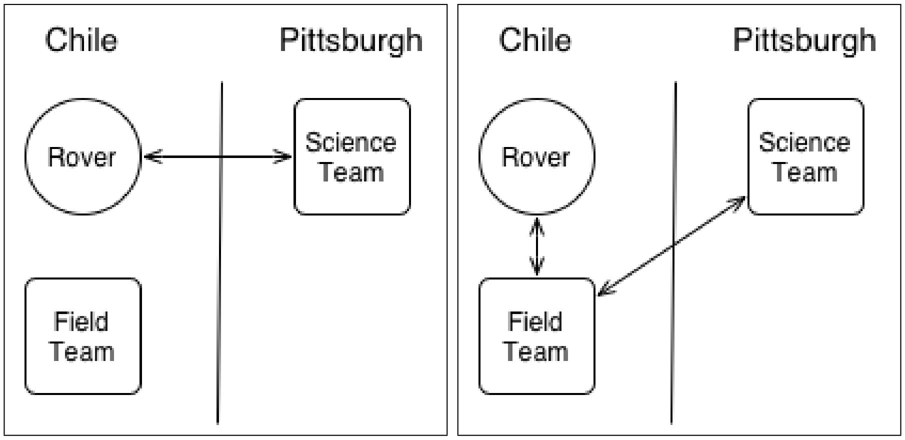


Figure 3.4: Science team members view data returned from the robot.

Figure 3.5: Members of the engineering team work to repair the robot.

ments that were not yet on-board the robot, and to ensure that the science team was able to gather data successfully (Figure 3.5). During this study, Zoë was a semi-autonomous system under constant development, which required the engineering team to act as an intermediary between the science team and the robot. Thus, the science team sent plans for the robot to the engineering team. The engineering team then interpreted the plans, commanded the robot directly to collect the necessary data, manually packaged the data, and sent the data back to the science team.



(a) The science team tries to build common ground with the robot.

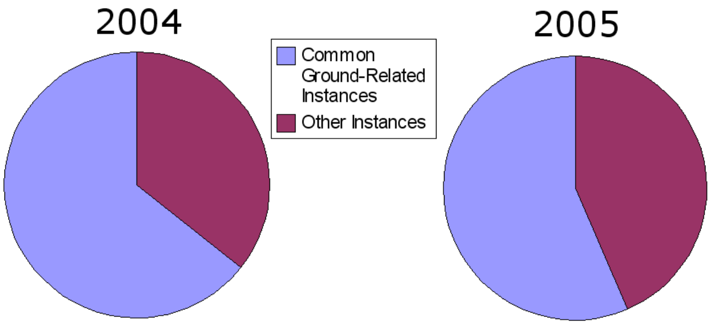
(b) The science team's interactions are mediated by the engineering team.

Figure 3.6: This study examined the grounding process (a) between the science team and the robot and (b) between the science team and the engineering team.

The goal of conducting these observations is to better understand the interaction between the science team and the robot (Figure 3.6(a)). The interactions between the science team and the engineering team (Figure 3.6(b)) are studied to further inform the interactions between the science team and the robot.

In order to collect data about both sites, one researcher observed the science team in Pittsburgh while one to two other researchers observed the engineering team and robot in Chile. The observation process involved writing detailed field notes, drawing diagrams, and taking photographs and video clips. Communication between observers across sites was limited in order to allow each observer to focus completely on the local situation and to better understand the perspective of the group that she was observing at the time. The people observed were told that the aim of the research was to gain a better understanding of how scientists and engineers work with remote rovers and that we would be observing them throughout field operations. During the 2004 field season, 138 hours of observations were conducted in Pittsburgh and 241 hours were conducted in Chile. In 2005, 254 hours of observation were conducted in Pittsburgh and 239 hours of observation were conducted in Chile.

Field notes, combined with artifact documents, which included PowerPoint presentations, emails, and robot plans generated by the science team, formed the data set. An initial reading of the data revealed many com-



	2004	2005
Common Ground-Related Instances	74	120
Other Instances	41	93
Total	115	213

Figure 3.7: Number of problem instances related to a lack of common ground in 2004 and 2005.

munication and coordination problems between sites. Next, the specific errors and miscommunications which occurred were identified and classified (e.g., "Error in plan sent to robot," "Miscommunication regarding interpretation of plan"). These errors and miscommunications are referred to collectively as "problems." Those problems related to common ground were identified based on whether the science team and robot lacked mutual knowledge, and, if so, what kind (e.g., "Missing contextual information," "Lack of transparency into robot's behavior"). A coding of the 2004 data revealed 57 separate common ground problems that occurred during the two weeks of remote science operations [Stubbs et al., 2006c,b]; 91 common ground problems occurred during the 23 days of remote science operations in 2005.

As shown in Figure 3.7, more than half of the problem instances in 2004 and 2005 related to a lack of common ground. We then used the data to trace the causes of these problems, particularly those problems related to the robot's autonomous capabilities [Stubbs et al., 2007]. Improving the grounding process is the main motivation for this thesis.

3.3.2 Autonomy and Common Ground in Human-Robot Interaction

In order to better understand the grounding process between the science team and the remotely located robot, this work focuses specifically on the impact of the robot's level of autonomy on the grounding process. The analysis of robot autonomy presented here is based on the work of Sheridan and colleagues, who define automation as “a device or system that accomplishes (partially or fully) a function that was previously, or conceivably could be, carried out (partially or fully) by a human operator” [Parasuraman et al., 2000, p. 287]. They distinguish between types and levels of autonomy, describing four basic types: information acquisition, information analysis, decision selection, and action implementation. Within robotics, these types of autonomy are commonly collapsed into three and referred to as sensing, planning, and acting (e.g. [Veloso, 2002]):

- Autonomous sensing (information acquisition & filtering)—making observations and refining information
- Autonomous planning (information interpretation & decision selection)—reacting to information or deciding actions and schedule
- Autonomous acting (action implementation)—executing a planned task or producing reflexive reactions

Information analysis is decomposed into data transformation during sensing and interpretation during planning. One robotic system can have a different level of autonomy of each type (sensing, planning, and acting). In this work, the level of each type of autonomy is evaluated as either low, moderate, or high; the metric will be the extent of external guidance required for the system to function. Low autonomy will mean that some basic automation may be present, but both information and procedures must be provided externally. For moderate autonomy, some required information will be provided by an external source, such as intermediate steps or proper system settings, but all procedures function independently. High levels of autonomy will be characterized by systems that can both derive needed information and proceed independently over extended periods. The most significant contribution of the work reported here is a better understanding of how different levels and types of autonomy affect grounding between people and robots, particularly teams of people and a remote robot.

Figure 3.8 depicts the type (sensing, planning, acting) and level of Zoë’s autonomous capabilities throughout this study. During “regular operations” in 2004 and 2005, the robot executed plans sent by the science team with a low degree of autonomous sensing or planning. In 2005, a science autonomy system was introduced which allowed the robot to collect data on its own without specific commands from the science team about where to do so. This resulted in much higher levels of autonomy with respect to sensing,

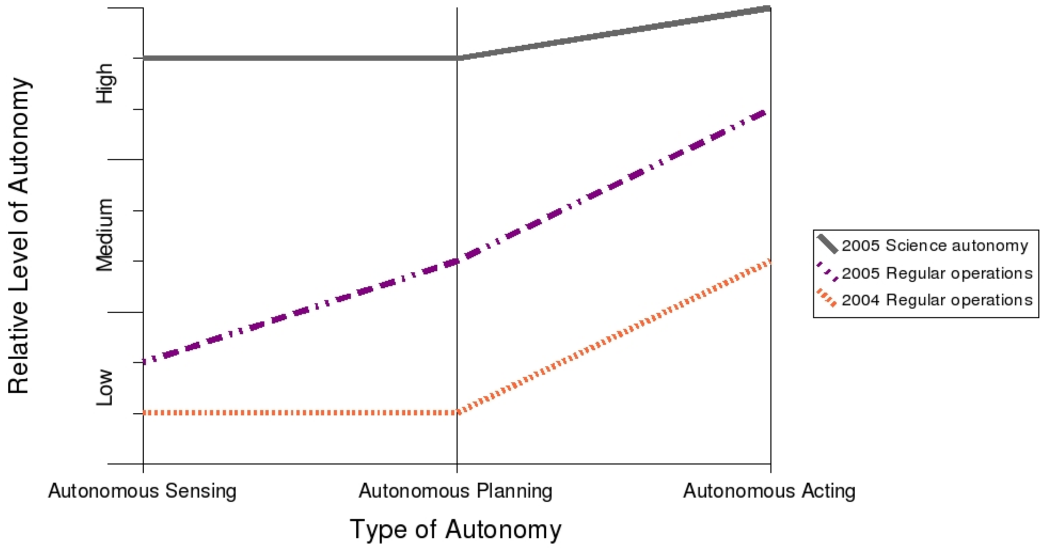


Figure 3.8: Types and levels of autonomy in 2004, 2005, and with the science autonomy system.

planning, and acting. The impact of Zoë's different levels of autonomy on the grounding process is described in the following sections.

Regular Operations—Low to Moderate Autonomy

2004 Results. Zoë's autonomous capabilities were very limited in 2004. Zoë autonomously recorded data about its internal state, could detect some failure conditions, and could detect obstacles but had difficulty accurately estimating its position over the long term. (GPS was not used due to its lack of relevance to planetary missions.) The robot did not interpret any science data and possessed only a basic level of planning for scheduling science actions. Autonomy with respect to planning and acting was low, as engineers were often required to drive the robot manually and to command instrument operations. Figure 3.8 illustrates Zoë's levels of autonomy for sensing, planning, and acting in 2004.

The types of problems which were identified in the 2004 data were predominantly problems of being able to understand references to objects of interest brought on by a lack of co-presence between the science team and the robot. Clark and Brennan argue that when people are not co-present, grounding becomes more difficult; this is supported by Cramton's work on geographically distributed teams [Clark and Brennan, 1991; Cramton, 2001].

According to Clark and Brennan, missing contextual information jeopardizes shared understanding because "the addressee has to imagine appropriate contexts for both the sender and the message" [Clark and Brennan, 1991, p. 143]. The observations revealed numerous problems with contextual information that bear on challenges users face when interacting with a remote robot. Receiving erroneous data from a robot is always a possibility. Without sufficient information about data and the context from which it is collected, making sound scientific judgments can be challenging. In one instance, the team received a fluorescence image in which nearly half of the field of view appeared to be fluorescing, signaling the possible presence of life. This caused a great deal of excitement and confusion, as it was unclear whether the team had indeed found life, whether the camera had malfunctioned, or whether some other unforeseen event had occurred. After nearly a day spent investigating the image, the team concluded that sunlight was responsible for the strange glow they had observed. In this case, a lack of information about the data and its context resulted in confusion and much time spent trying to deduce what could have gone wrong.

Effective reference in communication requires "perspective-taking"—that is, the speaker must take into account the listener's perspective when formulating a referring expression [Krauss and Fussell, 1996]. When two people are physically separated, it is difficult to have insight into the other's perspective. In particular, feedback is less immediate, more difficult to interpret, and may not happen at all. Feedback, in particular feedback about how well the speaker's messages are being understood, is crucial to conversational grounding [Traxler and Gernsbacher, 1992]. In 2004, the science team lacked enough information from the robot to effectively take the robot's perspective, and the robot had no means to detect or improve that situation.

Discussion. During the 2004 season, the science team primarily relied upon the data collected by the robot as well as information from the engineers located with the science team in order to build common ground with the robot. At a basic level, the science team could determine what data had and had not been collected; however, because feedback about errors or instrument failures that occurred was not easily accessible to the science team, the science team turned to the engineers collocated with them, who had the ability to contact engineers in the field, in order to obtain additional contextual information about what was happening in Chile. Had these resources not been available, the grounding process would have been further impaired.

The most significant constraint on the grounding process at these low levels of autonomy was being able to understand the robot's perspective—the context in which it was operating. Lack of copresence contributed significantly to this gap. Had the science team been able to observe the robot's executing commands in the desert, they would have had enough contextual information to disambiguate problems. However, the inability to observe the robot in context combined with a lack of feedback from the robot about its actions inhibited grounding and led to frustration and errors. This observation is similar to studies of situation awareness, although we add to this work by considering the "conversation" between the science team and the robot, where the breakdowns occurred, and how the science team attempted to create common ground with the robot. In particular, feedback from the robot was missing as was an awareness of and adjustment on the part of the robot to the science team's confusion. In common ground parlance, the "acceptance phase" of the conversation was missing. The robot engaged in the "presentation phase" by providing information, but it did not seek evidence of the science team's understanding; therefore, the conversation was incomplete and led to misunderstandings [Clark and Brennan, 1991].

2005 Results. In 2005, Zoë's autonomous navigation capabilities improved substantially during regular operations. Zoë could sense nearby obstacles, develop basic plans to avoid them, and act on those plans with minimal human intervention. This allowed Zoë to drive autonomously between locations specified by the science team. In addition, a new science autonomy system was added to the robot, but because this system was used separately from regular operations, it is discussed and analyzed later. In addition, as a result of problems in establishing common ground during the 2004 field season, the LITA engineers established the practice of sending a daily "robot report" to the science team as a proxy for the information that the robot should have provided autonomously. The robot report included information about exactly which actions were executed, which actions had and had not succeeded, instrument failures, and other contextual information.

One of the strategies that the LITA science team utilized both in 2004 and in 2005 to improve their understanding of the robot context was taking a "context image," a photograph taken by the robot's SPI camera of an area that had already been examined using the fluorescence imager (FI). This provided the scientists with additional information about the larger area within which the FI had been taken. However, these context images were not always taken correctly by the robot, and the science team had to work to detect these errors and determine what had happened. This problem occurred on days 3, 4, and 10 (see Section C.1 for details on this scenario).

Throughout this scenario, the scientists relied on the data from the robot and the robot report to establish common ground regarding how the robot was operating, and they used this information to adjust the commands they sent to the robot. This process mirrors conversational grounding between people in that the science team attended to the feedback provided by the robot and adjusted their communications in hopes of being more effective. Unlike most communication between people, however, the adjustment was one-sided. The robot did not learn how to better communicate with the science team and, as a result, the science team was not always successful at deducing the robot's actions.

In a second scenario, the science team expressed a desire to understand exactly how far the robot traveled and exactly where data products were collected. This was complicated by the fact that the distances the scientists could measure in the plan creation tool, the distances shown in the human-readable plan, the "odometric distance" that Zoë reported to have traveled, Zoë's estimate of how far Zoë traveled ("telemetry"), and the actual distances Zoë moved were all different and were all computed in different ways.

The data suggest that some members of the science team had an understanding of what Zoë's odometry and telemetry data were. However, this was less helpful to them when planning paths for the robot because the distances they measured in the planning tool were not necessarily the same as the distances that appeared in the plan that was generated or the distances that Zoë reported or actually traveled. The science team used the robot report as a definitive source of information about how far the robot traveled between locales. This may have been due to the fact that the robot report was the only easily accessible source of this kind of information.

Discussion. In dealing with problems that occurred during regular operations, the science team relied primarily on the data returned from the robot and the robot report (the proxy for the information that the robot might have autonomously returned itself). Without the benefit of copresence, the science team used the robot reports and the data from the robot as their main sources of information about what had happened in the field, but this feedback was still inadequate to establish common ground with the robot. The science team was not able to understand the robot; the robot did not verify the science team's understanding through an acceptance phase, nor did it learn how to better refer to objects, locations, and other environmental factors so that it and the science team could expand their common ground.

In the 2004–2005 regular operations data, the major issues of copresence and inadequate feedback appear to be most associated with moderate to high levels of autonomous acting. The robot was acting autonomously (albeit sometimes at low levels) by driving and deploying instruments with little or no human interaction. Without contextual information or adequate feedback, it was difficult for the science team to understand these autonomous actions. The robot had no means to maintain its end of the conversation by detecting that the science team was having difficulty understanding the information that it presented to them.

In summary, in 2004 and 2005 regular operations, the grounding process between the science team and the robot as it operated at these low to moderate levels of autonomy was primarily characterized by:

- A need by the science team to understand exactly what the robot had and had not done
- A reliance by the science team on data returned from the robot and about the robot
- A grounding process primarily constrained by a lack of copresence
- Inefficiency and errors on the part of the science team as a result of not being able to establish common ground with the robot, understand what it was doing, and command it more effectively

Science Autonomy—High Autonomy

A science autonomy system was available on Zoë during portions of the 2005 field season. It consisted primarily of software to collect and interpret sensor, camera, and instrument data and software to plan a response, if any, to these observations. The software was designed to allow the robot to collect science data as it traveled between locations of scientific interest. The science autonomy system allowed the science team to request autonomous collection of normal camera images and chlorophyll-only fluorescence images (FIs); if the robot detected that a chlorophyll-only fluorescence image showed evidence of life, the robot could *follow up* by taking a full fluorescence image set. The science autonomy system gave Zoë a much greater level of autonomy of all three types than the robot possessed during regular operations in 2004 and 2005 (Figure 3.8). Sensing, planning, and instrument deployment were all accomplished with little to no human intervention. When using the science autonomy system, the science team was forced to adopt a different strategy for grounding. In particular, issues arose around why the robot made certain decisions in addition to recurring questions about objects of reference as described in regular operations.

On days 1, 2, 3, 4, and 15, the science team discussed the fact that the robot was not performing follow-ups when it should have. The science team attempted to find out why the robot was not initiating follow-ups (see Section C.2 for details on this scenario). In contrast to the examples from regular operations in 2004 and 2005, this is a case in which the science team understands what has and has not been done but is baffled about *why* the robot made particular decisions. They attempt to reason amongst themselves and with an engineer about what Zoë might be "thinking," but they do not have an adequate understanding of the robot's decision-making algorithms or enough feedback from the robot to communicate well enough to get the data they want. The robot, for its part, has no means to represent or reason about why the science team has chosen particular actions. The robot is thus unable to ensure that the rationale for its actions is understood and its decisions are consistent with the science team's larger goals.

In the field, the engineering team was aware that at times there were problems with the follow-up mechanism due to water being present on rocks or due to sunlight shining under the robot. Because the science team lacked this information about the context within which the follow-up FIs were taken, they had to try and deduce why the robot decided not to conduct follow-up FIs based on the data available to them. Their grounding strategies included examining the chlorophyll FIs to see how strong the signal was and calculating the time of day at which images were taken to see if sunlight might have been an issue. In this example, there is evidence of breakdowns in both shared perspectives between the science team and the robot about what was located where and why things were or were not done.

Discussion. With the high levels of sensing, planning, and action autonomy that the robot possessed when using the science autonomy system, the science team's problems were less focused on exactly *what* the robot was doing; rather, they were primarily concerned with *why* the robot was making particular decisions.

Copresence continued to be a constraint and was particularly pronounced as the science team attempted to understand the robot's high levels of autonomous sensing and action. In addition, we observed that transparency became a constraint with high levels of planning autonomy. Even if the science team had been watching the robot as the science autonomy system was working, they would not necessarily have had enough information to determine why the robot stopped in particular locations or why it did or did not perform follow-up FIs. The science team not only had to understand how the robot would react to positive or negative evidence of life, but they also had to try to understand the robot's analysis process.

Based on the strategies used by the science team to understand the science autonomy system, it appears that transparency into the robot's decision making process became the primary constraint on the grounding process. The robot report provided only factual information and nothing about why the robot performed measurements or follow-ups as part of the science autonomy system. Instead, the science team used the data to determine what may have happened and then relied on engineers to explain the algorithms behind how the robot made decisions.

Transparent interactions have been defined as those in which a user can "see through" the logic behind a machine's operation. Some researchers have focused on users' understanding [Sinha and Swearingen, 2002], others on the explanations provided by the robot [Herlocker et al., 2000], and others on making the system transparent enough that no mental model of it is required [Goodrich and D. R. Olsen, 2003]. Consistent with the common ground framework, this work approaches transparency as a dynamic feature of the interaction between the science team and the robot. Transparency therefore refers to the process of developing common ground between the science team and the robot about the robot's logic. Bardram and Bertelsen [Bardram and Bertelsen, 1995] similarly suggest that transparency can not be understood as a static feature but must reflect a deliberate formulation and refinement of understanding during the course of human-computer interaction. Although people certainly ask questions and converse about reasons for their thoughts and actions, this idea of understanding someone's logic is not well articulated in current work on common ground. From these observations of the LITA project, it can be argued that the dynamic creation of transparency–promoting transparency throughout the course of interacting with the robot—becomes a more crucial element for creating common ground as robots acquire higher levels of autonomy, particularly autonomous planning.

This shift from a focus on missing contextual information to a lack of transparency can be seen in Figure 3.9. This figure shows, of the 148 problems related to common ground which we identified from the 2004 and 2005 data, the problems for which missing contextual information or a lack of transparency was the most significant cause. As the graph indicates, the nature of the problems shifted almost entirely away from problems with missing context to issues of transparency about the robots' decisions and logic. It is also important to note that each common ground problem may have occurred on multiple days; and, in fact, problems related to a lack of transparency generally took more days to resolve than those related to missing contextual information.

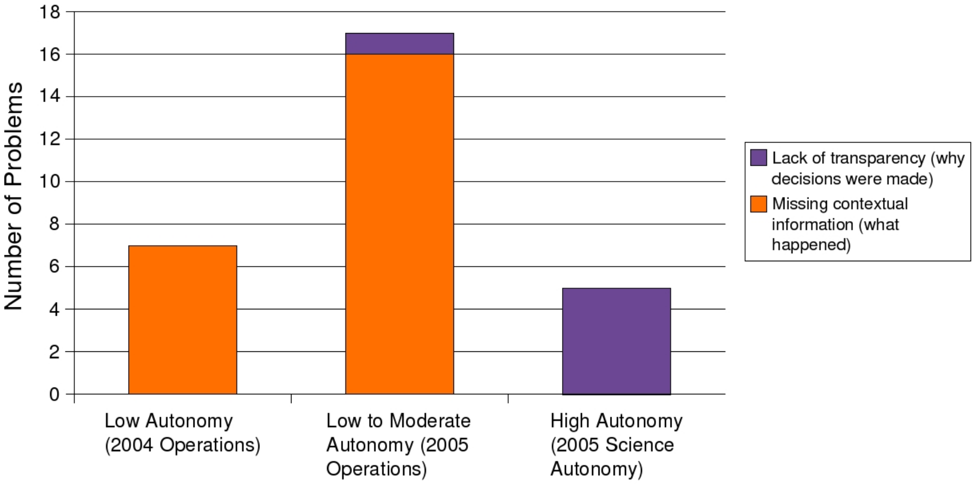


Figure 3.9: Common ground issues for problems relating to autonomy.

Conclusion

For this study, the researchers spent a total of over 800 hours observing the Life in the Atacama (LITA) exploration robotics mission and documenting the grounding process of the LITA science team and the remotely-located robot, Zoë. These observations suggest that the factors that disrupted grounding changed as the robot exhibited more autonomy. As autonomy increased, rather than being confused about the context in which the robot was collecting data, members of the science team became confused about why the robot was doing what it was doing. The observations also indicated that the grounding process became more complicated when the entire team attempted to work together with the science autonomy system. The data suggest that a team's shared mental model of an autonomous robot is more complex, more variable, and needs to be more consistent across teammates than mental models of simple devices.

Higher autonomy did not necessarily lead to better or more error-free interaction. Common ground problems emerged whether autonomy was low or high. The data suggest that designers need to be aware of how autonomy changes the type of information needed from the robot and the type of "conversation" that is necessary between the robot and the people with whom it is interacting. For grounding to occur with robots that have low levels of autonomy, contextual information and feedback are particularly critical; at high levels of autonomy, particularly for autonomous planning, users need transparency with respect to the robot's decision-making processes.

3.3.3 Impact on Formation of Robot-Proxy Grounding

The errors and miscommunications which were documented in the course of this study served as motivation to work to improve human-robot interaction in this problem domain. The concept of the "robot proxy" developed from an analysis of the challenges inherent in conducting remote exploration robotics. Within the domain of remote exploration robotics, the cost of mistakes in data collection is extremely high. Data that is not useful to the science team wastes valuable time and resources. However, delay costs are extremely low: the science team does not pay a penalty in terms of data return if the team spends time revising the plan (given that the plan is sent to the robot after the robot finishes daily operations). Therefore, this thesis introduces a software system that can participate in the grounding process during plan creation before the plan is sent to the robot. This system acts as a proxy for the robot, providing crucial feedback to the science team and supporting transparency without consuming time or resources during plan execution. The beauty of such a robot proxy-based system is that it improves upon conversational grounding between people, which requires the availability of both parties. It is possible to exactly recreate the behavior of the robot through this system without the actual robot's being part of the conversation. This enables people interacting with a remote robot to understand exactly how it would respond to their requests and provides the immediate feedback so critical to the grounding process.

Based on the observations of how the science team utilized Zoë in 2004 and 2005, this thesis presents a formalized goal representation which is used to encode the scientists' higher-level goals. Through Robot-Proxy Grounding, the robot works to determine whether the goals are physically possible and internally consistent, promoting common ground between the scientist and robot.

When developing evaluations for the robot proxy concept and the actual implementation, the Life in the Atacama project served as the basis for the tasks participants would complete. The tasks in both studies involve driving a robot around a desert, deploying instruments to collect data about objects of interest. While these studies were not nearly as complex as the scientists' work in the LITA project, the narrative was sufficiently compelling that nearly every participant was motivated to complete the task successfully.

3.4 Summary

These studies of two different human-robot systems, the Personal Exploration Rover museum exhibit and the Life in the Atacama project, serve as the foundation of this thesis. The Personal Exploration Rover studies focused particularly on the mental models of individuals interacting with the robot [Stubbs et al., 2005, 2006a], an important aspect of common ground. The Life in the Atacama studies demonstrated the errors and miscommunications that can occur when collaborators lack common ground [Stubbs et al., 2006c,b]. In addition, analysis of the impact of robot autonomy on the development of common ground forms the basis for the introduction of the robot proxy as a means to promote common ground between a scientist and a remotely-located robot [Stubbs et al., 2007].

Chapter 4

Characterization of Common Ground for Human-Robot Interaction

This chapter introduces a novel characterization of common ground for human-robot interaction which focuses on the mutual knowledge held by a human and robot about the robot's current state and its current context. These aspects of common ground are not well-represented in previous work in common ground theory. This chapter also includes an analysis of how this characterization generalizes across two exploration robotics systems.

In examining common ground for human-human interaction, Clark posits that the common ground between two participants in a joint activity can be divided into three parts [Clark, 1996, p. 43]:

1. Initial common ground, which represents the background knowledge that participants assumed they shared when they began the joint activity.
2. Current state of the joint activity, which represents what participants believe the state of the activity is at the current time.
3. Public events so far, which represents the history of events that participants believe have occurred in public before the current state.

Much of the initial common ground between two people can be derived from basic knowledge about human beings or from cultural norms. As Clark argues, "All of us take as common ground, I assume, that people normally have the same senses, sense organs, and types of sensations" [Clark, 1996,

p. 86]. When interacting with a robot, however, a person cannot take for granted how the robot functions. Robots may possess a variety of sensors which do not have a human analogue, such as infrared detectors. Understanding the sensing and actuation mechanisms in a robot can be very challenging, which can result in difficulties in predicting and interpreting robotic behavior over time. This suggests that a more detailed characterization of common ground might be helpful for understanding human-robot interactions.

4.1 Common Ground for Human-Robot Interaction

Given the initial common ground and the history of public events, additional detail regarding the current state of the joint activity is provided by the following two components:

1. The **current state of the robot**, which represents the properties of the robot itself. The current state of the robot includes what sensors/actuators it has, what sensors/actuators are currently functioning, the robot's current goal/task, the pose of the robot, and the pose of the robot's sensors/actuators. The current state of the robot also includes the current capabilities of the robot, which represent the commands that the robot can execute given what sensors and actuators are currently functioning. Some aspects of the robot's current state may be visible to an external observer (whether or not its wheels are moving), while others may be invisible (the temperature of the robot's processing unit).

2. The **current context in which the robot is operating**, which represents the properties of the environment in which the robot is located. This information includes what the robot's position is with respect to the environment, where obstacles are located, what type of terrain the robot is situated on, etc.

This characterization includes as part of the robot state those commands which are executable at the current time; that is, the set of commands which the robot can execute given which sensors/actuators are functioning. If a particular sensor or actuator breaks entirely or only has limited functionality, the set of commands which the robot can execute may change over time. This is particularly relevant in the case of remote exploration robotics, in which it may not be possible to repair the robot.

This two-part characterization is a novel contribution of this thesis; it is compelling because it is important for HRI but not well-captured in Clark's work. Clark's representation of common ground between two people assumes that this level of specificity is not necessary because so much of the underlying information is well-known to people: humans can infer a great deal about a collaborator based on the fact that they are both human. People can interpret one another's gestures and facial expressions, for example, and we have a great deal of knowledge about normal human capabilities and behavior. When a person is no longer collaborating with another human, however, much of this knowledge may be unhelpful. In order to capture what knowledge is necessary for a human and robot to interact successfully, it is important to consider common ground with respect to the robot's state and context.

While Clark's work does not specifically address these issues of state and context, Cramton has captured the importance of mutual understanding of context; however, her work remains limited in its applicability to HRI. In particular, Cramton has identified five failures of mutual knowledge which can occur between teams of people working in different physical locations [Cramton, 2001]. Cramton cites missing contextual information as a significant problem which can occur between teams: team members are not aware of the different contexts in which they are working, such as different national holidays or different business hours. This is very relevant to HRI because robots operate in the physical world and thus an awareness of their environment is critical to ensure a successful interaction. This is particularly true in exploration robotics as the robot and human do not necessarily share the same physical world.

Cramton's other mutual knowledge problems relate to the distribution, salience, speed of access to, and interpretation of information relating to the task at hand. When describing these problems, Cramton does not specifically differentiate between information about the team's task and information relating to team members' current states or to their expected behaviors. Much of this information may be assumed to be common ground when all participants are human, but a person interacting with a robot cannot make these same basic assumptions. Thus, for analysis of HRI systems, Cramton's work does not adequately capture the importance of mutual knowledge of the robot's state.

According to the novel characterization presented in this chapter, when a human and robot interact, the robot's state and context are part of their common ground. This implies that a successful interaction has three components:

- The robot must have **self-awareness**: The robot must have information about its current state and context. In order to function effectively in the world, a robot must possess information in some form about its own capabilities and task. Because the robot acts in the physical world, it must be able to sense whatever aspects of its environment are relevant to its task. Noisy or inaccurate sensor data may result in an inaccurate representation of the world, which may have deleterious consequences for robot performance. Robotics researchers continue to address these challenges by creating robots which are increasingly autonomous and more aware of themselves and their environments.

- The robot must support **transparency**: The user must have information about the robot's current state and context. Researchers from a variety of domains including robotics, human-computer interaction, and ergonomics are working towards the development of robot interfaces which support this awareness. In order to ensure that the user sends commands to the robot which are feasible and meet the user's goals, the user must be aware of the robot's capabilities, what sensors/actuators are currently functioning, etc. The user must also be aware of the robot's environment, as the environment may affect the result of an executed command.

- The person and robot must have **mutual knowledge**: Both the user and robot must know that they have information about the robot's current state and context. This is a novel concept for many roboticists, particularly in domains such as remote exploration robotics, in which human-robot interaction bears little resemblance to human-human, conversational interaction. Setting aside the issue of whether or not a robot can "know" something in the same sense that a person does, it is important for the person, at least, to be aware of the fact that both she and the robot have knowledge of the robot's state and context. As Clark and colleagues have demonstrated, this mutual knowledge is the foundation of successful collaborations [Clark and Wilkes-Gibbs, 1986; Clark and Marshall, 1981]. In order for two individuals to work together, they must coordinate what they do and when they do it; this can only happen if both parties appeal to their current common ground [Clark and Brennan, 1991].

As a simple example, consider the case of a robot which has been deployed in an abandoned mine. The user, located outside of the mine, wishes to direct the robot to a particular corridor inside the mine in order to inspect it for damage. As the robot begins to enter the mine, if it does not have **awareness** of its own state, it cannot tell whether its wheels are moving and it is making progress. If it lacks awareness of its context, the robot cannot tell where the walls of the mine are, and it risks colliding with them. Similarly, the robot must provide the user with **transparency** into the robot's state and context. If the user lacks information about the robot's state, she will not know whether the robot is functioning properly or needs to be retrieved from the mine. If the user lacks information about the robot's context, she will not know where the robot is located or if it has reached the goal location. **Mutual knowledge** is required in order for the user and robot to coordinate and adapt to unforeseen circumstances. Suppose the user and the robot share the knowledge that the user is trying to create a map of a particular room in the mine. If the robot is driving towards the room and finds that its path is blocked by an obstacle, it can plan a different route to its destination. The mutual knowledge between the user and robot allows them to coordinate a solution to the problem.

The following analysis provides examples which illustrate how the characterization of common ground in terms of a robot's state and context generalizes to different types of exploration tasks. As documented in Chapter 3, there are a wide variety of problems which may occur over the course of a human-robot interaction which stem from a lack of common ground between the person and robot. These problems may vary depending on what type of information the person lacks. The following two sections describe specific interactions observed in the context of the Life in the Atacama and the Personal Exploration Rover systems. These examples illustrate how the interactions between people and robots vary in part due to different levels of common ground with respect to the robot's state and context.

4.2 Analysis of the Life in the Atacama Project

As described in Section 3.3, the goals of the LITA project are twofold: to develop Mars-relevant robotic exploration technologies to generate new scientific knowledge about the Atacama desert itself. Analysis of the Life in the Atacama project identified seven types of errors and miscommunications which stemmed from the science team's missing information; four of these seven types of problems can be related to a lack of common ground regarding the robot's state or context. At times, the science team explicitly stated a desire for particular information about the robot's state and its context. Other miscommunications occurred because the science team had misunderstandings about the robot's capabilities, which included inaccurate information about the robot's state. The science team also incorrectly estimated the position of the rover, indicating a lack of correct contextual information. On some days, the science team sent plans to the rover which contained errors, suggesting they had inaccurate knowledge of the robot's current capabilities. This illustrates the variety of problems which occurred that related to a lack of common ground regarding the robot's state and context.

Two of the problems identified in the Atacama analysis are presented below as more detailed examples of how the science team was impacted by a lack of knowledge about the robot's state and context.

Problems interpreting context images

One of the strategies that the LITA science team utilized both in 2004 and in 2005 to improve their understanding of the robot context was taking a "context image," a photograph taken by the robot's SPI camera of an area that had already been examined using the fluorescence imager (FI). This provided the scientists with additional information about the larger area within which the FI had been taken. However, these context images were not always taken correctly by the robot, and the science team had to work to detect these errors and determine what had happened. This problem occurred on days 3, 4, and 10 (see Section C.1 for details on this scenario).

In this scenario, the scientists lacked information about the robot's **context**: they did not have a good understanding of where the robot's instruments are pointing with respect to the ground. The team used information from the history of the interaction in order to diagnose the problem: namely, the data that had been returned from the robot. Most significantly, the scientists lacked correct information about the robot's **state**: the scientists did not seem to realize that the robot would execute a marker plow after the FI, so they did not drive the robot up far enough to take a context image of the FI. As a result, the science team had difficulties collecting their desired data.

Missing fluorescence image follow-ups

A science autonomy system was available on the robot during portions of the 2005 field season. It consisted primarily of software to collect and interpret sensor, camera, and instrument data and software to plan a response, if any, to these observations. The software was designed to allow the robot to collect science data as it traveled between locations of scientific interest. The science autonomy system allowed the science team to request autonomous collection of normal camera images and chlorophyll-only fluorescence images (FIs); if the robot detected that a chlorophyll-only fluorescence image showed evidence of life, the robot could follow up by taking a full fluorescence image set.

On days 1, 2, 3, 4, and 15, the science team discussed the fact that the robot was not performing follow-ups when it should have. The science team attempted to find out why the robot was not initiating follow-ups (see Section C.2 for details on this scenario). They attempted to reason amongst themselves and with an engineer about what the robot might be "thinking," but they did not have an adequate understanding of the robot's **state** to obtain the data they wanted. In particular, the scientists did not have enough information about what the state of the robot was when it decided not to take follow-ups, so they had difficulty determining why the problem occurred. The scientists also did not know that the numbers in their commands were being rounded, which resulted in missing follow-ups.

In the field, the engineering team was aware that at times there were problems with the follow-up mechanism due to water being present on rocks or due to sunlight shining under the robot. Because the science team lacked this information about the **context** within which the follow-up FIs were taken, they had to try and deduce why the robot decided not to conduct follow-up FIs based on the data available to them (part of the history of their interactions with the rover). Without a good understanding of the robot's state and context, it was difficult for the science team to predict when the robot would collect data and construct plans accordingly.

These scenarios illustrate the challenges faced by the science team because they lacked common ground with respect to the robot's state and context. Without this information, it was difficult for the scientists to determine what the robot had done, interpret the data they received, and accurately predict the robot's behavior for future plans.

4.3 Analysis of the Personal Exploration Rover Project

The Personal Exploration Rover museum exhibit represents a very different type of exploration robotics interaction from the LITA project. As described in Section 3.2, the Personal Exploration Rover (PER) was designed as a tool to educate the public about certain aspects of NASA's Mars Exploration Rover (MER) mission. Eighteen museum employees who worked with the

PER and its exhibit were interviewed from December 2003 through June 2004. The following interview excerpts illustrate how different people possessed different levels of common ground with respect to the robot's state and context. This analysis provides support for this characterization in a task domain which involves collocated human-robot interaction not driven by scientific data collection.

Misunderstanding the PER's capabilities

When one museum employee was asked what is the hardest thing for the PERs to do, he responded, "I think the hardest thing I've seen [is for them] to be able to accurately judge where their wheels are sometimes. They end up dragging along the side of a rock sometimes..." This comment suggests that the employee believes that the robot is attempting to determine the position of its wheels with respect to obstacles in the environment. This is an incorrect belief about the robot's **state**, as the robot does not have any way to sense this information, nor does the robot attempt to "accurately judge" wheel position while executing missions.

Lack of knowledge of the PER's environment

Two of the employees at the National Air and Space Museum expressed frustration that the PER would work correctly upstairs in their offices but not downstairs on the exhibit floor. These employees were not able to troubleshoot what was going wrong—they lacked information about some part of the robot's environment (**context**) as well as an understanding of how that feature of the robot's environment could affect the robot's performance (poor information about the robot's **state**). Because they lacked this knowledge, the employees were not able to develop a remedy for the problem.

Differing perceptions of the PER

When asked about whether or not the PER is intelligent, most employees agreed that the PER was intelligent in some way. However, their explanations of *why* they considered the PER to be intelligent reveal significant differences in the employees' knowledge and perceptions of the PER. Consider the following interview excerpts:

> *Employee A*
> **Interviewer**: Do you think the PERs are intelligent?
> **A**: I think the software's intelligent. I mean yeah, I do.
> **Interviewer**: Do you think it's more or less intelligent than a dog?
> **A**: More.
> **Interviewer**: Why?
> **A**: Because it can do exactly what you tell it to do.
> **Interviewer**: So dogs kind of fail at that cause they don't always do exactly what you tell them.
> **A**: Yeah.

> *Employee B*
> **Interviewer**: Would you say that the PER is intelligent?
> **B**: Yes. Like I was talking about the autonomy of it, it's so amazing that it can actually scan around and look for the rock. I mean, I've seen it sometimes when it was really far off, at least a couple if not three or four feet away from the rock, and it manages to find it and approach it. I think that's pretty neat there....
> **Interviewer**: How about [comparing the PER to] a car?
> **B**: It's smarter than a car. A car will back into something, you know, if you tell it to go backwards, it'll back into that tree, it's not gonna stop before it hits it.

Employee A seems to perceive the robot as an obedient worker which obeys the commands it is given. It is certainly true that the PER attempts to travel according to the instructions provided by the user in order to reach a target rock. However, Employee B appears to have richer information than Employee A: Employee B recognizes that the PER is able to adjust its course in order to approach the target rock. Employee B seems to be able to recognize when the robot is in the **state** of trying to find a target rock and that it will modify its behavior based on where it senses the rock is located. The location of the rock is part of the robot's **context**, and Employee B recognizes the relationship between the location of the PER, the location of the rock, and the robot's behavior.

These excerpts illustrate how employees working with the PER sometimes lacked information about the robot's state and context and how different employees developed different perspectives about the robot; these perspectives can be distinguished in part because of the differing levels of common ground with respect to the robot's state and context.

4.4 Summary

This chapter has introduced a novel characterization of common ground for human-robot interaction based on work with the Life in the Atacama and Personal Exploration Rover projects. In addition to considering the initial common ground between human and robot and the public history of events, it is also important to consider:

1. The current state of the robot
2. The current context in which the robot is operating

Previous work in common ground theory by Clark and Cramton does not sufficiently address the challenges which people face when interacting with robots, particularly in remote exploration. As roboticists strive to make robots more self-aware and create interfaces which allow people to use them, this chapter argues that it is also important to consider the mutual knowledge between the person and robot. An analysis of scenarios from the Life in the Atacama and Personal Exploration Rover projects illustrates the problems which can occur when users lack information about the rover's state and context and how users' perceptions of the robot can vary based on this information. This analysis supports the characterization presented here as being generalizable across different exploration tasks.

Chapter 5

Robot-Proxy Grounding

The interactive process by which common ground is established is referred to as "grounding." More specifically, Clark and Brennan describe grounding as follows [Clark and Brennan, 1991, p. 129]:

> The contributor and his partners mutually believe that the partners have understood what the contributor meant to a criterion sufficient for current purposes. This is called the *grounding criterion.* Technically, then, grounding is the collective process by which the participants try to reach this mutual belief.

The methods which participants may use in order to ground are dependent upon constraints on the grounding process as well as the costs of grounding, as outlined in [Clark and Brennan, 1991]. These characteristics with respect to exploration robotics are described in detail in Section 5.1 below.

According to Clark and Brennan, grounding takes place through a two-phase process, the presentation-acceptance process [Clark and Brennan, 1991] (Section 5.2.1). Based on the constraints and costs of grounding in exploration robotics, this thesis adapts the presentation-acceptance process for use with a *robot proxy*, a software system with which the user can "converse" in real time while he formulates a plan. The robot proxy can check whether the goals the user has specified are internally consistent and also consistent with the robot's capabilities. The process by which the conversation between the user and robot proxy takes place, referred to as *Robot-Proxy Grounding*, is described in Section 5.2.2.

5.1 Characteristics of Grounding in Exploration Robotics

This section focuses on the key characteristics of the grounding process for exploration robotics: the constraints on the grounding process and the costs associated with grounding for both the user and the robot.

5.1.1 Constraints on Grounding

The constraints that are imposed on the grounding process by the interaction domain affect the strategies that people use when grounding. The eight constraints presented by Clark and Brennan in [Clark and Brennan, 1991], as reframed for human-robot interaction, are:

- Copresence: The user and robot share the same physical environment; each can see/hear what the other is doing/looking at.
- Visibility: The user and robot are visible to each other; they may be able to see each other without seeing what the other is doing/looking at.
- Cotemporality: The robot receives communications at roughly the same time the user produces them and vice versa.
- Audiability: The user and robot can hear each other.
- Simultaneity: The user and robot can send and receive communications at once and simultaneously (full duplex communication with zero latency).
- Sequentiality: The user's and robot's turns cannot get out of order.
- Reviewability: The robot can review messages from the user.
- Revisability: The user can revise messages before they are sent to the robot.

Considering the exploration robotics domain in particular, the grounding process is constrained by a lack of copresence, visibility, cotemporality, audiability, and simultaneity.

Without **copresence**, the user cannot tell exactly what the robot is doing or what its environment is like. If the user is not copresent with the robot, the user may lack contextual information about the environment the robot is working in and the user may have difficulty interpreting silence from the robot (see [Cramton, 2001]). From the robot's perspective, not being collocated with the user may make it more difficult for the robot to communicate feedback as well as more difficult for the robot to establish references to targets of interest.

Without **visibility**, the robot cannot detect the user's gestures, facial expressions, or pose. In addition, the user cannot receive visual feedback directly from the robot by observing the position and orientation of sensors and actuators, indicator LEDs, video screens, etc.

Without **cotemporality**, the user must send a set of actions to be executed to the robot, the user cannot make corrections to these actions at run time, and the user receives feedback only after all actions are completed. Because the user does not receive data at the same time that the robot produces it, the user experiences a delay in data return and feedback. Because the robot does not receive plans at the same time that the user generates them, the user must construct a set of actions without any feedback from the robot as the actions are executed, and the user is unable to make corrections at run time. Cotemporality is generally present in teleoperation: the robot responds to commands roughly instantaneously and returns data roughly instantaneously. However, in the case of many remote exploration robotics problems, such as space robotics, the user and robot lack cotemporality.

Without **audiability**, the user and robot cannot interact through speech. The user cannot use speech to command the robot, and the robot cannot process audible backchanneling. In addition, the robot cannot provide spoken feedback to the user.

Without **simultaneity**, the user cannot react to the robot while the robot is executing actions or returning data and vice versa: there is no support for backchanneling. Simultaneity is often present in teleopration; the user is able to receive data from the robot while commanding it, such as observing a live video feed while driving the robot.

The grounding process does possess sequentiality, reviewability, and revisability, which are utilized in the Robot-Proxy Grounding process.

5.1.2 Costs to Participants

The strategy that participants will adopt when grounding is not only affected by the constraints on the grounding process, but it is also affected by the costs associated with the media being used [Clark and Brennan, 1991]. For exploration robotics, we are particularly concerned with the costs to a user and robot in terms of time, effort, energy, and bandwidth; the costs for both the user and robot are summarized in Table 5.1.

The highest costs to the user are the costs of formulation, production, understanding, and faults. Formulation costs represent the time and effort required to formulate or reformulate plans, and more complicated plans have a higher formulation cost. Production costs include the cost of the act of producing a plan, including the costs of using the planning interface. Understanding costs represent the costs of understanding what is meant by data returned from the robot; because the user must understand the data returned from the robot before sending the next plan, the time and effort required to understand this data may have a significant negative impact on the planning process. Formulation, production, and understanding costs to the user are all costs which accrue “offline,” when the robot is not necessarily executing any actions. Fault costs are related to the robot’s behavior; fault costs include the wasted time and bandwidth which occurs when a plan contains mistakes.

Conversely, the user experiences low costs for start-up, delay, speaker change, display, and repair. Start-up cost represents the cost of starting a new plan; generally this is a simple function of the user interface software, and so it is not a significant time investment. Delay cost is the cost of delaying the plan to revise it further before sending it to the robot; since, at least in the Life in the Atacama project, the scientists developed a plan after the robot had stopped executing actions for the day, delaying the plan by another hour or two in order to improve it did not significantly affect remote operations. Speaker change cost, the cost of sending the plan to the robot such that the robot becomes the “speaker,” is also relatively low; it is possible for the user to continue working on other tasks while the plan is being uploaded to the robot. Display cost, the cost of presenting something to the robot, is also fairly low, given that the only means to “present” information to the robot is by constructing a plan. Repair costs are also low and incurred offline: it is relatively easy for the user to make changes to the plan before it has been sent to the robot.

For the robot, the highest costs are production, delay, display, fault, and repair. The production of science data is very costly as it requires a lot of time and energy. The cost of delaying data collection or return is also expensive because delays may result in less overall data being collected. Display cost, the cost of presenting something to the user, is also high as a much larger amount of bandwidth is needed to return data than to send plans. The costs of mistakes in data collection, referred to as fault costs, are also high: one incorrectly executed action may result in numerous subsequent data products which are unusable by the user, wasting time, energy, and bandwidth. Repair costs to the robot are also high as the robot must stop data collection in order to make repairs.

The lowest costs to the robot are those of formulation, understanding, start-up, and speaker change. The formulation cost, or cost to generate an ordered sequence of actions based on information sent from a user, is fairly low as this is not a difficult problem. The cost to interpret the plan from the user, understanding cost, is also fairly low. The cost of starting a new execution cycle (start-up cost) is minimal, especially if the robot performs only one new execution cycle each day. Finally, the speaker-change cost, or cost for the robot to send data to the user so that the user becomes speaker, is relatively low since the robot only performs this action once per day.

In general, fault costs are extremely high. Data which is not useful to the science team wastes valuable time and resources. Due to the constraints and costs of the grounding process in this domain, it is not possible for the human and robot to interact directly in real-time, as is the case for human conversational grounding. Thus, this thesis introduces the novel concept of a robot proxy. The robot proxy is a software system with which the user can interact in real-time as the plan is being developed. The robot proxy can provide the user with feedback regarding whether his goals are internally consistent and also consistent with the actual robot's capabilities. By promoting common ground between the user and a robot proxy, rather than the actual robot, this thesis seeks to improve the quality of plans before they are sent to the robot. The use of the robot proxy is intended to reduce formulation, production, repair, and understanding costs to the user (Table 5.2). The process by which the user and robot proxy build common ground is referred to as Robot-Proxy Grounding. The exact steps which occur during Robot-Proxy Grounding are based upon the human-human grounding process described in the following section.

5.2 Presentation-Acceptance

In order to determine how the "conversation" between the user and robot proxy will proceed, it is important to understand how common ground is built between two people engaged in conversation (Section 5.2.1). This thesis adapts the traditional presentation-acceptance process for use with a robot proxy working in an exploration robotics domain (Section 5.2.2).

Table 5.1: Summary of Grounding Costs to User and Robot

Highest Costs to User	
Type	*Definition*
Formulation	Create or edit plans
Production	Act of producing plan, including use of planning interface
Understanding	Interpreting what is meant by data from robot
Fault	Mistakes in plan
Lowest Costs to User	
Type	*Definition*
Start-up	Starting a new plan
Delay	Delaying the plan to revise it further before sending it
Speaker change	Sending the plan to the robot
Display	Presenting information to the robot by sending a plan
Repair	Making changes to the plan before sending it
Highest Costs to Robot	
Type	*Definition*
Production	Collection of science data
Delay	Waiting to collect or return data
Display	Presenting information to the user by returning data
Fault	Mistakes in data collection
Repair	Making changes to the plan at execution time
Lowest Costs to Robot	
Type of Cost	*Definition*
Formulation	Generating an ordered sequence of actions based on information from user
Understanding	Interpreting the plan from the user
Start-up	Starting a new execution cycle (assuming one execution cycle per day)
Speaker change	Returning data (assuming one upload per day)

Table 5.2: Impact of Robot Proxy on Grounding Costs to User

Highest Costs to User	
Type	*Impact of Robot Proxy*
Formulation	Reduced: Robot proxy assists user in developing plans
Production	Reduced: Robot proxy assists user in developing plans
Understanding	Reduced: Plans with fewer mistakes result in data which is easier to interpret
Fault	Reduced: Robot proxy works to ensure plan consistent with user's goals and robot's capabilities
Lowest Costs to User	
Type	*Impact of Robot Proxy*
Start-up	
Delay	No significant increase: Robot proxy interacts with user before plan is sent
Speaker change	
Display	
Repair	Reduced: User edits plan based on conversation with robot proxy

5.2.1 Traditional Common Ground Presentation-Acceptance

Clark and Brennan argue that grounding takes place through an exchange called the presentation-acceptance process [Clark and Brennan, 1991]. This is a two-phase process by which a contributor (A) conveys information to his partner (B) (see Figure 2.1). First is the presentation phase: A presents some information to B. A does this under the assumption that if A receives a certain amount of evidence from B, A can believe that B understands what A means. This is followed by the acceptance phase, in which B accepts the information from A by giving evidence that B understands what A means. B assumes that once A receives this evidence, A will recognize that B understands what A means [Clark and Brennan, 1991].

Clark and Wilkes-Gibbs have found that in conversation, when a speaker presents an initial reference which is not acceptable, either the speaker or the listener may repair, expand, or replace the reference (or request such a repair, expansion, or replacement) [Clark and Wilkes-Gibbs, 1986]. For example, if speaker A comments to listener B, "I heard that John is going on vacation tomorrow,", and listener B is unclear about exactly who A is referring to, B might signal the need for an expansion by asking “Who?” or “John Green or John Smith?” Clark and Wilkes-Gibbs also observe that the presentation-acceptance process is recursive: that is, a repair, expansion, or replacement might itself need to be repaired, expanded, or replaced. The process continues until the speaker receives sufficient evidence from the listener that the listener understood what the speaker meant.

5.2.2 Presentation-Acceptance with a Robot Proxy

As a starting point for the development of the Robot-Proxy Grounding process, this thesis argues that the presentation-acceptance process can be utilized to drive interactions at the level of the individual actions and parameters to be sent to the robot. In particular, Clark and Wilkes-Gibbs's detailed description of the acceptance process provides specific guidance for interaction design at this low level [Clark and Wilkes-Gibbs, 1986]. In conversation, when a speaker presents an initial reference which is not acceptable, either the speaker or the listener may repair, expand, or replace the reference (or request such a repair, expansion, or replacement). Within the context of exploration robotics, we can consider an individual action and its parameters to be analogous to a reference in conversation. If need be, a scientist may repair an action by editing its parameters, expand an action by providing additional information such as a name for a target, or replace an action in the plan with a different action. The presentation-acceptance process for one action may then proceed as follows:

In this process, the scientist presents an action which is either accepted by the robot proxy or not accepted. If the action is not accepted, the robot proxy indicates this to the scientist and requests a repair for the action. If the scientist needs additional information about the repair, the scientist can request that the robot proxy present additional information. Once the scientist has received enough information to make the repair, the scientist presents the repair to the robot proxy. The process then repeats, with the robot proxy checking the repair. This continues until the robot proxy accepts what the scientist has presented.

By generalizing this algorithm, it is possible to promote common ground across all three spacial levels of plan development (the action, the locale, and the entire plan). Thus, in this thesis, the implementation of Robot-Proxy Grounding for exploration robotics consists of presentation-acceptance processes which take place across the three spacial levels:

- Action level: The scientist presents an individual action to the robot (Algorithm 1).

Algorithm 1 Robot-Proxy Grounding presentation-acceptance process for a single action.

```
Scientist presents action $a_i$.
Robot proxy checks if $a_i$ is adequate (free of errors, consistent with other actions, etc.)
if $a_i$ is adequate then
  Robot proxy accepts $a_i$
  Robot proxy provides positive evidence of acceptance.
else
  Robot proxy presents negative evidence.
  Robot proxy requests a repair (a revision, expansion, or replacement).
while Scientist needs information do
  Scientist requests an expansion (further information about the inadequacy).
Scientist presents the requested repair, $a_i'$.
Let $a_i = a_i'$. Repeat.
```

- Locale level: The scientist presents a set of actions (which all take place at the same locale) to the robot.
- Plan level: The scientist presents a set of locales to the robot.

Beyond allowing the user to simulate the robot's actions (see [Hine et al., 1995]), this allows the robot to build common ground with the scientist regarding the relationships between different actions and the environment. In the future, the information gained during the grounding process could be provided to the robot for use at execution time. For example, in the event that an action fails in the field, the robot could then take advantage of this information to repair its plan in a manner consistent with the scientists' goals.

By utilizing a robot proxy and by structuring the grounding process as in Algorithm 1, it is possible to promote common ground between the scientist and robot within the grounding constraints described in Section 5.1.1; this also helps to minimize the costs to the scientist (Section 5.1.2). Robot-Proxy Grounding takes advantage of the **sequential** nature of the interaction by allowing the scientist and robot to present information in turn while allowing the other the opportunity to accept that information. The entire Robot-Proxy Grounding process takes place before the plan is sent to the robot because the scientist has the opportunity to **revise** the plan freely during this time. All of the information gathered during the grounding process can then be made available to the robot to **review** at execution time and utilize in case of automatic replanning.

In terms of the costs to the scientist, Robot-Proxy Grounding helps to reduce the cost of formulating and producing a plan by shifting some of the burden to the robot proxy. By promoting common ground with respect to the robot's capabilities and behaviors, the scientist is likely to develop a more accurate mental model of the robot, which should help to reduce the cost of understanding the data returned from the robot. Robot-Proxy Grounding also helps to prevent plans containing mistakes from being sent to the robot, which reduces the potential for fault costs.

5.3 Summary

As a starting point to develop a method to promote grounding between a scientist and a remotely-located robot, this chapter presents an analysis of the constraints on the grounding process for exploration robotics. This analysis suggests an approach of focusing on improving common ground before the plan is sent to the robot: in particular, the use of a "robot proxy," a novel method to promote common ground between a scientist and robot when real-time interaction is impossible. The robot proxy and scientist interact according to the Robot-Proxy Grounding presentation-acceptance process. This process occurs at three levels: individual actions, sets of actions, and the plan as a whole. Robot-Proxy Grounding thus allows the scientist and robot to build common ground while reducing the likelihood of failures due to erroneous plans.

In order to empirically examine the impact of a robot proxy on an exploration robotics task, a proof-of-concept study was designed and executed. In particular, this study focused on identifying whether a robot proxy helped improve task performance, fostered the development of more accurate mental models on the part of the user, and affected participants' perceptions of their performance at the task. The design and results of this study are presented in the following chapter.

Chapter 6

Proof-of-Concept Study

This chapter presents a proof-of-concept study of robot-proxy grounding in which a user and remotely-located robot collaborated on an exploration task. In the studied scenario, the user possesses scientific expertise but not necessarily detailed knowledge of the robot's capabilities, resulting in very little common ground between the user and robot. Because the robot is not available during mission planning, a robot proxy is introduced to build common ground with the user. This study demonstrated that the use of the robot proxy resulted in improved performance and efficiency on an exploration task, more accurate mental models of the robot's capabilities, a stronger perception of effectiveness at the task, and stronger feelings of collaboration with the robotic system.

6.1 Study Design and Method

In this experiment, a robot proxy-based interface which could provide feedback to users about their plans for the robot before execution was compared with with an interface that could only pass plans from the user to the robot. The study used a between-subjects design: each participant was randomly assigned to one of two conditions, the Robot Proxy condition or the Control condition. No physical robot was used in the study and all data were simulated.

The goals of the study were to understand the impact of a robot proxy-based interface on three particular areas relevant to common ground and exploration robotics tasks:

- *Task performance.* Which group is more efficient at completing the task successfully? How many correct and incorrect plans does each group send to the robot?

- *Mental model development.* After completing the task, which group knows more about the robot's capabilities and can make accurate predictions about the robot's behavior in novel situations?

- *Self-evaluation of performance.* How does the robot proxy-based interface affect participants' perceptions of their own performance and their feelings of collaboration with the system?

It is important to note that within the exploration robotics domain, particularly planetary exploration, communication with the robot is often infrequent and costly. In the case of the Life in the Atacama project, the science team in Pittsburgh could only communicate with the robot twice a day: once to receive data from the robot and once to transmit a new plan to the robot. Because of this asynchrony, the amount of time required by the planning process is a less significant concern than in other human-robot interactions. To be consistent with this aspect of the exploration robotics domain, my investigation of task efficiency focuses primarily on how many communication cycles are required to complete the task rather than the amount of time spent by the user to create plans for the robot.

6.1.1 Participants

Thirty-six participants were recruited from Carnegie Mellon University; eighteen were assigned to each of the two conditions. All participants were graduate students or staff selected for their background in computer science (e.g., members of the School of Computer Science). Participants were compensated for their time upon completion of the study; they received either refreshments or US$10 cash.

6.1.2 Procedure

After arriving at the lab, each participant was seated at a desktop computer. The experimenter explained that the computer would provide a description of the task and guide the participant through the task. The computer displayed the following scenario to each participant: "In this game, you will work with a Personal Exploration Rover (PER) which is located at an archeology site. Scattered around the site are fragments of a stone tablet covered in dirt. Each piece of the tablet contains words which can be combined to form a message. You must use the robot's plowing abilities to scrape the dirt

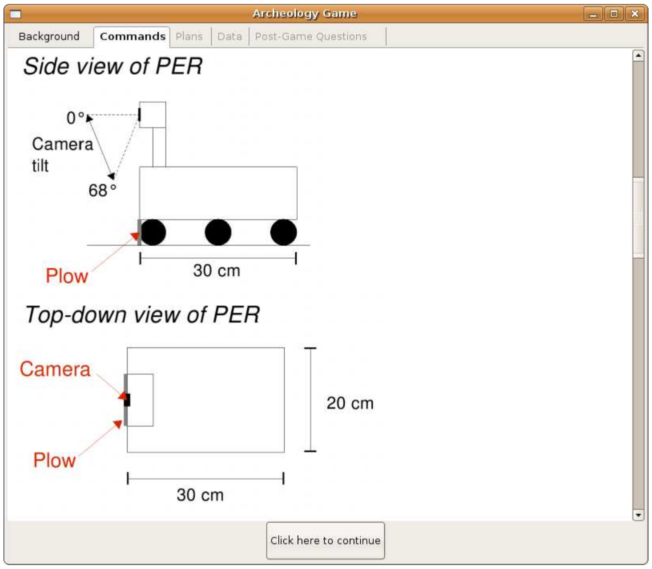

Archeology Game
Background
Commands
Plans
Data
Post-Game Questions
Side view of PER
0°
Camera tilt
68°
Plow
30 cm
Top-down view of PER
Camera
Plow
20 cm
30 cm
Click here to continue

Figure 6.1: The side and top-down views of the robot as presented to study participants.

off of the tablet fragments and reconstruct the message. Once you have examined all of the fragments, you will be asked what you think the complete message is.” The participant was then shown a summary of the entire procedure for playing the game (instructions only relevant to the Robot Proxy group are shown here in boldface text; these instructions were not shown to the Control group):

1. The computer will present you with a set of plans for the fragment.
2. You choose the plan that you want the PER to execute. It’s multiple choice: you choose one plan from a set of five plans.
3. Once you have selected a plan, you choose **whether you would like feedback on the plan or** whether you are ready for the robot to execute the plan.
    - **If you choose to receive feedback, the system will analyze your plan and provide you with additional information about it. You can then tell the robot to execute the plan, or you can select a different plan. You may request feedback *no more than two times* before telling the robot to execute a plan.**

4. When you command the robot to execute the plan, it will execute the plan and return a picture. After it has executed the plan, the PER automatically "resets" (goes back to the location where it was before it executed the plan).

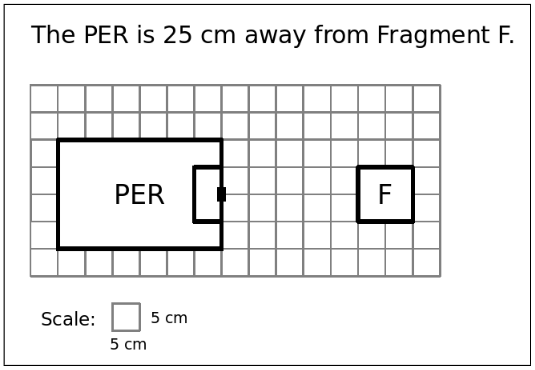
5. You decide whether to:
    - Send another plan to study the current fragment.
    - Go back to a previously visited fragment. The PER can automatically navigate to any fragment you have already seen so you can study it again. However, you may study the same fragment no more than three times all together.
    - Go on to a new fragment. The PER can autonomously navigate to a new fragment so that you can send a plan to study it.

The participant was also provided with a specific list of robot commands available to use as well as diagrams depicting the robot's shape and size, which were available to the participant throughout the game (Figure 6.1).

The archeology site contained three fragments. For each fragment, the participant was given a map indicating the location of the robot and the nearest fragment as well as a set of five possible plans the robot could execute (Figure 6.2). The participant was asked to choose one of these five plans for the robot to execute given that only one plan was correct (only the correct plan would result in a complete picture of the fragment). Additionally, participants in the Robot Proxy group had the option of requesting feedback about a possible plan up to two times per fragment. This feedback consisted of an image containing a scale drawing of the robot, the location of the targeted fragment, and the field of view of one of the robot's instruments (Figure 9.8). The feedback was designed to provide both contextual information about the robot's surroundings (the location of the fragment) as well as to encourage an accurate mental model of the robot's capabilities (the field of view of the instrument).

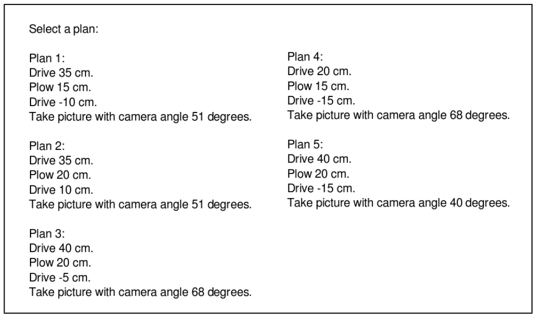
After the participant selected a plan to be executed, she was shown the resulting image and given the opportunity to review this data. Figure 6.4 shows the image returned from the correct plan for the first fragment. The participant could send up to three plans for each fragment; however, each time a participant chose to re-send a plan for a fragment, the participant was given a new set of five plans from which to choose. This prevented participants from using a process of elimination to find correct plans.

Thus, during the activity, each participant completed two major activities multiple times: selecting a plan for the robot (planning) and examining



The PER is 25 cm away from Fragment F.

(a) An example overhead map showing the location of the robot and the fragment.



Select a plan:

Plan 1:
Drive 35 cm.
Plow 15 cm.
Drive -10 cm.
Take picture with camera angle 51 degrees.

Plan 2:
Drive 35 cm.
Plow 20 cm.
Drive 10 cm.
Take picture with camera angle 51 degrees.

Plan 3:
Drive 40 cm.
Plow 20 cm.
Drive -5 cm.
Take picture with camera angle 68 degrees.

Plan 4:
Drive 20 cm.
Plow 15 cm.
Drive -15 cm.
Take picture with camera angle 68 degrees.

Plan 5:
Drive 40 cm.
Plow 20 cm.
Drive -15 cm.
Take picture with camera angle 40 degrees.

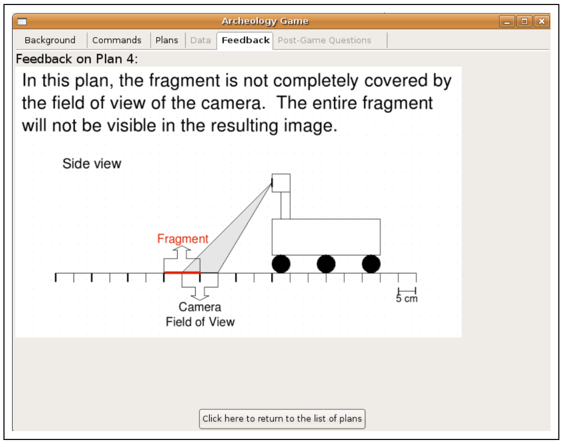
(b) An example list of plans.

Figure 6.2: For each fragment, participants were given (a) an overhead map and (b) a set of five possible plans.

the image that was returned from a selected plan (data review). A *cycle* is defined as as one planning session followed by one data review session. Each participant examined three different fragments for a total of three trials. After completing all three trials, the participant was asked a set of questions about her experiences. These questions included self-evaluation questions and questions intended to evaluate the participant's mental model of the robot (Section 9.5). The entire process lasted approximately thirty to forty-five minutes per participant.

6.1.3 Simulation

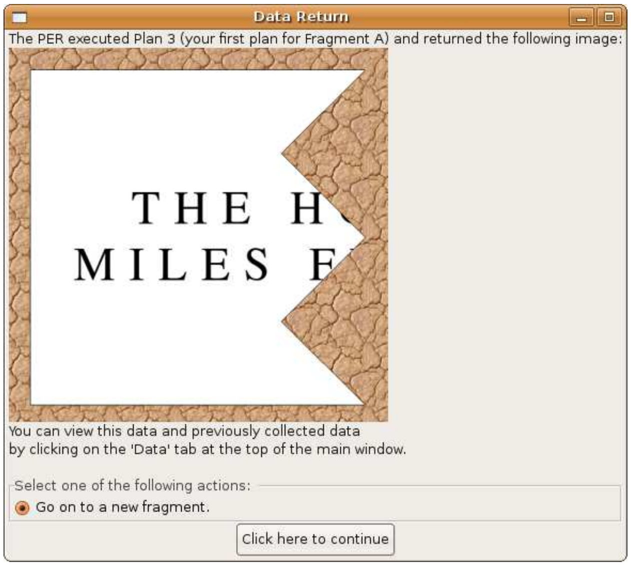
Because the task involved a non-collocated robot, software could be used to simulate a physical robot without sacrificing the fidelity of the human-robot interaction. Software was used to simulate the robot's actions and


Archeology Game
Background
Commands
Plans
Data
Feedback
Post-Game Questions
Side view
Fragment
Camera
Field of View
5 cm
Click here to return to the list of plans

Feedback on Plan 4:

In this plan, the fragment is not completely covered by the field of view of the camera. The entire fragment will not be visible in the resulting image.

Figure 6.3: An example of the type of feedback shown to participants in the Robot Proxy group.


Data Return
THE H
MILES F
Click here to continue

The PER executed Plan 3 (your first plan for Fragment A) and returned the following image:

You can view this data and previously collected data
by clicking on the 'Data' tab at the top of the main window.

Select one of the following actions:

● Go on to a new fragment.

Figure 6.4: A screenshot containing an image of the first fragment after it has been completely cleaned.

the data returned from the robot. This simulation was not a grid world; the grid shown in Figure 6.2(a) was provided to participants to allow them to measure distances. At the end of the experiment, the participant was informed that she had been using a simulated robot. It is important to note that the simulated robot's actions were not stochastic: the robot always executed plans consistently and perfectly; poor-quality images of fragments were solely the result of the incorrect plans chosen by participants.

6.1.4 Dependent Variables

Table 6.1 illustrates the dependent variables measured in this study. The performance and mental model variables were derived from the requirements of the task itself. Mental model questions included questions about the robot's physical properties as well as questions about how the robot would perform in situations similar to (yet slightly different from) those seen during the task. For all six self-evaluation questions, participants were given a Likert scale from 1 to 5 and asked how strongly they agreed with a particular statement (1 = "Strongly disagree", 5 = "Strongly agree"). Factor analysis was used to confirm that one question on perceived efficiency and three questions on confidence could be combined into a coherent factor "Effectiveness"; the Cronbach's alpha of this factor was calculated to be 0.74, which suggests that the factor is internally consistent. The question about the participants' feelings about collaborating with the robot was motivated by work by Hinds et al. on human-robot collaborative tasks [Hinds et al., 2004].

6.2 Results

My data analysis focused primarily on understanding differences between the Robot Proxy and Control participants and how participants' performance changed over the three trials for the dependent variables given in Table 6.1. The multivariate correlations between dependent variables are shown in Table 6.2.

Table 6.1: Dependent Variables

Variable	**Measure**
Task Performance	
Accuracy	Did the participant successfully identify the secret message?
# Cycles	How many cycles were required in order for the participant to reveal the entire secret message?
Review-Data Ratio	What proportion of the participant's time spent on the task was used to review data from the robot?
Mental Model Development	
Quiz Score	What percentage of questions about the robot's capabilities did the participant answer correctly? (14 questions, included both true-false and multiple-choice questions)
Self-Evaluation of Performance	
Effectiveness	To what extent did the participant agree or disagree that she was efficient at performing the task and felt confident during the task? (4 questions)
Fun	To what extent did the participant agree or disagree with the statement, "I had fun playing this game."?
Collaboration	To what extent did the participant agree or disagree with the statement, "When developing plans, I felt I was collaborating with the system."?

Table 6.2: Multivariate Correlation

	# Cycles Trial 1	# Cycles Trial 2	# Cycles Trial 3	Review-Data Ratio	Quiz Score	Effectiveness	Fun	Collaboration
# Cycles Trial 1	1.000	**0.450**	0.319	**0.352**	-0.168	-0.272	0.127	-0.060
# Cycles Trial 2		1.000	**0.481**	**0.499**	-0.290	-0.224	0.045	-0.143
# Cycles Trial 3			1.000	**0.613**	**-0.581**	-0.274	0.190	-0.270
Review-Data Ratio				1.000	-0.252	-0.259	0.222	-0.204
Quiz Score					1.000	**0.355**	-0.235	-0.134
Effectiveness						1.000	0.005	0.283
Fun							1.000	0.230
Collaboration								1.000

Statistically significant ($p < .05$)

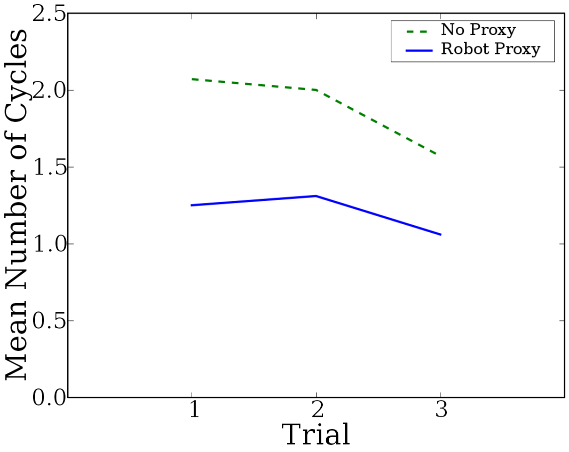


Figure 6.5: Mean number of cycles per trial for participants who successfully completed the task.

6.2.1 Task Performance

Overall, 30 of 36 participants successfully completed the task by revealing the entire secret message; a plot of the mean number of cycles used per trial is shown in Figure 6.5. To better understand participants' performance, a two-way repeated measures analysis of variance (ANOVA) was conducted on the data from the 30 participants who were successful with condition as a between-subjects variable and trial number as a within-subjects factor. There was a main effect for condition ($F[1, 28] = 37.52$, $p < .001$), indicating that participants in the Robot Proxy group needed significantly fewer cycles than those in the Control group. The main effect of trials was also significant ($F[2, 56] = 4.44$,$p < .05$). This shows that participants required significantly fewer cycles during the later trials, which indicates that learning occurred over the trials. There was no significant interaction effect between condition and trials.

A two-way repeated measures ANOVA was also conducted on the number of correct plans sent to the robot with condition as a between-subjects variable and trial number as a within-subjects factor. There was a main effect for condition ($F[1, 34] = 11.17$, $p < .01$): participants in the Robot Proxy group sent significantly more correct plans to the robot. 91% of trials in the Robot Proxy condition resulted in a correct plan being sent to the robot as opposed to 59% of trials in the Control condition. There was no

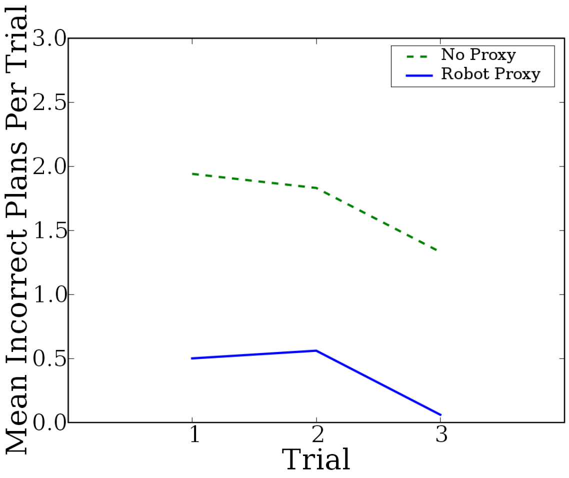


Figure 6.6: Mean number of incorrect plans sent to the robot per trial.

significant main effect of trials; this was the expected result because each trial ended as soon as one correct plan was sent to the robot. There was also no significant interaction effect between condition and trials.

The ANOVA on the number of incorrect plans also showed a main effect for condition ($F[1, 34] = 38.4$, $p < .001$), meaning that participants in the Robot Proxy group sent significantly fewer incorrect plans to the robot than participants in the Control group (Figure 6.6). There was a significant main effect for trials ($F[2, 68] = 3.46$, $p < .05$), which indicates that participants sent significantly fewer erroneous plans to the robot during the later trials, which provides further evidence of learning. There was no significant interaction effect between condition and trials.

In addition, a two-way repeated measures ANOVA was conducted on the review-data ratio (the proportion of time spent on data review) with condition as a between-subjects variable and trial number as a within-subjects factor. The main effect of condition was highly significant ($F[1, 34] = 36.4$, $p < .001$), meaning that participants in the Robot Proxy group used much less of their time reviewing data from the robot than did participants in the Control group (Figure 6.7). There was no significant main effect of trials nor a significant interaction effect between condition and trials. One possible explanation for the main effect of condition is that participants in the Control group may need more time to review the data because they must both interpret the data and use it to improve their mental models. By contrast, robot proxy users may update their mental models based on

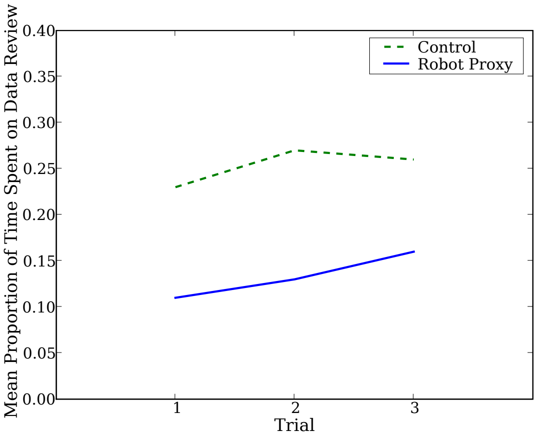


Figure 6.7: Mean proportion of time spent reviewing data from the robot per trial.

the feedback they receive during the planning process and so do not need to spend as much time reviewing the data. This finding is also supported by the correlation analysis, which indicates that the proportion of time spent reviewing data was significantly positively correlated with the number of cycles used in each trial (Table 6.2).

6.2.2 Mental Model Development

After completing the activity, participants were asked fifteen questions related to the robot's physical structure and capabilities. One question of the fifteen was not answered correctly by any participant and was therefore dropped from the analysis. A plot of the least squares mean score of each question by condition is shown in Figure 6.8. A multiple analysis of variance was conducted on participants' scores for each quiz question with condition as a between-subjects variable and question number as a within-subjects factor. The results indicated that there was no main effect for condition ($F[1,34] = 2.55$, $p > .1$). There was a significant main effect for question number ($F[12,23] = 15.26$, $p < 0.001$) as well as a small interaction effect between question score and condition ($F[12,23] = 1.86$, $p < 0.1$). This indicates that, while there was no significant difference in average total quiz score between the groups ($M_{RP} = 54\%, SD_{RP} = 15\%$, $M_{Control} = 45\%, SD_{Control} = 18\%$), whether or not a participant answered a particular question correctly was related to her group membership. This

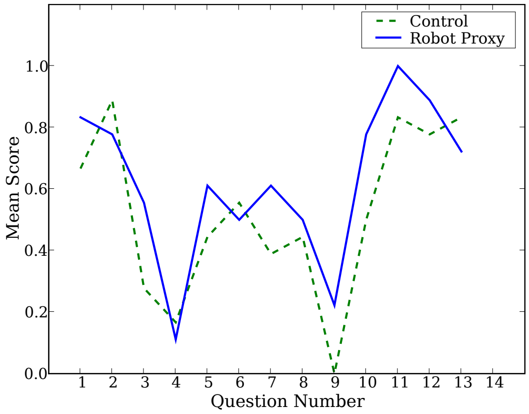


Figure 6.8: Least squares mean of score on each question by condition.

result is also reflected in Figure 6.8: while the average difficulty of questions varied, members of the Robot Proxy group tended to score higher on most questions.

Also observed was a negative correlation between total quiz score and the number of cycles required for each trial; the magnitude of the correlation increased over time (Table 6.2). This suggests that participants who scored higher on the quiz needed fewer cycles to complete the task. This provides evidence that a higher score on the mental model quiz was associated with better performance on the task.

6.2.3 Self-Evaluation of Performance

Regression analysis was used to ascertain if the presence of the robot proxy could explain the differences in participants' self-evaluation of their performance on the task. The regression of efficiency on condition was significant ($M_{RP} = 3.82$, $M_{Control} = 3.08$, $r^2 = 0.19$, $p < .01$) (Figure 6.9). The regression of collaboration on condition was also significant ($M_{RP} = 3.0, M_{Control} = 2.28$, $r^2 = 0.11, p < .05$) (Figure 6.10). There was no significant difference with respect to participants' ratings of the task as fun. This shows that participants' ratings of their own effectiveness and feelings of collaboration with the system were strongly impacted by their interaction with a robot proxy.

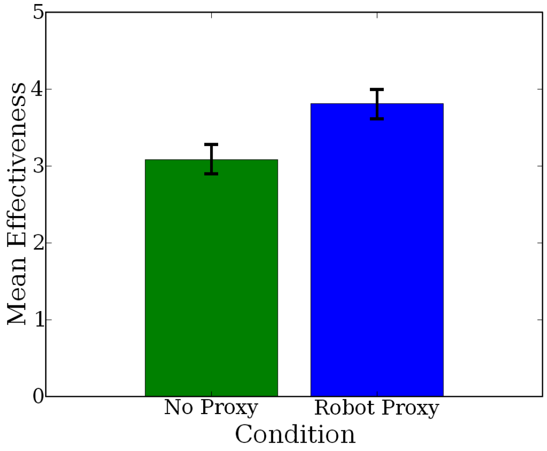


Figure 6.9: Least squares means of self-evaluation of effectiveness.

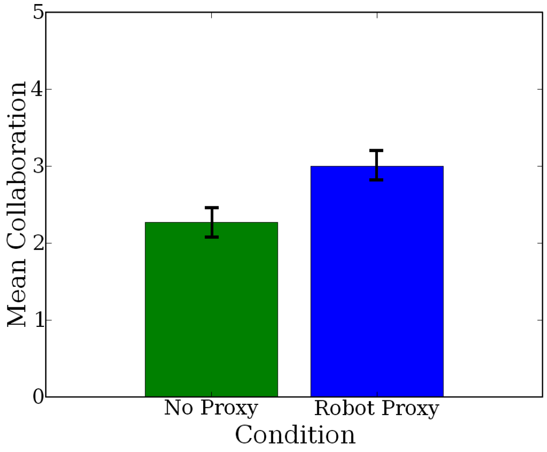


Figure 6.10: Least squares means of self-evaluation of collaboration with the system.

6.3 Discussion

The participants were representative of the highly-trained scientists who currently participate in robotics exploration missions due to participants' strong mathematics backgrounds and lack of direct experience remotely controlling robots for scientific exploration. The task was fairly short and straightforward: there were only three trials, and most participants successfully determined the secret message. While the results indicated that participants did learn over the course of the three trials, the Robot Proxy group performed significantly better than the Control group on the first trial, a strong example of one-trial learning. With a longer, more complex task, one would expect to see more differences between the two groups in terms of learning, performance, and mental model development; this could be verified in a future experiment. One would also expect more significant differences between the groups based on the type of the feedback provided; different forms of feedback (i.e. three-dimensional images, video, etc.) could be compared in future studies. However, the basic feedback used in this study was still sufficient to highlight the benefits of a robot proxy.

This study has demonstrated proof-of-concept for using a robot proxy to increase common ground between a user and a remotely located robot as they complete an exploration task. Users who could request feedback about their plans before those plans were sent to the robot were more accurate (sent many fewer incorrect plans) and more efficient (required fewer cycles in order to successfully complete the task). Robot proxy users were also able to develop a better mental model of the robot, which was correlated with improved efficiency. The use of the robot proxy also explains participants' stronger feelings of effectiveness at the task and collaboration with the system, the benefits of which have been shown in [Hinds et al., 2004]. In addition, this study found that that individuals who completed the task in fewer cycles also spent less time reviewing data from the robot relative to their total amount of time spent on the task.

The results of this proof of concept study validated the use of a robot proxy as presented in Chapter 5; after the full robot proxy implementation was completed, it was also evaluated as described in Chapter 9.

Chapter 7

Robot Proxy Requirements

This chapter introduces the three major components of the proposed robot proxy for HRI and presents the requirements which the robot proxy should meet in order to promote common ground with the user. In particular, the three major components of a robot proxy are a goal representation, which captures the plan as well as additional, higher-level information; and a goal validation system, which ensures that the goals are internally consistent and consistent with the robot's capabilities; a robot model, which captures the robot's geometry and capabilities. The robot proxy system architecture and implementation details for each of these components are provided in the next chapter.

7.1 Robot Proxy Overview

In remote exploration robotics, it is not possible for the user and robot to interact directly in real-time. An analysis of the other constraints and costs which are associated with building common ground between the user and robot indicates that a robot proxy can be used to "converse" with the user in the robot's place (Chapter 5). The robot proxy is a software system which runs off-board the robot and is available to the user during mission planning, before the plan is sent to the robot for execution. In order to be able to promote common ground, the robot proxy software system must be able to explicitly represent users' goals, model the physical robot's capabilities and behaviors, and use this information to reason about whether the goals are physically possible and internally consistent. Thus, the robot proxy has three major components: a goal representation, a goal validation system, and a robot model. The justification for each of these components and the requirements which each component must meet are presented in the following sections.

7.2 Goal Representation

In order for the robot proxy to ensure that the actions which will be executed meet the scientist's needs, the robot proxy must be able to explicitly represent the scientist's goals. This goal representation allows the robot proxy to capture knowledge in addition to a sequence of actions and reason about whether the plan will meet the scientist's needs. The following sections discuss a) why a sequence of actions alone is not sufficient in order to build common ground and b) the requirements which the robot proxy goal representation must meet.

7.2.1 The Need for Goal Representation

The goal representation presented in this thesis provides a way to represent information in addition to a normal robot plan (sequence of actions). This goal representation serves as the foundation for the "conversation" which takes place between the scientist and robot proxy. The following example illustrates how the lack of explicit goal representation adversely affected the LITA project in 2005. Without an explicit representation of the scientists' goals, it was virtually impossible for the science team to ensure that the engineering team and the robot made decisions that were consistent with the science team's desires.

This example focuses on the Standardized Periodic Sampling Unit (SPSU). The SPSU was a set of actions intended to perform the same set of actions in multiple locations. The design and acquisition of this complex data product was a major focus of the science team in 2005. The scientists' intent in creating the SPSU was indicated by the following conversation between three scientists (Riley, Sidney, and Taylor) on Sol 4:

> Riley tells Sidney and Taylor to think of a "standard data product" to do at each geologic unit. Taylor asks if "we" can do the standard transect. Riley says that "we" could.

The term "Standard periodic sampling unit" was coined in the plan to be executed on Sol 4. The initial SPSU was described in this plan as:

> "Standard periodic sampling unit" rationale = 180m transect w/max 6 RGB/chlor FI, max 2 full FI follow-ups, max 70 flex

> minutes for FI follow-ups, R_Navcam every 8m, ~9 SPI w/SOTF, ~6 high-res RGB SPI context (el-70) waypoints.

This SPSU involved taking a full set of images with the fluorescence imager (FI) at the starting location, a certain number of chlorophyll FIs on the way to the ending location, and a full FI at the ending location. However, there was no way for the science team to group these actions together to represent their constraints. In order for the data product to be standard, all of the actions needed to be completed. As these scientists discussed on Sol 9:

> Riley says that the field team skipped the last FI. "Aren't they supposed to do FI?" Sidney asks. Taylor suggests that "we" need to reiterate that "we" need FIs at either end of an SPSU. Riley suggests doing the FI the next morning.

Despite the fact that the term "SPSU" was included in human-readable robot plans beginning with the plan for Sol 4, the engineering team appeared to be unaware of the specifics of what the science team was trying to accomplish. The first recorded use of the term "SPSU" by an engineer was not until Sol 16. Without an explicit representation tying together the various actions and locations that form an SPSU, the engineering team struggled to decide about what actions should be kept or dropped, as indicated in this conversation between engineers (Avery, Bailey, and Casey) on Sol 17:

> Avery asks about the rest of the day. Bailey explains that the plan is to drive a kilometer, do an FI, drive another kilometer and do an FI. Avery says that he thinks they should start here and the science team will have to deal with the fact that they only have half of a 180 meter transect. Casey radios Avery and says that his recommendation is that if they're going to skip the first half of the transect, they should skip it all. He says that the science team has said that there isn't any value to half of a transect. Avery asks where Casey has seen this. Avery asks where Casey is getting this information. Casey says that he isn't sure, but it was either in the support doc or in the science team report. Avery asks if it was in support doc today and Casey says that it wasn't, but he thinks it was labeled as part of the same SPSU. Avery says that he still wants to start from this point and, he says, "maybe we'll get feedback from the scientists that that was a bad idea, but I'd like to just move forward."

When the science team received the data that resulted from this decision, they were displeased with the result:

> “This is an invalid SPSU,” Riley says. Taylor says that “we” need to tell the field team that. ...Taylor says, “Our SPSU is stinky, it’s more like a P-U.”

Overall, 10 of 201 problem instances (5%) in 2005 involved invalid SPSUs. More generally, 8.4% of instances involved situations in which the robot or engineering team were unable to correctly interpret the scientists’ plan. This demonstrates the need for an explicit representation of scientists’ goals. In this case, the engineering team lacked awareness of what the science team was trying to do through the plans that were sent to the robot. In an actual space exploration mission, the engineering team would not be collocated with the robot, and the robot would have to make autonomously the same kinds of decisions that the engineering team made. Representing scientists’ goals, including the relationships between actions and constraints on actions, and providing this information to the robot helps to promote common ground between the science team and the robot and to ensure that the actions executed in the field are consistent with what the science team is trying to achieve.

7.2.2 Goal Representation Requirements

In addition to a list of specific actions to be executed, the robot proxy needs access to information about the relationships between actions (constraints). The representation of the user’s goals presented in this thesis derives from analysis of how the LITA science team thinks about exploration (their exploration paradigm) and their discussions of the types of constraints and relationships between data products. In particular, the use of logistical, spatial, and ordering constraints was observed. This list is not necessarily exhaustive, but LITA scientists did use or discuss every item. The formulation of these constraints and relationships allows them to be represented on-board a robot, a basic step which is needed to build common ground between the robot and science team.

Logistical Constraints

Scientists’ Exploration Paradigm
According to observations of the LITA science team during 2004 and 2005,

in general, the scientists' exploration paradigm can be described in terms of three levels:

- Study of a specific point on the ground (target)
- Study of a location at which the robot is situated (locale)
- Study of a geologic unit

During the first few days at a site, one scientist would create a geologic map based on a satellite image, dividing the terrain in the image into a number of geologic units (contiguous areas believed to have similar properties, formation, history, etc.) To create a geologic map, the scientist divides the orbital image into some number of partitions. Each partition is associated with one geologic unit (multiple partitions may be associated with the same geologic unit). Thus, each pixel in the satellite image belongs to exactly one geologic unit. (See Figure 7.1 for an example of a geologic map.)

When the robot stopped at some place (locale), the scientists would often request instrument measurements to be taken of specific points (targets) as well as measurements that provided general information about the locale (e.g. panoramas). This data was then used to infer properties of the geologic unit containing the locale. Scientists also chose locales in different units and talked about comparing/contrasting the units. The science team produced a report every sol during 2005, summarizing their investigations and findings. Of the 12 sols for which a complete daily summary exists, all 12 mention specific locales and targets, and 11 of 12 mention specific units.

In addition, the scientists tended to talk about performing groups of actions together (which they referred to as using a "template"). The same set of actions was often repeated within a sol or across multiple sols, as was the case with the SPSU (requested in 14 of 20 plans) and the HASTA (another type of standard transect, requested in 7 of 20 plans).

The robot proxy supports both user-defined targets as well as automatic target identification.

In order to limit the scope of the problem, this work focuses primarily on activities that take place at a locale. This thesis does not address the problem of moving locales within or between units.

In addition, it is important to note that every action may or may not have a target, but every action will have a locale. An additional constraint is that all actions with the same target must take place at the same locale.

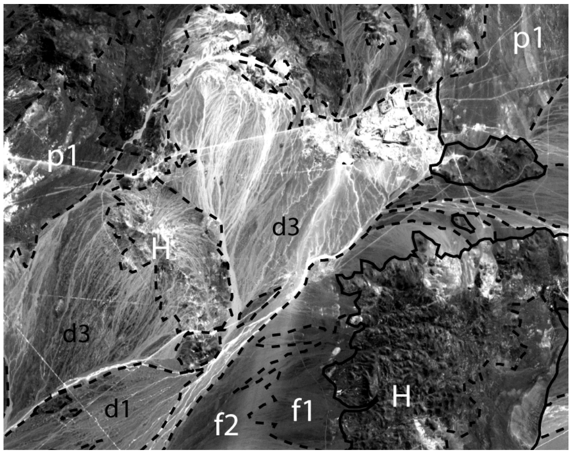


Figure 7.1: Part of a geologic map of Site F created by one of the members of the LITA science team.

Inclusivity Constraints
The fact that the science team was unable to specify that a group of actions was “all-or-nothing” made it difficult for them to ensure the collection of a complete SPSU, as described above. This type of constraint, a group of actions which all must be executed successfully in order to be worthwhile, is referred to as an inclusivity constraint.

Spatial Constraints

Collocation Constraints
As discussed above, the science team frequently attempted to coordinate multiple actions. This was most challenging for them to plan when multiple instruments were to be targeted at the same point. Every plan the scientists created also involved multiple actions’ being requested at the same locale. Using the current planning software, it was only possible to explicitly plan for this second case.

The first locale of the Sol 1 plan illustrates both of these scenarios:

> L10 atmosphere pan for weather,full FI, drive to “look at toes spi and spec”,plow 2 m, FI chlor on either side of furrow to check for overturned rocks, spec pan for part of plowed area, TIR on rock targets, move up the furrow 0.5m, FI chlor on either side, move up furrow 0.5m, FI chlor on either side, move up furrow 0.5m, FI chlor on either side.

All together, there were 19 science actions requested at this locale. Three of these actions were targeted on the same point (the first full FI, SPI, and spec). This illustrates the importance of being able to associate multiple actions with a target or locale to ensure that the data covered the same portion of the physical scene.

Distance Constraints

As seen through the design of the SPSU, the science team was interested in specifying the distances between actions. This included distances at the locale level (taking two FIs so many centimeters apart) as well as distances between locales (SPSUs consisted of two 90-meter segments). The degree of accuracy required was also important: a high degree of accuracy was needed for SPSUs in order to ensure that the data returned would be usable for statistical analyses.

Ordering Constraints

The planning software used by the science team forced them to specify the exact order of all actions. Ordering was often necessary in order to achieve the scientists' goals (taking an FI of a plow requires that the plow be completed before the FI).

Locales were also ordered, which represented the order in which the robot was to travel between them; the goal representation also requires that locales be ordered.

Requirements Summary

This analysis indicates that, in addition to representing actions and parameters, the goal representation utilized by the robot proxy must support:

- Logistical constraints: The actions to be executed, the targets they may have, and the locales to which the robot will be sent
- Inclusivity constraints: Which actions should be carried out as an "all-or-nothing" group
- Spatial constraints: Where each action should be located, whether actions or locales should be separated by a certain distance
- Ordering constraints: The order in which actions should be executed and locales should be visited

7.3 Goal Validation

The goal representation described above provides an explicit means to represent the actions the scientists wish to execute and also the constraints between them. In order for a scientist to communicate a set of goals to the robot proxy, the scientist uses a graphical interface to provide the following information:

- A sequence of locales placed on an orbital map which indicates where the robot should travel and in what order
- Optionally, a set of distance constraints between locales indicating how far they should be apart at what accuracy
- At each locale, a sequence of actions and their parameters
- Optionally, at each locale the scientist may specify:
  - Which actions are associated with which targets
  - Distance constraints between actions at the locale
  - Which actions are members of inclusive groups

Once this information has been provided to the robot proxy, the robot proxy must ensure that the information is both internally consistent and consistent with the robot's current capabilities. The following section discusses why goal validation is needed in order to promote common ground between a scientist and robot. The details of the goal validation algorithm will be presented in Section 8.4.

7.3.1 The Need for Goal Validation

The goal representation presented above provides the means for a scientist to specify explicitly what actions should be executed and the relationships between those actions. Goal validation is the process by which the robot proxy checks that these actions are physically possible and meet the specified constraints given the robot's current capabilities. This is especially useful in the case in which the science team desires a complicated sequence of actions that rely on simultaneous knowledge of the location of an instrument, the position of the instrument relative to some target, and the position of the instrument relative to the robot or other landmarks in the environment.

One example of this situation that was observed was the science team's attempt to take an FI over a plowed trench. This was a strategy for finding life that the science team developed in 2004 and continued to use in 2005. It involved using the robot's plow to remove dirt from the surface of the ground and then taking an FI of that area. Finding exactly the right parameters to use to coordinate these activities proved to be challenging, as indicated by this conversation on Sol 3 (scientists Riley, Sidney, and Taylor; engineers Dakota and Emerson):

> At 6:48 p.m., Sidney says that Dakota forwarded him an e-mail from Emerson. Sidney says that it says that the plow is 24 cm wide and generates a 20 cm trench. Sidney says that "we" were assuming twice that. Taylor agrees, saying that x equals zero is not zero relative to the robot. Riley replies, "It's a good thing it's called zero though, that makes sense." Riley laughs, and Taylor and Sidney laugh with him. They talk about how x equals zero is when the imager box is in the center of the robot, not the camera lens. To take a centered image, they need to take the FI at 6 cm. Sidney and Riley talk about what parameters to use to do fluorescence imaging over plow furrows. Sidney says that Emerson expects that 170 and -70 would work, but Sidney says "that doesn't make sense." Dakota suggests that Emerson meant millimeters. Sidney agrees that that would make more sense, but he says that 17 cm would still be too big. "We just have to keep this in mind," Sidney says.

In order to accomplish the task, the science team must simultaneously consider not only the size of the FI field of view and the size of the trench, but also the relationship between the position of the FI camera and the imager box, the relationship between the position of the imager box and the center of the robot, and the relationship between the center of the robot and the trench. Keeping track of all of the necessary parameters is difficult even for people with extensive robot experience, as illustrated above.

Allowing a robot proxy to verify that the user's goals are feasible based on a model of itself reduces the burden on scientists to try and remember all of the constraints that must be met in order to generate an error-free set of actions. The robot proxy can verify not only that the actions are physically possible but also that they will generate robot behavior that will meet the specified constraints. In this instance, once the robot has the information that scientists are trying to target the FI over a plowed trench, it can suggest alterations to the parameters to ensure that the plan can meet this goal. The goal verification process may help to resolve problems which arose due to the science team's not having correct information about rover capabilities (6.1% of instances) and those that arose due to errors in the scientists' plan (5.2%); fewer erroneous plans would also help to reduce the number of bad data products (26.3%) (Table B.1).

This feedback loop also allows scientists to learn more about the robot's capabilities and refine their plans without having to wait an entire execution cycle to see whether or not they chose appropriate parameters. The day after the above discussion took place, the scientists learned that the plan had been unsuccessful. "We tried to get an image of the plow trench, but we're way off," one scientist commented. The proposed goal validation system will help to reduce the likelihood of this type of situation.

At this time, there is one other similar tool currently under development: the Mission Repair Feature of the *MissionLab* mission planning tool [Moshkina et al., 2006]. This tool is a graphical user interface designed to support the creation of missions for autonomous mobile robots; example tasks include moving to a certain location or surveying a room. After a user has specified a mission, he can "play back" the plan and observe robots' behavior. If the user observes erroneous behavior, he can use the Mission Repair Feature. This feature uses a wizard-style interaction to identify and correct errors in the plan. Moshkina et al. demonstrate through user studies the effectiveness of this technique in reducing the number of unsuccessful missions and increasing perceived ease of use of the *MissionLab* software. By contrast, the goal validation system presented in this thesis does not require users to sit and watch an entire simulated execution cycle, nor does it rely on users to find all errors themselves. Moshkina et al. [Moshkina et al., 2006] do provide empirical evidence that providing users with tools which support repairing plans results in fewer erroneous plans. Instead of providing a visualization tool, as *MissionLab* does, this thesis includes explicit modeling of users' goals and of the robot itself because this allows for automated detection of errors, generation of remedies, and building of common ground. Given the challenges inherent in planning for a complicated robot, the goal validation system helps to reduce the burden on users to create a perfect plan while increasing their knowledge about the robot and its capabilities.

7.3.2 Goal Validation Requirements

Based on the structure of the goal representation, goal validation needs to occur at three levels:

- At the action level: Given the current functionality of the robot and its instruments, are the parameters of the actions acceptable?

- At the locale level: a) Given the targets which the scientist as specified as well as targets which have automatically been detected, do the actions associated with those targets correctly cover them? b) Given the distance constraints between pairs of actions at the locale, will those actions be separated by the correct distance when executed on-board the robot?
- At the plan level: Given the distance constraints between pairs of locales, will those locales be separated by the correct distance when executed on-board the robot?

The process of goal validation should then proceed as follows:

1. The robot receives a set of goals from the scientist.
2. The robot checks that these goals are internally consistent and consistent with the robot model.
3. Depending on the outcome of these checks, feedback about the user's goals is generated and provided to the scientist. In the event that a check fails, feedback will be displayed to the user which includes information about why the check failed, the relevant actions and constraints involved, and at least one possible course of action.

Before the final plan is sent to the robot, this validation cycle provides a means for the robot proxy to detect errors in actions, errors in constraint satisfaction or targeting, and inconsistencies in the plan; and suggest modifications to actions, constraints, or targets to eliminate these errors/discrepancies.

Common ground between the science team and the rover is built over time as the science team validates, edits, and re-validates their goals. Throughout this process, the robot proxy continues to gain more information about the scientists' goals, and the science team receives more information about the robot and its capabilities.

This goal validation helps assure that all actions are free of errors and that all constraints are satisfied. The robot's representation of the scientist's goals is updated as the scientist provides additional information through the user interface, increasing the common ground between the robot and the scientists. The system also promotes grounding by providing clear, easily understandable information about the errors and discrepancies it detects. The feedback helps the science team learn about the robot's capabilities as it helps them improve their plans.

7.4 Robot Model

The purpose of the robot model is to represent the robot's physical attributes and capabilities. In order to check whether a proposed plan can be executed on the robot, the robot model must include:

- A geometric model of the robot and the locations of its instruments.
- Mathematical models which can be used to compute the field of view of each instrument given a set of action parameters.
- The range of acceptable values for each action parameter.
- Current information about the robot's capabilities (e.g. which instruments are not currently functioning).
- A model of the robot's environment, such as the interaction between the robot's wheels and the terrain, as this affects the motion of the robot.

This will allow the Goal Validator to check whether the robot can physically execute a particular command as well as to determine the field of view of the robot's instruments.

7.5 Summary

This chapter introduces the three major components of a robot proxy: an explicit representation of scientists' goals, a goal validation system which can check if the goals are internally consistent and consistent with the robot's capabilities, and a model of the robot's geometry and capabilities. An explicit goal representation is necessary in order for the proxy to be able to reason about what the scientist is trying to do and for the robot to be able to make better decisions at execution time if replanning is needed. This representation must include support for actions, targets, locales, and the relationships between them. A goal validation system is required so that the robot proxy has a way to check that a sequence of actions will meet the constraints given by the user. The goal validation must take place at the level of individual actions, sets of actions at locales, and sets of locales (the entire plan). Through the use of a goal representation, goal validation system, and robot model, the robot proxy is able to promote common ground between the user and robot.

Chapter 8

Robot Proxy Design and Implementation

This chapter presents the design and implementation of a robot proxy. The design of the proxy focuses on three major components: a goal representation, a robot model, and a goal validation system.

8.1 System Overview

The most important classes in the system implementation are shown in Figure 8.1. The robot proxy itself is comprised of the Goal Validator and all classes underneath it.

The Plan Manager is responsible for storing the science goals and coordinating the scientist interface with the robot proxy. The Plan Manager contains a Science Goals object (Section 8.2.3) which holds all of the current information about the user's goals. When the system is started, the Plan Manager instantiates the Scientist Interface, Goal Validator, Robot Manager, and Command Translator.

The Scientist Interface is a graphical user interface which allows the user to choose the locales the robot will visit, the actions which will be executed at each locale, and view the data returned from the robot. The Scientist Interface also allows the user to request feedback from the robot proxy on a specific locale or the entire plan.

The Science Goals class provides an explicit representation of the user's goals, including actions and their parameters, locales, and the constraints between them.

The Goal Validator contains the methods necessary to check that an

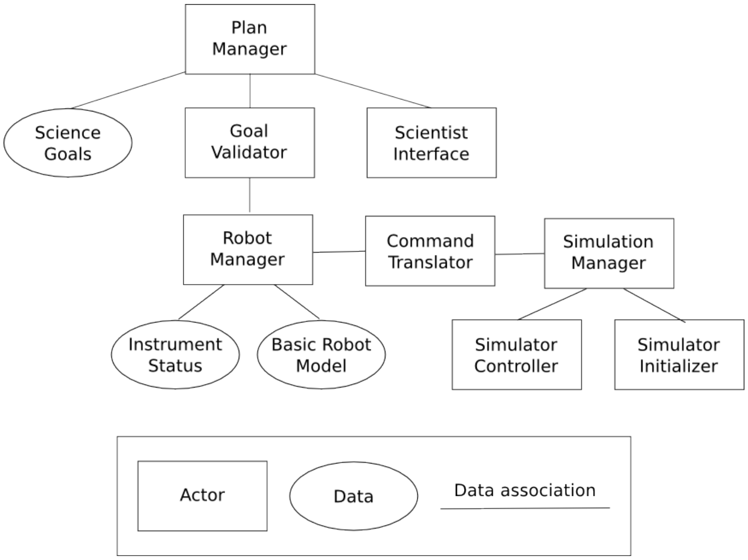

Plan Manager
Science Goals
Goal Validator
Scientist Interface
Robot Manager
Command Translator
Simulation Manager
Instrument Status
Basic Robot Model
Simulator Controller
Simulator Initializer
Actor
Data
Data association

Figure 8.1: System diagram

individual action is physically possible, that the actions at a locale meet the user's constraints, and that a set of locales meets the user's constraints.

The Command Translator is responsible for starting up the simulator, simulating actions, and requesting information about the state of the robot in the simulator. When the Simulation Manager receives a request from the Command Translator, it delegates to the Simulation Controller. The Simulation Controller communicates directly with the *gazebo*[1] simulation and calls the appropriate *gazebo* method to simulate an action or to request information about the state of the simulation.

As the user adds locales and actions, the Plan Manager updates the Science Goals accordingly. When the user requests feedback, the Plan Manager calls the appropriate method in the Goal Validator. Figures 8.2 and 8.3 illustrate this procedure for checking an action and a locale, respectively. The Goal Validator relies on the Robot Manager to provide it with information about the robot's capabilities. If a locale or entire plan is being checked, the Robot Manager may request information from the robot simulation via the Command Translator. The Command Translator communicates the request to the Simulation Manager, which delegates to the Simulation Controller. After simulation has finished, the needed information is passed back from the

---

[1] *gazebo* is available at `playerstage.sourceforge.net`.

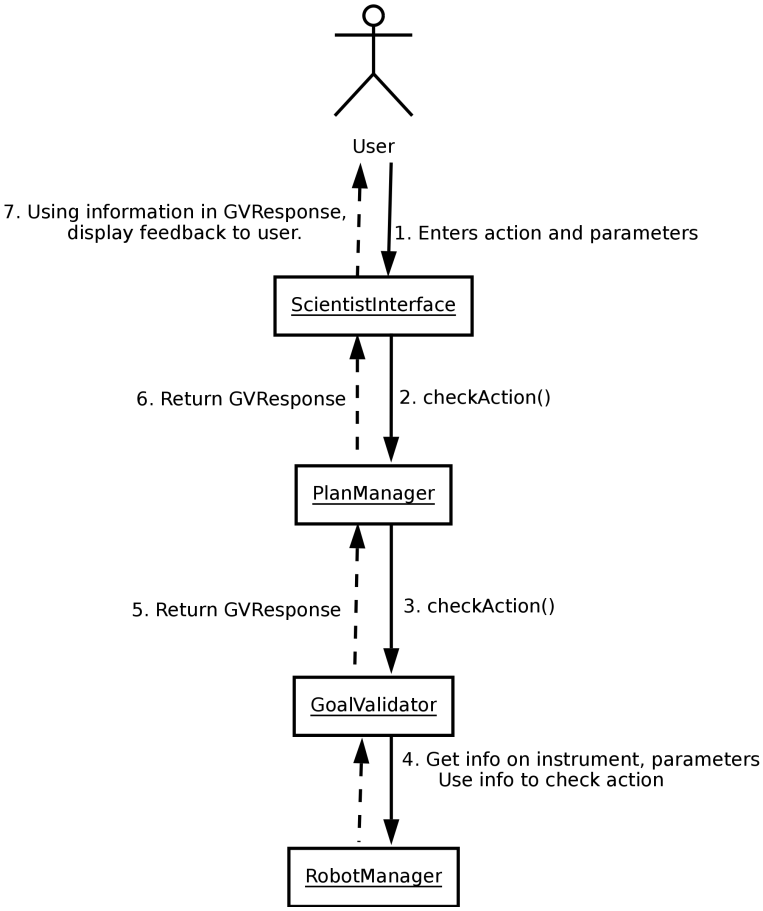

User
7. Using information in GVResponse, display feedback to user.
1. Enters action and parameters
ScientistInterface
6. Return GVResponse
2. checkAction()
PlanManager
5. Return GVResponse
3. checkAction()
GoalValidator
4. Get info on instrument, parameters Use info to check action
RobotManager

Figure 8.2: A sequence diagram illustrating the overall process of checking an action.

Simulation Controller to the Simulation Manager to the Command Translator, from the Command Translator back to the Robot Manager, and then back to the Goal Validator. The Goal Validator uses this information to complete the validation process, returning feedback which is then displayed to the user by the Scientist Interface.

All of the classes shown in Figure 8.1 are written in the Python programming language except for the Simulator classes. Because *gazebo* does not provide a Python interface, the Simulator classes are written in C++

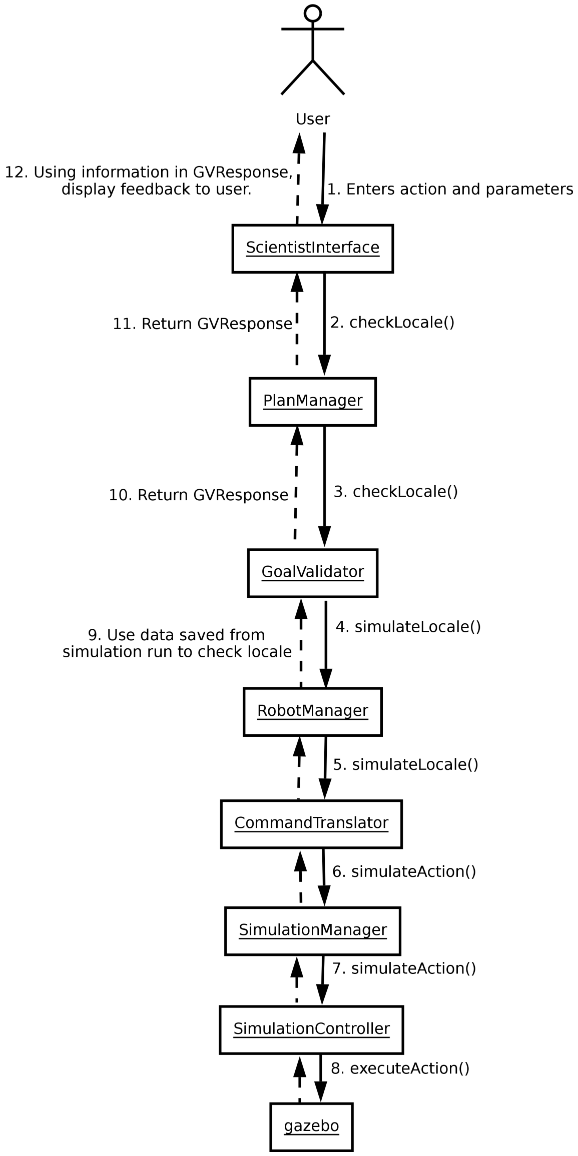

User
12. Using information in GVResponse, display feedback to user.
1. Enters action and parameters
ScientistInterface
11. Return GVResponse
2. checkLocale()
PlanManager
10. Return GVResponse
3. checkLocale()
GoalValidator
9. Use data saved from simulation run to check locale
4. simulateLocale()
RobotManager
5. simulateLocale()
CommandTranslator
6. simulateAction()
SimulationManager
7. simulateAction()
SimulationController
8. executeAction()
gazebo

Figure 8.3: A sequence diagram illustrating the overall process of checking a locale. Checking a plan follows essentially the same sequence.

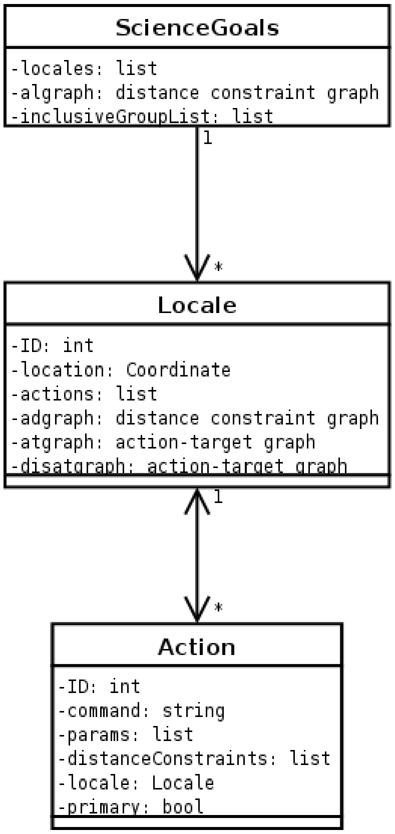

ScienceGoals
-locales: list
-algraph: distance constraint graph
-inclusiveGroupList: list
1
*
Locale
-ID: int
-location: Coordinate
-actions: list
-adgraph: distance constraint graph
-atgraph: action-target graph
-disatgraph: action-target graph
1
*
Action
-ID: int
-command: string
-params: list
-distanceConstraints: list
-locale: Locale
-primary: bool

Figure 8.4: A Unified Modeling Language (UML) class diagram of the three major components of the goal representation [Booch et al., 1999].

and then *thrift*[2] is used to pass data between the Command Translator and the Simulation Manager.

8.2 Goal Representation Implementation

The science goal representation has been implemented in Python using an object-oriented design. The key components of the implementation are shown in Figure 8.4. As this diagram illustrates, one Science Goals object may be associated with any number of Locale objects, and each Locale object may be associated with any number of Action objects.

---

[2] *thrift* is available at `http://developers.facebook.com/thrift`.

8.2.1 Action

An Action object contains the following:

- ID, a unique integer identification number
- command, a string representing the name of the action
- params, a list of the parameter values necessary to execute the command
- distanceConstraints, a list containing the distance constraints associated with this action
- locale, the Locale at which this Action should be executed
- primary, a Boolean representing whether or not this action is the action which defines a particular target

The Action class also defines methods for getting the ID value, getting and setting the command, parameters, locale, distance constraints, and primary value.

8.2.2 Locale

A Locale object contains the following:

- ID, a unique integer identification number
- location, set of coordinates representing where this Locale is in the world
- actions, an ordered list of the Actions to be executed at this Locale
- adgraph, a graph of the distance constraints which exist between actions at this Locale
- atgraph, a list of which actions are associated with which targets at this Locale
- disatgraph, a list of which actions have been explicitly dissociated with targets at this Locale

The Locale class also defines methods for getting the ID value, adding and removing Actions, adding and removing distance constraints, and associating or dissociating actions with targets.

8.2.3 Science Goals

A Science Goals object contains the following:

- locales, an ordered list of the Locales in this plan
- algraph, a graph representing distance constraints between Locales in this plan
- inclusiveGroupList, a list of the inclusive groups which have been created

The Science Goals class also defines methods for getting the list of Locales, adding and removing Locales, adding and removing distance constraints, and adding and removing items from inclusive groups.

A Science Goals object contains all of the information needed to execute a plan (which commands to execute where with what parameter values) as well as additional information, such as distance constraints and targets. This additional information is used to provide feedback to the user; it could eventually provided to the robot such that the robot may be able to make better decisions about what to do in the event an action fails.

8.3 Robot Model Implementation

The information necessary for the Robot Model is represented through both a Robot Manager class as well as a set of configuration files; one of these files is read by the Robot Manager class, and the other is read by the robot simulator *gazebo*.

8.3.1 Robot Manager

The Robot Manager class contains a set of private variables which encode many of the properties about a particular robot:

- instrumentList, a list of the instruments on-board the robot
- instrumentStatus, a dictionary structure which maps from an instrument name to a string indicating whether the instrument is fully functional, partially functional, or nonfunctional
- commandToInstrument, a dictionary structure which maps from a command name to a list of the instruments utilized by that command

- commandToParameters, a dictionary structure which maps from a command name to Parameter objects (which keep track of the name, type, and possible values for each parameter needed by the command)
- commandCanDefineTarget, a dictionary structure which maps from a command name to a boolean indicating whether or not the command can be the primary action when constructing a target
- commandCanCoverTarget, a dictionary structure which maps from a command name to a Boolean indicating whether or not the command can cover an existing target
- canGenerateData, a dictionary structure which maps from a command name to a Boolean indicating whether or not the command will create a data product to be returned to the user
- instrumentToIfaces, a dictionary structure which maps from a command name to the names of the interfaces used by that command in *gazebo*
- commandTranslator, an instance of the Command Translator class which is responsible for communicating with *gazebo*

8.3.2 Configuration Files

Most of the variables described above are initialized using a configuration file, which makes it easy to reconfigure or change robots. In addition, the Robot Manager is associated with an XML file called a world file. The world file is loaded by the *gazebo* simulator. This file is used to define the layout of the robot's environment, the physical model of the robot itself, and the instruments which are attached to the robot. Any *gazebo*-compatible world file can be used; any valid robot model placed in the world file is thus compatible with the robot proxy implementation. *gazebo* also provides a basic environmental model and is able to model the interaction between the robot's wheels and its environment. For use with a real robot, this model may need to be modified and updated according to the terrain in which the robot is operating.

Fields of view for the fluorescence imager and spectrometer used in the final evaluation of the system were calculated based on images taken by those instruments in *gazebo* and then cached (to prevent recalculation every time the instruments are used).

8.4 Goal Validation Implementation

The role of the Goal Validator is to ensure that a particular action, locale, or entire plan is consistent with the Science Goals and with the robot's capabilities. The principal methods in the Goal Validator which are used for this are 'checkAction', 'checkLocale', and 'checkPlan', respectively. Each of these is outlined in detail in the following sections.

8.4.1 Check Action

The Check Action method (Algorithm 2) is called every time the user enters a parameter for an action or attempts to save an action. The method checks that the Locale has been correctly associated with the Action, that all instruments needed by the action are functional, and that all parameters provided for the action are acceptable given the instruments' current capabilities.

Algorithm 2 Check Action method in the Goal Validator.

```
Require: Locale $l$, Action $a$
  {Check 1: Locale correctly assigned}
  if $a$.getLocale() $\neq l$ then
    return Exception
  {Check 2: Action must use a functional instrument}
  if any instrument in robotManager.getInstrumentByAction($a$) is not functional then
    return NonfunctionalInstrument
  {Check 3: All parameters must be acceptable given the current state of the instrument}
  for each parameter in $a$.getParameters() do
    if robotManager.checkParam($a$,paramter) == False then
      return InvalidParameter
  return positive GVResponse
```

8.4.2 Check Locale

The Check Locale method (Algorithm 3) updates the list of targets associated with the locale, checks that every distance constraint between actions at the locale is associated both with a distance and an accuracy value, checks to see if one or more actions associated with a target does not correctly cover that target, and checks to see that all distance constraints at the locale are satisfied.

Algorithm 3 Check Locale method in the Goal Validator.

```
Require: Locale $l$, Action $a$
  robotManager.simulateLocale($l$) {Simulate the actions at this locale}
  updateTargets($l$) {Update the associations between actions and targets at this locale}
  adgraph ⇐ $l$.getADGraph() {Get the distance constraints between actions at this locale}
  for each distance constraint in adgraph do
    if the distance constraint has no accuracy value then
      return MissingAccuracy
  targetList ⇐ all targets at this locale
  for each target in targetList do
    for each action associated with the target do
      if isGoodTarget(action) == False then
        badActionList.append(action)
    if length of badActionList $> 0$ then
      return BadTargeting
  adgraph ⇐ $l$.getADGraph() {Get the distance constraints between actions at this locale}
  for each distance constraint in adgraph do
    {Use the simulator to get the actual distance between the actions}
    if distance between the actions does not meet the distance constraint then
      return BadDistance
  return positive GVResponse
```

A target is defined by a set of coordinates on the ground plane which are the focus of a scientific investigation (that is, the user desires for multiple instrument readings to cover the same area). In the Robot Manager configuration file, every instrument on the robot which can take data must be specified as either being able to **define** a target, **cover** a target, or neither.

If an action defines a target, that means the target's coordinates are equal to the coordinates of the intersection of that instrument's field of view with the ground plane. If the instrument's field of view moves (either because parameters have changed or because the robot moves), the target's coordinates update accordingly. If an action can cover a target, that means the action can be associated with a target, and so the system will check for overlaps between the instrument's field of view and any targets which have been previously defined. In this particular implementation, the robot's fluorescence imager can define a target and the robot's spectrometer can cover a target.

The method by which the Goal Validator determines what targets are at a locale is shown in 4. Essentially, the method uses two criteria to determine what targets should exist at the locale: (1) whether the user has explicitly specified that an action should (or should not) define a target or be associated with a target and (2) whether an action is sufficiently close to a target such that they should be associated with each other. The user's directions (if any) are given first priority before the system will create or assign targets automatically. Two fields of view are said to be sufficiently close if they overlap each other, or, if the fields of view were scaled by a hand-coded factor of 125%, they would overlap.[3]

The isGoodTarget method of Algorithm 3, which the Goal Validator uses to determine if an action correctly covers a target, is shown in Algorithm 5. The polygon operations are provided by the Python library *Polygon*. If the field of view of the action completely covers the target or vice versa, or if the area of the intersection is very close to the area of the field of view and the target, there is good targeting.[4] However, in the final evaluation, this algorithm was adjusted so that it would return true if the field of view and target overlapped enough to cover a rock (a 12 cm by 12 cm square).

8.4.3 Check Plan

The Check Plan method (Algorithm 6) checks that every distance constraint between locales has an accuracy value and that the distance constraints between locales are met.

8.4.4 Goal Validation Responses and Remedies

The return value of each main Goal Validation method (checkAction, checkLocale, and checkPlan) consists of a GVResponse object (Figure 8.5). The GVResponse object contains a string, *evidenceType*, which is set either to 'positive' (indicates the check was successful) or 'negative' (indicates the check failed). The *level* variable is set to either 'action', 'locale', or 'plan',

---

[3] For the final evaluation, however, the system assumed that all instruments were attempting to target a single rock; thus, the sufficientlyClose method always returned true.

[4] The value used is hard-coded to 0.005.

Algorithm 4 Update Targets method in the Goal Validator.

```
Require: Locale $l$
  atgraph ⇐ $l$.getATGraph() {The graph which contains explicit associations between actions and targets}
  disatgraph ⇐ $l$.getDisATGraph() {The graph which contains explicit dissociations between actions and targets}
  releventActions ⇐ All actions at $l$ which can define or cover a target
  for action in relevantActions do
    targets ⇐ action.getTargets()
    if action.isPrimary() == True then
      {Ensure that the action is associated with exactly one target}
  for action in relevantActions do
    if action.isPrimary() == False then
      allPrimaryTargets ⇐ atgraph.getAllPrimaryTargets()
      for primary in allPrimaries do
        if primary in targets then
          {If the action was dissociated, remove the action from the target}
        else
          if action's field of view is not sufficiently close to the target then
            return BadTargeting
      for primary in allPrimaries do
        if primary not in targets then
          if primary not in dissociated then
            {If the action is sufficiently close to this primary target, associate them}
  if action can define a target and it hasn't already been associated then
    if action is not associated with any targets then
      {Make this action a primary action and create a new target}
```

to clarify whether the GVResponse object was generated inside the checkAction, checkLocale, or checkPlan method, respectively.

Each GVResponse object contains all relevant contextual information such that the interface can display an appropriate message. It also contains a list of GVRepair objects which indicate appropriate repairs to the Science Goals.

As shown in Figure 8.5, the five types of GVResponses are:

- InvalidParameter: An Action has an unacceptable parameter value. The object contains a list of which instruments are used by the action,

Algorithm 5 isGoodTarget method in the Goal Validator.

```
Require: Coordinates of action's field of view $fov$, target $target$
  $fovPoly \Leftarrow$ polygon representation of $fov$
  $targetPoly \Leftarrow$ polygon representation of $target$
  $andPoly \Leftarrow fovPoly \& targetPoly$ {Compute polygon which represents the overlap between the field of view and the target}
  if $fovPoly$.covers($targetPoly$) or $targetPoly$.covers($fovPoly$) then
    return True
  else
    $fovArea \Leftarrow$ area of $fovPoly$
    $targetArea \Leftarrow$ area of $targetPoly$
    $andArea \Leftarrow$ area of $andPoly$
    {There is good targeting if there is virtually no difference between the area of the field of view, the area of the target, and the area of their intersection.}
    if (abs($andArea - fovArea$) $< 0.005 * fovArea$) and (abs($andArea - targetArea$) $< 0.005 * targetArea$) then
      return True
    else
      return False
```

Algorithm 6 Check Plan method in the Goal Validator.

```
Require: ScienceGoals $scienceGoals$
  algraph $\Leftarrow l$.getALGraph() {Get the distance constraints between locales}
  for each distance constraint in algraph do
    if the distance constraint has no accuracy value then
      return MissingAccuracy
  for each distance constraint in algraph do
    $givenDistance \Leftarrow$ what the distance between the locales should be according to the distance constraint
    $accuracy \Leftarrow$ accuracy value according to the distance constraint
    $distance \Leftarrow$ straight-line distance between the two locales
    $lowerBound = givenDistance - accuracy$
    $lowerBound = givenDistance - accuracy$
    if ($actualDistance < lowerBound$) or ($actualDistance > upperBound$) then
      return BadDistance
  return positive GVResponse
```

the command associated with the action, the name of the parameter with the unacceptable value, a boolean indicating whether the action is currently using modified values (indicating an instrument is not fully functional), and a list of the currently acceptable values for the parameter.

- MissingAccuracy: A distance constraint (the relevantObject) is missing an accuracy value as well as the list of associated actions (the relevantActionList).
- NonfunctionalInstrument: An Action has been selected which relies on an instrument which is currently not functioning. The object contains the name of the instrument and the command associated with the action.
- BadDistance: Two actions or locales are separated by a distance which is not acceptable according to their distance constraint. The object contains the relevant actions, locales, and distance constraint information.
- BadTargeting: An action is associated with a target, but the field of view of the action does not sufficiently cover the target. The object contains the relevent target, a list of actions which do not sufficiently cover it, and a map between each relevant action and its field of view.

Figure 8.6 illustrates the seven possible remedies to checks which fail. Each GVRepair object is associated with a type, indicating whether the repair involves editing a value, deleting a value, or getting a new value from the user.

- EditParameter: change the value of an Action's parameter
- EditActions: edit the list of actions at a locale
- DeleteAction: delete an action
- EditTargets: edit which actions are associated/dissociated with which targets
- EditDistanceConstraint: edit the distance and/or accuracy values of a distance constraint
- DeleteDistanceConstraint: delete a distance constraint
- GetAccuracyValue: get a missing accuracy value from the user

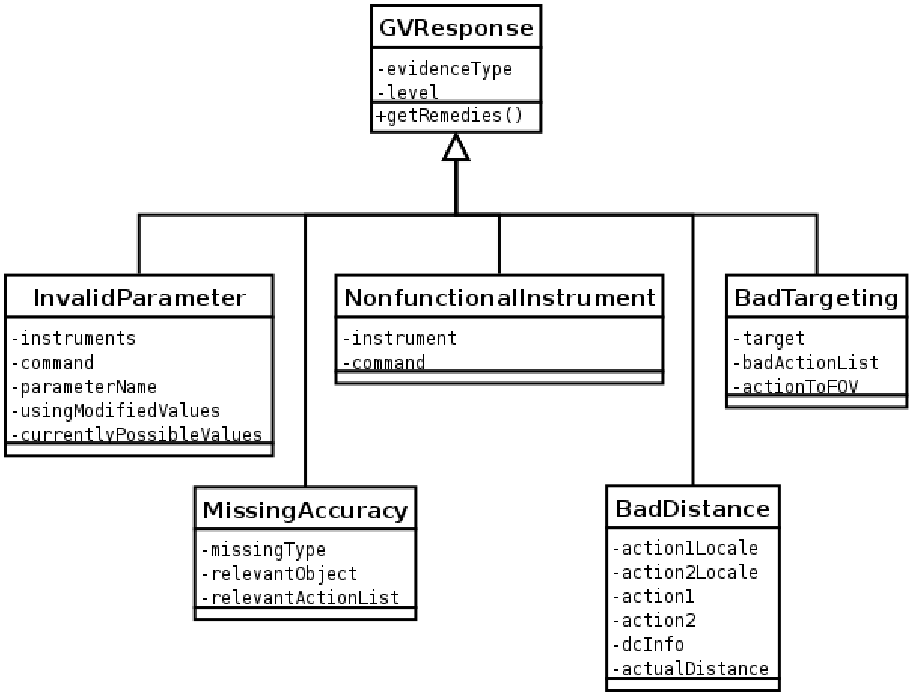

GVResponse
-evidenceType
-level
+getRemedies()
InvalidParameter
-instruments
-command
-parameterName
-usingModifiedValues
-currentlyPossibleValues
NonfunctionalInstrument
-instrument
-command
BadTargeting
-target
-badActionList
-actionToFOV
MissingAccuracy
-missingType
-relevantObject
-relevantActionList
BadDistance
-action1Locale
-action2Locale
-action1
-action2
-dcInfo
-actualDistance

Figure 8.5: A Unified Modeling Language inheritance diagram illustrating the possible responses from the Goal Validator [Booch et al., 1999].

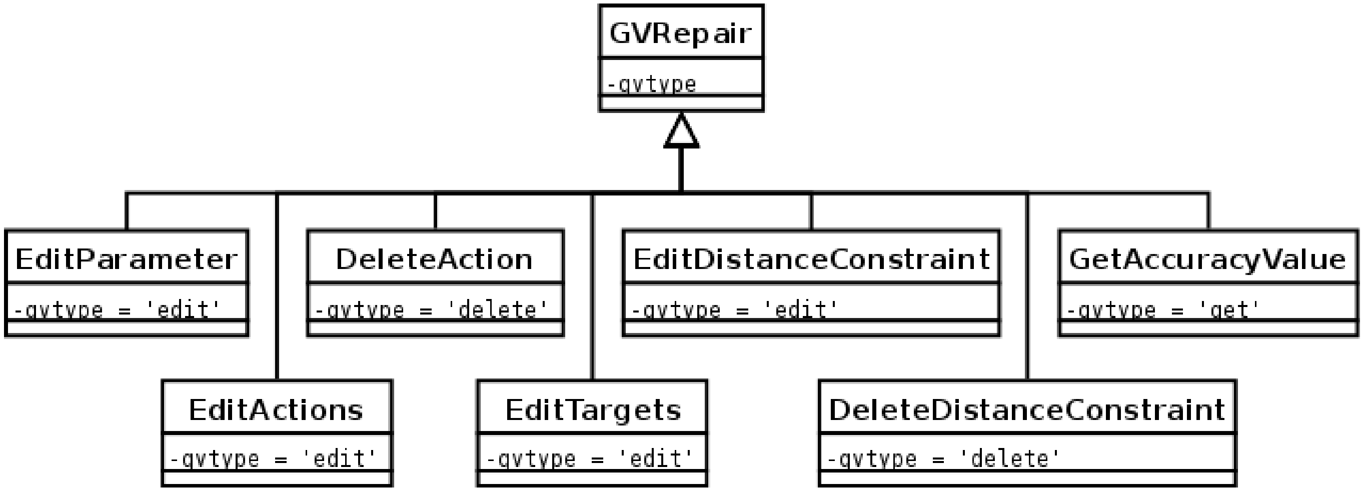

GVRepair
-gvtype
EditParameter
-gvtype = 'edit'
DeleteAction
-gvtype = 'delete'
EditDistanceConstraint
-gvtype = 'edit'
GetAccuracyValue
-gvtype = 'get'
EditActions
-gvtype = 'edit'
EditTargets
-gvtype = 'edit'
DeleteDistanceConstraint
-gvtype = 'delete'

Figure 8.6: Unified Modeling Language inheritance diagram illustrating the possible options to repair a failed goal validation check [Booch et al., 1999].

8.5 Summary

As indicated by the analysis of the Life in the Atacama project, the robot proxy requires three main components: a goal representation, which captures not only a sequence of actions but also additional information relevant to targets and other constraints; a robot model, which contains information about the robot's capabilities; and a goal validator, which allows the user to check an action, locale, or plan to determine if the science goals are internally consistent and consistent with the robot model. These components have been implemented in Python according to an object-oriented model, represented by the Science Goals, Robot Model, and Goal Validator classes, respectively.

Chapter 9

Evaluation Study Design

This chapter describes an extension to the proof-of-concept study described in Chapter 6 in which participants utilized the full robot proxy implementation in order to conduct an exploration robotics task. The most significant findings of this study indicated that participants who utilized the robot proxy performed more efficiently, collected higher-quality data, and established greater common ground with the robot. This study suggests Robot-Proxy Grounding can improve efficiency in exploration robotics tasks.

9.1 Study Design and Method

In this experiment, a robot proxy-based interface was compared against an encoder-decoder interface that could only pass plans from the user to the robot. As in the earlier, proof-of-concept study, a between-subjects design was used: each participant was randomly assigned to one of two conditions, the Robot Proxy condition or the No Proxy condition. In order to maximize the number of participants, a simulation was used in lieu of a physical robot.

The goals of the study were to understand the impact of a robot proxy-based interface on three particular areas relevant to common ground and exploration robotics tasks:

- *Task performance.* Which group is more efficient at completing the task successfully? Which group captures higher-quality data?
- *Self-evaluation of performance.* How does the robot proxy-based interface affect participants' perceptions of their own performance and their feelings of collaboration with the system?

- *Establishing common ground.* Which group has a better understanding of the robot's state and its context?

As in the proof-of-concept study, investigation of task efficiency focuses primarily on how many communication cycles are required to complete the task rather than the amount of time spent by the user to create plans for the robot. The number of communication cycles is a better metric because in remote exploration HRI, communication with the robot is very expensive and limited.

9.2 Participants

Sixteen participants were recruited from Carnegie Mellon University; eight were assigned to each of the two conditions. All participants were graduate students in the Robotics Institute. Participants were compensated US$50 for their time upon completion of the study.

9.3 Procedure

After arriving at the lab, each participant was seated at a desktop computer. The participant was given three handouts to read: a handout containing background information on the task, a handout containing diagrams of the robot used in the experiment (Figure 9.1), and a handout describing the commands and parameters that the participant could use during the task (Figure 9.2). The background information handout contained the following description of the task:

> In this task, you will work with a Pioneer 2DX robot which is located in the Atacama Desert. Scattered around the desert are rocks, each of which may host a variety of living organisms. You must use the robot's imaging abilities to take pictures of each rock and figure out what kinds of life live on the rock and what the distribution of each type of organism is. You will study one rock at each of three different sites (locales). At the very end of the task, you will be asked to complete a short questionnaire.
>
> The robot has the following capabilities:
>
> - It can drive in a straight line forwards or backwards (DRIVE)
> - It can turn in place (TURN)

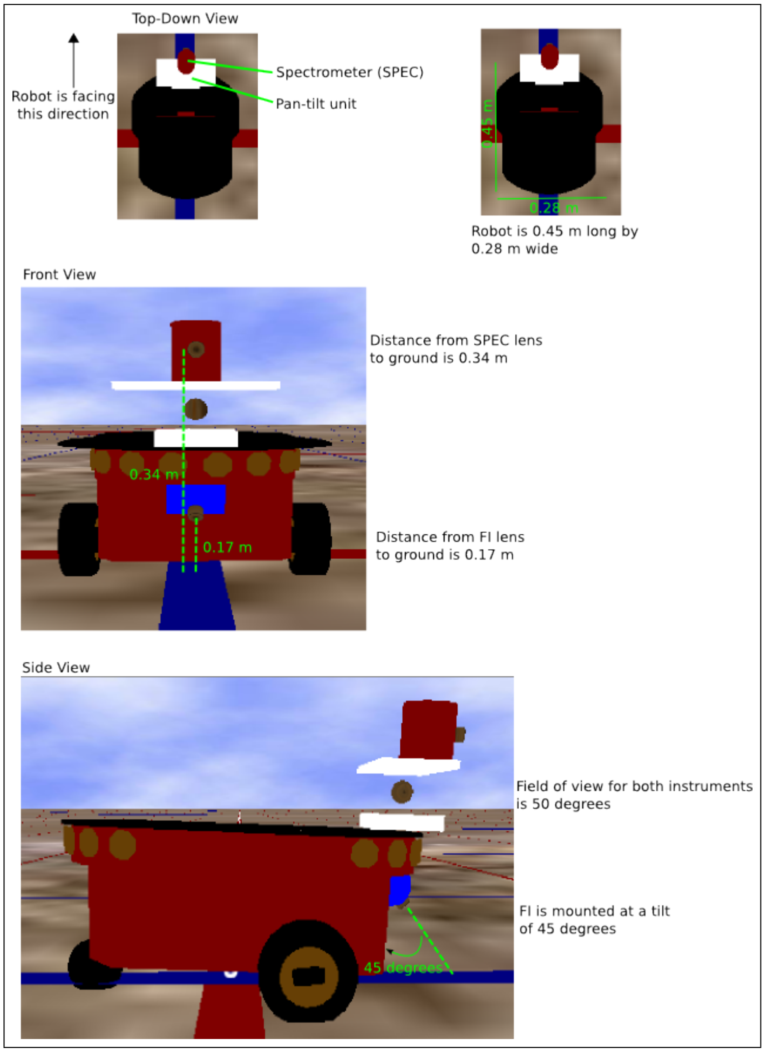

Top-Down View
Robot is facing this direction
Spectrometer (SPEC)
Pan-tilt unit
0.45 m
0.28 m
Robot is 0.45 m long by 0.28 m wide
Front View
Distance from SPEC lens to ground is 0.34 m
0.34 m
0.17 m
Distance from FI lens to ground is 0.17 m
Side View
Field of view for both instruments is 50 degrees
FI is mounted at a tilt of 45 degrees
45 degrees

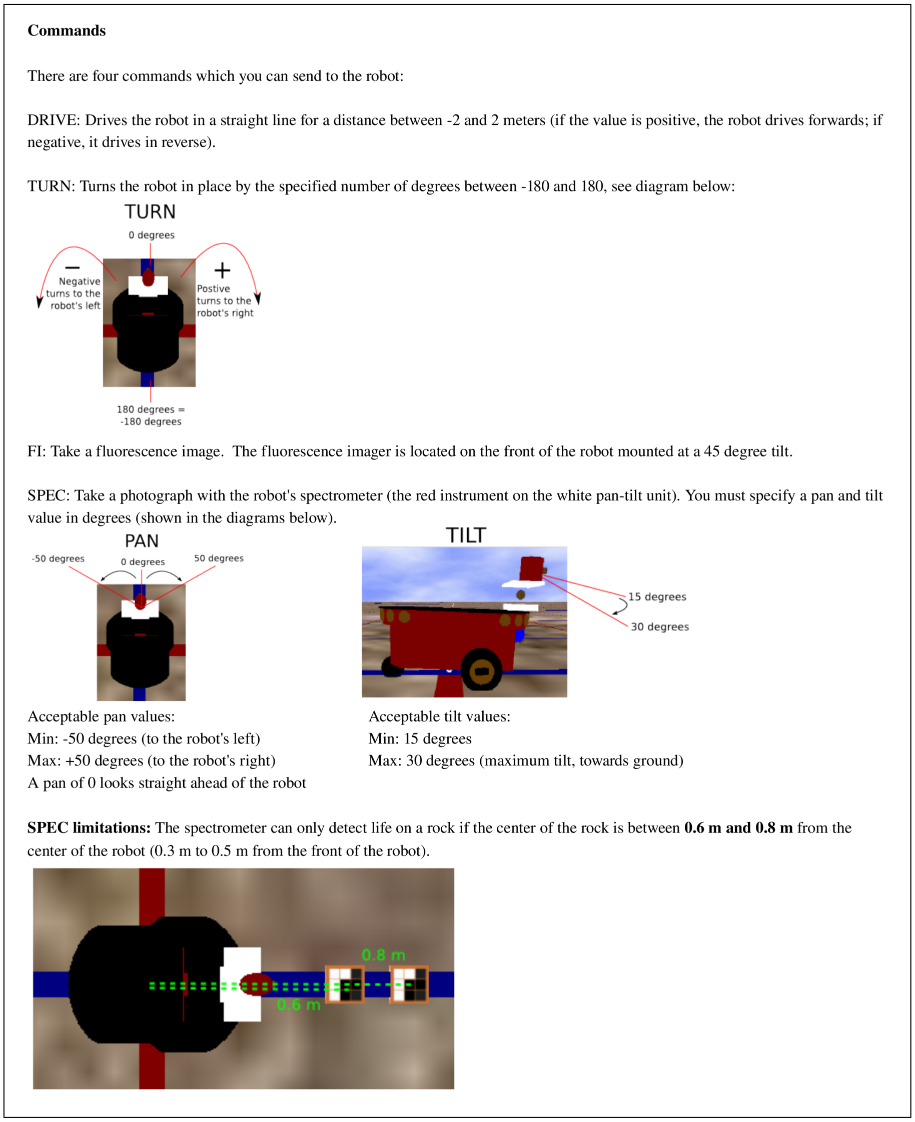
Figure 9.1: Handout that was presented to study participants on the dimensions of the robot.

- It can take an image using its fluorescence imager (FI)
- It can take an image using a spectrometer (SPEC)

The spectrometer is mounted on the robot's white pan-tilt unit.

The fluorescence imager is mounted on the front of the robot.

You will assemble sets of actions (called *plans*) that the Pioneer



Commands

There are four commands which you can send to the robot:

DRIVE: Drives the robot in a straight line for a distance between -2 and 2 meters (if the value is positive, the robot drives forwards; if negative, it drives in reverse).

TURN: Turns the robot in place by the specified number of degrees between -180 and 180, see diagram below:

TURN
0 degrees
−
Negative turns to the robot's left
+
Postive turns to the robot's right
180 degrees = -180 degrees

FI: Take a fluorescence image. The fluorescence imager is located on the front of the robot mounted at a 45 degree tilt.

SPEC: Take a photograph with the robot's spectrometer (the red instrument on the white pan-tilt unit). You must specify a pan and tilt value in degrees (shown in the diagrams below).

PAN
-50 degrees
0 degrees
50 degrees
TILT
15 degrees
30 degrees

Acceptable pan values:
Min: -50 degrees (to the robot's left)
Max: +50 degrees (to the robot's right)
A pan of 0 looks straight ahead of the robot

Acceptable tilt values:
Min: 15 degrees
Max: 30 degrees (maximum tilt, towards ground)

SPEC limitations: The spectrometer can only detect life on a rock if the center of the rock is between **0.6 m and 0.8 m** from the center of the robot (0.3 m to 0.5 m from the front of the robot).

Figure 9.2: Handout that was presented to study participants on the commands which could be sent to the robot.

will execute in order to collect data. In general, your exploration will have two phases:

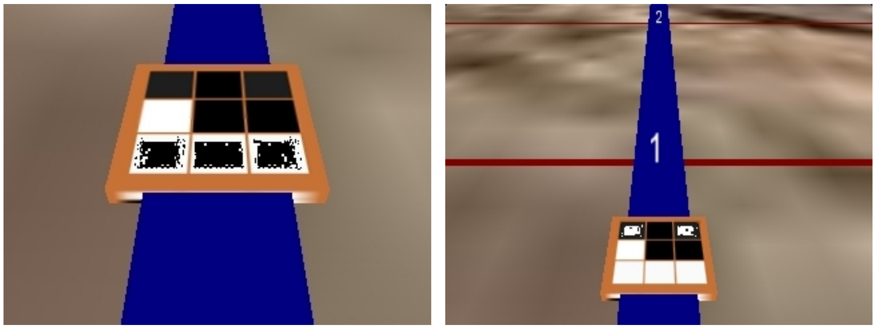
First, you will be asked to add one new locale where you want the robot to explore. This locale must be 100 meters away from

> the robot's current position (+/- 5 meters), located with a region of geologic interest. After you execute this plan, the robot will move to the new locale, automatically take a panoramic image with its camera, and return the image. **Every time you move the robot to a new locale, it must be 100 m +/- 5 m away from the robot's previous locale and in a new region.**
>
> Second, using the overhead image taken at the locale, you will add actions to the robot's current locale so that it will take images of the rock. It is very costly in terms of bandwidth to communicate with the robot, so you should try to **minimize the number of execution cycles** needed to study each rock. Every action also drains the robot's batteries, so you should try to **minimize the number of actions** in each plan. You are allowed a maximum of three execution cycles at each locale before the robot will have to move on to a new locale.

The participant was also provided with a specific list of robot commands available to use as well as diagrams depicting the robot's shape and size, which were available to the participant throughout the game (Figures 9.1 and 9.2).

The experimenter then presented a twenty-minute tutorial describing how to use the interface to complete each of the two phases described above. The experimenter then read through an additional handout describing how to interpret the images taken by the robot (example images shown in Figure 9.3).

The participant was presented with an orbital map which displayed the location of the robot as well as five areas of interest (Figure 9.4). In Phase 1, the participant was required to choose a new location (locale) for the robot to visit that was inside one of these areas as well as being 100 meters from the robot's starting location (plus or minus five meters). Participants could use the mouse to place a new locale on the orbital map. Additionally, participants in the Robot Proxy group had the option of adding a distance constraint specifying that the locations should be 100 meters plus or minus five meters apart. Participants in the Robot Proxy group could then use the "Check Plan" button to receive feedback about their locale placement (Figure 9.6(b)). Once the participant was satisfied with the locale placement, she clicked the "Execute Plan" button (Figure 9.6). Before the robot executed the plan, the participant was asked for an estimate of how far apart the two locales were and how confident she was in that estimate.



(a) An example FI image. Microbes detected by the FI appear as black on a white background.

(b) An example SPEC image. Microbes detected by the SPEC appear as white on a black background.

Figure 9.3: Sample images taken by the robot using (a) the fluorescence imager (FI) and (b) the spectrometer (SPEC).

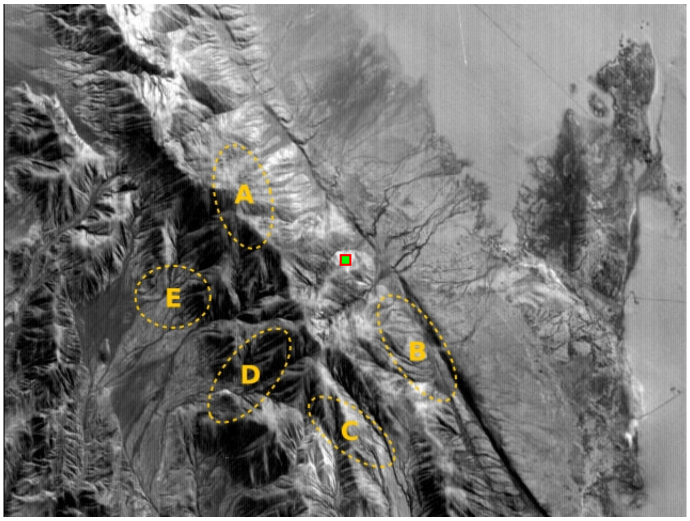


Figure 9.4: The orbital map presented to study participants showing the location of the robot (square) and five areas of interest labeled A through E.

The robot then executed the plan and moved to the new location. The interface displayed an overhead map containing the robot and the rock (see Figure 9.5). This was the end of Phase 1 for the locale.

The participant then used the "Edit Locale" button to add actions to the rover's current location such that it would approach the rock and image it

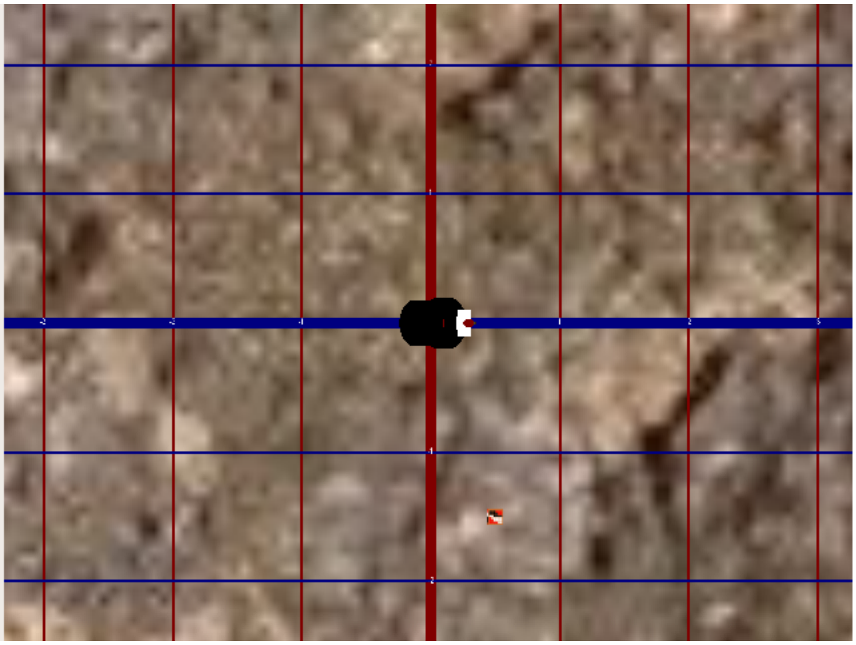
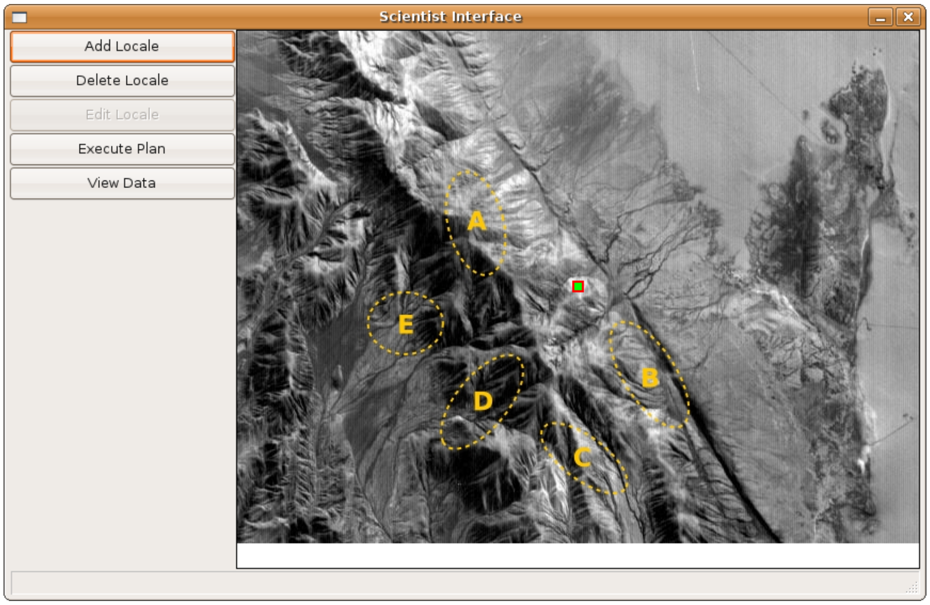


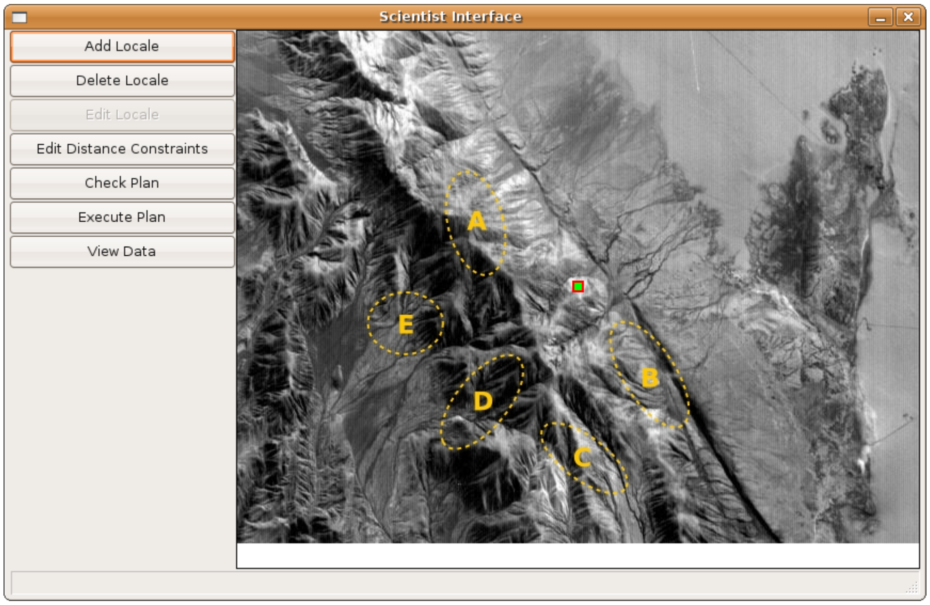
Figure 9.5: An example overhead map showing the location of the robot and the rock.

with both the fluorescence imager (FI) and the spectrometer (SPEC) (Figure 9.7). Participants in the Robot Proxy group had the option of requesting feedback about their plans (Figure 9.7(b)). This feedback consisted of a message indicating if the instrument readings were not overlapping enough to cover a rock as well as a graph depicting where the instruments' fields of view would cover the ground (Figure 9.8). Once participants were satisfied with their plan, they would click the "Execute Plan" button. Before the plan was executed, participants were asked to estimate a) which instrument readings would overlap with each other, if any, b) how far they estimated the center of the robot would be from the center of the rock after the plan was executed, and c) how much of the rock would be covered by each instrument reading ("All", "Some", or "None"). Participants were also asked to provide confidence values for each estimate on a Likert scale from 1 to 5 (1 = "Not confident", 5 = "Very confident")

The robot would then execute the plan and the interface would display an updated overhead view as well as any images taken by the robot during the execution of the plan. Participants had the option of executing up to three plans at each locale before they were required to move to the next

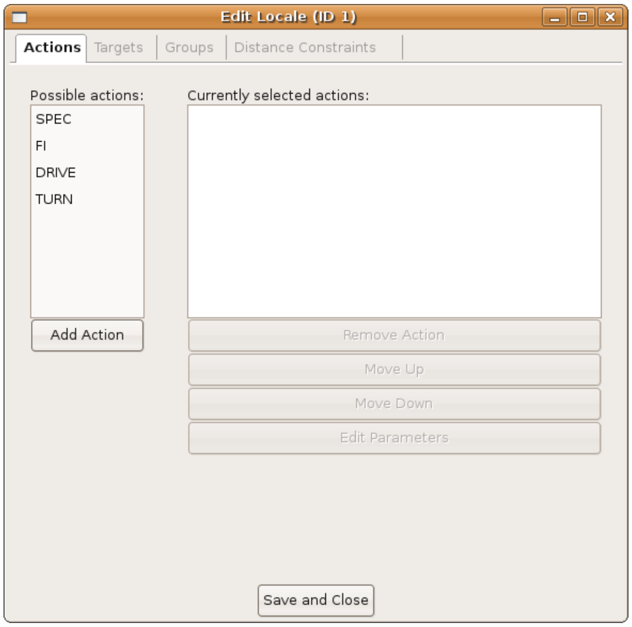

Scientist Interface
Add Locale
Delete Locale
Edit Locale
Execute Plan
View Data
A
B
C
D
E

(a) Scientist Interface for participants in the No Proxy group.

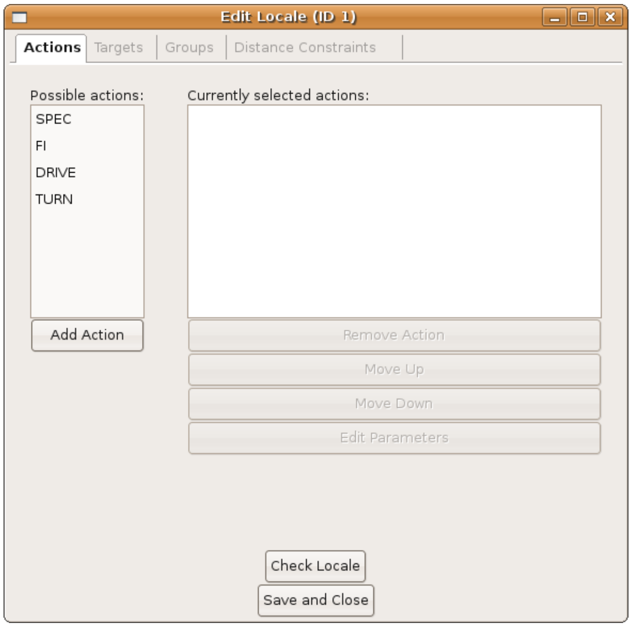

Scientist Interface
Add Locale
Delete Locale
Edit Locale
Edit Distance Constraints
Check Plan
Execute Plan
View Data
A
B
C
D
E

(b) Scientist Interface for participants in the Robot Proxy group, which includes buttons for entering distance constraints and checking the plan.

Figure 9.6: The main interface display for participants in (a) the No Proxy group and (b) the Robot Proxy group.

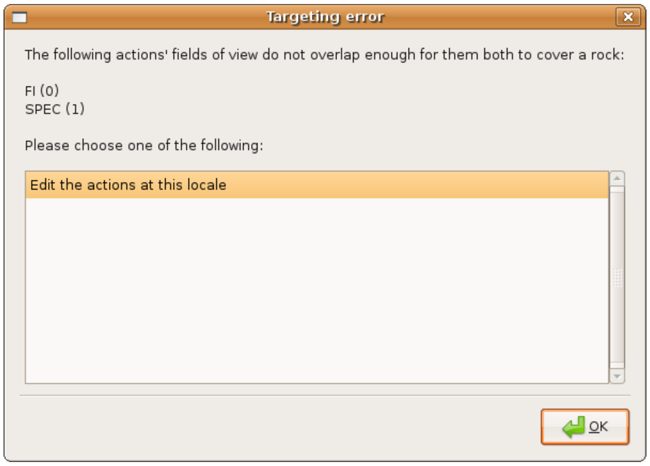

Edit Locale (ID 1)
Actions
Targets
Groups
Distance Constraints
Possible actions:
Currently selected actions:
SPEC
FI
DRIVE
TURN
Add Action
Remove Action
Move Up
Move Down
Edit Parameters
Save and Close

(a) Edit Locale Interface for participants in the No Proxy group.


Edit Locale (ID 1)
Actions
Targets
Groups
Distance Constraints
Possible actions:
Currently selected actions:
SPEC
FI
DRIVE
TURN
Add Action
Remove Action
Move Up
Move Down
Edit Parameters
Check Locale
Save and Close

(b) Edit Locale for participants in the Robot Proxy group, which includes a button for checking the actions at the locale.

Figure 9.7: The interface used to add actions to a locale for participants in (a) the No Proxy group and (b) the Robot Proxy group.


Targeting error
OK

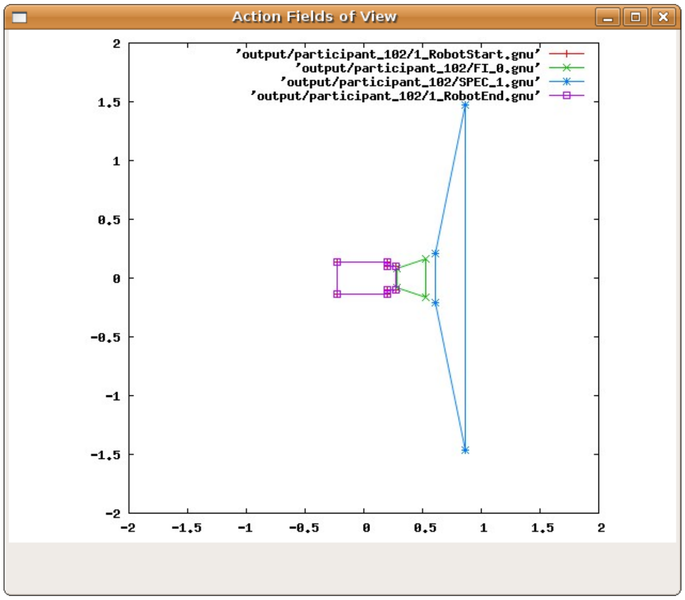
The following actions' fields of view do not overlap enough for them both to cover a rock:

FI (0)
SPEC (1)

Please choose one of the following:

Edit the actions at this locale

(a) Written feedback provided to participants indicating which actions do not overlap enough to cover a rock.


Action Fields of View
'output/participant_102/1_RobotStart.gnu'
'output/participant_102/FI_0.gnu'
'output/participant_102/SPEC_1.gnu'
'output/participant_102/1_RobotEnd.gnu'
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2

(b) Graphical feedback provided to participants which displays the robot's starting and ending locations as well as the area on the ground which will be covered by each instrument reading.

Figure 9.8: The two screens of feedback provided after the "Check Locale" button is clicked: (a) a description of which actions do not overlap and (b) a graph which displays the starting position of the robot, ending position of the robot, and the fields of view of all instrument readings.

locale.

Thus, during each of three trials, each participant completed two major activities: selecting a locale for the robot to visit (Phase 1) and studying a rock at that location (Phase 2). No matter where participants chose to send the robot during Phase 1, the location of the rock with respect to the robot was the same for all participants in Phase 2. During this second phase, the participants constructed a plan for the robot (planning), executed the plan, and examined the images returned from the robot (data review). A *cycle* is defined as one planning session followed by one data review session. Each participant examined three different rocks for a total of three trials. After completing all three trials, the participant was asked a set of questions about her experiences. The entire process lasted approximately sixty to ninety minutes per participant.

9.4 Simulation

Because the task involved a non-collocated robot, software could be used to simulate a physical robot without sacrificing the fidelity of the human-robot interaction. The robot simulation software *gazebo* was used to simulate the robot's actions and to generate the data returned from the robot. At the end of the experiment, the participant was informed that he/she had been using a simulated robot. The robot was not stochastic except for the fact that it tended to under turn (although it was consistent in doing so); thus, the robot executed plans consistently, if not perfectly. This is an advantage over the proof-of-concept study in that the simulated robot the robot behaved more like a physical robot.

9.5 Dependent Variables

Table 9.1 illustrates the dependent variables measured in this study. The performance and common ground variables were derived from the requirements of the task itself. For all self-evaluation questions, participants were given a Likert scale from 1 to 5 and asked how strongly they agreed with a particular statement (1 = "Strongly disagree", 5 = "Strongly agree"). Factor analysis was used to confirm that both statements on collaboration could be combined as a coherent factor "Collaboration"; the Cronbach's alpha of this factor was calculated to be 0.86, which suggests that the factor is internally consistent. For questions which asked about participants' confidence in an answer or estimation, participants were given a Likert scale from 1 to 5 and asked how confident they felt (1 = “Not confident”, 5 = “Very confident”). For the two questions which asked participants to rate how responsible they were for successful plans (those in which the entire rock was visible) and failed plans (those in which the entire rock was not visible), participants assigned responsibility by choosing one of the following: “I was 0% responsible, the system was 100% responsible”, “I was 12% responsible, the system was 88% responsible”,“I was 33% responsible, the system was 66% responsible”, “I was 50% responsible, the system was 50% responsible”,“ I was 66% responsible, the system was 33% responsible”, “I was 88% responsible, the system was 12% responsible”, “I was 100% responsible, the system was 0% responsible”, or “N/A”.

Table 9.1: Dependent Variables

Variable	**Measure**
Task Performance: Phase 1 (Locale Placement)	
Accuracy	How close was the distance between each pair of locales to 100 meters?
Task Performance: Phase 2 (Rock Examination)	
Accuracy	Did the participant successfully determine the distribution of each type of life on each rock?
# Cycles	How many planning/execution cycles were used in examining the rock?
Data Quality	For each piece of data returned from the robot: A) What proportion of the rock face was covered by the instrument? B) What proportion of pixels in the image contained the rock face?
Review-Data Ratio	What proportion of the participant's time spent in this phase was used to review data from the robot?
Self-Evaluation of Performance	
Effectiveness	To what extent did the participant agree or disagree that she was efficient at performing the task and felt confident during the task? (4 questions)
Collaboration	To what extent did the participant agree or disagree with the statements: A) When developing plans, I felt I was collaborating with the system. B) I relied on the system for help.
Responsibility for Successes/Failures	How do participants assign responsibility for their successes/failures between themselves and the system?
Establishing Common Ground	
Understanding of the robot's state	Immediately after each plan is executed but before the data is displayed to the participant, the participant is asked whether each pair of instrument readings overlap and how confident she is in those answers.
Understanding of the robot's context	A) Immediately after each plan is executed but before the data is displayed to the participant, the participant is asked to estimate the distance between the robot and the rock and how confident she is in that estimate. B) Immediately after each plan is executed but before the data is displayed to the participant, for each instrument reading, the participant is asked how much of the rock (all, some, or none) will appear in the resulting image and how confident she is in those answers.

Chapter 10

Analysis of Evaluation Results

The data analysis for this study focused on understanding the differences is task performance, self-evaluation of performance, and establishing common ground between participants who used a Robot Proxy and those who did not.

10.1 Task Performance

The experimental task consisted of two phases: 1) choosing a locale for the robot to visit and 2) examining a rock. The performance of participants on both phases of the task was examined to determine if there were any differences between the Robot Proxy and No Proxy groups.

10.1.1 Phase 1: Locale Placement

During Phase 1 of each trial, the participant was required to place a new locale on the orbital map at a distance of 100 meters (plus or minus five meters) from the robot's starting position. The results showed that there was no significant difference in performance on this part of the task between the two groups. The absolute error of each participant's locale placement was calculated as the distance by which the participant's locale placement was outside of this 110-meter window. In order to examine the difference in this absolute error between participants, a two-way repeated measures analysis of variance (ANOVA) was conducted from all 16 participants with condition as a between-subjects variable and trial number as a within-subjects factor. There was no significant main effect for condition nor trial and no interaction effect. This shows that participants placed their locales with approximately the same error, indicating that there was no difference in performance. Although the Robot Proxy group received feedback about the distance between the locales that they placed while the No Proxy group received no feedback, the No Proxy participants had to determine the distance between locales by calculating it themselves. This meant that both groups had a fairly good idea of how far apart locales were, resulting in similar error rates.

10.1.2 Phase 2: Examining Rocks for Signs of Life

During Phase 2 of each trial, the participant was allowed to send up to three plans in order to capture an image of a rock using both the robot's spectrometer (SPEC) and fluorescence imager (FI). For each rock, participants were asked to identify the frequency and distribution of two types of life (each instrument could detect only one type). Five of eight participants in the No Proxy group and seven of eight participants in the Robot Proxy group correctly identified the proportion and distribution of all life; there was no significant difference between the groups. This indicates that participants were sufficiently motivated to collect enough data during the task to correctly identify life.

A two-way repeated measures ANOVA was conducted with condition as a between-subjects variable and trial number as a within-subjects factor in order to better understand any differences in participants' strategies in performing the task. Examining the number of actions executed on the robot showed no significant main effect for condition or trial and no interaction effect. The fact that participants executed similar numbers of actions suggests that participant adopted a similar strategy as they examined the rocks. Also examined was the proportion of time participants spent reviewing data from the robot as opposed to planning. The ANOVA indicated only a marginal main effect for condition ($F[1,42] = 3.19$, $p < 0.1$); participants in the Robot Proxy group spent only marginally significantly more time reviewing data than participants in the No Proxy group. This is opposite of our finding in the pilot test, perhaps because utilizing the feedback from the system required the use of the overhead map, which was included as part of the data in this study; the overhead map was not included in the data review time in the previous study.

These results suggest that participants utilized similar strategies and were ultimately able to achieve success at correctly identifying life. Mistakes

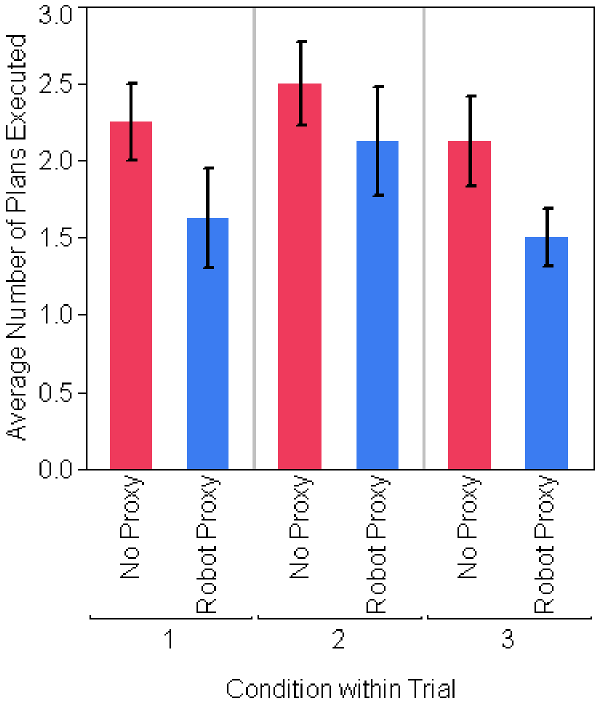


Figure 10.1: Mean and standard error of number of plans executed for both conditions across all three trials.

participants made did not lead to complete failure; rather, the data suggest that participants without the robot proxy were more inefficient at the task and lacked common ground with the robot.

Task efficiency was examined by measuring the number of plans each participant executed in Phase 2 as well as the quality of the data that the robot returned to them. Examining the number of plans executed using a two-way repeated measures ANOVA with condition as a between-subjects variable and trial number as a within-subjects factor showed a significant main effect for condition ($F[1, 42] = 5.45$, $p < 0.05$). As shown in Figure 10.1, this indicates that participants in the Robot Proxy group used significantly fewer cycles than those in the No Proxy group. Given that every execution cycle during an exploration robotics mission requires a large amount of resources in terms of time, bandwidth, and energy, these results suggest that the use of a robot proxy could result in significant cost reductions.

To examine the differences in data quality, a two-way repeated measures

ANOVA was conducted with condition as a between-subjects variable. For each image, both rock size in the image (the fraction of pixels in the image which contained the rock face) as well as rock coverage (the fraction of the rock face visible in the image) were calculated. With respect to rock size, there were no significant differences for either instrument between the groups. This was largely a result of the design of the robot used in the simulation: In order to for the rock to be in the image, it had to be within a set, fairly narrow range from the robot, meaning that if a rock was in the image it would generally appear to be about the same size. With respect to rock coverage, the results indicated that participants in the Robot Proxy group took significantly better FI data on average (more of the rock was covered by the FI) ($F[1, 42] = 4.95$, $p < 0.05$) (Figure 10.2). There was no significant difference between the groups for the SPEC data, most likely due to the fact that the SPEC had a wider field of view and it was thus easier to capture the entire rock face in a SPEC image. The fact that the Robot Proxy group was able to obtain higher-quality FI data also demonstrates an improvement in efficiency over the No Proxy group.

10.2 Self-Evaluation of Performance

Regression analysis was used to ascertain if the presence of the robot proxy could explain the differences in participants' self-evaluation of their performance on the task. The regression analysis of efficiency on condition was marginally significant ($M_{RP} = 4.38$, $M_{NP} = 3.38$, $r^2 = 0.2$, $p < 0.1$). The multivariate correlations between participants' self-evaluations and their performance (Table 10.1) also showed that participants who rated themselves as more efficient at the task did demonstrate better performance: they executed fewer plans and their FI images covered more of the rocks.

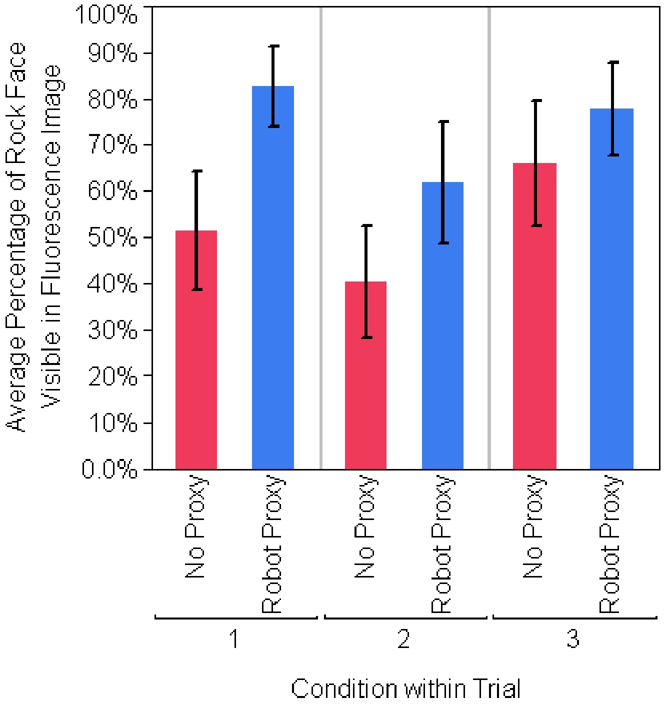


Figure 10.2: Mean and standard error of percentage of rock face visible for both conditions across all three trials.

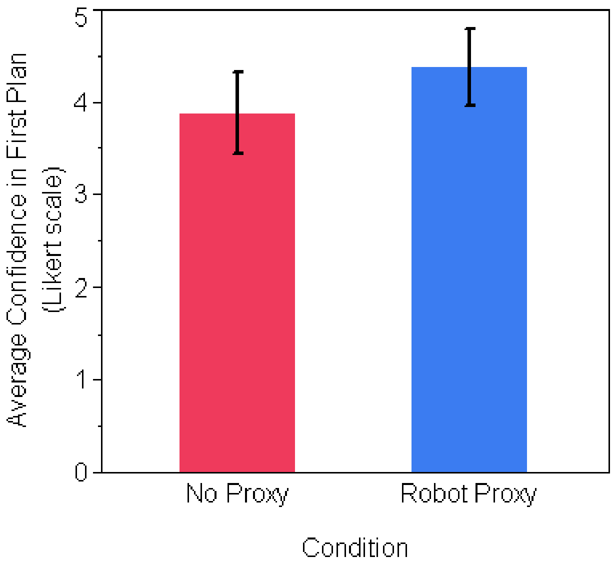
Table 10.1: Multivariate Correlations (Self-Evaluation of Performance and Actual Performance)

	Avg # Plans Executed	Avg % Rock Face Visible in FI	Efficiency	Confidence in First Plan	Confidence in Last Plan	Responsibility for Success	Responsibility for Failure	Collaboration
Avg # Plans Executed	1	**-0.33**	**-0.29**	-0.04	0.15	**0.22**	**-0.55**	0.15
Avg % Rock Face Visible in FI		1	**0.84**	-0.09	**0.36**	**-0.35**	-0.23	0.16
Efficiency			1	-0.23	**0.6**	**-0.66**	-0.26	**0.41**
Confidence in First Plan				1	-0.21	0.43	-0.09	-0.01
Confidence in Last Plan					1	**-0.69**	-0.46	**0.7**
Responsibility for Success						1	0.03	**-0.38**
Responsibility for Failure							1	-0.46
Collaboration								1

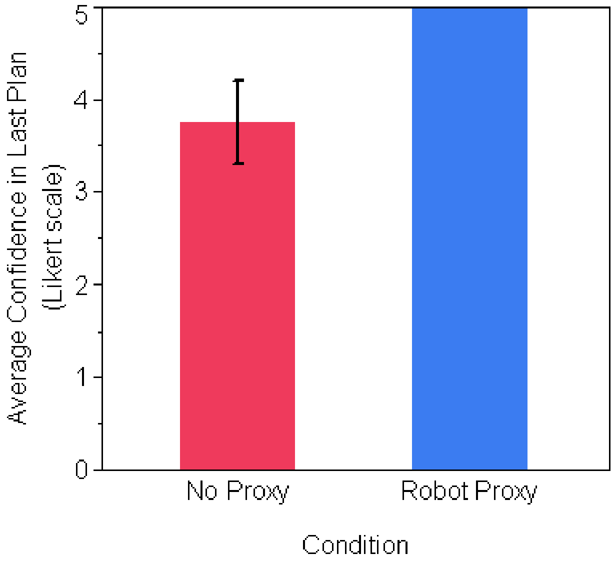
Statistically significant ($p < 0.05$)

Participants were also asked to rate to what extent they agreed or disagreed with the statements "The very first/last time I sent a plan to the robot, I felt confident that I knew what I was doing." While there was no significant difference between participants about their confidence while sending the first plan, there was a significant difference about their confidence while sending the last plan ($M_{RP} = 5$, $M_{NP} = 3.75$, $r^2 = 0.35$, $p < 0.05$). As shown in Figure 10.3(b), every participant in the Robot Proxy group strongly agreed that s/he felt confident while sending the last plan.

Participants' evaluation of their own performance also indicated that participants in the Robot Proxy group felt a stronger sense of collaboration with the system. Factor analysis was used to confirm that one question on collaboration with the system and one question on reliance on the system could be combined into a coherent factor of "Collaboration"; the Cronbach's alpha of that factor was calculated to be 0.86. The regression analysis of collaboration on condition was significant ($M_{RP} = 4.44$, $M_{NP} = 3.25$, $r^2 = 0.32$, $p < 0.05$); that is, participants in the Robot Proxy group agreed more strongly that they collaborated with the system. The multivariate correlations between participants' answers to the self-evaluation questions indicated that participants who agreed more strongly that they collaborated with the system also rated themselves as more efficient at the task, more confident in the last plan they sent to the robot, and rated themselves as less responsible for their successes (they rated the system as more responsible for their successes) (see Table 10.2). In other words, participants who felt they were collaborating with the system rated themselves as more efficient at the task, more confident at the end of the task, and gave the system more credit for successful plans.



(a) Mean and standard error of participants' confidence with respect to the first plan they sent to the robot.



(b) Mean and standard error of participants' confidence with respect to the last plan they sent to the robot.

Figure 10.3: Participants' ratings of how confident they were when sending (a) their first plan to the robot and (b) their last plan to the robot.

Table 10.2: Multivariate Correlations (Self-Evaluation of Performance)

	Efficiency	Confidence in First Plan	Confidence in Last Plan	Responsibility for Success	Responsibility for Failure	Collaboration
Efficiency	1	-0.23	**0.6**	**-0.66**	-0.26	**0.41**
Confidence in First Plan		1	-0.21	0.43	-0.09	-0.01
Confidence in Last Plan			1	**-0.69**	-0.46	**0.7**
Responsibility for Success				1	0.03	**-0.38**
Responsibility for Failure					1	-0.46
Collaboration						1

Statistically significant ($p < 0.05$)

These feelings of collaboration extended to the point that the participants in the Robot Proxy group gave the system significantly more credit for successful plans than members of the No Proxy group ($M_{RP} = 0.80$, $M_{NP} = 0.56$, $r^2 = 0.36$, $p < 0.05$); as shown in Figure 10.4(a), no member of the Robot Proxy group claimed more than 66% responsibility for successful plans (every Robot Proxy participant gave the system at least 33% responsibility for success). There was no significant difference with respect to assigning blame for failures, although the range of responses from the Robot Proxy group was smaller (Figure 10.4(b)); no member of the Robot Proxy group felt more than 66% responsible for failed plans.

10.3 Establishing Common Ground

As demonstrated in Chapter 4, two of the key components of common ground in human-robot interaction are the robot's state and its context. In order to estimate participants' common ground with the robot, before each plan was sent to the robot for execution, participants were asked to predict:

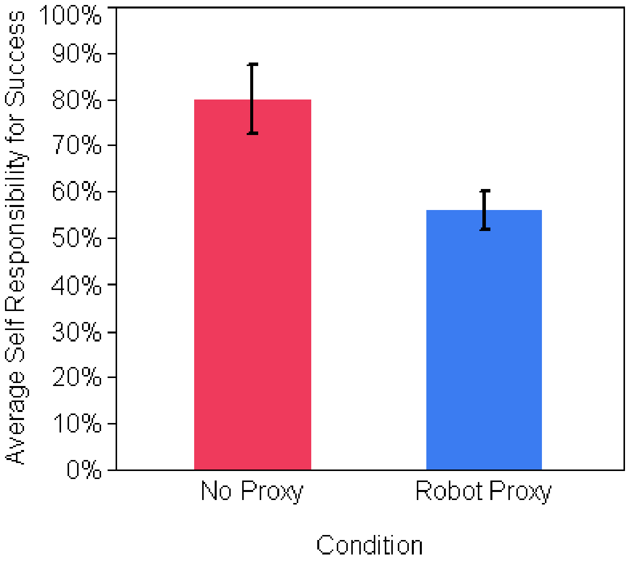
- which instruments' fields of view would overlap with each other (robot state)
- whether each image taken would contain all, some, or none of the rock (robot context)
- the distance between the center of the robot and the center of the rock at the end of the plan (robot context)

Participants also indicated how confident they were in each of their predictions.

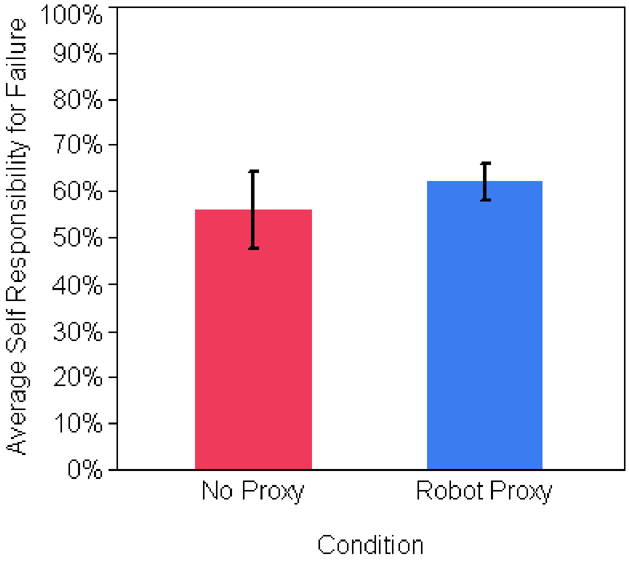
Overall, participants in the Robot Proxy group had a better understanding of the robot's state and context than participants in the No Proxy group. With respect to the robot's state, a two-way repeated measures ANOVA with condition as a between-subjects variable and trial as a within-subjects factor indicated that there was a main effect for condition: the Robot Proxy group made significantly more accurate predictions of whether or not instruments would overlap ($F[1, 39] = 10.00$, $p < 0.01$)[1] (Figure 10.5(a)). With respect to the robot's context, the Robot Proxy group made significantly more accurate predictions of how much of the rock would appear in each

---

[1] The degrees of freedom differ for this measure as not every plan contained multiple instrument readings.



(a) Mean and standard error of participants' assignments of their own responsibility for successful plans.



(b) Mean and standard error of participants' assignments of their own responsibility for failed plans.

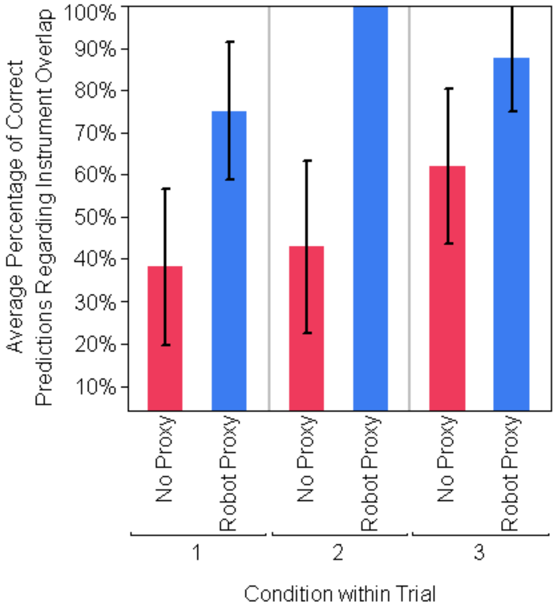
Figure 10.4: Participants' ratings of how much responsibility they assigned themselves for (a) successful plans and (b) failed plans.

image ($F[1, 42] = 26.56$, $p < 0.01$) (Figure 10.6(a)) and marginally significantly less error in estimating the final distance between the robot and the rock ($F[1, 42] = 3.85$, $p < 0.1$) (Figure 10.7(a)). No main effect for trial or interaction effect was found. This suggests that the Robot Proxy group had a better understanding of the robot's state and context than the No Proxy group and therefore more common ground with the robot. Participants in the Robot Proxy group were also significantly more confident in their predictions for each of the three prediction types: for predicting instrument overlap ($F[1, 39] = 43.01$, $p < 0.01$) (Figure 10.5(b), predicting how much of the rock would appear in each image ($F[1, 42] = 26.56$, $p < 0.01$) (Figure 10.6(b)), and predicting the distance between the robot and rock ($F[1, 42] = 20.93$, $p < 0.01$) (Figure 10.7(b).

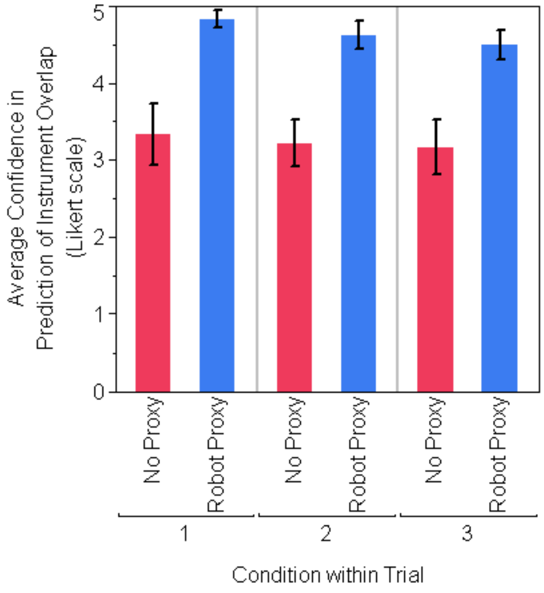
The multivariate correlations between participants' understanding of the robot's state and context and participants' performance demonstrate that greater common ground is correlated with better task efficiency (Table 10.3). Participants who scored better at predicting how much of the rock would be in a given image also, on average, captured more of the rock in their FI images and SPEC images. In addition, participants who scored better at predicting whether different instruments would overlap executed fewer plans.

10.4 Constraints and Limitations

In order to ensure that the task could be completed by participants within a reasonable length of time, the functionality of the robot proxy was somewhat limited. While the system had no knowledge of where rocks were located, the robot proxy provided feedback under the assumption that each locale contained exactly one 12cm by 12cm rock, such that any overlapping instrument readings must overlap by at least enough to cover a rock. (Participants in the Robot Proxy group were told this limitation during the tutorial.) Functionality supported by the Robot Proxy which was not needed to complete the task was disabled: participants could not add constraints between actions, define targets explicitly, or construct inclusive groups.

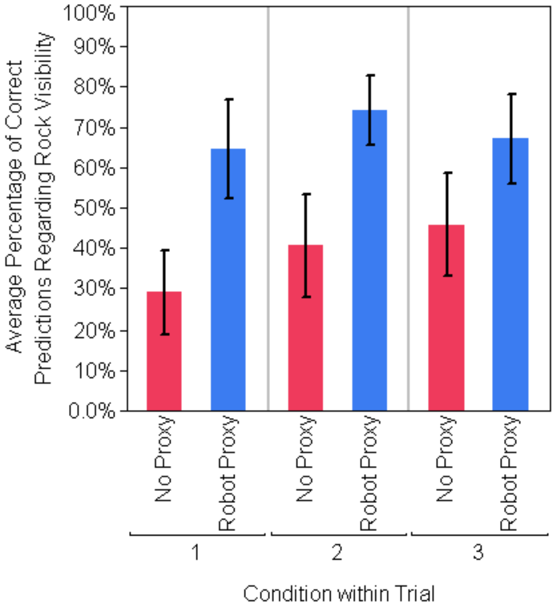


(a) Mean and standard error of participants' percentage accuracy in predicting instrument overlap.

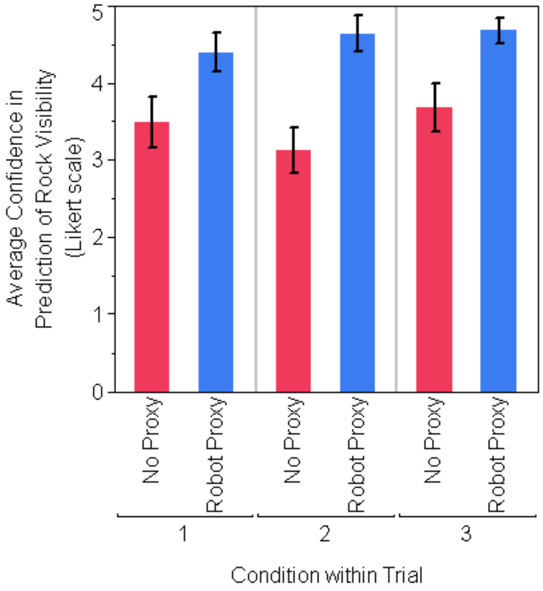


(b) Mean and standard error of participants' confidence in their accuracy in predicting instrument overlap.

Figure 10.5: Participants' (a) average percentage accuracy in predicting image overlap and (b) confidence in their predictions.

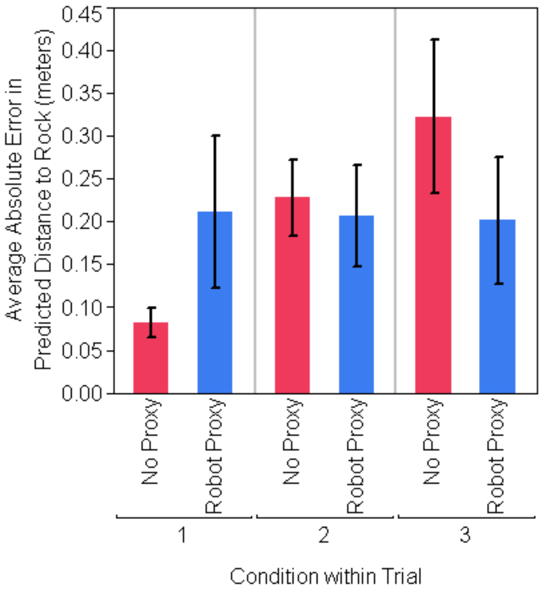


(a) Mean and standard error of participants' percentage accuracy in predicting how much of the rock would be visible in an image.

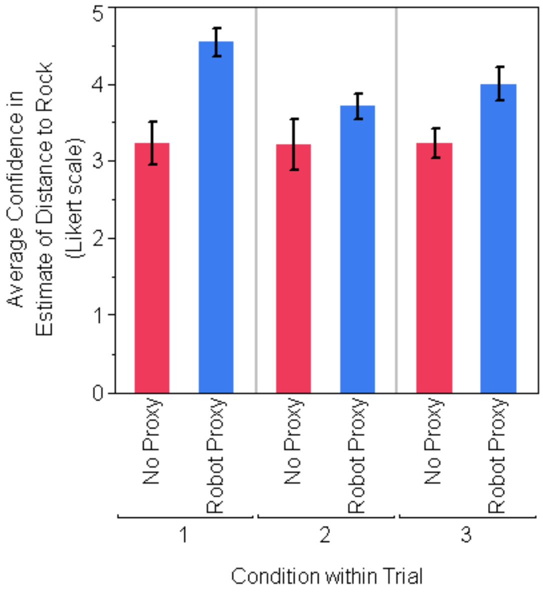


(b) Mean and standard error of participants' confidence in their accuracy in predicting rock visibility.

Figure 10.6: Participants' (a) average percentage accuracy in predicting how much of the rock would be visible in an image and (b) confidence in their predictions.



(a) Mean and standard error of participants' error in estimating the distance from the robot to the rock (meters).



(b) Mean and standard error of participants' confidence in estimating the distance from the robot to the rock.

Figure 10.7: Participants' (a) average error in estimated distance in meters between the robot and the rock and (b) confidence in this estimate.

Table 10.3: Multivariate Correlations (Establishing Common Ground and Task Performance

	Plans Executed	Avg % Rock Face Visible in FI	Avg % Rock Face Visible in SPEC	Rock Visibility % Predictions Correct	Instrument Overlap % Predictions Correct	FI Average Rock Pixels	SPEC Average Rock Pixels	Fragment Distance Average Absolute Error
Plans Executed	1	-0.61	-0.33	-0.19	**-0.41**	-0.37	-0.05	0.04
FI Average Rock Face		1	0.21	**0.52**	0.19	0.74	-0.12	-0.09
SPEC Average Rock Face			1	**0.27**	0.2	-0.09	0.26	-0.17
Rock Image Percent Predictions Correct				1	0.1	0.28	0.01	-0.32
Instrument Overlap Percent Predictions Correct					1	0.05	0	0.11
FI Average Rock Pixels						1	-0.04	0.17
SPEC Average Rock Pixels							1	0.09
Fragment Distance Average Absolute Error								1

Statistically significant ($p < 0.05$)

It is also important to note that participants in both conditions were prevented from entering invalid parameters when adding actions to their plans. As a result, in this study it was impossible to execute an action which was physically impossible for the robot. This seemed a sensible limitation given that the Life in the Atacama mission planning software also prevented users from entering erroneous parameter values.

10.5 Support for Robot-Proxy Grounding

While the participants in this study were more representative of expert roboticists than the type of highly-trained scientists who participate in scientific exploration missions, the results demonstrated the significant impact of the use of a robot proxy on task efficiency, self-evaluation of performance, and establishing common ground. The task used in this study was more challenging and open-ended than the task in the proof-of-concept study, yet nearly every participant was able to collect enough data to complete the task correctly (e.g. correctly identify life on every rock). However, the data did not indicate that there were any significant linear learning effects across the trials as were found in the proof-of-concept study; this may have been due to the more challenging nature of the task.

This study has demonstrated the effectiveness of using a robot proxy to improve task performance and promote common ground. Participants who utilized the robot proxy were more efficient at the task than participants without the robot proxy: members of the Robot Proxy group executed fewer plans on the robot, and they collected higher-quality data. Participants using the robot proxy perceived themselves as more efficient at the task (which was confirmed by their actual performance), and by the end of the task, every participant who used the robot proxy strongly agreed that he felt confident in performing the task. Participants in the Robot Proxy group also felt a stronger sense of collaboration with the system, to the point that they gave the system significantly more credit for successful plans.

Not only did robot proxy users perform more efficiently, more confidently, and with stronger feelings of collaboration, but the results also suggest that they established more common ground with the robot. In particular, participants in the Robot Proxy group had a better understanding of the robot's state and its context, two key aspects of common ground for human-robot interaction. This suggests that the Robot Proxy was successful in promoting common ground and improving task performance.

Chapter 11

Conclusion

The goal of this thesis is to improve upon the encoder-decoder model of communication utilized by traditional exploration robotics systems. According to this model, the user encodes her goals through an interface which translates them into machine-readable actions which are sent to the robot. At execution time, the robot uses planner to decode and schedule the necessary low-level commands and an executive process to direct the execution of the commands. The drawbacks to this model are that, particularly in the case of remote exploration robotics systems, the user only receives feedback from the robot at the end of its execution cycle (in the form of data return). However, time, energy, and communications bandwidth are extremely valuable resources. If a plan results in poor-quality data, these resources may be used inefficiently. In addition, given that the robot only receives a list of actions, it has no information about the user's higher-level goals. In the event that an action fails at execution, the robot does not have any additional information to guide its decision-making processes.

This thesis set out to answer the following question:

> In a system consisting of a human and an autonomous, mobile robot engaged in an exploration task, can we improve *efficiency* by helping the user develop a more accurate *mental model* of the robot and ensuring that the actions that the robot executes are consistent with the user's *underlying goals*?

The primary hypothesis of this thesis is that promoting common ground between the human and robot will result in fewer execution cycles required to complete a task, higher-quality data returned from the robot, better mental models formed by the user about the robot's behavior, and more information for the robot about the user's higher-level goals. Studies of the Personal Exploration Rover museum exhibit and the Life in the Atacama project have demonstrated the wide variety of problems which can result from a lack of common ground between user and robot, even as the robot becomes more autonomous. These studies also demonstrate the importance of the robot's state and context as part of this common ground.

Thus, the goal of this work is to promote common ground between a human and robot engaged in an exploration task. However, due to the constraints and costs of the grounding process for exploration robotics tasks, it is not possible for the human and robot to interact directly in real-time, as is the case for human conversational grounding. To solve this problem, this thesis introduces the *robot proxy*, a software system with which the user can "converse" in real time while she formulates a plan. A proof-of-concept study validated this approach.

The purpose of the robot proxy is to ensure that the plans which are sent to the robot are both physically possible and consistent with the user's higher-level goals. In order to accomplish this, the robot proxy needs a goal representation, robot model, and goal validation system. The goal representation presented in this thesis is derived from observations of the Life in the Atacama science team's needs and strategies for exploring the desert. The robot model is based on a physical robot as modeled in the *gazebo* robot simulator architecture. The goal validator checks that, in a given plan, individual actions are physically possible, that constraints between actions at the locale level are preserved, and that constraints between locales are preserved. Through the goal validation system, the user and robot build common ground as the robot gains information about the user's goals and the user receives information about the robot's capabilities and limitations. The user may then modify her plan based on this feedback. The grounding process continues during each iteration that the goal validation system is used as the robot and user receive more information about one another.

A final evaluation of the robot proxy implementation demonstrated that the robot proxy does answer the thesis question: in an example exploration task, the use of a robot proxy did reduce the number of execution cycles needed to complete the task, resulted in higher-quality data, and resulted in greater common ground between robot proxy users and the robot.

This work presented in this dissertation supports the hypothesis that by utilizing a common ground-based model of exploration planning, it is possible to: improve task efficiency, help the user generate more accurate mental models of the robot, and ensure that the actions which the robot executes meet the user's goals. In particular, the results of the final robot proxy evaluation indicated both that participants who utilized the robot proxy were more efficient at the task in terms of number of execution cycles and the quality of the resulting data and had more accurate mental models of the robot, specifically on the limits of its capabilities such as fields of view and position accuracy, than participants who did not use the robot proxy. The analysis of the Life in the Atacama field work indicates that there were times during remote science operations when the scientists and robot had no common ground with respect to what the scientists were trying to do. In situations when instruments malfunctioned, actions could not be completed, and other unexpected failures occurred, it became very difficult for the science team to get the data they wanted. The LITA data indicate that, in a real-world situation, it would be difficult and sometimes impossible for users to get the data they want if the robot does not have the additional goal information which results from promoting common ground.

11.1 Contributions

This thesis makes four contributions to the field of robotics:

- **Detailed analyses of human-robot systems, adding to the body of knowledge of how these systems function and what problems still need to be addressed.** As described in Chapter 3, careful analysis of the Personal Exploration Rover museum exhibit and the Life in the Atacama remote exploration robotics project have demonstrated how a lack of common ground can result in errors, miscommunications, and inefficiencies in human-robot tasks. These studies represent some of the first work which focuses specifically on common ground issues relating to exploration robotics. Critical to building common ground is mutual knowledge of the robot's state and context; these components of common ground are not well-captured by the existing common ground literature (Chapter 4).

- **The application of common ground theory to exploration robotics through the development of the concept of Robot-Proxy Grounding.** This thesis introduces the concept of Robot-Proxy Grounding and explains the implementation of a proxy consisting of a goal representation, a goal validation system, and a robot model (Chapters 5, 7, and 8). The robot proxy represents a novel approach to promoting common ground given the constraints and costs of grounding in exploration robotics. A proof-of-concept study is used to demonstrate that the use of Robot-Proxy Grounding improves task efficiency in an example exploration robotics task (Chapter 6).

- **A goal representation for a specific domain, which may later be generalized to other types of HRI problems.** This representation is primarily based on observations of the scientists working on the Life in the Atacama project (Chapters 7 and 8). The goal representation focuses on the exploration strategy and constraints which were of interest to the Life in the Atacama science team.
- **An implementation of a robot proxy system which has been demonstrated to improve task performance for an exploration robotics task.** This thesis includes an implementation of a robot proxy and an interface to support Robot-Proxy Grounding for developing plans for exploration robotics missions (Chapters 7 and 8). A user study demonstrates that the use of the robot proxy results in improved efficiency on an exploration task and higher-quality data as well as improving users' common ground with the system and engendering stronger feelings of collaboration with the system (Chapters 9 and 10).

11.2 Generalization

When considering the generalization of this work to other domains, two factors must be considered separately:

- The concept of a robot proxy, a software system which actively builds common ground with a user in place of a physical robot
- The specific presentation-acceptance process introduced here as a means to promote grounding between a user and a robot proxy

The applicability of each of these concepts to other human-robot interaction problems depends on specific characteristics of the interaction as described in the following sections.

11.2.1 Building Common Ground Using a Robot Proxy

The usefulness of the concept of a robot proxy to other problem domains depends on the following characteristics of the human-robot interaction:

- Interaction mediation

- Degree of robot autonomy
- Copresence and joint attention
- Synchronicity
- Communication cost

Interaction mediation. The term "interaction mediation" refers to the degree to which a user communicates with a robot via a software interface. Interactions involving full mediation include the use of robots in urban search and rescue (i.e. [Burke et al., 2004; Burke and Murphy, 2004]), in which operators use a keyboard and mouse or joystick interface to send commands to a robot. Other robots use partially mediated interactions, such as the *Roboceptionist* [Gockley et al., 2006]: users type to the robot using a keyboard (mediated interaction), but the robot can respond to the presence of a person using computer vision or a laser range finder (direct interaction). Many other social robots interact directly with users without computer mediation, such as *Paro* [Wada et al., 2003], which responds when users touch its body.

The concept of a robot proxy is most applicable to robots which have a high degree of interaction mediation. Because the robot proxy is a software-based third party to the interaction, it can be used most effectively when added to an existing piece of mediating software.

Degree of robot autonomy. In order to be effective, a robot proxy must be able to act autonomously in order to promote common ground. A robot proxy will be less useful in systems in which the robot is teleoperated by the user, as this style of interaction does not afford as many opportunities for a proxy system to intervene. It may be easier to integrate a robot proxy into systems which already support some autonomous actions, whether the system has sliding autonomy [Brookshire, 2004], partial autonomy [Stubbs et al., 2007], or full autonomy [Moshkina and Arkin, 2003]. Without the flexibility to analyze robot commands on its own, a robot proxy will be unable to engage in the grounding process with users.

Copresence and joint attention. As defined by Clark and Brennan, two collaborators are copresent if they "share the same physical environment" Clark [Clark and Brennan, 1991, p. 141]. Each participant has the opportunity to see and hear what her collaborator is seeing and doing. If participants are focused on the same task or object, they are said to have joint attention [Brooks and Breazeal, 2006]. In the case of remotely controlled robots, the user and robot are not copresent, which is a significant challenge to building common ground [Cramton, 2001]; a robot proxy can help to overcome this challenge. In other cases, the robot and human are copresent but do not always have joint attention; for example, a person may be running a cleaning robot, such as the *Roomba*, without paying attention to it at all times. Because the user and robot do not always share joint attention, they may not both share the same information about the state of the world or the task. Cramton has identified this type of common ground problem as an uneven distribution of information [Cramton, 2001], and a robot proxy may be useful to help address this.

Users and robots who are both copresent and who always share joint attention may share more of the same contextual information about their surroundings as well as information about the state of their joint activity; a robot proxy would be less useful in this situation.

Synchronicity. Synchronicity refers to the extent to which the user and robot are interacting in real time. (Clark and Brennan refer to this property as cotemporality [Clark and Brennan, 1991].) Interactions between social robots such as *Paro* and the *Roboceptionist* take place in real time and are thus fully synchronous. In the case of the Mars Exploration Rover mission, scientists sent plans to the robots once a day, and these plans were executed later; thus, the interaction was asynchronous.

A robot proxy will be most useful when communication is asynchronous, as it will be available to promote common ground even when the physical robot cannot interact with the user. With more synchronous interactions, the user is better able to interact with the robot directly and a proxy may not be helpful.

Communication cost. In the event that communication between a user and robot is expensive, a robot proxy can be extremely beneficial. In the case of remote exploration robotics, missions are often limited in terms of the amount of bandwidth available for communication and the amount of time available for communication [Stubbs et al., 2006b]. These costs are discussed in detail in Section 5.1.2. Rather than interacting directly with the robot and consuming these resources, users are free to interact with a robot proxy very cheaply; they can work with the proxy for as long as they like whenever the robot itself is inaccessible. For human-robot interactions in which communication is essentially free, such as many one-on-one social interactions, the robot proxy does not provide this kind of savings.

11.2.2 The Robot-Proxy Grounding Presentation-Acceptance Process

The applicability of the specific grounding approach used here and its implementation is largely dependent on the following interaction characteristics:

- Revisability
- Reviewability
- Sequentiality
- Number of human interactors
- Number of robot interactors
- Task characteristics

Revisability. A human-robot interaction supports revisability if it is possible for the user to revise a set of instructions before sending it to the robot [Clark and Brennan, 1991]. This is generally the case in remote exploration robotics, in which scientists develop a plan and revise it before sending a final version to the robot for execution. Robot-Proxy Grounding is most applicable to these types of interactions, as the robot proxy helps the user revise plans before they are sent to the physical robot.

Reviewability. Reviewability is another constraint on the grounding process introduced by Clark and Brennan [Clark and Brennan, 1991]; if contributions to an interaction leave behind artifacts which can be later examined by the contributors, the interaction has reviewability. In the case of the Atacama project, the plans that the scientists created could be viewed later by the scientists and by the robot after the plan had been sent; the planning process had reviewability. Robot-Proxy Grounding must have reviewability such that the robot proxy and user can consult the entire plan at any point in its development.

Sequentiality. Sequentiality refers to whether it is possible for collaborators' turns to "get out of sequence," or be interrupted by unrelated activities [Clark and Brennan, 1991, p. 141]. The Robot-Proxy Grounding process presented in this thesis is structured so as to preserve sequentiality and requires that the robot proxy not be interruptable. The grounding process will be less effective as the sequentiality of the interaction decreases.

Number of human interactors. This thesis focuses on a robot proxy which interacts with only one human user; the proxy is not designed to manage common ground between a robot and several different people.

Number of robot interactors. The Robot-Proxy Grounding process presented here assumes that the user is creating a plan for only one robot. However, in exploration robotics problems with asynchronous communication, it may be possible to easily extend the robot proxy such that it represents a group of robots.

Task characteristics. The specific grounding process outlined in this thesis is best suited for tasks which are structured in terms of spatial objectives (i.e., the robot should go to a particular location and perform some sequence of actions).

This analysis demonstrates that whether the concept of a robot proxy may be useful for other HRI problems or whether the specific implementation presented in this thesis generalizes to other tasks are highly dependent on a variety of characteristics of the particular interaction under consideration.

11.3 Future Work

One of the next logical steps in this line of research would be to comparing the results of the robot proxy studies when a real robot is used (as opposed to a simulated robot). One would expect that the use of a real robot would reveal further significant differences in task performance and grounding between participants who use a robot proxy during planning and those who do not.

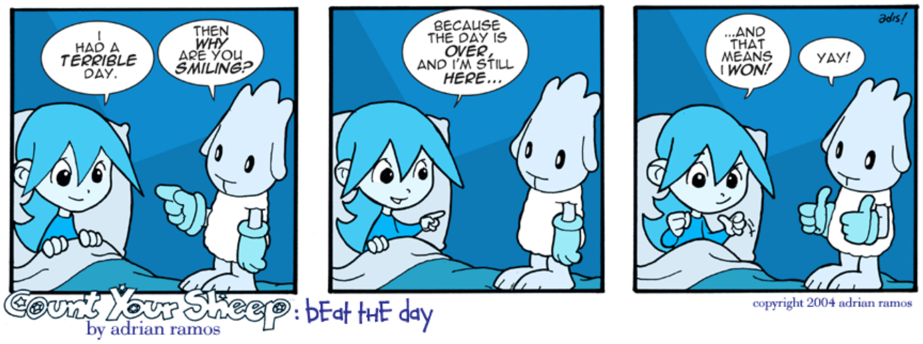
The results of this thesis suggest that for HRI grounding to occur, particularly with remote robots, the robots must learn and adjust their behaviors on the basis of "conversations" with people. Software systems can perfectly recall prior conversations with users, so robots might use this information to learn and adapt, just as humans do in conversational grounding. Implementing such adaptation might not be easy with current technology, but this thesis suggest this is a promising direction for future work and might address the recurring issues which were observed with missing contextual information and confusion about objects of reference.

In addition, an improved robot proxy could promote transparency by actively detecting errors in a user's understanding. When a user provides inappropriate responses to questions or expresses confusion, the robot proxy could detect these grounding problems and automatically disclose its logic by providing additional information, such as the evidence it used to make a particular decision. This would be a promising line of research as situation awareness research has not generally considered a robot with capabilities to detect and respond to grounding opportunities because situation awareness research historically has not focused on the conversation between users and the robot.

In order to fully reap the benefits of the robot proxy, future work could also include the creation of a plan repair system. This system would be used by the robot during its execution cycle in the event that planned actions fail; the plain repair system would be able to modify the original plan slightly in order to generate a sequence of actions which are consistent with the higher-level goals received from the user via the robot proxy. Such a plan repair system is extremely important in order for the physical robot to take full advantage of the common ground promoted by the robot proxy.

11.4 Summary

This thesis has presented a detailed analysis of human-robot systems which demonstrates the importance of common ground for successful human-robot interaction in exploration robotics tasks. In response to the constraints and costs associated with building common ground in exploration robotics, this thesis presents the robot proxy as a means to promote grounding. A robot proxy consisting of a goal representation based on extensive observations of a robotic exploration mission, a robot model, and a goal validation system, was implemented and evaluated. This evaluation supported the hypothesis that a robot proxy improves task performance and increases common ground between a human and robot cooperating on an exploration task.


I HAD A TERRIBLE DAY.
THEN WHY ARE YOU SMILING?
BECAUSE THE DAY IS OVER, AND I'M STILL HERE...
...AND THAT MEANS I WON!
YAY!
Count Your Sheep: bEat thE day
by adrian ramos
copyright 2004 adrian ramos

©2004 Adrian Ramos. Reprinted with permission.

Appendix A

Glossary

The following terms and their definitions are particularly salient to this thesis:

- **FI** - Fluorescence Imager
  An instrument on-board Zoë used to detect biological molecules such as DNA, chlorophyll, and protein. After water or dyes are sprayed on the ground, the imager emits light at certain wavelengths and captures images of other wavelengths; the molecules appear as bright spots in a grayscale image. The instrument has a field of view of ten square centimeters. A **full FI** uses water and all dyes, so a variety of biological molecules can be detected. A **chlorophyll FI** only uses water and can only detect chlorophyll.

- **locale** - Any place where the science team explicitly indicates in the plan that the robot should stop and perform some number of actions.

- **PAN** - Panorama

- **SPEC** - Spectrometer

- **SPI** - SPI camera
  One of three cameras at the top of Zoë's mast used for capturing single frames or panoramic images.

- **SPSU** - Standard Periodic Sampling Unit
  A data product designed by the LITA science team. By the end of the mission, the SPSU consisted of taking a full FI, taking some number of chlorophyll FIs over a 90-meter distance, taking another full FI,

taking some number of chlorophyll FIs over a 90-meter distance, and taking another full FI.

Appendix B

2005 Life in the Atacama Errors/Miscommunications

(See table on following page.)

Table B.1: Number of instances of each error type and miscommunication type from the LITA project in 2005.

Error/Miscommunication	**Instances**	**Percentage**
Lack of Common Ground (Science Team Missing Information)		
Miscom - desire for info	21	9.9
Miscom - rover capability	13	6.1
Error - bad position estimate	12	5.6
Error - bad plan	11	5.2
Error - bad data volume estimate	6	2.8
Total	63	29.6
Lack of Common Ground (Robot/Engineering Team Missing Information)		
Miscom - plan interpretation	18	8.4
Miscom - unaware of protocol	17	8.0
Miscom - data product file	9	4.2
Miscom - plan file	1	0.5
Total	45	21.1
Lack of Common Ground (Both Teams Missing Information)		
Miscom - priorities	10	4.7
Miscom - changed protocol	2	0.9
Total	12	5.6
Other		
Error - bad data product	56	26.3
Error - missing data product	17	8.0
Error - unrequested data product	15	7.0
Error - data product label	4	1.9
Miscom - human capability	1	0.5
Error - move in wrong dir	0	0.0
Error - rover move	0	0.0
Total	93	43.7
Summary		
Common Ground-Related Instances	120	56.3
Other Instances	93	43.7
Total	213	100.0

Appendix C

Excerpts from Life in the Atacama Field Notes

C.1 2005 Regular Operations (Low Autonomy)

Interpreting Context Images

On day 3, one scientist (X) mentioned that a context image, a stereo panoramic imager (SPI) image that was supposed to include the field of view of the fluorescence imager (FI), was not taken correctly:

> *X looks at a particular SPI image and says that "this is the messed up one." X says that this was supposed to be a context image. X reads the robot report. X says that the robot moved before taking the SPI image. X says, "I'm not sure why that happened."*

The scientists spent time trying to find the FI field of view in SPI context images, but sometimes the SPI images had not been taken correctly and this was impossible. The science team used both the images returned from the robot as well as the robot reports to figure out what had happened. On day 4, the science team talked about adjusting the commands sent to the robot to account for the fact that the robot moves back and plows 0.5 meters after an FI, before the context image is taken. Scientist X explained to scientist Y that the robot should have been moved only 1 meter, not 1.5 meters, before taking the context image:

> *At 2:09 p.m., X tells Y that "we" might have to adjust the drive precise command for the FI context image. X explains that after*

> *the FI, the rover moves back 0.5 meters for the marker plow. Y says that they are imaging the marker instead of the FI. X says that they might get the FI. X says that "we" may need to adjust. X says he thinks that the plow is right after the FI.*
>
> *At 4:14 p.m., X says that he and Y were talking. They talked about the fact that since the marker plow is done at the end of the FI, "we" need to adjust how much to move [the robot] back up. X says "we" should have asked to move 1 meter.*

After this, the science team adjusted their commands to move the robot one meter (days 5, 6, 7, 9, 10) and later commanded the robot to move 1.5 meters (days 9, 11, 12). On day 11, one scientist explained that the team realized they had to change back to requesting 1.5 meters instead of requesting 1 meter:

> *X says that they need the plow as a marker, so they found they did have to move up to 1.5 meters to get into the initial position.*

C.2 2005 Science Autonomy (High Autonomy)

Missing Fluorescence Image Follow-Ups

On day 1, one scientist (X) observed that the science autonomy system should have taken a full FI sequence in response to a positive chlorophyll signature (a "follow-up") but it did not. An engineer (Z) confirms that a follow-up should have been taken:

> *At 10:20 p.m., X is looking at a fluorescence image on the transect associated with locale 40 and asks, "Why didn't we have a follow-up on that?" X turns to Z and asks, "Shouldn't that have initiated a follow-up?" Z replies that yes, it should have.*

On day 1, an engineer (Z) explained that rounding errors contributed to the problem, and that the system was originally designed for transects that were much longer than what the scientists were using:

> *At 11:45 p.m., Z explains to X about some of the science on-the-fly problems that [the engineering team] had with the fluorescence imager. Z says the problem had to do with "round off"*

> *and "resource juggling." Z says that for fractional distances, the rover will always round up. X says that [the robot] went 180 meters. Z explains that the algorithm was designed for much longer distances. X explains that [the scientists] want to make the 180-meter traverse a standard procedure.*

On day 15, members of the science team and an engineer (Z) talked about other reasons why the follow-ups may not have been initiated:

> *Scientist A says that he is going to look at the transect between 800 and 810 to try and figure out why there were three full FIs and three chlorophyll only, but it doesn't look like there was a chlorophyll follow-up. A says this has happened before. Y suggests that it could be the result of the delta in the signal between the pre and post (the difference in the signal). Z says that the algorithm uses raw signal values.*

This technical discussion continued without resolving why the robot had not performed follow-ups as expected.

Bibliography

Bardram, J. E. and Bertelsen, O. W. (1995). Supporting the development of transparent interaction. In Blumenthal, B., Gornostaev, J., and Unger, C., editors, *Human-Computer Interaction, 5th International Conference, Selected Papers*, pages 79–90. Springer Verlag.

Bernstein, D. (2004). Parent, Docents and Robots: Examining Mediation at a Mars Rover Exhibit. in K. Crowley (chair), Islands of Expertise: An Approach to Exploring the Cognitive Ecology of Childhood. Symposium conducted at the meeting of the Visitor Studies Association, Albuquerque, NM.

Booch, G., Rumbaugh, J., and Jacobson, I. (1999). *The Unified Modeling Language User Guide*. Addison-Wesley.

Breazeal, C. (2003). Emotion and sociable humanoid robots. *International Journal of Human-Computer Studies*, 59:119–155.

Brennan, S. E. and Clark, H. H. (1996). Conceptual pacts and lexical choice in conversation. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 22(6).

Brennan, S. E. and Hulteen, E. A. (1995). Interaction and feedback in a spoken language system: A theoretical framework. *Knowledge-Based Systems*, 8:143–151.

Brooks, A. G. and Breazeal, C. (2006). Working with robots and objects: Revisiting deictic reference for achieving spatial common ground. In *Proceedings of the First Annual Conference on Human-Robot Interaction*, pages 297–304. ACM.

Brookshire, J. (2004). Enhancing multi-robot coordinated teams with sliding autonomy. Technical Report CMU-RI-TR-04-40, Carnegie Mellon University.

Burke, J. and Murphy, R. (2007). RSVP: An investigation of remote shared visual presence as common ground for human-robot teams. In *Proceedings of the 2007 ACM Conference on Human-Robot Interaction*, pages 161–168. ACM.

Burke, J. L., Murphy, R. L., Coovert, M. D., and Riddle, D. L. (2004). Moonlight in Miami: A field study of human-robot interaction in the context of an urban search and rescue disaster response training exercise. *Human-Computer Interaction*, 19(1–2):85–116.

Burke, J. L. and Murphy, R. R. (2004). Situation awareness, team communication, and task performance in robot-assisted technical search: Bujold goes to Bridgeport. Technical Report CRASAR-TR2004-23, University of South Florida.

Burridge, R., Graham, J., Shillcutt, K., Hirsh, R., and Kortenkamp, D. (2003). Experiments with an EVA assistant robot. In *Proceedings of the 7th International Symposium on Artificial Intelligence, Robotics and Automation in Space (i-SAIRAS-03)*.

Casper, J. and Murphy, R. R. (2003). Human-robot interactions during the robot-assisted urban search and rescue response at the World Trade Center. *IEEE Transactions on Systems Man and Cybernetics*, 33:367–385.

Clark, H. and Marshall, C. (1981). Definite reference and mutual knowledge. In Joshi, A. K., Webber, B. L., and Sag, I. A., editors, *Elements of discourse understanding*, pages 10–63. Cambridge University Press.

Clark, H. and Wilkes-Gibbs, D. (1986). Referring as a collaborative process. *Cognition*, 22(1):1–39.

Clark, H. H. (1996). *Using Language*. Cambridge University Press.

Clark, H. H. and Brennan, S. E. (1991). Grounding in communication. In Resnick, L. B., Levine, R. M., and Teasley, S. D., editors, *Perspectives on socially shared cognition*, pages 127–149. APA.

Clark, H. H. and Krych, M. A. (2004). Speaking while monitoring addressees for understanding. *Journal of Memory and Language*, 50:62–81.

Cramton, C. D. (2001). The mutual knowledge problem and its consequences for dispersed collaboration. *Organization Science*, 12(3):346–371.

Crowley, K. and Jacobs, M. (2002). Buildings islands of expertise in everyday family activity. In Leinhardt, G., Crowley, K., and Knutson, K., editors, *Learning Conversations in Museums*. Lawrence Erlbaum Associates.

Drury, J. L., Riek, L., and Rackliffe, N. (2006). A decomposition of UAV-related situation awareness. In *Proceedings of the First Annual Conference on Human-Robot Interaction*, pages 89–94.

Drury, J. L., Scholtz, J., and Yanco, H. A. (2003). Awareness in human-robot interactions. In *International Conference on Systems, Man and Cybernetics 2003*, volume 1, pages 912–918.

Endsley, M. R. (2000). Theoretical underpinnings of situation awareness: A critical review. In Endsley, M. R. and Garland, D. J., editors, *Situation Awareness: Analysis and Measurement*, chapter 1, pages 1–32. Lawrence Erlbaum.

Fong, T., Hiatt, L. M., Kunz, C., and Bugajska, M. (2006). The human-robot interaction operating system. In *Proceedings of the First Annual Conference on Human-Robot Interaction*, pages 41–48. ACM.

Fong, T., Nourbakhsh, I., and Dautenhahn, K. (2002). A survey of socially interactive robots: Concepts, design, and applications. Technical Report CMU-RI-TR-02-29, Carnegie Mellon University.

Fussell, S. R. and Krauss, R. M. (1992). Coordination of knowledge in communication: Effects of speakers' assumptions about what others know. *Journal of Personality and Social Psychology*, 62(3):378–391.

Gergle, D., Kraut, R. E., and Fussell, S. R. (2004). Language efficiency and visual technology: Minimizing collaborative effort with visual information. *Journal of Language and Social Psychology*, 23(4):491–517.

Gockley, R., Simmons, R., and Forlizzi, J. (2006). Modeling affect in socially interactive robots. In *Proceedings of the 15th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN 06), Hatfield, UK*, pages 558–563. IEEE.

Goodrich, M. A. and D. R. Olsen, J. (2003). Seven principles of efficient interaction. In *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, pages 3943–3948.

Goodrich, M. A. and Schultz, A. C. (2007). Human-robot interaction: A survey. *Foundations and Trends in Human-Computer Interaction*, 1(3):203–275.

Herlocker, J., Konstan, J. A., and Riedl, J. (2000). Explaining collaborative filtering recommendations. In *Proceedings of the 2000 ACM Conference on Computer-Supported Cooperataive Work*, pages 241–250.

Hinds, P. J., Roberts, T. L., and Jones, H. (2004). Whose job is it anyway? A study of human-robot interaction in a collaborative task. *Human-Computer Interaction*, 4(1–2):151–181.

Hine, B., Hontalas, P., Fong, T. W., Piguet, L., Nygren, E., and Kline, A. (1995). VEVI: A virtual environment teleoperations interface for planetary exploration. In *SAE 25th International Conference on Environmental Systems*, pages 615–628.

Issacs, E. A. and Clark, H. H. (1987). References in conversation between experts and novices. *Journal of Experimental Psychology: General*, 116(1):26–37.

Jones, H. and Hinds, P. (2002). Extreme work teams: Using SWAT teams as a model for coordinating distributed robots. In *Proceedings of Computer Supported Cooperative Work 2002, New Orleans, Louisiana*, pages 372–380.

Kiesler, S. (2005). Fostering common ground in human-robot interaction. In *Proceedings of the IEEE International Workshop on Robots and Human Interactive Communication (RO-MAN)*, pages 729–734.

Krauss, R. M. and Fussell, S. R. (1996). Social psychological models of interpersonal communication. In Higgins, E. T. and Kruglanski, A., editors, *Social psychology: Handbook of basic principles*, pages 655–701. Guilford Press.

Kraut, R. E., Fussell, S. R., and Siegel, J. (2003). Visual information as a conversational resource in collaborative physical tasks. *Human-Computer Interaction*, 18:13–49.

Li, S., Wrede, B., and Sagerer, G. (2006). A computational model of multi-modal grounding. In *Proceedings of the ACL SIGdial Workshop on Discourse and Dialog, in conjunction with COLING/ACL 2006*, pages 153–160. ACL Press.

Moshkina, L. and Arkin, R. C. (2003). On TAMEing robots. In *Proceedngs of the International Conference on Systems, Man and Cybernetics*, pages 3949–3959.

Moshkina, L., Endo, Y., and Arkin, R. C. (2006). Usability evaluation of an automated mission repair mechanism for mobile robot mission specification. In *Proceedings of the First Annual Conference on Human-Robot Interaction*, pages 57–63. ACM.

Murphy, R. R. (2004). Human-robot interaction in rescue robotics. *IEEE Transactions on Systems, Man, and Cybernetics*, 32(2):138–153.

Nourbakhsh, I., Hamner, E., Bernstein, D., Crowley, K., Ayoob, E., Lotter, M., Shelley, S., Hsiu, T., Porter, E., Dunlavey, B., and Clancy, D. (2005). The Personal Exploration Rover: Educational assessment of a robotic exhibit for informal learning venues. *International Journal of Engineering Education, Special Issue on Robotics Education.*

Nourbakhsh, I., Hamner, E., Bernstein, D., Crowley, K., Porter, E., Hsiu, T., Dunlavey, B., Ayoob, E., Lotter, M., Shelly, S., Parikh, A., and Clancy, D. (2004). The Personal Exploration Rover: The ground-up design, deployment and educational evaluation of an educational robot for unmediated informal learning sites. Technical Report CMU-RI-TR-04-38, Robotics Institute, Carnegie Mellon University.

Nourbakhsh, I. R., Crowley, K., Wilkinson, K., and Hamner, E. (2003). The educational impact of the Robotic Autonomy mobile robotics course. Technical Report CMU-RI-TR-03-18, Carnegie Mellon University.

Paek, T. and Horvitz, E. (1999). Uncertainty, utility, and misunderstanding: A decision-theoretic perspective on grounding in conversational systems. In *Psychological models of communication in collaborative systems: Papers from the AAAI Fall Symposium, November 5-7, North Falmouth, Massachusetts*, pages 85–92.

Parasuraman, R., Sheridan, T. B., and 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*, 30(3):286–297.

Schauble, L., Gleason, M., Lehrer, R., Bartlett, K., Petrosino, A., Allen, A., Clinton, C., Ho, E., Jones, M., Lee, Y., Phillips, J., Seigler, J., and Street, J. (2002). Supporting science learning in museums. In Leinhardt, G., Crowley, K., and Knutson, K., editors, *Learning Conversations in Museums*. Lawrence Erlbaum Associates.

Scholtz, J., Young, J., Drury, J., and Yanco, H. A. (2004). Evaluation of human-robot interaction awareness in search and rescue. In *Proceedings of the 2004 IEEE International Conference on Robotics and Automation*, pages 2327–2332.

Severinson-Eklundh, K., Huttenrauch, H., and Green, A. (2003). Social and collaborative aspects of interaction with a service robot. *Robotics and Autonomous Systems, Special Issue on Socially Interactive Robots*, 42(3-4).

Siino, R. and Hinds, P. (2004). Making sense of technology as a lead-in to structuring: The case of an autonomous mobile robot. In *Academy of Management Best Paper Proceedings*. OCIS E1–E6.

Sinha, R. and Swearingen, K. (2002). The role of transparency in recommender systems. In *Proceedings of the 2002 SIGCHI Conference on Human Factors in Computing Systems (CHI 2002)*. ACM.

Steinfeld, A. M., Fong, T. W., Kaber, D., Lewis, M., Scholtz, J., Schultz, A., and Goodrich, M. (2006). Common metrics for human-robot interaction. In *Proceedings of the First Annual Conference on Human-Robot Interaction*, pages 33–40. ACM.

Strauss, A. and Corbin, J. (1998). *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory*. SAGE Publications, 2 edition.

Stubbs, K., Bernstein, D., Crowley, K., and Nourbakhsh, I. (2005). Long-term human-robot interaction: The Personal Exploration Rover and museum docents. In *Proceedings of the 12th International Conference on Artificial Intelligence in Education*, pages 621–628.

Stubbs, K., Bernstein, D., Crowley, K., and Nourbakhsh, I. (2006a). Cognitive evaluation of human-robot systems: A method for analyzing cognitive change in human-robot systems. In *Proceedings of the 15th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pages 59–65.

Stubbs, K., Hinds, P., and Wettergreen, D. (2006b). Challenges to grounding in human-robot collaboration: Errors and miscommunications in remote exploration robotics. Technical Report CMU-RI-TR-06-32, Carnegie Mellon University.

Stubbs, K., Hinds, P., and Wettergreen, D. (2006c). Challenges to grounding in human-robot interaction: Sources of errors and miscommunications in remote exploration robotics. In *Proceedings of the First International Conference on Human-Robot Interaction*. ACM. Awarded Best Poster.

Stubbs, K., Hinds, P., and Wettergreen, D. (2007). Autonomy and common ground in human-robot interaction: A field study. *IEEE Intelligent Systems, Special Issue on Interacting with Autonomy*, 22(2):42–50.

Torrey, C., Powers, A., Marge, M., Fussell, S. R., and Kiesler, S. (2006). Effects of adaptive robot dialogue on information exchange and social relations. In *Proceedings of the First Annual Conference on Human-Robot Interaction*, pages 126–133. ACM.

Traxler, M. J. and Gernsbacher, M. A. (1992). Improving written communication through minimal feedback. *Language and Cognitive Processes*, 7(1):1–22.

Turkle, S. (1984). *The Second Self: Computers and the Human Spirit*. Simon and Schuster.

Veloso, M. M. (2002). Entertainment robotics. *Communications of the ACM*, 45:59–63.

Wada, K., Shibata, T., Saito, T., and Tanie, K. (2003). Effects of robot assisted activity to elderly people who stay at a health service facility for the aged. In *Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems*.

Yanco, H. A., Drury, J. L., and Scholtz, J. (2004). Beyond usability evaluation: Analysis of human-robot interaction at a major robotics competition. *Human-Computer Interaction*, 19(1–2):117–149.