

Collaboration, Dialogue, and Human-Robot Interaction

Terrence Fong¹, Charles Thorpe¹ and Charles Baur²

¹The Robotics Institute, Carnegie Mellon University, Pittsburgh, PA USA, {terry, cet}@ri.cmu.edu

²Ecole Polytechnique Fédérale de Lausanne, CH-1015 Lausanne EPFL, Switzerland, Charles.Baur@epfl.ch

Abstract

Teleoperation can be significantly improved if humans and robots work as partners. By adapting autonomy and human-robot interaction to the situation and the user, we can create systems which are easier to use and better performing. In this paper, we discuss the importance of collaboration and dialogue in human-robot systems. We then present a system based on collaborative control, a teleoperation model in which humans and robots collaborate to perform tasks. Finally, we describe our experiences using this system for vehicle teleoperation.

1. Introduction

1.1. Robot as partner

A robot is commonly viewed as a tool: a device which performs tasks on command. As such, a robot has limited freedom to act and will perform poorly whenever its capabilities are ill-suited for the task at hand. Moreover, if a robot has a problem, it has no way to ask for assistance. Yet, very often, the only thing the robot needs to get out of difficulty and to perform better is some advice (even a small amount) from a human.

Consider the situation in which a mobile robot is driving outdoors when its perception system classifies tall grass as a dangerous obstacle. Depending on the robot's autonomy, it may be unable to proceed or may decide to take a long, resource consuming detour. If, however, the robot is able to discuss the situation with a human, a better solution can be found. For example, if the robot asks "Is there really an obstacle ahead?" and displays a camera image, the human can help the robot decide that it is safe to move forward (i.e., to drive through the grass).

In general, even with advances in autonomy, we find that robots are more adept at making some decisions by themselves than others. For example, structured planning (for which well-defined processes or algorithms exist) has proven to be quite amenable to automation. Unstructured decision making, however, remains the domain of humans, especially when common sense is required [1]. In particular, robots continue to be poor at high-level perceptual functions, including object recognition and situation assessment [2].

It seems clear, therefore, that there are benefits to be gained if humans and robots work together. In particular, if we treat a robot not as tool, but rather as a partner, we find that we can accomplish more meaningful work and achieve better results. To do this, however, we need to enable humans and robots to collaborate: to engage in dialogue, to ask questions of each other, and to jointly solve problems.

1.2. Collaborative control

To address this need, we have developed a new system model for teleoperation called *collaborative control* [3]. In this model, a human and a robot work as partners (if not peers), collaborating to perform tasks and to achieve common goals. Instead of a supervisor dictating to a subordinate, the human and the robot engage in *dialogue* to exchange ideas, to ask questions, and to resolve differences.

We use the term collaborative control because it is analogous to the interaction between human collaborators. Specifically, when humans engage in collaboration, we encourage each collaborator to work with others towards a common goal. We also allow each collaborator to take self-initiative and to contribute as best he can. At the same time, however, we leave room for discussion and negotiation, so that potential solutions are not missed.

An important consequence of collaborative control is that the robot can decide how to use human advice: to follow it when available and relevant; to modify it when inappropriate or unsafe. This is not to say that the robot becomes "master": it still follows higher-level strategy set by the human. However, with collaborative control, the robot has more freedom in execution and can better function when the human is unavailable. As a result, teleoperation is better able to accommodate varying levels of autonomy and interaction.

The most significant benefit of collaborative control, however, is that it preserves the best aspects of supervisory control (use of human perception and cognition) without requiring time-critical or situation-critical response from the human. If the human is available, he can provide direction or assist problem solving. But, if he is not, the system can still function.

1.3. Key issues

In building collaborative control systems, we have found that there are four key issues which must be addressed. First, since the robot is free to use the human to satisfy its needs, the robot must have *self-awareness*. This does not imply that the robot need be sentient, merely that it be capable of detecting limitations (in what it can do and what the human can do), determining if it should ask for help, and recognizing when it has to solve problems on its own.

Second, the robot must have *self-reliance*. Since the robot cannot rely on the human to always be available or to always provide accurate information, it must have the capability to maintain its own safety. Specifically, the robot should be capable of avoiding hazards, monitoring its health and “safeing” itself when necessary.

Third, the system must have the capacity for *dialogue*. That is, the robot and the human need to be able to communicate effectively. Each participant must be able to convey information, to ask questions and to judge the quality of responses received. To an extent, traditional teleoperation has dialogue (i.e., the feedback loop), but the conversation is limited. With collaborative control, dialogue is two-way and requires a richer vocabulary.

Finally, the system must be *adaptive*. By design, collaborative control provides a framework for integrating users with varied skills, knowledge, and experience. As a consequence, the robot has to be able to adapt to different operators and to adjust its behavior as needed. For example, it should ask its questions based on the operator’s capacity to answer (e.g., a scientist and a child have vastly different domain knowledge). Similarly, it should handle information received from a novice differently than that received from an expert.

2. Background

2.1. Human-robot interaction

Humans have interacted with robots since the 1940’s. In the beginning, this interaction was primarily unidirectional: simple on-off controls or analog joysticks for operating manipulator joints and remote vehicles. Over time, as robots have become more intelligent, the nature of communication between humans and robots has becoming less and less like that of using a passive hand tool and more and more like the relationship between two human beings [4].

Human-robot interaction (HRI) can be defined as “the study of the humans, robots, and the ways they influence each other”. As a discipline, HRI regards the

analysis, design, modeling, implementation, and evaluation of robots for human use. HRI is strongly related to human-computer interaction (HCI) and human-machine interaction (HMI). HRI, however, differs from both HCI and HMI because it concerns systems (i.e., robots) which have complex, dynamic control systems, which exhibit autonomy and cognition, and which operate in changing, real-world environments.

HRI may occur through direct, proximal interaction (e.g., physical contact) or may be mediated by a user interface (“operator interface” or “control station”). In the latter case, the interface acts as a translator: it transforms human input (from hand-controllers or other control devices) to robot commands and provides feedback via displays. When the human and robot are separated by a barrier (distance, time, etc.) and information is exchanged via a communication link, then the interaction is called teleoperation.

Takeda et al. classify HRI into four categories [5]. Primitive interaction is communication via computer-based interfaces. Intimate interaction is direct, one-to-one interaction (e.g., gesture). Loose interaction is interaction at a distance. Cooperative interaction involves automatically introducing additional robots and people as needed by the interaction.

Milgram, Zhai and Drascic claim that for telemanipulation, human-robot communication can be classified into continuous and discrete languages[2]. Continuous language represents continuously distributed spatial or temporal information, e.g., analogue displays and input devices such as mice and joysticks. Discrete language consists of independent elements such as spoken commands and interface tools.

Laengle, Hoeniger, and Zhu discuss how humans and robots can function as a team [6]. Humans perform task planning, monitoring and supervision. Robots act as intelligent, autonomous assistants and interact symbolically and physically. This interaction is achieved via natural language, gestures, and touch.

Sheridan notes that one of the challenges for human-robot communication is to provide humans and robots with models of each other [4]. In particular, he claims that the ideal would be analogous to two people who know each other well and can pick up subtle cues from one another in order to collaborate (e.g., musicians playing a duet).

In recent years, much of the work in HRI has focused on making robots more “human”. Specifically, many researchers have been developing robots which perform human tasks [7][8][9], which exhibit human traits [10], and which can interact via natural language and gestures [11][12].

2.2. Human-computer collaboration

Collaboration is a process in which two or more parties work together to achieve shared goals. For collaboration to be effective, there must be: agreement of the goals to be achieved; allocation and coordination of tasks to be performed; shared context (keeping track of what has been done and what remains); and communication to exchange information and solutions. Numerous researchers have explored ways to allow humans and computers to share in this process.

Terveen writes that there are two major approaches to human-computer collaboration [13]. Human Emulation (HE) is closely tied to the artificial intelligence domain and assumes that the way to get computers to collaborate with humans is to endow them with human-like abilities, to enable them to act like humans. Human Complementary (HC) was developed by HCI researchers and assumes that computers and humans have fundamentally asymmetric abilities. The focus of HC, therefore, is to improve human-computer interaction by making the computer a more intelligent partner.

HE emerged from attempts to understand and model human communication. Most HE work, such as expert systems for trip planning or medical diagnosis, involve natural language processing (text and spoken dialogue). Consequently, much emphasis has been placed on designing representation formalisms (beliefs, goals, plans, etc.) and on developing techniques for systems to model and adapt behavior to different users.

HC is a more direct and pragmatic approach to human-computer collaboration. Much of the work in HC has focused on developing interface technology to make human-computer communication more efficient and natural. This requires understanding human cognitive and perceptual abilities, how people work as individuals and in groups, and how to represent and present information. The HC approach has proven to be effective for student tutoring and for user critiquing.

3. Dialogue

3.1. Communication and conversation

Dialogue is the process of communication between two or more parties. Dialogue is a joint process: it requires the sharing of information (data, symbols, context) and of control between among the parties. Depending on the situation (task, environment, etc.), the form or style of dialogue will vary. However, studies of human conversation have revealed that many

properties of dialogue, such as initiative taking and error recovery, are always present.

When humans and machines (computers, robots, etc.) communicate, dialogue is usually mediated by an interface. Some interfaces (e.g., computer command languages) offer great power and flexibility, though at an associated high learning cost. Other interfaces, such as menus, are easier for novices because they make few assumptions about what the user knows. The common interface models for human-computer dialogue are: command languages, form-filling, natural language (speech or text), question-and-answer, menus, and direct-manipulation (e.g., graphical user interfaces)

Lansdale and Ormerod describe dialogue as being controlled by five factors [14].

- *Linguistic competence* is the ability to construct intelligible sentences and to understand the other's speech. Human-computer dialogue requires a vocabulary which associates labels with concepts (e.g., command words) and sequences of actions (grammar).
- *Conversational competence* are the pragmatic skills necessary for successful conversation. Human-human dialogue tends to proceed with greater ease than human-computer dialogue due to mechanisms such as inference. A user interface must be designed such that users can unambiguously impart intentions and receive feedback.
- *Nonverbal skills* such as gestures are used for turn-taking, to differentiate between statement & question, etc. These skills add coherence to a dialogue, act as adaptation and control cues, and provide redundant information.
- *Medium constraints* such as communication links force a number of modifications on user behavior (slowing down, confirmation, etc.) With user interfaces, input devices and displays mediate conversation, thus technical limitations directly influence the nature of the dialogue.
- *Task constraints* can determine the structure of dialogue. For example, operative languages such as military communications and air traffic control use restricted vocabulary, domain specificity and economical grammar (e.g., acronyms) in order to avoid misunderstandings and to efficiently communicate complex information.

Perhaps the strongest factor, however, which influences the structure and flow of dialogue is expertise. Studies of experts have consistently shown that the

ability to assimilate and process information depends largely upon how much is already known about the structure of that information. This is because prior knowledge allows experts to categorize and compress incoming information (which may appear meaningless and uncorrelated to novices) into known patterns [14].

As a consequence, when experts and non-experts (e.g., doctor-patient) engage in dialogue, the expert usually takes initial control of the conversation. The ensuing exchange tends to follow a question-answer sequence. But, when both parties are experts, control is shared and the dialogue is more focused. This dichotomy appears in user interfaces as well. In particular, many human-machine dialogues are designed: (1) to give experts more control than novices and (2) to ensure that a specific procedure is followed when users cannot be trusted to do so by themselves.

3.2. Dialogue management

Unless the interaction is simple (e.g., fixed or limited grammar), human-computer systems require dialogue management. The basic function of dialogue management is to translate user requests into a language the computer understands and the system's output into a language that the user understands [15]. In addition, dialogue management must be capable of performing a variety of tasks including adaptation, disambiguation, error handling, and role switching [16].

Role switching occurs because at any stage in a dialogue, one participant has the initiative (control) of the conversation. In a sense, initiative is a function of the roles of the participants. Dialogue systems may allow the user or the computer to take the initiative, or may allow both to switch roles as required. The hardest dialogues to model, by far, are those in which the initiative can be taken at any point in the dialogue [17].

The two most common methods for dialogue management are graphs and frames. With graph-based management, dialogue consists of a series of linked nodes, each containing a limited number of choices the user can make (e.g., "For reservations, press 2"). Since the dialogue is structured and because the system always has the initiative, errors are limited. With frame-based management, database queries drive the dialogue. This approach allows more flexibility for role switching, but is also less robust than graphs.

3.3. User model

Dialogue cannot make sense unless the user and the system have a reasonable understanding of each other. Given a user model, dialogue adaptation can be per-

formed by referring to a user's expertise, knowledge, and preferences [15]. For example, the way in which information is collected (filtering, classification, etc.) and presented (text, graphics, speech) can be adapted to the user.

A user model (or profile) contains a set of attributes which describe a user or a group of users. Because dialogue is highly situation and task dependent, the attributes contained in a user model vary from system to system. Some common attributes are: skill in using the system, knowledge/expertise level, personal preferences, type of information needed/wanted, and responsiveness.

The stereotype approach is the most popular user modeling method. With the stereotype approach, a designer defines appropriate subgroups of the user population (the stereotypes), identifies user behaviors that enable a system to categorize users into a subgroup, and represents the set of features that characterizes each stereotype [13].

How to most effectively construct a user model is an open research area. Information about users may be acquired explicitly (by questioning the user) or implicitly (inferring information based on user actions). The former approach can be time consuming and users may not accurately characterize themselves. The latter approach is often impractical because it is difficult to determine when a user is starting a new task [15][13].

4. System Design

4.1. Architecture

We have implemented our collaborative control system as a distributed set of modules, connected by a message-based architecture (see Figure 1). Some of the modules run standalone and operate asynchronously. Other modules, particularly those which process sensor data or operate hardware, have precise timing or data requirements and are run synchronously in real-time (10 Hz).

The control system includes a safeguarded teleoperation controller which supports a wide range of operator interfaces [18]. In particular, the controller supports varying degrees of cooperation between the operator and robot. The controller has a modular architecture and includes localization, map building, safeguarding, sensor management, and speech synthesis.

Our current user interface is the PdaDriver which runs on WindowsCE¹-based Personal Digital Assistants [19]. It provides a variety of command modes

¹WindowsCE is a registered trademark of Microsoft, Inc.

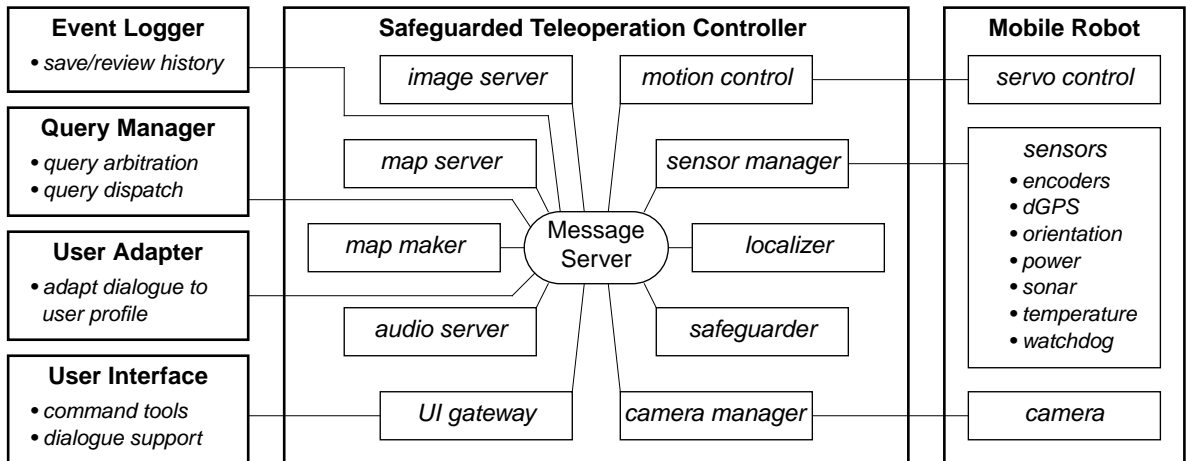


Figure 1: Collaborative control architecture

including image-based waypoint, rate/position control, and map-based waypoint. In addition, PdaDriver supports dialogue (i.e., the robot can query the user through the interface) and human-to-human interaction (audio and video).

We are using our collaborative control system to operate Pioneer² mobile robots. At present, we are using a Pioneer-AT and a Pioneer2-AT, both of which are skid-steered vehicles equipped with microprocessor-based servo controller, on-board computing (233 MHz Pentium MMX), 802.11 wireless ethernet and a variety of sensors (ultrasonic sonar, color CCD camera, differential GPS, and wheel encoders).

4.2. User model

We use stereotype user profiles in our collaborative control system. To create the user profiles, we addressed the following design issues:

- *attributes*: what information about a user is required and how should it be represented?
- *definition*: what are the appropriate subgroups of the users (the stereotypes)? how do we obtain information about each type of user?
- *use*: how will the system use information about different types of users (i.e., what useful behavioral adaptation can be made?)

4.2.1. Attributes

Although there are many attributes which can be used to describe teleoperation users, we have chosen to use the three listed in Table 1. We selected these attributes because they are well-suited for vehicle teleoperation

in unknown environments and provide a sufficiently rich basis for experimentation.

Table 1: User attributes

attribute	description
response accuracy	estimate of how accurately the user can answer a question
expertise	estimate of task skill / domain knowledge the user possesses
query interval	indicates how often the user is available/prefers to answer questions

An estimate of the user's response accuracy is important for when the robot needs to ask a safety-critical question. If a user is highly accurate, then the robot can place greater confidence in the response.

Similarly, an estimate of the user's expertise (whether general or domain-specific) is valuable for adapting dialogue and autonomy. If a user does not have expertise, then it is not useful to ask questions requiring such.

Finally, query interval can indicate several things about a user: availability (how much time he can dedicate to responding), efficiency (how quickly he can answer), and personal preference (how often he prefers to be interrupted with questions).

4.2.2. Definition

To support our initial experiments and evaluation of collaborative control, we defined three user stereotypes: novice, scientist, and expert. A novice is an inexperienced and untrained user. He does not know anything about teleoperation and has no special domain knowledge or task skill. Novices do not answer questions well (i.e., their responses are inaccurate) and prefer not to be interrupted.

²Pioneer is trademark of ActivMedia, Inc.

A scientist is also an inexperienced and untrained user. As with the novice, a scientist does not know anything about teleoperation. A scientist, however, does have domain expertise. Thus, a scientist can be expected to have difficulty answering teleoperation questions (e.g., “Is this an obstacle?”), but will be able to answer domain questions such as “Is this a rock?”.

An expert is defined as a user who knows everything. Thus, an expert is experienced, has training, and has both teleoperation and task expertise. Furthermore, an expert understands how the system is designed, how it is implemented, and how it functions. An expert can answer all questions quickly.

For each user stereotype, we assigned the attribute values shown in Table 2. In our current system, all user attributes are constant (no adaptation occurs during use) and reflect *a priori* assumptions about each user stereotype (e.g., novices take longer to respond).

Table 2: Stereotype attributes

stereotype	response accuracy	expertise	query interval
novice	30	30	60
scientist	50	100	30
expert	100	100	0

4.2.3. Use

We use each user profiles to configure human-robot interaction, to adapt the dialogue, and to modify how the robot acts. We configure interaction by modifying the user interface to fit the needs of each type of user. Each user profile defines which control modes are shown, what types of user input are allowed, and what types of feedback (displays) are presented.

We adapt dialogue by filtering messages with user attributes: only messages which are appropriate for each type of user are selected. We modify how the robot acts by configuring robot modules. Thus, the questions that the robot generates and how much autonomy the robot exhibits are user-dependent.

4.3. QueryManager

Under collaborative control, multiple robot modules may ask questions of the human at the same time. Thus, a collaborative control system needs query arbitration: a mechanism for choosing which questions to ask based on both immediate (local) needs and overall (global) strategy. In our system, the QueryManager performs this task with an attribute-based message filtering scheme [3].

Whenever a robot has a question to ask the human, it sends a message to the QueryManager. A message is defined by user attributes (see Table 1), query attributes (type, priority level, expiration time), and question-specific data (image, text, etc.) Our collaborative control system currently supports two query types: *y/n* (user must answer *y* or *n*) and *value* (user must provide a decimal value).

The QueryManager stores incoming messages into a cache, sorted by priority and by expiration. Whenever the cache contains unexpired messages, the QueryManager notifies the human (via a user interface) that there are pending queries. Then, when the human indicates that he is available to answer a question, the QueryManager selects a message by filtering the cache with the accuracy and expertise attributes.

Because the cache is priority-sorted, urgent questions have preference. Expired questions are discarded undelivered (i.e., the user is never asked a question which is no longer valid). Once a question is asked, the QueryManager waits until the query interval has expired before repeating the process.

4.4. Dialogue

In our system, dialogue arises from an exchange of messages between human and robot. Effective dialogue does not require a full language, merely one which is pertinent to the task at hand and which efficiently conveys information. Thus, we not use natural language and we limit the vocabulary and grammar to vehicle mobility issues (e.g., navigation).

At present, we are using approximately thirty messages to support vehicle teleoperation. Robot commands (translate, goto waypoint, etc.) and information messages (pose, status) are uni-directional. A query is expected to elicit a response (though the response is not guaranteed and may be delayed).

Table 3 lists the queries that a robot can ask. Several of these queries have variable response accuracy levels, which shows that the importance of a question can change with time or situation. For example, the accuracy of the “Stopped due to high temperature. What should the safety level be?” query varies from 0 to 100 percent. An accuracy value of zero means that the robot is willing to accept any response (i.e., any safety level). High accuracy values, however, indicate that the setting is critical to the robot’s continued health.

Three of the queries in Table 3 have non-zero expertise values. To answer these queries, therefore, the human must have a certain level of expertise. In our system, we do not distinguish between different types of experts (e.g., skilled pilot vs. geologist). This is

Table 3: Robot to user queries

query	type	response accuracy	expertise
Can I drive through (<i>image</i>)?	y/n	50	50
Is this a rock (<i>image</i>)? If you answer y, I will stay here.	y/n	0	50
The environment is very cluttered (<i>map</i>). What is the fastest I should translate?	value	0-100	50
My motors are stalled. Can you come over and help?	y/n	0	0
Motion control is currently turned off. Shall I enable it?	y/n	50	0
Safeguards are currently turned off. Shall I enable it?	y/n	50	0
Stopped due to collision danger. Disable safeguards?	y/n	100	0
Stopped due to high temperature. What should the safety level be?	value	0-100	0
Stopped due to low power. What should the safety level be?	value	0-100	0
Stopped due to rollover danger. Can you come over and help?	y/n	0	0

because our current work only requires distinguishing between trained (experienced) and untrained (novice) users. In general, however, we would use additional attributes to target queries to users having specific task or domain expertise.

5. Results

To gain insight into collaborative control, we briefly examined three vehicle teleoperation applications. In each of these applications, we observed that collaborative control enabled the robot to perform better than if left on its own. Specifically, we found that in situations where the robot does not know what to do, or in which it is working poorly, a simple human answer (a single bit of information) is often all that is required to get the robot out of trouble.

We should note that collaborative control does not force the human to respond or to stay ‘in the loop’. If the human is unavailable or cannot provide an accurate answer, the robot will still try its best to find a solution (as it would without collaborative control). However, if good human advice is given then the robot is able to perform better.

5.1. “A to B”

One of the basic tasks in vehicle teleoperation is “A to B”. That is, if the robot is at A, and if we know where B is located, the task is to control the robot’s actions so that it moves from A to B. As simple as this task may seem, successful execution is critical to many applications. In reconnaissance, for example, performance is directly related to being able to move accurately from point to point. Thus, it is important to make “A to B” as efficient and as successful as possible [3].

In the past, most vehicle teleoperation systems used only direct (manual) control. However, this mode of operation is known to have many performance limiting factors (e.g., operator fatigue). Moreover, direct control is not practical (or possible) for applications involving low-bandwidth or high-delay communications such as planetary exploration. Thus, many vehicle teleoperation systems now use some form of waypoint driving: the human specifies a set of waypoints which the robot then achieves on its own.

One problem with waypoint driving is that whenever the robot is moving autonomously, it may incorrectly identify an obstacle. Based on this false information, the robot would then be forced to look for an unobstructed path (which may take a long time or may not exist). Yet, if the robot is able to ask the human “Is this an obstacle?” and the human decides that it is not (based on his experience or his interpretation of the sensor data), then the robot can avoid the detour and take the more efficient solution (i.e., drive through).

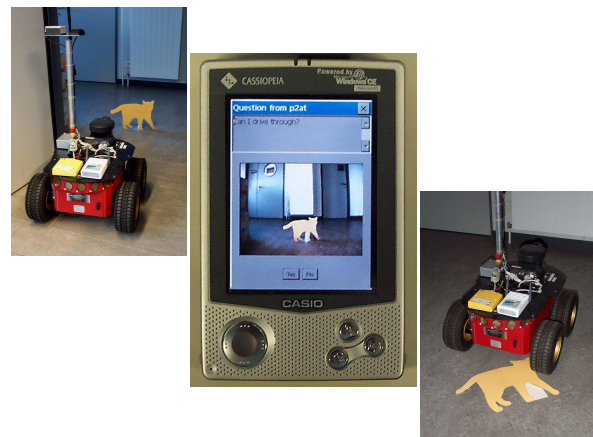


Figure 2: Query to the human: “Can I drive through?”

Figure 2 shows an example of this interaction occurring in an experiment we performed in an office environment. During this test, we placed a cardboard cat in the path of our mobile robot. The cardboard cat is detected as an obstacle by the robot’s sonar, thus forcing the robot to stop. At this point, the robot sends the human a camera image and asks whether or not it

is safe to drive forward. Based on image interpretation, the human answers “yes” and the robot rolls over the cardboard cat. Thus, by collaborating with the human, the robot is able to perform “A to B” more efficiently.

5.2. Collaborative exploration

Within the next thirty years, a human crew is expected to explore the surface of Mars. Considerable focus has already been given to human and robotic systems for planetary surfaces. Scant attention, however, has so far been paid to joint human-robotic systems. Yet, such systems offer significant potential to improve planetary missions. In particular, we believe that enabling humans and planetary rovers to work together in the field will increase productivity while reducing cost, particularly for operations such as material transport, sampling, survey, and site characterization [19].

To study collaborative human-robot exploration, we have developed a robot module, *RockFinder*, to autonomously locate “interesting rocks”. *RockFinder* is not intended for use in actual exploration, but merely serves as an example of the type of assistance rover could provide to an EVA crewmember (e.g., a geologist looking for samples). Thus, *RockFinder* does not actually examine geologic properties of specimens it encounters, but simply searches for objects having a pre-defined color signature.

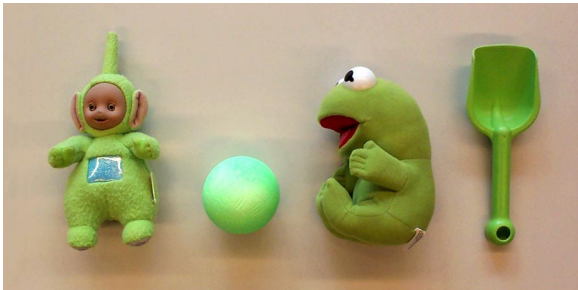


Figure 3: Some “rocks”

When it is running, *RockFinder* searches camera images for contiguous regions having a specified range of hue and saturation (i.e., a “color blob”). We use hue and saturation to reduce the impact of viewpoint, object geometry and changing illumination (i.e., hue and saturation are fairly constant if scene illumination color does not vary). Colored regions exceeding a pre-defined size are then marked as potentially being a “rock”. Figure 3 shows several of the “rocks” the *RockFinder* is designed to locate.

Whenever *RockFinder* detects a potential “rock”, it notifies the robot controller. At this point, the controller needs to decide what to do: Should it stop? Mark the location? Collect a sample? etc. Since it is difficult for

the robot to make this judgement, it must converse with the human in order to choose the proper action.



Figure 4: Collaborative exploration

Figure 4 shows an example of this interaction occurring during an experiment in a cluttered environment. Since there are many objects in the environment, it would be fatiguing and tedious for a human to manually search for the rocks. With collaborative control, however, the human can say to the robot: “Explore this area and tell me when you find an interesting rock”. Then, whenever the robot finds a candidate, it asks the human for his opinion. In this way, with human and robot collaborating, exploration is more efficient: the human is freed from performing a tedious task and the robot is able to search even though its on-board autonomy (i.e., rock finding competency) may be limited.

5.3. Multi-robot teleoperation

The American military is currently developing mobile robots to support future combat systems. These robots will be used to perform a variety of reconnaissance, surveillance and target acquisition (RSTA) tasks. Because these tasks have traditionally required significant human resources (manpower, time, etc.), one of the primary areas of interest is determining how a small number of operators can control a larger number of robots. We believe that collaborative control provides an effective solution for this problem.

For example, consider the situation in which a single operator needs to control multiple robots, each of which is capable of limited autonomous RSTA functions (e.g., “move to point Tango and collect imagery”). As they traverse unknown, unexplored or changing terrain each robot will likely have questions such as: “Is it safe to continue driving at this power level?”, “Is this obstacle dangerous?”, and “Can I drive over this terrain?”.

Since the human can only focus his attention on one robot at a time, we can use dialogue to unify and coordinate the multiple requests. Specifically, we arbitrate among the questions so that the human is always presented with the one which is most urgent (in terms of safety, timeliness, etc.) This allows us to maximize the human’s effectiveness at performing simultaneous, parallel control.

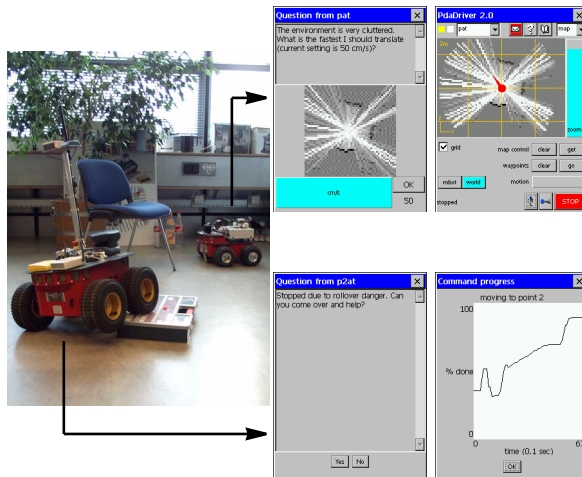


Figure 5: Multi-robot teleoperation

Figure 5 shows an example of this behavior occurring in multi-robot teleoperation. In this experiment, an operator is using two robots for reconnaissance. Collaborative control allows the human to quickly switch his attention between the two, directing and answering questions from each as needed. In our testing, we found that this to be an effective way to interact with independently operating robots. In particular, coordination arises naturally from query arbitration.

6. Discussion

6.1. Benefits

By enabling humans and robots to work as partners, we have found that teleoperation is easier to use and better performing. Collaboration enables the human and robot to complement each other, as well as allowing the robot to proceed when the human is unavailable. Dialogue lets us build systems which are user adaptive and which encourage unified human-robot teams.

We have observed that dialogue makes human-robot interaction adaptable. Since the robot is “aware” of to whom is speaking, it can dynamically decide whether or not to ask a question based on how much accuracy is required (i.e., how important it is to have a good answer). This is similar to what happens when humans have a medical problem. If the problem is

minor, we are willing to talk to a general practitioner. But, as a correct diagnosis becomes critical to our continued well-being, we insist on consulting specialists.

We have also found that there are situations for which dialogue (even a minimal amount) enables the robot to proceed or to perform better than without human intervention (i.e., than if left to its own devices). Moreover, this is true regardless of whether the human is a novice or an expert. What is important is to note is that even a novice can help compensate for inadequate sensing/autonomy.

Finally, it seems evident (though we have not yet confirmed this) that specific combinations of collaboration and dialogue are appropriate for multiple situations. In other words, it is possible that the interaction used for a specific user may be appropriate for all users when the system is constrained by factors such as bandwidth and delay.

For example, if communications are unconstrained, an expert may choose to use an interface which allows him to simultaneously generate multiple commands. In the same situation, the novice may prefer to use a more basic interface. However, if bandwidth is limited, both types of users may have to use the more sophisticated interface and to trust the robot to perform more steps. This is only one way the interface could vary.

6.2. Limitations

Although collaboration and dialogue provide significant benefits, we recognize that there are limitations. First, identifying which parameters are well-suited to a given task and assigning appropriate values for each query is difficult. If there are many tasks to perform or if task execution creates many questions, then dialogue may add considerable complexity to system design.

Second, if human-robot interaction is adaptive, then the flow of control and information through the system will vary with time and situation. This may make debugging, validation, and verification problematic because it becomes harder to precisely identify an error condition or to duplicate a failure situation.

Finally, working in collaboration requires that each partner trust and understand the other. To do this, each collaborator needs to model what the other is capable of and how he will carry out a given assignment. If the model is inaccurate or if the partner cannot be expected to perform correctly (e.g., a novice answering a safety critical question), then care must be taken. For the robot, this means that it may need to weigh human responses. For the human, this means the robot may not always behave as expected.

6.3. Future work

Although we have some evidence that adapting human-robot interaction to the user and to the task is beneficial, we cannot (as of yet) guarantee that this is always true. Thus, we feel it is important to evaluate the impact of dialogue on task performance. Some of the questions we would like to answer are: How is workload related to changes in dialogue? Does dialogue create task interference? To what extent does dialogue improve (or reduce) efficiency?

There are several ways in which we believe our existing system could be improved. Currently, we only arbitrate the queries the robot asks the human. A natural extension, therefore, would be to arbitrate commands from the human and from robot modules (e.g., obstacle avoidance). We could similarly arbitrate human responses (i.e., the robot would weigh responses based on the user providing them).

Another improvement would be to customize the way in which a question is asked, such as using icons or graphics instead of text, for different users. This would enable a wider range of users (children, handicapped, etc.) to have interact with the system.

Finally, instead of using statically defined user profiles, we could allow individual customization via a questionnaire or preference settings. This would enable finer control of human-robot interaction. A further refinement would be to learn (or to dynamically adapt) user profiles while the system is running.

Acknowledgements

We would like to thank the Institut de Systèmes Robotiques (DMT-ISR / EPFL) for providing research facilities. This work was partially supported by a grant from SAIC and the DARPA ITO MARS program.

References

- [1] Clarke, R. 1994. Asimov's Laws of Robotics: implications for information technology. *IEEE Computer* 26(12) and 27(1).
- [2] Milgram, P., Zhai, S., and Drascic, D. 1993. "Applications of Augmented Reality for Human-Robot Communication". International Conference on Intelligent Robots and Systems, Yokohama, Japan.
- [3] Fong T., Thorpe, C., and Baur, C. 1999. "Collaborative Control: a robot-centered model for vehicle teleoperation". AAAI Spring Symposium on Agents with Adjustable Autonomy. Stanford, California.
- [4] Sheridan, T. 1997. "Eight ultimate challenges of human-robot communication". IEEE International Workshop on Robot and Human Communication.
- [5] Takeda, H., et al. 1997. "Towards ubiquitous human-robot interaction". Working Notes for IJCAI-97 Workshop on Intelligent Multimodal Systems.
- [6] Laengle, T., Hoeniger, T., and Zhu, L. 1997. "Cooperation in Human-Robot-Teams". International Conference on Informatics and Control, St. Petersburg, Russia.
- [7] Baltus, G. et al. 2000. "Towards personal service robots for the elderly". Workshop on Interactive Robots and Entertainment, Pittsburgh, Pennsylvania.
- [8] Green, A. et al. 2000. "User centered design for intelligent service robots". International Workshop on Robot and Human Communication, Osaka, Japan.
- [9] Nourbakhsh, I., et al. 1999. An affective mobile robot educator with a full-time job. *Artificial Intelligence* 114(1-2).
- [10] Breazeal (Ferrel), C. 1998. "Regulating human-robot interaction using emotions, drives, and facial expressions". Agents in Interaction Workshop, Autonomous Agents, Minneapolis, Minnesota.
- [11] Rogers, T., and Wilkes, M. 2000. "The Human Agent: a work in progress toward human-humanoid interaction". International Conference on Systems, Man, and Cybernetics, Nashville, Tennessee.
- [12] Triesch, J., and von der Malsburg, C. 1998. "A gesture interface for human-robot interaction". IEEE Conference on Face and Gesture, Nara, Japan.
- [13] Terveen, L. 1994. An overview of human-computer collaboration. *Knowledge-Based Systems* 8(2-3).
- [14] Lansdale, M. and Ormerod, T. 1994. *Understanding interfaces: a handbook of human-computer dialogue*. Academic Press.
- [15] Goren-Bar, D. 2001. "Designing model-based intelligent dialogue systems". In Rossi, M., and Siau, K. *Information Modeling in the New Millennium*. Idea Group.
- [16] Abella, A. and Gorin, A. 1999. "Construct Algebra: analytical dialog management". Annual Meeting of the ACL, Washington, D.C.
- [17] Churcher, G., et al. 1997. Dialogue management systems: a survey and overview. Report 97.6, School of Computer Studies, University of Leeds.
- [18] Fong, T., Thorpe, C., and Baur, C. 2001. "A safeguarded teleoperation controller", International Conference on Advanced Robotics, Budapest, Hungary.
- [19] Fong, T., Cabrol, N., Thorpe, C., and Baur, C. 2001. "A personal user interface for collaborative human-robot exploration". International Symposium on Artificial Intelligence, Robotics, and Automation in Space, Montreal, Canada.