Experiments in Adjustable Autonomy

Michael A. Goodrich, Dan R. Olsen Jr., Jacob W. Crandall and Thomas J. Palmer
Computer Science Department
Brigham Young University

Abstract

Human-robot interaction is becoming an increasingly important research area. In this paper, we present our work on designing a human-robot system with adjustable autonomy and describe not only the prototype interface but also the corresponding robot behaviors. In our approach, we grant the human meta-level control over the level of robot autonomy, but we allow the robot a varying amount of self-direction with each level. Within this framework of adjustable autonomy, we explore appropriate interface concepts for controlling multiple robots from multiple platforms.

Introduction

The purpose of this research is to develop human-centered robot design concepts that apply in multiple robot settings. More specifically, we have been exploring the notion of adjustable autonomy and are constructing a prototype system. This prototype system allows a human user to interface with a remote robot at various levels of autonomy: fully autonomous, autonomous with goal biases, waypoint methods, intelligent teleoperation, and dormant. The objective is to allow a single human operator to interact with multiple robots and do so while maintaining reasonable workload and team efficiency. This objective is influenced by the desire to extend this work to allow multiple users to manage multiple robots from multiple interface platforms.

Related Literature

Relevant research in human-robot interaction can be loosely classified under five topics: autonomous robots, teleoperation, adjustable autonomy, mixed initiatives, and advanced interfaces. Of these topics, research in teleoperation is most mature; we refer to Sheridan's work for an excellent overview of these topics (Sheridan 1992). Perhaps the most difficult obstacle to effective teleoperation occurs when there are communication delays between the human and the robot. The standard approach for dealing with these issues is to use supervisory control. Work on teleautonomy (Conway, Volz, & Walker 1990) and behavior-based teleoperation (Stein 1994) are extensions to traditional supervisory

control that are designed specifically to account for time delays.

Alternative approaches to teleautonomy that focus on the operator include the use of predictive displays (Lane, Carignan, & Akin 2000) and the use of intelligent interface assistants (Murphy & Rogers 1996). Approaches that focus more on the human-robot interaction as a whole rather than in isolation include safeguarded teleoperation (Krotkov *et al.* 1996; Fong, Thorpe, & Baur 2001), mixed initiative systems (Fong, Thorpe, & Baur 1999), and adjustable autonomy-based methods (Dorais *et al.* 1998).

In addition to dealing with communication delays, adjustable autonomy has also been applied to problems where human workload and safety are considerations. The concept has been applied in both software (Pollack, Tsamardinos, & Horty 1999; Scerri, Pynadath, & Tambe 2001) and hardware agents (Perzanowski *et al.* 1999). Although promising, challenges in creating systems that effectively employ adjustable autonomy include issues in mixed initiatives (Ferguson, Allen, & Miller 1996; Perzanowski *et al.* 1999), intervention, responsibility, and trust(Inagaki & Itoh 1996). Researchers from aviation and other human-factors areas provide meaningful insights into the application of adjustable autonomy in the human-robot interaction domain (Inagaki 1995).

For many of the applications for which adjustable autonomy and mixed initiatives are appropriate, it is desirable to allow the human to interact with the robot as naturally as possible. This leads to research in advanced interfaces, such as gesture recognition (Kortenkamp, Huber, & Bonasso 1996; Voyles & Khosla 1995), emotive computing (Breazeal 1998), natural language-based interfaces, virtual reality-based displays (Steele, Thomas, & Blackmon 1998), and so on. Additionally, this also leads to research in robots learning from human operators (Boyles & Khosla 2001), and research in designing intelligent interface agents (Murphy & Rogers 1996).

In subsequent discussions, we elaborate on the differences between autonomous/semi-autonomous robots and mixed initiative human-robot systems. The key element in mixed initiative systems is the on-running dialogue between human and robot in which both parties share responsibility for mission safety and success. This work is well characterized by (Fong, Thorpe, & Baur 1999), who emphasize a robot-

centered view to human-robot interaction. Related concepts are also present in some approaches to shared control (Röfer & Lankenau 1999) as well as in situation-adaptive autonomy in aviation automation (Inagaki 1995).

Autonomous robot control and vehicle design has an extensive history. A complete review of the literature is beyond the scope of this paper, but we do note the seminal work of Brooks with behavior-based robotics (Brooks 1986). We further note the excellent textbooks on the subject by Murphy (Murphy 2000) and by Arkin (Arkin 1998). There are many approaches to behavior-based robotics, but in this paper we focus on approaches based on utilitarian voting schemes (Rosenblatt 1995) as well as artificial potential fields (Chuang & Ahuga 1998; Frixione, Vercelli, & Zaccaria 1998; Volpe 1994); the last of these papers has an excellent overview of pre-1994 work in the context of telemanipulation. Hierarchical approaches, which are the other major approach to designing autonomous vehicles, are characterized by the NIST RCS architecture (Albus 2000; 1991).

A related but relatively unexplored topic is that of collaborative teleoperation wherein multiple users control one robot (Goldberg *et al.* 2000). This work is important because it provides a foundation for multiple user/multiple robot interactions.

Autonomy Modes and Justification

The purpose of this section is to describe the levels of autonomy that are being included in our human-robot interaction system. Additionally, we present a justification for each of the autonomy modes we include. In the system we describe, the operator must switch between each autonomy mode but within each mode the robots have some authority over their behaviors.

Time Delays and Neglect

In designing an architecture that allows multiple users to interface with multiple robots, it is desirable to equip robots with enough autonomy to allow a single user to service multiple robots. To capture the mapping between user attention and robot autonomy, we introduce the neglect graph in Figure 1. The idea of the neglect graph is simple. Robot A's likely effectiveness, which measures how well the robot accomplishes its assigned task and how compatible the current task is with the human-robot team's mission, decreases when the operator turns attention from robot A to robot B; when robot A is neglected it becomes less effective.

A common problem that arises in much of the literature on operating a remote robot is time delays. Time delays between earth and Mars are around 45 minutes, between earth and the moon are around 5 seconds, and between our laptop and our robot around 0.5 seconds. Since neglect is analogous to time delay, we can use techniques designed to handle time delays to develop a system with adjustable autonomy. For example, when the operator turns attention from robot A to robot B, the operator introduces a time delay, albeit a voluntary one, into the interaction loop between the operator and robot A. Depending on how many robots the operator is
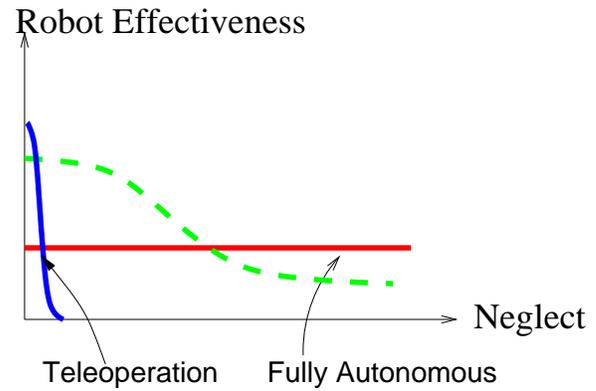

Figure 1: *The neglect curve. The x-axis represents the amount of neglect that a robot receives, which can be loosely translated into how long since the operator has serviced the robot. The y-axis represents the subjective effectiveness of the robot. As neglect increases, effectiveness decreases. The nearly vertical curve represents a teleoperated robot which includes the potential for great effectiveness but which fails if the operator neglects the robot. The horizontal line represents a fully autonomous robot which includes less potential for effectiveness but which maintains this level regardless of operator input. The dashed curve represents intermediate types of semi-autonomous robots, such as a robot that uses waypoints, for which effectiveness decreases as neglect increases.*

managing and depending on the mission specifications, it is desirable to adjust how much a robot is neglected. Adjusting neglect corresponds to switching between techniques for handling time delays in human-robot interaction.

As the level of neglect changes, an autonomy mode must be chosen that compensates for such neglect. In the literature review, several schemes were briefly discussed for dealing with time delays. Schemes devised for large time delays are appropriate for conditions of high neglect, and schemes devised for small time delays are appropriate for conditions of low neglect. At the lowest neglect level, shared control can be used for either instantaneous control or interaction under minimal time delays; at the highest neglect level, a fully autonomous robot is required.

We are now in a position to make two observations that appear important for designing robots and interface agents. First, the following rule of thumb seems to apply: *as autonomy level increases, the breadth of tasks that can be handled by a robot decreases*. Another way of stating this rule of thumb is that as efficiency increases tolerance to neglect decreases. Second, the objective of a good robot and interface agent design is to move the knee of the neglect curve as far to the right as possible; a well designed interface and robot can tolerate much more neglect than a poorly designed interface and robot.

Autonomy Modes

We have constructed (a) a set of robot behaviors and (b) an interface system that allows an interface agent running on

a laptop computer to interact with two Nomad SuperScout robots via a 11Mb/s wireless ethernet. A human can explicitly control the level of autonomy by selecting an appropriate mode from the interface, but once this mode is selected then the human can only influence the robot's behavior by issuing commands via the mediating interface agent. This inter-
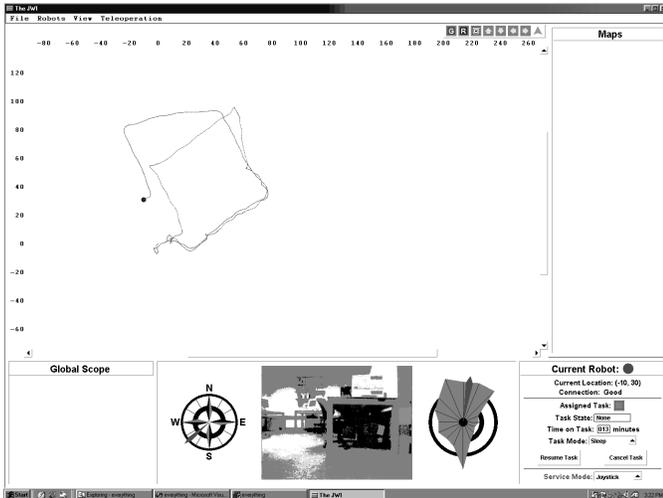

Figure 2: *A screen capture of the human interface.*

face, shown in the Figure 2, includes (a) a graphical depiction of robot behaviors and locations using a 2-D, god's-eye perspective, (b) a graphical depiction of sonar, compass and video data, and (c) icons, pull-down menus, and other tools for selecting robots, assigning tasks, and changing modes. Five levels of autonomy are supported: fully autonomous, goal-biased autonomy, waypoints and heuristics, intelligent teleoperation, and dormant. In this section, we discuss each of these levels (except dormant) in detail. More specifically, for each autonomy level, we (a) discuss the robot design technique (plus modifications) used to implement each autonomy level and (b) describe the expected neglect characteristics of this design. We discuss how we plan to validate the design shortly.

Full Autonomy   The system we have developed, which is based on a utilitarian-voting scheme similar to Rosenblatt's (Rosenblatt 1995), is designed to allow the robot to be situated in the environment and to initiate its own responses based on its perceptions. Our prototype system is built on a behavior-based scheme with a behavior arbiter responsible for selecting actuator settings via a voting method. This system uses vetoes and hijacks to constrain undesirable emergent behaviors that arise from the voting implementation while permitting desirable emergent behaviors to pass unhindered.

At this fully autonomous level, the robot's mission is to initiate responses to environmental stimuli and no human input is allowed to influence robot behavior. The purpose of a fully autonomous robot is to let the robot do what it needs to do and intervene when necessary. Under our implemen-

tation, the only fully autonomous behavior is for the robot to wander about and create a local map of its environment. Thus, it has low efficiency (although maps are helpful) but can tolerate a high level of neglect.

Goal-Biased Autonomy   In the voting method that we are using, it is possible for a human to specify a region of interest (by dragging-and-dropping a goal icon in the interface) or a region of risk (by dragging-and-dropping a threat icon in the interface). Furthermore, in the near future we expect to be able to tell the robot to wander in a particular direction until it finds a particular goal. In our design, these goal and risk regions do not directly control the robot's selected action, but they can be treated as any other behavior (where we use this term in the behavior-based robotic sense) and their vote is included in the action-selection mechanism encoded in the arbiter. This concept, which is compatible with Rosenblatt's work, is still in preliminary design stage.

Following the maxim, "an ounce of direction is worth a ton of intervention," it is desirable to allow a human to bias the autonomous behavior of the robot. By introducing goal/risk icons or by assigning a goal-seeking task, the user can guide the robot to a particular goal and thus achieve more user-specified tasks then the fully autonomous system. This increase in efficiency is accompanied by a decrease in the level of acceptable neglect since once the robot reaches the goal it will stop wandering and therefore stop generating local maps.

Waypoints and Heuristics   Included in the interface is the ability to specify not only goals/risks, but also heuristic directions wherein the human drags-and-drops iconic arrows in the interface to heuristically influence robot actions. Rather than implementing this level using the voting method of action selection, we use a potential-field-based approach wherein waypoints represent attractive potentials (that disappear when the robot reaches them), obstacles represent repulsive potentials, and heuristics represent constraints on the potential field (causing the resulting potential to be aligned along the direction of the heuristic). These constraints are tantamount to (hard and soft) waypoints, but are currently restricted to navigation-type tasks. Because of the problem with local minima in potential field approaches, we modify the conventional approaches by using satisficing decision theory to create local decision potentials. Essentially, these decision potentials always have a local attractive pull, normalized between zero and one, and a local repulsive push, also normalized between zero and one. Any decisions for which the attractive pull exceeds the repulsive push are satisficing, and any satisficing action can be chosen.

Introducing waypoints and heuristics improves the robot's ability to do a human-specified task whence efficiency increases. However, introducing waypoints and heuristics requires a more involved human whence the robot's level of autonomy is decreased and its tolerance for neglect decreases. This level of autonomy is, in essence, a task-automation approach, and can be coupled with a waypoint sequence that allows a robot to complete a more complicated task than possible using only potential fields.

Intelligent Teleoperation   At the teleoperation level, the human controls the robot via a Microsoft Sidewinder force feedback joystick while the interface displays video feedback and other robot information. Because (a) time delays exist between when a command is issued and when its effects are observed and (b) it is difficult to efficiently convey perfect situation awareness to a remote human operator, the robot treats human inputs (from the joystick) as desired directions, but the robot counter-balances this input by a robot-determined assessment of risks. Again, we use satisficing decision potentials to identify actions that are good enough in the sense that choices are directed by the human but modulated by the robot's sense of what is safe. This system is in the spirit of shared control (Röfer & Lankenau 1999), and includes safe-guarding which prevents the operator from running into obstacles (unless the operator persists long enough to cause the robot to acquiesce and execute the operator's command).

Although preliminary experiments demonstrate that this shared control system appears easy to use and appears to require less cognitive work from the operator than conventional master-slave teleoperation, the system can tolerate only minimal neglect from the human operator. Consequently, it's efficiency is high but neglect tolerance is low.

Summary   In Figure 3, we plot each of the autonomy modes discussed in this section. The trend as autonomy level
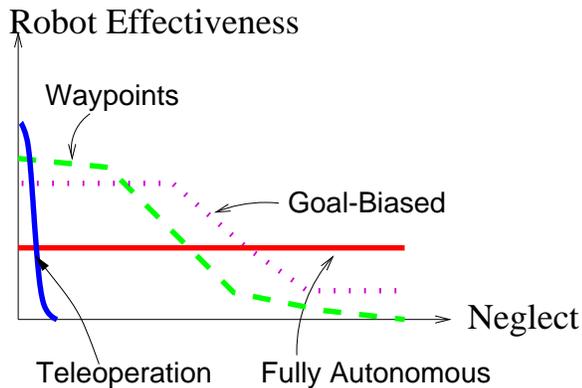

Figure 3: *Autonomy modes as a function of neglect. The teleoperation and fully autonomous levels are shown as in Figure 1. The waypoints level permit more user control and higher efficiency, but when the waypoints are exhausted then the efficiency drops off. The goal-biased autonomy allows less user control then waypoints, but can include some capability to build local maps even if neglected.*

is increased is toward flat curves situated in the middle of the efficiency axis. As operator control is increased, the curves reach higher on the efficiency axis but fall off more quickly as neglect increases.

Multi-Platform Interfaces

Our current interface runs on a laptop computer with a mouse and joystick as input devices. For systems with many robots and multiple users, this interface may be inefficient. In parallel with the development of interface-based adjustable autonomy, we have developed interfaces that include novel input-output modes which are platform independent. In this section, we discuss these interfaces and the underlying design framework.

Interface for Multiple Robots and a Single Human

One of the reasons to give a human meta-level control over the level of autonomy is to decrease human workload in human-robot interaction tasks. If workload is decreased significantly, a single human can interact with multiple robots provided that the interface facilitates such interaction. Although much work needs to be done before our interface is complete, we do have an operational interface that allows us to control a team of two robots. This interface currently includes a primitive ability to display information, placed on a service queue, about which robot needs servicing. Extending such a queue to multiple robots requires the ability for the interface agent to detect robots that have completed their assigned tasks (in the spirit of task automation), robots that have initiated behaviors that may need to be monitored (in the spirit of response automation), robots that are dormant, and robots that may be stuck. The interface agent will prioritize this queue, and robots being serviced will broadcast sensor information to help the human obtain an accurate situation awareness.

The interface will be extended to allow an operator to interrupt a robot's behaviors for a time and then allow the robot to return to its previous task. Furthermore, we will add the ability for the operator to specify a sequence of tasks for the robot to accomplish. We intend to use the idea of a goal-stack (Perzanowski *et al.* 1999) in our preliminary implementation.

X-Web Framework

XWeb is an architecture for collaborative interfaces that use many interactive modalities. Interaction is defined as the manipulation of some *shared set of information*. XWeb uses XML and a change language to represent the shared information and its modification. Multiple clients can subscribe to the information and modify it. We have developed a robust algorithm for resolving asynchronous conflicts in the information so that all clients maintain a consistent view.

In the context of human-robot teams, this shared information includes not only human-created goals and threats but also the robot status, position and information that robots have discovered. Robots behave as clients in receiving and updating the shared information.

We have created XView, which is an abstract language for representing interactive user interfaces. The heart of XView is a description of which data elements are to be presented as well as how they are labeled and organized. This representation is independent of any particular interactive modality. We have built and demonstrated complete XView clients that use a normal screen/keyboard/mouse, speech recognition and synthesis, pen-based interaction on a wall, laser-pointer interaction for shared use of a wall display, and glove-based interaction. Any of these modalities can collaborate with

any other and with any of the shared pieces of information. This allows the interaction with robotic control to adapt to any physical situation.

Validation: Experiments and Measurements

An important ingredient of human-centered robot design is validating how well the system works. In this section, we outline our proposed approach for validating the design of our robots and our interface agent.

The key concept in our approach to designing a system with adjustable autonomy is the relationship between neglect and time-delay. It is desirable to capture how much neglect a particular robot/interface can tolerate. Our approach is to conduct a series of experiments wherein a human subject manages a single robot. The subject will be asked to make the robot perform a series of tasks. In addition to accomplishing this goal, the subject will be assigned an additional task which is unrelated to controlling the robot but which requires cognitive resources. The secondary task will motivate the subject to neglect the robot, and the amount of neglect will be recorded by measuring how much of the secondary task the subject performs. We will then measure how well the subject operates the robot as a function of the level of robot autonomy given a particular level of neglect. The first experiment we are planning is one in which a human operator will use the teleoperation mode to guide the robot around the top floor of our building. The operator will perform this task while being asked to perform a cognitive task (iteratively subtract the number seven from the number 3653) while controlling the robot. This test will be repeated for two robot teleoperation techniques: conventional master-slave teleoperation and intelligent teleoperation.

Another important measurement is the amount of workload a human experiences when operating a robot. Behavioral entropy, a concept for measuring human workload in real time (Boer *et al.* 2001), is a likely candidate for measuring this workload. We are currently researching ways to measure the workload required to teleoperate the robot, add waypoints and goals, and manage the autonomy level. A second phase of this research direction is measuring how workload changes as a function of interface platform.

A Perspective on Mixed-Initiatives and Adjustable Autonomy

When humans and machines share responsibility for achieving a specific task, responsibility can be thought of as shifting between human and robot according to the timeline diagrammed in Figure 4. Initiation and termination of automation are functions of human desires and capabilities, and machine design and capabilities. An automated system must facilitate not only seamless transitions between automated and human skills, but also unambiguous assignment of authority to switch between these skills.

In this section, we discuss authority in the context of initiating and terminating automation. Within this context, we give an operational characterization of what it means to be a mixed initiative system. Throughout this section, it seems reasonable to view human-robot systems as composed of
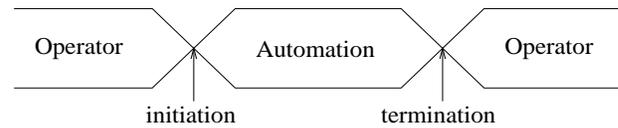

Figure 4: *Timeline of transitions between human operator and automation control (robot). (Time increases from left to right.) The timeline indicates who is given responsibility for performing a particular task. Automation responsibility begins at an* initiation *event, and ends at a* termination *event.*

three agents: a human operator, an interface agent, and a robot agent. The operator can set bounds within which the robot has authority to initiate behaviors, and the interface agent can initiate switches in these bounds.

Authority to Initiate

Sheridan identifies ten levels of automation in human-computer decision-making which range on a responsibility spectrum from the operator deciding on a task and assigning it to the computer, to the computer deciding on a task and performing the task without input from the operator (Har 1988). Based on these two extremes, automation that shares responsibility with a human operator can be broadly classified into two main categories (with examples from our system):

Task automation systems: The operator chooses to delegate a task to the automation to relieve some physical or mental burden. Setting waypoints is an example of such a system.

Response automation systems: The automation preempts human decision making and control and initiates a task to facilitate safety or efficiency. An interface agent that automatically changes the robot's autonomy level to relieve human workload is an example of such a system

The essential distinction between these two categories is how the automation is *initiated* and, more precisely, who has *authority to invoke a behavior*. In the first, the human operator initiates the automation whereas in the second, automation initiates itself.

Authority is a useful concept for identifying a mixed initiative system. *One characteristic of a mixed initiative system is that it grants a machine the authority to initiate a task*; the robot or interface agent has authority to initiate a behavior without waiting for human instruction.

Even when a human-robot system is mixed initiative, the operator may be required to switch levels of autonomy. Controlling levels of autonomy is tantamount to controlling bounds on the robot's authority. This meta-control task of controlling the autonomy level can itself be mixed initiative, as when an interface agent determines that the cognitive workload for the operator is outside of a predefined range and initiates a change in the robot's autonomy level.

Authority to Terminate

Automation will terminate if the assigned task is *completed* or if the human operator *intervenes*. Since *completion* and

intervention can both occur, it is important to design human-robot systems that help operators detect and respond to the limits of the automation. This observation leads to a second division among automation types. This division is exemplified by Sarter's automation policies (Sarter 1998):

Management by exception: When operators can easily detect and respond to the limits of automation, then the automation, once initiated, is responsible for system behavior unless and until the operator or interface detects an exception to nominal system behavior and terminates the automation. Examples of this termination policy include when a robot wanders and builds maps until the operator stops it.

Management by consent: When the limits of automated behaviors are not easily identifiable by operators or when the operator is neglecting the automation, then the automation, once initiated, must convey its limits to the operator or interface and clearly indicate when it self-terminates. This allows the operator to develop accurate and reliable expectations of automation termination by consenting to a limited scope of automated behavior. Examples of this termination policy include timed devices and systems that perform a task with a clearly identifiable state of completion (e.g. find goal x, sleep for five minutes, etc.).

The essential distinction between these two classes is how the automation is *terminated* and, more precisely, *who turns off the automation*. In the first (automation by exception), people terminate the automation whereas in the second (automation by consent) the automation terminates itself. *A second characteristic of a mixed initiative system is that the system can terminate a behavior,* even if the operator initiated the behavior.

Conclusions

Adjustable autonomy is an important concept in the human-robot-interaction community. By combining techniques from behavior-based robotics with human-centered automation, a usable interface that facilitates adjustable autonomy can be developed and applied to multi-human, multi-robot interaction.

References

Albus, J. S. 1991. Outline for a theory of intelligence. *IEEE Transactions on Systems, Man, and Cybernetics* 21(3):473–509.

Albus, J. S. 2000. 4-D/RCS reference model architecture for unmanned ground vehicles. In *Proceedings of the 2000 IEEE International Conference on Robotics and Automation*.

Arkin, R. C. 1998. *Behavior-Based Robotics*. Cambridge, Massachusetts: MIT Press.

Boer, E. R.; Futami, T.; Nakamura, T.; and Nakayama, O. 2001. Development of a steering entropy method for evaluating driver workload. In *SAE 2001 World Congress*. SAE paper #1999-01-0892.

Boyles, R. M., and Khosla, P. K. 2001. A multi-agent system for programming robots by human demonstration. *Integrated Computer-Aided Engineering* 8(1):59–67.

Breazeal, C. 1998. A motivational system for regulating human-robot interaction. In *Proceedings of the AAAI*, 54–61.

Brooks, R. A. 1986. A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation* 2:14–23.

Chuang, J.-H., and Ahuga, N. 1998. An analytically tractable potential field model of free space and its application in obstacle avoidance. *IEEE Transactions on Systems, Man, and Cybernetics — Part B: Cybernetics* 28(5):729–736.

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

Dorais, G. A.; Bonasso, R. P.; Kortenkamp, D.; Pell, B.; and Schreckenghost, D. 1998. Adjustable autonomy for human-centered autonomous systems on mars. In *Proceedings of the First International Conference of the Mars Society*.

Ferguson, G.; Allen, J. F.; and Miller, B. 1996. TRAINS-95: Towards a mixed-initiative planning assistant. In *Artificial Intelligence Planning Systems*, 70–77.

Fong, T.; Thorpe, C.; and Baur, C. 1999. Collaborative control: A robot-centric model for vehicle teleoperation. In *AAAI 1999 Spring Symposium: Agents with Adjustable Autonomy,*. Stanford, CA: AAAI.

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

Frixione, M.; Vercelli, G.; and Zaccaria, R. 1998. Dynamic diagrammatic representations for reasoning and motion control. In *Proceedings of the 1998 IEEE ISIC/CIRA/ISAS Joint Conference*, 777–782.

Goldberg, K.; Bui, S.; Chen, B.; Farzin, B.; Heitler, J.; Solomon, R.; and Smith, G. 2000. Collaborative teleoperation on the internet. In *IEEE ICRA 2000*.

Harry G. Armstrong Aerospace Medical Research Laboratory. 1988. *Engineering Data Compendium: Human Perception and Performance*. Vol. II, Section 7.3.

Inagaki, T., and Itoh, M. 1996. Trust, autonomy, and authority in human-machine systems: Situation-adaptive coordination for systems safety. In *Proc. CSEPC 1996*, 176–183.

Inagaki, T. 1995. Situation-adaptive responsibility allocation for human-centered automation. *Transactions of the Society of Instrument and Control Engineers* 31(3).

Kortenkamp, D.; Huber, E.; and Bonasso, R. P. 1996. Recognizing and interpreting gestures on a mobile robot. In *AAAI96*.

Krotkov, E.; Simmons, R.; Cozman, F.; and Koenig, S. 1996. Safeguarded teleoperation for lunar rovers: from hu-

man factors to field trials. In *IEEE Planetary Rover Technology and Systems Workshop*.

Lane, J. C.; Carignan, C. R.; and Akin, D. L. 2000. Advanced operator interface design for complex space telerobots. In *Vehicle Teleoperation Interfaces Workshop, IEEE International Conference on Robotics and Automation*.

Murphy, R. R., and Rogers, E. 1996. Cooperative assistance for remote robot supervision. *Presence* 5(2):224–240.

Murphy, R. R. 2000. *Introduction to AI Robotics*. MIT Press.

Perzanowski, D.; Schultz, A. C.; Adams, W.; and Marsh, E. 1999. Goaltracking in a natural language interface: Towards achieving adjustable autonomy. In *IEEE International Symposium on Computational Intelligence in Robotics and Automation: CIRA '99*, 208–213. Monterey, CA: IEEE Press.

Pollack, M. E.; Tsamardinos, I.; and Horty, J. F. 1999. Adjustable autonomy for a plan management agent. In *1999 AAAI Spring Symposium on Adjustable Autonomy*.

Röfer, T., and Lankenau, A. 1999. Ensuring safe obstacle avoidance in a shared-control system. In Fuertes, J. M., ed., *Proceedings of the 7th International Conference on Emergent Technologies and Factory Automation*, 1405–1414.

Rosenblatt, J. K. 1995. DAMN: A distributed architecture for mobile navigation. In *Proc. of the AAAI Spring Symp. on Lessons Learned from Implememted Software Architectures for Physical Agents*.

Sarter, N. 1998. Making coordination effortless and invisible: The exploration of automation management strategies and implementations. Presented at the 1998 CBR Workshop on Human Interaction with Automated Systems.

Scerri, P.; Pynadath, D.; and Tambe, M. 2001. Adjustable autonomy in real-world multi-agent environments. In *International Conference on Autonomous Agents*.

Sheridan, T. B. 1992. *Telerobotics, Automation, and Human Supervisory Control*. MIT Press.

Steele, F.; Thomas, G.; and Blackmon, T. 1998. An operator interface for a robot-mounted, 3d camera system: Project pioneer. In *Proceedings of the American Nuclear Society*.

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

Volpe, R. 1994. Techniques for collision prevention, impact stability, and force control by space manipulators. In Skaar, S., and Ruoff, C., eds., *Teleoperation and Robotics in Space*. AAAI Press. 175–208.

Voyles, R., and Khosla, P. 1995. Tactile gestures for human/robot interaction. In *Proc. of IEEE/RSJ Conf. on Intelligent Robots and Systems*, volume 3, 7–13.