The Peer-to-Peer Human-Robot Interaction Project

Terrence Fong,* Illah Nourbakhsh, Clayton Kunz, Lorenzo Flückiger, and John Schreiner

NASA Ames Research Center (ARC), Moffett Field, CA 94035, USA

Robert Ambrose and Robert Burridge

NASA Johnson Space Center (JSC), Houston, TX 77058, USA

Reid Simmons and Laura M. Hiatt

Carnegie Mellon University, Pittsburgh, PA 15213, USA

Alan Schultz, J. Gregory Trafton, and Magda Bugajska

Naval Research Laboratory, Washington, DC 20375, USA

Jean Scholtz

National Institute of Standards and Technology, Gaithersburg, MD 20899, USA

The Peer-to-Peer Human-Robot Interaction (P2P-HRI) project is developing techniques to improve task coordination and collaboration between human and robot partners. Our hypothesis is that peer-to-peer interaction can enable robots to collaborate in a competent, non-disruptive (i.e., natural) manner with users who have limited training, experience, or knowledge of robotics. Specifically, we believe that failures and limitations of autonomy (in planning, in execution, etc.) can be compensated for using human-robot interaction. In this paper, we present an overview of P2P-HRI, describe our development approach and discuss our evaluation methodology.

I. Introduction

A key element of NASA's Vision for Space Exploration is that humans and robots will work as partners, leveraging the capabilities of each where most useful.[1] Basic mission tasks, both in-space and on planetary surfaces, will demand close collaboration of humans and robots. But, because cost pressures and other mission constraints (e.g. risk minimization) will keep astronaut teams small, the effectiveness of human-robot interaction (HRI) will have a major impact on the productivity and performance of future missions.

The objective of the "Peer-to-Peer Human-Robot Interaction" (P2P-HRI) project is to significantly advance the state-of-the-art in HRI to facilitate sustained, affordable space exploration. Specifically, we are developing a range of HRI techniques so that humans and robots can work as partners across a range of team configurations: side-by-side, line-of-site remote, and far remote (over the horizon or even interplanetary distance). While improving both teleoperated and autonomous robot operations is important for space exploration, in this research project we focus on *peer-to-peer interaction* as a mechanism for facilitating human-robot coordination and teaming.

There are three primary components in our approach. First, we are developing a novel interaction framework called the "Human-Robot Interaction Operating System" (HRI/OS). The HRI/OS is based on the

---

*terrence.w.fong@nasa.gov

American Institute of Aeronautics and Astronautics

collaborative control model[2, 3] and is designed to enable humans and robots to engage in task-oriented dialogue and problem solving. Second, we are using computational cognitive architectures to model human behavior and make human and robot more understandable to each other, so that interaction becomes more human-compatible. Finally, we are developing a series of evaluations using research robots, analog environments, exploration relevant tasks (e.g., structural assembly), and quantitative HRI performance metrics.

In P2P-HRI, our work focuses on supporting tasks that are essential for basic mission operations and that are well-defined and narrow in scope. These include: shelter and work hangar construction, piping assembly and inspection, pressure vessel construction, habitat inspection, and *in-situ* resource collection and transport. Each of these "operational tasks" demands effective human-robot teamwork and requires extensive interaction between team members.[4]

An efficient and pragmatic approach to performing operational tasks is to first specify a high-level set of operations that can be performed by humans and robots working in parallel and then use interaction to resolve problems that arise during execution. This mode of operation has numerous benefits: it is similar to how human construction and maintenance crews operate; it significantly reduces the need for fine-grained contingency planning and resource scheduling; it will work with any level of robot autonomy; and it does not require the human to continuously engage in robot teleoperation or supervision.

II.   Peer-to-peer HRI

Conventional human-robot interaction is limited to "master-slave" commanding (i.e., goal/task specification) and monitoring (e.g., of status information). More precisely, the interaction model is essentially one-way: the human "speaks" and the robot "listens" (perhaps asking for clarification). As a result, system performance is strictly bound to the operator's skill and the quality of the user interface. To improve system capability, increase flexibility, and create synergy, human-robot communication needs to be richer and occur in both directions.

Our approach is to develop an interaction model in which humans and robots communicate as peers. Specifically, we are building a dialogue system that allows robots to ask questions of the human when necessary (urgent) and appropriate (human at a work breakpoint), so that robots are able to obtain human assistance with cognition and perception tasks. Two key benefits of this system are that it: (1) allows humans and robots to communicate and coordinate their actions and (2) provides interaction support so that humans and robots can quickly respond and help the other (human or robot) resolve issues as they arise.

A key challenge is enabling robots to perform tasks on their own, but giving them the ability to ask for (and make use of) human expertise and assistance when necessary. Another challenge is enabling robots to understand task-oriented commands in the same way that human teammates do. For example, human work crews routinely use spatial references (e.g., "move *that* panel to *my* left") when performing work.

III.   Human-Robot Interaction Operating System

In order for humans and robots to work effectively together, they need to be able to clearly converse about goals, abilities, plans and achievements.[5] Such communication is especially required for humans and robots to jointly solve problems or when situations exceed autonomous capabilities.

To address this need, the HRI/OS provides a structured software framework and set of core interaction services for building human-robot teams. We have designed the HRI/OS to support a variety of multi-modal and perceptual interfaces (including shared workspaces and handhelds) and to facilitate integration of third-party UI's and robots through an extensible API.

American Institute of Aeronautics and Astronautics

A. Related work

The HRI/OS is similar in some respects to interaction *infrastructures* that support non-traditional human-computer interaction.[6–9] In particular, the HRI/OS provides a variety of infrastructure services (event/data distribution, delegation, etc.) to distributed and mixed teams. The HRI/OS differs from infrastructures because the "devices" (i.e., robots) have physical embodiment and can move and act in the real world. As a result, the HRI/OS includes dialogue support services (e.g., spatial perspective taking) not normally found in infrastructures.

The HRI/OS is also related to a number of recent HRI architectures,[10–13] all of which are designed to facilitate task performance by human-robot teams. The HRI/OS, however, differs from these architectures in three ways. First, it is designed to seamlessly support human-robot collaboration across multiple spatial ranges. Second, the HRI/OS includes a task executive, which provides loose coordination between humans and robots working in parallel. Finally, it allows robot control authority to pass between different users (i.e., no operator has exclusive "ownership" of a robot). This improves flexibility because which robot(s) supports (i.e., works with) which human(s), and vice versa, can vary dynamically based on the situation.

B. Teamwork Model

With the HRI/OS, humans and robots work on tasks independently of each other. Tasks are delegated by a task executive, which assigns work to agents (i.e., human or robot) it believes capable of satisfying the task, and which are not currently performing other work. Agents are expected to take care of their own planning, execution, and monitoring. Once that task has been assigned to an agent, that agent is responsible for its execution and eventual completion. If it encounters a problem, however, an agent will first try to resolve it through dialogue, before reporting failure.

When an agent makes a request for help, the request is delegated to another agent (human, robot, or software) that is capable of providing assistance. Satisfying a "help" request typically requires communication. A human, for example, may request a tool or a service that can be provided by a robot, while a robot might require a human's image processing or reasoning ability. The HRI/OS provides dialogue services such as text-to-speech, speech recognition, spatial relationship context resolution (as a step in natural language processing), and contextual data transport. Embodied agents can make use of these services to facilitate communication with each other, with the ultimate goal of resolving a problem with their own assigned tasks.

If a robot is interrupted, it suspends its task before addressing the reason for its interruption. In the case where a human has requested the assistance of the robot, the robot will provide the needed assistance and then resume its previous task. With our teamwork model, as with purely human interactions, only one human is able to interrupt the robot at a time. A human cannot, for example, interrupt a robot while it is already providing assistance to another human. Instead, the human must wait until the robot becomes available.

In our system, human-robot dialogue is coordinated with an interaction manager. A robot can specify particular user interface modalities (e.g. a graphical user interface or a speech interface) as part of its request for dialogue. In general, humans are selected solely based on their published domain(s) of expertise. Agents working on behalf of the human can display robot dialogue using whichever interface modality is most appropriate for the human's situation, taking into consideration the available display, the human's workload, etc.

In order to take the best possible advantage of the particular skills of humans and of robots, it is important that robots be able to reason about how to communicate with humans for maximum effect. Consequently, the HRI/OS provides cognitive models and spatial reasoning capabilities to facilitate the use of natural, spatial language (e.g., "move the light to the left of the box").[14–16]

American Institute of Aeronautics and Astronautics

C. Implementation

The HRI/OS is implemented as a collection of agents using the Open Agent Architecture (OAA).[17] Embodied agents describe their skills at a coarse level, rather than with the detail typically used by robot planners. Software agents are designed to provide a single capability, rather than a large set of related capabilities. For example, an agent that tracks objects may provide pose information without handling coordinate frame transformations, which could be provided by other supporting agents. This design approach helps improve flexibility while reducing system brittleness.
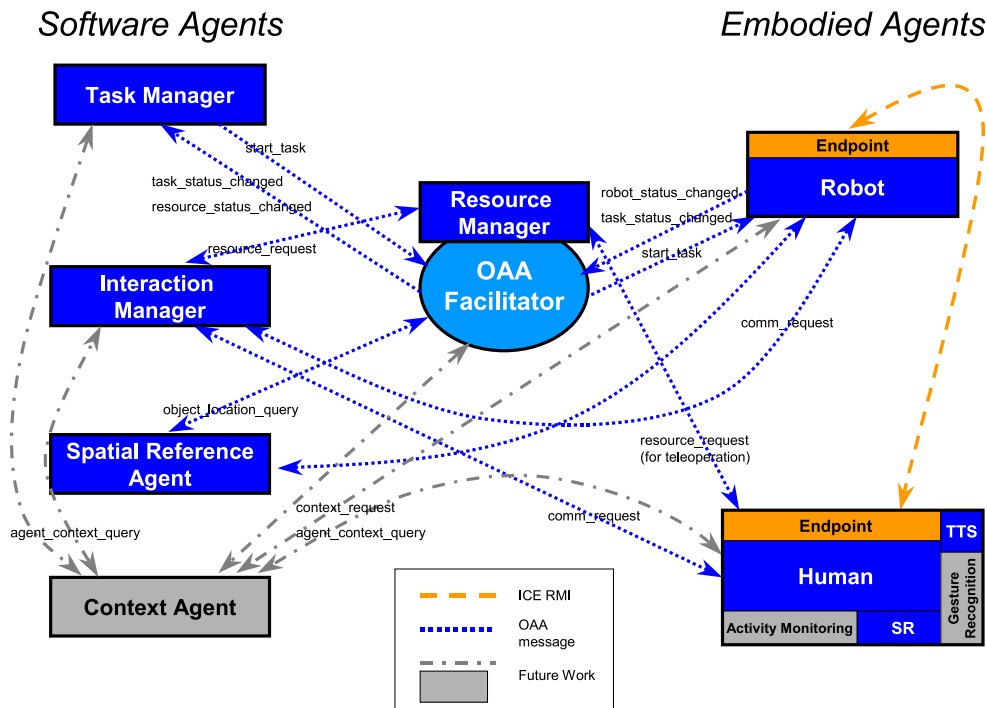

Figure 1. The Human-Robot Interaction Operating System (HRI/OS) is an agent-based system.

Figure 1 shows the primary components in the HRI/OS. Software and embodied agents communicate via OAA messages, which are delegated and routed via a central OAA facilitator. Direct, point-to-point communication (used primarily to transport non-text dialogue, such as images) is performed using the "ICE" object-oriented middleware.[18] The following sections describe the major agents that make up the core of the HRI/OS.

1. Task Manager

The Task Manager (TM) is a task executive that coordinates execution of well-defined, operational tasks by one or more agents. The TM also provides for limited recovery in the face of unforeseen task failures. For example, if any given task fails, the TM automatically respawns another instance. The TM is written in the Task Description Language (TDL), an extension of C++ that allows for principled and managed task execution, coordination and management.[19]

2. Resource Manager

The Resource Manager (RM) works with the underlying agent system to determine which agent is best suited for performing a given task or handling dialogue. As such, it receives all requests to be delegated and may reprioritize the list of agents that will be consulted to perform a task or answer a request. The RM is designed to consider numerous factors (including the relative positions of embodied agents, their workload, and statistics such as fuel level) in order to refine the decision about which agent should be assigned a task.

The RM also supports switching of robot control authority between *system operation* (execution of tasks assigned by the Task Manager) and *interrupt servicing* (i.e., temporary use of a robot). For example, a user can take temporary "ownership" of a robot (e.g., so that the user can teleoperate the robot for a specific use) by making a request to the RM. When the robot reaches a breakpoint, the RM will then grant "ownership" to the user. Multiple requests are pushed onto an interrupt queue, which allows multiple users to share "ownership" of a robot.

3. Interaction Manager

The Interaction Manager (IM) coordinates dialogue between humans and robots. Whenever an embodied agent needs to communicate with another agent, it contacts the IM, which works with the RM to generate a list of agents that are able to handle the communication request. Once it has this list, the IM informs the best-matching agent that it has an interaction request.

If this agent is a robot, then it immediately handles the request. If it is a human, however, he is notified that a message is waiting for him, and the IM waits for acknowledgment before passing along the robot's request. If the human does not respond in a reasonable amount of time, the IM iterates through the list of agents returned by the RM to see if another one is available. If none are, it then notifies the requesting agent that the request cannot be handled.

Whenever an agent responds to a request, it becomes the responsibility of the two parties to set up and handle the rest of the communication between them. The IM will only get involved further in the dialogue if the original requesting agent notifies the IM that the communication has failed. In this case, the IM will try to connect the agent to someone else in order to process its request.

4. Dialogue Agents

There are a number of supporting software agents in the HRI/OS that are used to enable and facilitate spoken dialogue. The *text-to-speech* agent (TTS) allows the robots' responses to humans to be verbalized for the humans to hear; the *speech recognizer* agent (SR) allows the humans to verbally respond to the robots; and the *spatial reference agent* (SRA) allows the robots to understand utterances such as "on your left" and "in front of me."

5. Robot Agents

Robot Agents (RA's) provide an interface between robot controllers and the HRI/OS. RA's are responsible for handling messages and requests from other agents, managing and executing their own atomic tasks, and enabling communication and interaction with others. The RA provides a programming interface for integrating robots into the HRI/OS. This interface includes support for registering robot capabilities with the Resource Manager, for broadcasting event and state information, for dispatching service requests and problem solving queries, and for processing query responses.

IV.   Computational Cognitive Architectures

For a robot to work side by side with an astronaut, collaborating in a shared workspace, the robot must be able to do something that humans do naturally: understand how another person perceives space and the

relative positions of objects around them — the ability to see things from another person's point of view. To give robots this ability, we are building computational cognitive models (CCMs) of certain high-level cognitive skills that humans possess and that are relevant to collaborative tasks. We then use these models as reasoning mechanisms for our robots.

Why do we propose using CCMs as opposed to more traditional programming paradigms for robots? We believe that by giving the robots similar representations and reasoning mechanisms to those used by humans, we will build robots that act in a way that is more compatible with humans.

In the P2P-HRI project, we are developing computational, cognitive, and linguistic models that can deal with spatial perspective-taking and frames of reference. Issues include dealing with constantly changing frames of reference, changes in spatial perspective, and maintaining common ground among team members. Perspective taking, in particular, is a critical cognitive ability for humans, particularly when they want to collaborate.

A. Spatial perspective in space

To determine just how important perspective and frames of reference were in collaborative tasks in shared space, we analyzed a series of tapes of two astronauts and a ground controller training in the Neutral Buoyancy Lab at NASA JSC for an assembly task for Space Station mission 9A. We performed a protocol analysis of several hours of these tapes focusing on the use of spatial language and commands from one person to another. We found that the astronauts changed their frame of reference approximately every other utterance.

As an example of how prevalent these changes in frame of reference are, consider this following utterance from ground control:

> ...if you come *straight down* from where you are, uh, and uh, kind of peek *down under* the rail on the *nadir side*, by *your right hand*, almost *straight nadir*, you should see the...

Here we see five changes in frame of reference (highlighted in italics) in a single sentence! These rates in the change of reference are consistent with work by Franklin, Tversky and Coon.[20] In addition, we found that the astronauts had to take other perspectives, or forced others to take their perspective, about 25% of the time.[16]

Obviously, the ability to handle changing frames of reference and being able to understand spatial perspective will be a critical skill for robots such as NASA's Robonaut[21] and, we would argue, any other robotic system that needs to communicate with people in spatial contexts (i.e., any construction task, direction giving, etc.).

B. Models of perspective taking

Consider the following task, as illustrated in Figure 2. An astronaut and his robotic assistant are working together to assemble a structure in a shared space. The human, who can see one wrench, says to the robot, "Pass me the wrench." Meanwhile, from the robot's point of view, two identical wrenches are visible, while the human has a partially occluded view and can only see one wrench. What should the
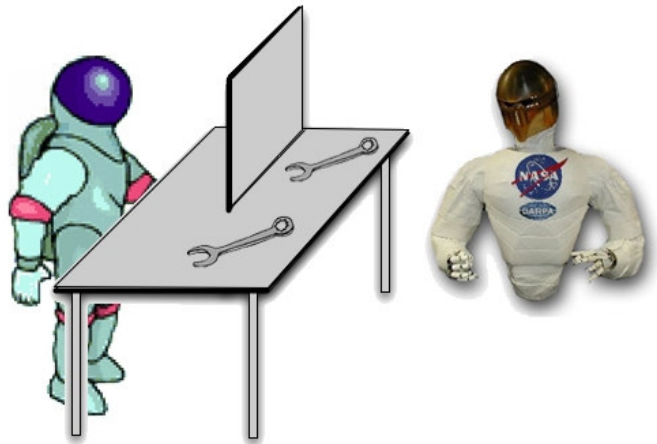
Figure 2. Perspective taking example. The astronaut can only see one wrench. The robot can see both wrenches. The astronaut asks the robot to "Pass me the wrench".

robot do? Evidence suggests that humans, in similar situations, will pass the wrench that they know the other human can see[22] since this is a jointly salient feature.

We have developed several models of perspective-taking using different cognitive modeling systems.[14–16] Currently, we are building a model of perspective-taking that can handle the above scenario in a general sense. The approach uses the Java version of the ACT-R[23] cognitive architecture system, jACT-R, to model frames of reference and perspective-taking.

In essence, whenever the model wishes to take the perspective of a person, it performs the equivalent of a mental simulation, virtually placing the robot in the human's position in order to reason about the human's perspective (e.g., what objects the human can see). This mental simulation is accomplished using the Stage component of the Player/Stage robot simulation system.[24]

Specifically, we model the current world in the Stage simulation using information obtained from the robot (e.g,. current pose) and other sensors (e.g., object trackers). When we need to understand the human's perspective, or resolve a frame of reference, we "imagine" a scene from an appropriate perspective and then resolve any ambiguous references, as shown in Figure 3.
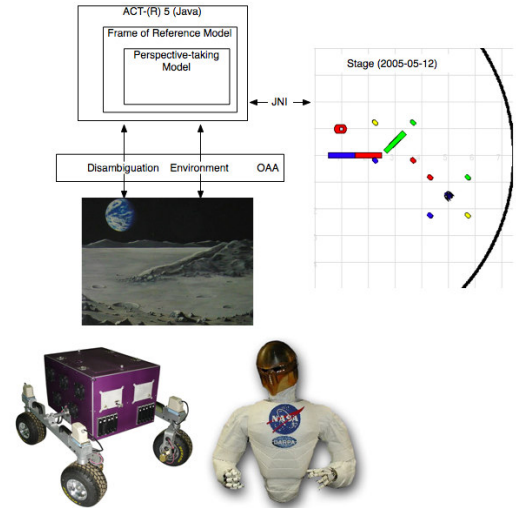

Figure 3. Computational cognitive model for perspective taking and spatial referencing.

V.   Evaluation

A.   HRI metrics

The development of human-robot systems needs to be based on proven theories and guidelines. Although metrics from other fields (human-computer interaction, human factors, etc.) can be applied to satisfy specific needs, HRI has characteristics (e.g. physical interaction by an embodied robot) that set it apart. Thus we need to develop metrics and evaluation procedures that are appropriate for HRI.

Our approach focuses on assessing our overall progress in improving the productivity of EVA with human-robot teams. Thus, we have chosen metrics to evaluate task effectiveness, teamwork efficiency, and astronaut workload. As our research progresses, we intend to implement techniques for assessing situational awareness of both humans and robots.

To develop our evaluation method, we first defined a hierarchy of metrics and ground truth data to be collected (Table 1). For each of these metrics we then identified lower level measures that could potentially inform these metrics. For example, to assess teamwork, we can look at measures such as the number and types of problems that occur during a task and determine how these problems impact overall success.

Table 1. Top level metrics

Metric	Measured by
Effectiveness	Task success
Teamwork efficiency	Analysis of breakdowns, subjective questionnaire data, time measures
Astronaut workload	NASA TLX[25]

American Institute of Aeronautics and Astronautics

Efficient teamwork measures include several workflow measures:

- Number of problems (breakdowns) encountered during the task

- The percentage of breakdowns detected by the robot and the percentage detected by the human

- The level(s) of autonomy used to perform the tasks and a classification of events that necessitate changes in autonomy level.

Breakdowns are further divided into four categories: *robot-task*, *robot-tool*, *human-robot*, and *robot-human*. In the robot-tool category, we focus on the ability of the robot to use the tool correctly, assuming the tool is adequate for the task. In our current work, an appropriate tool is always used. In future work, however, when novel tasks are presented, this may not be the case. For the other three categories, we will examine lower level dialogue measures and system logs. Breakdown classifications are summarized in Table 2.

Table 2. Breakdown classifications

Category	Representative breakdowns
Robot-task	Number of tasks robot does not understand
	Number of tasks for which robot does not have enough information
Robot-tool	Tool failure
	Robot does not understand or use the tool correctly
	Tool is inadequate for the job
Human-robot	Insufficient information
	Message unclear
	Message not received
	Message sent to inappropriate robot
Robot-human	Message not processed in a timely manner
	Appropriate context not given
	Message sent to inappropriate person

We will use time measures to determine if there is any relationship between workload and efficient teamwork. The time measures that we will calculate include: time to complete the task, percentage of time the robot worked alone, percentage of time the human and the robot interact, percentage of time the human worked alone, and percentage of time the human is able to provide assistance remotely (e.g., from inside the habitat).

To examine breakdowns that result from communication issues, we will use a variety of dialogue measures. These include: number of dialogue turns, percentage of turns used by human and by robot, percentage of turns that contain content, percentage of turns needed for clarification (how, what, where, who, or when), percentage of inappropriate messages, and percentage of communications not handled in time.

By design, these lower-level measures can inform multiple higher-level metrics.

B. Methodology

During the P2P-HRI project, we plan to enact several use cases with our human-robot team. During each of these scenarios we will collect both video and audio data, log files from the system, body position of the astronauts, position of the robots, and position of the tools. In addition, the astronauts will be given the

American Institute of Aeronautics and Astronautics

NASA TLX questionnaire to assess workload and a questionnaire to assess their opinion of the efficiency of the teamwork. Finally, observers will be present to make notes of any unexpected events or breakdowns that occur. The early scenarios that we enact will be scripted so we will be able to easily note deviations that occur.

We will then divide the overall scenario into smaller tasks. Example tasks might be a robot performing a weld task, a robot inspecting a weld, or an astronaut indicating a location for a robot to weld. We will calculate the measures per task and will also collect our higher level metrics of success, workflow questionnaire, and workload by task. Once we have calculated these measures, we will determine how the lower level measures correlate with our observations of breakdowns, the astronauts' ratings of workflow, and the NASA TLX scores for workload. We will do this over the range of primitive tasks. This should enable us to identify the best measures, i.e., those that correlate well and are lowest cost to collect.

Dialogue measures, for example, are much more expensive to compute than time measures. Workflow measures can be obtained in real-time by observers, but necessitate several observers who have been trained in the coding scheme. Log files of the actions of the robots, including dialogue interactions, are inexpensive to calculate as they can be processed automatically. As we conduct evaluations with more scenarios we will follow the same procedure. That is, we will collect multiple measures to determine those that allow us to discriminate tasks that have successful human-robot interactions from those that do not compared to our ground truth measures.

C.   Initial use case

During Fall 2005, we will study a use case that centers on seam welding and inspection by a human-robot team. Seam welding is a task that will be required for building and maintaining a variety of structures on planetary surfaces.[26–29] For example, linear welds might be used to construct pressure vessels, work hangers, and emergency shelters too large to raise into space in one piece.

Figure 4.   Left, the K-10 rover (NASA Ames) is a low-cost rover designed to operate at human interaction speeds; right, Robonaut-B (NASA JSC) is a mobile manipulation system attached to a Segway RMP differential-drive base.

In this study, three humans (two in EVA and one in a habitat) will work with two robots (Figure 4), the ARC K-10 and JSC Robonaut-B, to weld* panels to a truss. The humans will act as *master welders* and provide initial panel mounts (e.g, spot welds). The robots will perform two types of tasks. Robonaut-B

---

*Because it is not our goal to study and/or improve robotic welding, actual welding will not be performed. Instead, we plan to use a "mock welding" process (e.g., spray painting for seaming).

American Institute of Aeronautics and Astronautics

will work as a *junior welder* and seam weld once the panels are placed. K-10 will work as a *seam inspector* and inspect the quality of the welds. Humans and robots will work in parallel, supporting each other as necessary.

This work scenario provides numerous opportunities for dynamic and flexible human-robot interaction. For example, a variety of communication acts are useful: human generated commands, questions from the robots to the human, etc. Additionally, the human may remotely interact with the robots (e.g., he may deploy the inspection robots via a control room inside a habitat) as well as work side-by-side with others (e.g., leading welders to a new site and showing them where to weld).

VI.   Conclusion

We believe that peer-to-peer HRI will enable more effective and productive human-robot teams for space exploration. In particular, we believe that tools such as the HRI/OS and computational cognitive models will enable humans and robots to work efficiently and effectively together, regardless of spatial distribution, communication channel, and user interface.

In more general terms, we expect that peer-to-peer HRI will be appropriate whenever humans and intelligent systems must collaborate in order to execute complex tasks such as inspection and maintenance. It is also probable that in low-bandwidth situations and scenarios (e.g., lunar robots operated from terrestrial ground control) that the interaction technology developed by P2P-HRI will help reduce data transmission demands.

Thus, during the next few years, our goal will be to apply peer-to-peer HRI directly to a wide variety of mission systems, including the Crew Exploration Vehicle (CEV) and lunar landers.

Acknowledgments

References

[1] NASA, "The vision for space exploration," Tech. Rep. NP-2004-01-334-HQ, NASA, Washington, DC, 2004.

[2] Fong, T., Thorpe, C., and Baur, C., "Collaboration, dialogue, and human-robot interaction," *Proc. 10th International Symposium on Robotics Research*, Springer, 2001.

[3] Fong, T., Thorpe, C., and Baur, C., "Multi-robot remote driving with collaborative control," *IEEE Transactions on Industrial Electronics*, Vol. 50, No. 4, 2003.

[4] Fong, T. and Nourbakhsh, I., "Interaction challenges in human-robot space exploration," *ACM Interactions*, Vol. 12, No. 2, 2005, pp. 42–45.

[5] Fong, T. and Nourbakhsh, I., "Peer-to-peer human-robot interaction for space exploration," *Proc. AAAI Fall Symposium: The Intersection of Cognitive Science and Robotics: From Interfaces to Intelligence*, Oct 2004.

[6] Gajos, K., Weisman, L., and Shrobe, H., "Design principles for resource management systems for intelligent spaces," *Proc. Second International Workshop on Self-Adaptive Software*, 2001.

[7] Roman, M., Hess, C., et al., "Gaia: a middleware infrastructure to enable active spaces," *IEEE Pervasive Computing*, Vol. 1, No. 4, 2002, pp. 74–83.

[8] Sousa, J. and Garlan, D., "Aura: an architectural framework for user mobility in ubiquitous computing environments," *Proc. IEEE/IFIP Conference on Software Architecture*, 2002.

[9] Winograd, T., *HCI in the New Millennium*, chap. Interaction spaces for 21st century computing, Addison Wesley, 2001.

[10] Bradshaw, J. et al., *Software Agents*, chap. KAoS: Toward an industrial-strength open agent architecture, MIT Press, 1997.

[11] Clancey, W. et al., "Automating capcom using mobile agents and robotic assistants," *Proc. AIAA 1st Space Exploration Conference*, 2005.

[12]Martin, C., Schreckenghost, D., et al., "An environment for distributed collaboration among humans and software agents," *Proc. 2nd International Conference on Autonomous Agents and Multi-Agent Systems*, 2003.

[13]Tews, A., Mataric, M., and Sukhatme, G., "A scalable approach to human-robot interaction," *Proc. IEEE International Conference on Robotics and Automation*, 2003.

[14]Cassimatis, N., Trafton, J., Bugajska, M., and Schultz, A., "Integrating cognition, perception, and action through mental simulation in robots," *Robotics and Autonomous Systems*, Vol. 49, No. 1-2, Nov 2004, pp. 13–23.

[15]Hiatt, L., Trafton, J., Harrison, A., and Schultz, A., "A cognitive model for spatial perspective taking," *Proc. 6th International Conference on Cognitive Modelling*, 2004.

[16]Trafton, J., Cassimatis, N., Bugajska, M., Brock, D., Mintz, F., and Schultz, A., "Enabling effective human-robot interaction using perspective-taking in robots," *IEEE Trans. on Systems, Man and Cybernetics, Part A*, Vol. 49, No. 4, July 2005, pp. 460–470.

[17]Cheyer, A. and Martin, D., "The Open Agent Architecture," *Journal of Autonomous Agents and Multi-Agent Systems*, Vol. 4, No. 1, March 2001, pp. 143–148.

[18]Henning, M., "A New Approach to Object-Oriented Middleware," *IEEE Internet Computing*, Vol. 8, No. 1, 2004, pp. 66–75.

[19]Simmons, R. and Apfelbaum, D., "A task description language for robot control," *Proc. Conference on Intelligent Robots and Systems*, 1998.

[20]Franklin, N., Tversky, B., and Coon, V., "Switching points of view in spatial mental models," *Memory and Cognition*, Vol. 20, No. 5, 1992, pp. 507–518.

[21]Ambrose, R., Aldridge, H., et al., "ROBONAUT: NASA's Space Humanoid," *IEEE Intelligent Systems Journal*, Aug 2000.

[22]Clark, H., *Using language*, Cambridge University Press, 1996.

[23]Anderson, J. and Lebiere, C., *Atomic components of thought*, Erlbaum, Mahwah, NJ, 1988.

[24]Gerkey, B., Vaughan, R., and Howard, A., "Player/Stage Project: Tools for multi-robot and distributed sensor systems," *Proc. 11th International Conference on Advanced Robotics*, 2003, pp. 317–323.

[25]Hart, S. and Staveland, L., *Human Mental Workload*, chap. Development of NASA TLX (Task Load Index): results of empirical and theoretical research, North-Holland Elsevier Science, 1988.

[26]Johnson, S. and Leonard, R., "Evolution of concepts for lunar bases," *Lunar Bases and Space Activities of the 21st Century, Proc. Lunar and Planetary Institute*, 1985, pp. 47–56.

[27]Reynerson, C., "Design considerations for remotely operated welding in space: Task definition and visual weld monitoring experiment," Tech. Rep. M.S. Thesis, Dept. of Ocean Engineering, MIT, Cambridge, MA, 1993.

[28]Reynerson, D. and Reynerson, C., "Design considerations for remotely operated welding processes," *Proc. 10th International Conference on Robotics and Remote Systems for Hazardous Environments*, Mar 2004.

[29]Russell, C. et al., "Considerations of metal joining processes for space fabrication, construction and repair," *Proc. SAMPE Technical Conference*, 1991.