I want what you've got: Cross platform usability and human-robot interaction assessment.

Julie L. Marble, Ph.D.*, Douglas A. Few, & David J. Bruemmer
Human, Robotic, & Remote Systems
Idaho National Engineering and Environmental Laboratory
Idaho Falls, ID
{marbjl, fewda, bruedj}@inel.gov

ABSTRACT: Human-robot interaction is a subtle aspect of design that must be investigated during the creation of both the human-robot interface and robot behaviors if the team is to effectively meet the complexities of the real world. The goal of testing is not only to determine whether the system can achieve the goal for which is has been designed, but more importantly, the purpose of testing is for the system creators to learn how humans and robots can, will, and should work together in the complex, real world to achieve their goals. The lack of human-centered robot interface design, the rigidity of sensor configuration, and the platform-specific nature of research robot development environments are a few factors preventing robotic solutions from reaching functional utility in real word environments. Often the difficult engineering challenges of implementing adroit reactive behavior, reliable communication, trustworthy autonomy, and usable interfaces are overlooked in favor of far-reaching research aims. The result is that many robotic systems never reach a level of functional utility necessary even to evaluate the efficacy of the basic research, much less result in a system that can be used in a critical, real-world environment. This paper discusses the challenges inherent to the conduct of a recent human factors testing on a variable autonomy control architecture using a simplified search and rescue scenario within a complex, real-world environment. It discusses the need to compare behaviors, architectures, and interfaces within a structured environment that contains challenging real-world tasks, and the implications for system acceptance and trust of autonomous robotic systems for how humans and robots interact in true interactive teams.

KEYWORDS: *robots, human-robot interaction, cross-platform compatibility, usability, mixed-initiative.*

1. INTRODUCTION

True human-robot teaming requires that all team members be aware of the goal, and work toward that goal with or without input from the other members. For the control architecture of a robotic system to be useful and usable, it must allow the human team member to build trust in the system, regardless of the intelligence inherent in the robotic system. System trust can only be enhanced when the system is designed to meet actual users' needs, abilities, and limitations within the constraints of the task; such design requires true user testing, not simple designer evaluation.

Often, research in human-robot interaction has assumed that the human will always be the ultimate decision maker and the goal seeker, while the robot is seen as a tool that is not aware of the task goal. As such, robots are substitute task performers under some level of supervision by the human. The robot may be designed to meet the physical requirements of the task, but the complexity of the task also yields requirements and constraints on human interaction with the system that are often neglected altogether. When the needs of the human are considered, these needs have not been a focus of the design process, because it has not been possible to compare architecture-architecture or interface-interface between systems because differences in architectures frequently meant differences in interfaces.

Human-robot interaction is a subtle aspect of design that must be investigated during the creation of both the human-robot interface and robot behaviors if the team is to effectively meet the complexities of the real world. Usability testing is one tool available to help roboticists design systems to meet these goals. The goal of testing is not only to determine whether the system can achieve the goal for which is has been designed, but more importantly, the purpose of testing is for the system creators to learn how humans and robots can, will, and should work together in the complex, real world to achieve their goals. The presence of reliable, transparent to the user robot systems in the field of Human Robot Interactions (HRI) is lacking. This lack of reliable technology has resulted in the majority of HRI studies to be completed in simulation. In the few cases where real robots systems have been used the lack of control in robot behavior implementation from one system to another has reduced experiment results to subjective observations. This paper discusses the challenges inherent to the conduct of human factors tests of robotic control architectures within a complex, real-world environment. It discusses the need to compare behaviors, architectures, and interfaces within a structured environment that contains challenging real-world tasks, and the implications for system acceptance and trust of autonomous robotic systems for how humans and robots interact in true interactive teams.

Figure 1. The ATRV Jr. and component sensors

2. HUMAN CENTERED INTERFACE DESIGN AND TESTING

The lack of human-centered robot interface design, the rigidity of sensor configuration, and the platform-specific nature of research robot development environments are a few factors preventing robotic solutions from reaching functional utility in real word environments. Often the difficult engineering challenges of implementing adroit reactive behavior, reliable communication, trustworthy autonomy, and usable interfaces are overlooked in favor of far-reaching research aims. The result is that many robotic systems never reach a level of functional utility necessary even to evaluate the efficacy of the basic research, much less result in a system that can be used in a critical, real-world environment.

Yanco, Drury, and Scholtz [2] have identified two major shortcomings in prior HRI evaluations. First, HRI evaluations typically fail to test the intended user of the system; rather, the designers of the system are also the test users. Such evaluation is flawed, because system designers possess a much higher system understanding and proficiency than do the actual users of that system. In short, system designers have a unique "mental map" of the interface that is based on how the system works --- understanding that the typical end user may never need or want to derive. For example, most people do not fully comprehend *how* the engine in their cars works; however, these same people may be very highly capable drivers. Thus, due to their specialized insight, designers represent an upper bound of expected performance, and evaluations may fail to identify the difficulties that an actual HRI user might experience.

The second shortcoming noted by Yanco et al. [2] is that HRI evaluations are commonly informal, precluding careful empirical control. As a consequence, most HRI evaluations fail to provide objective or conclusive results. Yanco et al. do not dismiss offhand the value of current HRI evaluation methods. Rather, they aim to complement existing HRI evaluation methods by pointing the way toward more effective evaluation.

In order to realize the broad use of robotic systems in hazardous environments, shortcomings in robot interfaces, control system configurability, and overall usability must be addressed. The Idaho National Engineering and Environmental Laboratory (INEEL) has made a concerted effort to build a foundation of well-engineered communication, perception and autonomous behavior, robust to changing, unstructured environments and which could be reused across different robot geometries and sensors.[1]

At present, the INEEL has performed several formal and semi-formal usability tests of our HRI and behavior control architecture. These studies are discussed more fully elsewhere see [3, 4]. These tests have examined the role of prior experience with remote systems on usability and interaction with the system, the effects of age gender, and more simply, users' expectations for system performance and robot behaviors.

As suggested by Yanco et al. [2], we avoided evaluating the interface with system designers or seasoned operators. Instead, we enlisted novice users of robotic systems in our evaluation. First, we are designing for multiple applications, including countermine operations, remote characterization of high radiation environments and military reconnaissance. We believe that by opting for novice users, we maximized both the relevance of our study to multiple applications and our evaluation's sensitivity to interface shortcomings. Second, because we believe that incorporation of autonomous robots into these types of tasks will inherently change not only the structure of the task, but the humans' role in these tasks, we must design the HRI to support novice users. For example, use of autonomous robots may eliminate the need for humans to enter high radiation environments; therefore, the rules that keep the human safe in the high rad environment may no longer apply. If we design a system that plays by rules that serve no purpose, we limit the system. Evaluation with novice users does not preclude the necessity of further evaluation with the actual target users when the system is devoted to a single task domain. An evaluation of novice users does, nonetheless, provide a baseline performance measure using a greater number of participants than would otherwise be possible.

With this distinction aside, we believe that we have much more in common with Yanco et al. than not. Like Yanco et al., we firmly believe that robotic systems must be designed with as much environmental, and task realism as possible. Furthermore, we also believe that formal, iterative system testing is the only route that ensures a system the supports the capabilities and needs of the users. In previous work, we proposed the following set of guidelines for usability testing.

2.1 Guidelines for testing usability of a human-robot interface or architecture [5]

1. Simplification of the environment to allow problem solution can corrupt the ability of the human-robot system to achieve its goals; therefore, the system must be tested in real world conditions to determine if it accurately meets these real world needs.
2. Robotic systems will be effective only if the behaviors they use to achieve task goals are comprehensible and predictable to the human team members; therefore, system design must assess how the human will work with the system.
3. The test environment must reflect the complexities of the real-world environment in which it will be used.
4. The test environment must incorporate uncertainty regarding the environment or the goal that will be seen in the true task.
5. The task cannot be designed to exploit the capabilities of the robot; rather the robot's capabilities must be designed to exploit aspects of the environment and the task should emphasize the complexities encountered in the real world.
6. To accurately reflect the complexity of the task, testing must involve users who are similar to those who will put the system to actual use, not only those operators who are most familiar with the control architecture.
7. Testing must incorporate the need for an operator to maintain a level of awareness in more than one environment.
8. Issues of teaming and the ability of the human to trust the robot enough for effective teaming must be addressed and assessed in the testing.
9. Tasks constraints may dynamically change with the incorporation of human-robot teams. However, these constraints may still shape how the human expects the system to behave.

3. CROSS-PLATFORM COMPATIBILITY

The above guidelines, as are apparent from reading, focus on usability testing of a single system. They do not provide guidance for making comparisons *between* HRIs and control architectures. Comparison of the sufficiency of HRIs between designers, or the comparison of control architectures between platforms, is inherently complex. How is one to assess the performance of one control architecture separate from the advantages yielded by the mobile platform itself? Did failure occur because the interface did not provide the user sufficient information to maintain situation awareness or did failure occur simply because the platform selected is not agile enough or the size is incompatible for the environment or task?

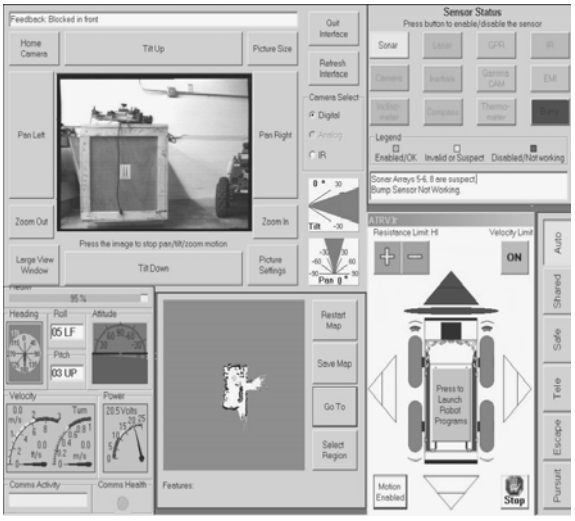

Figure 2. Human-robot interface.

The INEEL has focused on the need to increase human-centered design and usability through an emphasis on consistency, simplicity, and low bandwidth communication. A human-centered approach requires that robot interface, behaviors and perceptions be designed such that the robot's particular characteristics are transparent to the user. To support this aim, the INEEL has developed a control system that uses a level of middleware abstraction to support robust perception and autonomous behavior for a wide variety of robotic systems. The abstractions allow for the easy addition of new robot systems as well as providing a method for developing behaviors on one platform that transfer with no source code changes to all other platforms, despite differences in size, bounding shape, or sensor configuration.

Recently the INEEL team incorporated several major systemic changes to the robot control architecture. The first was to completely abstract all data and function calls with respect to robot specific geometry, sensor suite, and development environment from the robot control architecture. Doing so required funneling all robot sensor data into standard constructs. The constructs contain robot sensor information in a form generic to ground vehicles enabling the easy addition of future platforms into the INEEL architecture. Additionally all evidence of proprietary robot development environments (i.e., iRobot's 'Mobility,' ActivMedia's ARIA) have been removed from the behavioral content of the control architecture. The combination of these efforts resulted in a system capable of being transferred from one robot to another with out the need of porting or compiling the robot control architecture. The added benefit of this effort is the ability to develop and modify behaviors in complete abstraction allowing for behavior modification and development that applies the each platform in the INEEL control architecture as well as robots owned by other institutions. Recently, the INEEL has ported the "universal" architecture to unmanned systems

owned and operated by the Army, Navy, and DOE as well as robots used at other research institutions.

The sensor abstractions insure not only that code can be ported from one robot to another, but also provide a means for a standardized, custom communication protocol over a reliable, low-bandwidth communication architecture. The information sent to and from the interface is not dependent on a particular sensor configuration or robot geometry, allowing novice users with no knowledge of robot size, capabilities and sensors to accomplish complex tasks. In order to support different levels of operator trust and skill, the interface is designed with several distinct modes of operator intervention that complement scalable levels of robot autonomy. The system also provides continuous sensor analysis and allows for dynamic sensor reconfiguration – a capability that proved very useful in the competition when sensors actually failed during operation.

A testament to the strength of the INEEL architecture is that the robot behavior demonstrated in the competition was developed on a platform of different make, size and sensor configuration than the one deployed in the Robot Rescue competition. Upon reaching Acapulco, the behaviors developed in Idaho where transferred to the intended robot with no source code compilation required. In fact, no code changes were necessary before or during the competition.

The technologies recently developed under the Advanced Robotic Control Architecture initiative at the INEEL provide such a structured test environment because they allow for the easy porting of robot behaviors from one robotic system to another. The net result is the ability for different robot systems to utilize the same algorithms for control. Additionally, the Advanced Robotic Control Architecture is of particular interest to the HRI community because the interface is entirely decoupled from the robot behaviors. That decoupled aspect of the Control Architecture makes it possible for multiple interfaces to be developed utilizing the same control intelligence setting the stage for a truly first-of-its-kind HRI study: A study where all robots utilize the same behaviors, wherein it can be determined if an interface implementation is beneficial or a coping mechanism for a previously poor robot control behavior.

4. REFERENCES

[1] Few, D., Bruemmer, D., & Marble, J. (2003). Usability and Portability Lessons Learned from the 2003 AAAI Robot Rescue Competition. In *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence (IJCAI-2003)*, AAAI Press.

[2] H. A. Yanco, J. L. Drury, and J. Scholtz, "Beyond usability evaluation: Analysis of human-robot interaction at a major robotics competition," *Journal of Human-Computer Interaction*, Vol. 19, pp. 117-149, 2004.

[3] Marble, J. L., Bruemmer, D. J., & Few, D. A. (2003). Lessons learned from usability tests with a collaborative cognitive workspace for human-robot teams," *IEEE Conf. on Systems, Man, and Cybernetics*.

[4] Bruemmer, D., Boring, R., Few, D., Marble, J., & Walton, M. (in press). "I Call Shotgun!": An Evaluation of Mixed-Initiative Control for Novice Users of a Search and Rescue Robot. To appear in *Proceedings of the IEEE 2004 Conference on Systems, Man & Cybernetics.*

[5] Marble, J., Few, D., Bruemmer, D., & Dudenhoeffer, D. (2004). Challenges in the Design and Conduct of Usability Testing of Human-Robot Control Architectures. In Proceedings of the ANS 10th International Conference on Robotics and Remote Systems for Hazardous Environments, Gainesville, FL.