Perspectives on Human-Robot Team Performance from an Evaluation of the DARPA Robotics Challenge

DARPA selected Boston Engineering to lead a team of independent researchers to monitor DRC events and to identify effective HRI methods that can be applied to enhance robotic design guidelines beyond DRC events. This report provides HRI recommendations that have implications in the public and private sectors.

Authors:

David Shane and Willard Ober
Boston Engineering

Adam Norton
New England Robotics Validation and Experimentation (NERVE) Center, University of Massachusetts Lowel

Lisa Baraniecki
AnthroTronix Inc.

Anna Skinner
Black Moon, LLC

Holly A. Yanco
Computer Science Department, University of Massachusetts Lowell

BOSTON ENGINEERING™
Imagine the Impact™

Table of Contents

BOSTON ENGINEERING™
Imagine the Impact™

Summary of Human-Robot Interaction (HRI) Research and Findings

Background

To maximize research opportunities for the latest DARPA Robotics Challenge (DRC), DARPA selected Boston Engineering to lead a team of independent researchers to monitor the DRC events and to identify effective HRI methods that can be applied to enhance robotic design guidelines across the public and private sectors. Importantly, the 2015 DRC Finals provided a forum to conduct the industry's most extensive study of how human-robot team performance is affected by levels of robot autonomy and interaction.

The HRI evaluation team established a novel methodology that enabled analysis of performance based on the task requirements and the complexity of the actions required. The evaluation included:

1. Identifying correlations between interaction methods and team performance (e.g., number of task attempts, speed, etc.),
2. Evaluating team communications, data transmit/receive, and the connection to autonomy, and...
3. Predicting team performance based on their interface designs, strategy and training aspects, and lessons learned from the DRC Trials in 2013.

Snapshot:

Led by Boston Engineering, the HRI evaluation team identified strong connections among the human-robot interaction Techniques (an HRI evaluation team metric that includes low-bandwidth adaptability, robot stability, and task strategy), Training (type and amount prior to the event), and Performance (incidents, successful vs. failed subtasks, speed, and more) at the DRC events.

Key Outcomes and Findings

Developed an algorithm that predicted team scores with 71% accuracy (within 1 point). This is deemed especially successful as an initial prediction because the teams' self-predictions (gathered from pre-event interviews) were only 45% correct and a randomized prediction would only prove 31% correct. The HRI evaluation team generated the predictions to:

1. **Show a quantifiable impact on performance based on defined HRI criteria**
2. **Determine the effectiveness of the prediction algorithm developed by Boston Engineering**

Established a correlation between interaction methods and performance for manipulation and mobility.

Successful HRI "level of effort" techniques enable operators to focus on higher-level task strategies instead of lower, more tactical levels of robot control and situational awareness. Importantly, "level of effort" is not necessarily an indication of the amount of autonomy, as top teams used markedly different levels of autonomy. The major difference between the successful and unsuccessful teams was in the subtle connection between interaction methods and performance that enables effort within the task-level context vs. robot maneuverability/operation context.

Summary of Human-Robot Interaction (HRI) Research and Findings

Showed that team success was not dependent on how much autonomy they used. Even though the event was staged to promote teams with more autonomy, there was little to suggest that this was the case. Some successful teams were able to modify their operator-in-the-loop processes to adjust to the data limits set for the event, while others did not base their interactions on those limits at all. The analysis of the teams' data use supported this conclusion that there was little correlation between autonomy and performance. Restricting data did, however, improve the event's realism, given the expectation of working within a degraded and unstructured environment.

Identified potential causality between combined interface techniques and performance. Across all comparisons of team performance and interface techniques (e.g., sensor fusion, simulation before execution, and the number of operators), only the comparison among teams using simulation before execution and those that did not were statistically significant, meaning sensor fusion and the number of operators have a more complicated connection to performance. **Findings include:**

- Teams with high levels of training that used simulation before execution were more than 2.5X more successful at completing tasks vs. those without
- Teams with high-level technique that used simulation before execution had more incidents and were more successful at completing tasks vs. those without
- All teams with high "Training" also used higher levels of sensor fusion
- Only one team with high "Technique" did not have sensor fusion

Observations showed that success was dependent on high levels of 'Technique' and 'Training' in addition to using simulation before execution and implementation of high sensor fusion for tasks.

HRI Design Guidelines

The HRI evaluation team applied its findings from the DRC Finals and DRC Trials to provide guidance for robot developers and component developers to be used in machines/robotics.

- **Multiply and divide sensor fusion:** Provide multiple perspectives through intuitive sensor fusion to increase the operator's situational awareness
- **Balance operator and robot responsibility:** Use methods that enabled "shared" mental models between the operator and robot to streamline interfacing
- **Separate mobility and manipulation tasks, but use similar control methods:** Control complexity demands a separation of these types of tasks (at least for the DRC Finals) – tune the interface to optimal control and feedback for specific task strategies.
  Note: The HRI evaluation team considers this a short-term guideline and expects near-term advances to close gaps in this area
- **Maintain situational awareness using minimal bandwidth:** Use a variety of techniques to maintain situational awareness when new data isn't available rather than wait for new data
- **Decrease the number of operators and/or divide a large number of operators:** Reduce the number of operators or delegate tasks by function so that those in control maintain appropriate situational awareness
- **Design holistically:** Instead of focusing on single aspects of an interaction, consider one that consists of every aspect, including the operator, robot, and the mission they are performing when developing the interaction design

Introduction

The 2015 DARPA Robotics Challenge (DRC) was easily the largest display of semi-autonomous, remote humanoid robots performing a set of standard tasks with a human operator in the loop. Some teams used DARPA-provided robots (i.e., Boston Dynamics Atlas) while some provided their own systems, but every team had to engineer their own control schemes, interfacing techniques, and autonomy, all comprising their system's human-robot interaction (HRI) methods. Our team designed and conducted HRI analyses for both the DRC Trials (Yanco et al., 2015) and Finals (Norton et al., 2017), where performing teams were observed on the field (with the robot) and in the control room (with the operators; only teams that consented to be part of the study were observed in the control room). Our analyses categorized each competition task by the type of manipulation and mobility activities needed to complete it.

We then crossed team performance on each of these subtasks with the interaction methods they used, categorizing them by control methods levels of effort, sensor fusion, and operator layout. From our analyses, several HRI components, specifically those related to the human operators, contributed to team performance: interaction modalities (e.g., sensor fusion, control methods), operator distribution (e.g., active and passive operators, fixed or rotating layouts), and interface layout (Norton et al., 2017).

Based on our studies, it is apparent that HRI had a significant impact on performance, with some teams utilizing certain techniques that outperformed others. Our findings on effective HRI techniques used at the DRC have implications outside of the competitions in the real world. To that end, this article comprises our perspectives on human-robot team performance, based upon our observations of the DRC competitions, considering the human operators, the capabilities of the robots, the operational context under which tasks were performed, and the relationships between each factor.

Throughout each section, we discuss these factors of the human-robot team through examples observed at the DRC, and their implications on current and future developments in the world of response robots. This article does not cover each factor comprehensively (see Yanco et al. [2015] and Norton et al. [2017] for the detailed presentation of the results of the Trials and Finals, respectively), but instead focuses on those we found to have had the most impact on performance at the DRC.

Given the time that has passed since the DRC Finals, many of the participating teams have published papers discussing their HRI design approaches. Where appropriate, we cite those publications or online media like YouTube videos as examples of concepts being discussed. It should be noted that other teams might have also used similar methods, but only teams who have publicly disclosed their methods since the DRC Finals in publications are identified in this paper, given the Institutional Review Board protocol approved at the University of Massachusetts Lowell.

Background and Definitions

This article heavily references two existing publications on HRI analyses at the DRC (Yanco et al., 2015; Norton et al., 2017); this section provides a brief review of the terminology used in those studies, which are used in this article. A range of performance metrics were used in evaluation of the different teams, including success of attempts and relative duration to complete tasks or subtasks.

In the development of our study methodology, we determined that it was inappropriate to focus on only a single performance metric (e.g. overall score, the competition's metric), as this is only one indication of the teams' abilities. Other metrics, such as speed, number of attempts, and number of incidents, were introduced to provide a more holistic view of performance. The possible deeper-level meanings of these metrics are discussed in the referenced publications.

To increase the granularity of our analyses, we divided each task into subtasks, which are actions or milestones that needed to be performed in order to complete the task. Each subtask and task was then categorized by the type of manipulation and mobility activities required, referred to as "subtask functions."

Six subtask functions were defined:

- Unobstructed Traverse (UT): Mobility over flat, open ground (e.g., walking from the Valve to the Wall).
- Obstructed Traverse - Foot (OTF): Mobility over ground with obstructions that pose challenges to the robot's lower extremities (e.g., walking over the blocks in Rubble-Terrain).
- Obstructed Traverse - Robot (OTR): Mobility over ground with obstructions that pose challenges to the robot's entire body (e.g., walking through the Door).
- First Order Manipulation (FOM): Fine or coarse manipulation and use of the end effector (e.g., rotating the Valve wheel).
- Second Order Manipulation (SOM): Interacting with a non-affixed object, guiding the end effector of the object (e.g., moving the drill to cut the Wall).
- Third Order Manipulator (TOM): Manipulating a system with its own control loop (e.g., driving the Vehicle).

Each team's interaction method was distilled into levels of effort based on the amount of interaction from the operator required coupled with the level of automation needed from the robot and interface. We defined levels of effort for manipulation and mobility activities. For manipulation, the levels of effort are defined as:

- Manipulation Level of Effort 1: Pre-defined action or script based on contextual information, such as the use of an object model or template, that generates manipulator trajectories; usually a single click or button press per action, sometimes the entire execution of a task is performed with a single action (e.g., turning the valve with a single wrist rotation).
- Manipulation Level of Effort 2: Maneuvering an end effector (or interaction marker) using a keyboard, mouse, or game controller (generally visualized through an avatar of the robot using a Cartesian transform tool) which uses inverse kinematics and generates manipulator trajectories; if an object model or template is used it may provide contextual information (e.g., where to place fingers when grasping an object).
- Manipulation Level of Effort 3: Sending individual joint angles using a keyboard, mouse, or game controller (sometimes using a Cartesian transform tool); does not use any contextual information.

For mobility, the levels of effort are defined as:

- Mobility Level of Effort 1: Placing a waypoint or "ghost" avatar for the robot to walk to and the footsteps are automatically generated.
- Mobility Level of Effort 2: Pre-defined action or script to step in a specified direction a number of steps; two-dimensional directional control for traversing in a direction either continuously or incrementally (similar to that of wheeled robot teleoperation).
- Mobility Level of Effort 3: Manual placement and adjustment of individual footsteps; generally only used for tasks that involve changing elevations, such as Rubble-Terrain or Stairs.

We also defined a common interaction technique for the placement of object models and templates into a camera view or point cloud display, used to add context to an autonomous action. For example, the operator would guide the robot's manipulator to the drill by placing a virtual model of the drill into the point cloud. This technique is referred to as model/template placement.

Interfacing Techniques

For teleoperated robots, the most ubiquitous display is a video feed streaming from a camera. The placement of the camera on the robot dictates what it can see in relation to the robot, such as the environment in front of the robot or an exocentric view of the robot's manipulator. Some systems also include multiple cameras.

Many systems also feature simulated robot avatars that visualize the robot's pose in 2D or 3D. This data is gathered from encoders on the robots and from lidar sensors that provide point clouds. Both types of data displays provide situation awareness (SA) of the environment and the robot to the operator.

Response robots are most often controlled via a combination of joysticks, directional pads, and buttons. Control is also typically only possible if there is sufficient communications bandwidth, due to the operator largely relying on a live display of camera data. At the DRC, higher complexity robots (i.e., humanoids) called for more complex data displays and control methods, as did the implementation of forced degraded communications links between the operator and the robot while performing certain tasks. In this section, we discuss some of these techniques and their implications.

Data Presentation

It is uncommon to find instances of sensor fusion in today's commercially-available response robot interfaces that combine multiple data displays using a common reference frame. The type of displays typically available vary in dimensionality and perspective (i.e., fixed perspective of a 2D camera image with a 3D robot avatar), making them difficult to fuse properly. Chen et al. (2007) addresses similar issues that can impact human performance for remote operation, which include image bandwidth, time lag, and 2D views.

In such scenarios, the operator is exposed to potential pitfalls due to poor presentation, which can be mitigated with proper design consideration. Stereo cameras, lidar sensors, or 3D structured light sensors can be used to provide proximity data about the environment, from varying perspectives that the operator can manipulate. While not commonly found on today's response robot platforms, most all of the teams at the DRC had systems equipped with one or more of them. These types of sensors generally provide high fidelity 3D spatial awareness in the form of a point cloud. Based upon resolution of the focal area, refresh rate of the sensor, and other viewing conditions, this technique can generate realistic 3D data that the operator can use to interpret a scene and command the robot within it.

Multiple versions of point clouds and lidar visualizations were used during the Finals, spanning from very "novice friendly" visualizations to "super-user" systems. Of the teams in our study, all 20 used camera views, 19 used a point cloud display, 18 used a simulated robot avatar display, and 19 used sensor fusion to combine multiple data displays to share a common reference frame (Norton et al., 2017). Teams that used Atlas robots all had the same Carnegie Robotics MultiSense sensors in the robot's head with stereo cameras and a rotating lidar, typically using similar interface displays of this data (see Figure 1 for an example of MIT's visualization from the Vehicle task). With many data streams fused into a single view, the display may not appear to be "novice friendly," at least compared to present day response robot interfaces.

However, in the example shown in Figure 1, even a new operator should be able to understand where the robot is in relation to the car on the display. Training would be needed to understand the context of the display with respect to the relationship between the simulated robot avatar and the actual position of the robot in the physical world, though.

In contrast, a team with a more "super-user" based approach was THOR. The images of the lidar system (see Figure 2) included a variety of additional blocks and vertical lines that were intended to indicate points of interest and/or extension limitations of the robot. As an outside observer, the relationship between the objects and lines within the image were not easy to correlate to the edges or surfaces of objects within the environment. However, this method could have been an adaptation to which the operators had become accustomed, or even preferred by one of the operators.

This interface is one example of many for teams generating highly detailed interface approaches that could be very difficult to operate by a novice operator; most interfaces was designed to be used in the context of the competition (i.e., made and operated by a developer).

In a way, THOR's distance display is similar to more traditional 2D lidar displays that provide a visualization of boundaries in the environment around a top-down avatar of the robot, as in Keyes et al. (2010), but viewed from a different perspective, drawing the boundaries as bars from the ground up. Any issues stemming from presenting distance data in this manner may come from the fact that 2D information is being presented in a 3D manner alongside parts of the scene that are displayed truer to their actual shape in the real world (e.g., the robot avatar is shown in 3D).

Sensor fusion displays with variable perspectives are not typically found on present-day response robots, so why were they so prevalent at the DRC? One reason may be due to the complexity of controlling a remote humanoid robot, with increased degrees of freedom across two legs, two arms, two hands, and the body, and issues of balance. A bipedal humanoid robot is inherently unstable, always at risk of falling.

Today's wheeled and/or tracked response robots are typically not at risk of falling if they bump into something due to being heavy and bulky (or if they are lighter then they may be designed to operate while upside down or have the ability to easily flip over). For instance, driving a wheeled ground robot through a doorway likely poses very little risk of falling. Doing so with a walking humanoid robot poses many risks of bumping into the doorframe, potentially causing it to fall. By using data displays that provide 3D representations of the environment, the robot, and the distance between them, the operator is able to better understand the position of the robot so as to prevent these types of issues.

The operator can collect a point cloud, plan the movements of the robot in simulation, and view the planned movements from multiple perspectives before they are executed on the physical robot.

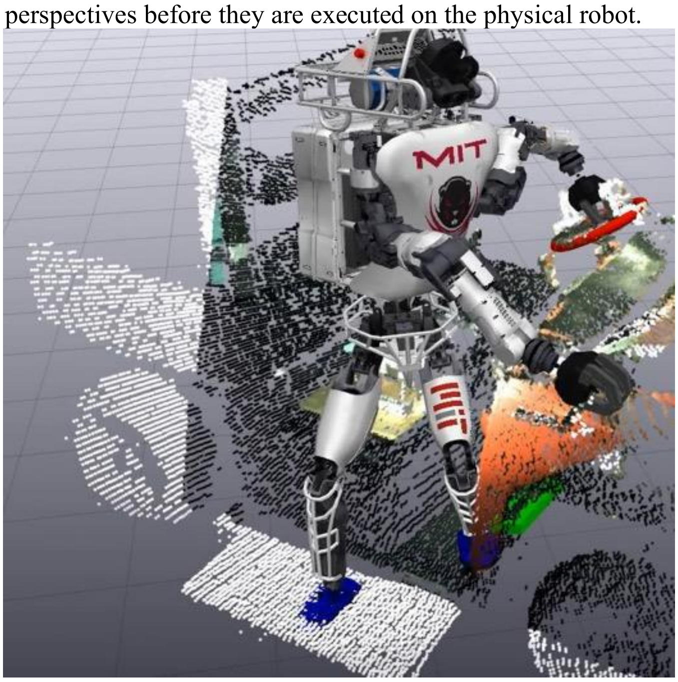


Figure 1. Team MIT's sensor fusion display showing combined camera images, lidar point cloud, and simulated robot avatar, while operating the car. Image from YouTube https://www.youtube.com/watch?v=em69XtIEEAg (accessed August 2017).

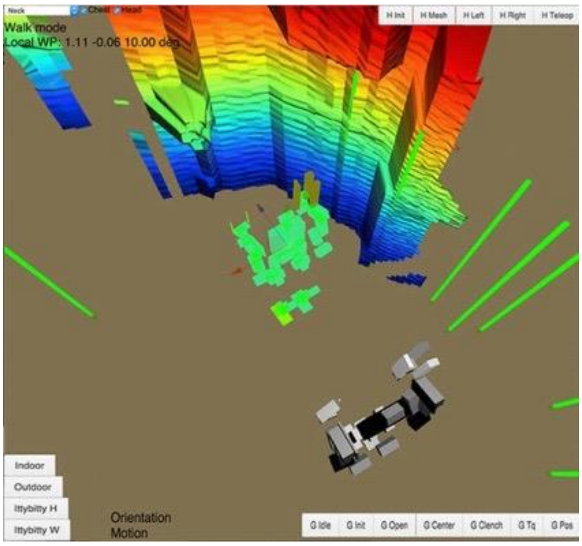


Figure 2. Team THOR's interface display showing a simulated robot avatar in a display with blocks and vertical bars to correspond with edges or surfaces in the environment. Image from McGill et al. (2017).

Many interfaces provided with today's response robots use 3D robot avatars to provide pose information of the robot; point cloud displays could be used in conjunction to provide more information about the robot's relationship to the environment, although not currently done. The 3D robot avatars are also typically able to be viewed from multiple perspectives, generally relying on the same gamepad controller that is used to drive the robot.

Changing the perspective of the point cloud displays by the DRC teams was largely performed using a mouse and keyboard, input devices not commonly found on response robots. If point clouds with robot avatar displays were to be introduced more commonly, consideration would have to be given to how and when the point cloud decays (replacing old data with new data) and the robot would have to be able to localize itself, continually, while maneuvering within the point cloud.

This type of data is certainly more complex than traditional 2D camera views, but the operator will no longer need to mentally fuse a series of camera images to mentally construct the full scene. Proper implementation can increase the fidelity of an operator's understanding of a scene and potentially increase the operator's sense of spatial presence (Stepanova et al., 2017). This also follows the ecological interface design paradigm as described by Nielsen et al. (2007) by fusing data displays along a common reference frame and providing an adjustable perspective.

Input and Output

When maneuvering a present-day response robot, the operator typically uses a gamepad to give directional commands and watches the camera displays on the interface, anticipating changes that correlate to the robot's movements (e.g., when driving the robot forward, it is expected for the video from a forward-facing camera on the robot to move "forward" into the video). If there are degraded communications, the operator likely needs to wait for updated camera images in order to regain situation awareness.

The sensor fusion displays discussed in the previous section were simultaneously utilized for robot control by DRC teams, manipulating the robot avatar's limbs into a desired position using a keyboard and mouse to plan a movement trajectory before execution. These types of control methods were very beneficial in enabling continued robot operation during periods of degraded communications. During these periods, there were less frequent updates to the higher bandwidth displays that provided contextual information about the environment to the operator and robot (e.g., camera images and point clouds).

However, a low bandwidth communication line (9600 baud) was constant regardless of where the robot was in the test course, enabling the operator to maintain situation awareness of the robot so long as the simulated robot avatar updated properly within the virtual representation of the environment around the robot (i.e., using the latest point cloud retrieved and joint encoder values from the robot).

Some teams implemented techniques that enabled them to continue to perform during periods of degraded communications. One common approach was to record the joint angles and velocities of the robot's pose, which could be sent on the low bandwidth communication line, then present them as a "ghost" within the virtual space created by the old camera images, point cloud, and/or robot avatar (essentially displaying two robot avatars; one of the robot's current state and the other of the robot's state when the last camera image or point cloud was retrieved; see Figure 3 for an example).

By doing so, teams could take individual camera images or point clouds and continue to estimate robot position and distance from the objects in the environment as they continued to operate the robot, even when few updates were present. As suggested by Jameson (2001), this approach addressed the challenge of communications issues, which is a critical design factor for the support of situation awareness.

This technique was improved when teams assigned virtual models/templates within the virtual space. One example of a team that took advantage of this was IHMC on the Valve task (see Figure 4). Once the operator identified the valve with a virtual model/template in the interface, the manipulator movements would be mapped within that virtual scene, independent of the stale visual feedback.

With this design, IHMC was able to perform the Valve task successfully with only minimal updates to the visual presentation data. This method is useful only if the robot can have accurate localization using only joint angles. Such a design is an advanced method of enhancing robot proprioception, which alleviates information analysis duties from the operator and allocates them to a system better suited given the scenario (Chen et al., 2007; Parasuraman et al., 2000).

Some teams relied heavily on scripting and autonomous processes to perform the DRC tasks, requiring less contextual information about the robot's positioning relative to other objects in the environment. For example, at least one team did not use a simulated robot avatar within their point cloud display – or at all on their interface. These teams could theoretically simplify the display; however, such features could be helpful during oversight or error handling. Teams using sensor fusion displays with variable perspectives performed better than those that did not (Norton et al., 2017), so it follows that including more contextual information for both the robot and operator is beneficial for effective operation.

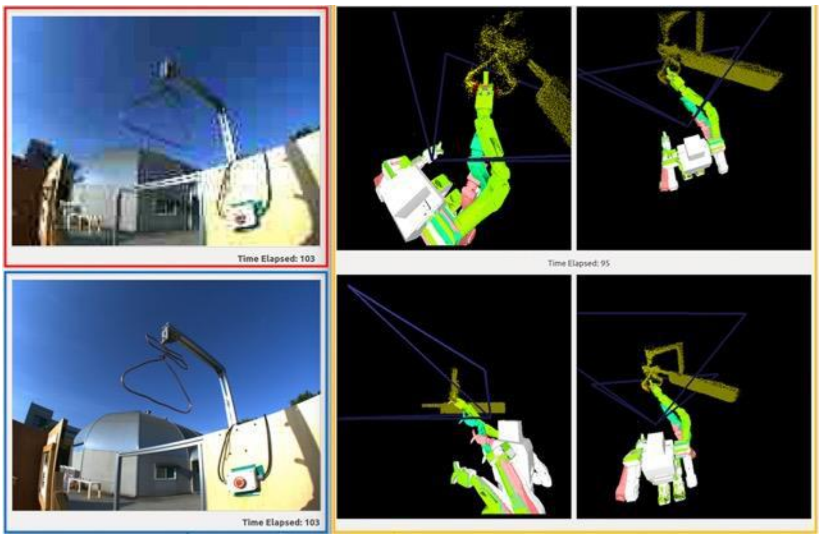


Figure 3. Team KAIST's "ghost" robot avatar within a point cloud display (right) alongside a camera image (left) of one of the prospective Surprise tasks. Image from Lim et al. (2017).

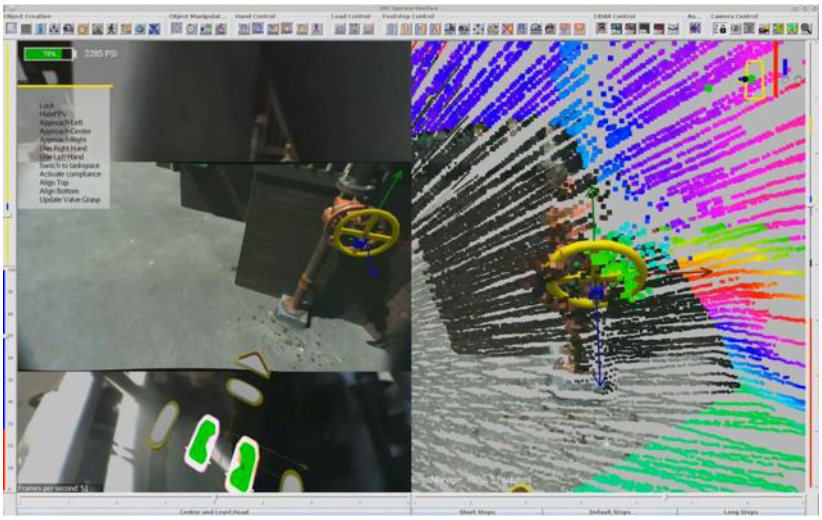


Figure 4. Team IHMC's user interface while performing the Valve task, with a model/template of the valve wheel placed into their sensor fusion display. Image from YouTube https://www.youtube.com/watch?v=TstdKAvPfEs (accessed August 2017).

However, providing contextual information for the robot and operator did not always function properly. We observed a few poor examples in different areas, one being when the accuracy of point clouds and the placement of models/templates was not updated automatically by the system. These errors occurred mostly when communications were degraded, with some teams incorrectly executing manipulation due to incorrect or insufficient movement (e.g., the robot performs the actions necessary to reach and grasp the valve, but the valve is actually beside the robot's hand rather than within it). This type of error can be caused by poor localization of the robot while maneuvering within a stale point cloud.

While old camera images and point clouds would typically decay once new ones were received, the addition and removal of virtual objects was largely the responsibility of the operator. We observed several instances of inappropriate timing for simulated objects to exit the sensor fusion display. Most notably, this was seen during a run where the "ghost" image of the valve wheel appeared in the interface while the team was performing other tasks. This was a frustrating, but minor, distraction given the competitors' exhaustive knowledge of the test course, but in a real-world scenario with limited prior knowledge about the environment, this could be a significant issue with which the operator would need to contend.

Teams implemented many different mitigation approaches to assist with robot stability, especially for the robots that were not statically stable (e.g., Atlas). One approach was placing cameras at the robot's knees to provide the operator with significantly more situation awareness regarding the robot's position in relation to the environment and the quality of foot placement (see Figure 5). In providing this additional modality in the interface, the operator was able to verify a potential mismatch between the footstep plan the robot was supposed to take compared to what it actually performed. This technique proved effective; the team that implemented it was the only team using an Atlas robot which did not fall during the Finals (Atkeson et al., 2015).

If the operator trusted the robot's capability to remain localized while operating within stale point clouds during periods of degraded communications, then he/she essentially did not need to be concerned with the state of communications. However, complete blackout periods in a real-world deployment would call for much more autonomy from the robot.

While the ability for the operator to observe the robot moving in a simulated display via a "ghost" robot avatar and point cloud display would assist with keeping the operator in the loop with respect to the robot's status, during a communications blackout the robot avatar would only be updating according to the commands that were sent, not necessarily the result of those commands on the physical robot, which is even more reason to present both the last robot state and the "ghost" of the command state. It should be noted that this technique will have less success in a highly dynamic environment where both the world and the robot's position within the world is changing.

Figure 5. Team WPI on the Terrain task. From top left, clockwise: the robot on the test course, camera image from the robot's knee, and two point cloud with robot avatar displays showing a mismatch between the robot's plan and the actual step that was taken. Image from Atkeson et al. (2015).

Based on our observations at the DRC, no robots operated without some input from the operator. Most automated manipulation performance (i.e., control method level of effort 1) was used in conjunction with models/templates placed by the operator. Prior to the competition, teams were able to program and train their systems to perform using these virtual objects, enabling them to act autonomously during the competition. This speaks to the relationship between the robot and operator.

Robots are good at performing known tasks (or known parts of tasks) and the operator is good at specifying the variable aspects of those known tasks. The robot/interface can present data to the operator in a fused manner, and the operator can interpret it in a meaningful way. Robots are also good at sensing/detecting things more precisely and more quickly than humans, and while they are slow with interpretation, low-level autonomy such as reacting to obstructions to perform obstacle avoidance is plausible.

Team MIT implemented a dynamic stabilization technique that enabled the robot to autonomously deploy a rapid foot placement in an attempt to maintain its stability when it perceived that a fall was imminent (Atkeson et al., 2015). This approach is much more advanced as it does not require operator input to be used. Further developments in these kinds of reactive behaviors for robots could significantly aid operators in terms of managing workload, in addition to preventing failures.

Robot Morphologies

Seventeen teams were responsible for building or procuring their own robots for the Finals, while six teams were granted the use of a Boston Dynamics Atlas robot by DARPA (one other team also used an Atlas, but procured it privately). Of the non-Atlas robots, eleven different robot makes/models were used. The characteristics of each team's robot varied across many characteristics, such as their overall size, degrees of freedom, and mobility methods (e.g., biped, quadruped, wheeled, tracked).

The characteristic that appeared to have the most significant impact on performance was balance: robots that had a statically stable configuration filled four of the top five finisher slots at the DRC Finals. See Figure 6 for some examples of statically stable (SS) and statically unstable (SU) platforms, including examples of reconfigurable robots that morphed between statically stable and unstable configurations.

Mobility and Stability

A statically stable (SS) configuration was very useful for maintaining balance and moving quickly during the execution of both mobility and manipulation tasks at the Finals. Seven teams used robots that were either SS by default or had the ability to change to a SS pose, and the remaining sixteen always operated in statically unstable (SU) positions.

Based on the metrics and comparison methods used in our previously published analyses of robot performance at the Finals (Norton et al., 2017), on average, teams with SS robots generally performed better than the SU robots, and, in many cases, these differences in performance were statistically significant (see Table 1). Due to the disparity of scores between robots of each type, only the most relevant tasks (Door, Valve) or subtask functions (UT, FOM, all mobility subtask functions, or all manipulation subtask functions) for each metric are presented.

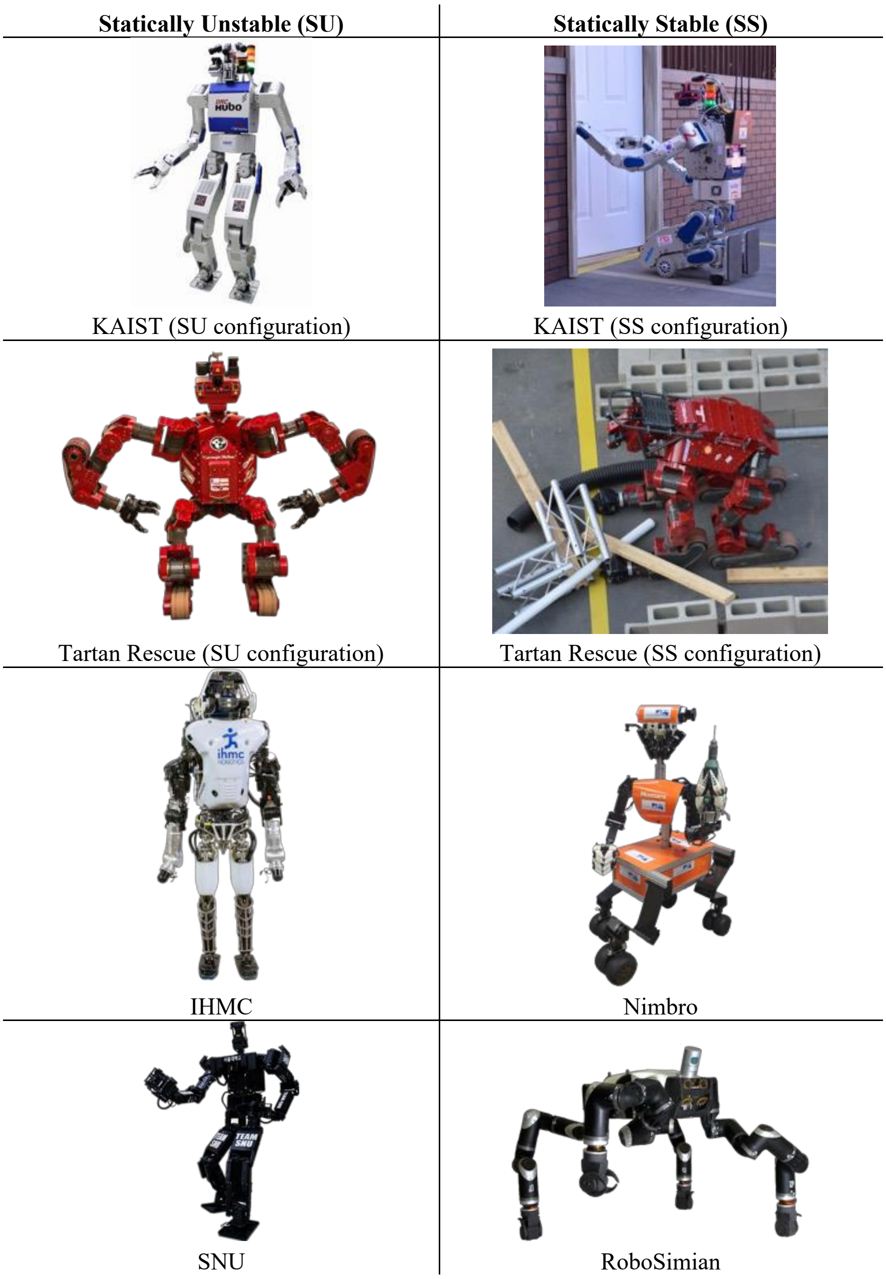


Figure 6. Some of the robots used in the DRC Finals, showing varying morphologies. Photos from http://archive.darpa.mil/roboticschallenge/ (accessed August 2017)

Metric	Task or Subtask Function	Statically Stable (SS) Robots			Statically Unstable (SU) Robots			Comparison
		avg	stdev	n	avg	stdev	n	SS vs. SU
Percentage of Failed Attempts	UT	0.0%	0.0%	61	7.1%	25.9%	84	SS*
	Door	2.1%	7.7%	28	20.3%	33.9%	74	SS**
Percentage of Successful Tasks/Subtasks	UT	100.0%	0.0%	61	92.9%	25.9%	84	SS*
	Door	100.0%	0.0%	28	80.2%	38.9%	74	SS**
Relative Duration to Complete Tasks/Subtasks	UT	77.4%	49.6%	60	114.5%	101.6%	79	SS*
	FOM	79.5%	58.2%	44	109.0%	67.9%	70	SS*
	Valve	49.8%	32.2%	20	121.5%	86.6%	39	SS**
Falls Per Attempt	All Mobility	2.4%	14.5%	105	10.3%	29.7%	149	SS*
	All Manipulation	0.0%	0.0%	63	6.7%	25.2%	119	SS*

Table 1. Comparison of performance between teams that used SS robot configurations and those that used SU robot configurations. For the percentage of failed attempts, each team's attempt at a task or subtask was recorded as successful or failed and expressed as a percentage of the total attempts made. The same goes for falls per attempt (i.e., each attempt presents an opportunity for the robot to fall). The percentage of successful tasks/subtasks does not consider failed attempts, only ultimately successful performance (e.g., the robot can fail multiple attempts to grasp the door handle to generate a percentage of failed attempts metric on that subtask, but ultimately still end up successfully completing the Door task).

Relative duration is calculated by dividing a team's duration on a task or subtask by the average of all the teams that completed that task or subtask (i.e., <100% is below average, >100% is above average). Each grouping of team performance data points (i.e., performance of SS or SU robots) was then averaged together and compared. The category of robot with better performance is noted in the "Comparisons" column; if the difference in performance is statistically significant from performing unpaired t-tests, they are indicated by * ($p < 0.05$) or ** ($p < 0.01$).

Overall, the SS robots had fewer failed attempts, more successful tasks/subtasks, faster relative times to complete tasks/subtasks, and fewer falls per attempt. As highlighted in Table 1, the SS robots made significantly fewer unobstructed terrain (UT) errors and completed significantly more UT subtasks than the SU robots; this was due to the primarily flat nature of the ground and non-confined spaces in which many of the UT subtasks took place in the competition. The teams with SS robots had the ability to complete all mobility tasks in a wheeled or tracked mode, enabling them to control their humanoid robots in a manner similar to a traditional ground robot, involving inputting simple directional commands (mobility level of effort 2).

In some cases, these teams also employed waypoint navigation, requiring even lower levels of interaction and effort on the part of the operators (mobility level of effort 1). These lower level-of-effort control methods combined with the stable base provided a distinct advantage over the bipedal SU robots, as evidenced by significantly faster times on all UT subtasks and for all manipulation subtasks, particularly first order movement (FOM). Stable robots also performed the Door task with significantly fewer errors and more completed subtasks, and performed the Valve task significantly faster than their SU counterparts. Finally, the general lack of falls for SS robots was notable, including significantly fewer falls across all mobility tasks/subtasks and all manipulation tasks/subtasks.

These results imply that the use of a statically stable base was extremely beneficial in the competition, and resulted in better performance on the majority of tasks spanning various subtasks and metrics. While this generally proved to be the case, two of the SS robots scored 0 points, and there were multiple SU robots that performed exceptionally well, including two teams that achieved 7 points and one that achieved all 8 points, as shown in Figure 7.

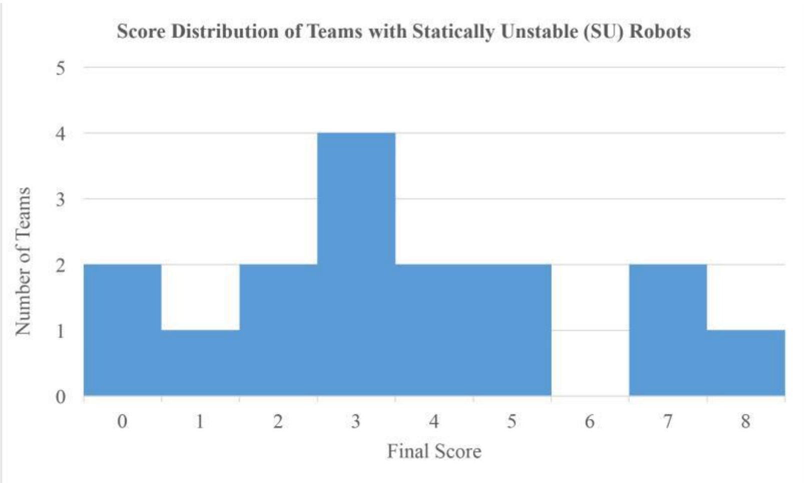
Also, highlighted in Figure 7 is the stark contrast in distribution between SS and SU robots. The SS robots only came in at the extreme ends of the scoring spectrum, with five high-scoring teams and two extremely low-scoring teams. On the other hand, the SU robots' scores are more normally distributed. It is worth noting that there were twice as many teams with SU robots, which likely influences the distributions.

In order to further explore potential contributing factors to the overall success or failure of teams with both SS and SU robots, we can discuss a few select teams performing two of the tasks that required considerable amounts of both mobility and manipulation: Door and Wall.

We specifically selected these two tasks because they represented a task that was performed by most teams (Door) and a task that was only performed by the higher scoring teams (Wall). As the Door task was one of the first tasks, and required completion to progress to the remaining tasks (i.e., it couldn't be skipped), we were able to compare data for both high-scoring (HS) and low-scoring (LS) teams using SU and SS robot configurations (i.e., HS-SU: IHMC, HS-SS: KAIST, LS-SU: VALOR, LS-SS: AERO). As the LS teams didn't even attempt the Wall task, our analyses for this task focus only on HS teams with both SU and SS robots.

These more in-depth analyses focused on the relative duration metric for the selected tasks in an effort to better understand the time teams spent completing a certain task relative to the average time of completion across all teams. Relative duration is calculated by dividing a team's duration on a task or subtask by the average of all the teams that completed that task or subtask.

Since relative durations are based on all the teams that completed the task, the relative duration metrics for the later tasks such as Wall only take into account the HS teams, whereas the relative durations for the Door task are relative to average durations that include both HS and LS teams.



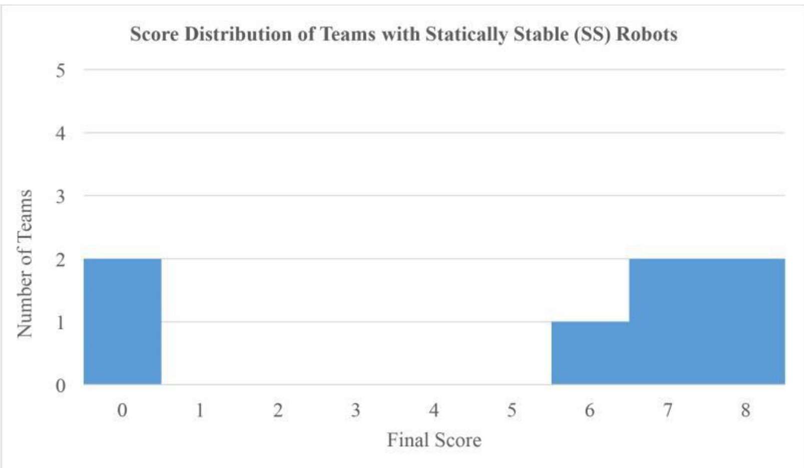


Figure 7. Scores distribution of teams with SU robot configurations (top) and teams with SS robot configurations (bottom). If a team used their robot in both a SS and SU configuration, they are only included in the SS graph

See Table 2 for a side by side comparison of relative durations for the Door task for AERO, VALOR, KAIST, and IHMC on Day 2 of the competition. The subtask functions of the Door consisted of first-order manipulation (FOM) to manipulate the handle and open the door ("Open door" subtask), and obstructed traverse – foot (OTF) and obstructed traverse – robot (OTR) to traverse through the doorway ("Traverse through door" subtask).

In general, both HS robots performed faster than the LS robots, although AERO traversed through the doorway in the same amount of time as IHMC. Comparing the two HS teams, KAIST was faster than IHMC, except when it came to traversing to the door, which was done after performing the Egress task. During this time, KAIST had to transition from their SU configuration to their SS configuration. While both LS teams had some difficulty with opening the door, both subtasks involving traversal were much quicker for AERO than VALOR, who didn't even get to traverse through the doorway due to time expiration.

Relative Duration of Door Task (Day 2) for Select Teams				
Task/Subtask Breakdown	**LS-SS: AERO**	**LS-SU: VALOR**	**HS-SS: KAIST**	**HS-SU: IHMC**
Door	127.6%	472.0%	31.9%	51.0%
Traverse to door (UT)	57.5%	172.6%	57.5%	28.8%
Open door (FOM)	184.6%	784.6%	23.1%	46.2%
Traverse through doorway (OTF, OTR)	33.3%	n/a	16.7%	33.3%

Table 2. Relative duration for low-scoring (LS) and high-scoring (HS) teams with statically stable (SS) or statically unstable (SU) robots performing the Door task and its subtasks on Day 2 of the competition. Relative duration is calculated by dividing a team's duration on a task or subtask by the average of all the teams that completed that task or subtask. A relative duration under 100% means the team was faster than average, while a relative duration over 100% indicates that the team was slower than the average performance on that task or subtask across all teams completing it.

The robot used by KAIST (who took first place in the competition and was able to perform tasks in both SS and SU configurations) performed the opening of the door and traversing through the doorway while in an SS configuration, doing so faster than the highest scoring SU team (IHMC). This is likely due to the stable nature of their base, allowing them to be less concerned about balance or whole body control while manipulating the door handle. Also, when moving through the doorway, there was a lower chance of KAIST's robot colliding with the doorframe due to a shortened height (by getting "on its knees" for the SS configuration) and lack of side-to-side movement while traversing (many humanoid robots swing their hips while walking, as IHMC, the highest scoring SU team, did with the Atlas robot).

However, it should be noted that IHMC did not fail any attempts at the Door task, nor were there any critical incidents. Also of note on the Door task, the second-highest scoring SS robot from team Tartan Rescue demonstrated the exceptional value of having a reconfigurable design in some cases for error recovery. Tartan Rescue's robot fell while performing the Door task, but managed to complete the task without a reset by reconfiguring itself from a bipedal to a quadruped stance and righting itself, as shown in Figure 8.

See Table 3 for a side by side comparison of relative durations for the Wall task for KAIST and IHMC on Day 2 of the competition. The Wall task was comprised of subtask functions FOM for grasping the drill, UT for traversing to the wall while carrying the drill, and second-order manipulation (SOM) to operate the drill to cut into the wall.

IHMC performed this task slightly faster than KAIST, with the inverse only being true when it came to the subtask of actually cutting the hole in the wall. IHMC was faster to traverse to the shelf, to grasp and activate the drill, and to traverse to the wall with the drill in hand. In fact, they had positioned themselves while grasping the drill such that minimal locomotion was needed to align with the wall and begin cutting the wall. The difference in performance time may be indicative of the options available for each robot type: SU robots had a higher risk of falling when the robot traversed than SS robots, allowing the SS robots to move more freely.

Overall, we found that a large impact on performance was made from robots falling down. There were considerable time consequences for falling down in the DRC; robots were penalized with a 10-minute delay before they could continue their task. A particularly rough fall could damage the robot preventing them from getting up at all. In the real world, critical errors may cause extreme loss of situation awareness and may not be recoverable. Tartan Rescue's self-recovery after a fall is a good example of resilience. In general, robots that utilized SS platforms were very resistant to falling down.

Relative Duration of Wall Task (Day 2) for Select Teams		
Task/Subtask Breakdown	**HS-SS: KAIST**	**HS-SU: IHMC**
Wall	88.8%	68.3%
Traverse to shelf (UT)	55.2%	27.6%
Grasp and activate drill (FOM)	113.9%	81.3%
Traverse to wall with drill in hand (UT)	54.9%	27.5%
Cut opening in wall (SOM)	58.7%	78.3%

Table 3. Relative duration for two high-scoring (HS) teams with statically stable (SS) or statically unstable (SU) robots performing the Wall task and its subtasks on Day 2 of the competition.

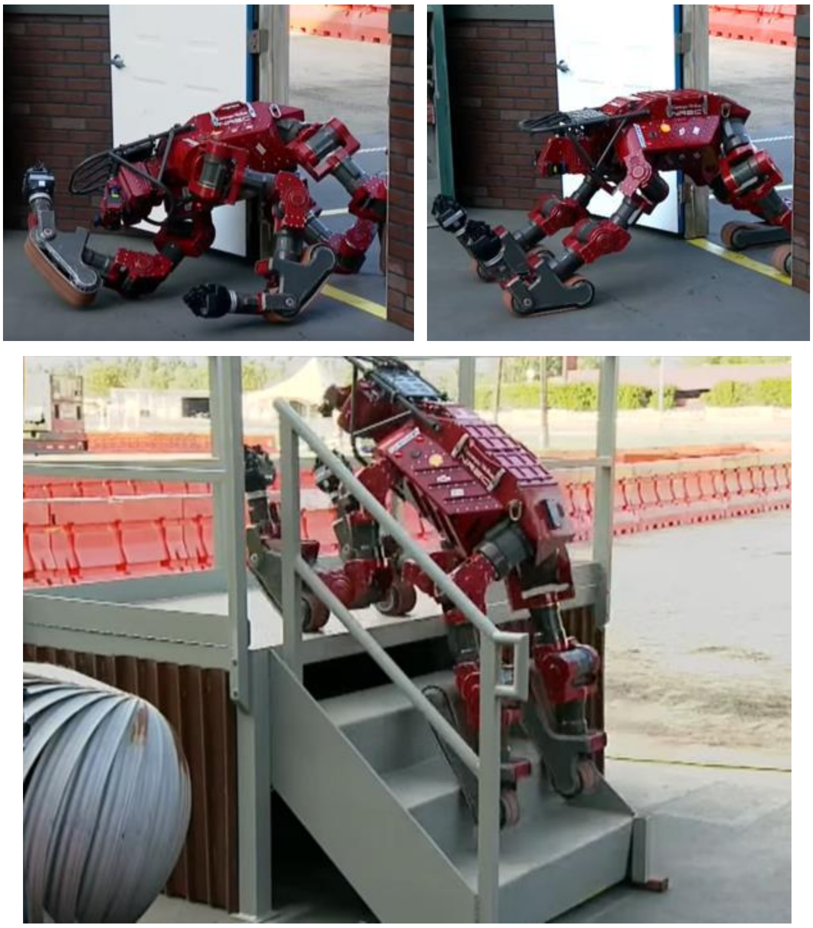


Figure 8. Team Tartan Rescue taking advantage of their unique robot morphology to recover after falling through the doorway (top) and to climb the stairs moveable limbs with embedded wheels and tracks (bottom). Photos from https://www.youtube.com/watch?v=FRkYOFR7yPA (accessed August 2017).

Versatility (multiple options/configurations) appeared to be beneficial for optimal performance on the DRC tasks. Different means of mobility were superior across many tasks. Most teams employed robots with biped configurations for the Stairs task, although Tartan Rescue displayed that this is not the sole method/configuration for performing stair-climbing (see Figure 8).

Legs or heavy-duty tracks are useful for uneven terrain. Tracks are also suitable for rubble as these teams could force their way through debris, and work well for UT, but they have the limitation of a single DOF. Omni-directional mobility is more efficient and could be beneficial in small spaces. KAIST demonstrated this limitation, by having to perform a multi-point turn in order to align with the Wall. While wheels are great for UT, omni-wheels could be even better. Overall, KAIST used versatility very effectively, using tracks when appropriate and legs when it was more suitable.

These two aspects, resistance to critical errors and versatility, appeared to greatly affect performance with respect to scoring at the DRC. They also impacted other factors that are relevant to human-robot teaming in teleoperation scenarios. Having a robot that has fewer risks posed against it and/or is resilient in the face of engaged risks eases the burden on the operator allowing them to focus on other tasks.

A versatile robot gives the operator additional options to choose from; this is beneficial when a situation calls for a specific tool-set. It is further advisable to have easy control methods for switching between configurations as it eases the load on the operator. These recommendations are in line with the adaptive mobility requirements put forth by Blitch (2003), specifically that for recovery from tumbles. One apparent drawback of having these multiple configurations is that the operator has to be good at each configuration (i.e., a "super-user"). If a "super-user" is absent, multiple operators might be required to control each configuration.

Obstructed Traversal

During the Finals, teams could choose to perform either of the Rubble tasks: Terrain or Debris. Both tasks were aimed at exercising robot mobility over or through obstructions.

The Terrain task consisted of four rows of pitched cinder block steps with a flat step in the middle. The humanoid robot capabilities needed to perform this task included planning footfalls over surfaces of varying elevation and pitch, as well as balancing while ascending and descending. The robots that performed this task only did so in statically unstable (SU) configurations. As such, we classified the Terrain tasks as subtask function OTF. From our analyses, all teams performing the Terrain task placed individual footsteps for the robot, adjusting the placement and pitch of each step (mobility level of effort 3).

See Figure 9 for an example of IHMC, a SU robot, on the Terrain. In some teams, this footstep planning was sometimes preceded by placing a model/template of the Terrain into a point cloud view, then placing a waypoint either on the flat blocks in the middle or on the flat ground at the other end (mobility level of effort 1), with the robot generating a footstep plan based on the model/template and/or the point cloud data. The operators then adjusted the model/template of the Terrain and the planned footsteps based on their interpretations of the environment, using a combination of the available interface data displays (e.g., camera views, lidar, point clouds, and/or sensor fusion displays of both).

Using this method, a balance was struck between the robot and the operator: the operator interprets the environment to place and adjust a model/template that matches, the robot plans footsteps in the environment, the operator adjusts those footsteps to match their interpretation, and the robot executes the footsteps while maintaining balance. Extra care was also likely taken to perform this task due to the potential severity of failure for this particular task (i.e., falling from an elevation).

The Debris task used a series of objects (2x4s, 4x4s, metal truss, etc.) to obstruct a path. For the Finals, the robot could either traverse through the pile or move the objects out of the way to reach the other side. The only criteria for scoring a point was that the entire body of the robot be over a line on the other side of the task, so the robot could traverse through the pile by either stepping over the objects, push the objects out of their way using their legs/mobility base, and/or manipulate the objects out of the way. All of the robots that performed the

Debris task did so using a statically stable (SS) mode, driving through the pile to push the objects out of the way, and no manipulation of the obstructions was observed.

We classified its subtask function as OTF. Placing waypoints and 2D directional control (mobility levels of effort 1 and 2, respectively) were used by all teams when performing Debris, as no individual footsteps needed to be planned. See Figure 9 for an example of KAIST, a SS robot, in the Debris. No walking/footsteps were used at all, due to the change in robot locomotion method. The balance between the operator and robot in this case much more relied on the operator trusting that the robot, in its SS configuration, would be able to withstand colliding with the obstructions.

No teams in our study were observed using models/templates to perform the Debris task (it should be noted that at least one team that performed the Debris task did not consent to be in our study). They would likely only be used if the robot had to place footsteps through the pile, which might require a model/template of each individual obstruction to be placed.

During the Finals, on average, the Terrain task was completed in 7:40 and the Debris task was completed in 4:47 (Norton et al., 2017). This difference in time is due to the design of the tasks, each requiring the HRI methods previously described. It also correlates to the control methods levels of effort exhibited by teams on each task: placing footsteps (level 3) takes many more actions, and introduces more opportunities for error, and adjustment than placing waypoints and 2D directional control (levels 1 and 2, respectively). Also, no robots performing the Debris task appeared to be autonomously assisting the operator with the decision of which obstructions would be easier or more optimal to push out of the way, unlike the robot planning initial footstep plans on the Terrain task.

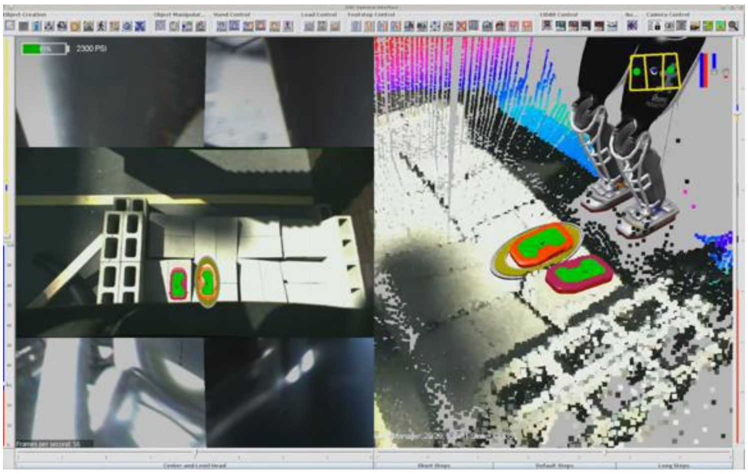


Figure 9. IHMC on Terrain task, descending the pitched cinder blocks. Top: Robot on the test course. Photo from http://darparoboticschallenge.com/ (accessed December 2015). Bottom: Interface with planned footfalls to descend in fused point cloud, camera data, and robot avatar display. Image from YouTube https://www.youtube.com/watch?v=TstdKAvPfEs (accessed August 2017).

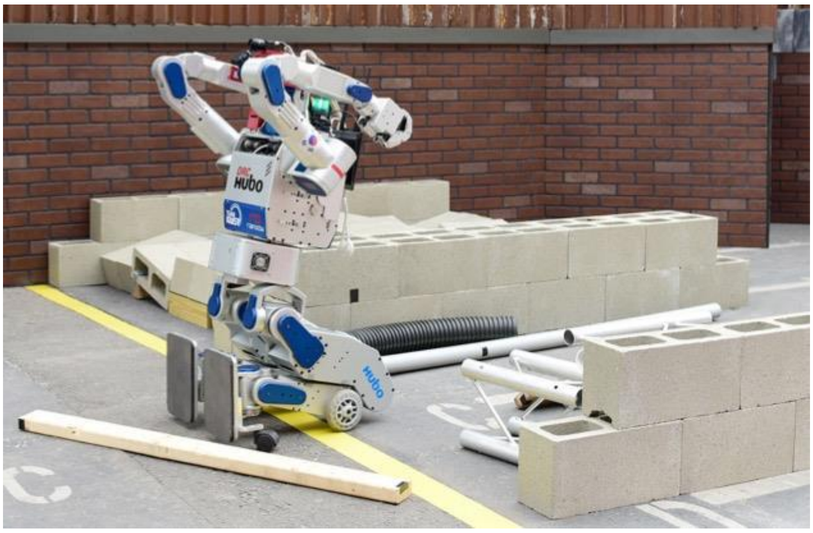


Figure 10. KAIST on Debris, in a wheeled, statically stable mode pushing the objects out of the way. Photo from http://darparoboticschallenge.com/ (accessed December 2015).

Why didn't any robots attempt to perform the Debris by walking through it? Both the Terrain and Debris tasks spanned the same ground path length, and could theoretically be achieved using the same number of footsteps. From our observations, the Debris pile appeared to be traversable by a human stepping throughout, with no manipulation of the objects necessary.

Most robot footsteps taken were short strides; for the Terrain, only strides of the length of the two cinder blocks (~40 cm) were required, but also involved ascending or descending a surface. The widest item that a robot would have had to step over in the Debris pile was the truss (~25 cm), which would involve the robot's foot and all parts of its leg fully clearing the volume of it to step over, requiring a much higher lifting of the foot than was observed during the competition.

Ultimately, if the team dedicated their robot development to include a SS configuration (implementing both the physical components and the automation to change between SS and SU configurations), they would likely choose to perform the Debris task due to its design. A real-world deployment of a remote humanoid robot, though, could benefit from both types of capabilities.

The inspiration for the Terrain and Debris tasks come from experiences with tracked ground robots responding to the Fukushima Daiichi disaster (Nagatani et al., 2013). In the field, the operator may encounter an obstructed path and have an understanding of the weight and dimensions of the obstructions such that they could use a robot to push them out of the way (e.g., pushing 2x4s using a SS configuration, like in the Debris task). However, debris piles can contain twisted and destroyed objects, some of which may not be movable. As such, the ability for a robot to step through and over debris and non-flat terrain is desirable.

While the operators at the DRC could apply models/templates to aid their robot in task performance, the variability of the real world calls for more robust methods. This further justifies the need for the operator to remain in the loop and provide their interpretation of the environment through the interface, even more so in unknown environments. For capabilities like walking through debris, the operator will likely need more situation awareness of the actual configuration of the layout.

Camera images and point clouds may be limited in the data they provide due to the viewing angle from where the sensors are situated on the robot. Additional time may need to be spent scanning the environment to build a better representation of the world such that the robot can effectively act within it. Much of this knowledge is not needed when using a wheeled/tracked platform, as development focus for those systems has largely relied on building hardened and durable mobility methods, rather than those that need to be concerned with balance and purchase (i.e., walking with legs). A combination solution, like that used by team Tartan Rescue, may prove the most fruitful, wherein wheels/tracks are embedded within the feet so both locomotion methods can be utilized (see Figure 8).

Operational Context

The goal of the DRC was to bring the challenges of operating a robot in a disaster scenario to light. The competition was structured to focus on several challenges in this context that were most critical to performance. For example, maintaining a data connection to a robotic system is very challenging in scenarios, like those presented during the Fukushima Daiichi reactor disaster (Nagatani et al., 2013), driving the need for greater levels of autonomy in the absence of constant operator control.

During the competition, the teams were forced to find a solution as the data connection was purposely limited to reflect this challenge. Working in an unknown, unstructured, and variable environment also presents significant challenges especially when combined with communication issues – the operators cannot rely on any significant amount of a priori knowledge to help them achieve their goals. Finally, environments like nuclear power plants are built specifically for human operation – tools, controls, gauges for reactor status, etc. all take forms that are intended for human interaction and interpretation. These characteristics are not necessarily ideal for robotic systems to work with. In this section, we discuss why these factors make operation in a disaster scenario so challenging, as demonstrated during the DRC.

Manipulation in Human-Centric Environments

Why does operating in a human-centric environment present challenges? We believe that this is due in large part to the manipulation aspect of the task performance. The requirements for effective mobility seem to be fairly well understood (if not addressed all that effectively), and while balancing is a challenging problem (as exemplified by the number of falls at the DRC), we believe this is due to insufficient control, balancing, and mobility algorithms. Even the Debris task, originally meant to illustrate the challenge of mobility in a damaged environment, was addressed primarily via maneuvering of the debris pile.

To characterize manipulation in an environment made for humans, we define the task in terms of the systems involved from the perspective of the operator. Our breakdown of subtask functions from both of our studies broke down manipulation tasks into "orders of manipulation" (first, second, and third; FOM, SOM, and TOM, respectively). This method of characterization was used because it effectively captures the required feedback loop to the operator based upon what is normally a simple hand-eye coordination task for a human.

For example, grasping and rotating the valve at the DRC was categorized as an FOM subtask because most teams had already programmed in the expected rotary motion of the valve. Once grasped, a simple script to drive the manipulator through a known trajectory was sufficient. The valve to which the wheel is attached constitutes another system in this example, but that system did not impact the control of the robot. In systems terms, the control loop was closed around the robot's proprioceptor sensors.

SOM can be broadly defined as "indirect manipulation," meaning the requirements of the task demand that the robot not only directly interact with something, but indirectly act through something. The best example of SOM at the Finals is the Wall task. In this task, the operator had to command the robot to grasp the drill and then use the drill to cut a pattern out of the wall. The drill introduces a gap between the robot's system and the output of task performance (i.e., the drill bit must call the wall, as controlled by the robot), and is considered a second order action for that reason.

To effectively cut the pattern on the wall, the operator had to close the control loop not around the robot's proprioceptors, but on the path of the drill bit, which is a step apart from the proprioceptors. In reality, most of the teams were able to turn this into a FOM subtask by using a preprogrammed trajectory that they refined trough empirical testing. In this case, the only real SOM aspect came into play when validating whether the path was correct or not. The drill is not considered a "system" per se because it has a known output given a known input (hence how people were able to treat it as FOM). However, it is still a gap between input and output that requires some understanding of the effect on the environment, constituting it as SOM.

TOM, on the other hand, introduces an entire system between the robot's system and the output. The output of this additional system is unknown given a known input, requiring the operator to close the control loop around the final output instead of internal proprioception. In other words, TOM requires the operator to predict or anticipate the effect of the robot system's actions on the other system, which is considered level 3 situation awareness (Endsley, 1995).

This understanding and projection into the future is gradually built while the task is being accomplished and awareness is gained. The Vehicle task is a great example of TOM (see Figure 11). The operator commanded the robot to turn the steering wheel and press on the gas (first system) to get a response of the vehicle (second system). The link between the gas pedal and steering wheel to the vehicle output constitutes a second system that is unknown to the operator. To effectively control the vehicle, the operator had to take his/her feedback from the output of the combination of these two systems: the vehicle motion. In systems terms, the control loop was closed around the output of the secondary system.

Some teams even used a steering wheel with foot pedals as input devices to control the robot while operating the car, like team DRC-HUBO at UNLV (Oh et al., 2017). Another relevant example for operation in a nuclear reactor scenario would be the control of a crane (i.e., the robot is used to operate the crane controls and visual feedback is provided to the operator) or, more simply, rotation of a valve to achieve a value that registers on a nearby gauge (as opposed to rotating a predetermined amount, like what was seen in the DRC).

If we look at these concepts in the context of a human environment, we see that people use TOM in this sense all the time. People are very good at hand-eye coordination and also at quickly understanding the link between inputs to an unknown system and the output (e.g., driving a car or operating a crane). Our environment has been built around this skill that comes very easily for us, but is very difficult for a human controlling a robot to attain through an interface. The bottom line here is that for a human-robot team to be effective in a human environment, they need to be effective at TOM.

Figure 11. Team DRC-HUBO at UNLV performing the Vehicle task. Top: The robot operating the car. Photo from http://www.drc-hubo.com/ (accessed August 2017). Bottom: The operator's interaction method for driving the car, showing the car trajectory on the left and the steering wheel and foot pedal input devices on the right. Image from Oh et al. (2017).

Environmental Characteristics

Given the complexity of unknown environmental characteristics for a human-robot team, the DRC Trial tasks had well specified, known characteristics. Teams knew everything about each task before attempting to complete them, including dimensions and locations of the valves, size and shape of the debris objects, and exact terrain conditions. This information was necessary to ensure the tasks were actually achievable by the teams. However, this availability of information is clearly not realistic for an actual scenario.

Variation of the environment is a core component of achieving a realistic scenario. We defined three categories of effort discussed throughout our evaluations, first presented in Yanco et al. (2015) based on observations at the Trials, then refined in Norton et al. (2017) to evaluate the Finals. These categories help us to understand the impacts of changing characteristics for the tasks. Based on our findings, teams that were able to streamline robot control (level of effort 3) and situational assessment (level of effort 2) performed better because they didn't have to put effort into anything but accomplishing the task at hand (level of effort 1). If we look at the techniques used by the teams to increase the effectiveness of situational assessment, we see that it was very limited in its application.

Understanding the environment includes developing a higher-level understanding of the things within the environment with which the robot can interact (level 2 situation awareness; Endsley, 1995). In a system with effective HRI, the operator should able to quickly understand the environment around the robot using tools such as camera feeds and lidar data. However, to effectively control the robot, this information must be put into terms that the robot can understand.

We found in the Trials that much of the effort applied in the HRI task was in developing this mutual understanding between the operator and the robot. From both studies, we saw that teams using interaction techniques with displays of simulation over live data generally performed better, as this method enables effective mutual understanding of the environment and control from the operator. However, the flexibility of this technique in developing higher-level understanding is limited because it was based off of exact dimensions and constraints provided to the teams prior to the competitions, limiting its use to a static and known environment.

It required some effort from the operator to gain situation awareness of the environment using acquired sensor data to place the 3D models correctly, but not as much as it would have in a variable environment. Operators already had well understood mental models of the task environments, as they had been training on them throughout their systems' development, and therefore they were not required to gather much level 2 situation awareness during task execution.

To make the tasks more comparable to a realistic scenario, variation and unknowns need to be introduced. For example, instead of specifying the size and location for every piece of debris, a range of characteristics could have been provided; instead of locating the valves and doors in a very specific location, tolerances for those locations or even changes in height and/or ordering could have been provided; instead of specifying exactly what valves or doors had to be opened, a list of possibilities could have been provided.

This would have forced teams to develop tools beyond what they used in the Trials to address flexibility. By adding variability to the task setup, operators would have to initially gain situation awareness of task-specific dimensions and locations when a task is encountered and adapt to it on-the-fly, as one would have to in a real-world scenario. In the next section, we describe the small amount of variation that was implemented at the DRC (the Surprise task), and how teams designed their HRI to perform flexibly.

Surprise Manipulation Tasks

During each day of the Finals competition, a different Surprise task was used: the Lever the first day of competition, and the Plug on the second. Given the differing tasks each day, teams had to be competent at performing both in order to maximize their chance of obtaining a high score on either day of the competition.

To perform the Lever task, the robot had to make contact with the handle and effectively swing its hand/arm down to push the lever down. Some teams had to perform this action multiple times in order to move the lever down far enough to complete the task, but no fine manipulation or grasping was required. The task did not require fingers and was generally performed with a single static end effector.

As such, its subtask function was classified as FOM. Typically, teams performing this task maneuvered the robot avatar's end effector with the robot then planning trajectories (manipulation level of effort 2) after placing a model/template of the lever into the point cloud. The simplistic nature of the Lever task did not require much in the way of a feedback loop between the operator and the robot, aside from understanding whether or not the lever had been pushed down far enough for the task to be considered completed.

To perform the Plug task, the plug end and/or the cable it attaches to must be grasped, pulled back and out of the left receptacle, repositioned (taking into account the flexible nature of the plug), and inserted into the right receptacle. Removing the plug requires fingers/grippers to grasp it, unlike the Lever task. The subtask function for the Plug task was classified as SOM.

Operators typically placed models/templates of each receptacle on the wall and the plug/cable, then maneuvered the end effector of the robot avatar towards those models/templates such that the robot could plan trajectories (manipulation level of effort 2). The model/template for the plug end was typically static, meaning it did not move in the simulated display when the robot moved the physical plug in the real world. At least one team did update the placement of the model/template for the plug end once it was in the robot's gripper: AIST (Cisneros et al. 2016; see Figure 12).

In doing so, the operators used the situation awareness provided by a camera in the robot's other hand/arm for another viewing angle of the plug end's position within the robot's gripper to aid with task planning. With proper camera placement, point cloud generation, and fusion of the two, then interpreted by a trained operator, they were able to successfully perform the Plug task. This is another example of a balance between the operator and the robot: the operator remains in the loop and focused, providing contextual understanding of the situation (placing the model/template) while the robot plans what to do with the conveyed information (adjust its grip and plan trajectories to maneuver the plug to the right receptacle).

On average, completing the Lever task took 5:36 minutes and the Plug task took 11:25 minutes, more than twice the Lever task. Of all attempts at the Lever, 14.3% percent involved critical incidents other than falls or resets, compared to 29.6% on the Plug (Norton et al., 2017).

The Plug task had more opportunities for failure than the Lever, given that it had the possibility of being dropped. Compare this to the Door task; if the robot loses grip on the door handle, regrasping is possible using the same trajectory of movements that were used to grasp it the first time due to the handle being attached to the door. The plug cable was attached to a wall behind the receptacles, but if dropped, the end of the plug was now much lower to the ground near the robot's knees. This resulted in teams abandoning that task attempt; no robots were observed attempting to regrasp the plug once it had been dropped.

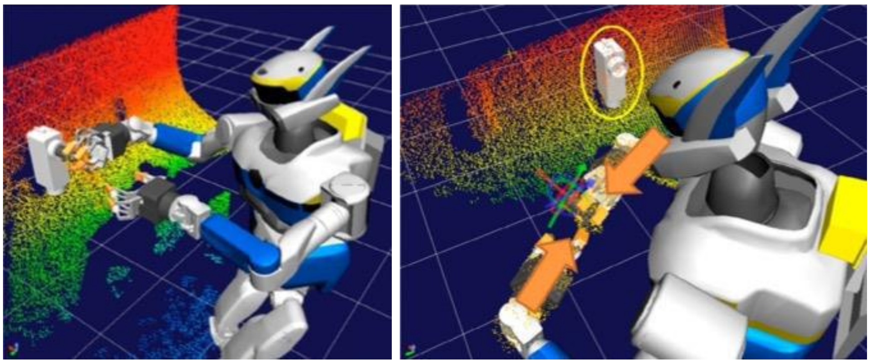


Figure 12. Team AIST's interface display while performing the Plug task. Left: A model of the plug and left receptacle can be seen in the point cloud with the simulated robot avatar grasping it. Right: The model of the plug end being adjusted in the robot avatar's hand. The model of the left receptacle has been removed and a model of the right receptacle has been added.
Images from Cisneros et al. (2016).

Regardless of these differences in complexity, teams largely used the same HRI techniques for both Surprise tasks. This approach is understandable given that teams were not made aware of which Surprise task was to be in play until the day before each run. It reflects the unknown nature of a real disaster scenario, wherein the tasks that must be performed to remediate the situation may not be known ahead of time, only learned in situ once the robot is downrange.

If operating in an environment that the operator is familiar with (e.g., known valve sizes, doorway dimensions, stair angles, etc.), a similar technique could be used. For operating in a more potentially unknown environment, similar techniques for balancing operator and robot responsibilities can be employed, but the type of information provided by the operator may need to be more malleable. Rather than a model/template of a specific object type, more robust interaction markers for planning grasps, finger placements, and approach angles could be provided by the operator for the robot to use for planning.

Discussion

From our analysis of the DRC Finals, we distilled a set of HRI characteristics that successful teams possessed, which we used as a basis to generate a set of design guidelines for HRI with remote, semi-autonomous humanoid robots. The guidelines are as follows:

- Balance the capabilities of the operator and the system to effectively perform the task.
- Keep the operator in the loop. Design HRI that requires steady interaction from the operator that supports and benefits from the autonomy of the robot.
- Maintain operator awareness of robot state and use consistent control methods that function regardless of bandwidth.
- Duplicate sensor fusion displays using different perspectives. Increased sensor fusion with common reference frames from an adjustable perspective is beneficial for remote teleoperation, and even more so by displaying two varying perspectives of the same data streams to increase the operator's situation awareness.
- Allow time for operator training and specialization. At this stage, humanoid robots are too complex such that general-purpose interfaces could be designed to be usable without training. (Norton et al., 2017).

The focus of the DRC was not on HRI development, but all teams did have to use an interface to control their robot and understand the environment around it. Given the competition design, the solutions they produced are developer-focused, not necessarily designed with consideration for transfer to novice users. With that said, there are still lessons to be learned and knowledge to be gained with respect to what was most effective. Through the analyses we have performed and the perspectives reviewed in this article, we can derive how the robotic advancements made at the DRC can inform human-robot teaming for real world response robotics.

Lessons Learned from the DRC to the Real World

Consider the target end user: a subject matter expert (SME). The target end user is a SME that is competently knowledgeable on the task being performed (e.g., navigating through a disaster area) likely with non-expert robot knowledge, nor sufficient amounts of robot training (Murphy, 2014). SMEs hold knowledge that cannot necessarily be easily programmed into a robot, not to mention the difficulty of equipping a robot with the perception needed to hold such knowledge.

The concept of using a humanoid robot was to provide a form that can operate in a world designed for humans. To that end, the operator should be able to use the robot to perform tasks in the same way that he/she would if he/she were physically there in place of the robot, at least with respect to physical motions like rotating valves and crossing debris. Given his/her expertise, the operator should also be focused on performing the task, not controlling the robot. SMEs can also provide the contextual understanding of a scene that a robot does not possess, keeping the operator engaged and in the loop, so long as effective data displays and control methods are used.

Using object models and templates. The use of models/templates as part of a robot control method is new for response robot HRI. A robot needs to obtain 3D depth information of its environment in order to use them properly, and 3D sensors are also not commonly implemented on response robots. The use of models/templates during the DRC was particularly effective, and was the closest interaction method we observed where the operator was interacting "directly" with the task, rather than controlling the robot. This follows one of the principles of efficient HRI as proposed by Goodrich and Olsen (2003): an efficient interface should enable the operator to "directly manipulate the world."

This method was demonstrated by some robots autonomously planning actions based on how the operator manipulated the virtual object to a desired end state (e.g., rotating a virtual model of the valve wheel which the robot uses to plan its actions to achieve that end state). The success of this technique was largely due to the fact that the design of the DRC tasks were known a priori. For use in a more unknown environment, these interaction methods need to be more generalized. Reducing the models/templates to a series of primitive movements and spatial relationships may enable this generalization.

Many of the models/templates used by teams were likely built on these elemental capabilities anyway, and some were even expressed in more general terms, but it is not clear how robust they are at being used outside of the DRC at this stage of development.

Representing the environment with 3D spatial data. Many of the data displays at the DRC used 3D spatial information and renderings, providing a very obvious connection to the real world that was being represented (i.e., if scaled and localized correctly, the dimensions of the robot avatar, the environment, the virtual models/templates of objects, and their proximity to one another is approximate to what it actually is in the real world). When only camera images and a robot avatar are provided (unfused, as typical with present day response robots), the operator is burdened with piecing together an understanding of the scene and building a mental representation to work from.

Some techniques that only use cameras to create more panoramic views for foveated vision have been implemented, like at the Trials competition (Yanco et al., 2015). However, when using only camera data, the operator must maneuver the robot's cameras and/or the limbs they are attached to in order to decipher the scene. Any robot movement introduces possibilities for error, or in the case of unstable humanoid robots, a risk of falling. In the case of using 3D representation data, the operator can instead manipulate the perspective of the interface display to build their understanding of the scene, requiring no additional maneuvering of the robot and thus posing less risk. This builds on recommendations from Keyes et al. (2006) for providing an exocentric view of the robot to increase an operator's situation awareness, by doing so while also making that viewing angle manipulable.

Presenting robot status information through a simulated display. When the robot status is conveyed through a 3D display, it is done in simulation and is dependent on the robot's understanding of its own position as informed by joint encoder values. When using a live video feed from a camera with an exocentric view of part of the robot, the operator watches the actual robot moving in the real world. In current response robot interfaces, there is very little simulation; everything being displayed is tied directly to the actual current state of the robot. While the 3D robot avatars at the DRC were able to remain updated throughout task completion, this was only possible due to the constant low-bandwidth line that was available regardless of communications degradations.

In the real world, where communications blackouts could occur, this simulated 3D avatar could still function in a way that camera images could not: continuing to simulate the position in which the robot was supposed to be. Continuing to provide this type of information to the operator, even though it might be inaccurate, may be a worthy endeavor if it keeps him/her engaged in the interaction. A combination of both types of data may be necessary so the operator can draw distinctions between the position the robot thinks it is in (the avatar display, possibly showing a "ghost") and where it actually is (the camera images). The displays can be fused together with a common reference frame to reduce cognitive workload, as was demonstrated at the DRC.

Continued operation in degraded communications. Continuing to operate in a degraded communications environment is a largely new concept for robot operations by emergency responders. Intermittent communications can be expected in a disaster scenario, but typically operators are acting incrementally whenever bandwidth is available, pausing when it is not.

Many of the effective HRI techniques discussed in this article can continue to be used when low bandwidth is available, which can increase operational tempos and keep the operator engaged. The introduction of these types of control loops will be new to today's end users, so we will need to be sensitive to its introduction. One challenge will be separating data that is "known" and that which is "estimated," and providing that awareness back to the operator. These all rely on some level of robot autonomy, which will also be a new concept. Current response robots barely have behaviors to autonomously assist an operator during teleoperation, like avoiding obstacles while being driven.

If we work towards implementing low-level autonomy into today's platforms, we can ease the cognitive burden on the operator and enhance the HRI of the human-robot team in the present. The autonomy demonstrated at the DRC on very advanced robots can be considered the state-of-the-art, representing the future of robotics, but it is a long way from becoming hardened. We can't expect to be effective in those high levels of autonomy if we don't have effective solutions at the lower levels first.

Designing HRI for varying levels of autonomy. Proper interface and interaction design is the means through which optimal function allocation can be accomplished. The various levels of autonomy observed across the systems at the DRC present many examples of how such a concept can be applied. This is highlighted even more through the nature of the different tasks, which many teams approached with their own strategy for optimal autonomy.

Regardless, some teams used an approach largely focused on autonomy, as witnessed via highly functioning robotic systems with limited human consideration. However, some teams reported taking a more human-centric approach to the overall system design. The latter demonstrates viewing the challenges presented in the DRC as an overall interaction scenario, as opposed to a single-domain problem (e.g., entirely a software problem, an autonomy problem, etc.) (Parasuraman et. al., 2000). Including requirements dictated by the interaction system as a whole (namely, human considerations) promotes an interchangeable balance in the way humans and machines collaborate during high-risk, chaotic missions.

Impact of HRI in the Context of the DRC

The ability to predict an outcome is a powerful method of testing a theory. In our analysis of the Finals, a (limited) model was developed by which a prediction of the results of the competition could be generated to show the true value of HRI in human-robot teaming. The model led to a 71% accurate prediction of the teams' scores within +/- 1 point, compared to an accuracy of 45% based on the teams' predictions of their own success and 31% if randomly guessed (Norton, et al., 2017). The model was based entirely on the findings from our evaluation of the Trials (Yanco et al., 2015), which enabled us to connect key performance metrics to specific interaction techniques.

However, there are other strategic impacts to human-robot team performance outside of the isolated HRI factors. Some of these have been presented previously throughout this article (e.g., robot stability). Our prediction included a second model of performance based upon these other factors, capturing aspects of the teams' capabilities such as robot stability, bandwidth adaptation, training, etc. As part of our evaluation, we conducted an independent assessment of all of the major performance-impacting factors to identify any clear patterns or themes, both HRI-related and non-HRI related.

These consisted of key issues with the robot, operator, interaction, and team strategy. For example, some teams did not effectively compensate for low-bandwidth communications and therefore could not complete tasks during periods of degraded communications, resulting in much slower times compared to teams that did. While these teams performed poorly overall, it is possible that the underlying autonomy, robot capability, and interface techniques would have been very effective otherwise.

These critical performance factors tended to fall into one of four team capability groups:

- Operator specific: Not a function of the robot's capability, autonomy, etc., but possibly something that could be mitigated through interface techniques to better inform the operator.
- Robot capability: Hardware, balance, etc. that cannot fundamentally be overcome to complete a task, or that were a contributing factor to critical lapses in performance.
- HRI specific: Limitations or gaps in the interaction that either prevented the operator from effectively controlling the robot or receiving feedback, or was a detriment to performance.
- Software, autonomy, and/or communications: Any limitations that would prevent the operator from using an otherwise effective interface and robot.

Some specific instances of these issues that were observed are captured in Table 4. The issues presented are by no means an all-inclusive list or even mutually exclusive, but rather are meant to capture a large part of the primary factors impacting performance.

Capability Group(s)	Critical Performance Factor, Description, and DRC-Specific Examples
Operator specific	**Logistical shortfalls**. Aspects of managing the tasks (such as following procedures correctly) that directly resulted in a fault or other issue impacting performance. Example: Missing a step in changing the robot's state, causing a critical error.
Robot capability	**Robot capability shortfalls**. Strictly based on the hardware capability of the robot. Examples: Inherent instability causing falls. Insufficient strength to rotate valve.
Robot capability	**System robustness shortfalls**. Not a capability, but rather hardware and software robustness that impacts ability to complete the tasks. Examples: Antenna breaks off because of a fall. Servo software failure. Sensor data stops coming through unexpectedly.
Robot capability HRI specific	**Fundamental gaps in robot situation awareness**. The sensor suite itself or the manner in which the feedback is presented (or lack thereof) is ineffective at providing information to the operator such that he/she can effectively control the robot. Example: Lack of motor feedback preventing understanding of motion issues.
HRI specific	**Poor execution of autonomy/interaction tools**. Autonomous processes and other tools that were ineffectively implemented, resulting in the operator making an error during task execution, rather than the error occurring in the autonomy or interface itself. Example: Missing a step in changing the robot's state, causing a critical error.
HRI specific	**Fundamental gaps in environmental situation awareness**. The sensor suite itself and/or the manner in which the feedback is presented (or lack thereof) is ineffective at providing situation awareness of the environment around the robot, causing critical incidents or forcing the operator to focus on developing SA, slowing down operations. Examples: Running into barriers with the vehicle. Attempting to turn the valve without actually having gripped it first.
Software, autonomy, and/or comms	**Perception-based issues**. Not autonomy, but inaccurate localization and identification. Includes ineffective implementation of what would normally be effective perception (meaning this is not related to the ability of the perception algorithms themselves or the sensors). Example: Repeated failed attempts to autonomously execute door task because of error in autonomous tasking
Software, autonomy, and/or comms	**Lack of supporting cognitive autonomy**. This is in the form of scripts for pre-set behaviors, closed-loop control through simulated objects, etc. Example: Extended periods of time when the operator is manually operating the robot to grasp/manipulate something.
Software, autonomy, and/or comms	**Poor compensation for low-bandwidth**. Impairing the ability of the operator to control the robot, either because he/she is unable to receive effective feedback because it is limited too much or because he/she is unable to send commands. Example: Extended periods of time when operation of the robot is significantly slowed because the operator is waiting for high-bandwidth transmission.

Table 4. Some of the critical performance factors observed during the Finals competition.

Considering all of the impacts to performance, and the characteristics of operations in real world scenarios, the following lists key capabilities that we believe will be required of robotic systems to enable effective team performance:

- Third order manipulation (TOM): This is a characteristic of operating in human-centric environments and is required to enable robots to interact with human-based systems as an effective extension of the user.
- Wide range of interaction tools: Enables a system to be more capable of functioning in realistic disaster scenarios that have unknown and unpredictable environments.
- Variable robot capabilities and associated interaction tools: The ability to change morphologies to match the scenario, ideally in a dynamic way, changing on-the-fly as needed.
- Variable levels of autonomy: Enables a system to interact with the environment without reliable communication channels while maintaining the connection to the operator's own expertise and contextual awareness.
- Connection between the environment and control techniques: As exemplified by the simulated object/live object interaction technique, this enables a 'mutual understanding' between the robotic system and operator.

Conclusion

This article presents a series of perspectives on the factors influencing human-robot team performance at the DARPA Robotics Challenge Trials and Finals competitions, as informed by our evaluation of the events (Yanco et al., 2015; Norton et al., 2017). While not an exhaustive list of all the recommendations for effective HRI techniques for response robots, humanoid or otherwise, those provided are examples of how the results of the competition can continue into the research community and beyond. Effective human-robot teams are still needed using the highly teleoperated robots of today, let alone more capable and advanced robotic systems in the future.

While the focus of our studies was on effective HRI techniques for remote humanoid robot control, many factors had an impact on performance, including the operator's role and expertise, the varying robot morphologies, and the context of the event. Some of these critical performance factors, like robot balance instability and poor autonomy for object identification, can continue to be developed and eventually solved with better algorithms and more accurate sensors. However, all of these factors can only be harnessed through effective HRI, and cannot be considered after the fact. HRI design is an important component to making an effective human-robot team, if not the most essential.

Acknowledgements and References

Acknowledgements

This research has been supported in part by DARPA under W31P4Q-13-C-0136. The views, opinions, and/or findings expressed are those of the authors and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government.

The authors would like to thank DARPA, the staff members of the DRC, and all affiliated personnel that enabled this research. Thank you to Josh Peri for his assistance with data analysis. In particular, thank you to all of the teams that consented to participate in our studies over the years.

References

Atkeson C G, Babu B P W, Banerjee N, Berenson D, Bove C P, Cui X, DeDonato M, Du R, Feng S, Franklin P, Gennert M (2015) No falls, no resets: Reliable humanoid behavior in the DARPA robotics challenge. In: 2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids), 623-630.

Blitch J G (2003) Adaptive mobility for rescue robots. In: Proceedings of SPIE Sensors, and Command, Control, Communications, and Intelligence (C3I) Technologies for Homeland Defense and Law Enforcement II (5071):315-321.

Chen J Y, Haas E C, Barnes M J (2007) Human performance issues and user interface design for teleoperated robots. In: IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 37(6):1231-1245.

Cisneros R, Nakaoka S I, Morisawa M, Kaneko K, Kajita S, Sakaguchi T, Kanehiro F (2016) Effective Teleoperated Manipulation for Humanoid Robots in Partially Unknown Real Environments: Team AIST-NEDO's Approach for Performing the Plug Task during the DRC Finals. In: Advanced Robotics 30(24):1544-1558.

Endsley M R (1995) Towards a Theory of Situation Awareness in Dynamic Systems. In: Human Factors, 32-64.

Goodrich M A and Olsen D R (2003) Seven principles of efficient human robot interaction. In: IEEE International Conference on Systems, Man and Cybernetics (4):3942-3948.

Jameson S M (2001) Architectures for distributed information fusion to support situation awareness on the digital battlefield. In: 4th International Conference on Data Fusion, 7-10.

Keyes B, Casey R, Yanco H A, Maxwell B A, Georgiev Y (2006) Camera Placement and Multi-camera Fusion for Remote Robot Operation. In: Proceedings of the IEEE International Workshop on Safety, Security and Rescue Robotics, 22–24.

Keyes B, Micire M, Drury J L, Yanco H A (2010) Improving Human-Robot Interaction through Interface Evolution. In: Human-Robot Interaction, Chugo D (ed), 183-202.

Lim J, Lee I, Shim I, Jung H, Joe HM, Bae H, Sim O, Oh J, Jung T, Shin S, Joo K (2017) Robot System of DRC-HUBO+ and Control Strategy of Team KAIST in DARPA Robotics Challenge Finals. In: Journal of Field Robotics 34(4):802-829.

McGill S G, Yi S J, Yi H, Ahn M S, Cho S, Liu K, Sun D, Lee B, Jeong H, Huh J, Hong D (2017) Team THOR's Entry in the DARPA Robotics Challenge Finals 2015. In: Journal of Field Robotics 34(4):775-801.

Murphy R R (2014) International Cooperation in Deploying Robots for Disasters: Lessons for the Future from the Great East Japan Earthquake. In: Journal of the Robotics Society of Japan 32(2):104-109.

Nagatani K, Kiribayashi S, Okada Y, Otake K, Yoshida K, Tadokoro S, Nishimura T, Yoshida T, Koyanagi E, Fukushima M, Kawatsuma S (2013) Emergency response to the nuclear accident at the Fukushima Daiichi Nuclear Power Plants using mobile rescue robots. In: Journal of Field Robotics 30(1):44–63.

Nielsen C W, Goodrich M A, Ricks R W (2007) Ecological interfaces for improving mobile robot teleoperation. In: IEEE Transactions on Robotics 23(5):927-941.

Norton A, Ober W, Baraniecki L, McCann E, Scholtz J, Shane D, Skinner A, Watson R, Yanco H (2017) Analysis of Human–Robot Interaction at the DARPA Robotics Challenge Finals. In: The International Journal of Robotics Research 36(5-7):483-513.

Oh P, Sohn K, Jang G, Jun Y, Cho B K (2017) Technical Overview of Team DRC-Hubo@ UNLV's Approach to the 2015 DARPA Robotics Challenge Finals. In: Journal of Field Robotics 34(5):874-896.

Parasuraman R, Sheridan T B, Wickens C D (2000) A model for types and levels of human interaction with automation. In: IEEE Transactions on systems, man, and cybernetics-Part A: Systems and Humans 30(3):286-297.

Steinfeld A, Fong T, Kaber D, Lewis M, Scholtz J, Schultz A, Goodrich M (2006) Common metrics for human-robot interaction. In: Proceedings of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction (HRI) 33-40.

Stepanova E R, von der Heyde M, Kitson A, Schiphorst T, Riecke B E (2017) Gathering and Applying Guidelines for Mobile Robot Design for Urban Search and Rescue Application. In: International Conference on Human-Computer Interaction 2017 562-581.

Yanco H A, Norton A, Ober W, Shane D, Skinner A, Vice J (2015) Analysis of Human-Robot Interaction at the DARPA Robotics Challenge Trials. In: Journal of Field Robotics 32(3):420-444.

BOSTON ENGINEERING™
Imagine the Impact™

These SBIR data are furnished with SBIR rights under Contract No. W31P4Q-13-C-0136. For a period of 4 years, unless extended in accordance with FAR 27.409(h), after acceptance of all items to be delivered under this contract, the Government will use these data for Government purposes only, and they shall not be disclosed outside the Government (including disclosure for procurement purposes) during such period without permission of the Contractor, except that, subject to the foregoing use and disclosure prohibitions, these data may be disclosed for use by support Contractors. After the protection period, the Government has a paid-up license to use, and to authorize others to use on its behalf, these data for Government purposes, but is relieved of all disclosure prohibitions and assumes no liability for unauthorized use of these data by third parties. This notice shall be affixed to any reproductions of these data, in whole or in part.

Company/university logo or name used in connection with DRC events does not suggest or imply endorsement, sponsorship, promotion, association or affiliation with the U.S. Government or DARPA.

Please contact us for additional information:

Email us at:
info@boston-engineering.com

Visit our website at:
www.boston-engineering.com