Assistive Mobile Manipulation: Designing for Operators with Motor Impairments

Phillip M. Grice and Charles C. Kemp
Department of Biomedical Engineering
Georgia Institute of Technology
Atlanta, Georgia 30332
Email: phillip.grice@gatech.edu

***Abstract*—Individuals with severe motor impairments can operate mobile manipulators for physical assistance with activities of daily living. However, relatively little attention has been given to designing these systems with the explicit aim of reducing operator workload and increasing usability. This is especially important for operators with limited input bandwidth as a result of motor impairments. To address this challenge, we have worked regularly with Henry Evans, an individual with severe motor impairments, and Jane, his wife and primary caregiver. With their collaboration, we have developed an assistive robotic system that can be teleoperated via a modern web browser and a single-button mouse or equivalent. This increases accessibility for individuals with motor impairments already using assistive Human-Computer Interfaces. Our system uses a video-centric approach to reduce complexity, provides clear, easily switchable modes of interaction, and enables the use of task-level planning and task-relevant undo. We performed a preliminary evaluation of this system via remote operation by Henry Evans performing the Action Research Arm Test (ARAT), a clinical test for manipulation capability. In this evaluation, the system provided a clinically significant improvement relative to Henry's own capability.**

I. Introduction

Many people have severe motor impairments that make it difficult for them to perform activities of daily living. For those without cognitive impairments, assistive mobile manipulators have the potential to provide physical assistance, while the user is available to provide guidance and understanding to support the robot's performance. We have worked closely with Henry Evans, an individual with severe motor impairments, and Jane Evans, his wife and primary caregiver, to develop this kind of assistive capability for robots.

Throughout our development process, we have enabled novel assistive physical interactions, but these systems often created a high cognitive workload for users, reducing usability. Drawing upon guidelines for usable and accessible design [31, 30, 20], we have worked to alleviate these challenges, developing novel user interface methods in the process. We have also developed a task-level planning system to provide cognitive support for users during complex or long-running tasks. Using this task-level planning framework, we also present an implementation of undo that is relevant to real-world tasks, including cases where the complete system state is not immediately and/or directly observable or controllable by the robot. Using our robotic system remotely, Henry Evans was able to achieve a clinically significant improvement in his performance on the Action Research Arm Test (ARAT), a standardized clinical test of manipulation.

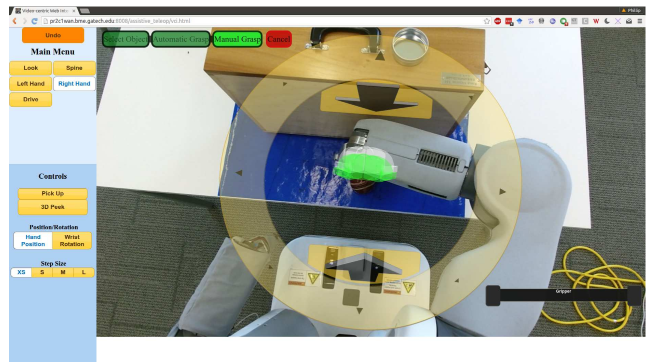


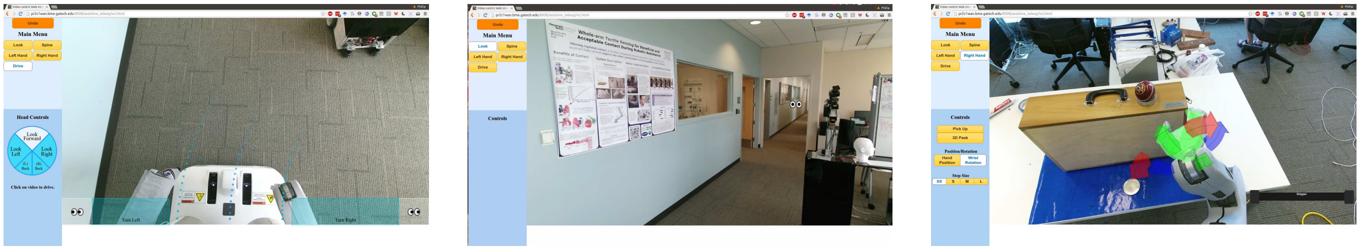
Fig. 1: Our novel, web-based interface for teleoperation by individuals with severe motor impairments.

Additionally, while the interface is designed for use by individuals with motor impairments who have difficulty providing inputs to a computer system [28], many of the methods we developed are applicable to non-motor-impaired operators, and so are representative of universal design [16]. We have made the complete source code for the system open-source and freely available [8], but it remains under active development.

II. Related Work

This work is a direct continuation of the Robots for Humanity Project, begun in 2011 [3]. In particular, it is motivated by our systems for enabling self-care tasks around the head [11] and for whole-arm tactile sensing for safe physical interaction [9], developed as part of Robots for Humanity. These systems highlighted the challenges of designing interfaces for motor impaired operators. We also draw on lessons from work of other collaborators on the project, including efforts on control of a mobile manipulator for in-home tasks [5] and for accessible pointing interactions [24].

[18, 21, 40, 42, 41] and [12] present various techniques to improve interaction between individuals with impairments and robotic systems, and [2] and [19] do so for robotic systems in general. However, these systems tend to focus on grasping, mobility, or telepresence, rather than general mobile manipulation for potentially arbitrary tasks. [37] and [4] present especially condensed, actionable recommendations for the design of robot interfaces. Recently, the DARPA Robotics



(a) Driving Mode

(b) Looking Mode

(c) Arm Rotation Controls Mode (right arm)

Fig. 2: The web-based user interface.

Challenge has produced a variety of control interfaces for operation of mobile manipulators by able-bodied, expert users for performing diverse tasks [44].

We draw our conceptual structure for hierarchical task-planning from [36]. Regarding undo, [43] discusses the use of history list systems in robotics, and indicates the difficulty associated with providing undo capabilities at a semantic level, especially given 'impossible to undo' actions. More recently, [35] attempts to address this issue for manipulation tasks by introducing an 'oracle' state into a state machine that can revert the system to an earlier available state and query human users for assistance as necessary.

III. Long-term User Involvement

Henry Evans is a significant contributor and collaborator in our work, and the originator of the project name "Robots for Humanity." Henry suffered a brain-stem stroke in August 2002, which has left him with severe motor impairments. Henry operates a computer using an assistive head-tracking device [27] which tracks the motion of a reflective dot on his glasses to direct cursor movement on the screen. In addition, Jane Evans, Henry's wife and primary caregiver, also provides regular feedback and insight.

Since early 2011, we have worked with Henry regularly to develop and improve various systems for enabling individuals with severe motor impairments to control a general purpose mobile manipulator for performing assistive tasks. Using our web-based interface, Henry is able to evaluate new design iterations remotely from his home in California. Over the past 1.5 years he has done so approximately monthly. Henry's perspective as a motor impaired user provides valuable insights, improving our effectiveness in designing technology for real-world use cases. Additionally, since late 2013, Henry and Jane have also attended weekly group meetings to enhance collaboration and ensure that their perspective is integrated into ongoing work. Jane's insight as a caregiver is important, as it has been shown that including not only users, but also caregivers, in the development of assistive technologies improves adoption [22]. This is a point that Henry emphasizes in what he calls 'the Caregiver Principle,' which states that assistive technologies must make the lives of caregivers easier, or else the technology will not be adopted (unless ordered by a physician).

IV. System Description

A. Accessible Input via Human-Computer Interfaces

A primary challenge in designing assistive technologies for individuals with motor impairments is the reduced ability of these users to provide computer input through traditional means. To address this, we design our robotic system for use by individuals who can control a computer using any device that emulates a single-button mouse. Many individuals with motor impairments already use computers via a variety of assistive Human Computer Interface (HCI) devices which allow this level of interaction. Available inputs include head trackers [27], eye trackers [38], sip-and-puff devices, and others [1]. This provides a clear design requirement and allows us to focus our efforts on enhancing the system without working to support additional inputs. Additionally, the system is directly usable by able-bodied users while still providing access to a portion of the motor-impaired population.

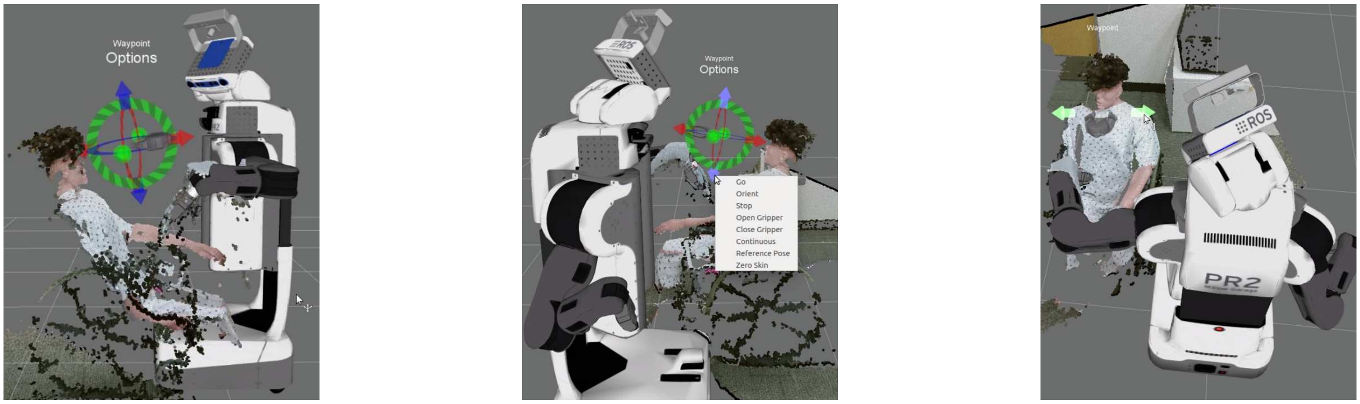
B. Web-based Interface

Users control our system via a web-based interface which can run in any modern web browser (tested in Chrome, Firefox, and Opera, see Figs. 1, 2). This reduces the complexity for the user, as they are not required to download or install software. Instead they can use a web browser with which they are likely already familiar. This simplifies support and development, as cross-platform browser behavior is generally consistent across operating systems. Modern web standards have also reduced the challenges associated with cross-browser compatibility. Our system makes use of the open-source Robot Web Tools [39] suite to stream data and video between the interface and robot in real time.

C. Video-centric Interface

In our prior study [9], even able-bodied users with experience in virtual 3D modeling had difficulty controlling a robot effectively using a 3D-rendered virtual interface (the ROS RViz suite [7], see Fig. 3), despite training and limited practice. As the task was easily understood, and the robot consistently performed as commanded, we believe that the RViz-based interface made it difficult for users to operate the system.

While potentially powerful, this interface presents a few distinct challenges. First, the virtual camera perspective requires additional mental effort in understanding the scene. For example, moving the virtual camera results in a novel perspective



(a) Default Display (b) Right-click Menu Active (c) Position controls active

Fig. 3: Multiple views of the same scene from the RViz-based interface used in [9], highlighting challenges that motivate this work.

after each move. Additionally, multiple important commands and modes are hidden from the user. Specifically, right and middle clicks each perform necessary functions, but perform different functions depending upon where the click occurs on the scene. Further supporting these conclusions, [25] uses a similar RViz-based interface, and notes that "the operator's comfort with a general 3D GUI and related operations such as positioning a virtual camera proved to be very important" for effective task performance.

Our current system addresses these challenges with a number of specifically designed features.

1) Video-centric Display: The new web interface is designed around a large display of the live, color video feed from the Kinect One camera on the head of the robot (see Fig. 1). Each interface mode augments this primary display in different ways, but all retain a consistent visual perspective and structure. By restricting the user's perspective on the space to the view from the robot's head, we eliminate the requirement for the user to manually position and orient a virtual camera in 3D space. The consistent perspective also aids the user in assuming the role of the robot, as this is similar to the perspective from which individuals experience their own world in daily life.

2) Depth Perception: A significant challenge introduced by this single-camera perspective is the lack of depth perception it provides. While display technologies such as virtual reality head-mounted displays or 3D displays may present effective solutions, these techniques currently require specialized hardware, such as glasses, which would limit the accessibility of our system and increase cost. Instead, we have attempted to provide depth information in a few different ways. First, we added a pair of small LED lights to the inside of the gripper on a battery-powered, custom printed circuit board. These lights are sufficiently bright that their reflection is visible off of many objects in the camera view when those objects are directly in front of the gripper (i.e. in a position where they can be grasped).

We also provide a novel, augmented reality style '3D Peek' which uses the RGB-D data from the Kinect One sensor to

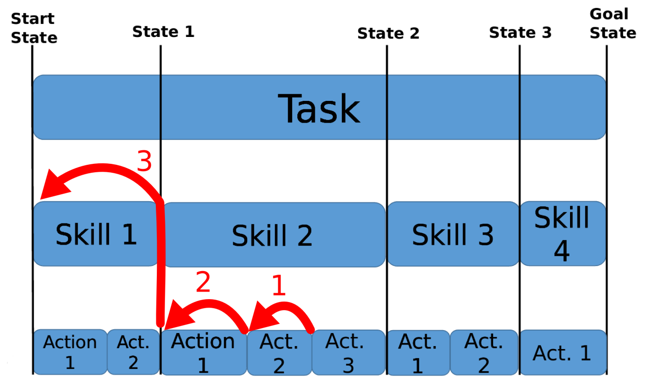


Fig. 4: Conceptual hierarchy of tasks used in designing task PDDL domains. Orange path shows undo reverting to prior task state when necessary, rather than directly reversing prior actions.

present a 3D view of a region of interest near the robot's hand. When the user clicks the '3D Peek' button, the display renders a live, filtered point cloud, aligned overtop of the video display in the interface. The virtual camera then shifts downward to the same vertical height as the gripper over 0.4 seconds. The virtual camera holds this position for 2.8 seconds, and then shifts back to match the live camera view over 0.4 seconds before the point cloud is removed.

This effect allows the user to virtually view the region near the robot's hand from the side, rather than from above, where the camera view is typically located during manipulation. The rendered point cloud is aligned with the camera view at the beginning and end of the sequence, and the darkened camera view remains in the background throughout the sequence. This overlay gives the impression of the region around the hand being lifted up at an angle for inspection by the viewer and then being replaced. By using a brief, scripted visualization, users can quickly ascertain the relative depth of items visible to the camera without manually positioning and orienting a virtual camera.

3) Modal Interface: Because our target users have a limited ability to provide input to the system, we use a modal interface, presenting a small sub-set of the available controls at any one time. While modal interfaces allow multiplexing of the user's available input, they also add the requirement for mode switching. This can introduce delays and reduce usability [13]. In our system mode switching occurs rapidly once the user selects the desired mode from a set of constantly visible buttons. This is possible because the robot's back-end system components (i.e. controllers, perception systems) are all active concurrently, so the modes and associated switching are only present within the interface.

Additionally, modal interfaces introduce opportunities for mode errors [31], where a user issues a command intended for one mode while in a different mode, with an undesired result. Increasing user awareness of the current mode can help alleviate mode errors. To this end, each of our interface modes are visually distinct (see Fig. 2).

In the 'Driving' mode, the video remains unobstructed except for lines showing the path the robot will take based on the cursor location, including turning in place. This is similar to other remote driving interfaces such as the Beam by Suitable Technologies®, which is used regularly by individuals with motor impairments [6]. The 'Driving Mode' also allows the user drive the robot along a straight line in any direction, taking advantage of the holonomic base of the PR2, a feature not found on 2-wheeled devices such as the Beam. The camera view is also unobstructed in the 'Looking' mode, and the mouse cursor over the video display is replaced by an icon showing a pair of eyes. In the 'Right Arm' and 'Left Arm' modes, the interface displays controls to direct the end-effector overlaid directly around the gripper, and the head automatically tracks the appropriate end-effector, centering it in the view.

These distinctions seek to limit mode errors by making the modes, and especially the clickable controls within each mode, visually distinct. The interface also provides lock-outs with respect to many controls for the modes that are not currently selected, further limiting the risk from mode errors.

4) End-effector Control: All arm motions use our model-predictive controller [17, 9] with low stiffness gains at the joints. To command the position of the end-effector, the user clicks on a wide ring overlaid on the image around the gripper (see Fig. 1). Clicking anywhere on the ring commands the gripper to move in that direction parallel to the floor by one of four user-selectable step sizes (2.5cm, 5cm, 10cm, and 25cm). Buttons inset within the ring show arrows pointing upward and downward that allow step-wise commands in the vertical direction. This novel ring layout allows the user to leverage both degrees of freedom in the mouse position to command the gripper along arbitrary directions in the horizontal plane. We placed the ring control on the horizontal plane to aid in table-top manipulation.

The orientation of the end-effector is another feature of the robot that is challenging to control intuitively on a 2D interface. After testing multiple designs, we created an augmented reality interface, which overlays the camera view with semi-transparent, virtual, 3D curved arrows that track the gripper in space (see Fig. 2 (c)). When the user clicks each arrow, the system adjusts the orientation of the fingertips by moving the wrist in the corresponding direction. By tracking the gripper, the arrows always point in the direction of motion they command, reducing the mental workload associated with mapping static controls to the gripper as it moves.

Extending the augmented reality approach, we display a green, semi-transparent 3D virtual model of the robot's gripper at the goal pose when a command is received by the arm controller, and remove it once the goal is reached. This allows the user to understand the behavior of the controller and preview the arm's motion, as recommended in [4]. Lastly, the gripper can be opened or closed using parallel sliders in the bottom left or right corner in the arm control modes. While closing, the gripper attempts to grasp items gently but securely using the open-source implementation of [34]. These features seek to leverage the benefits of direct manipulation interfaces [15], while leaving the video relatively clear around the gripper to provide visual feedback for manipulation.

D. Task-level Planning

Even cognitively unimpaired individuals are susceptible to being overwhelmed by complex tasks. This may be especially true for robotic teleoperation, where users must direct the robot through steps that would not normally be part of a human's performance of a task, such as explicitly indicating an object to be manipulated. In our system for shaving the face of a user with motor impairments [11], we developed a number of independent modules for specific sub-tasks. One challenge we identified with this system was the effort required for the user to coordinate and sequence these actions. After significant practice and testing in his home, Henry Evans agreed that he could use the system effectively, but disagreed that the system was easy and intuitive to use. He also reported high mental demand and effort on a NASA Task Load Index (TLX), and "reported that he would prefer a step-by-step 'wizard-like' process as opposed to a set of distinct tools" [11].

To address this, we have developed a task-level planning system based on the Planning Domain Definition Language (PDDL) [29] and the Fast-Forward (FF) planner [14]. The PDDL description of a given task domain provides the robot with specific, actionable knowledge regarding the states of the world that are relevant to that task, and the actions it can take to alter those states. With this information, the robot is able to quickly plan a sequence of high-level actions to complete a given task. This high-level plan allows the robot to guide the user through the task, providing cognitive support with the aim of reducing cognitive workload. The robot also aids the user via automatic mode switching, presenting the appropriate interface mode for each step, to reduce the switching cost associated with a modal interface. This is intended to spare the user's cognitive resources for performing other functions, such as properly identifying relevant goals and accurately monitoring progress.

Based on [36], we defined a hierarchy for decomposing task understanding (see Fig. 4). This hierarchy includes tasks, skills, and actions. Tasks represent high-level goals, often activities of daily living (ADL's) or instrumental activities of daily living (IADL's). Each task is completed by performing a sequence of one or more independent skills. A skill is a sequence of one or more actions that changes the state of the task based on the PDDL domain description. These task-level state changes are either not directly observable or controllable by the robot (i.e. dropping an object, pouring the contents of a cup, etc.) In contrast, the actions of which skills are composed are immediately both observable and controllable by the robot (i.e. motions of the robot's joints, handling of software-only states, etc.)

One must design PDDL domains for each complete task to allow planning at a high level of abstraction. Furthermore, a task domain may represent a component skill within another task domain. For instance, we treat 'pick' and 'place' each as their own tasks, but both are skills within the 'pick-and-place' task, which is specified at a higher level of abstraction. This hierarchical structure allows for modification of the component sub-domains within a given domain, and allows the user to focus only on the immediately active sub-task. During operation, the interface displays the current task sequence in the top-left corner of the view, with the current action within the sequence highlighted, and prior and future actions in the sequence dimmed, but visible (see Fig. 1). Additionally, a cancel button allows the user to cancel the current task plan at any time.

E. Undo

Donald Norman states that "perhaps the most powerful tool to minimize the impact of errors is the Undo command in modern electronic systems" [31]. In software-only systems, undo can be readily implemented because it is often possible to fully observe and control the complete state of the system. However, because robots interact with the physical world, and have potentially noisy and/or unreliable sensing and actuation, it is not always clear what it would mean to undo an action, or how to accomplish this in general.

Despite these challenges, undo could be a useful capability in the context of robot teleoperation. To support the use of undo, we define the state of interest as the state of the robot itself (joint positions) for low-level actions in our task hierarchy. In these cases, the robot has essentially complete knowledge of and control over the state, and undo is relatively simple.

For undo in the context of tasks involving interaction with the environment, we instead define the state of the system based on the PDDL domain description for that particular task. We then express undo as a command to return the system to the most recent prior state within the task domain, and re-plan in the task domain to find a sequence of actions to return to that state (see Fig. 4, numbered sequence). For example, in the pick-and-place domain, opening the gripper to release an object results in the task state transitioning from 'OBJECT GRASPED' to 'OBJECT PLACED.' Because the task-level state has changed, a subsequent undo command will not simply re-close the gripper (which will often not re-grasp the object, e.g. if it was dropped away from the gripper). Instead undo will re-plan with the goal state of 'OBJECT GRASPED,' and so activate the complete 'pick' sub-task, including re-identifying the object's location. This provides a mechanism for users to command an undo action with respect to a complex task which is both meaningful and actionable, even if a complex sequence of actions may be required to undo the effects of a prior action.

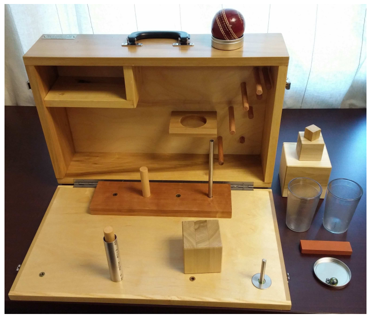
V. Evaluation

We performed a preliminary evaluation of our system via remote teleoperation by Henry Evans. We obtained informed consent from Henry and approval from the Georgia Tech Institutional Review Board. Henry teleoperated a PR2 robot in the Healthcare Robotics Lab at Georgia Tech, Georgia, USA from his bed in his home in California, USA. Henry accessed the interface via Google Chrome on an Apple laptop placed on an overbed table. He controlled the mouse cursor position using a Tracker Pro [27] head-tracking mouse and clicked by using his left thumb to activate the button of a traditional mouse placed in his left hand. A researcher also remotely accessed a BeamPro robot [6] in Henry's home to provide verbal instructions.

We evaluated the system via the Action Research Arm Test (ARAT), using a commercially available kit [33]. The ARAT is a clinical measure of manipulation capability typically used to assess upper limb recovery in patients with cortical damage [26]. The ARAT consists of 19 sub-tasks, each scored from 0-3, where 0 is a complete inability to perform the task, and 3 is 'normal' human performance, for a possible score range of 0-57. The test is typically administered independently for each arm of a subject, but we test only the right side with the robot to reduce testing time, as the controls are identical for the left arm. We administered the ARAT according to [45], which specifies many details where ambiguity is present in Lyle's original paper [26]. We increased the distance from the robot to the near edge of the table to allow the robot's elbow clearance from the shelf; used plastic beads in place of water for the pouring task (to protect the robot in case of spills); and used a mannequin seated in a wheelchair for the tasks requiring Henry to bring his hand to locations around his head. We followed [45] closely so the achieved results would be meaningful with respect to ARAT results from other contexts, and interpretable to others familiar with the test, especially clinicians.

Henry completed the ARAT twice. On May 11th, 2016, we administered the ARAT strictly according to [45], including a 1 minute cutoff for completion of each sub-task. On June 9th, 2016, we administered the test again, without a time limit and skipping the 10cm block, washer, and ball-bearing tasks, as the robot hardware cannot grasp these items. Because of the robot's slow speed, the 1 minute time limit creates significant temporal pressure, which is not typically present during ADL's. We therefore also administered the ARAT without this constraint, which still meets Lyle's original instruction of giving 2 points to someone who "can complete the test but takes abnormally long" [26].

(a) PR2 setup for ARAT



(b) ARAT Test Kit from [33]

Fig. 5: Remote evaluation using the ARAT.

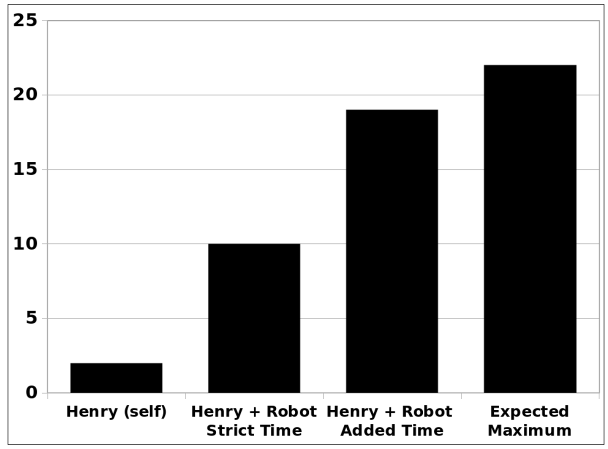


Fig. 6: Henry's performance on the ARAT.

VI. Results

After reviewing each of the ARAT items with Henry and Jane, Henry self-reported that he believed he would achieve a score of 2 with his left arm (as he can raise his left hand to his mouth), and 0 with his right arm, if completing the test himself. Immediately before testing with the robot, we gave Henry the opportunity to practice using the task-level planning for pick-and-place tasks, which he had not previously used. Henry practiced, primarily using the task-level planning, for approximately 30 minutes before he asked to proceed. Henry then completed the ARAT, achieving a score of 10 (see Fig. 6). Within the 1 minute time limit for each sub-task, Henry successfully grasped, lifted, and placed the 5cm wooden cube and the cricket ball (2 pts each), grasped and lifted the tumbler of beads and 2.25cm tube (1 pt each), reached part way to the top of the head and the mouth (1 pt each), and fully reached the back of the head (2 pts). Henry attempted to use the task-level planning and automated picking and placing behaviors for six of the 19 tasks, and earned 5 of his 10 total points during 3 of these 6 tasks. Henry then completed a NASA TLX [10] and a custom questionnaire regarding his use of the interface.

Henry later completed the ARAT without the time constraints, with no additional practice, and earned a score of 19. Henry successfully placed the 2.5cm, 5cm, and 7.5cm wooden cubes, cricket ball, and sharpening stone (2 pts each). He also successfully poured the beads between the cups, reached all three locations on the mannequin's head (2 pts each), and lifted the 2.25cm tube (1 pt). Henry performed each of these tasks (including time spent before giving up when trying to place the 2.25cm tube, grasp the 1cm tube, or grasp the marble) in an average of 197.4s $\pm$ 57.1s (mean $\pm$ std, range: 110s – 298s).

On the NASA TLX, after the time-constrained test, Henry reported very low Physical Demand and Performance, low Mental Demand and Frustration, low-to-moderate Effort, and very high Temporal Demand, indicating that Temporal Demand and Performance were the greatest contributors to his workload. His overall weighted workload score after was high, at 70/100. Henry stated that "time was the biggest hurdle, which is an artificial constraint, because ... for quadriplegics, how long something takes is irrelevant. What matters is the end result," which partially motivated the second test. After the second test, Henry reported very low Physical and Temporal Demand and Frustration, and low Mental Demand and Effort, along with high Performance. His overall weighted workload score after the second test was low, at 20/100. This is lower than reported by individuals performing reading comprehension and word-search tasks [32].

Henry did not use the undo feature during the performance of any of the components of the ARAT during either test, explaining that "[he] was not convinced it would save time." He also states that he used the 3D Peek feature "often ... to see the height of the gripper."

We asked Henry to compare his experiences with three different interfaces he has used to control the PR2 for manipulation tasks (the 3D RViz visualization, a prior version of the web interface, and the interface presented here) via ten 7-point Likert items. Henry expressed no preference among the three "for understanding the environment in 3D," stating that "none are particularly good at 3D," and that the "biggest weakness of the New Web Interface is that the main camera is no better at 3D." For 'controlling the orientation of the grippers,' Henry disagreed that he preferred RViz, and strongly disagreed that he preferred the old web interface, to the interface presented here, stating that the "New Web Interface is especially strong at this." Henry strongly disagreed that he preferred either RViz or the old web interface to the new interface for 'performing long, complex tasks,' indicating that the "New Web interface is much more natural." Additionally, "for overall ease of use" and "for overall usefulness," Henry strongly disagreed that he preferred either RViz or the old web interface to the new interface, and states that "[he is] dying to try the [New Web Interface] at home," and that "RViz felt like a science project by comparison."

VII. Discussion

[23] establishes a change of 10% of the maximum score, or 5.7, as the minimal clinically important difference when evaluating stroke recovery using the ARAT. Thus, based on scores of 10 and 19 (when relaxing time constraints), the use of the robot would likely make a clinically significant improvement in Henry's ability to manipulate his environment.

The artificial time constraint in the strictly administered ARAT was the greatest source of workload as reported in

Henry's NASA TLX results, and he commented that "the time was so short and the robot so slow that if even one thing went wrong ... it was over." Given additional time for each item, Henry achieved a higher score on a test still within Lyle's original design, and likely more representative of everyday use.

While Henry did not express a preference between the interfaces for interpreting the world in 3D, he did use the '3D Peek' feature regularly to gauge the height of the end-effector relative to the task objects, stating that it "immediately tells you whether you have to raise or lower the gripper." This consistent use indicates the value of this capability in helping to overcome the lack of depth perception. The absence of any preference relative to the full 3D rendering of the RViz interface indicates that this system conveys depth information effectively.

Henry used automated task planning regularly in both test scenarios. In cases where the automated grasping was successful, Henry generally scored well, earning half of his earned points in these tasks in the time-constrained test. By failing over to manual control within the task sequence, in multiple instances Henry was able to benefit from partially-autonomous task execution. When it failed, he would often complete portions himself, and then allow the automated system to resume control once the specific difficulty was overcome. This represents an interesting form of task-centric shared control. Task-level planning may provide additional benefits when supporting users during more complex tasks and by enabling the task-relevant undo function.

Henry did not use the undo function, as he did not believe it would speed up the overall performance at any point. Despite this, it may prove more helpful in the context of more complex tasks.

VIII. Conclusions

Our preliminary evaluation demonstrates that the system can improve the performance of an expert user with severe motor impairments by a clinically significant margin on a clinically validated manipulation test. We have presented our system for assistive teleoperation of a mobile manipulator by individuals with severe motor impairments, and highlighted the novel design features intended to reduce operator workload and so enable more effective control. To the same end, we have also also developed a high-level task-planning system and task-relevant undo for robotic systems with the aim of enhancing operator performance in more complex tasks.

IX. Acknowledgments

The work presented here was supported in part by an NSF Graduate Research Fellowship, NSF CAREER Award IIS-1150157, NIDILRR Grant H133E130037 as part of the TechSAge RERC, and the Residential Care Facilities for the Elderly Authority of Fulton County. We would also like to thank Henry and Jane Evans and Newton Chan for their assistance with this work.

References

[1] N Browning. Mouse alternatives: Software and hardware options. *Occupational Therapy Now*, 7(4), 2005.

[2] JYC Chen, EC Haas, and MJ Barnes. Human performance issues and user interface design for teleoperated robots. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 37(6):1231–1245, 2007.

[3] TL Chen, M Ciocarlie, S Cousins, PM Grice, K Hawkins, K Hsiao, CC Kemp, C-H King, DA Lazewatsky, A Leeper, H Nguyen, A Paepcke, C Pantofaru, WD. Smart, and L Takayama. Robots for humanity: Using assistive robotics to empower people with disabilities. *IEEE Robotics & Automation Magazine*, March 2013.

[4] W Chou and T Wang. The design of multimodal human-machine interface for teleoperation. In *Systems, Man, and Cybernetics, 2001 IEEE International Conference on*, volume 5, pages 3187–3192, 2001.

[5] M Ciocarlie, K Hsiao, A Leeper, and D Gossow. Mobile manipulation through an assistive home robot. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012.

[6] H Evans, K Krishnaswamy, and E Rapacki. Telework: The accommodation that opens new doors to people with disabilities. Technical report, August 2015.

[7] D Gossow, A Leeper, D Hershberger, and M Ciocarlie. Interactive markers: 3-D user interfaces for ros applications [ros topics]. *Robotics & Automation Magazine, IEEE*, 18(4):14–15, 2011.

[8] PM Grice. *GT-ROS-PKG: hrl_assistive*. Georgia Tech, 2016. URL www.github.com/gt-ros-pkg/hrl-assistive.

[9] PM Grice, MD Killpack, A Jain, S Vaish, J Hawke, and CC Kemp. Whole-arm tactile sensing for beneficial and acceptable contact during robotic assistance. In *Rehabilitation Robotics (ICORR), 2013 IEEE International Conference on*, pages 1–8. IEEE, 2013.

[10] SG Hart. NASA-task load index (NASA-TLX); 20 years later. In *Proceedings of the human factors and ergonomics society annual meeting*, volume 50, pages 904–908. Sage Publications, 2006.

[11] KP Hawkins, PM Grice, TL Chen, C-H King, and CC Kemp. Assistive mobile manipulation for self-care tasks around the head. In *2014 IEEE Symposium on Computational Intelligence in Robotic Rehabilitation and Assistive Technologies (CIR2AT)*, pages 16–25, 2014.

[12] R Hazlett-Knudsen, MA Smith, and A Behal. Knowledge based design of user interface for operating an assistive robot. In *Human Centered Design*, pages 304–312. Springer, 2011.

[13] L Herlant, R Holladay, and S Srinivasa. Assistive teleoperation of robot arms via automatic time-optimal mode switching. In *Human-Robot Interaction*, March 2016.

[14] J Hoffmann and B Nebel. The FF planning system: Fast plan generation through heuristic search. *Journal of Artificial Intelligence Research*, 14, 2001.

[15] EL Hutchins, JD Hollan, and DA Norman. Direct

manipulation interfaces. *Human-Computer Interaction*, 1(4):311–338, 1985.

[16] S Iwarsson and A Ståhl. Accessibility, usability and universal design — positioning and definition of concepts describing person-environment relationships. *Disability and rehabilitation*, 25(2):57–66, 2003.

[17] A Jain, MD Killpack, A Edsinger, and CC Kemp. Reaching in clutter with whole-arm tactile sensing. *International Journal of Robotics Research*, 32(4), April 2013.

[18] S Jain, A Farshchiansadegh, A Broad, F Abdollahi, F Mussa-Ivaldi, and B Argall. Assistive robotic manipulation through shared autonomy and a body-machine interface. In *Rehabilitation Robotics (ICORR), 2015 IEEE International Conference on*, pages 526–531, 2015.

[19] S Keates, J Clarkson, and P Robinson. Designing a usable interface for an interactive robot. In *ICORR 99: 6th International Conference on Rehabilitation Robotics*, pages 156–162, 1999.

[20] B Keyes, M Micire, JL Drury, and HA Yanco. Improving human-robot interaction through interface evolution. 2010.

[21] D-J Kim, R Lovelett, and A Behal. Eye-in-hand stereo visual servoing of an assistive robot arm in unstructured environments. In *Robotics and Automation, ICRA'09. IEEE International Conference on*, 2009.

[22] A Kintsch and R DePaula. A framework for the adoption of assistive technology. *SWAAAC 2002: Supporting learning through assistive technology*, 2002.

[23] CE Lang, DF Edwards, RL Birkenmeier, and AW Dromerick. Estimating minimal clinically important differences of upper-extremity measures early after stroke. *Archives of physical medicine and rehabilitation*, 89(9): 1693–1700, 2008.

[24] DA Lazewatsky and WD Smart. Context-sensitive in-the-world interfaces for mobile manipulation robots. In *RO-MAN, 2012 IEEE*. IEEE, 2012.

[25] AE Leeper, K Hsiao, M Ciocarlie, L Takayama, and D Gossow. Strategies for human-in-the-loop robotic grasping. In *Proceedings of the seventh annual ACM/IEEE international conference on Human-Robot Interaction*, pages 1–8. ACM, 2012.

[26] RC Lyle. A performance test for assessment of upper limb function in physical rehabilitation treatment and research. *International Journal of Rehabilitation Research*, 4(4):483–492, 1981.

[27] Madentec Limited. *Tracker Pro User Guide*.

[28] J Mankoff, A Dey, U Batra, and M Moore. Web accessibility for low bandwidth input. In *Proceedings of the fifth international ACM conference on Assistive technologies*, pages 17–24. ACM, 2002.

[29] D. McDermott and the AIPS-98 Planning Competition Committee. Pddl - the planning domain definition language. Technical report, Yale University, 1997.

[30] J Nielsen. Usability inspection methods. In *Conference companion on Human factors in computing systems*, pages 413–414. ACM, 1994.

[31] DA Norman. *The Design of Everyday Things, Revised and Expanded Edition.* Basic Books, New York, NY, 2013.

[32] JM Noyes and DPJ Bruneau. A self-analysis of the nasa-tlx workload measure. *Ergonomics*, 50(4), 2007.

[33] *ARAT Test Kit*. Rehab Solutions, LLC, 2014. URL http://www.aratkits.com.

[34] J. M. Romano, K. Hsiao, G. Niemeyer, S. Chitta, and K. J. Kuchenbecker. Human-inspired robotic grasp control with tactile sensing. *IEEE Transactions on Robotics*, 27(6):1067–1079, Dec 2011.

[35] B Sankaran, B Pitzer, and S Osentoski. Failure recovery with shared autonomy. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 349–355. IEEE, 2012.

[36] M Schwarz, J Stückler, and S Behnke. Mobile teleoperation interfaces with adjustable autonomy for personal service robots. In *Proceedings of the 2014 ACM/IEEE International Conference on Human-robot Interaction*, HRI '14, New York, NY, USA, 2014.

[37] A Steinfeld. Interface lessons for fully and semi-autonomous mobile robots. In *Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on*, volume 3, pages 2752–2757. IEEE, 2004.

[38] *PCEye Mini - Computer access through gaze interaction*. Tobii Dynavox, 2015. URL www.tobiidynavox.com/pceye-mini.

[39] R Toris, J Kammerl, D Lu, J Lee, OC Jenkins, S Osentoski, M Wills, and S Chernova. Robot Web Tools: Efficient messaging for cloud robotics. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015.

[40] K Tsui, H Yanco, D Kontak, and L Beliveau. Development and evaluation of a flexible interface for a wheelchair mounted robotic arm. In *Proceedings of the 3rd ACM/IEEE international conference on Human robot interaction*, pages 105–112. ACM, 2008.

[41] KM Tsui, JM Dalphond, DJ Brooks, MS Medvedev, E McCann, J Allspaw, D Kontak, and HA Yanco. Accessible human-robot interaction for telepresence robots: A case study. *Paladyn: Journal of Behavioral Robotics*, 6.

[42] KM Tsui, D-J Kim, A Behal, D Kontak, and HA Yanco. I want that: Human-in-the-loop control of a wheelchair-mounted robotic arm. *Applied Bionics and Biomechanics*, 8(1):127–147, 2011.

[43] HFM Van der Loos and LJ Leifer. The design and use of history list systems for rehabilitation robots: Enhancing safety and performance through activity recording and analysis. *Technology and Disability*, 5(2):177–196, 1996.

[44] HA Yanco, A Norton, W Ober, D Shane, A Skinner, and J Vice. Analysis of human-robot interaction at the darpa robotics challenge trials. *Journal of Field Robotics*, 32 (3):420–444, 2015.

[45] N Yozbatiran, L Der-Yeghiaian, and SC Cramer. A standardized approach to performing the action research arm test. *Neurorehabilitation and Neural Repair*, 22(1): 78–90, 2008.