Space Construction: an Experimental Testbed to Develop Enabling Technologies

H.C. Schubert and J.P. How

Aerospace Robotics Laboratory
Stanford University, Stanford CA 94305

ABSTRACT

This paper will discuss a new testbed developed at the Stanford Aerospace Robotics Laboratory (ARL) to address some of the key issues associated with semi-autonomous construction in a hazardous environment like space. The new testbed consists of a large two-link manipulator carrying two smaller two-link arms. This macro/mini combination was developed to be representative of actual space manipulators, such as the SSRMS/SPDM planned for the Space Station. This new testbed will allow us to investigate several key issues associated with space construction, including teleoperation versus supervised autonomy, dextrous control of a robot with flexibility, and construction with multiple robots. A supervised autonomy approach has several advantages over the traditional teleoperation mode, including operation with time delay, smart control of a redundant manipulator, and improved contact control.

To mimic the dynamics found in space manipulators, the main arm was designed to include joint flexibility. The arm operates in 2-D, with the end-point floating on an air-bearing. This setup allows cooperation with existing free-flying robots in the ARL. This paper reports the first experiments with the arm which explore the advantages of moving from teleoperation or human-in-the-loop control to the human supervisory or task-level control. A simple task, such as capturing a satellite-like object floating on the table, is attempted first with the human directly driving the end-point and second with the human directing the robot at a task-level. Initial experimental results of these two control approaches are presented and compared.

Keywords: joint flexibility, teleoperation, task-level control, macro/mini manipulator, redundant manipulator

1. INTRODUCTION

In the hazardous environment of space, the use of robotics for on-orbit construction, servicing, assembly, and repair should be more cost effective and safer than using human EVA. In particular, the space construction task is difficult because it requires that tasks such as insertion and precise alignment of parts be routinely performed. These complex tasks, combined with the viewing constraints of space, the dynamics of the manipulator itself, and the demanding construction schedule make space construction with robots extremely difficult.

The Space Station Remote Manipulator System (SSRMS), a large two-link arm like the current RMS, is being developed. The SSRMS can carry the Special Purpose Dextrous Manipulator (SPDM), two smaller 7 degree of freedom arms for finer manipulation.[1] This macro/mini system will allow a large variety of tasks to be performed roboticly, from moving a large truss structure to closing a hatch. However, to take full advantage of more capable hardware, control software and interfaces are required.

Currently, space robots are controlled directly by a human via teleoperation. This takes advantage of the human's excellent perception, judgment, and learning capabilities. However, for a large, flexible robot, it can take months of training for a human to learn to perform one task reliably. On-orbit astronaut time could be spared if the robot were controlled from Earth, but time delay makes teleoperation more difficult.[2] The SSRMS/SPDM is a three arm system, but the hand controller can only control one arm at a time,[3] limiting the tasks that can be performed. Teleoperation of such a complex manipulator takes a large amount of attention and can be very fatiguing to the operator. An alternative to providing constant joystick inputs is to have the human issue high-level task commands to be automatically controlled by the robot. This supervised autonomy approach allows the human to remain in

Other author information:
H.C.S.: Email: heidi@sun-valley.stanford.edu
J.P.H.: Email: howjo@sun-valley.stanford.edu

control while taking advantage of advanced computer control. It has been conjectured that using a task-level control structure would result in faster completion time, more accurate control, and be less tiring to the human operator.[4] This research experimentally explores the difference between task-level control and teleoperation.

While numerous other testbeds exist on the ground, few combine the macro/mini manipulator combination which is essential for correctly assessing the autonomy/teleoperation trade-off for station construction tasks. Building on the extensive robotics research in the Aerospace Robotics Lab (ARL), we have built a new macro/mini testbed. The new testbed consists of a two degree of freedom main arm that carries two smaller arms. The testbed will be used for comparing teleoperation versus supervised autonomy, dextrous control of a robot with flexibility, and construction with multiple robots. This paper presents preliminary results of teleoperation and task-level control comparisons. A simple task, representative of construction tasks, is performed first with teleoperation and second with task-level control.

The paper begins with a discussion of the task-level control approach. This is followed by a description of the manipulator system used for the research. Then, the experiment is outlined, followed by the results.

2. TASK-LEVEL CONTROL AND TELEOPERATION

Space robots, such as the shuttle RMS, are currently being controlled by teleoperation, wherein a human must explicitly command every movement of the robot. In contrast, a fully autonomous robot could be given a mission, such as repairing a broken radiator, and figure out how to safely do that. Full autonomy requires artificial intelligence far beyond current technology. However, current technology is very good at simpler tasks, such as closing a control loop around a hold point, controlling force, or calculating an efficient trajectory. The idea of task-level control is that a human does the intelligent sequencing and planning of large operations and the robot carries out simpler, commanded tasks. For example, in repairing a space station, the human would issue tasks to the robot, such as "remove that box" and "move box to storage port". These tasks are then broken up into a sequence of commands that are performed under supervision of the operator.

The concept of having the computer close lower level control loops and the human command at a higher level has been explored by several researchers. Sheridan[5] defined supervisory control as any control scheme where any control loops are closed by a computer, not by a human. One developed architecture is the NASA/NBS Standard Reference Model Telerobot Control System Architecture (NASREM).[6,7] NASREM is a six-level hierarchy, with sensory processing, world modeling, and task decomposition at each level. The lowest level of NASREM is servo control for all actuators, with the highest level a mission planning level, the user can input at any level. Another architecture is the 3T architecture developed at JSC.[8] The 3T architecture is based on 3 tiers, the lowest tier being a set of reactive skills that handle low-level control of the robot. The next tier is a sequencer that activates sets of skills in order. The top tier is a high level planning tier. The 3T architecture has been applied to several robotic experiments, including several mobile robots.

The ARL has also developed its own control architecture called Object-Based Task-Level Control (OBTLC).[4] This framework is different from other architectures because the user specifies desired behavior of an object manipulated by the robot or robots, rather than specifying desired behavior of the robot itself. The human/robot team completes the missions by the human issuing consecutive tasks that the robots are capable of carrying out. Thus the human handles high-level planning and problem solving and the robot executes simple planning algorithms and closes low-level control loops. This architecture has been used in the lab for many experiments, including very flexible manipulators, free-flying space robots, and underwater vehicles. The OBTLC architecture consists of three layers, the user interface, the strategic controller, and the dynamic controller as shown in Figure 1.

The highest level of the OBTLC architecture is the user-interface. The user interface displays real-time position and force information for the robot and objects in the environment as well as status information such as fuel usage. This can be displayed with any combination of graphics, live video, and figures. The user requests actions by selecting the appropriate buttons and icons with a mouse. For example, the user can select to move an object to a new location by clicking the appropriate button and dragging the object's icon to the new location. For a complex task, the robot can plan out the required movements for the task and display the planned action to the user for approval before executing. After selecting to execute the task, the user can supervise the task and stop execution if necessary at any time.

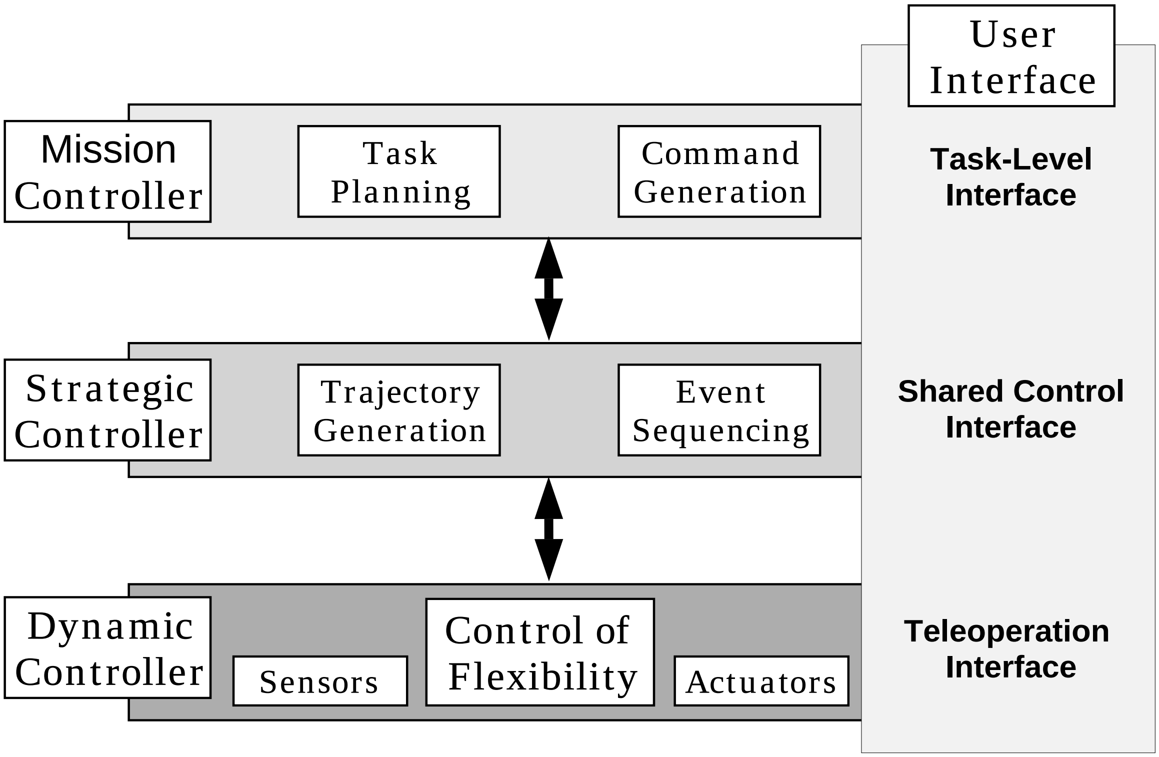

Figure 1. The task-level control architecture, nominally the user inputs only into the top tier, but can input at a lower level.

The inputs from the user interface communicate directly with the middle layer, the strategic controller. The strategic controller has a finite state machine that executes different low-level operations based on inputs either from the user or from the robot's sensors. The strategic controller does the required planning for the robot, for example, collision free trajectory planning, coordination between two robots, and dual-arm manipulation planning. The strategic controller also provides a software safety layer by stopping execution and/or notifying the operator if safety is compromised, like the robot reaching the edge of its workspace or exceeding a velocity limit.

The dynamic controller handles the high-bandwidth feedback control loops for low-level operation of the robot as commanded by the strategic controller. Using the available sensors, the dynamic controller commands the actuators to achieve precise, quick behavior of the robot. Smart control laws take into account structural flexibility of the manipulator or non-linear actuators.

Although it is possible to do so, the user does not usually communicate directly with the dynamic controller. Therefore, the user does not worry about the details of how the robot carries out low-level control, such as position and force regulation. The user's attention is focused more on oversight of safety and performance rather than low-level operation. Thus the operator must still be present to supervise, but can time-share with other tasks. Although task-level control sounds like an ideal control architecture, some issues remain unresolved. For example, the controller may not be robust to changes in the manipulator dynamics, sensor failures, or changes in the environment. Also, it is unknown if task-level control can achieve the same or better performance than teleoperation for a variety of tasks.

In contrast to task-level control, teleoperation requires that the human be an integral part of the control loop. Every motion of the robot must be explicitly indicted by the human through an input device. This takes advantages of a human's superb sensory perception and ability to learn. Also, leaving the human in the loop is more robust to sensor dropout and computer failures. The SSRMS/SPDM is planned to be operated with several teleoperation modes, either operating each joint individually, or commanding end-point rates of a single arm.[1] However, some more automated modes are planned for when the arm is unloaded, specifically an automated preplanned trajectory following and an object tracking mode.

Proponents of task-level control architectures have stated that it will have many benefits, such as increased speed and repeatability. However, few studies have been done on actually comparing the performance of a teleoperated robot versus one controlled by a computer, in particular for the complex tasks required for space construction. The macro/mini manipulator system developed in the ARL provides an excellent testbed for such a comparison.

3. HARDWARE DESCRIPTION

3.1. Manipulator System

The new manipulator system has been designed to model the dynamics, capability, and operation of a space-based robotic system. The shuttle RMS has significant joint flexibility, with the first brakes-on natural frequency of about 0.5 Hz.[9] The manipulator planned for the space station has a main arm that carries two smaller arms. In order to model the space based manipulator system, the experimental manipulator was designed to be a macro/mini manipulator with joint flexibility.

The manipulator system consists of one large main arm carrying two smaller mini arms. Because of the difficulty of counteracting gravity with suspension systems, the robot is designed to operate only in the 2-D plane perpendicular to gravity. The end-point of the main arm floats on an air-bearing on a 6 ft by 12 ft granite table. The flat, smooth, stable granite surface provides a very low drag environment.

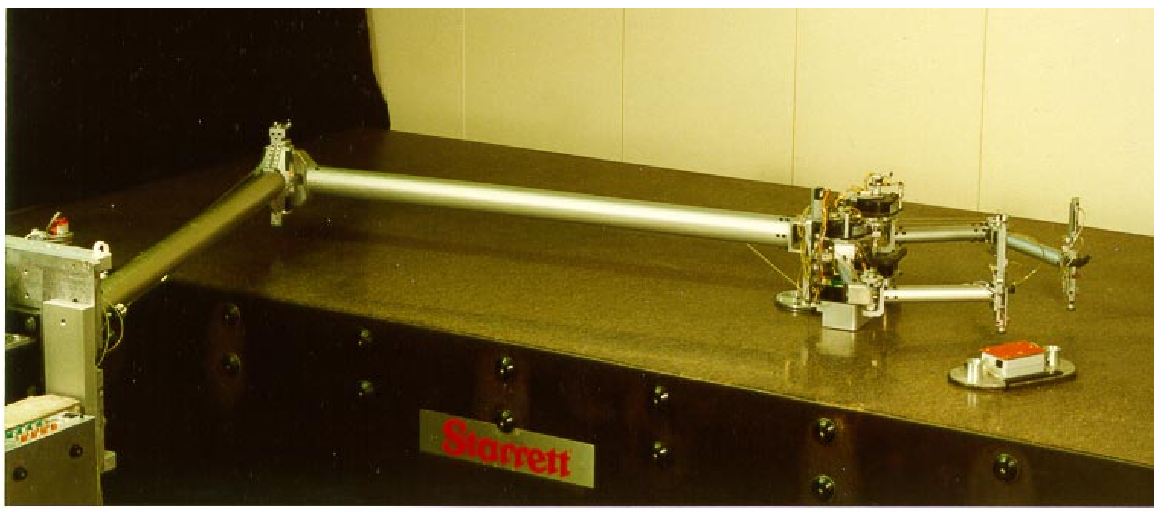

Figure 2. The two-link manipulator with two mini arms mounted at its end-point.

Each of the two links of the main arm is 5 ft long, giving the manipulator a workspace larger than the table the end-point is floating on. The shoulder of the manipulator is attached to a large concrete block, housing both the shoulder and elbow motor. The elbow motor drives the elbow joint via an idler pulley pivoted at the shoulder joint, reducing the effective inertia of the arm by locating the motor off-board. Both motors have a peak torque of 11 Nm, with a 2.44:1 gear reduction on the elbow drive train and a 5.91:1 gear reduction on the shoulder drive train. Both of the cable drives have springs mounted in-line to give the joints exaggerated flexibility, mimicking the joint dynamics of space manipulators. The brakes-on natural frequency of the elbow joint is 0.5 Hz and the shoulder joint is 0.7 Hz, similar to the RMS. Encoders on both sides of the flexibility measure both the motor position and the link position. The links of the arm are rigid tubes, but could later be changed to flexible links to further simulate the dynamics of space manipulators.

The main arm carries two smaller mini arms, identical to the mini arms carried by the free-flying vehicles in the ARL.[10] Each link of the mini arms 1 ft long, giving the mini a smaller workspace, but higher bandwidth than the main arm. The elbow motor is located on the shoulder, driving the elbow through a 1:1 cable drive, thus reducing the effective inertia of the arm. The shoulder motor has a peak torque of 1.0 Nm and the elbow motor a peak torque of 0.4 Nm. Joint angle information is provided for each link by an RVDT. Each arm has a pneumatic gripper at its end. To capture an object, the gripper plunges into an object port. A ball bearing in the gripper allows it to control translational degrees of freedom while imparting no moment on the object.

Global vision information is given by an overhead camera that tracks LEDs. A pattern of three LEDs on the end-point of the main arm uniquely identify it as well as giving position and orientation of the end-point. Each mini arm has a single LED for position tracking.

The manipulator can pick up one of several free-floating objects on the table. For this experiment, a 7.0 kg, 12 in diameter object is used (Figure 3). This object has a its own flotation system and two gripper ports. The height of the object is small enough that if the grippers are in the up position the arms pass over the object, but if the grippers are down they collide with the object. On top of the object is a pattern of 3 LEDs for tracking by the vision system. This object can be thought of as a satellite or part that the manipulator must catch or move around.

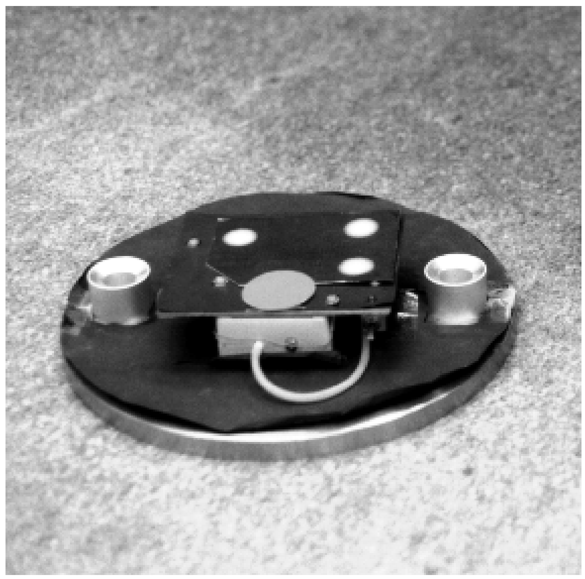

Figure 3. The free-floating object used in the experiments, note the two gripper ports.

3.2. User Interface

3.2.1. Teleoperation Mode

The input device for each of the arms on the SSRMS/SPDM consists of two hand controllers, one providing pitch-roll-yaw information and one providing x-y-z information. Each arm on this experimental manipulator has only two degrees of freedom, x-y, so only a two degree of freedom input device is needed. A Wingman Warrior joystick is used as the input device. The pitch/roll of the joystick provide desired rates in x and y directions in the table coordinate frame. Three thumb buttons on the joystick select between the main arm, mini right arm, and mini left arm. A finger trigger fires the plunger up/down when either of the mini arms is selected.

In the first few trials, the operator observes the robot simply by standing by the table and directly viewing the robot. To simulate cases where direct observation is not available, the user also can operate by observing a live video stream. The video camera is placed to give the operator a good, oblique angle, view of the end-point of the arm but not the entire arm. If necessary, the camera can be manually moved. Viewing only the television screen is designed to simulate the case where viewing is limited, either by only viewing through a window like an astronaut on-orbit or through just a camera view like a remote operator.

3.2.2. Task-Level Control Mode

In contrast to the teleoperation mode, the task-level control mode does not require a joystick for an input device. Instead, the keyboard and mouse on the Sun workstation are used to input commands to the robot. The workstation also displays any desired information on the manipulator, such as position and joint torques. As with teleoperation, the user has access either to direct visual with the robot, or a televised view.

4. CONTROL DESIGN

4.1. Low-Level Control

The low-level PD control loop for both the main arm and the mini arms runs at 50 Hz. The main arm control law is simply a PD loop using joint angle sensors. This controller is quite primitive and does not give very good end-point performance because it ignores joint flexibility and does not use the end-point measurement. In the future, better performance will be achieved through the use of an end-point control law, similar to the ones previously implemented in the ARL.[11] This model based control law uses end-point sensing for better performance. The advantage to using a simple PD control law for this experiment is that the task-level controller and teleoperation both use the same low-level control law.

The mini arms can be considered to have rigid joints. The control law for each of the arms is a PD control using end-point vision sensing. This results in desired end-point forces which can be mapped into joint torques through the inverse Jacobian.

The current control design has limited performance because the macro and mini controllers were designed separately. Movements of the mini arm are seen as disturbances to the main arm. This results in uncoordinated motion, where the macro moves significantly for any mini motion. This can be improved through either a combined control law or using feedforward to the main arm to compensate for mini movements. Building on previous work in the ARL,[12] the next controller implemented will use feed-forward for improved performance.

4.2. Task-Level Control

The task-level control design consists mainly of building the strategic controller, a finite state machine on top of the low-level control law. The finite state machine for the task of picking up the object and moving it to its final point is described. When the task is issued, the macro arm moves toward the object, and the mini arms wait until the object is within its workspace. When the object is within reach, the mini arms switch to tracking the object while the main arm holds position. When the mini arms are aligned with the gripper ports, the plungers are fired and the object is captured. This event triggers the main arm to move towards the final position with the mini holding the object. When the main arm final position is reached, the mini arms position the object exactly, releasing as it is positioned. Then the manipulator waits for the next command from the user. The user can intervene if necessary, for example if the mini arms miss capturing the object.

4.3. Teleoperation

The teleoperation mode is designed to be similar to the end-point control mode designed for the SSRMS/SPDM.[1] Here the user commands end-point velocities in the global frame with the joystick. The other teleoperation mode planned for the SSRMS/SPDM is a joint rate mode, where each joint is commanded to move at a time. This mode was not examined as it is difficult for an untrained operator.

The joystick allows the operator to command the end-point velocity of one of the three arms. The same PD control law is used, with the position gains set to zero. However, when the mini arms are operated, the macro arm is commanded a position hold and when the macro arm is operated, the mini arms are commanded a position hold mode. When the mini arms are holding an object, the object constrains the motion of the arms. For this reason, when one mini arm is operated, the other mini arm is commanded only a zero velocity, not a position hold. This avoids the two mini arms placing a large internal force on the object.

5. EXPERIMENT

As an example of the type of comparisons that can be performed on this testbed, a task was chosen that is representative of those needed on-orbit and that necessitates the use of each robot arm. The chosen task is to pick up the object and place it in a certain position and orientation (Figure 4). The object is placed on the table in a predefined position and orientation, with the manipulator in a starting position with the object out of the workspace of the mini (4.1). The macro manipulator must move its end-point so the mini can reach the object (4.2). The object is then captured by the mini, by plunging each gripper into the two gripper ports (4.3). Then the macro must move the mini to the drop-off point (4.4) where the two mini arms place the object in its correct position and orientation

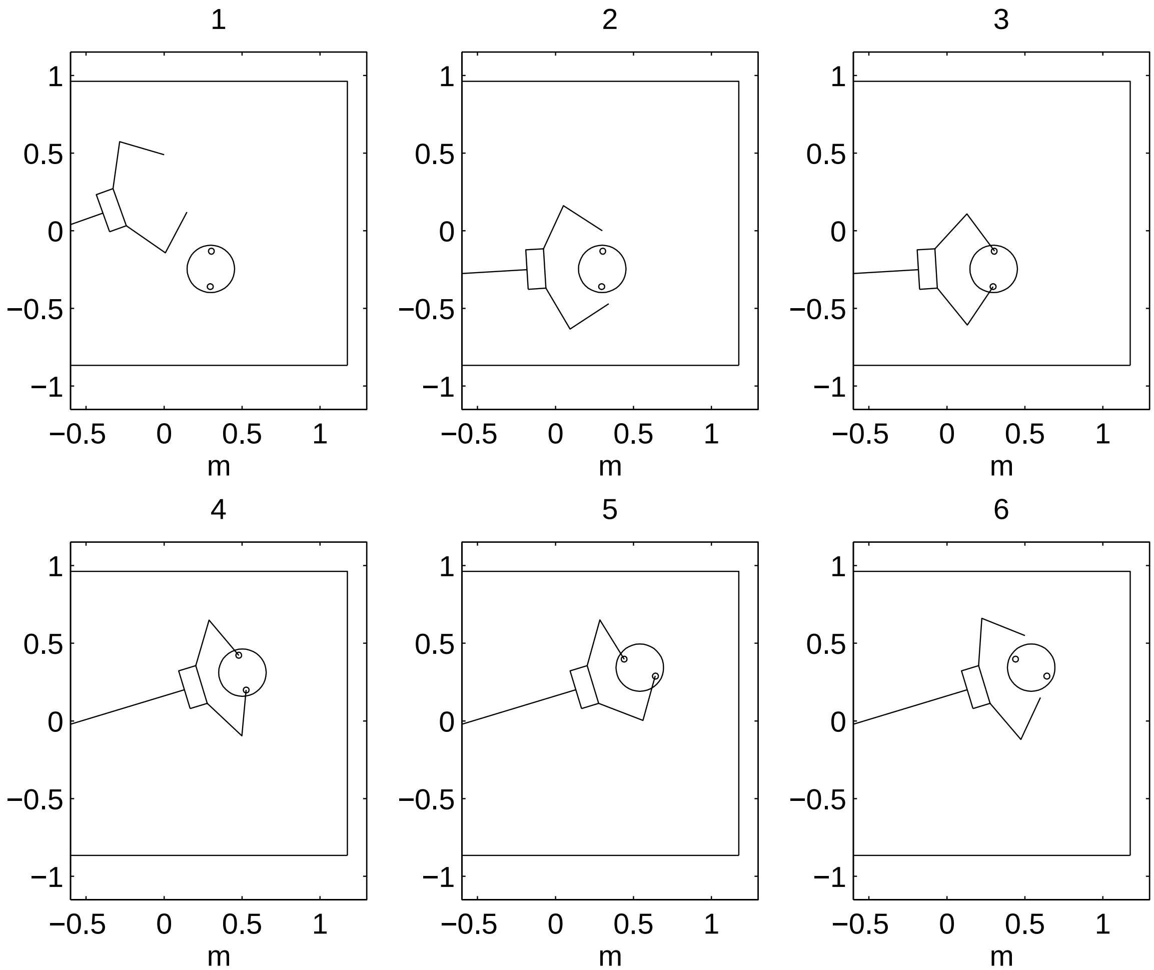

Figure 4. A sequential top view of the end of the arm performing the task of moving the object

(4.5). This tasks tests several different operational elements, such as precisely positioning the arm over the gripper port and coordinating the mini arms to place the object correctly.

The teleoperation mode was tested with nine test subjects, all engineering graduate students, both male and female. None of the test subjects had previously operated the arm. Each received brief instruction on the task to be performed and operation of the joystick. Each user attempted the task four times. The first two trials were done with the user standing at the edge of the table. The second two trials were done with the user around the corner looking only at a live video on a 26” television.

The task-level control requires only minimal input from the user, telling the robot to pick up the object and move it to the new location. Therefore, to not burden the test subjects, the task-level control trials were tested separately. For the task-level control trials, the computer performed the same task nine times.

Three metrics were chosen to compare the performance of the two different control architectures. The first metric, the time to complete the task, is a measure of speed and efficiency. The second metric, the number of times the gripper missed its target, is a measure of accuracy. The third metric, a weighted sum of control torques, is a measure of power used and input effort.

6. EXPERIMENTAL RESULTS

Nine volunteers performed four teleoperation trials each, with the first two trials done standing right by the table and the last two watching only a monitor. As the first trial was the first time anyone had operated the robot, it was expected that the second trial would show improved performance. It was also expected that performance would degrade when only having access to the television monitor.

Comments from the operators verified that controlling the robot while viewing only the television screen was more difficult. Depth perception and occlusion were cited as the difficulties. Depth perception was a problem when trying to position the gripper directly above the port, although most operators were able to judge it better after a few misfires of the gripper. Three different operators were unable to complete the task on one of their trials while watching the television only. The task was not completed because the object port was missed by the gripper, then the arm was moved such that the extended gripper pushed the object away too rapidly to be recaptured.

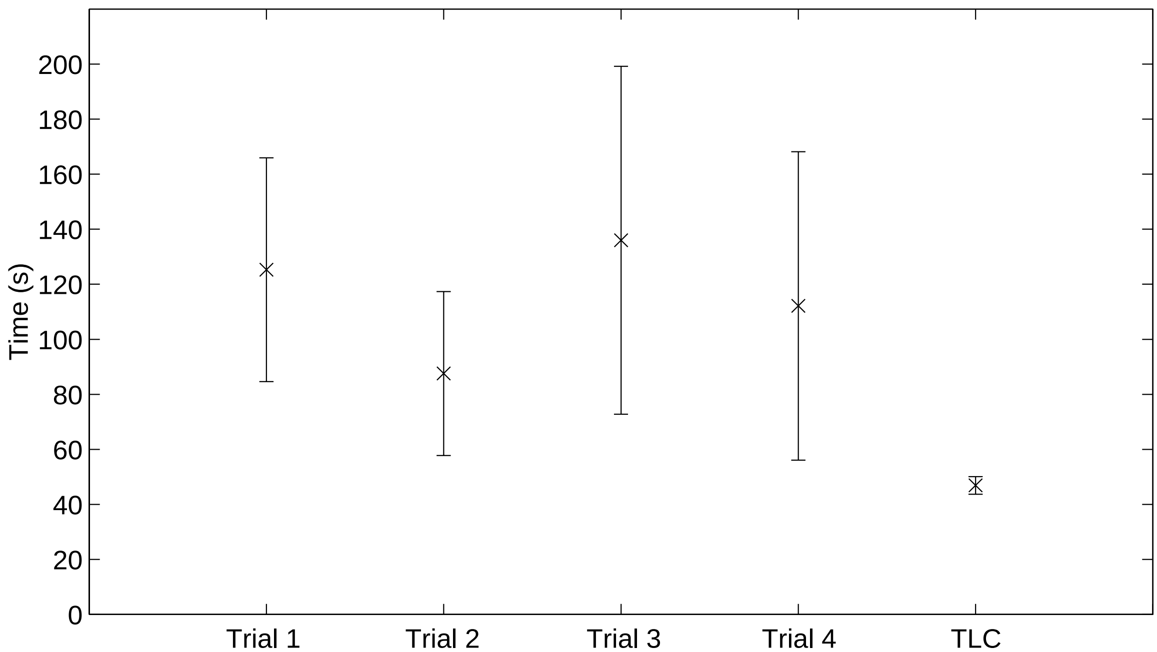

Figure 5. Shows the average time to complete task for each of the four teleoperation trials and the task-level control runs. The x is the mean value and the error bars show the standard deviation.

Results from the first measure of performance, time to complete the task, are shown in Figure 5. As expected, the mean time for the second trial is less than the first, demonstrating that the operator learned from the first to the second trial. In fact, all but two of the subjects improved their time on the second try. The mean time for the third trial increased as the user now has a restricted view, every subject's third time increased over the second. The standard deviation is quite large for all the trials, as some subjects had more difficulty than others. For the last two trials, the standard deviation increased, showing a larger spread among the operators. The mean of the task-level control runs was 47 seconds, while the fastest time for a teleoperator was 50 seconds. In addition, the standard deviation of the task-level control runs was quite low, showing good consistency between each run.

Another measure of success is how many times the gripper missed the gripper port when fired. The gripper itself is 2.2 cm in diameter, while the target port is 3.9 cm. The accuracy of placing the gripper directly over the port seemed mostly a function of visual cues. For the first two trials, the gripper missed 25% of the time, while for the last two the gripper missed 58% of the time. Several subjects stated that depth perception and occlusion made it harder to accurately position the grippers when looking only at the television. Under task-level control mode, the computer control missed 0% of the time. The computer fired both grippers simultaneously and did not fire until the vision system verified that both were positioned accurately enough.

The third metric is a measure of control effort. This measures the power used to perform the task, as well as showing how much user effort was required. The control cost for each motor is found by summing the squares of the input torque. For example, the cost for the mini left arm elbow motor, J_{LE} is found from its torque, u_{LE} using the following equation.

$$J_{LE} = \int_0^{T_{final}} u_{LE}^2 dt$$

The overall cost is found by a weighted sum of each control cost. The subscript LS stands for the left mini arm shoulder, RE and RS for the right mini arm elbow and shoulder, and ME and MS for the main arm elbow and shoulder motor.

$$J = (J_{LE} + J_{LS} + J_{RE} + J_{RE})/.4^2 + J_{ME} + J_{MS}$$

The peak torque for each of the mini motors was limited to 0.4 Nm and the peak torque for each of the main arm motors was limited to 1.0 Nm. Each term in the control cost is normalized by its peak torque.

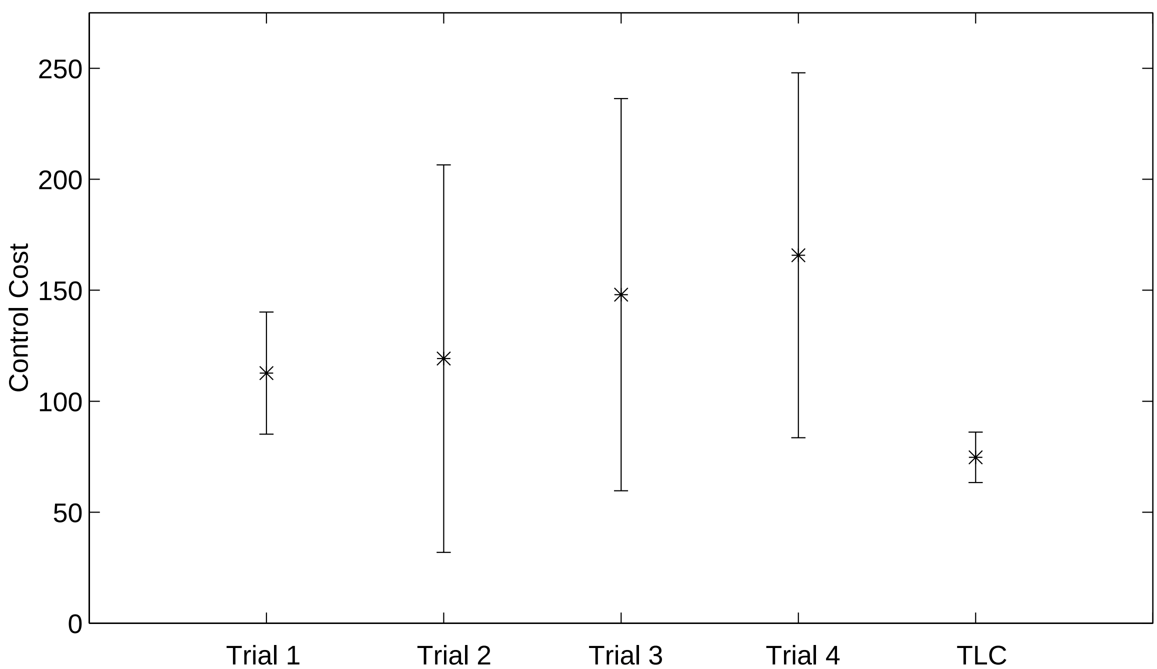

Figure 6. Shows J, the control cost for each of the four teleoperation trials and the task-level control runs. The * is the mean value and the error bars show the standard deviation.

The control cost for each teleoperation trial and the task-level control run are shown in Figure 6. Unlike the completion times, the control effort does not show a decrease from trial 1 to trial 2. However, the last two runs do show a slightly increased control effort. The task-level control run shows a lower mean and standard deviation for control effort.

7. CONCLUSIONS

The purpose of this paper was to experimentally evaluate the difference between a task-level control strategy and teleoperation. The task of retrieving, moving, and placing a free-floating object using a macro/mini manipulator was carried out using both teleoperation and task-level control. The task-level control performed better on average than the teleoperators in all three performance measures: completion time, accuracy of gripper placement, and control effort. In particular, the task-level controller had much better accuracy for gripper placement, never missing a gripper port in all nine runs.

These initial results look promising for the ability of the task-level controller. Future work will focus on improving the capabilities of the task-level controller. The initial PD control design has a quite limited performance, a more capable end-point controller for the main arm needs to be implemented. In addition, coordinated control is needed for the main and mini arms, especially when the mini captures a large object. Future comparisons with teleoperators will include more complex tasks, such as assembly of objects. These experiments used only untrained teleoperators, it is unknown how much better a well-trained teleoperator would perform. Future tests could include more sophisticated control techniques and trained teleoperators. Additional performance metrics, such as disturbance rejection, operator fatigue, and force regulation could also be compared.

REFERENCES

1. C. Trudel, D. Hunter, and M. Stieber, "Control and operation of space manipulator systems," in *Advanced Guidance and Control Aspects in Robotics*, pp. 5.1–5.19, AGARD-LS-193, 1994.
2. M. Ottensmeyer, J. Thompson, and T. Sheridan, "Telerobotic surgery: experiments and demonstration of tele-surgeon/assistant cooperation under different time delays and tool assignments," in *Telemanipulator and Telepresence Technologies III*, *Proc. SPIE* **2901**, pp. 156–166, 1996.
3. D. Cooke and D. Hunter, "Operations procedure planning tools for space station robotics task analysis," *Space Technology* **12**(1), pp. 35–44, 1992.
4. H. Stevens, E. Miles, S. Rock, and R. Cannon, "Object-based task-level control: a hierarchical control architecture for remote operation of space robots," *Conference on Intelligent Robotics in Field, Factory, Service and Space* **1**, pp. 264–73, 1994.
5. T. Sheridan, *Telerobotics, Automation, and Human Supervisory Control*, The MIT Press, Cambridge, MA, 1992.
6. R. Lumia, "Using NASREM for real-time sensory interactive robot control," *Robotica* **12**, pp. 127–135, 1994.
7. R. Lumia, "Using NASREM for telerobot control system development," *Robotica* **12**, pp. 505–512, 1994.
8. R. Bonasso, R. Firby, E. Gat, D. Kortenkamp, D. Miller, and M. Slack, "Experiences with an architecture for intelligent, reactive agents," *Journal of Experimental and Theoretical Artificial Intelligence* **9**, pp. 237–256, 1997.
9. T. Alberts, H. Xia, and Y. Chen, "Dynamic analysis to evaluate viscoelastic passive damping augmentation for the space shuttle remote manipulator system," *Journal of Dynamic Systems, Measurement, and Control* , pp. 468–474, 1992.
10. M. A. Ullman, *Experiments in Autonomous Navigation and Control of Multi-Manipulator Free-Flying Space Robots*, Stanford, CA 94305, March 1993. Also published as SUDAAR 630.
11. M. Hollars and R. Cannon, "Experiments on the end-point control of a two-link robot with elastic drives," in *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, pp. 19–27, 1986.
12. H. Stevens and J. How, "The limitations of independent control design for multiple-link flexible macro-manipulator carrying a rigid mini-manipulator," in *Proceedings of the 2nd ASCE Conference on Robotics for Challenging Environments*, pp. 93–99, 1996.