

# A Novel Teleoperated Long-Reach Manipulator Testbed and its Remote Capabilities via the Internet

Wayne J. Book, Hobson Lane, Lonnie J. Love, David P. Magee, Klaus Obergfell

The George W. Woodruff School of Mechanical Engineering  
Georgia Institute of Technology  
Atlanta, Georgia 30332-0405  
www: <http://davinci.marc.gatech.edu/>

## Abstract

*During the 1995 DOE Robotics Forum at Albuquerque, New Mexico, Georgia Tech's Intelligent Machine Dynamics Laboratory demonstrated remote display capabilities of its telemanipulated long-reach manipulator testbed through the Internet. This paper describes the testbed and addresses the methods, limitations, and capabilities of remote operations.*

## 1. Introduction

The dynamics and control of flexible manipulators has been an active research area for the past 20 years [1]. A long-reach flexible manipulator testbed has been developed at Georgia Tech's Intelligent Machine Dynamics Laboratory (IMDL) during the last decade for the verification of theoretical developments in a realistic setting. Previous and ongoing work with the testbed include: modeling [4], joint control [12], end-point position sensing and control [9], macro/micro manipulation [5], and command filtering [8]. During the last three years the issues of human/machine interfaces and remote capabilities for long-reach manipulators have also been investigated [6],[7]. Little work had previously been published on human interfaces for compliant manipulators.

Nuclear waste restoration efforts by the Department of Energy (DOE) highlight the need to investigate such remote capabilities. The use of long-reach manipulators for tank waste retrieval from large underground storage tanks is currently being investigated by national laboratories. Work space requirements for this application are enormous, for example, underground storage tanks at DOE's Hanford site are 23 m in diameter and 10-16 m high. Applications in such hazardous environments also require the isolation of the human operator. In the specifications for the nuclear waste restoration project, the operator may be located miles

from the contamination site. Our testbed attempts to simulate this real world scenario and provide further insight into remote manipulation using long reach manipulators.

Some of the issues associated with remote teleoperation of compliant manipulators were addressed during a demonstration at the 1995 DOE Robotics Forum in Albuquerque, New Mexico. The IMDL long-reach manipulator testbed at Georgia Tech was telemanipulated from our isolated control room, while data was sent through the Internet to Sandia National Laboratories in New Mexico to animate a telegrip model of the testbed in real-time. The use of product names in this paper implies no endorsement, but will provide the reader with information on availability and compatibility or lack thereof for this undertaking.

## 2. Testbed Overview

The manipulator testbed of the Intelligent Machine Dynamics Laboratory at Georgia Tech consists of several modules that are used individually or together to study advanced manipulator controls and human interfaces: the long-reach manipulators RALF and BERTHA, the articulated manipulator SAMII, and the haptic interface HURBIRT.

RALF (Robotic Arm, Large and Flexible) is a hydraulically actuated, two degree-of-freedom manipulator operating in the vertical plane. The design may be indicative of manipulators used in the nuclear waste restoration process. It consists of two cylindrical links with a span of 3 m each and a parallel link mechanism for actuation of the second joint. Total weight of the link structure is 45 kg while the payload capacity of the manipulator is 27 kg. Several sensors are used to measure rigid and flexible states of the manipulator. Joint angles are calculated from measurements of the length of the hydraulic cylinders

using linear displacement transducers. Modal data is available from strain gages mounted to the links. Deflections of individual links can be measured using lateral-effect photodiodes. End-point position measurements are available from a landmark tracking system.

BERTHA (Big Elastic Robotic THree dimensional Arm) consists of a single flexible link which is positioned using a three degree-of-freedom base from a Cincinnati Milacron T3-646 industrial robot. The wrist assembly was removed from the robot and replaced with a 6 m cylindrical link to give a large three dimensional workspace.

SAMII (Small Articulated Manipulator II) is a hydraulically actuated, seven degree-of-freedom arm that consists of a rotating base, two rigid links, a three degree-of-freedom wrist, and a gripper. SAMII weighs approximately 25 kg and can be mounted at the end of RALF or BERTHA.

HURBIRT (Human Robot Bilateral Research Tool) is a two degree-of-freedom haptic interface used for force-reflecting telemanipulation and adaptive impedance control.

### 3. Sensing and Control Environment

In addition to joint sensors and strain gages, two unique sensor systems are available at the IMDL to measure the end-point position of long-reach manipulators: Landmark Tracking System (LTS) and lateral-effect photodiodes.

The LTS is an off-the-shelf vision system adapted for the tracking of retroreflective landmarks [9]. It consists of a Dickerson Vision Technologies, Inc. Stinger 70 integrated vision system, optics, strobe unit, and retroreflective landmarks. The Stinger 70 is a self-contained unit containing CPU and vision hardware communicating with a client through a RS-232 serial interface. Strobe and retroreflective landmarks are used to obtain high contrast images at a fast rate. For end-point position measurements of RALF, the LTS is mounted stationary with a line of sight perpendicular to the manipulator's plane of motion and tracks a retroreflective target attached to the tip of the manipulator. Using a 6 mm telephoto lens and a distance of 5.7 m between LTS and the plane of motion, a 2.7 m x 2.7 m area within the manipulator's workspace can be observed with a position resolution of 2 mm. Through calibration, the output of the LTS was related to the end-point position with respect to the manipulators base with an accuracy of approximately 5 mm. Using software that maintains a small window around the expected target location the sampling rate of the LTS is approximately 50

Hz. This enables the LTS to track the robot's end-point at speeds greater than 2 m/s.

A second, manipulator based, end-point position measurement system was developed, because the LTS observes only part of the workspace [10]. It utilizes lateral-effect photodiodes and joint angle sensors. A lateral-effect photodiode, a focusing lens and a signal conditioning amplifier are attached to the base of each of RALF's links and an infrared-LED is attached to the other end of each link. The output of the photodiode is proportional to the translation of the LED and the link deflection can be obtained through calibration. The position of the tip of the arm is then computed from joint angles and link deflections. However, any compliance not monitored by this system will cause errors in the tip position estimate, e.g. base deformation. To improve the manipulator positioning accuracy it is therefore proposed to combine this sensing system with a tip mounted LTS.

The programming environment for control software development consists of a UNIX host machine and a target CPU board within a VMEbus chassis. Our particular host is a Sun IPC Workstation with Solaris 2.4 as its operating system and our target CPU is a Motorola MVME167 board running a VxWorks 5.2 shell. The two computers are able to communicate with one another via a direct serial line connection or over the Ethernet.

Most host/target computer arrangements contain dissimilar CPU chips that require a specific cross compiler for software execution. For each particular system, a cross compiler must be obtained to provide the correct object code for execution on the target CPU. Unfortunately, VxWorks is only a C-shell and it only supports C source code. However, a pseudo-C++ compiler can be obtained through shareware sources. The C++ compiler will not support some important features such as `new()` and `delete()` memory allocation and data streaming. Name mangling of C++ function names will also occur.

Once the source code has been compiled to object code, it can be downloaded to the target CPU where it is linked with any other relevant object code and transformed into executable code. At this point, programs can be executed within the shell by typing the appropriate function name. The real-time multitasking operating system will permit the execution of many programs spawned as individual tasks. By prioritizing the tasks, a user can then control the order of task execution and the amount of CPU time each task receives. Tasks running during the remote demonstration where prioritized in the following order: joint control (PD or PD with command filtering are usually used for telemanipulation), communication between master and slave, socket communication, and communication with landmark tracking system.

In general, VMEbus systems contain many different types of boards for gathering, manipulating and outputting information. Our particular system has Analog-to-Digital converter boards, Digital I/O boards, Digital-to-Analog converter boards, a carrier board with a DSP chip and the target CPU board. To tailor these boards to our specific needs, low level drivers were developed and written for each board. Although development can often take several months, the resulting drivers are usually more efficient and reliable than ones obtained directly from the board manufacturer. For high speed applications such as real-time control, it is very important that the software to the interface boards be as efficient and reliable as possible.

#### 4. Bilateral Telemanipulation

The testbed configuration for bilateral telemanipulation is shown in Figure 1. The system consists of a master robot (HURBIRT) scaled to human arm motion and a slave robot (RALF) that has a workspace approximately fifty times the master robot's workspace. To isolate the human operator from the slave environment, the master and slave robots are located in different labs in the same building. This configuration allows the investigators to control the visual, acoustical, and tactile information that the operator might experience.

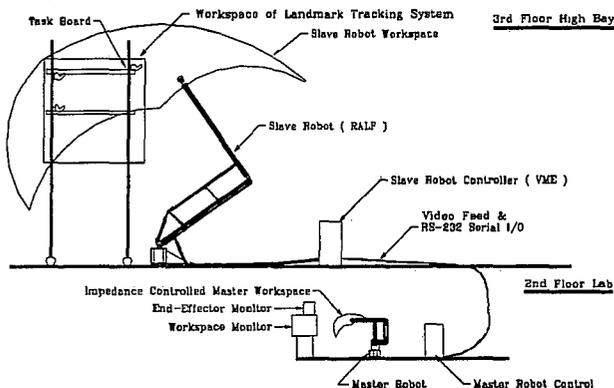


Figure 1: Long-reach teleoperation testbed

A modular scaffold next to the slave robot permits simple modifications to the slave robot's workspace and tasks. This task board can be configured for tasks such as teleoperated pick-and-place, constrained manipulation, remote path following, and basic assembly such as the peg-in-the-hole insertion problem. The operator views the motion of RALF on two monitors that display black

and white camera views of the slave robot's workspace. The first camera records a 6 m x 4.5 m vertical plane of motion from the side with a line of sight perpendicular to the robot's plane of motion. A 63 cm diagonal monitor displays this plane of motion. The second camera is mounted at the tip of the second link of RALF. This provides visual feedback of the robot's end-effector. A 23 cm diagonal monitor displays roughly a 36 cm x 25 cm rectangle in the plane of the end-effector.

To facilitate the teleoperation tasks, the controller for HURBIRT computes its tip position and scales the position from the space of the master robot to the space of the slave, RALF. Currently, a 7:1 position amplification permits comfortable mapping of RALF's full workspace into the workspace of the human operator. Once the desired tip position for RALF is calculated, the desired joint position is computed and then transmitted to the VMEbus for input to the slave robot's controller. Currently, data is transmitted via a high speed serial communication port every 10 ms at 38,400 baud.

Impedance control of the master robot facilitates force reflection and virtual fixtures in an intuitive way. The workspaces of the master and slave manipulators in Figure 1 are dissimilar. Simple tasks such as moving the slave robot to its home position prove to be difficult by visual cues alone. The target impedance of the master robot, using the philosophy of superimposing impedances [3], is augmented with virtual walls that constrain the operator from commanding the slave robot outside its workspace. The target impedance for the robot is defined in (1).

$$\mathbf{M}_t \ddot{\mathbf{x}}_m + \mathbf{B}_t \dot{\mathbf{x}}_m + \mathbf{F}_{vf} = \mathbf{F}_h + \frac{1}{A} \mathbf{F}_s \quad (1)$$

The target mass,  $M_t$ , and damping,  $B_t$ , control the ease with which the operator moves the master robot. The virtual force,  $F_{vf}$ , represents the repulsive force produced by deforming the virtual fixtures, in this case stiff walls constraining the effective workspace of the master robot, as illustrated in Figure 2. Four compliant circles replicate the limits of the slave robot's workspace mapped inside the master robot's workspace. If the operator manipulates inside the scaled slave robot's workspace, the robot effectively "feels" like a mass moving through a viscous fluid. However, if the human attempts to command the robot outside its workspace, the virtual walls attempt to push the operator back into the workspace. Extensions of this example can include constrained paths for guidance as well as sophisticated forms of obstacle avoidance. While this is not a direct form of force reflection, the use of virtual fixtures provides physical sensation of characteristics of the remote environment beyond the scope of direct force feedback.

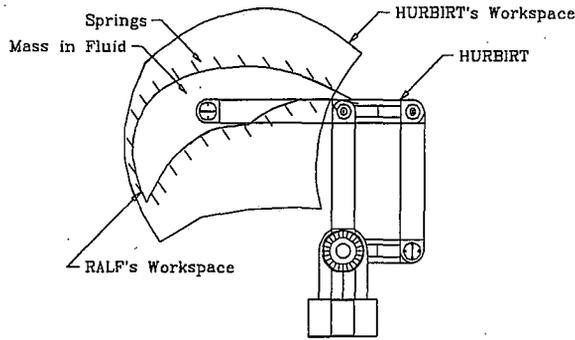


Figure 2: Virtual walls for bilateral teleoperation

The forces acting on slave and master are grouped on the right side of (1). The human force,  $F_h$ , is the force applied by the operator to the master which is measured using a two degree-of-freedom force sensor at the tip of HURBIRT. The environment force,  $F_e$ , provides the operator with tactile feedback of the interaction between slave and environment. It is measured using a three degree-of-freedom force sensor at the tip of RALF. Position commands from the master robot and force feedback from the slave robot are scaled by the amplification factor,  $A$ .

$$\mathbf{x}_s = A\mathbf{x}_m \quad (2)$$

## 5. Remote Capabilities

A remote display of the telemanipulated long-reach manipulator testbed, updated with real-time data through the Internet, was demonstrated at the 1995 DOE Robotics Forum in Albuquerque, New Mexico.

### 5.1. Modelling

In order to view a mechanical system in operation effectively, a three-dimensional computer model must be created which can be rapidly updated to reflect the changing system. A convenient package for modeling robotic operations is the Telegrip software package [11]. A model can be generated using standard CAD procedures to create geometric parts or CAD files can be directly imported. These parts are then assembled into a device and devices are arranged to form a workcell. Before realistic motion is possible the kinematics of the required devices must be established. This can be accomplished by entering explicit mathematical formulas for each of the devices' degrees of freedom or by allowing the software to automatically generate formulas

using supplied inverse kinematics algorithms. In this way, an entire work cell can be modeled and animated.

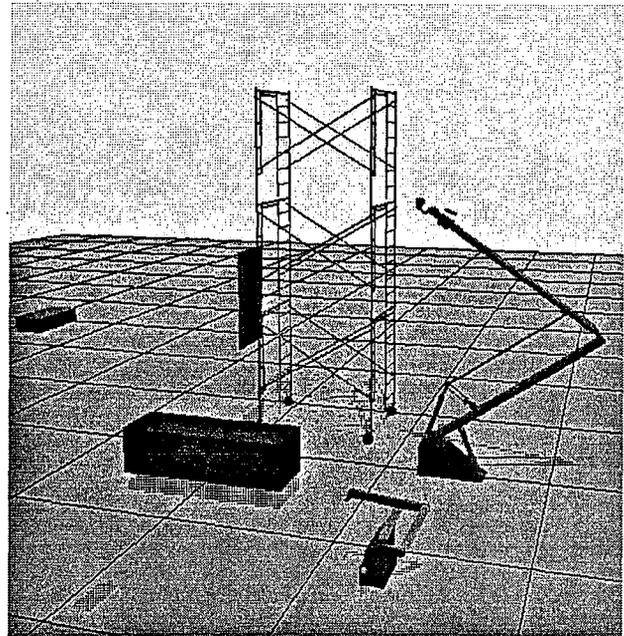


Figure 3: Telegrip screen capture of master robot HURBIRT (foreground) and slave robot RALF

Figure 3 shows a screen capture from a Telegrip session representing a typical display of RALF and HURBIRT. A model with this degree of complexity can be animated at refresh rates of approximately 3 Hz using an SGI Indigo2-XZ workstation. Simplified models that include only unadorned representations of the robotic devices, without fixed workspace props such as the scaffolding, achieved update rates of approximately 30 Hz. Both these rate measurements exclude data transport and acquisition times. During display of a graphical model a user may conveniently move the display's vantage point to any position in the workcell or smoothly translate and rotate the camera through the virtual world. The user might even invoke multiple views at once. In this way, a single computer screen can display the views seen by the operator of a telerobotic device as well as any other viewpoints that are required. Additional features include the ability to detect collisions between devices, simulate digital I/O communications logic between the various devices in a cell, and the ability to import data from external sources in real time. It is this last capability that was utilized for the demonstration at the Robotics Forum.

## 5.2. Communication Interface

The Internet is one medium through which data from a remote location can be transported. By far the most standardized approach for sending information across the Internet is through the use of TCP/IP sockets that send and receive packets of data using error checking algorithms that ensure exact transmission of the data. The length of these packets of data can be established by the user before beginning a session or may be allowed to fluctuate during an individual socket session. Before data may be transported, a socket connection must be established. One site must be designated as the client that initiates all exchanges of information and the other as a server that merely responds to any messages received from the client. Generally, the observing site is established as a client to the server at a site where the physical activity is taking place. Functions which implement a protocol to establish this link are standard to a wide variety of operating systems.

One disadvantage of the TCP/IP communications scheme is the variable amount of time required to transport an individual packet. Depending on the traffic at intervening Internet sites and the physical extent of the network between the client and server, an individual packet may require between 3 ms and several seconds to reach its destination. However, a transport time of several seconds is extremely rare and 50 to 100 ms is a much more average time for a packet's round trip across the entire United States. These time estimates represent the time required to transport a small packet where the amount of data is insignificant--less than one Kbyte. Packet rates can be improved by using specialized packet formats. Zero-copy packets allow numerical information to be transported without converting it into ASCII text during transport. Asynchronous packets do not preserve the chronological order of packets as they are sent and received. However, the standard TCP/IP protocol is sufficient for refresh rates in the 10 Hz range. For example, the transfer of a 256 byte packet from the VMEbus system at Georgia Tech (Atlanta) to Sandia (Albuquerque, NM) took approximately 50 ms on the day before the demonstration when network traffic was normal and the VMEbus system performed no control tasks.

The method by which a link is established between the graphical modeling software and the socket connection depends upon the particular modeling package. The Telegrip software provides the capability to incorporate user-written C functions into any workcell animation. This allows for the animating program to directly connect to the remote location where the physical parameters are being generated and request a packet of this information whenever it is possible to update the

model. The memory locations of the changing parameters can then be directly modified before requesting a simulation or graphics update. This method provides a high degree of efficiency and stability, allowing the observing user to easily connect and disconnect from the remote site using a graphical interface.

Regardless of the method selected for communicating data with the remote site, the controlling processor will be heavily tasked. The superfluous socket communications activity with the remote display site must be synchronized with any other processes which are required to control and monitor the device in motion. For our particular application, four separate processes were simultaneously and independently running on one CPU board within the VMEbus system: (1) low-level slave manipulator control, (2) receiving desired slave positions from the master arm and returning sensed force values via serial line, (3) maintenance of the socket communications connection with the remote site, and (4) end-point position readings from the landmark tracking system via serial communications. No one process can be allowed to monopolize the CPU and a proper prioritization of tasks must be implemented in order to avoid dangerous conflicts between processes.

## 5.3. Remote Demonstration Example

Three graphical models of devices at Georgia Tech were simultaneously viewed and animated at Sandia while they were being operated in Atlanta: RALF, HURBIRT, and a tip location pointer based on end-point position feedback from the landmark tracking system. Each of the three devices was allowed only two degrees of freedom. As a result fewer than 120 bytes of data were required to update the graphical model. Using a detailed model and an SGI Indigo2-XZ workstation an update rate of approximately 1 Hz was observed. This rate was inadequate for observing the vibrational motion that a flexible structure such as RALF undergoes, but did provide a vivid representation of the telerobotic operations performed and the static deflection of RALF's tip position.

In addition to the Telegrip simulation a video feed from the IMDL was displayed at Sandia at a rate of approximately 8 images per second using public domain CU-SeeMe software [2]. This software runs on IBM PC or Macintosh computers using very inexpensive video hardware. An image of size either 320x240 (used during demonstration) or 160x120 pixels with 4-bit gray levels is produced which is transmitted over the Internet. Compression allows for transmission of only the image pixels which change. The live video greatly enhances the sense of realness of the demonstration and facilitates

communication between the parties at the two locations. Further info is available via anonymous ftp from the host cu-seeme.cornell.edu in the directory /pub/video.

## 6. Conclusions and Recommendations

The predominant factors governing the display rate at the remote viewing site were the graphics refresh rate of the computer model implemented, the CPU load of the VMEbus system, and the amount of network traffic at the day of the demonstration. A faster graphics card or a less complex model would improve the graphics refresh rate significantly while a faster or an additional CPU board would reduce the load of the VMEbus system. With these improvements it should be possible to achieve display rates equal to that of the video feed even when network traffic is high. Less data is sent to update the computer model than to transmit images.

Remote supervisory control of the experiment configuration (choice of algorithms or parameters used) or the remote generation of command trajectories from the remote viewing site might also be implemented without much modification. This was demonstrated by others at the same Robotics Forum but using dedicated communication lines. This would allow for the excitation and observation of vibrations from the remote site directly. Sending data for remotely plotting graphs is another effective display that others have used.

Because of the flexibility of implementation and the inherent efficiency of data transport, remote modeling and viewing of robotic processes using this method provides many opportunities for development of useful applications and impressive visual demonstrations without unusual or extremely expensive additional hardware.

## Acknowledgments

This research was sponsored in part by Sandia National Laboratories with the cooperation of Pacific Northwest Laboratory and Oak Ridge National Laboratory through contract AK-9037 funded by the DOE Robotics Technology Development Program.

## References

- [1] Book, W. J. (1993) "Controlled Motion in an Elastic World", *ASME Journal of Dynamic Systems, Measurement, and Control*, Vol. 115, No. 2, June 1993, pp. 252-261.
- [2] CU-SeeMe for Windows Ver. 0.65b2, 7/5/95, Copyright Cornell University 1993, 1994, 1995.
- [3] Hogan, N. (1985) "Impedance Control: An Approach to Manipulation: Part III - Applications", *ASME Journal of Dynamic Systems, Measurement, and Control*, Vol. 107, No. 1, March 1985, pp. 17-24.
- [4] Huggins, J. D., Kwon, D.-S., Lee, J. W., and Book, W. J. (1987) "Alternative Modeling and Verification Techniques for a Large Flexible Arm", *Proceedings of the Conference on Applied Motion Control*, Minneapolis, Minnesota, USA, June 16-18, 1987, pp. 157-164.
- [5] Lew, J. Y. and Book, W. J. (1994) "Bracing Micro/Macro Manipulators Control", *Proceedings of the IEEE International Conference on Robotics and Automation*, Vol. 3, San Diego, California, USA, May 8-13, 1994, pp. 2362-2368.
- [6] Love, L. J. (1995) *Adaptive Impedance Control*, Ph.D. Dissertation, Georgia Institute of Technology, July 1995.
- [7] Love, L. J. and Book, W. J. (1995) "Environment Estimation for Enhanced Impedance Control", *Proceedings of the IEEE International Conference on Robotics and Automation*, Vol. 2, Nagoya, Aichi, Japan, May 21-27, 1995, pp. 1854-1859.
- [8] Magee, D. P. and Book, W. J. (1995) "Filtering Micro-Manipulator Wrist Commands to Prevent Flexible Base Motion", *Proceedings of the American Control Conference*, Vol. 1, Seattle, Washington, USA, June 21-23, 1995, pp. 924-928.
- [9] Obergfell, K. and Book, W. J. (1992) "End-Point Control of a Two-Link Flexible Manipulator Using the Landmark Tracking System", *Proceedings of the Japan/USA Symposium on Flexible Automation*, Vol. 1, San Francisco, California, USA, July 13-15, 1992, pp. 701-706.
- [10] Obergfell, K. and Book, W. J. (1994) "End-Point Measurements of Long-Reach Flexible Manipulators", *Symposium on Robot Control '94*, Vol. 2, Capri, Italy, September 19-21, 1994, pp. 669-674.
- [11] TELEGRIP Ver. 2.3, Deneb Robotics, Inc., Auburn Hills, Michigan, USA, 1994.
- [12] Yuan, B. S., Book, W. J., and Huggins, J. D. (1990) "Control of a Multi-Link Flexible Manipulator with a Decentralized Approach", *Proceedings of the 11<sup>th</sup> Triennial IFAC World Congress*, Tallinn, (formerly) USSR, August 13-17, 1990, pp. 235-241.