Integration of a Sensor Based Multiple Robot Environment for Space Applications: The Johnson Space Center Teleoperator Branch Robotics Laboratory

James Hwang, Perry Campbell, Mike Ross
Lockheed Engineering & Science Company
Houston, Texas 77058

Charles R. Price, Don Barron
Johnson Space Center
NASA

ABSTRACT

The Teleoperator Systems Branch at JSC has developed a robotics laboratory for space robotics technology development. An integrated operating environment was designed to incorporate three general purpose robots, sensors and end-effectors, including Force/Torque Sensors, Tactile Array sensors, Tactile force sensors and Force-sensing grippers. This paper describes the the design and implementation of: (1) the teleoperation of a general purpose PUMA robot, (2) an integrated sensor hardware/software system, (3) the force-sensing gripper control, (4) the host computer system for dual Robotics Research arms, and (5) the Ethernet integration. The space applications for the above projects will be discussed and the future development for the laboratory will be presented.

INTRODUCTION

The JSC telerobotic program aims at achieving higher levels of autonomy in space operations and to demonstrate the potential of significantly enhancing the agency's current robotic operational capabilities. The near term objective is to provide a more versatile telerobotics system that interacts compliantly with its environment. It is also desirable to have a higher safety system by providing fault-tolerance and obstacle avoidance schemes. Relative to the these goals, the program elements consist of:

- sensor hardware and software technology: vision,tactile, proximity, force/torque.
- fault diagnosis and planning.
- computing architecture and planning.
- telerobotic demonstration.

The Robotics Laboratory for the Teleoperator Systems Branch was founded in April, 1987. The objective for this Lab is to provide a flexible hardware/software testbed environment for various technology demonstrations and to fulfill the goal of supporting the Space Shuttle and the Space Station Freedom Program [1].

* This paper presents the initial results of work carried out at the JSC, LESC, under Contract No. NAS9-17900, JO. 22-110, sponsored by the National Aeronautics and Space Administration.

LABORATORY CONFIGURATION OVERVIEW

The Robotics Lab of the Teleoperator Systems Branch is located in Building 16, Room 2000 at Johnson Space Center. This Lab currently contains 3 large robots, including a PUMA 762, a Robotics Research (RR) 1607, and an RR 2107. There are also several small educational robots, such as four Microbots, one early model PUMA 250, and a Hero 2000, for internal training purposes. Various sensors, end-effectors and grippers have been installed on these robots. The following section will give a brief description for each of these sub-systems.

PUMA 762 ROBOT

The PUMA 762 manipulator is an industrial robot made by Unimation Corp. It is a six degree of freedom (DOF) robot with about 5' reach and 44 lb. of payload capacity. The controller supplied by the manufacturer is an integrated control and programming unit. The robot can be controlled by a teach pendant or by a program sequence through the VAL II programming language. Since the robot was designed basically for pick-and-place type of applications, only point-to-point motions are supported directly from the VAL II language.

Robotics Research 1607 and 2107 Manipulators

The Robotics Research robots 1607 and 2107 are made by Robotics Research Corp. The 2107 has about 2100 mm reach and the 1607 has 1600 mm. The 2107 can hold only 4 lb of payload while 1607 is able to handle 50 lb. These two robots have 7 DOF and a very unique roll-pitch-roll-pitch-roll-pitch-roll configuration. The extra degree of freedom provides the capability of ORBITING the elbow while keeping the end-effector frame at a fixed position and orientation. The Type II controller with the robot provides basic teach and replay controls of the robot. No robot programming language is available for control sequence generation. Also only the point-to-point motion control is supported at this time.

Force Torque Sensors and Tactile Array Sensors

The PUMA manipulator is equipped with a LORD 125/600 Force/Torque Sensor. It has 125 lb of force capacity and 600 in-lb torque capacity. The other two manipulators are equipped with JR3 UFS-4A-XX Force/Torque Sensors. One JR3 sensor has 100 lb capacity and the other one has 25 lb capacity. The transducers of these force/torque sensors are similar, consisting of a maltese-cross based strain gauge assembly which provides a set of signals which contain the force/torque data. In addition to force/torque sensors, a tactile array sensor made by Lord Corp. was also interfaced to the host computer. The tactile array sensor consists of a 10 by 16 element grid array of closely spaced deflection-measuring sensors. These sensors yield information about the contours of the object they are in contact with. All three force/torque sensors and the tactile array sensor are communicating with the host control computer through RS-232 serial lines.

Host Computer

For the initial build, each of three robots were integrated to a PC/AT 286. These PC's acted as sensor controllers and programming interfaces. As mentioned above, all of those robots were designed for industrial applications, so they are not easily adapted to sensor interfacing. Because a generic programming environment was required to interface to different types of sensors and hand controllers, as well as using this sensor information to control the robot motion, a host computer was needed. For instance, a robotic system may be required to obtain the motion command from the hand controller; to collect data from the force/torque sensor, the video sensor and the tactile sensor; to display sensor data to the screen, pass the motion command to the robot, and to command the motion of the end-effector. All these activities should be controlled by a master program so that all the interactions will be coordinated. The robot programming language usually can only handle the manipulator motion well but not the sensor or end effector control. To coordinate all activities of the robot, sensors, and end-effectors, a PC/AT was used to host all these control programs and the "C" language was used to develop programs.

Communications

There were at least two classes of communication needed in order to implement a telerobotic system in the Lab. The first class was the communication from local sensors to the control computer and the second was the communication among several robot control computers and the workstation. Most sensors in this Lab had RS-232 serial links (some of them even with analog output) and all three robots in the Lab were also equipped with RS-232 ports, therefore the RS-232 was used for the local communication among them. Because a regular PC/AT supported only two serial ports at 19.2 K Baud, a more capable communication device was needed. The Advanced Communication Link (ACL) by Stargate Corp. was selected to build the RS-232 communication sub-system. The ACL is actually a front-end communication processor which has a CPU on board and can handle up to 8 serial ports with Baud rate up to 38.4 K Baud. Since it has its own memory buffer for each channel and the processor on board will service the message transmission and receiving in background, the main CPU has very little overhead. A special ACL software driver was designed for all the sensor interfacing.

One of the major goals for this Lab is to provide a real manipulator operating environment for various Display and Control Workstations other than the simulated graphic scene generator. For providing a communication vehicle among these workstations and controllers, peer-to-peer networking is the most cost-effective approach. The EtherNet and the TCP/IP protocol are good candidates to implement such a network. The physical speed limit of EtherNet is 10 Mbit/sec, although 1 Mbit/sec is a more realistic estimate of the actual data rate with the overhead from many layers of protocol. The EtherNet provides a very convenient way to add a new node without disturbing the existing communication. The communication from any workstation to any manipulator can also be re-configured without any hardware change. The TCP/IP protocol adds further versatility of networking among different kinds of computers with a single programming interface. The intelligent EtherNet controller, Thick Net EtherNet, and the TCP/IP C socket library by ExceLan Co. were then selected to build this network. Software drivers for message passing were designed and implemented on the PC/AT and several workstations.

Hand Controllers

The hand controller is still an indispensable part of the whole telerobotics system. Two kinds of hand controllers, position and rate, were tested and integrated to manipulator systems. The "Space Mouse" was the first one been tested in the position controller group. It was constructed from a 6 DOF electro-magnetic field sensor made by McDonnell Douglas, named "3-SPACE ISOTRACK". Only a thin cable is connected between the free floating hand-controller and the sensor control box. The RS-232 link was used for sensor-to-control computer communication. Another type of position hand controller in the process of being integrated into the system is the Schilling Minimaster controller. A pair of Schilling 6-DOF hand controllers mounted on a common base were delivered to the Display & Control Lab. Using the Minimaster to drive a graphic simulation has been tested. The next step will be interfacing this hand controller to manipulators through the EtherNet. The other group, rate hand controllers, was also tested. Two MSI hand controllers, one a rotational hand grip, and the other a translational T bar, were interfaced to the PUMA control computer. The phase one design used an A/D interface board to measure the deflection of the MSI controller since only raw voltage output from potentiometers was available. The phase two design is still in progress; it will use a digital serial interface instead of an analog interface, and an embedded single board computer will be used to perform the analog-to-digital conversion and digital data transmission function.

End-Effectors

In order to make the robot interact with the environment and any size of payload, a general purpose end-effector is needed. The gripper, instead of a special tool, was identified as the most generic form of a end-effector. Such a gripper needs to be electrically servo driven with accurate position feedback, and with force sensing capability. When a survey was conducted to locate this kind of industrial gripper, it was a surprise that very few manufacturer made general purpose electrical servo grippers, and the TeleRobotics Inc.(TRI) was the only vendor could provide both the position and the force feedback. Two TRI EP100/30 grippers were then acquired to be used on the PUMA and the RR 1607 manipulators. These two grippers can exert 66 lb. of gripping force and their fingers can open up to 4". One EP90/05P gripper was acquired for the RR 2107 manipulator. It is lighter (weighs about 3 lb without cable) and shorter for very light payload capability of the RR 2107 manipulator. All fingers of the above three grippers have strain gauges to measure the gripping force. The gripper controller offered by the vendor is a stand-alone control box which is controlled from a serial port through Text String commands. There are, however, some major design flaws of this gripper controller. For examples, it does control the force actively but only stops the gripper when the force has exceeded a preset threshold. In other words, it is not capable of gripping a rigid object while maintaining a constant force. Some other flaws exist like not always responding to a position interrogation command on time, and closing the gr$per without recognizing the force limit. Finally, it was decided an in-house gripper controller should be designed and built. More descriptions of this new controller design will be discussed in the following section.

PROJECTS and SPACE APPLICATIONS

Compound Arm Project and Teleoperation Of PUMA

The first project, named "Compound Arm Project", built in this Lab was to implement the teleoperator mode for the PUMA robot and to control a special end-effector constructed from two Microbots mounted on a common bracket, which was then installed on the tool flange of the PUMA [2].

The task was to control the PUMA by the Space Mouse Hand Controller and to move the PUMA around until the tool point had entered an imaginary washout sphere, whereupon an auto sequence would take control of robots, and execute a changeout procedure of some batteries on a mockup satellite. This simulated the idea of having a smaller dexterous dual arm robot carried by a long boom manipulator to do some servicing work, such as the Flight Telerobot Servicer (FTS) or Special Purpose Dexterous Manipulator (SPDM) on the end of the Space Station Manipulator. Additionally, the PUMA continually acquired arm position data and passed that data back to the display PC which would draw the PUMA's configuration on the screen.

The first problem which had to be solved was that the VAL II controller for the PUMA itself does not support any kind of communication utility even though several serial ports are available. In order to implement a teleoperator mode, it was necessary to establish communication between the hand controller and the robot controller. It was decided to use a host computer (a PC/AT) to control the traffic. Two sets of communication software drivers were developed: one for the VAL II controller, the other for the ACL processor installed on the PC/AT. The driver for the VAL II was written in the LSI-11 assembly language as an interrupt handler, and it was then assembled by a cross-assembler on a VAX11/785 and finally POKEed into memory by a VAL program. A segment of memory was allocated to build four ring buffers for four serial channels. Another VAL program was also developed to pull the message from the ring buffer and to execute motion commands according to the message. On the PC side, a similar program was written to submit messages to the memory ring buffer in the PC, then the message was passed to the serial port by the ACL Communication Processor. After the communication channel was established, the control PC was able to acquire 6 position data from the Space Mouse. That data, being very noisy, was filtered, processed, and appended with error checking code before being passed to the PUMA to direct the arm's position. The display PC would receive the PUMA's joint angles from the VAL controller and draw a view of the PUMA and its environment. This was to aid the operator in remote teleoperation.

Sensor and Force Sensing Gripper Integration

After the Compound Arm Project was accomplished and demonstrated many times, several drawbacks were found during the demonstrations. The most significant problem was that the motion of the PUMA was commanded through a string of "MOVES" VAL commands, which only direct the arm through a sequence of point-to-point motions. No continuous velocity control was available, so that the motion appeared very sluggish and fine motions were very difficult to control. Another problem was that there was no force sensing capability at the end-effector, some battery mock-ups were demolished.

The second project was to integrate all the available sensors to the robot control system and to develop a more accurate control scheme. Software drivers were developed for all of the sensor and the gripper. The force/torque sensor information was displayed in a bar-graph form; the tactile sensor data was displayed as a 16 by 10 array with color representing the displacements of each sensing element; the gripper position was displayed as a picture of moving fingers; then all of these graph were shown on a windowed display screen. A demonstration was developed to show actual application of the force sensing gripper. The robot was guided through and picked up a number of different objects, including a real egg shell, a brick, and a soft drink can. The demonstration displayed the necessity to use a force sensing gripper to hold a fragile egg without breaking it, lifting a heavy brick without losing the grip, and finally picking up a soft drink can, pouring the contents into a cup, then crushing the can and dropping it into a trash can [3].

The above sequence was still controlled through a set of pre-programmed points and the PUMA was still operated under Point-to-Point Move mode. In order to control the PUMA with a smooth, continuous trajectory, a synchronous continuous path control mode was developed. Although the PUMA, or any industrial robot, was not designed to be operated from a Hand Controller, it did provide a special synchronous control mode called "Alter". The major benefit of the synchronous mode over the asynchronous point-to-point mode is that it can control all the intermediate points between the end-points and the time between each intermediate point is fixed. In this mode, the control computer has to be 'synchronous' with the robot, and the robot controller will ask for a set of command position every fixed time period (28 ms for the PUMA), then the control computer must respond to the robot controller by giving the next command position and also receive a current position report. The control computer, however, has to keep up with the speed of the robot and not lose the synchronization, otherwise the motion will come to a stall. The Alter mode for the PUMA was designed for altering the programmed path by applying the sensor data. To adapt this mode for the Hand Controller application, the PUMA was programmed to move to the same position forever and then the Alter mode was used to feed the Hand Controller command. This mode was tested with the Force/Torque Control and some demonstrations were developed for these studies. A separate paper will present results of this study [4] [6].

Workstations Integration

As mentioned in the earlier section, this Lab has the reponsibility to support the Display and Control section to establish a reconfigurable manipulator operation environment for their workstations. Two projects were in process of integration of workstations and the manipulator systems. The first one is the Vision Integration Project (VIP), which is to integrate the manipulator system with a Computer Aided Design (CAD) model based workstation. The workstation was supposed to know the 3D model of every object in its view. A 3D vision system, including some cameras, frame grabbers, and pattern recognition software, will identify the object, calculate the Cartesian coordinate of the object and send it to the workstation. The workstation will re-generate the scene of the world out of vision data and display the scene on the screen. The raw camera image will also be available to the operator. The operator can then manipulate the robot from the simulated world scene with much more rich information than the raw video image can contain. For example, the operator can rotate the viewing angle, look inside of the object, or predict the center of mass before any motion. To date, the communication and teleoperation portion of the project have been accomplished. The workstation (a IRIS 4D/70GT) was used to control the teleoperation and vision data interface. The EtherNet and TCP/IP protocol were used to implement the networking between the robot control computer and the IRIS. The IRIS was able to command the motion of the PUMA robot across the EtherNet. The Vision Process is still being developed [5].

Another project is to integrate the Multi-Purpose Applications Console (MPAC), which is being developed by the Display and Control section, to the robot system. Since the host computer (a microVax) for the MPAC is also on the same EtherNet and using TCP/IP, the similar protocol as the VIP will be used for this project.

SRMS Advanced Control Project Support

The Shuttle Remote Manipulator System (SRMS) Advanced Force/Torque Control Study is a new project in the Teleoperator systems branch. The objective for this project is to demonstrate the Advanced Closed Loop Control by application of OAST/JPL developed Force/Torque sensor and control algorithms to the Shuttle RMS in order to influence definition of future RMS upgrades. The role of the Lab for supporting this project is to provide a laboratory manipulator environment to test and validate the algorithm developed. For the initial build, the Robotics Research Manipulator and JR3 Force/Torque sensor will be used to implement these algorithms. Since the dynamic charateristics are not well defined for RR arms, model identification was the first task for this project. Some tests were done on identifying the servo control parameters, more tests have been scheduled to identify the link inertias and friction.

In order to simulate the RMS with an industrial arm like RR robot as close as possible, some issues have to be resolved. The most unique characteristics of the RMS is that it uses joint rate control with tachometer feedback instead of the common joint position control for most industrial robots. Therefore a plan has been set up to implement a rate control mode by modifying the RR controller.

NASREM Implementation

NASA/NBS Standard Reference Model (NASREM) Architecture is a proposed model to implement the Space Station Telerobot Control System [7]. It was developed by the joint effort of NASA and the National Bureau of Standards. It defines the functional requirements and high level specifications of the control system for the NASA space Station Flight Telerobot Servicer. The recommended architecture is, however, very generic and suitable for any space telerobotic control system. In this Lab, a project has been planned to implement most levels of control architecture of the NASREM model using the existing manipulators and sensor systems. The objective of this project is to verify the NASREM model in a real manipulator environment to demonstrate the advantage of a hierarchical control structure, as well as to investigate any issues arising in the implementation. These studies will assist in establishing the telerobot operation procedures and the specifications of the future space station robots.

In order to implement the NASREM model on existing industrial robots, many problems have to be overcome. The regular RR robot controller does not provide the capabilities of tuning the servo parameters, interrogating the rate and torque feedback, and controlling the manipulator at joint rate or joint torque level. For higher level functions, such as dual arm coordination control and collision avoidance, the processors in the RR controller just can not provide the high number crunching capablity required. A plan has been drafted to overhaul the RR controller so that it will provide the taps to control and measure the signals of absolute positions, joint rates, and joint torques in digital format. The acquisition of the high level host computer, an Iris 4D/70G workstation, for the RR controller has also been initiated. To match the NASREM architecture, several levels of processors will be used to implement different control levels in the Model.

The lowest level in NASREM is the servo level. To implement this level, the servo level interface upgrade will be acquired from the Robotics Research Corp. The upgrade will provide the digital taps to control the robot at the rate or the torque level. However, the functions of low level servo analog compensators, which keep the arm stable, must to be realized digitally every 2 ms. The original processor can not provide this computation power, so another dedicated processor has to be used for the servo transfer functions. The TMS320C25 digital signal processor (DSP) by Texas Instruments was selected to do the low level servo functions. Initial simulations studies have shown very encouraging results: the DSP is able to complete the calculation of current analog servo functions for 7 joints within 500 micro-second. It means that not only a programmable servo controller can be realized, also more advanced control algorithm might be incorporated as well.

The level a step higher than servo level is the Primitive. At this level, the joint trajectory is decomposed to small motions and sent to the servo controller. Sometimes the inverse kinematics is also solved in this level. The loop time for this level is about

25 ms, which is still synchronous. This level of control is planned to be implemented by a MultiBus embedded processor board. This processor will be equipped with a complete disk operating system (DOS) for the high level language program development. At this level, the Force/Torque Sensor data will be integrated to the control law.

The E-move level and the Object level will be implemented in the IRIS computer. At these two levels, the world model will be established, the Vision data will be incorporated, the collision avoidance will be planned from the model, and the dual coordination task will be executed.

There are three more levels above the Object level, the Task, Service Bay, and the Mission. Some functions of the task level controller can still be done by the IRIS computer, while the other higher control levels will need an AI based machine to coordinate the operations of workstation, the operator side, and the operation of the manipulator, the manipulator side. These functions will be expanded in the future.

IN-HOUSE GRIPPER CONTROLLER DESIGN

The gripper controller which came with the TRI gripper was not sufficient to implement all the functions needed. Therefore, an in-house design of the gripper controller was planned. The phase one design of this project was to build a controller based on PC using off-the-shelf components as much as possible. A PC based motor controller with optical encoder decoder and a PWM power amplifier were the only hardware components needed to build such a controller. The force sensing circuit on the gripper was modified to incorporate the A to D circuits on board. After these modifications, all the sensing signals and control signals were digital and could be controlled by the PC directly. A software driver was written for this controller to incorporate all the functions in the old controller in C functions format. New functions were also added such as the active finger force control, the motion abort capability, the synchronous positioning, and error code returning. The phase two design will be moving all the hardware into a stand alone control box with a processor much like the old controller design. However, all the improvement and added functions will be integrated to the new design. The control computer will communicate to this controller through a serial link or some digital I/O lines for predefined motions.

CONCLUSION

This paper has given a overview for the Robotics Laboratory of Teleoperator Systems Branch in JSC. Various space telerobotic application research projects have been initiated in this Lab. For the first year and half, most effort has been focused on setting up an integrated sensor based manipulator testbed environment. In order to further support the Space Shuttle and the Space Station Freedom Programs, the spectrum of research has been directed toward the technology development for demands from space programs. In the future, the lab will emphasize on the following research aspects:

- Hierarchical robot control architecture implementation.

- Coordinated multisensory perception (sensor fusion).
- Force/torque and tactile sensing for dexterous manipulator.
- Expert system for automated task planning, and collision avoidance.
- Model-based Telerobotic Control.
- Multiple arms coordination.
- Fault-tolerant manipulator systems.
- 2D, 3D vision and range imaging.

REFERENCES

[1] C. J. Hwang, "PROPOSAL FOR OBJECTIVES OF ROBOTICS LABORATORY", LEMSCO, Houston,TX, July 1987.

[2] M. Ross, P. Campbell, "ROBOTICS LABORATORY COMMUNICATION SUPPORT FOR COMPOUND ARM PROJECT", LEMSCO-25238, June 1988.

[3] M. Ross, P. Campbell, "ROBOTICS LABORATORY SENSOR AND GRIPPER INTEGRATION PROJECT", LEMSCO-25246, July 1988.

[4] P. Campbell, "ROBOTICS LABORATORY FORCE FEEDBACK PROJECT", LEMSCO-25251, September 1988.

[5] M. Ross, "PROTOTYPE OF REMOTE NETWORK MANIPULATOR CONTROL FOR EF2 VISION INTEGRATION PROJECT", LEMSCO-26071, October 1988.

[6] P. Campbell, "REAL-TIME CARTESIAN FORCE FEEDBACK CONTROL OF A TELEOPERATED ROBOT", NASA Conference on Space Telerobotics, Pasadena, CA, January 1989.

[7] J. S. Albus, H. G. McCain, R. Lumia, "NASA/NBS STANDARD REFERENCE MODEL FOR TELEROBOT CONTROL SYSTEM ARCHITECTURE (NASREM)", National Bureau of Standards Robot Systems Division, December, 1986