CONF-8310134--2

A SYSTEM FOR INTELLIGENT TELEOPERATION RESEARCH

Nancy E. Orlando
Automation Technology Branch
NASA-Langley Research Center
Hampton, Virginia

CONF-8310134--2

DE85 902189

Presented at the AIAA Computers in Aerospace IV Conference

Hartford, Connecticut
October 25, 1983

A SYSTEM FOR INTELLIGENT TELEOPERATION RESEARCH

Nancy E. Orlando
NASA Langley Research Center
Hampton, Virginia

Abstract

The Automation Technology Branch of NASA Langley Research Center is developing a research capability in the field of artificial intelligence, particularly as applicable in teleoperator/robotics development for remote space operations. As a testbed for experimentation in these areas, a system concept has been developed and is being implemented. This system, termed DAISIE (Distributed Artificially Intelligent System for Interacting with the Environment), interfaces the key processes of perception, reasoning, and manipulation by linking hardware sensors and manipulators to a modular artificial intelligence (AI) software system in a hierarchical control structure. Verification experiments have been performed: one experiment used a blocksworld database and planner embedded in the DAISIE system to intelligently manipulate a simple physical environment; the other experiment implemented a joint-space collision avoidance algorithm. Continued system development is planned.

Research Objectives

The Automation Technology Branch (ATB) of NASA Langley Research Center is presently involved in researching the use of teleoperator (remotely controlled manipulation) devices for remote space application.[1] The expense, the time limitations, and the high personal risk of astronauts doing extravehicular activities in space are necessitating the development of alternative means of accomplishing such tasks as satellite servicing and station construction. A logical alternative is the development of teleoperator devices that can accomplish the tasks through remote operator input, and that can also be expanded through automation techniques toward the realization of autonomous, or robotic, devices. It is expected that teleoperator device capability will increase with technological advances, the eventual goal being man as a high-level, goal-setting supervisor of such a robotic device. Toward this realization, the ATB has established a system level research program in teleoperation and robotics, with the initial focus on a satellite servicing system. The system level research is conducted using a Teleoperator/Robotic System Simulation (TRSS) which is a modular software simulation coupled to a reconfigurable teleoperator control station. Preliminary results from experiments run with this simulation indicate that TRSS serve as an invaluable tool in identifying the necessary specifications for a working space teleoperator system.[2&3] The branch has also developed the Intelligent Systems Research Lab in which actual hardware is evaluated and which allows verification of the software modules of TRSS. Within the systems level research, areas under investigation include: 1) Manipulator dynamics and control, 2) End effectors, 3) Sensors, 4) Operator-machine interface to automated systems, 5) Distributed computers and network systems and 6) Artificial intelligence. The research in artificial intelligence and its interaction with the other on-going research in the ATB is the subject of the remainder of this paper.

Langley's automation research group is convinced that some minimal degree of supervisory control/automation will be necessary for the successful completion of space teleoperation missions. Most teleoperator devices that exist today are simple, awkward extensions of different manned control. In the performance of even the simplest task, the operator is still totally responsible for tedious, repetitious, and needlessly time-consuming task primitives, such as inserting a screw or removing a nut from a bolt. The development and application of machine intelligence techniques will enable the human operator to move to higher and higher levels of supervisory control of the teleoperator system. Eventually, refinement of artificial intelligence applications will lead to relatively autonomous, goal-driven robotic devices that will be capable of sophisticated remote operations such as satellite servicing and space construction. However, it must also be stressed that a totally autonomous or automatic system is not felt to be required at present for successful mission completion. It is still an open research question as to what degree of machine intelligence is necessary for the successful completion of a particular task.

For a teleoperator system to deal effectively with delicate remote applications, it must be able to simultaneously and accurately perceive, reason about, and interact with its environment. Many artificial intelligence research programs deal with verifying one specific technique, and usually only interface to the "real world" through language, i.e., terminal I/O. Typically these programs cannot easily be adapted to encompass generalized real operations. On the other hand, most manipulator control systems are rather narrowly defined processes, and their capability of expansion towards artificial intelligence horizons is limited. Other research groups, notably Sheridan at MIT, have experimented with limited supervisory control techniques for manipulators.[4] A few efforts have been made, particularly at Stanford and at MIT, to actually run certain AI programs dealing with robot planning and collision avoidance using manipulator hardware. In addition, many industries have researched the use of machine vision with manipulators. However, very little systems level research has been done in teleoperator/robotics control. Therefore, the objective of Langley's artificial intelligence research is to provide an integrated system that will encompass all facets of interfacing the reasoning capabilities of the system with its environment of interest. This will include developing algorithms for reasoning processes, knowledge acquisition, and database construction and management. Such a system will ultimately provide a testbed for AI techniques and algorithms that will be useful in supervisory and autonomous control of remote manipulation systems. Specifically, research is planned in such areas as:

1. Knowledge representation - What type of information, in how much detail, is needed by each element of the system to give satisfactory performance by the system? How can this information be structured to reduce access time and yet provide for expansion as the system acquires more knowledge? Is "time-tagging" of information necessary, i.e., should there be temporary versus permanent types of information, or any degree in between?

2. Intelligent planning - How is the control of a distributed intelligent system divided among different processing components to provide both the speed and the accuracy necessary for remote operations? How do proposed goal-planning systems perform in a time- and materials-critical environment? How would an effective planning algorithm be structured?

3. Operator-machine interface - How much information, of what nature, is required by the operator to control such a device, and how is this displayed? As the machine increases in autonomy by degrees, how much control is left to the operator at each level and how is this control effectively provided? How can the operator override computer inputs when necessary, and conversely, how can the computer protect a delicate environment from erroneous operator inputs?

4. System integration - What type of control structure is necessary or preferred for such a system? How are perception, reasoning and manipulation components interconnected to be satisfactorily operational and yet to maintain a modular system? How is a query-driven structure balanced with an interrupt-driven structure in an AI-controlled hardware system?

System Implementation

System Structure

The intermediate goal of the artificial intelligence research of the Automation Technology Branch at Langley is the development of a system that will integrate the three components of perception, reasoning, and manipulation. The approach has been to build a segmented module which consists of the "reasoning" portion of the system and that interfaces with the perception and manipulation elements of the Teleoperator/Robotic System Simulation (TRSS) already being developed by the ATB. This segmented module resides in a relatively powerful processor that can coordinate communication from other processors and devices. The module encompasses segments that deal with "strategic" task planning, database management of the machine's concept of the environment, supervisory monitoring of the teleoperator control station, and the interfaces to various "tactical" controllers. This approach has been formulated into the concept of a total system, which has been termed DAISIE, for Distributed Artificially Intelligent System for Interacting with the Environment.

A predominantly hierarchical control structure has been chosen for the DAISIE system. This structure is similar in concept to Albus' work at National Bureau of Standards and implements successive levels of task decomposition with feedback at every level.[5] This structure is realized through careful construction of software and hardware components within "strategic" and "tactical" control modules. The terms strategic, and tactical, borrowed from military terminology, require specification within the scope of this paper. Obviously, strategic and tactical are relative terms, depending upon the level of command. Each level tends to view itself as "strategic" in some sense, with all lower levels being "tactical" in some sense. For the purpose of this paper, however, the system will be viewed as a totality, and the terms strategic and tactical will be applied to global system control and local unit control respectively. These two sublevels of the hierarchy can be conceived as similar to the voluntary/involuntary responses in animal systems. In such systems, the strategic, or voluntary, level is considered to have the higher level of intelligence, yet perhaps be unable to respond in real time. The tactical, or involuntary, level must respond more primitively to a real, time-critical environment. It will become apparent as the system development is explained that within these two sublevels of the hierarchy are many more successive layers of hierarchical control as well. This type of high-level structure can allow the system to be neat and modular, thus readily adaptable to new algorithms. However, within the outer hierarchical structure, particularly within the strategic module, methods should exist for utilizing current artificial intelligence algorithms using data-driven, event-driven, and model-referencing techniques.

Figure 1 represents the generic structure and interaction between modules of the system. One primary module of this system is the strategic planner, which handles all the "intelligent" functions of the system, and which interfaces with an extensive world model database. This strategic module interfaces through a communications coordinator to the several tactical planning modules of the system. These tactical planners appear as software modules in distributed processors which actually control the peripheral sensors and actuators, such as the manipulator processors, the vision module processor, and the end-effector processors. Commands are generated at the strategic level, and result in a chain of commands at each tactical level, each command in turn resulting in the suitable activation of the hardware. Status is then returned to the appropriate command level for interpretation, and the distribution of the next set of commands is initiated.

Hardware Structure

The development of this system is being done in the Intelligent Systems Research Lab (ISRL), which was developed and is supported by the Automation Technology Branch at NASA Langley Research Center. The particular resources pertinent to this development are listed below:

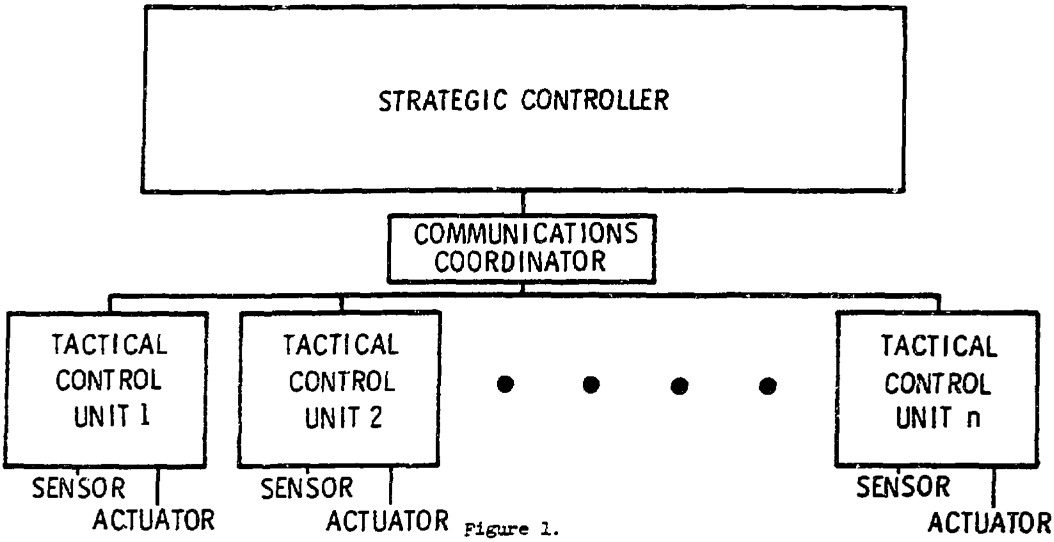

Figure 1.

1. Computer resources - The ISRL has a loosely-coupled network of 7 PDP 11/02 computers, which do the majority of control work for the various special purpose peripherals. This network, called RTNET, was developed in-house and provides uniform communications among the processors. In addition, a VAX/750 has been acquired for major software development tasks and supervisory control problems. Language support in the lab is extensive and includes several dialects each of LISP, FORTRAN, PASCAL, and Assembler, as well as many software development tool packages. Links also exist between the ISRL computers and the Langley central computing facility resources.

2. Manipulators - The ISRL has two Unimate 600 robot arms, known as PUMAS. Each PUMA has a LSI 11/2 as a controller, which can be used as a stand-alone unit or interfaced into the lab network. Each controller in turn supervises six 6502 microprocessors, each of which controls the servo mechanism of a single joint of the PUMA. FORTRAN-callable subroutines developed at NASA Langley are available to move the arm point-to-point by terminal input, or in a joint-by-joint or a rate-controlled mode by the joysticks of the teleoperator control station.

3. End-effectors - The ISRL is presently developing a version of the University of Rhode Island (URI) parallel-jaw-gripper end-effector which is equipped with proximity sensors and which is microprocessor-controlled. Until that system is fully operational, the DAISIE routines are using an early URI design of a surface-adapting vacuum-gripper end-effector.[6] This end-effector has been interfaced to a PUMA control unit and can be activated by specific servo commands.

4. Vision capabilities - The ISRL has a Stanford Research Institute (SRI) vision module with the accompanying software package. Work is in progress to adapt this system to provide vision feedback for control of the manipulators. Another effort is also underway to develop a full tactical controller for a vision module, using an Octek image processor.

5. Voice capabilities - The ISRL has both a VOTERM voice recognition module and a VOTRAX voice synthesis unit. Plans include the eventual control of certain functions by voice input, and the alerting of the operator to certain error conditions by voice synthesis.

6. Reconfigurable operator control station - A control station containing all operator displays and controls is in development in the ISRL. This station will eventually play a major role as an integrated tactical unit of the DAISIE system. Through this facility, many questions of human factors in the control of intelligent teleoperation systems can be resolved.

Software Structure

The DAISIE structure is designed to be language independent, allowing research and implementation of algorithms in the language deemed to be best suited for a particular task. Different software modules of the system currently exist in different languages, predominantly in LISP, FORTRAN, and assembler; modules are also coming on-line in PASCAL and PATH PASCAL.

The software for the basic DAISIE system is presently divided into three sections; the strategic control portion, the tactical control portions, and the communication coordinator. Each of these sections in turn has several modules.

Strategic Controller. The strategic controller consists of four distinct modules:

1. Strategic presentation layer - This module exists as a small set of LISP functions that package the low-level LISP commands into structured command strings for transmission to the tactical controllers. These functions also receive requested status information from the communications coordinator and make this status available to the strategic controller.

2. Lisp command bases - These modules are sets of low-level LISP functions which can be called from higher-level stategic commands, and which result in the generation of commands for each tactical unit. These sets also include functions for requesting status information and for packaging it for use by higher levels of control. Note that a single LISP command base typically corresponds to a functionally unique tactical control unit.

3. World model database - This module incorporates all pertinent information about the system environment, allowing the strategic planner to make informed decisions about maneuvers. This information is also available upon demand to the human operator. Interfaces with various sensors are critical for this function, and the issues of knowledge representation and access are critical research items.

4. Planning elements - These modules form templates for various algorithms that can be researched using the DAISIE system. Here is where goal-oriented planning, supervisory control interfaces, learning, automatic programming, and generalization techniques can be implemented to develop higher and higher degrees of autonomous control.

Tactical Controllers. The exact form of each tactical controller will vary somewhat depending upon the function of the unit. An attempt has been made, however, to design a structure that will be useable with a heterogeneous combination of tactical control hardware. For example, the manipulator tactical control unit can be conceptualized as containing three software modules:

1. Tactical presentation layer - These FORTRAN subroutines accept command strings from the communications coordinator and call the appropriate tactical control sequence. This module also responds to the communications coordinator with the appropriate success/status information. An important function of this module is to convert spatial coordinates between those needed for local unit calculations, and those used at a global level for system coordination.

2. Manipulator controller - These FORTRAN subroutines initiate control modes and manipulator functions, including actual movement of the manipulators. These subroutines interface with a library of assembly language routines which perform actual servo command sequence generation. This controller does the computations necessary for point-to-point trajectory movements and for soft limits of movements for equipment protection.

3. End-effector controller - These FORTRAN subroutines activate the end-effector and receive status from the various sensors associated with the end-effector.

This tactical control software resides in the processor of the PUMA manipulator. Other tactical control units would have similar structure, but would necessarily be configured for different functions.

Communications Coordinator. The Communications Coordinator module exists as a FORTRAN subroutine that accepts a structured command string from the LISP communications functions. This command string undergoes minor interpretation and repackaging, and is sent to the appropriate tactical control unit. The communications coordinator is also responsible for receiving status information from the tactical control units and relaying this information to the Lisp controller upon demand. This module originally used a serial line connection between processors, but is being reconfigured to use the mailbox system of the RTNET network. This mailbox system reserves and structures areas of memory for data transfer purposes.

Figure 2 represents a specific concept for the integration of such modules.

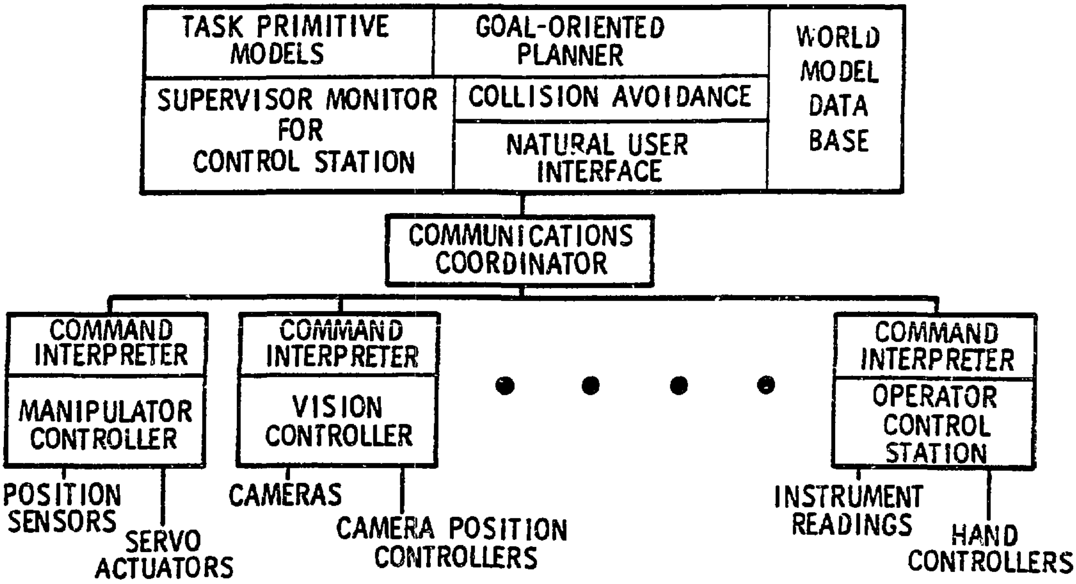

Figure 2.

System Verification

In March 1983, a task was performed with the present implementation of the DAISIE system. A simple blocksworld planner (i.e., routines to plan the stacking and unstacking of simple geometric shapes) and a minimal database structure were embedded in the LISP software. DAISIE successfully performed block-stacking operations using hollow plastic cubes of differing sizes, and was controlled by user terminal input of high-level commands. This task used a single PUMA manipulator arm running tactical control software in its processor. Attached to the manipulator was a surface-adapting vacuum-gripper end-effector implemented from a URI design. The database for this task was a simple property list recording the existence and characteristics of the cubes, of the arm, and of the gripper. Geometrical and positional information of each object was stored. Support characteristics of each object, i.e., what supported the object, what did the object support, were also maintained. The "intelligent planner" was a hierarchically structured set of functions that called tactical routines to perform hardware movements, and that maintained the integrity of the database information. The basic concept of the planner is similar to that of SHRDLU by Winograd[7] but the software has been modified to provide a hierarchical control structure.

This blocksworld implementation demonstrated the success of the basic structure of the control and communication sequences, and proved that it was remarkably simple to interface high-level intelligence algorithms with the DAISIE system.

In April 1983, Richard Wallace, a doctoral candidate in Artificial Intelligence at Carnegie Mellon University, developed a joint-space approach to manipulator collision avoidance. This research dealt with recognizing potential collisions of the manipulator as a total unit, consisting of a series of continuously re-orienting convex polygons. Thus the problem became much more complex: to find a collision-free path for the total arm/end-effector configuration through space, not just a path for the tip of the end-effector. The algorithms involved the generation of a joint-space representation of objects in space, and a search for the optimal path through the remaining free-space. Details of the algorithm are available.[8]

Mr. Wallace spent a week at the Langley Intelligent Systems Research Lab and easily interfaced his procedures with the DAISIE system, thus successfully demonstrating his algorithms on actual hardware. The demonstration consisted of automatically defining paths to avoid known obstacles, and moving the manipulator along those paths. The obstacles were chosen and placed to force reconfiguration of the manipulator joints in order to avoid the obstacle. This demonstrated that the algorithms considered the entire manipulator geometry as opposed to merely the end-effector position. The entire writing of the algorithms, implementation on the DAISIE system, and formal presentation of the project took five working days. The resulting collision avoidance program interfaced to DAISIE with a single command.

This task also used a single PUMA manipulator arm running tactical control software in its processor, as well as some of the planning and database software of the blocksworld experiment.

Future Plans

An integrated system encompassing the remainder of existing lab elements is expected to be fully operational by the end of 1984. The system will include control of both PUMA manipulators, of the vacuum-gripper and the parallel-jaw end effectors, and of vision perception. Supervisory monitoring of the teleoperator control station should also be operational, as should be a flexible user interface within the intelligence module, thus allowing research on man-machine interfacing at different levels of supervisory control. The planner and database will be extended to encompass realistic situations, including remote space and satellite repair environments. The interaction of both manipulators can be included to do necessary research in collision avoidance among multiple manipulators. A graphics capability will also be necessary in order to dynamically monitor the machine's concept of the environment.

As independent processors, each tactical unit has the capability of performing concurrently. Indeed, this is seen as vital to DAISIE's success. Coordination of the communications will be extended to better monitor tactical activities, thus freeing the strategic controller to continue plan generation until intelligent interaction is required by the subsystems. This will allow more distributed processing while also providing high-level supervision when necessary.

A grant proposal has been accepted to construct a suitable database by which DAISIE can store and retrieve knowledge about the world. Dr. Jerry Potter of Kent State University in Ohio has proposed REES, a Remote Environment Expert System, which would consist of three modules:

1. Perception component - This module contains the sensory hardware and the software interfaces required to obtain necessary information about the environment.

2. Database component - This module contains all pertinent information about the environment stored in the necessary degree of detail.

3. Expert component - This module is in effect the logic and arithmetic portion of an expert system which uses the database component, and which serves as an interface between the database and the planner.

Conclusions

Preliminary results indicate that the DAISIE system concept is a successful integration of reasoning, perception, and manipulation components, easily expandable to encompass different algorithms and different hardware. It is anticipated that the DAISIE system will serve as a testbed for development and testing of artificial intelligence techniques as aids to increasing the autonomy of teleoperator/robotic devices. Many pioneering algorithms, such as those dealing with the capacity to learn, to generalize, and to deal with unforeseen circumstances, can then be incorporated into the system, and their applicability to teleoperator/robotic device control can be effectively researched. In addition, as a total system containing modules of perception and manipulation as well as of reasoning, DAISIE could serve as a testbed for techniques in all related areas of technology, and as a vehicle for researching the neglected aspect of system interaction of teleoperator/robotic devices.

References

1. "Remote Orbital Servicing System Concept", by A. J. Meintel, Jr. and R. T. Schappell, presented at the Satellite Services Workshop, NASA Johnson Space Center, June 1982.

2. "NASA Research in Teleoperation and Robotics", by A. J. Meintel, Jr. and R. L. Larsen, presented at the Society of Photo-Optical Instrumentation Engineers Conference, San Diego, California, August 1982.

3. "A Rate Controlled Teleoperator Task with Simulated Transport Delays", by J. E. Pennington, NASA Langley Research Center, TM-85653, September 1983.

4. "Supervisory Control of Remote Manipulators, Vehicles, and Dynamic Processes: Experiments in Command and Display Aiding", by T. B. Sheridan, March 1983, Contract N00014-77-C-0258, Work Unit NR 196-152, ONR.

5. "Theory and Practice of Hierarchical Control", by J. S. Albus, A. J. Barbera, and R. N. Nagel, NBS, 23rd IEEE Computer Society International Conference Conference, September 1981.

6. "General Methods to Enable Robots with Vision to Acquire, Orient, and Transport Workpieces", Seventh Report, Decemer 1981, by J. Birk et al., URI, prepared for NSF, Grant DAR78-27337.

7. LISP, by P. H. Winston and B. K. P. Horn, Addison-Wesley Publishing Company, Inc., 1981, pp. 179-188.

8. "Two-Dimensional Path Planning and Collision Avoidance for Three-Dimensional Robot Manipulators", by R. Wallace, Representation and Processing of Spatial Knowledge, University of Maryland Technical Report TR-1275, May 1983.