

Operobotics

Richard P. Paul, Craig P. Sayers, Julie A. Adams
General Robotics and Active Sensory Perception Laboratory
The University of Pennsylvania
Philadelphia, PA 19104, U.S.A.

A distinguishing feature of teleoperation is that, given time, tasks are generally successfully completed. Robotic systems are now being developed that achieve the same level of reliability by keeping the operator "in the loop" at an appropriate level of interaction during task execution. While the operator may start out at the highest level in specifying a task, he or she may soon find it necessary to interact at much lower levels. The operator moves between levels as appropriate. We might call these robots "operabots" and the system which allows an operator to successfully complete every task that is attempted "operobotics."

This paper introduces the "operobotics" paradigm and provides a brief overview of two specific implementations. In developing these systems the emphasis has moved from correctness to correctability. While the former has proven an elusive goal in robotics, the latter is well within the domain of software engineering.

INTRODUCTION

Two distinct fields, teleoperation and automation, bear on the scope and definition of the field of robotics. Teleoperation [6, 14] has existed since the early 1950's and is representative of manual work. An operator uses tools operated by a master/slave manipulator system which provides high quality bilateral control of the tool. In high quality teleoperation systems the teleoperator provides almost transparent control of the tool. Task feedback is almost always in the form of visual feedback, either direct or by means of video cameras and monitors. In many tasks force feedback greatly facilitates task performance. Task performance by teleoperators closely resembles manual task performance: little planning is done, the task is assessed, the appropriate tools selected and attempts to perform the task are immediately undertaken. Watching a teleoperator perform a task is similar to watching a human perform the same task. Task execution is usually successful even though it might follow a somewhat meandering course. With teleoperation, as with human task execution, there is an expectation of success. Some form of feedback is necessary, to help perform each task step and to confirm to the operator that the task step has been successfully carried out. The reasoning brought to bear on a task by a human operator is, while almost subconscious, vast compared to any available computing systems. Furthermore, it is "common sense" reasoning, an elusive goal of artificial intelligence systems. Sensor

based control is used exclusively, no coordinate systems are imposed on the task nor enter into the control of the task execution.

Automation is the application of external power and organization to perform a task. It is generally based on position control and the necessary control of position tolerances of all components. While sensing is used in automation systems to determine that each sub-task has been successfully performed, sensing is rarely used in the performance of the task since it slows down the rate of operation. Automation systems, while they might perform the same task that an operator once performed, do so in a quite different manner and usually orders of magnitude faster. There is no provision for error recovery, other than detecting that an error occurred and rejecting the part or stopping the machine. The occurrence of errors is an indication that either the machine is incorrect or that the tolerances of the component parts are incorrect. Modifications of one or both take place to eliminate the problem. Automation is frequently associated with the development of considerable infrastructure and extensive study of all factors. Automation, like manual work, is satisfying to watch with the same expectation of success, although in the case of automation many months of debugging may be necessary to reach production rates.

What then of robotics? Initially, seen as an intelligent activity based on sensing, planning, and execution in mimicry of an intelligent human, this initial approach soon ran into trouble as the vast intellectual resources casually brought to bear on a task by a human were not present in a computer. The substitutions, deductions from disparate clues, exploratory actions, rejections of false images, were simply not in the tool box of artificial intelligence systems. While a human might be able to use a teleoperator to perform a task, a computer was simply lost. Based on this initial experience the techniques of automation were applied to robotics. The quality of image processing was to be improved so that quality information could be extracted directly from the image without requiring the elaborate deductive process humans employ. Manipulators were position controlled to allow for precise dispatch to given locations. Execution errors could be detected and, based on the general purpose nature of the equipment, actions taken to correct the errors pro-

grammed. Unfortunately, the range of possible errors was vast, hardly to be tackled by the application of the if-then-statement [13]. Robotic task execution normally ended in the detection of yet another error, which could only be taken care of by extending the program further. By severely limiting the task, as in automation, such robots have found application in industry as a form of very flexible, although low precision, automation in the food, electrical machinery, and semiconductor industries.

OPERABOTICS

What we will call “operabotics” are systems beginning to appear which make use of many of the components of robotics in combination with an operator [4, 8, 7, 12]. The robotic components, or sub-systems: image processing, Cartesian motion control, force control, and path planning are used to perform the “grunt” work for an operator working in a teleoperator mode; the operator provides the missing component of intelligence of the robotics pantheon. Why do this? Teleoperation, so direct, provides no productivity improvement. It requires an operator in constant attendance - every motion made by the teleoperator is made by the operator. While some tasks might totally engage the full attention of an operator many teleoperation tasks are of a repetitive nature and soon lose the attention of an operator; consider the task of dismantling a nuclear reactor which might extend over years. And after years of such teleoperation no trace of the work performed or the skills acquired will have been recorded than if the work had been directly performed manually. Is it possible to provide an operator with some tools to improve his or her productivity and to reduce boredom and consequently reduce errors?

Some tasks are impossible to perform using current teleoperation technology. While teleoperation implies task execution at a distance, that distance is in reality extremely limited. Even in tasks which do not involve any separation of master and slave system, time delay in the control loop can cause instability. If the distances become large, as in outer-space or the velocity of signal propagation becomes low, as in acoustic communication under water, then stability becomes a problem and conventional teleoperation becomes impossible. It is in tackling this problem that teleoperation systems based on task models and robotic technology components have evolved. While it might seem that such systems were limited to the domain of time delayed teleoperation their applicability is much wider and could provide the basis for the necessary productivity improvement in time-delay-free teleoperation.

We will describe two such systems being developed at the University of Pennsylvania, teleprogramming and the MASC system. Teleprogramming is a system devel-

oped to operate a manipulator via delayed communications channels. This is our approach to the classic time-delayed teleoperation application above and is the basis of operabotics. We will then describe the MASC system which is designed to provide a multi-level operator interface to a nominally autonomous system of multiple cooperating mobile robots.

SUBSEA TELEPROGRAMMING

In applying the teleprogramming paradigm [5] to the task of subsea teleoperation our goal was to develop the means to permit an operator ashore to control a manipulator on an unmanned untethered submersible. This required that we overcome two main problems. The first is that the subsea environment is unpredictable and poorly defined. It is far removed from the safety and repeatability of the laboratory. The second difficulty is that available communications schemes are very limited. Using acoustic modems we can expect delays of as much as 10 seconds with bandwidths as low as 10 kbits/second.

When using the current implementation of the teleprogramming system, the operator’s first task is to calibrate the subsea cameras (see figure 1) and then make use of images from those cameras to update the master station model of the remote environment [9]. He or she may then begin to command the remote manipulator. During this time the operator works entirely within a simulation of the remote site. The system observes this interaction and translates his or her actions into symbolic commands for transmission to the real slave robot. Sent at an average rate of only 1 or 2 per second, these commands encode a higher-level representation of the task than is common in teleoperation.

As the operator works the system provides active assistance through the use of *synthetic fixtures* [10]. These provide force and visual clues to aid the operator. By guiding him or her toward making deliberate, unambigu-

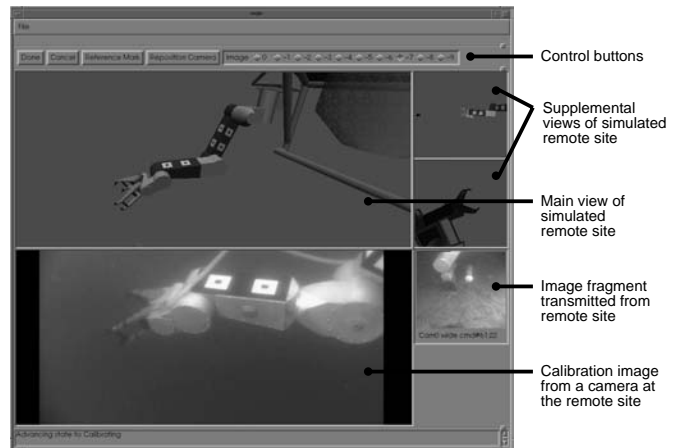


Fig. 1: Teleprogramming operator interface.



Fig. 2: Examples from a sequence of image fragments displayed to the operator during subsea trials [11]. Each image is 192x160 pixels, has been passed through contrast enhancement and JPEG compression/decompression. Note how the system automatically switches cameras while altering windowing and subsampling parameters.

ous, actions they also simplify the task of translating motions into commands.

At the remote site the slave system interprets and executes each command as it is received. However, due to the unpredictability of the environment and the practicality of real world interaction it is unreasonable to expect that every command will execute as planned. While some problems may be avoided by generating commands which are tolerant of remote site uncertainty there will still be cases where execution fails. These situations may be detected by having the slave compare its actual sensory readings with those predicted (and encoded within the generated commands) by the operator station. When such an error occurs the slave pauses and sends an error message stream to the operator station. This error stream contains a preliminary error diagnosis and description of the current slave state along with a detailed record of recent slave motions.

At this point it is the operator's responsibility to analyze the data and determine a strategy for recovery. The system can aid him or her in this task by providing simulations of expected and actual prior slave motion alongside any corresponding real imagery transmitted from the remote site [9] (see Figure 2). The operator may step forward and back through past commands and, if necessary, correct any invalid inferences the system may have made from measured sensory data. For example, the operator may advise the system that it did not grasp an object even though the slave may "think" that it did.

This system is well suited to cases where the environment is unpredictable or where tasks are repeated relatively infrequently. However, if operating in a more structured man-made environment then it is appropriate to consider a greater level of autonomy. Our approach to these situations is realized in the MASC implementation.

MULTIPLE AGENT SUPERVISORY CONTROL SYSTEM (MASC) INTERFACE

The MASC interface has been developed for use with the multiagents project [3]. The purpose of this project has been to investigate the coordination and monitoring of multiagent systems for intelligent material handling. The system is composed of four mobile platforms, each equipped with various sensory and manipulative capabilities, and the MASC interface. The MASC interface

(shown in Figure 3) permits the individual robotic agents and their processes to be controlled by the human supervisor. Our objective is to create a semi-autonomous system which successfully completes assigned tasks in a dynamic environment.

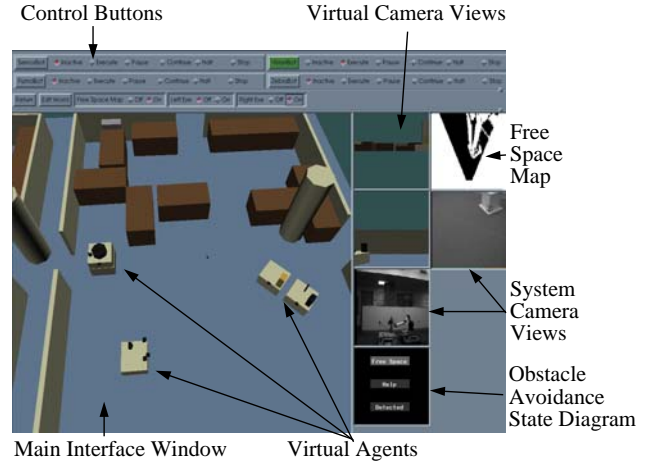
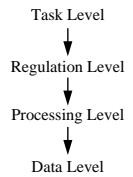


Fig. 3: The MASC system interface.

The human's primary task is to "supervise" the agent's actions during task execution while observing sensory data and images. Each agent is composed of multiple control and processing levels, therefore MASC must permit the supervisor to interact with these levels for the successful semi-autonomous task execution. This interaction will permit the supervisor to revise incorrect agent decisions and reconfigure the system after partial system failures.

A *mediation hierarchy* consisting of four levels (see inset figure) has been formulated [1, 2]. This hierarchy defines the various intervention types which furnish the supervisor with the capability to interact with all the robotic system levels. This interaction occurs when assistance is requested by the agents or when the supervisor detects a situation in which she or he deems intervention necessary.



The highest level in the *mediation hierarchy* is termed the *task level*. This level permits the supervisor to specify the actions an agent, or a group of agents, are to execute to complete an assigned task.

The second level, the *regulation level* is composed of three interaction types: *control interaction*, *request interaction* and *specification interaction*. *Control interaction* provides the supervisor with the capabilities to directly control the individual robotic agent's actions. *Request interaction* permits the supervisor to request the sensory data and processed information from the agent's. *Specification interaction* provides the supervisor with the means to interactively specify information pertinent for a pro-

cess' execution prior to the commencement of processing.

The third level of the *mediation hierarchy* is the *processing level*. This level of intervention permits the supervisor to aid a process when it is unable to arrive at a decision either upon the process' request or as determined by the supervisor through monitoring.

The final level of the *mediation hierarchy* is the *data level*. This level permits the supervisor to ensure correct data is passed up through the system for interpretation and processing. It also allows the supervisor to reconfigure the system during a hardware failure.

The MASC system interface has been designed to encompass the *mediation hierarchy*. It permits the human operator to control a multiple mobile robotic system in a dynamic environment.

CONCLUSIONS

It has been interesting to put together systems which "work." We have developed implementations "from the bottom up"; starting with low-level components and moving toward higher-level representations with the goal of proposing, rather than defining, solutions. The operator has always remained in control of the system. He or she may interact at different levels to diagnose and correct those problems which will inevitably occur.

It has been surprising how effectively operators can compensate for missing or incorrect sensory information; relying on experience and intuition to perform tasks under difficult conditions. Most of our "intelligence" had been in attempting to understand what the operator intended to do so that we could help in the most useful manner. While not robotics, operabotics, makes use of much of what is good in robotics to provide useful, productive, systems.

ACKNOWLEDGEMENTS

The authors wish to gratefully acknowledge the assistance of the Woods Hole Oceanographic Institution, especially Dana Yoerger, Louis Whitcomb and Josko Catipovic.

This work was funded by NSF Grant BCS-89-01352 and in part by Sandia National Laboratories Grant AN-2317, ARPA Grant N00014-92-J-1647, ARO Grant DAAL03-89-C-0031, ARPA/NSF Grant IRI94-12913, and NSF Grants CISE/CDA-88-22719 and STC-SBR8920230.

Any opinions, findings, conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of any funding agency.

REFERENCES

- [1] J. A. Adams and R. Paul. Human-managed, hierarchical control of multiple mobile agents. In *Proceedings of the 33rd IEEE Conference on Decision and Control*, pages 3524 – 3529. IEEE, Dec. 1994.
- [2] J. A. Adams and R. Paul. Human management of a hierarchical control system for multiple mobile robots. In *Proceedings of the 1994 IEEE International Conference on Systems, Man and Cybernetics*, pages 2780 – 2785. IEEE, Oct. 1994.
- [3] J.A. Adams et al. Cooperative material handling by human and robotic agents: Module development and system synthesis. In *IEEE/RJS International Conference on Intelligent Robots and Systems*. IEEE/RJS, Aug. 1995.
- [4] A. K. Bejczy and W. S. Kim. Predictive displays and shared compliance control for time delayed telemanipulation. In *IEEE International Workshop on Intelligent Robots and Systems*, Ibaraki, Japan, July 1990.
- [5] Janez Funda, Thomas S. Lindsay, and Richard P. Paul. Teleprogramming: Toward delay-invariant remote manipulation. *Presence*, 1(1):29–44, Winter 1992.
- [6] Ray C. Goertz. Manipulators used for handling radioactive materials. In E. M. Bennett, editor, *Human Factors in Technology*, chapter 27. McGraw Hill, 1963.
- [7] S. Hirai, T. Sato, and T. Matsui. Intelligent and cooperative control of telerobot tasks. *IEEE Control Systems*, 12(1):51 – 56, Feb. 1992.
- [8] G. Hirzinger, J. Heindl, and K. Landzettel. Predictive and knowledge-based telerobotic control concepts. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1768–1777. IEEE, May 1989.
- [9] Craig Sayers, Angela Lai, and Richard Paul. Visual imagery for subsea teleprogramming. In *IEEE International Conference on Robotics and Automation*, pages 1567–1572, May 1995.
- [10] Craig Sayers and Richard Paul. An operator interface for teleprogramming employing synthetic fixtures. *Presence*, 3(4):309–320, 1994.
- [11] Craig P. Sayers. *Operator Control of Telerobotic Systems for Real World Intervention*. PhD thesis, The University of Pennsylvania, 1995. (in preparation).
- [12] Thomas Sheridan. Space teleoperation through time delay: review and prognosis. *IEEE Trans. Robotics and Automation*, 9(5):592–606, October 1993.
- [13] R. H. Taylor, P. D. Summers, and J. M. Meyer. AML: A Manufacturing Language. *The International Journal of Robotics Research*, 1(3):19–41, 1982.
- [14] Jean Vertut and Philippe Coiffet. *Robot Technology*, volume 3: Teleoperations and Robotics. Prentice-Hall, 1986.