Proceedings of the 1994 IEEE International Conference on Systems, Man, and Cybernetics
San Antonio, TX Oct. 1994

Human Management of a Hierarchical Control System for Multiple Mobile Agents [1]

Julie A. Adams and Richard Paul

General Robotics and Active Sensory Perception Laboratory
Department of Computer and Information Science
The University of Pennsylvania
3401 Walnut Street, Room 301C Philadelphia PA 19104-6228

Abstract

This paper defines a hierarchy of human supervisory mediation levels into a robotic system. This interaction creates a semi-autonomous system which should enable completion of all feasible tasks. We propose that the increased interaction between the supervisor and all levels of a robotic system will permit a human supervisor to maintain the system in a stable state. We have incorporated these hierarchical levels in MASC, a Multiple Agent Supervisory Control system which provides the human supervisor with a three dimensional graphical user interface. Our current test bed includes four mobile robots, of which two are equipped with manipulators.

1 Introduction

In order to take advantage of autonomous robotic systems, and yet to ensure successful completion of all feasible tasks, we propose a mediation hierarchy in which an operator can interact at all levels of a system. Robotic systems are not robust in handling un-modeled events. Reactive behaviors may be able to guide the robot back into a modeled state and to continue. Reasoning systems may simply fail. Once a system has failed it is difficult to re-start the task from the failed state. Rather, the rule base is revised, programs altered, and the task re-tried from the beginning.

Human-machine interfaces have been developed for applications in the areas of nuclear power plants, aviation, and telerobotics [Chr93, HB93, She92], however, these systems are not generally considered autonomous with the operator providing a "supervisory" role.

One aspect of the system we are developing permits the human operator, when necessary, to interact with all levels of a system to correct errors in processes. This interaction encompasses all areas of a semi-autonomous system from the processes which would be considered fully-autonomous to those which would be consider telerobotic.

Our system, MASC - Multiple Agent Supervisory Control system, permits the agents to work autonomously until the human supervisor is requested to take control or a problem is detected by the supervisor. Our design strategy is to develop a general system which is applicable to various robotic systems. We combine the advantages of autonomous systems with the human's ability to control a system through a human-machine interface. MASC provides the supervisor with tools to interact with all processing levels of the robotic system. These interactions may correct corrupted data or process decisions which would typically cause an autonomous system to enter an unstable state. We desire to create a more comprehensive semi-autonomous sys-

[1]This research is funded in part by: ARPA Grants N00014-92-J-1647, DAAH04-93-G-0419; ARO Grants DAAL03-89-C-0031PRI, DAAL03-92-G0153; Gateway Grant 9109794; NASA Grants NGT-50729, NGT-70359; NIH Grant 3R01LM0521703S1; NSF Grants BCS92-16691, BCS92-21796, CISE/CDA-88-22719, CDA91-21973, CDA92-11136, CDA92-22732, GER93-55018, IRI92-10030, IRI92-09880, IRI93-03980, IRI93-07126, MSS91-57156-A 02; University Research Foundation Grant 370892; and The Whitaker Foundation

tem based on this interaction which will successfully complete the execution of task assignments.

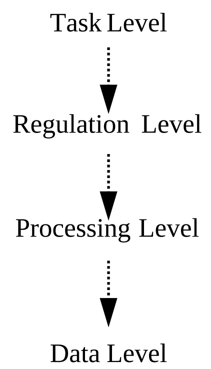

Figure 1: Hierarchical levels of human interaction.

We have defined four hierarchical levels of supervisory interaction with the various levels of a robotic system, see Figure 1. MASC permits the supervisor to specify task assignments, teleoperate agents, display sensory data, override process conclusions and reconfigure the system during sundry sensory and agent failures.

We will briefly describe the MASC system in section two. Section three formally defines the levels of the mediation hierarchy and briefly explains the reasoning which lead to the formulation of this hierarchy. Section four discusses the current and future implementations of the hierarchy.

2 MASC System Description

MASC is a human-machine interface which interacts with any number or type of robotic agents, see Figure 2. The individual robotic agents, their associated manipulators and processes may be controlled by the supervisor through MASC. Our objective is to create a semi-autonomous system which successfully completes its assigned tasks.

The primary task of the human is to “supervise” the actions of the agents during execution [She92]. Through MASC, the human supervises the system, observes sensory data and images. The supervisor is permitted to assist the agents when requested and may assume control of an agent when necessary. Each agent is composed of multiple control and processing levels. In order for the successful semi-autonomous execution of feasible task, MASC must permit the supervisor to interact with these levels. This interaction will permit the supervisor

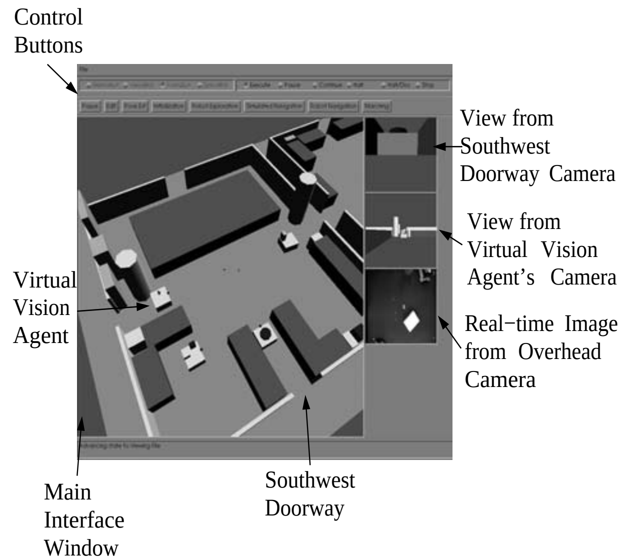

Figure 2: The MASC system interface.

to revise incorrect agent decisions and reconfigure the system after partial system failures. We have organized the supervisor’s interactions with the many system levels into a hierarchy of mediation.

The supervisor creates path plans for the agents employing a local path planner [Wan93] or a global path planner [Bec94]. The supervisor may teleoperate the agents. The supervisor may also display the agents’ sensory information.

Our test bed consist of four mobile agents, each of which are equipped with differing capabilities. The sensor agent possesses ultrasound and infrared sensors, a structured-light sensor and camera, and a stereo camera pair [MM93]. MASC can portray sensory information in a descriptive manner for the supervisor. The vision agent performs visually guided obstacle avoidance and tracking using a stereo camera pair and a single camera on a turn table. MASC displays the free space map created by the visually guided obstacle avoidance. This free space map [KB93] may be overlaid onto the virtual model. Two manipulator agents complete the system. The Puma agent is equipped with a Puma 260 manipulator and the Zebra agent is equipped with a Zebra manipulator. We intend to provide the supervisor with the ability to control the manipulators and monitor the coordination of locomotion and manipulation [YY92].

The supervisor communicates with the agents through MASC. We have provided display push buttons, termed control buttons see Figure 2, which allow the supervisor to specify system information. The top set of con-

Published: 1994 IEEE International Conference on Systems, Man and Cybernetics

trol buttons permit the supervisor to define which agent s/he wishes to control at a specific instance. There are two sets of buttons on this level. The left hand set lists the agents which are active in the system. The right hand set specifies the current execution state of the agent chosen in the left hand set. The options include pausing a current action, continuing a paused action, halting an agent and removing all further commands from its command array and an emergency stop. The bottom set of command buttons permit the supervisor to alter the current system state and execute various processes. For instance, MASC is equipped with initialization, exploration and navigational states. In the exploration state, the supervisor may teleoperate an agent using the mouse. The agent autonomously executes commands while in the navigational state.

The supervisor may request any agent's sensory data while in any system state. The agents transmit odometry and heading readings, sensory readings, raw image data, and processed data. This information is used by MASC to create various system displays. The current positions of the respective virtual agents are maintained with the odometry and heading readings. Image data may be displayed in windows to the right of the main interface window, see Figure 2 and some images may be overlaid onto the virtual environment model. Processed data, such as the free space map originating from the visually guided obstacle avoidance process, can be displayed in a window or overlaid onto the virtual model.

MASC combines autonomous and telerobotic control. While an agent proceeds autonomously it may petition assistance from the human. This assistance may be a request for information about the environment. MASC determines if it possess the requested data and transmits it accordingly. The agent may require a decision from the supervisor before it can continue execution. When this request is made, a request window appears on the supervisor's screen. The supervisor must acknowledge the request and then furnish the proper information.

3 Definition of Mediation Hierarchy

We have formulated a mediation hierarchy consisting of four levels. These levels define the various types of intervention into the differing levels of a robotic system. The hierarchy furnishes the supervisor with the capabilities to interact with all such levels. This interaction should permit the supervisor to correct situations which would cause a fully autonomous system to become unstable and possibly fail in its task execution. It is important to note the supervisor only interacts with the agents when assistance is requested or when the supervisor detects a situation where s/he deems it is necessary to intervene on behalf of an agent. We purpose that this hierarchy will create a semi-autonomous system in which we should be able to execute all feasible tasks to completion.

3.1 Task Level

There are numerous tasks which one would purpose to assign to a robotic system. One manner by which to break up a task and assign the proper set of actions to each agent is to "hard code" the tasks and actions into the system. Unfortunately, this approach inevitably limits the number of tasks the system can execute and does not create a general system. In order to create a general system which executes various tasks, the supervisor, or a task planner, must derive the proper assignments. Since the system will not execute a task until these actions are taken, the *task level* resides atop our mediation hierarchy, see Figure 1.

The *task level* will permit the supervisor to specify the actions an agent, or a group of agents, are to execute to complete an assigned task. Tasks may include exploration of the environment to assist with model building, following an assigned path to a goal, observing the execution of a task assigned to another agent, marching in a configuration, carrying items such as pallets, and the navigation necessary to transport items from one location to another.

Since humans are better equipped for such tasks we currently assign this undertaking to the supervisor, however, we plan to incorporate a planner to assist the supervisor.

3.2 Regulation Level

There exist minimal interactions which are necessary between a human-machine interface and a robotic system. If an agent is on the verge of colliding either with another agent or an obstacle, the supervisor should be able to prevent such a collision. If it is necessary for one agent to complete a task before another agent begins its task, the second agent may need to be informed to wait for the first agent to complete its execution. The supervisor possess a means of monitoring an agent's actions. This may include video images, displays of sensory data or positional updates. It is essential that the interface provide a means for the supervisor to choose such information for monitoring purposes. Also, in such a system, the agent's processes may require information from the supervisor in order to begin processing. The interface must facilitate the means of providing this information. The *regulation level* couples these interactions into one mediation level. We have developed three types of interaction on this level, *control interaction*, *request interaction* and *specification interaction*, which we define below.

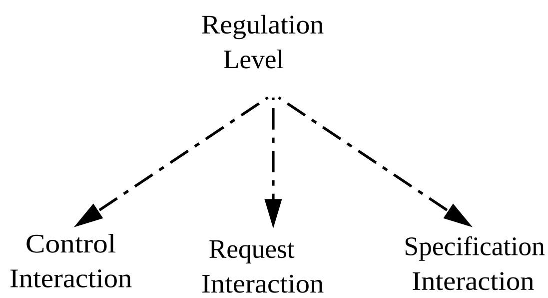

Figure 3: Interactions on the *regulation level*.

3.2.1 Control Interaction

MASC provides the supervisor with the capabilities to cope with situations when an impending collision must be avoided, or if one agent should be instructed to wait for another agent to complete its task assignment through *control interaction*. Included in this interaction are situations when an agent finds itself unable to find a means with which to execute its set of assigned actions. The agent would then alert the supervisor to its need of assistance. This assistance may require the supervisor to teleoperate the agent to a new position before the agent can continue the task execution autonomously. Formally, *control interaction* provides the supervisor with the ability to control the progress of the agent while executing a task either for the purpose of deterring progress or assisting progress.

3.2.2 Request Interactions

Systems contain varying information which may be of use to the supervisor at different times throughout the system execution which are not necessary during the entire system execution. The objective is to avoid overloading the supervisor with too much information[She92, Wha92]. *Request interaction* permits the supervisor to request the sensory data and processed information from the agent's only when it is needed for error detection or monitoring purposes. Once the supervisor has completed the actions for which the information was requested, s/he can inform the agent's processes to cease transmission.

Formally, the *request interaction* permits the supervisor to request information directly related to the current task. This information is then used by the supervisor to monitor the execution of a task. If the supervisor believes a process is making incorrect decisions, s/he may request the raw data which is being processed. This information may include images, ultrasound sensors or vehicle position information. The supervisor then reviews this information and draws conclusions as to why the process behaves as it is.

3.2.3 Specification Interaction

Various processes require information from the supervisor before they can begin processing. Such a process may be a path planning process for which the supervisor must specify the starting, intermediary and goal points of the path. The process will then use this information and return a path for the supervisor to approve. The *specification interaction* permits this form of interaction between a process and the supervisor. Formally, *specification interaction* provides the supervisor with the means to interactively specify information pertinent for a process' execution.

3.3 Processing Level

One can think of situations where a process my be incapable of reaching a correct decision based on ambiguous information and must therefore request assistance from the supervisor. There are also situations where a process will formulate a correct decision in a local context but the decision will not be correct in the global scheme, therefore the supervisor should override the decision formulated by the process.

While observing an agent's actions based on a particular process the supervisor may determine the process is formulating its calculations based upon an incorrect interpretation. The supervisor may then intervene in the process to clarify the information, override a decision or allow it to continue with its processing. Assume an agent is employing visually guided obstacle avoidance and another agent momentarily passes within its viewing field, in this case the visual agent would interpret it as an obstacle and begin the avoidance process. The *processing level* permits the supervisor to override the decision to avoid the "obstacle" and instruct the agent to continue with its original assignment. Formally, the *processing level* permits the supervisor to aid a process when it is unable arrive at a decision and to rectify incorrect decisions deduced by a process. This level of interaction will protect the agents from entering unstable states.

3.4 Data Level

It is known that mechanical devices fail from time to time, and that the automatic reconfigurations for such failures are not always successful, therefore, the supervisor should be provided with the means to reconfigure the system. The *data level* of the mediation hierarchy will permit the supervisor to reconfigure the system when an automatic reconfiguration has failed.

The outcomes determined by the higher level processes are dependent upon correct input data. If this data is not correct, the processes will likely formulate incorrect decisions and commands which may force the agent into an unstable state. The *data level* will also supply the supervisor with the ability to ensure processes receive correct data for their interpretations. For instance, as mobile agents move throughout the environment executing an assigned task, they accumulate errors in their positional and heading readings. If the automatic reset fails, it may become necessary for the supervisor to reset the readings. Alternatively, assume the focus of a camera from a camera pair has been corrupted, this may hinder the retrieval of information in the process using the images. The supervisor should be able to inform the process to stop processing and instruct the agent to rely on another form of sensing to complete its task assignment.

Formally, the *data level* permits the supervisor to ensure correct data is passed up through the system for interpretations and processing. It also allows the supervisor to reconfigure the system during a hardware failure. This type of interaction implies that as data flows upward through the system, the system will correctly interpret the data implying correct actions will be executed which in turn imply the successful execution of task assignments.

4 Implementation and Future Work

MASC currently implements the above described *control*, *request* and *specification interactions* of the *regulation level* and provides the supervisor with the abilities to intervene and override the decisions of the system processes on the *processing level* of the mediation hierarchy. The supervisor may teleoperate the agents as well as control their actions by instructing them to pause, halt or continue further actions or to stop immediately. The supervisor may request information and may also specify the necessary information for all processes requiring such in our system.

The supervisor has not yet been incorporated into all processes of our system on the *processing level*. We plan to continue working with the *processing level* interactions and will provide the supervisor with more authority within these processes to override process decisions. Once we have sufficiently developed the *processing level* we will begin implementation of the *data level*. We will incorporate a planner to assist the supervisor with task decomposition and action set assignments to the proper agents.

Once all levels of interaction have been implemented we will test MASC on various tasks, such as navigational and follow the leader behaviors, detecting an object, and the picking up and transporting of objects. These experiments will either verify or negate our hypothesis that providing a supervisor with such interactive capabilities through the mediation hierarchy permits a semi-autonomous system to successfully complete assigned tasks.

5 Conclusions

The mediation hierarchy will provide traditional autonomous systems an added capability. The hierarchy will supply the supervisor control of the overall system, directing all levels of the system to ensure successful task completion. The hierarchy permits the agents of the system to work autonomously until they request assistance from the supervisor or the supervisor feels the situation warrants direct intervention. Once the human completes the necessary interactions, the agents may be instructed to proceed autonomously.

We have described the MASC system and our test bed. We have also defined the levels of intervention in the mediation hierarchy. We believe that when completed this mediation hierarchy will create a semi-autonomous robotic system which will permit the agents to successfully complete all feasible tasks.

6 REFERENCES

[Bec94] Craig Becker. *Internet Path Planner Server Protocol*. Stanford University, January 1994.

[Chr93] Brian Christensen. Virtual environments for telerobotic shared control. In *Proceedings SPIE - The International Society for Optical Engineering*, pages 74 – 83. SPIE, 1993. v. 2057.

[HB93] T. Hancke and R.J. Braune. Human-centered design of human-machine systems and examples from air transport. In *Proceedings of the 12th World Congress, International Federation of Automatic Control*, pages 343 – 346, 1993. v. 7.

[KB93] Jana Kosecka and Ruzena Bajcsy. Cooperation of visually guided behaviors. In *Proceedings of the IEEE 4th International Conference on Computer Vision*, pages 502 – 506, 1993.

[MM93] Robert Mandelbaum and Max Mintz. Active sensor fusion for mobile robot exploration and navigation. In *Proceedings SPIE - The International Society for Optical Engineering*, pages 120 – 141. SPIE, 1993. v. 2059 Sensor Fusion VI.

[She92] Thomas Sheridan. *Telerobotics, Automation, and Human Supervisory Control.* MIT Press, Cambridge, Mass., 1992.

[Wan93] Chau-Chang Wang. Local path planner. University of Pennsylvania. Personal Communication, Sept. 1993.

[Wha92] Susan Whalley. The human computer interface - designing for process control. In *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, pages 827 – 832. IEEE, 1992.

[YY92] Yoshio Yamamoto and Xiaoping Yun. Coordinating locomotion and manipulation of a mobile manipulator. In *Proceedings of the 31st Conference on Decision and Control*, pages 2643 – 2648. IEEE, December 1992.