INTRODUCTION

The Robotic Aid now under development at Stanford University [1] will fulfill its potential as an aid for the physically disabled when two things come about: (a) users can speak to the device in ordinary English, and (b) users can teach the device to perform tasks of their choosing. A robot that can understand ordinary English will be accessible to those many people who are not trained in computer programming. And with a robot that can be taught new tasks, the user will be able to adapt the device to his or her individual abilities and needs.

This paper describes recent progress we have made towards these two goals. The work builds on our earlier experience in providing a natural-language interface to the mobile base of the Robotic Aid [2,3,4].

ROBOT DESIGN AND OPERATION

The Robotic Aid consists of a manipulator (a Unimation PUMA 260) and simple gripper mounted on an omnidirectional vehicle. In the current phase of our work we are looking at the robot’s use in an office-like environment, a room containing a table and bookshelf on which there are books of various sizes and shapes. We want the user to be able to teach the robot to perform organizational tasks such as sorting the books, packing them into or out of the bookshelf, and rearranging the order of the books on the shelves.

The robot starts with several basic motor Actions: it can open and close its gripper; it can move its gripper from one point to another; it can pick up a single book; it can put a book down on the table or place it upright on a shelf; and it can push a book along the surface of the table.¹

The robot can also perform certain Tests: it can tell whether the gripper is closed or open; it can check whether or not its gripper is at a specified location; it can tell whether something is in its gripper or not; and it can tell whether or not it is moving its gripper or arm. Tests return the value true or the value false when executed.

There is no vision system on the robot at present and so we equip the robot with a 3-dimensional map of its environment showing the initial location and orientation of all the objects in the room. Other relevant perceptual properties such as size and color are also made known to the robot. This store of information is called the robot’s Database. The robot updates the Database whenever it moves a book from one location to another. Access Functions allow the robot to get information from the Database. There are three basic types of Access Function: one returns information on all objects of a given type (all books, for instances); another returns information on all objects with a given property value (all books that are red in color, for instance, or all books that are thinner than a given size); and the third returns information on all objects that stand in a given spatial relation to some other objects (everything that is on the top shelf, for instance, or everything that is next to the thin book, or everything that is between the big book and the small book).

When an English command is issued to the robot, it is accepted by a program, called the Interpreter, that produces an Interpreted Command. The Interpreted Command specifies three things:

1. The Database Access Functions that allow the robot to determine the location of any object or area referred to in the English command (the book on your left, the red book, etc.)

¹Some of these motor functions are hard for any robot to do well and will not work perfectly in all circumstances (the “pick-up” action, for instance, will not initially be successful if the book cannot easily be grasped or is at the bottom of a pile, and a heavy book may slip in the robot’s grasp and have to be put down and grasped again more firmly). The user will eventually be able to give corrective commands to modify the robot’s behavior in such circumstances.
next to the thin red book, to the left of the big book, on the top shelf, for instance).

2. The basic Actions and Tests that the robot must perform to satisfy the user's request. So, for instance, the command Move the red book from the table to the top shelf specifies that the red book is to be picked up if it is not already in the robot's gripper, that the arm is to be moved to an empty spot on the top shelf, and that the book is to be put down in that spot.

3. The temporal and logical order in which the Access Functions and robot Actions and Tests must be performed. There are seven different control structures for specifying the order. 

$$\text{(SEQ } S_1 \ldots S_n \text{)}$$ designates the sequential execution of $S_1$ to $S_n$. 

$$\text{(PAR } S_1 \ldots S_n \text{)}$$ designates the parallel or simultaneous execution of $S_1$ to $S_n$. 

$$\text{(IF } x \text{ then } S_1 \text{ ELSE } S_2 \text{)}$$ designates the execution of $S_1$ if the Test $x$ returns true; otherwise if $x$ returns false, $S_2$ is executed. The remaining four control structures are described in [3]. The command Put the red book on the shelf will produce the following:

$$\text{(SEQ find-location-of-red-book}$$

$$\text{ (PAR (IF red-book-not-in-gripper}$$

$$\text{ pick-up-red-book)}$$

$$\text{ find-free-space-on-shelf)}$$

$$\text{ put-book-down-in-free-space)}$$

The Interpreter consists of a parser and a sentence-level grammar. The parser applies the rules of the grammar to the user's command to produce the Interpreted Command. The parser is a CommonLisp version of the D-PATR parser [5] that we have extended to allow interaction with the user and with the robot's Database. If the command is unclear in any way—if, for example, the user says to the robot Pick up the big book and there are several that qualify as big books—the user will be asked for more information. The D-PATR formalism allows both the structural properties of sentences to be described (their syntax) and their informational content (their meaning or how they should be interpreted). It is the informational content of the sentence that is represented in the Interpreted Command.

In addition to the Interpreter, there is also a Dialogue Manager, a program that accepts commands from the user, monitors the robot's progress as it obeys the commands, and tells the robot what it must remember so that it can learn from the user's instruction. For instance, if the user tells the robot how to arrange the books in piles according to their size, the robot must remember what it was told and how it did it so that it can later repeat the same behavior when asked. In this way, the user can build up a set of learned Actions that will be added to the robot's basic Actions of "pick-up," "put-down," "push," and so on.

**IMPLEMENTATION**

The computer onboard the robot is a Digital Equipment Corporation LSI 11/73. The basic robot Actions and Tests are programmed in MicroPower PASCAL (a dialect of PASCAL that explicitly supports multi-tasking) and are executed at a fixed rate of 15 Hz to guarantee smooth motion of the robot. When an Interpreted Command is produced by the Interpreter, it is acquired by a program called the Scheduler. The Scheduler invokes robot Actions and Tests as specified by the Interpreted Command by sending a series of coded commands to the robot over an RS232 cable connection. When one of the Tests returns a value, the Scheduler is activated. It examines the Interpreted Command to determine what to do next—usually to invoke or terminate the execution of some other Action and/or Test. The Interpreter, Scheduler, and Database are all implemented in CommonLisp on a Symbolics 3650.

**ACKNOWLEDGMENTS**

This work was funded by the United States Veterans Administration through the Rehabilitation Research and Development Center of the VA Medical Center in Palo Alto. The authors acknowledge the use of facilities in the Center for Design Research and the Center for the Study of Language and Information at Stanford University.

**REFERENCES**


Address: Institute for Mathematical Studies in the Social Sciences (IMSSS), Stanford University, CA 94305, USA.
Robots controlled by computers are replacing jobs that involve packing or moving of goods. An example is the Amazon robots that move all products to people who package and mail the products to customers. Below is a video of these robots at work. These robots help Amazon and its employees get packages shipped out faster than their competition. The results of a recent experiment to slow the effects of the aging process in mice amazed scientists in Boston, USA. The scientists increased the amount of an enzyme called telomerase in the cells of the mice. Telomerase is an important enzyme because it repairs DNA. With increased telomerase in their cells, the mice’s fertility improved, their fur began to look healthier, even their brains worked better. Of course, if scientists ever do succeed in developing drugs that combat the aging process we will need to ask ourselves whether it is right to use them. For instance, should we keep people young and healthy artificially when, already, there are far too many people on the planet? What if you can’t wait for these future developments though? A robot is defined as a mechanism that can understand its surrounding environment, make decisions and move automatically. The little spider-bots can certainly do that. They can walk, turn left and right and even create their own products. One day, such microscopic devices could actually be used to build tiny computer chips or to detect and treat diseases such as cancer at a molecular level. It is interesting to note that the nano-spiders are made of DNA molecules. The spider’s body is composed of a common protein called streptavidin. Unlike a real spider, however, the nano-spider has only got Using English to instruct a robotic aid: An experiment in an office-like environment. In Proc. of the International Conference of the Association for the Advancement of Rehabilitation Technology (ICAART-88), pages 466–467, Montreal, Canada, 1988. Google Scholar. A potential application in early education and a possible role for a vision system in a workstation based robotic aid for physically disabled persons. In R Foulds, editor, Interactive Robotic Aids-One Option for Independent Living: An International Perspective, Monograph 37, pages 18–23. World Rehabilitation Fund, 1986. Google Scholar. 17.