In Proceedings of the Sixth International Conference on Cognitive Modeling, 337-338.
Mahwah, NJ: Lawrence Earlbaum.

ADAPT: A Cognitive Architecture for Robotics

D. Paul Benjamin (benjamin@pace.edu)
Computer Science Department, Pace University
1 Pace Plaza, New York, NY 10038 USA

Damian Lyons (dlyons@fordham.edu)
Computer Science Department, Fordham University
340 JMH Fordham University, 441 E. Fordham Rd., Bronx, NY 10458

Deryle Lonsdale (lonz@byu.edu)
Linguistics Department, Brigham Young University, Provo, UT 84602

Introduction

ADAPT (Adaptive Dynamics and Active Perception for Thought) is a cognitive architecture specifically designed for robotics researchers. ADAPT is in the initial state of development. It manipulates a network of schemas that implement a formal model of concurrent, distributed real-time control. Perception is active, which means that all perceptual processing is goal-directed and context-sensitive, even down to the raw sensory data.

The development of ADAPT is motivated by robotics researchers who want their robots to exhibit sophisticated behaviors including use of natural language, speech recognition, visual understanding, problem solving and learning, in complex analog environments and in real time. Many robotics researchers have realized that programming robots one task at a time is not likely to lead to a robot with such general capabilities, so interest has turned to cognitive robotic architectures to try to achieve this goal. ADAPT is a collaborative effort combining Pace University's Robotics Lab – csis.pace.edu/robotlab, Fordham University's Robot Lab – www.cis.fordham.edu/~lyons/rcvlab and the linguistics department of Brigham Young University – linguistics.byu.edu/nlsoar.

Robotics researchers are faced with a major obstacle when attempting to use cognitive architectures to control robots: the architectures do not easily support certain paradigms of perception and control that are mainstream in robotics. This paper focuses on two such paradigms: adaptive dynamics and active perception.

One common assumption that is made by many robotics researchers is that perception is an *active* process, rather than a passive one. This means that perception is goal-directed and context-sensitive at every stage, including the initial processing of input sensory data. Active perception, and in particular active vision (Blake & Yuille, 1992), is a major research area in robotics. For example, if a robot is crossing the street, it will not process all of its visual frame equally, but will filter the data for large blobs that are moving towards the robot, processing in a very coarse way the rest of the visual field. Such a robot will perceive cars very differently depending on the goal and the situation. Existing cognitive architectures can be altered to perceive in this way (Byrne, 2001) but this is the exception rather than the rule. EPIC in particular does not perceive this way, and this is a major limitation in its application to robotics.

A second assumption made by many robotics researchers is that a robot's actions must be modeled by concurrent, distributed, real-time processes. A robot typically possesses a large number of moving components, e.g. gripper joints, wheels or legs, pan-tilt-zoom camera units, microphones, sonars, etc. and these components may all be active simultaneously. In addition, the robot must model processes occurring simultaneously in the environment.

As a result, many robot programs are the composition of concurrent, communicating real-time behaviors that are not typically organized in a hierarchy. Cognitive architectures such as Soar (Rosenbloom, Laird, & Newell, 1993), EPIC (Kieras, Wood, & Meyer, 1997) and ACT-R (Anderson, 1996) are relatively weak in this area. These models focus primarily on sequential search and selection, and their learning mechanisms focus on composing sequential rather than concurrent actions. Furthermore, these architectures tend to model actions hierarchically (with a stack), not in a truly distributed fashion. It is not that these architectures forbid the use or learning of concurrent distributed programs, but rather that this is not a central feature of them, and it is of primary importance in robotics.

Our Approach to Perception and Planning

ADAPT resembles existing architectures in many ways. It is a production-based architecture with a working memory, and matches productions against working memory during each cycle in the same manner as these architectures. All the matching productions fire, as in Soar, and place their results in working memory. ADAPT possesses a long-term declarative memory, as ACT-R does, in which it stores sensory-motor schemas that control its perception and action. All perceptual processors fire in parallel, as in EPIC, but place their low-level sensory data into working memory, where it is processed by the cognitive mechanism.

We draw a distinction between the goals that are task goals, e.g. "find the blue block", and those that are goals of the architecture, e.g. "start the schema that scans the environment for a segment of a given color". Similarly, we distinguish between task actions, e.g. "pick up the block", and architectural actions, e.g. "start the gripper-closing schema". Our goal is to reason about concurrent goals and actions *in the task*, so we partition the actions and goals into an architectural part and a task part. We restrict the architectural part to one active goal (it has a goal stack) and one architectural action at a time. The architectural part is procedurally represented, i.e. the system can execute the actions but cannot examine their internal representation. We represent the task goals and actions declaratively in working memory as well as procedurally. There can be as many active task goals and actions in working memory as the system wants. We call these goals and actions "schemas".

An Implementation of ADAPT

An initial implementation of ADAPT has been completed. Testing has just begun using a Pioneer P2 robot that is equipped with stereo color vision, microphone and speakers, sonars and touch sensors. The implementation is in Soar, because of the similarity between ADAPT and Soar. A declarative memory has been added to Soar for the general schemas, together with operators that instantiate a general schema, transform it into a set of chunks, and control its execution. Schemas are executed by a runtime system that implements the RS language (Lyons & Arbib, 1989) in the Colbert language, which is a behavior-based language provided with Pioneer robots.

Currently, ADAPT has access only to sonar and bump-sensor readings. This permits simple navigation and pushing behaviors, but nothing more. The existing version of ADAPT has a cycle time of 50ms and is very successful at guiding the robot at basic navigation tasks such as moving from one room to another and avoiding obstacles.

The language component will be provided by NL-Soar (Lonsdale, 2001), which is being ported to Soar8 as part of our collaboration with Brigham Young University. It will be available in July and will be integrated into the system.

The vision component consists of two pieces: a bottom-up component that is always on and is goal-independent, and a top-down active component. Both components exist but the full integration within ADAPT is not complete. We anticipate a full integration by September.

Testing and Evaluating ADAPT

We have selected an important and flexible class of mobile robot applications as our example domain: the "shepherding" class. In this application, one or more mobile robots have the objective of moving one or more objects so that they are grouped according to a given constraint.

This class of tasks is attractive for two reasons. The first is that it includes the full range of problems for the robot, from abstract task planning to real-time scheduling of motions, and including perception, navigation and grasping of objects. This is ideal, because it creates a situation in which complex hierarchies of features and constraints arise.

The second reason is that we can embed lots of other problems in it, especially those that have been examined by cognitive psychology. For example, we can create an isomorph of the Towers of Hanoi task by having three narrow enclosures, with three boxes in the leftmost enclosure. The robot must move them to the rightmost enclosure one at a time. We add the constraint that no object can be in front of a shorter object (so that all objects are always visible by the observer). The three enclosures act as the three pegs in the Towers of Hanoi, and the constraint is isomorphic to the constraint that no disk can be on a smaller disk in the Towers of Hanoi. In this way, the robot can be presented with a Towers of Hanoi problem in a robotic setting with perceptual and navigational difficulties, rather than just as an abstract task. This permits us to evaluate the robot's problem-solving and learning capabilities in a way that may permit comparison with human data.

Summary

Cognitive psychology and robotics have much to offer each other. The development of robot cognitive architectures is an attempt to apply the results of cognitive modeling to the difficult problems faced by robotics research. ADAPT is a cognitive architecture specifically designed to permit robotics researchers to utilize well-known robotics research in areas such as active perception and adaptive dynamics within a cognitive framework. This architecture is still in its infancy, and in particular has not yet been integrated with vision or language. The goals of this research are to expand the capabilities of robots and simultaneously to expand and generalize the capabilities of existing cognitive models.

Acknowledgements

We would like to thank Frank Ritter for many helpful comments and pointers to relevant work.

References

Anderson, J. R. (1996). ACT: A simple theory of complex cognition. *American Psychologist* , 51, 355-365.

Blake, A., and Yuille, A., (Eds,) (1992). *Active Vision* , MIT Press, Cambridge, MA.

Kieras, D.E., Wood, S.D., and Meyer, D.E. (1997). Predictive engineering models based on the EPIC architecture for a multimodal high-performance human-computer interaction task, *ACM Transactions on Computer-Human Interaction* 4, 230-275.

Lonsdale, Deryle, (2001). An Operator-based Integration of Comprehension and Production, *LACUS Forum XXVII*, pages 123–132, Linguistic Association of Canada and the United States.

Lyons, D.M. and Arbib, M.A., (1989). A Formal Model of Computation for Sensory-based Robotics, *IEEE Transactions on Robotics and Automation* 5(3), June.

P. S. Rosenbloom, J. E. Laird, and A. Newell, (1993). *The SOAR Papers*. MIT Press.