Information Sciences

Algorithm Would Enable Robots to Solve Problems Creatively

A control architecture is based on hypotheses concerning natural intelligence.

Lyndon B. Johnson Space Center, Houston, Texas

A control architecture and algorithms to implement the architecture have been conceived to enable a robot to learn from its experiences and to combine knowledge gained from prior experiences in such a way as to be able to solve new problems. The architecture is an abstraction of an interacting system of relatively simple components that, when properly interconnected, should enable the spontaneous emergence of behaviors from the complete system that would not necessarily be expected from the individual components. These emergent behaviors should enable a robot to interact robustly and intelligently with a complex, dynamic environment.

Departing from the customary approach to artificial intelligence, the theoretical foundation of this architecture includes a hypothesis that has gained some acceptance in research on natural intelligence in animals. Stated succinctly, the hypothesis is that intelligence requires embodiment, situatedness, development, and interaction. Applied to a computer, the hypothesis states that to be able to develop intelligence, a computer must be able to, among other things, manipulate the environment and sense the results; in other words, the computer must be equipped with a robotic interface with the physical world. A related hypothesis is that sensory-motor coordination forms the basis for intelligent behavior, and, hence, intelligence. This follows from the demonstrated ability of sensory-motor coordination information to self-organize into categories.

The architecture represents a set of parallel distributed computational modules (software objects) that communicate with each other through message passing. The principal objects loosely correspond to the (presumed) common computational modules of mammalian brains. A set of sensory processing modules continually updates a spatio-temporally indexed short-term memory structure denoted the sensory ego-sphere (SES). Depending on the task context, an attentional mechanism determines the saliency of incoming sensory information. In response to changes in the state of motion of the robot, the time series of sensory information is partitioned into episodes. Episodes are encoded in vector form and stored in a database, denoted the database associative memory (DBAM), that comprises the long-term memory of the robot.

The DBAM has the implicit mathematical structure of a linear vector space that is constructed as the direct sum of three subspaces. One subspace encodes a change in motor state (a motor event). Another subspace encodes the state of the sensors immediately prior to the motor event — in other words, the sensory preconditions of the motor event. The third subspace encodes the sensory state subsequent to the motor event (sensory post-conditions). The vectors in the DBAM and the external events from which they are formed are called sensory-motor coordination (SMC) events.

Several research projects have demonstrated that over time, as a robot senses and acts, clusters form within a vector space of SMC descriptors. Categories emerge. Thus, it becomes possible to select an exemplar that describes an equivalence class of SMC events. The encoding of SMC events as motor events with sensory preconditions and sensory post-conditions makes it possible to use simple vector-space distances as measures of dissimilarity or similarity of complete SMC events. Vector-space distances can also serve as measures of dissimilarity or similarity between the sensory post-conditions of one event and the sensory preconditions of another.

The architecture utilizes a control strategy, called topological action mapping, that appears to be employed by mammals to achieve specific goals. In this architecture, action maps are implemented by a spreading activation network (SAN). (A SAN is one of a particular class of graph traversal algorithms that selects optimal paths.) For each goal to be reached by the robot, there corresponds an SMC event. Moreover, the current state of the robot is indicated by an SMC event. Given the goal event, the SAN searches through the DBAM, matching its sensory preconditions to the sensory post-conditions of other events. The search is constrained by a number of global state variables and previously formed linkages. Upon finding a sufficiently close match, the SAN iterates the search until the current state is reached. The current motor state (hence, the current behavior) is maintained until the sensory post-conditions trigger a change of state. Thus, the robot proceeds through a sequence of behaviors until the goal is reached.

One key aspect of this architecture is that the robot cannot be programmed through conventional means: it must be taught to perform tasks, either through direct control or by example. The robot must practice each task several times to enable the formation of clusters of SMC events and to learn the sequences of these events. Such repetition enables the robot to perform the task later, without supervision. However, this does not, in itself, enable creative problem solving.

To be able to solve problems creatively, the robot must engage in a process analogous to dreaming. After intervals of continuous motor activity, the robot performs computations in which it plays back data on recently performed sequences of tasks. SMC events are compared quasi-randomly to other events in the DBAM. If the sensory post-conditions of a given event are sufficiently similar to the sensory precondition of another event, links are formed between them. Such a link is formed, even if the two behaviors have never occurred in sequence before. This link gives the robot a new possible behavior transition that it could make, given the appropriate sensory trigger. Thus, the robot learns to anticipate possible sequences of events and learns strategies for the solving of problems that it has not encountered but that could occur.

The architecture is still undergoing development. An SES has been implemented in the NASA Robonaut (a developmental anthropomorphic robot intended to serve as an astronaut's assistant). Production-quality software to implement the architecture has not yet been written. A major problem in writing this software is that of efficient representation of sensory data in vector spaces.

This work was done by Alan Peters of Vanderbilt University for Johnson Space Center.

In accordance with Public Law 96-517, the contractor has elected to retain title to this invention. Inquiries concerning rights for its commercial use should be addressed to:

Christopher D. McKinney, Director
Office of Technology Transfer
and Enterprise Development
Vanderbilt University
1207 17th Avenue South, Suite 105
Nashville, TN 37212
Phone: (615) 343-2430
E-mail: chris.mckinney@vanderbilt.edu

Refer to MSC-23489, volume and number of this NASA Tech Briefs *issue, and the page number.*

Hypothetical Scenario Generator for Fault-Tolerant Diagnosis

This is a means of performing diagnostic reasoning when data are missing.

NASA's Jet Propulsion Laboratory, Pasadena, California

The Hypothetical Scenario Generator for Fault-tolerant Diagnostics (HSG) is an algorithm being developed in conjunction with other components of artificial-intelligence systems for automated diagnosis and prognosis of faults in spacecraft, aircraft, and other complex engineering systems. By incorporating prognostic capabilities along with advanced diagnostic capabilities, these developments hold promise to increase the safety and affordability of the affected engineering systems by making it possible to obtain timely and accurate information on the statuses of the systems and predicting impending failures well in advance.

Prognosis is tightly coupled with diagnosis. The simplest approach to prognosis by an artificial-intelligence system involves the use of a diagnostic engine in a controlled feedback loop to project from the current state of the affected engineering system to future states that are elements of scenarios that are discovered hypothetically. A hypothetical-scenario generator is a key element of this approach. A hypothetical-scenario generator accepts, as its input, information on the current state of the engineering system. Then, by means of model-based reasoning techniques, it returns a disjunctive list of fault scenarios that could be reached from the current state.

The HSG is a specific instance of a hypothetical-scenario generator that implements an innovative approach for performing diagnostic reasoning when data are missing. The special purpose served by the HSG is to (1) look for all possible ways in which the present state of the engineering system can be mapped with respect to a given model and (2) generate a prioritized set of future possible states and the scenarios of which they are parts. The HSG models a potential fault scenario as an ordered disjunctive tree of conjunctive consequences, wherein the ordering is based upon the likelihood that a particular conjunctive path will be taken for the given set of inputs. The computation of likelihood is based partly on a numerical ranking of the degree of completeness of data with respect to satisfaction of the antecedent conditions of prognostic rules. The results from the HSG are then used by a model-based artificial-intelligence subsystem to predict realistic scenarios and states.

To avoid the need to create special models to generate hypothetical scenarios, the HSG uses the same model that is used to perform fault-detection and other diagnostic functions but interprets the results generated by the model in a manner unique to the generation of hypothetical scenarios. An important additional advantage of this approach is that a future state can be diagnosed by the same model as that used to diagnose the current state.

This work was done by Mark James of Caltech for NASA's Jet Propulsion Laboratory. Further information is contained in a TSP (see page 1).

The software used in this innovation is available for commercial licensing. Please contact Karina Edmonds of the California Institute of Technology at (626) 395-2322. Refer to NPO-42516.