Controlling Situated Agent Behaviors with Consistent World Modeling and Reasoning

by

Keith L. Doty† and Akram Bou-Ghannam *

†Machine Intelligence Laboratory
Department of Electrical Engineering
University of Florida, Gainesville, FL
email: doty@mil.ufl.edu

*IBM Boca Raton, Florida

Abstract

The control architecture for an intelligent, fully autonomous mobile robot is based on the philosophical view of combining reflexive behaviors and cognitive modules into situated agents in a complementary fashion, supported by a knowledge-based perceptual system. Behavior, cognitive, and perceptual subsystems make up the agent's intelligence architecture. Each of these three principal components decompose into layered, independent, parallel, distributed functions. The behavior based component of the system provides the basic instinctive competences for the robot while the cognitive part manipulates perceptual knowledge representations and a reasoning mechanism which performs higher machine intelligence functions such as planning. Cognitive control directly affects behavior through motivation inputs to behavior functions and through output behavior arbitration. A cognitive planning activity can execute plans merely by setting the motivation state of the robot and letting the behavior-based subsystem worry about the details of plan execution. The perceptual system offers a general framework for sensory knowledge generation, abstraction, and integration. All information in the perception knowledge bases derives from the fusing of real-time sensor data in a model-driven, multi-sensor system. This approach provides a current, consensual, and consistent interpretation of the environment observed by the agent. The perceptual module effects behavior through virtual sensor inputs supplied to behavior algorithms. The cognitive and perceptual modules can be influenced by behavior status inputs.

With this control architecture the agent gains the real-time performance of a behavior-based system without losing the effectiveness of a general purpose world model and planner. World models based on the robot's own sense perception appear to be essential for intelligent interaction with the environment. Experimental operation of a mobile robot implemented with this control paradigm in an unknown laboratory environment demonstrated the feasibility of the proposed behavioral-cognitive-perceptual architecture to explore and map the environment.

Keywords: autonomous agents, cognitive, perception, knowledge-based, artificial intelligence, behavior-based, situated agents, synthetic ethology, machine intelligence, control architecture,. parallel computation

INTRODUCTION

Animals live in a dynamic environment and tailor their actions based on their internal state and their perception of the external environment. Animal interaction with the environment becomes more complex as one ascends the hierarchy of organisms. Animals lower in the hierarchy behave reactively to stimuli where those at the top end of the hierarchy employ learning and intelligence. Where in the animal hierarchy should one look for guidance and inspiration in developing an autonomous agent? Should the engineer design 1) a machine modeled on human behavior and intelligence that is oriented towards symbolic reasoning and symbolic models of the world, or 2) a machine that derives inspiration from *insect intelligence* and features reactive and instinctual behaviors? The majority of *AI* researchers support option 1) and are too numerous to mention here. The behavior-based researchers [Brooks 86] [Brooks 90] [Connell 89] [Payton 86] [Anderson and Donath 88] [Agre and Chapman 90] advocate endowing autonomous agents with low-level behaviors that react to sensory information in a non-cognitive manner. Later, [Maes 90] and [Mataric

92] (the work here predates that of Mataric, however) demonstrated that reactive agent behavior can also be goal driven. The early fervor and vigorous assertions of the *situated-agents-grounded-in-reality* research group has resulted in recognition from the larger *AI* community of the value.

This paper, based on Bou-Ghannam's doctoral dissertation [Bou-Ghannam 91], represents an early attempt at a synthesis of the two views, or, perhaps more correctly, to bridge the architects and realization gap between them. About the same time, [Gat 91] published his dissertation apparently aiming for the same unification.

LESSONS FROM ETHOLOGY

In designing autonomous mobile robots, researchers can gain valuable insights from animal behavior. Animals successfully interact with their environment, including other individuals of their own species. Survival to reproduce constitutes the sole criterion for success. Animals appear to possess a combination of inherited instinctive responses to certain environmental conditions and the ability to adapt to new situations. Ethologists, who study animal behavior in their natural habitat, view animal behavior as largely a result of innate responses to certain environmental stimuli. Behavioral psychologists, on the other hand, study animal behavior under controlled laboratory conditions, and observe that animals learn as well. Observations by [Manning 79] support the existence of both learned and innate behaviors in animals. Animals with a short life span and small body size, such as insects, seem to depend mostly upon innate behaviors for interacting with the environment. Animals with longer life span and larger body size support larger amounts of neuronal tissue and depend more on learned behavior.

Reflexive behavior, perhaps the simplest from of animal behavior, produces a stereotypical response in the presence of specific environmental stimuli. The intensity and duration of the behavior depends only on the intensity and duration of the stimulus. Reflexive responses allow an animal to quickly adjust to sudden environmental changes and appear to be instinctive. Other reactive behaviors include orientation responses, where an environmental stimulus either attracts or repels the animal, and fixed action patterns elicited by specific stimuli [Beer 90].

The primitive, reactive behaviors mentioned in the previous paragraph may depend on the animals internal state as well as external stimuli. Motivated behavior, such as feeding, depends upon the animals state of hunger as well as the presence of food (external stimuli). The motivation potential of a motivated behavior varies with the level of arousal and satiation. In addition, such behaviors can occur in the complete absence of any external stimuli and can outlast those behaviors generated by external stimuli [Beer 90].

Given a diverse set of sensory information about the environment and a diverse behavioral repertoire, how does an animal select which information to respond to, and properly coordinate its many possible actions into a coherent behavior needed for its survival and reproduction? For example, [Anderson and Donath 90] reports observations by [Lettvin *et al* 70] about the frog's visual system as specialized in detecting movements of small, dark circular objects at close range. The frog cannot, however, detect stationary food objects or large moving objects. In general, animals possess a number of sensors with specialized sense modalities that assist survival and reproductive behaviors.

How do animals handle behavior conflict where certain stimuli and motivations cause the tendency to simultaneously perform more than one activity? Observations [Manning 79] suggest animals posses a behavior hierarchy where some behaviors take precedence over others while some behaviors are mutually exclusive. The animals internal state and environmental conditions determine the switches between behaviors. Behavior mode switches may be fuzzy and the relationship between behaviors non-hierarchical with partial overlap [Beer 90]. Anderson and Donath [Anderson and Donath 90, pp 151-152] provide an excellent summary of relevant ethology research as a model for autonomous robot control. Our response to these observations and our own suggests an

Intelligent Agent Design Manifesto

Intelligent, autonomous machines will include 1) a multi-layered, parallel, knowledge-based perception system, 2) a multi-layered, parallel, behavior subsystem, and 3) a cognitive system responsible for generating motivation, learning, planning and goals. In particular,

(A) Machine-embodied, autonomous, mobile agents (MEAMAS) must possess a set of innate behaviors which allow it to survive and respond to different environmental conditions. These include reflexive, phobic, tropic, and fixed-action-pattern behaviors.

(B) Adaptable Boolean, rule based, neural network and fuzzy switching mechanisms, conditioned by internal and external stimuli, will arbitrate behaviors.

(C) Multi-level, deterministic and non-deterministic finite state machines and Petri net controllers will provide behavior sequencing. A multi-level finite state machine consists of a finite state machine whose "states" consist of finite state machines.

No theoretical advantage accrues from a multi-level finite state decomposition, only a practical one. We suggest limiting multi-level finite controllers to three levels 1) cognitive, 2) motivated behavioral, and 3) reactive. The cognitive level defines goals, reasons about situations, invokes lower behaviors, and, sometimes, modifies lower behaviors (reinforced learning). Motivational behaviors receive motivation from either inherent survival requirements or from the cognitive level. Survival take precedence over the latter. Reactive behaviors only respond to direct internal and external stimuli. We will not discuss Petri nets any further in this paper, but will report on its application in future work.

(D) MEAMAS will, typically, only respond to a small subset of available sensory data and will be equipped with real-time sensors specialized to detect specific conditions suitable for its survival.

(E) Knowledge generators will fuse, organize, categorized, reduce, and transform sensory data. Most sensory information will be trashed. Data relevant to the MEAMAS will be structured in a manner that provides operational convenience and advantage to the agent.

(F) Multiple, specialized, knowledge generators, cognitive algorithms, sensor modalities, and behaviors will operate in parallel with emphasis on independent operation and minimum interfacing between parallel layers.

(G) Behaviors, perception modules, cognition processes, and knowledge generators will be modularized, layered, functional and operate in parallel.

The agent architecture should support many simultaneous processes where each process performs a single function. Processes that depend upon inputs from other processes act like Petri nodes, when all input conditions become satisfied, the node executes, in this case the process. Complicated message passing or sequencing can, thus, be avoided.

In the remainder of the paper we will elaborate on the various components and items listed in the manifesto and provide further motivation of our position.

BEHAVIOR-BASED VS. TRADITIONAL CONTROL OF AUTONOMOUS AGENTS

The traditional approach of control establishes a serial decomposition of the perception-action control path , for example, [Crowley 85] [Kriegman 89] [Moravec 85]. The behavior-based approach establishes a decomposition of multiple, parallel, perception-action control layers [Brooks 86] [Connell 89] [Payton 86]. Each perception-action layer performs its own task independent of the other layers, although one layer may inhibit outputs or suppress inputs of another. The parallel layered technique offers a flexible, modular approach with the advantage of direct, real-time perception to action.

Actually, the layered, parallel approach to the computation of behaviors applies equally as well to sense perception, knowledge generators, and cognitive manipulations. For example, multiple, dedicated, specialized vision algorithms (corner detection, line detection, blob detection, etc.) can process independently and in parallel. Specialized knowledge generators can independently collect inputs from sensors or other knowledge generators and create domain specific information. Cognitive algorithms that decide a plan of action can operate in parallel with algorithms that learn from past experiences.

PERCEPTION-BASED BEHAVIOR AND COGNITIVE CONTROL

Embodied, autonomous mobile agents may not need to keep a world models for low-level reactive behavior, such as wandering around while avoiding obstacles, but, for intelligent interaction, cognition, and planning, world models appear essential. We assert that both views can and should be accommodated into a single agent in order for

researchers to realize the next level of intelligent machine.

Figure 1 illustrates the decomposition of our agent control model into cognitive, perceptual and behavioral components. The behavioral (Fig. 2) and perceptual modules (Fig. 3) perform their respective tasks in parallel, while the cognitive modules perform serial tasks requiring greater data processing. We will detail the first two modules in subsequent paragraphs.

Although not implemented, cognitive modules admit of parallel operation as well. The dominant cognitive module might be defined anthropomorphically as the *I* of the agent and the others the agent's *subconscious*. A robot controlled by the combined architecture offers the real-time performance of a behavior-based system while maintaining the effectiveness and goal handling capabilities of a planner acting on a sense-generated, not synthetic, world model. By setting the motivation state of the robot, the cognitive module activates specific behaviors that bias the response towards achieving desired goals. The cognitive algorithms only specify the goals, not how to achieve them. Details of plan execution emerge from the resultant behavior-based actions induced by the goals.

To unite reactive and cognitive control, we chose to realize a mobile agent designed to generate a map of its environment and navigate by it. Model accuracy and consistency became a primary concern.

ISSUES IN WORLD MODEL CONSTRUCTION

The environment of a mobile robot is often unstructured and contains objects either to avoid or manipulate. In the tradition *AI* approach, a mobile robot must build and utilize models of its environment. This model should be accurate and remain consistent as the robot explores new areas or revisits old ones [Chatila and Laumond 85]. Our approach to consistent world modeling utilizes a spatially-indexed method similar to [Kent *et al* 87] and a 2-D line representation of [Crowley 87]. Kent employs a 2-D tessellation of space wherein each cell is tagged, *empty, occupied, unknown. (EOU)* Occupied cells are tagged further with object identification information. Crowley suggested a representation of a 2-D floor plan in terms of line segments. A difficult problem is to maintain a consistent model given inaccurate information and uncertainties. A variety of consistency checks help resolve these difficulties. Our consistent world modelling has been implemented as part of our *Sensory Knowledge Integrator* framework (Fig. 3) [Bou-Ghannam and Doty 90].

A great deal of this research effort focused on the agent developing an accurate spatial model in order to solve the problem of mapping and navigating in a 2-D space. While we achieved a measure of success with this approach, today we would not be as rigid or as concerned with precise measurements and modelling [Doty and Van Aken 93] [Doty and Seed 94] [Caselli *et al* 94]. Agents now replace precision with repeated sense measurements aimed at reducing the difference between its current and goal states. How novel (heavy irony)! The real twist in attitude from traditional approaches, however, is that agent failure is tolerated and agent success is neither guaranteed nor catastrophic if not achieved. With this attitudinal change, agent recovery from failure takes on a new significance. It becomes essential!

Our current embodied agents [Doty and Seed 94] [Doty and Harrison 93] [Jantz *et al* 94], not the one reported here, live with ambiguity and contradictions as well. Consistent and accurate world modeling reduces in importance, often being ancillary to the real objectives of the agent. Our argument does not eliminate the need for precise, consistent world models, just that research efforts in that direction might be misguided and should be carefully evaluated in terms of the agent's real objectives. While these comments may appear obvious, implementation is often far from obvious. In the Machine Intelligence Laboratory (MIL) we recommend the researcher perform an imaginary *Vulcan Mind Meld* with the robot agent in order to perceive the world as the agent perceives it, not as humans perceive it! Only then does it become possible to formulate sensible, realistic behaviors and tasks for the agent.

BEHAVIOR CONTROL FRAMEWORK

Figure 2 illustrates a layered, parallel, control architecture for autonomous agent behaviors. Agent behaviors are grouped according to competence level. Low competence level behaviors perform reflexive and fixed action patterns such as *obstacle avoidance*, *wall follow*, and *wander* and, thus, provide direct perception to action. Higher level behaviors depend on *motivation* factors and perception knowledge data bases.

Figure 4 illustrates the behavior control framework implemented on the Cybermotion K2A (Figure 5). The low level behaviors *wander, avoid obstacles,*

and *boundary follow* on *Competence Level 1* provide the basic instinctive competences. *Target navigation* and *curiosity* on level two depict higher level behaviors requiring *motivation* and *virtual sensory* inputs. Virtual sensors perform computations on data from the perceptual knowledge bases and, typically, "sense" conditions computed from other virtual or real sensory data. Virtual sensory information can have the immediate effect of direct sensory perception on behavior, as indicated in Figures 2 and 3. *Motivation* and *virtual sensory* inputs allow the cognitive system to modify the execution of various behaviors and, thus, provide great flexibility in the agent's response to environmental conditions.

The planning module reasons about knowledge generated by the perception system and plans tasks. We agree with [Agre and Chapman 90] in that plans serve as information not as procedures or commands. In our implementation the planning module employs a rule-based system to compute an arbitration and actuation strategy. To realize the planning system, we used CLIPS (C Language Implementation Production System), a knowledge-based system shell, as an implementation tool [Bou-Ghannam and Doty 91]. The planning module does not have to wait for a highly processed representation of the environment before it can effect action. Instead, it can select a behavior based on current perceptual knowledge and behavior status. In addition to motivation changes, the planning module implements its behavior decisions via the *Arbitration Network* and the *Actuation Resultant* module. The latter then computes a resultant value for the actuators. For example, if the agent's perception subsystem assigns virtual attraction and repulsion to objects in the environment, thus, generating a virtual potential field, the resultant actuation will generate motion that follows the negative gradient of the resulting potential field [Khatib 86]. Of course, gradient methods can lead to local minimum that, if used solely, could trap the agent into oscillatory behavior. For our agents, the gradient technique constitutes only one of many behaviors and cannot dominate action for any length of time. The planning module detects cyclic behavior and redirects the activity. Detection is not foolproof. One can conceive of long time-period cyclic behaviors that will not be detected by any planning system. The cycle time only needs to exceed the memory capability of the agent. An agent's last resort to detecting cyclic behavior is *boredom* ! The robotic agent *Riker* [Doty and Seed 94] constructs landmark maps by locating, contouring and identifying landmarks. During contouring, Riker may fail to make precise enough measurements to determine when he has completely traveled around the landmark. After a length of time, called the *boredom factor*, *Riker* simply abandons the current contouring task and heads out in an arbitrary direction! In animals, when conflicts arise that appear to have no resolution based on the current state of things, the animal also redirects its activity to break the deadlock. The resultant behavior often appears curious. The male stickleback fish, for example, will resort to nest building when confronted with a competitor at its boundary where the drive to flee balances the drive to attack the other male [Anderson and Donath 91]. *Redirected behavior, therefore, whether out of boredom, curiosity, cycle detection, or whatever, constitutes an extremely important behavioral capability.*

KNOWLEDGE BASED PERCEPTION

A powerful multi-sensor system must rely on extensive amounts of knowledge about both the domain and the problem solving strategy effective in that domain [Feigenbaum 77]. Domain knowledge can compensate for the inadequacies and uncertainties in sense data and help generate reasonable interpretations of domain object features.

The *Sensory Knowledge Integrator (SKI)* schema appearing in Figure 3 depicts a general model for sensor data fusion and perception [Bou-Ghannam and Doty 90] and follows a blackboard framework [Nii 86a,b]. *SKI* organizes domain knowledge and provides a strategy for applying that knowledge. Data-driven and model-driven, independent, parallel, knowledge generators transform information at one or more levels of abstraction into another level of abstraction. Data in the observations database range from sonar intensity and depth arrays at lower levels of abstraction, to lines, edges, regions, and surfaces at intermediate levels, to objects and their relationships at higher levels of abstraction. Partitioning the data into application dependent hierarchies or levels of abstraction makes it easy to modularize the knowledge base. Execution of both data and model knowledge sources produces new data which invoke execution of other knowledge sources until a high level description of the environment results.

The control module handles conflict resolution among the knowledge sources and, thus, determines what knowledge source or group of knowledge sources to apply next. The control module

determines the *focus of attention* of the system and is implemented in CLIPS.

The perceptual schema posed here has some similar properties to that of [Arkin 90], but our system differs significantly from his. Our agent does not integrate perception into the motor schemas and it does construct a map and plans. On the surface, this approach to perception appears to be symbolic *AI*. Actually, the map and plans do not dictate behavior, but only provide information to the behavior algorithms which posses a direct line from perception to action. If the perception inputs are not complete, goal attainment becomes less probable, but the agent continues to act , to the best of its abilities, without locking up or endangering its existence.

Figure 6 illustrates the map builder perception knowledge generator implemented on the Cybermotion robot according to the *SKI* model. From the sonar and pose data, four observations panels are generated. These observation panels serve as inputs to data-driven knowledge sources that generate higher levels of data abstraction. For example, conditioned sensory inputs drive the *Empty-Occupied-Unknown Knowledge Source (EOU-KS)* that generates an occupancy grid of a room in which the robot finds itself. The raw data also supplies information to the *Filtered Data KS (FD-KS)*. The *EOU-Representation* also serves as input to the *FD-KS*. The *FD-KS* generates the filtered sense data from which the *Line Find-KS* can determine the observed lines, hence, boundaries and layout of the room. The *Observed Lines* observation panel allows the robot to re-reference its pose (*New Pose Panel)* . The *Match/Merge-KS* generates and updates the *Model Lines* model panel, a composite of line data accumulated over time, from the *Observed Lines* observation panel. The *Match/Merge-KS* compares the expected scene to the *Observed Lines* observation panel. A match exists when the observed line roughly agrees with the model line. The *Match/Merge-KS* then fuses the two into a better estimate of the line. The *Re-Reference KS* uses the difference in pose between the observed and fused line to estimate the error in position and orientation of the robot. A new line observation will not match any model line. The *Match/Merge KS* will post the model hypotheses, to be confirmed or denied, that the line is new and not a sense error.

EXPERIMENTAL RESULTS

The Cybermotion K2A robot (Fig. 5) started its autonomous mission near the center of an unknown laboratory space about 10 feet by 30 feet. The agent's motivating influence to map its environment with the 12 sonars crowning its head in a semicircle initiated the *curiosity* behavior Curiosity spins the robot 360 degrees in place while taking a sonar scan before allowing the agent to venture into the unknown. The end of curiosity triggered the simultaneous execution of *avoid*, *wander*, and *boundary follow* behaviors. The map builder *SKI* works concurrently with the behaviors, continually building representations of the world by assimilating sensor data from the various locations visited by the robot.

As the robot collects sonar data the map builder gradually creates a line image of the room boundaries. The cognitive unit examines the image and ascertains regions where the agent has collected little to no information and triggers the *target-navigation* behavior to approach those areas. After the agent has visited all targets, the agent disables the *target-navigation* behavior and resorts to generalized wandering. A second pass of the room generates another line image of the room boundaries and the agent fuses this image with the previous one to obtain an even better estimate of the true boundary.

The agent successfully navigated, even when obstacles unexpectedly appeared in its path. A time-out prevented the agent from getting trapped when all paths to its target became impassable.

The implemented control architecture performed as intended during the experiment, guiding the robot safely through an unknown and unstructured environment without operator intervention. While the behavior based system reacted to immediate sensor readings, the map builder generated models of the world and navigation targets, and the planning module determined further actions in order to achieve the goal of building a map of the environment.

The behavior based system was implemented on an IRIS 2400 graphics workstation which communicated to the robot controller, an IBM PC AT, via an RS232C serial link. The robot controller, in turn, relayed the commands to the robot via a radio modem. This arrangement, while clumsy, allowed much quicker implementation than would have been possible otherwise.

SUMMARY AND CONCLUSIONS

We have described an architecture which allows for the implementation of a wide range of planning and control strategies, including those requiring timely information about the environment, and those requiring processing and assimilation of sensor data

over time for general purpose map-based planning. The architecture exploits the advantages of both behavior-based systems and traditional *AI* control architectures and was verified experimentally in the context of a mobile robot agent exploring and mapping a room unknown to it.

A design manifesto, derived from considerations of ethology, *AI*, neurobiology, behavioral psychology, and allied fields, outlines what we believe to be essential, guiding principles for the construction of an intelligent, autonomous machine. The intelligence mechanism should consist of 1) a multi-layered, parallel, knowledge-based perception system, 2) a multi-layered, parallel, behavior subsystem, and 3) a multi-layered, parallel, cognitive system responsible for generating motivation, learning, planning and goals. Layers in each of the principal subsystems should interconnect through inputs and outputs only and operate independently on current inputs, constantly revising and updating observations and derive knowledge. The perceptual and cognitive systems provide motivation and virtual sensors as input connections to behavior and actuation. Behaviors, in turn, provide status to the cognitive system to enable cognition to evaluate behavior responses.

Experiments with physically embodied agents adds to the excitement of this field. We encourage real world experimentation even for modestly simple tasks, not only to avoid the assumptions inherent in simulations, but also to speed up progress in a field where researchers do not agree about the fundamental principles involved in the construction of an intelligent autonomous agent.

REFERENCES

[Agre and Chapman 90]
Agre,P.E. and Chapman, D. "What are Plans For", *Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back* , Pattie Maes, Editor., 1990, pp 17-34.

[Anderson and Donath 88]
Anderson, Tracy L. and Donath,Max, "Synthesis of Reflexive Behavior for a Mobile Robot Based Upon a Stimulus-Response Paradigm," *Mobile Robots III, SPIE Proceedings,* Vol. 1007, 1988, pp198-211.

[Anderson and Donath 90]
Anderson, Tracy L. and Donath,Max, "Animal behavior as a paradigm for developing robot autonomy," in Designing Autonomous Agents edited by Pattie Maes, MIT Press, 1990.

[Arkin 90]
Arkin,R.C, "Integrating Behavioral, Perceptual and World Knowledge in Reactive Navigation",*Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back*, P. Maes ed., MIT Press, 1990.

[Beer *et al* 90]
Beer, R.D., Chiel, H.J., and Leon S., "A Biological Perspective on Autonomous Agent Design," *Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back*, P. Maes ed., MIT Press, 1990.

[Bou-Ghannam and Doty 90]
Bou-Ghannam, A., and Doty, Keith L., "A General Model for Sensor Data Fusion." 3rd Annual Conf. on Recent Advances in Robotics, May, 1990.

[Bou-Ghannam and Doty 91]
Bou-Ghannam, A., and Doty, Keith L., "A CLIPS Implementation of a Knowledge-Based Distributed Control of an Autonomous Mobile Robot," *SPIE Technical Conference*, Orlando FL, April 1-5, 1991.

[Bou-Ghannam 91]
Akram Bou-Ghannam, *Intelligent Autonomous Systems: Controlling Reactive Behaviors with Consistent World Modeling and Reasoning,* Doctoral Dissertation, University of Florida, 1991.

[Brooks 86]
Brooks, R.A., "A Robust Layered Control System for a Mobile Robot, " *IEEE Journal of Robotics and Automation,* Vol RA-2(1):14-23,1986.

[Brooks 90]
Brooks, R.A., "Elephants Don't Play Chess",*Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back*, P. Maes, ed., MIT Press, 1990.

[Caselli *et al* 94]
Caselli, Stefano; Doty, Keith L.; Harrison, Reid and Zanichelli, Francesco, "Mobile Robot Navigation in Enclosed Large-Scale Space", Proceedings *IECon94*, Bologna, Italy, Sept. 5-9, 1994.

[Chatila and Laumond 85]
Chatila, R. and Laumond, J.P., "Position Referencing and Consistent World Modeling for Mobile Robots," *Proc. IEEE Int. Conf. on Robotics and Automation,* St. Louis MO, 1985, pp 138-145.

[Connell 89]
Connell, J.H. , "A Behavior-Based Arm Controller", IEEE Trans. on Robotics and Automation, Vol. 5, No. 6, December 1989, pp. 784-791.

[Crowley 85]
Crowley, J.L., "Navigation for an Intelligent Mobile Robot", IEEE Trans. on Robotics and Automation, Vol. RA-1, No. 1, March, 1985, pp 31-41.

[Crowley 87]
Crowley, J.L., "Dynamic World Modeling for an Intelligent Mobile Robot Using a Rotating Ultra-Sonic Ranging Device, " *Proc. 1987 Workshop on Spatial Reasoning and Multi-sensor Fusion,* A. Kak and S. Chen ed., Morgan Kaufmann Pub., pp 293-302.

[Doty and Seed 94]
Doty, Keith L.and Seed, Steven L., "Autonomous Agent Map Construction in Unknown Enclosed Environments", *Proceedings MLC-COLT Workshop on Robot Learning*, Rutgers University, New Brunswick, N.J., July 10,1994, pp. 47-55.

[Doty and Harrison 93]
Doty, Keith L. and Harrison, Reid, "Sweep Strategies for a Sensory-Driven, Behavior-Based Vacuum Cleaning Agent", *AAAI Fall Symposium Series on Instantiating Real-World Agents*, Raleigh, NC, Oct 22-24, 1993, pp. 42-50.

[Doty and Van Aken 93]
Doty, Keith L. and Van Aken, Ronald E., "Swarm Robot Materials Handling Paradigm for a Manufacturing Workcell", *IEEE Int.. Conf. on Robotics and Automation,* May 2-6, 1993, Atlanta GA., pp. 778-782.

[Feigenbaum 77]
Feigenbaum, E. A. 1977, "The Art of Artificial Intelligence: Themes and Case Studies of Knowledge Engineering," *Proceedings of the Fifth International Joint Conference on Artificial Intelligence (IJCAI 77)*, pp. 1014-29.

[Gat 91]
Gat, E., Reliable Goal-Directed Reactive Control of Autonomous Mobile Robots, Ph.D. Dissertation, Virginia Polytech and State, 1991.

[Jantz *et al* 94]
Jantz, Scott D.; Yim, Eunbin; Harrison, Reid; Van Aken; Doty, Keith L.,"Swarm Robot Materials Handling in a Manufacturing Workcell", Poster and Demonstration presented in ALIFE4 Poster/Demo Session, MIT, Boston, MA,July 7, 1994.

[Kent *et al* 87]
Kent, E.W., Shneier, M.O. and Hong, T.H., "Building Representations from Fusions of Multiple Views," *Proc. IEEE Int. Conf. on Robotics and Automation,* 1987, pp. 1634-1639.

[Khatib 86]
Khatib, O., "Real-Time Obstacle Avoidance for Manipulators and Mobile Robots," *Int. J. Robotics Res.* 5(1), 1986.

[Kriegman *et al* 89]
Kriegman, D.J., Triendl E. and Binford, T.O., "Stereo Vision and Navigation in Buildings for Mobile Robots,"*IEEE Trans. on Robotics and Automation,* 5(6): 792-803.

[Lettvin *et al* 70]
Lettvin, J.Y. *et al,* "What the Frog's Eye Tells the Frog's Brain,"*Embodiments of Mind,* W. McCullock, ed.,MIT Press, Cambridge, MA, pp. 230-255.

[Maes 90]
Maes, P., "Situated Agents Can Have Goals," *Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back* , Pattie Maes, Editor., 1990, pp 49-70.

[Manning 79]
Manning, A., *An Introduction to Animal Behavior*, 3rd, ed., Addison-Wesley, Reading MA, 1979.

[Mataric 92]
Mataric, M.J. , "Integration of Representation into Goal-Driven Behavior-Based Robots", *IEEE Trans. on Robotics and Automatio*n, Vol. 8, No. 3, June 1992, pp. 304-312.

[Moravec and Elfs 85]
Moravec, H.P., and Elfs, A.,"High Resolution Maps from Wide Angle Sonar," *Proc. IEEE Int. Conf. on Robotics and Automation ,* St Louis, MO, pp. 116-121.

[Nii 86a]
Nii, H.P.1986, "The Blackboard Model of Problem Solving," *AI Magazine* 7(2): 38-53.

[Nii 86b]
Nii, H.P. 1986, "Blackboard Systems Part Two: Blackboard Applications Systems," *AI Magazine* 7(3): 82-106.

[Payton 86]
D.W. Payton, "An Architecture for Reflexive Autonomous Vehicle Control", IEEE Int. Conf. on Robotics and Automation, San Francisco, CA, 1986.

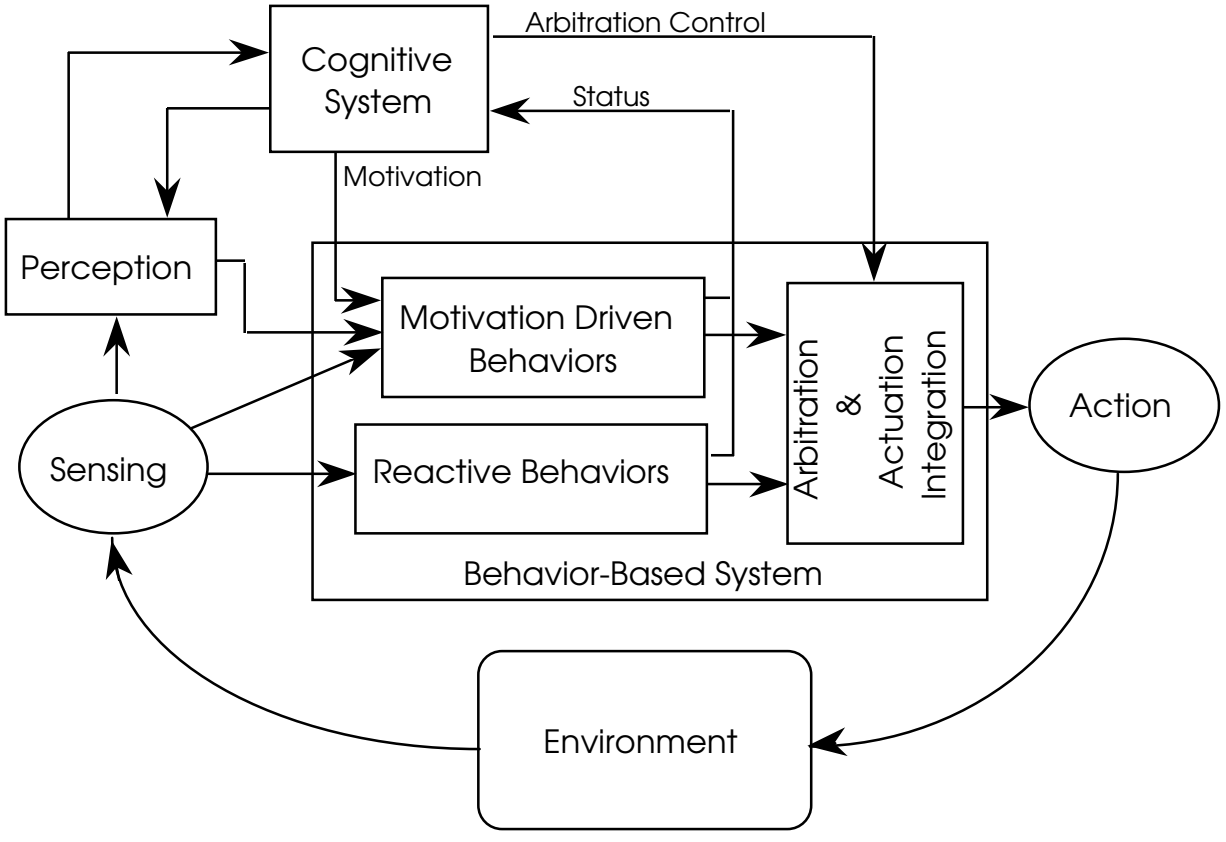

Figure 1 Combined Behavior-Based and Cognitive Control Paradigm

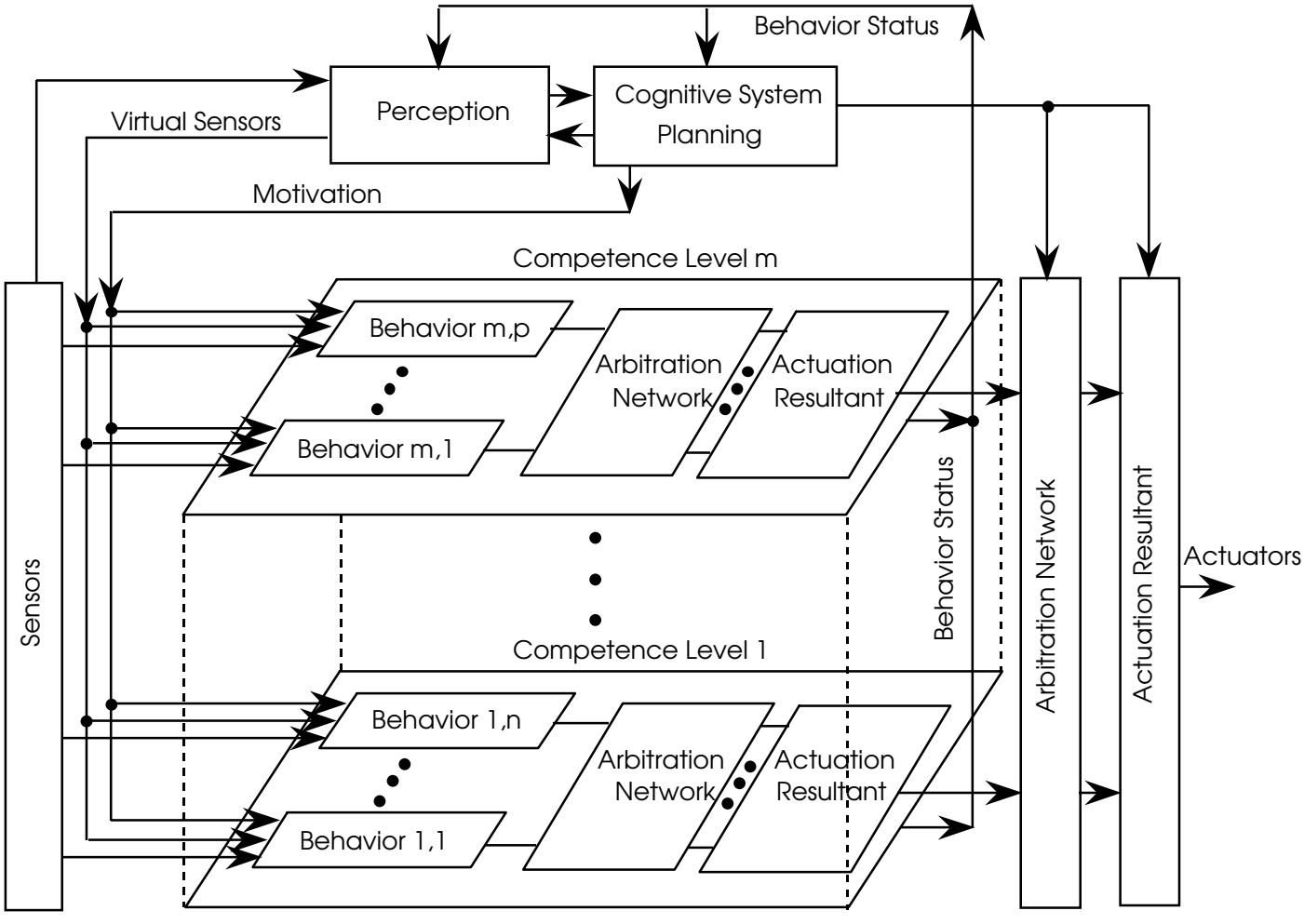

Figure 2 Details of the behavior control architecture for autonomous agents.

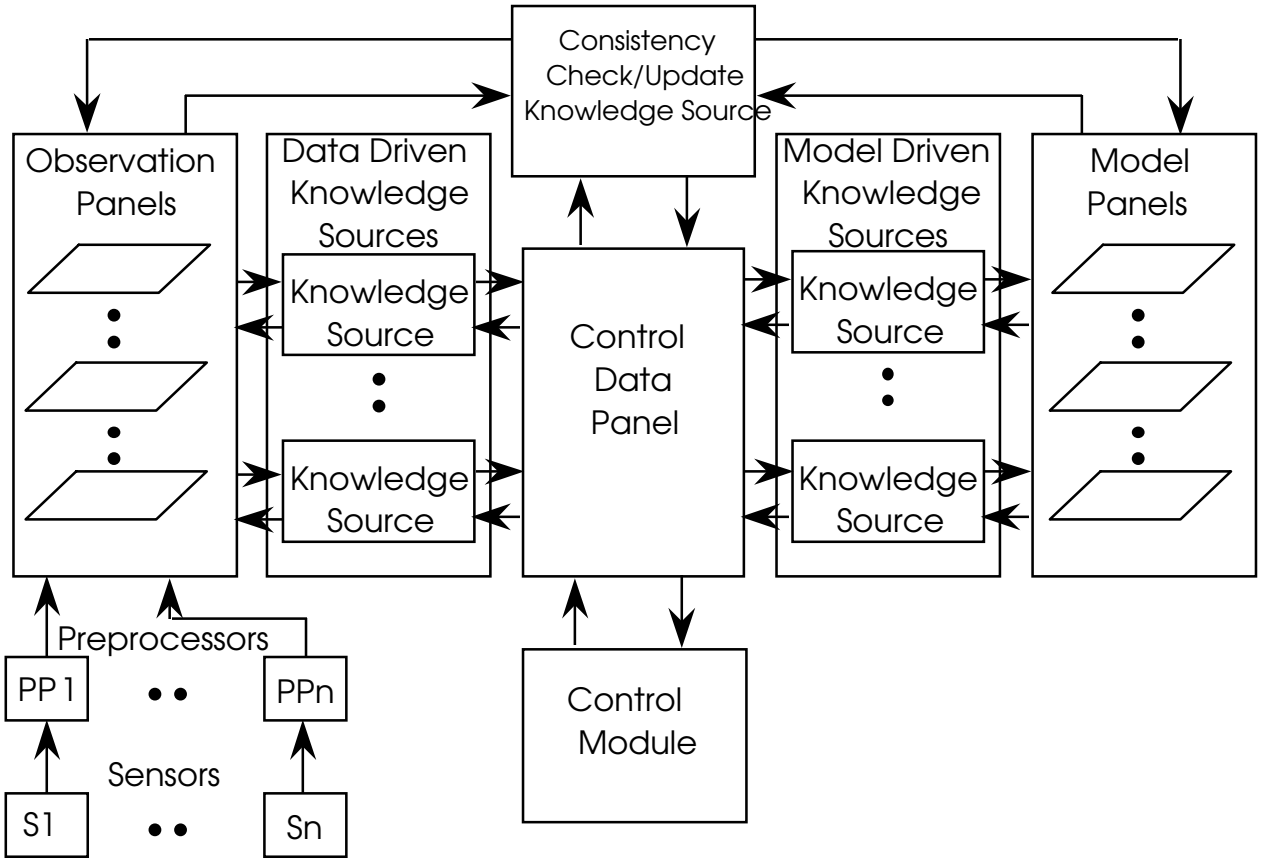

Figure 3 *Sensory Knowledge Integrator* implementation of a perception system.

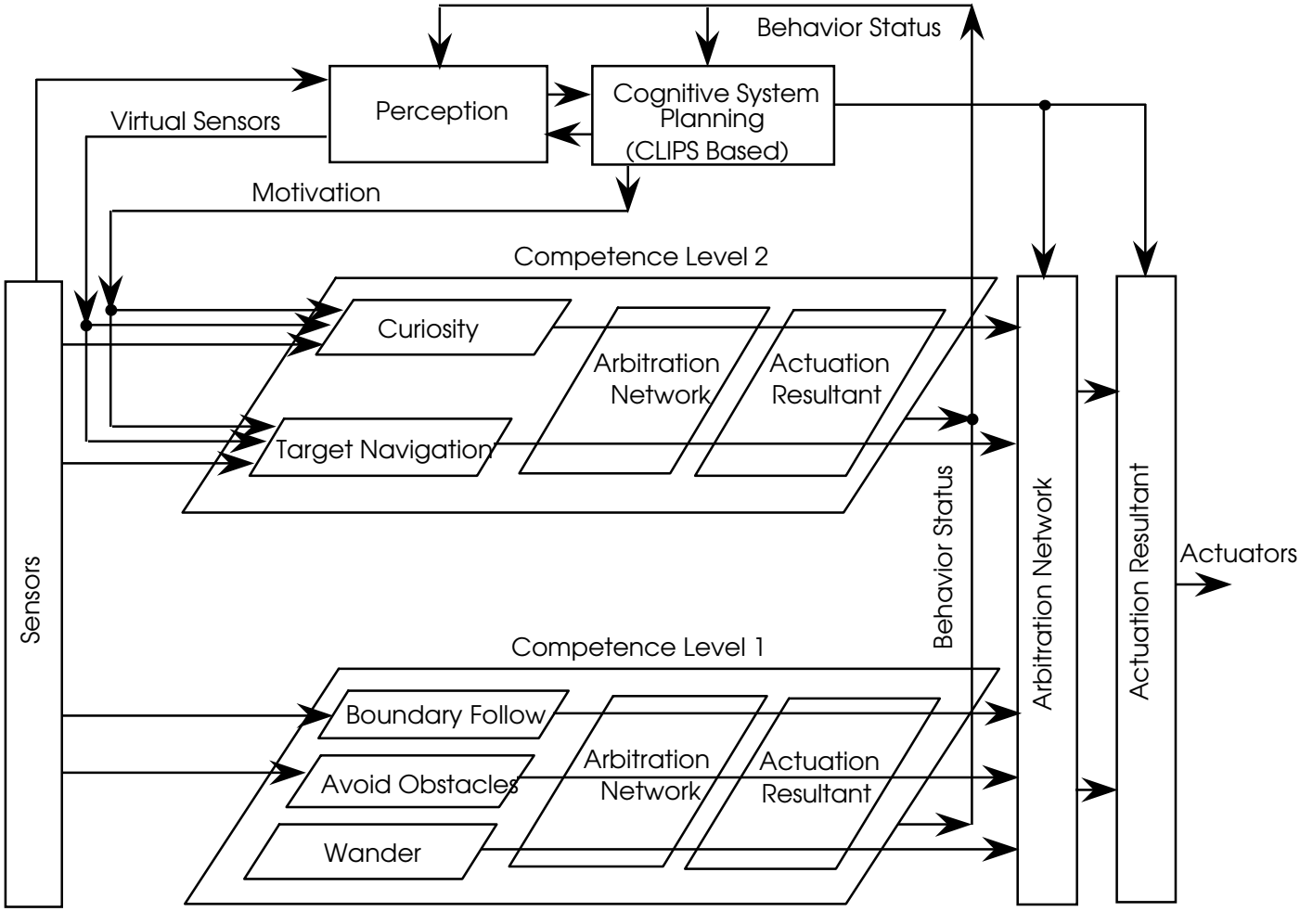

Figure 4 Specific implementation of a behavior control architecture for the Cybermotion K2A robot.

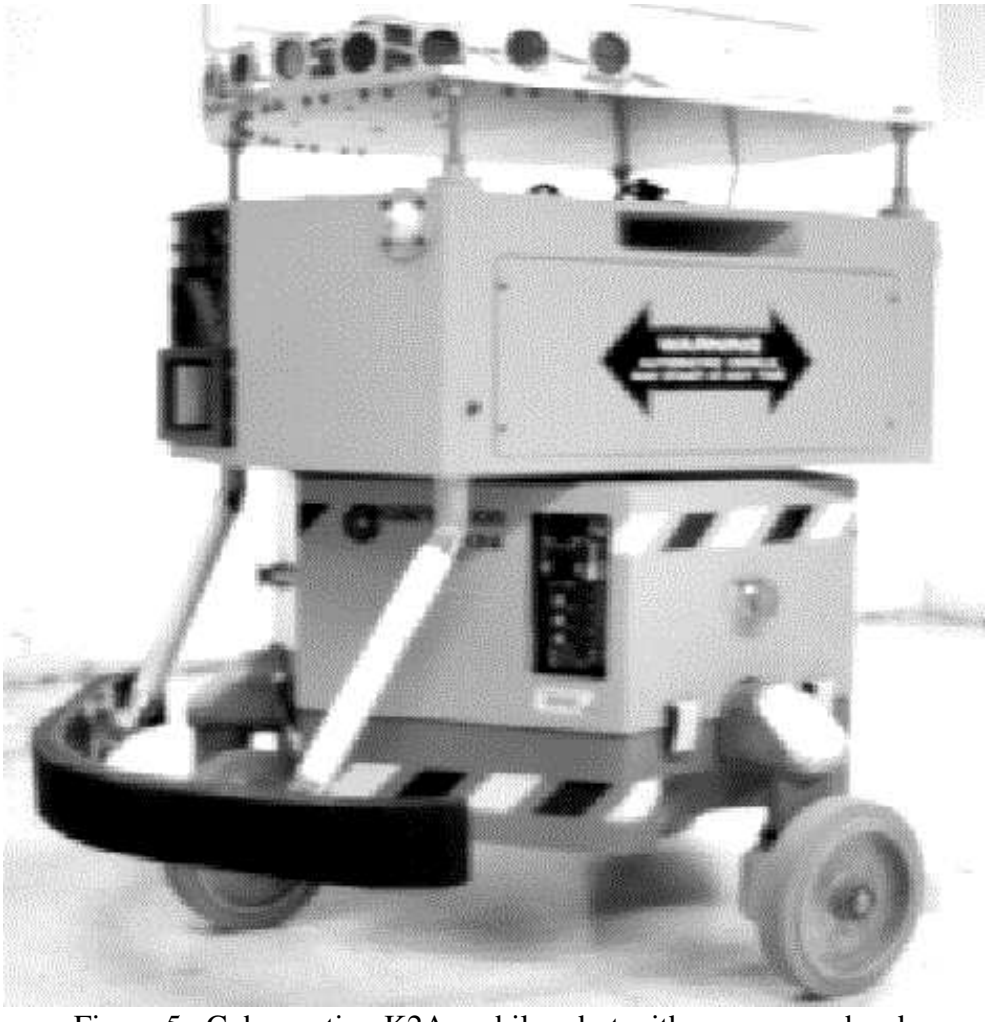

Figure 5 Cybermotion K2A mobile robot with sonar array head.

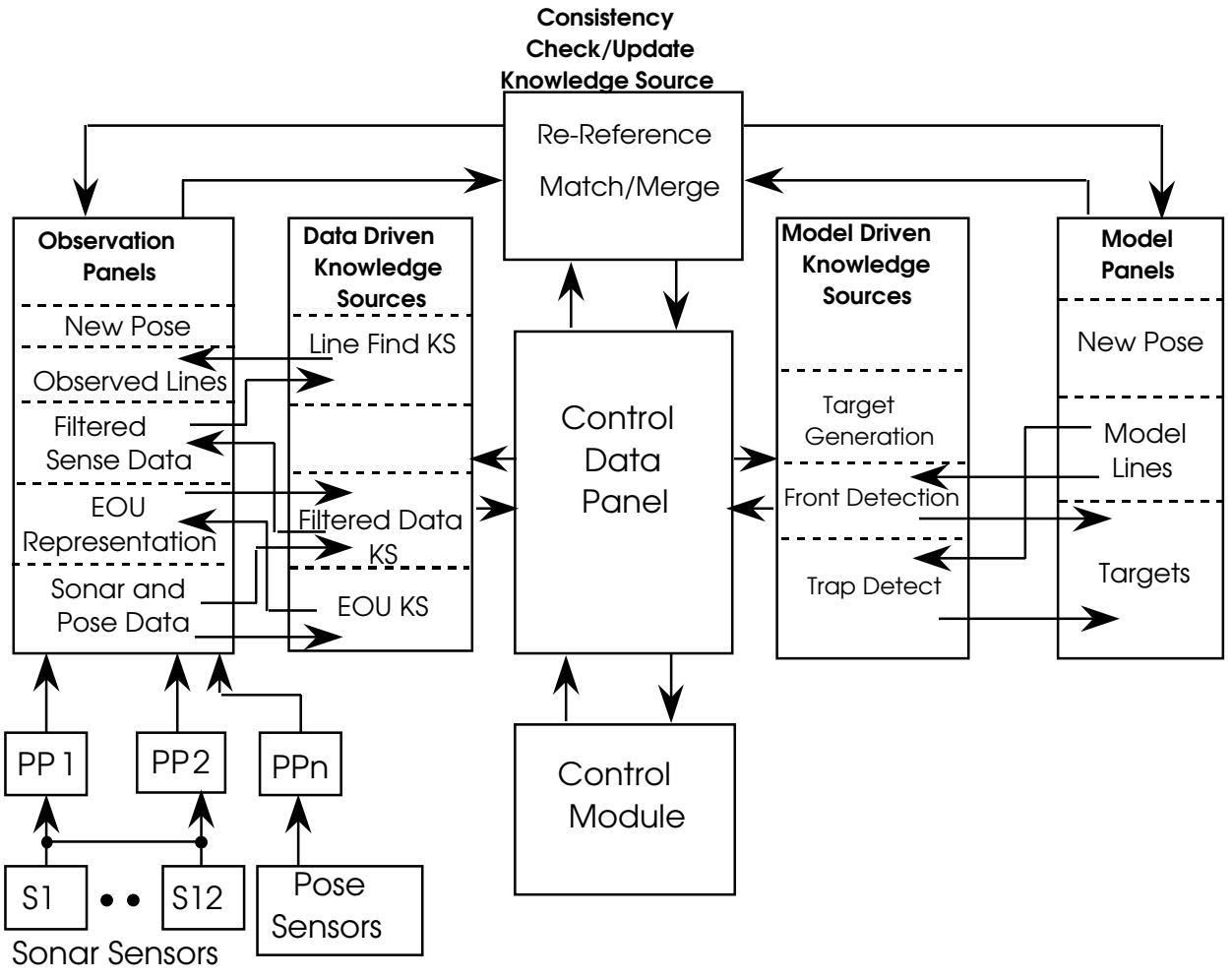

Figure 6 A map builder *Sensory Knowledge Integrator* architecture for the Cybermotion K2A robot.

POSITION PAPER ON LESSONS LEARNED FROM IMPLEMENTED SOFTWARE ARCHITECTURES FOR PHYSICAL AGENTS

by
Keith L. Doty

Director, Machine Intelligence Laboratory
Department of Electrical and Computer Engineering, University of Florida
email: doty@mil.ufl.edu

PROLOGUE

In the Machine Intelligence Laboratory at the University of Florida we have realized several physically embodied mobile agents from the primitive to the relatively elaborate. The implemented software architectures on the reactive machines become embedded within more elaborate structures as the robot architecture increases in complexity. The data and function encapsulation and information hiding techniques promulgated by the Object-Oriented architecture paradigm appear applicable, but have only informally been applied to our robots. Good software engineering practices generally apply: modular design, well defined interfaces, no implicit coupling, etc. However, since many modules must have access to the same sensory data, such data appears as global information, sometimes resulting in undesirable, implicit coupling of function modules.
Real-time operation proves to be a critical problem for realizing low level behaviors.

REALIZED AGENTS

The software architectures for the following realized agents will be discussed:

1. Riker 2. Gator 3. G2 4. Gestalt
5. Cybermotion K2A

The first four agents were designed and built at the Machine Intelligence Laboratory (MIL) while the fifth one is a commercial product. The MIL robots are small-- all will fit into a cubic foot. The K2A fits into 27 cubic feet. Of the MIL designs, G2 is the most elaborate physically and contains two microcontrollers. Gator and G2 have a forklift gripper while Riker and Gestalt do not posses manipulation capabilities. All the robots possess wheel encoders and bump sensors. All but the Cybermotion K2A possess IR. Only Riker and the K2A possesses sonar.

We generally kept the K2A tethered during operation for the purpose of RS-232C communication with a PC. All the other robots functioned without tethers.

COORDINATION

Behavior Arbitration

Behaviors on the MIL robots are time sliced when active. Finite state control behavior dictates which behaviors are active. Activation was determined by the state of the robot and its current sensory inputs. Technically, none of them are strictly reaction machines. Behavior arbitration is determined by sensory inputs and the time sequence of their appearance.

The K2A robot utilizes a CLIPS rule based system to plan and arbitrate the robots behaviors. For complex system this appears to be a viable approach. We have also successfully employed fuzzy logic control to arbitrate primitive behaviors for Medusa, another MIL robot not mentioned above. A fuzzy logic implementation consumes considerable memory and computational resources as the number of variables and the number of intervals for each variable increase.

Agent Cooperation

No explicit agent cooperation has been implemented to date. Current agent cooperation manifests as emergent behavior. Future plans do include explicit agent cooperation.

Agent Communication

Gator listened to IR beacons, but otherwise does not communicate. G2 has an IR communication link coupled with an IR communication beacon tracking head for direct line of sight communication. The K2A has a radio communications link which we used to download code to the robot and upload data to a PC.

Central vs. Distributed Coordination

Low level software architectures required of the "insect level" behaviors match either the real-time,

event driven architecture or a sensory selected execution of behavior subroutines. The K2A architecture employed a CLIPS rule based system to coordinate the activities of the robot. The decision rule predicates derived from sensory data and the state of the robot.

A microcontroller on the G2 robot was dedicated to the IR communications link and IR head tracking mechanism. A second microcontroller on the G2 controlled all other behaviors. The two microcontrollers communicated on a high speed serial bus, although the bandwidth requirements for the current application does not require high speed. The software systems on the two microcontrollers are completely disjoint.

With a single processor one could say the above control is central, but this observation is technical only. Basically, our approach has been to load up a processor with behaviors until its computational bandwidth saturates and then add another processor. Our underlying approach, then is distributed coordination. We plan to employ distributed coordination with future work on swarm robots as well.

Intervention of cognitive algorithms in future designs might be considered centralized control, but this will be true only in a weak sense. Cognitive algorithms provide goals, guidance and judgements upon which future action may be based, but will not necessarily detail how to achieve the goals.

Sequential Machine Realization

A state machine driven by sensory data and internal conditions controls the agent behaviors. For the simplest systems, the computational bandwidth of a microcontroller permits us to eliminate the use of interrupts.

INTERFACES

We categorize external interfaces into three groups: Agent-Agent, Agent-Human, Human-Agent communication.

Most of our currently implemented agents interact with the world only through their sensor inputs, with the exception of G2 and K2A. The latter have communications links which allow external agents to send information and mission goals. G2 communicates to a non-mobile agent that we call a machine. The machine sources and consumes parts carried by the robot. We plan to implement agent-agent communications into swarm behavior as well. K2A communicates a sonar data base from which a 2-dimensional sonar image is created by an off-board computer.

We have plans where our agents will name identified objects and communicate the name to a person via a radio link. The person may rename the object and inform the robot of the new name. The robot and the person will then communicate in terms of the person specified name. In other words, a shared vocabulary will be constructed.

We believe Human-Agent natural language communication beyond current voice recognition technology will not be needed for a long time. Many exciting voice possibilities have yet to be tried with what we have available now.

Lack of useful Agent-Human interfaces cause considerable experimental distress. Direct observation of the agent's behavior may not reveal enough information for debugging purposes. Several *ad hoc* solutions such as piezoelectric speakers, LEDs of different colors, seven segment displays and LCDs help. In the future, we plan to transmit the agents data bases to an off-board computer for interpretation and graphical display of the agent's perceptions. Some of our future agents will also use voice generation to tell the observer what sequence of actions the robot is taking at any particular time.

REPRESENTATIONS

In K2A sensory data drives the lower levels of a knowledge data base lattice.The minimum nodes of the sensory knowledge lattice contains the raw data. Higher nodes of the knowledge lattice maintain a particular level of abstraction of the data. Each node, in fact, is an abstract object with its own data processing algorithms and representation form. In this sense we believe that real agents must, in general, support multiple representations. Uniform message passing between objects and data integrity techniques may eventually mimic those developed in data base research. The K2A supports our most elaborate knowledge structures while only Riker, among the other four, possess a knowledge structure generated from sensory information.

Our opinion is that no single representation will be adequate or even necessary. We would like to emphasize the obvious, perhaps. Only knowledge derivable from sense data and model based hypotheses tested by experiment should find its way into any internal representation. Experimental verification may be done externally by human intervention and interpretation or internally by machine learning techniques. A common source of

agent behavior failure and experimenter frustration stems from asking an agent to perform a task that cannot be related to its knowledge bases.

STRUCTURAL

I favor distributed architectures both in our realizations and for future designs. Monolithic software or hardware designs have little future from our experience. The designer should strive to implement independent computational units. Interconnections and communications between computational units should be minimized to reduce traffic, complexity and overhead. Highly parallel computation will be useful for large complex autonomous agents, but again in a distributed manner. For example, the sense modalities of vision, audition, and smell (chemical detection) may benefit by separate parallel computer groups for each sense. In general, however, I am not particularly enthusiastic about SIMD type architectures.

Sensor and control architectures in K2A are multilevel. Ideally, most of the sensor and control levels operate in parallel, an objective not yet realized. The control architecture does not appear to require more than three principal levels (1) reactive, (2) behavioral, and (3) cognitive. Within these three levels, the designer may, and probably will, define sublevels which vary according to the dictates of the particular control being implemented.

The sensor knowledge bases may have many levels of representation, depending upon the sensor. A bumper switch indicates a collision and possibly locates the contact point on the agent's corpus. One level of structure for a bumper switch is adequate. Vision sensing, in contrast, may require many levels of abstraction above the raw pixel data, from blob representation, to edge detection to object identification.

PERFORMANCE

I would like to emphasize that autonomous agent performance evaluation must be an experimental process. Performance goals and metrics will typically be dictated by the application and often are qualitative, especially for primitive behaviors. For example, G2's success depends upon it (1) avoiding obstacles, (2) unloading parts from workcell machines, and (3) loading machines with parts. Meeting these basic goals qualifies G2 as a materials handler in a manufacturing workcell. With the basic behaviors established, however, we can then obtain quantitative information such as the number of machines serviced per unit of time by the mobile agent or the total throughput of a workcell serviced by such agents. The relation between these performance variables and the behaviors of the robot may be difficult to ascertain.

PSYCHOLOGY

Perhaps the concept of short-term and long-term memory derived from human memory function will have the greatest immediate impact. Our small agents are memory limited. Since most sense data have only a brief useful lifetime, most information can be thrown away after a short time. Information that must persist over longer periods of time, Riker's and K2A's map generation, for example, require more permanent memory. Certain information can be time-stamped and allowed to fade away with time. How to make the data fade will be an interesting problem to solve. Critical knowledge should be stored in EEROM or EPROM. Less critical information in battery backed-up RAM.

I personally am not interested in developing androids at this time. The gap between insect intelligence and android intelligence is too large. Much interesting and exciting work remains to be done at the insect level, work that should keep us busy for a long time. Current android work will be perceived in the future as visionary, but such observations are illusory. The work will only appear visionary with the technologies of the future in hand. Seen in the light of current technologies such work appears premature. Should there be no android research then? No, I do not suggest that. Android activity will serve as a guide towards the ultimate intelligent machine and indicate where the trouble areas lie. The level of effort and resource commitment, however, should be steady and at a fairly low level.

Our work couples strongly with some of the principles and techniques of animal behavior and behavioral psychology and mutual sharing will probably advance all three fields. Our experience demonstrates that the relationship between perceived behavior vs programmed behavior can often be subtle and utterly surprising. For example, translating low level behaviors into precise wall following proved to be more difficult than first imagined. Crude wall following was easy to implement. Following walls with concave and convex corners of various degrees proved more challenging. Through tremendous amounts of experimentation and adjustments, G2 can now contour a segment of solid cardboard fence arranged in any open curve configuration using only

IR sensors! Of course, the fence convolutions must allow for the turning radius of the robot. If not, G2 will not perceive the variation in the fence or will find itself in a narrow dead-end from which it backs out.

SIMULATION

Simulation has played an important role in our research. Simulation provides a field to test agent behavior algorithms under ideal conditions. Those algorithms that fail under ideal conditions can, in general, be eliminated for realization. Rapid agent behavior evolution becomes possible and many ideas can be tested. On the other hand, algorithms that show promise in simulation can fail in their realization for a number of reasons, (1) the sensors and actuators do not provide ideal operation, (2) the algorithm requires too much sense data or control precision, (3) the algorithm makes inaccurate assumptions about the environment, (4) the algorithm encounters unanticipated environmental conditions, (5) the algorithm takes too much time, etc. Simulation, therefore, cannot be a substitute for realization and experimental verification. In the final analysis, this discipline must be *experimental* for real validation of behaviors predicted by theory.

LEARNING

One aspect of learning involves information gathering. Riker, for example, explores his environment looking for navigational landmarks. As Riker encounters landmarks, contours them to identify them, and then stores them in a data base, he increases his knowledge of his environment and is able to act upon that knowledge. As the environment changes Riker's knowledge about that environment changes. From the viewpoint of an outside observer, Riker appears to learn about his environment and act upon it.

We have not yet implemented reinforced learning but plan to in the future. At low levels of performance, reinforced learning does not appear to be as important as it is at intermediated and high levels. Model based reasoning plays even a lesser role at the low and intermediate levels of robot operation. Experientially derived strategies appear to have more bearing on problem solving at these two levels then model based reasoning.

As we perfect the agent's basic hardware and software platforms, reinforced learning will play a more significant role in its performance. Our hypothesis, based on limited experience, is that a robot will support a multitude of different learning modules, each tuned to a specific set of sensors and behaviors. I do not see a universal, all encompassing learning model applicable to all situations. I propose that learning, like behaviors, should be individualized, particular and independently operant. I call them abstract learning operators. For example, one vision algorithm may only be responsible for identify looming objects greater than a certain size, while another might be dedicated to finding regions of the visual field possessing a particular color. Ideally, all sensor abstraction algorithms operate in parallel to provide real-time information. In this example, both abstract learning operators transform the raw pixel data. This is not always the case. Knowledge generated by one learning algorithm might be useful to a higher (more abstract) operator.

In K2A sensor and control algorithms access information and knowledge at a variety of levels. One way to make all levels simultaneously available is through a shared memory. In most cases an algorithm operates only on abstract data of one class. For example, the K2A EOU (Empty, Occupied, Unknown) algorithm operates only on the raw sonar data to create an EOU knowledge base. Navigation algorithms, in contrast, only operate on the EOU knowledge base and not the raw data. The raw data may supply different operators.

ACTUATION AND CONTROL

I postulate three principle layers of control (1) primitive reactive, (2) behavioral, (3) cognitive directive. The first layer must be real-time and respond at about a 10 to 50 HZ rate. Behavior responses vary from 1 to 10 HZ rate and cognitive directives (goal setting, planning, judgement), from 1 to 0.001 HZ, or even less for some cognitive activities. Other aspects to control and actuation have been elaborated in previous paragraphs.

SENSING

I propose an abstract sensory knowledge lattice where nodes at higher levels hold increasingly more abstract versions of the data. The edges between nodes, the lattice relationship, depicts data dependency. The minimum nodes of the lattice represents the raw data of a particular sensor. Higher nodes, whose information content necessarily derives from lower level nodes, represent data abstractions and generated knowledge. Access to the lattice should not be

dictated by the lattice structure, which only serves to identify the levels of abstraction to the designer. Access should be random at the node level. Within a node, node specific functions and access mechanisms apply.