Modular Adaptivity and Behavior Based Control

Joanna Bryson

Laboratory for Cognitive Neuroscience and Intelligent Systems
Edinburgh University
Level 8, Appleton Tower
Edinburgh EH8 9JZ, UK
Joanna.Bryson@ed.ac.uk
http://www.ed.ac.uk/~isl/Lab_home.html

Abstract. Learning in animals is modular (though not encapsulated), and there is growing interest in modular, non-homogeneous models of learning within the neural networks community. Behavior-based artificial intelligence, with its emphasis on modular components, seems an ideal platform for implementing such systems in autonomous agents. However, historically this community has shunned representation issues, and there is no established system for combining modular control and modular adaptivity. This paper describes a behavior based architecture and development approach designed to enable research in modular adaptivity.

1 Introduction

The Brain is an adaptive controller with modular construction. This is not to say that the various processes of the brain are fully encapsulated. Nerve cells appear to have their growth partly regulated by actively seeking correlated signals, which may come from neighboring or even distant regions (Willshaw & von der Malsburg 1976). Over the course of evolution, the functionality of brain regions shift as they become exploited and enervated by different areas (Livesey 1986). Nevertheless, different regions of the brain have markedly different architectures and computational properties.

Recent work in AI and neural networks emphasizes the utility of having multiple cooperating learning processes(McClelland, McNaughton & O'Reilly 1995, Moll & Miikkulainen 1997, Turney, Whitley & Anderson 1996). Behavior-based artificial intelligence (BBAI), which models intelligence as interacting modules of specific behaviors (Minsky 1985), would seem the ideal platform for constructing such systems. However, historically, the first and most successful approach to implement BBAI, subsumption architecture (Brooks 1991), emphasized a lack of representation and learning. All adaptivity arose from the flexibility of the architecture and changes in external state, not changes in the agents' behaviors. Although more recent work has included adaptive elements (Maes, Mataric, Meyer, Pollack & Wilson 1996), these elements tend to be either a single specialized learning component or a monolithic system to control or design the entire agent. No principled methodology has been established for combining modular behaviors and modular learning.

Bryson (1995) demonstrated a system based on subsumption architecture with several behaviors consisting of neural networks modules, and suggested that adaptivity requirements should be the basis of behavior decomposition for such systems. This paper presents a significant development of these ideas. We present the notion of a behavior library developed under object oriented design methodology. The classes for this object library are based entirely on the state requirements for a modularized learning system. Behavior primitives are based on methods of these classes.

2 Architecture

The structure for a behavior-based system is provided by an architecture. Architectures are generally considered to be a set of rules describing the interactions of behaviors within such a system. We consider this description incomplete. An architecture implies a methodology. This implication is made explicit in our work. In our architecture, Edmund, this methodology addresses not only the organization of the behaviors, but the process for their development. We also designed Edmund specifically to enable modularized learning. Such considerations are important in BBAI, because the approach depends almost entirely on the hand-coding of an agent's behavior (Wooldridge & Jennings 1995).

Edmund is designed to address the issue of scaling the behavior-based approach to building intelligent systems without resorting to a hybrid symbolic / reactive architecture (Bryson under consideration). Besides addressing issues of design methodology and learning, it also introduces elements of control state for behavior sequences and flexible pseudo-hierarchies (see figure 1). Sequences are necessary because strict parallelism provides insufficient bias for action selection and goal persistence. Animal intelligence appears to provide sequencing primitives for both representation and action (McGonigle & Chalmers 1996, Tanji 1996). Clustering behaviors into 'competences' further assists with action selection. This priming or selectively attending to behaviors that are useful within a particular context results in behavior selection that appears hierarchical, but Edmund does not constrain reactivity by maintaining stack nor does it pass error values between levels. New sequences are always selected as a result of the environment and internal motivation, the appearance of hierarchical behavior is dependent entirely on continuity of context.

The behavior library described in the introduction is Edmund's Nomad Robot Library (ENRL). Thirty-two sensing and action primitives for the architecture are based on methods from six classes (see figure 2). Each class is based on a knowledge-set representing attributes of the external world or the agent's mechanism. Both Edmund and ENRL are currently implemented in C++, though they were prototyped in Lisp (CLOS) and Perl 5.

As its name implies, ENRL is implemented on a commercially available Nomad 200 mobile robot that operates in a standard laboratory environment of five rooms connected by two hallways with much scattered furniture and experimental equipment. Its processing is conducted on-board with a 486 processor

Sense	A perceptual primitive.
Act	An action primitive (both external and internal acts).
Trigger	A sequence of primitives executed atomically, for selecting actions. Should have no impact external to agent.
Lap	An action pattern, a sequence of primitives which may have external impact. The *L* is for “learnable”, not yet implemented.
Competence	A prioritized set of Laps or other Competences and their triggers. Includes a goal lap that, if successfully triggered, indicates completion.
Drive	A type of competence (may also contain Drives) that runs its components in parallel. Generally has no goal.

Fig. 1. The classes composing the Edmund's control architecture.

Direction	For moving and detecting motion-relevant information, knows current and preferred directions. Instances correspond to faces of the robot.
PMem	Perceptual Memory, a short list of recent combined sensor readings. The main source of sensing. No significant instances.
Bump	Provide information about local “invisible” obstacles. Instances represent collision points.
Wave	Detect people waving at sonar (a source of communication).
DpLand	Decision point landmarks, the basis for navigating (vs. wandering.) Instances represent choice points and the decisions made there.
DpMap	A loose grid-like structure for holding DpLand-marks. Instances represent squares of space containing the marks.

Fig. 2. The classes composing Edmund's Nomad Robot Library (ENRL).

running Linux OS. It is approximately 1m high and 0.5m in diameter. It has 16 sonar and infrared sensors arrayed evenly around its circumference, and 20 bump sensors located in two rings at its base. The sonar rings are at the top of the robot, while the IR is just above the bumpers. The Nomad can also detect its own battery level and approximate odometric information in Cartesian coordinates. It has 3 degrees of motion of which ENRL currently uses two, translation and rotation. It can also speak via a speech synthesizer.

Since our Nomad has no manipulator, the tasks we address are almost entirely navigational[1]. Tasks or ‘personae’ for the robot are coded in scripts that specify the ordering of primitive behaviors (either actions or senses) from the library into the higher level classes of the control architecture. Specifying control in scripts

[1] Although Edmund was prototyped in a blocks-world simulation (to demonstrate its ability to control complex behavior despite a reactive architecture), that library contained no learning elements. ENRL, implemented on a real robot, required adaptive elements immediately to cope with the noise and complexity of live sensor data.

both facilitates the quick programming of new tasks and the development and debugging of the behavior library.

3 Learning Modules Under the Behavior Based Architecture

Modular adaptivity has been central to ENRL since it was first able to control the robot. For example, the sonar sensors pick up each other's signals at arbitrary intervals, thus sonar readings must be checked against a short perceptual memory to sanitize the behavior from the reactive architecture. The sonar is sensed at 8-12Hz, the sequencing of the firing has been set so that misreadings seldom persist more than a tenth of a second. If a reading changes by over 8 inches within one cycle, its true value is stored but not perceived. The perceived value, that used by other behaviors, is based on the previous value plus a trend. If the new reading persists for 3 cycles, it then enters perception. This system is robust enough to pass through intricate doorways[2] and still avoid humans to the extent that the Nomad can be easily herded into desired locations.

Similarly, there can be no way for a fully reactive robot to take into account bumper information, since the robot must move away from the object it has impacted (thus losing the signal) before it can proceed to avoid it. Subsumption architecture robots stored the necessary state to *back up and turn* in the finite state machines composing their behaviors. The Nomad under Edmund stores bump information as a body-based coordinate which is factored into the range perception of the sonars and IRs. This allows for a smooth corrected trajectory, and could provide a basis for a consolidated, long-term memory of the obstacle, or at least of a path that avoids it.

We are currently using ENRL to implement a biologically inspired navigation model with multiple memory elements. The model combines short and long term memory via a highly indexical working-space (Teyler & Discenna 1986, McClelland et al. 1995, Moll & Miikkulainen 1997). We are using this model to implement navigation (as opposed to wandering) on our robot. LTM in our implementation is composed of DpLand objects representing recorded decision points with trajectories. The working memory is a simple (though constructive) grid that organizes discovered landmarks into square meters, each a DpMap object. This facilitates a 1-nearest-neighbor algorithm by providing a set of nodes within a nine square radius that are likely to be nearest. Episodic memory is also DpLand objects, which in this case are simply stored off the assumed relevant nodes of the working memory.

Having episodic memory present allows for additional strategies and enhancements to the navigation algorithm. If the robot has been led into a different trajectory than normal by a change in its environment, it may encounter a region not well served by its navigation strategy. This can be detected by a growth of episodic memory elements in the same location as the robot. The robot can

[2] Inset right angles such as are found in doorways often confuse sonar sensing.

react to this situation by trying an alternative strategy, such as going to an open location in the direction of more landmarks. Before episodic memory is disposed of by the robot, a process can search for episodes that resulted in new trajectories if such novelty is one of the indices or dimensions of the working space. The robot then performs a cheap, constructivist form of consolidation: it simply adds the relevant nodes to the LTM. Thus experience can result in new nearest neighbors.

It is important to note that the new memory model is *additional*, it in no way bypasses or modifies the established perception and memory routines of ENRL. Obstacle avoidance and the emergency response to collisions has not had any perceptual or state information altered by this process. The robot is still reactive to the world around it; it simply has a new element to select direction rather than random wandering and persistence.

The relation between memory modules is overlapping but orthogonal. The functions of the sensing and acting primitives are almost entirely composed of methods on the various ENRL classes holding Edmund's internal state. However, several key primitives, such as the navigational *pick-dir*, or the motion sensing routine *compound-sense* rely on multiple memory modules. As mentioned above, this violation of module encapsulation is also fundamental to brain organization. Almost no behaviors are independent of any learning module, the exceptions are simple signaling routines: open-loop actions to communicate to the user. This should be expected, BBAI behaviors are supposed to be tightly-coupled perception/action routines, and in animals there is no perception without memory (Carlson 1994).

4 Discussion and Conclusion

The memory model we are exploring has many implications for artificial intelligence. We are interested in looking at the interaction between episodic recall and the learning of the indexical structure stored in long-term memory, the additions of more special-purpose memory systems such as the amygdala, and the voluntary access of episodic memory apparent in humans. Historically, such questions are generally explored via thought experiments and introspection. By grounding our research in systems that cope with real perceptual and actuation issues, we hope to establish more substantiated results than might otherwise be achieved. Our experience with the constraints and practicalities of implementing on a real robot substantially altered Edmund's control architecture. We expect the same to be true of our memory and perception model as well.

In summary, we believe that the autonomous agents approach an important part of research into artificial intelligence, particularly in systems that try to model biological processes. So far, the most successful approach to intelligent robot programming has been behavior-based systems, but no principled system for combining a biologically inspired, modular model of adaptivity into a behavior based control system has been established. This paper has described an architecture for a behavior library based around state for modular learning, with examples from our implementation on a mobile robot. We hope this approach will enable research into memory as well as increasing the capabilities of autonomous robotics.

References

Brooks, R. A. (1991), 'Intelligence without Representation', *Artificial Intelligence* **47**, 139–159.

Bryson, J. (1995), The Reactive Accompanist: Adaptation and Behavior Decomposition in a Music System, *in* L. Steels, ed., 'The Biology and Technology of Intelligent Autonomous Agents', Springer-Verlag.

Bryson, J. (under consideration), Agent Architecture as Object Oriented Design, *in* M. P. Singh, ed., 'The Fourth International Workshop on Agent Theories, Architectures, and Languages (ATAL97)', Springer-Verlag.

Carlson, N. R. (1994), *Physiology of Behavior*, 5 edn, Allyn and Bacon, Boston.

Livesey, P. J. (1986), *Evolutionary Processes*, Vol. 1 of *Learning and Emotion: A Biological Synthesis*, Lawrence Erlbaum Associates, Hillsdale, NJ.

Maes, P., Mataric, M. J., Meyer, J.-A., Pollack, J. & Wilson, S. W., eds (1996), *From Animals to Animats 4: Proceedings of the Fourth International Conference on Simulation of Adaptive Behavior*, MIT Press, Cambridge, MA.

McClelland, J. L., McNaughton, B. L. & O'Reilly, R. C. (1995), 'Why there are complementary learning systems in the hippocampus and neocortex: Insights from the successes and failures of connectionst models of learning and memory', *Psychological Review* **102**(3), 419–457.

McGonigle, B. & Chalmers, M. (1996), The Ontology of Order, *in* L. Smith, ed., 'Piaget: A Critical Assessment', Routledge.

Minsky, M. (1985), *The Society of Mind*, Simon and Schuster Inc., New York, NY.

Moll, M. & Miikkulainen, R. (1997), 'Convergence-Zone Episodic Memory: Analysis and Simulations', *Neural Networks*. in press.

Tanji, J. (1996), Involvement of motor areas in the medial frontal cortex of primates in temporal sequencing of multiple movements, *in* R. Caminiti, K. Hoffmann, F. Lacquaniti & J. Altman, eds, 'Vision and Movement: Mechanisms in the Cerebral Cortex', Vol. 2, Human Frontier Science Program, Strasbourg, pp. 126–133.

Teyler, T. J. & Discenna, P. (1986), 'The hippocampal memory indexing theory.', *Behavioral Neuroscience* **100**, 147–154.

Turney, P., Whitley, D. & Anderson, R. (1996), 'Evolution, Learning, and Instinct: 100 Years of the Baldwin Effect', *Evolutionary Computation*. special issue.

Willshaw, D. J. & von der Malsburg, C. (1976), 'How Patterned Neural Connections Can Be Set Up by Self-Organization', *Proceedings of the Royal Society of London B* **194**, 431–445.

Wooldridge, M. & Jennings, N. R. (1995), 'Intelligent Agents: Theory and Practice', *Knowledge Engineering Review*.