Learning Hierarchical Representations and Behaviors

Robert A. Hearn
Dartmouth College
Neukom Institute for Computational Science
Sudikoff Hall, HB 6255
Hanover, NH 03755, USA
robert.a.hearn@dartmouth.edu

Richard H. Granger
Dartmouth College
Department of Psychological and Brain Sciences
6207 Moore Hall
Hanover, NH 03755, USA
richard.granger@dartmouth.edu

Abstract

Learning to perform via reinforcement typically requires extensive search through an intractably large space of possible behaviors. In the brain, reinforcement learning is hypothesized to be carried out in large measure by the basal ganglia / striatal complex (Schultz 2000; Granger 2005), a phylogenetically old set of structures that dominate the brains of reptiles. The striatal complex in humans is integrated into a tight loop with cortex and thalamus (Granger and Hearn 2007); the resulting cortico-striatal loops account for the vast majority of all the contents of human forebrain. Studies of these systems have led to hypotheses that the cortex is learning to construct large hierarchical representations of perceptions and actions, and that these are used to substantially constrain and direct search (Granger 2006) that would otherwise be blindly pursued by the striatal complex (as, perhaps, in reptiles). This notion has led to construction of a modular system in which loops of thalamocortical models and striatal models interact such that hierarchical representation learning in the former exerts strong constraints on the trial-and-error reinforcement learning of the latter, while reciprocally the latter can be thought of as testing hypotheses generated by the former. We report on explorations of these models in the context of learning complex behaviors by example, in simulated environments and in real robots.

Software framework. The essential components of the system are shown in Figure 1. A given *cortical module* comprises a *perception* component and a *target* component. At a conceptual level, these components correspond to, respectively, the superficial and deep layers of cerebral cortex: superficial layers receive, process, and represent input, whereas deep layers send processed output to subcortical structures. In this abstract model, the perception component performs unsupervised learning: it observes statistical properties of its inputs, and learns a compressed state representation capturing the most essential information in its current input. The target component, on the other hand, learns to predict the next state of the perception component, by sequential learning: it learns a function mapping current perception state plus contextual cues to a distribution over next perception states (see, e.g., (Granger 2006)).[1] In modules

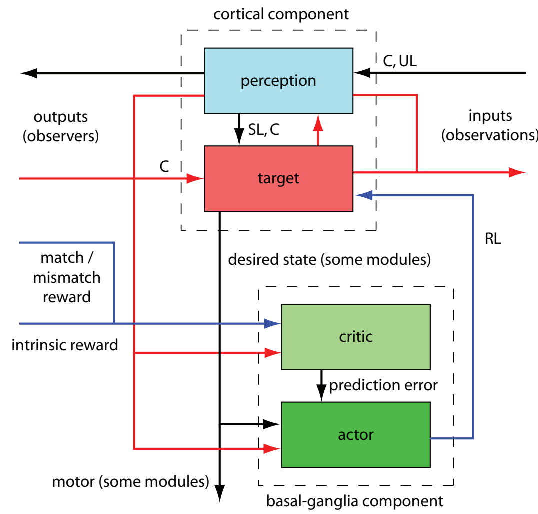

Figure 1: Cortical and basal-ganglia modules. Legend: UL = unsupervised learning, SL = sequential learning, RL = reinforcement learning, C = compute new state. Black arrows represent primary or "driving" input; red arrows represent contextual or modulatory input; blue arrows indicate reward signals.

serving a primarily sensory function (roughly, corresponding to posterior cortex), the role of the target component is to aid recognition of ambiguous input; perception is biased by the expectation of future input provided by the target. The full role of the target in modules that relate to action, broadly construed (corresponding to anterior cortex), is more complex, and is detailed below, but can be taken to represent action selection at an abstract level.

The perception-component representation of one module serves as input to downstream modules. Different modules have different integration windows over which they observe their inputs (generally coming from multiple upstream modules) in order to produce a state representation. In consequence, an entire sequence of inputs can be represented atomically as a single state in a downstream module. In turn, target state serves as context for upstream modules: perceptions flow downstream; predictions flow upstream.

Figure 1 also contains *basal ganglia modules*, which perform reinforcement learning: they learn associations between cortical states and appropriate actions, where "appropriate" corresponds to generating reward via the midbrain

[1] Both perception and target states are represented as real vectors of the same length, abstracted from any particular interpretation that may additionally be present. Target representations may be superposed (vector addition); this can be taken to represent a distribution over predicted perception states.

dopamine system in real brains. In this model, the basal ganglia modules operate by training the cortical-module target components, also providing random exploratory inputs at times. Some cortical modules are "primary motor" modules and are connected directly to effectors. In those, target representation is fixed; target state directly specifies motor output. Thus, prediction becomes command. When inappropriate commands are generated in a given context, the associated basal-ganglia modules train the target to select a different state in the future. Over time, appropriate actions are learned in varied contexts.

Goal bootstrapping. Primary rewards represent built in targets such as sleep, satiation, etc.; reinforcement learning enables the system's own predictions to define novel rewards at a more abstract level. The system then bootstraps itself into automatically representing a goal hierarchy. When a given "action"-oriented module happens to perceive a current state that matches its predicted target, a reward is generated for lower-level modules that were responsible for generating the actions that led to that perception. This rule has the effect of transforming the role of target in action modules into action selection at varying levels of abstraction. Thus, prediction becomes command, at an abstract level: a module can effectively select a "goal" state based on its current context, trained by basal-ganglia modules, and trust that other modules will act so as to achieve that state.

Learning-algorithm selection. The basic model outlined above is agnostic as to the specific unsupervised, sequential, and reinforcement learning algorithms used. In our computational experiments so far we have used many standard algorithms, including: self-organizing maps, k-means clustering, and series of nets of winner-take-all clusters for unsupervised learning; linear associators for sequential learning; and actor-critic versions of temporal-difference learning for reinforcement learning. These are all intended as tests of how these mechanisms may behave when embedded in an appropriate large-scale brain architecture.

Typical representations used in reinforcement learning are non-hierarchical, and thus do not richly represent relations among concepts. Recent work on "hierarchical reinforcement learning" attempts to address this shortcoming by applying reinforcement methods to hierarchically structured domains (Precup and Sutton 1998; Andre and Russell 2002). The work we describe here extends and elaborates this direction via its focus on the learning of the hierarchical representations themselves.

Computational experiments. We have used the above architecture to teach an AIBO robot dog to stand up from arbitrary initial postures (Hearn and Granger 2007). This is a standard task of known difficulty, which is far from trivial given the size of the search space. What the system learns is to perform a sequence of small motions that eventually leads to a standing posture, from any starting posture, such that each individual motion is within the physical capability of the servomotors from that configuration.

This system embodies the ideas outlined above, using a hierarchy of six cortical modules. Primary sensory information from the robot is relayed wirelessly to the computer running the simulation, and drives the state of the S1 module. S1 states are clustered into more abstract unified representations in the downstream module S2. Primary motor modules drive the front and rear legs; target states are relayed wirelessly to the robot servos. Downstream of all of these is a "posture" module that represents a combination of sensory and motor state. Finally, a further downstream "planning" module is hard-coded to represent a desired standing posture. As the system runs, it first learns abstract representations of posture. When, by chance, the overall posture more closely matches a standing posture, the current target mapping in the posture module is rewarded. In turn, when the current posture matches the target posture, whatever that may be, the front and rear leg modules reward their current target mapping—it was that mapping that led to the motor acts making the posture perception match posture target.

Eventually, the system learns to select a series of high-level postural targets that will result in a standing posture; the front and rear legs learn the appropriate steps to transition from one posture to the next. The significance of this application is that a general-purpose architecture for learning hierarchical state representations and behaviors was used.

Work is in progress on a broad array of applications involving learning to produce motor outputs that reproduce aspects of perceived inputs. In each case, the goal is to enable learning of appropriate representations, and to match motor representations to sensory representations. The system is intended to model how humans and other animals perform these tasks, and, it is hoped, to identify a general system for achieving perceptual-motor learning from example.

Conclusion. Traditional approaches to building intelligent systems have used constraints from psychology and behavior; it is hoped that adding serious constraints from the architecture and operating rules of the only existing intelligent systems (brains) may aid in the search for intelligent systems that work. We proffer one example in the form of systems that combine a type of reinforcement learning, a la the brain's striatal system, with construction of coherent representations of the space being explored, via models of the brain's thalamocortical system. The integration of these systems results in a coherent model that incorporates trial and error search but becomes increasingly directed by growing semantic knowledge. The model's components, architecture, and integration correspond to the brain's largest regular architectural structure, the cortico-striatal system.

References

Andre, D., and Russell, S. J. 2002. State abstraction for programmable reinforcement learning agents. In *Eighteenth national conference on Artificial intelligence*, 119–125. Menlo Park, CA, USA: American Association for Artificial Intelligence.

Granger, R. H., and Hearn, R. A. 2007. Models of thalamocortical system. *Scholarpedia* 2(11):1796.

Granger, R. H. 2005. Brain circuit implementation: High-precision computation from low-precision components. In Berger, T., and Glanzman, D., eds., *Replacement Parts for the Brain*. MIT Press. 277–294.

Granger, R. 2006. Engines of the brain: the computational instruction set of human cognition. *AI Mag.* 27(2):15–32.

Hearn, R. A., and Granger, R. H. 2007. Basal-ganglia-inspired hierarchical reinforcement learning in an AIBO robot. Demonstration at Neural Information and Processing Systems 2007.

Precup, D., and Sutton, R. S. 1998. Multi-time models for temporally abstract planning. In *Advances in Neural Information Processing Systems 10*. MIT Press.

Schultz, W. 2000. Multiple reward signals in the brain. *Nat. Rev. Neurosci.* 1(3):199–207.