A Cognitive Architecture for Instrumental Learning in Smart Agents

Wandemberg Gibaut[1], Ricardo R. Gudwin[1]
[1]Control and Automation Department (DCA), FEEC, UNICAMP

Introduction and Hypothesis:

The research field of Cognitive Architectures relates to getting inspiration from the different models of the human brain / human mind coming from neuroscience and neuropsychology and building smart artificial agents able to reproduce the cognitive capabilities found in men and animals. Among these capabilities, instrumental learning is a kind of learning where an agent learns from observing the results of its own actions on the environment. This work proposes a Cognitive Architecture, inspired in neuroscience findings, for the control of intelligent agents able to learn autonomously, based on the results of its own actions on the environment. To accomplish this, the Architecture counts on different techniques, such as Reinforcement Learning [1], Neural Networks and Episodic Memory [2], which allows the system to generate expectations for its actions and to retrieve past experiences. To validate the architecture, we developed a performance test, using the virtual environment of the game Minecraft, available through the Mälmo Platform [3].

Objective:

The objective of the present work was to propose a Cognitive Architecture, grounded on findings from neuroscience (see Figure 1), for the smart control of an artificial creature in a computer game with a high degree of freedom on its actions. The chosen platform, Minecraft, is a sophisticated 3D environment where, due to a huge state space, conventional control techniques are not usually suitable. In particular, we sought to analyze how models of Episodic Memory may help in a cognitive agent's learning and decision making processes in a way that it can explore and learn about it's surroundings, showing behaviors that make good use of the available information seeking to maximize its performance indicators. To implement that, we used unsupervised learning techniques (Reinforcement Learning, for example) and Neural Networks.

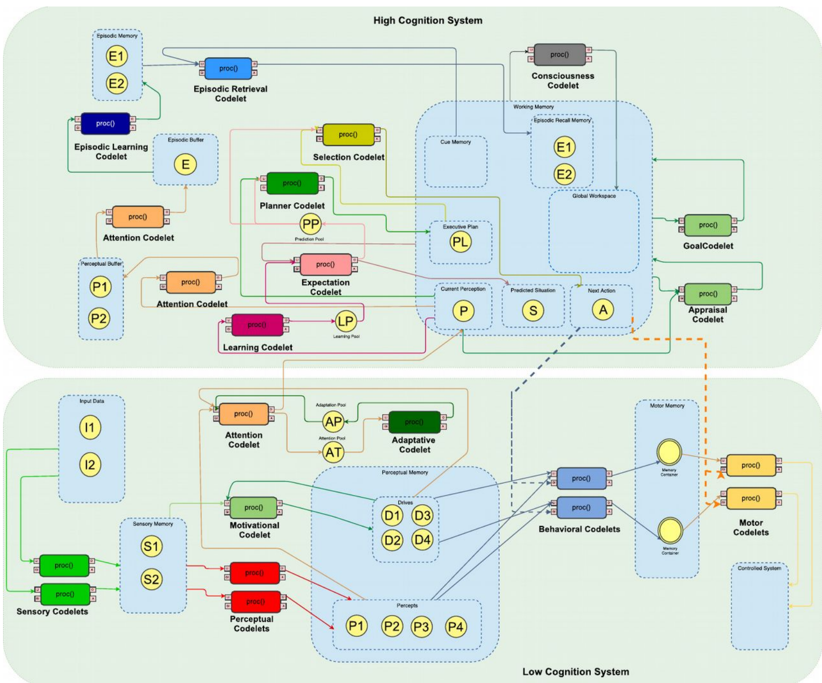

Figure 1 –The proposed Cognitive Architecture.

Methods:

We took as methodology the exploration of the space state and checking the agent's learning through measures of quality and speed of adaptation to the found conditions in the environment. We used five types of controllers:

- Simple Look-up table Reinforcement Learning with ε-greedy policy: representing a classic method of decision-making process with guaranteed convergence. The results obtained in the other proposals were compared primarily to this one.
- Cognitive controller purely based on Expectations: controller model whose decisions are simply based on a reinforcing structure (with ε-greedy policy) with a neural network as an approximation.
- Cognitive Controller with Episodic Memory: similar to the previous one, but using Memory to improve the decision-making process.
- Cognitive Controller with Exploration: the same as the previous one, but with a given chance. The agent decides to explore new situations in detriment of what it already knows. This controller opts for exploration in 100% of the cases.
- Complete Cognitive Controller: In addition to the modules presented in the previous sections, this version can save and partially optimize sequences of actions that lead to positive reward points. As a way to achieve a compromise between discovery and benefit, the agent chooses to follow the best plan so far in 50% of cases and to explore new possibilities in the remaining ones.

To validate the full controller and the Architecture, and measure its performance compared with other proposals, we defined some metrics to get an overview of the agent's learning with each controller. They are:

- Total number of victories per round: Metric that indicates how many times the agent was able to reach the previously established mission goal. Although it seems redundant with the previous item, due to the fact that the mission can be terminated by time, this value evaluates the ability to correctly exploit available space for positive rewards.
- Average Victory Execution Time: Represents, in seconds, the total time it takes for the agent to reach the mission goal.
- Number of commands to victory per round: Because programs have different structures and complexities, execution time might not always fully reflect their capabilities. Thus, this measure shows the number of commands needed to achieve the goal, when it was successfully completed.

Relevance:
This artificial mind project wraps together many know techniques in the research areas of Neurotechnology and Artificial Intelligence . This makes possible to the controllers to obtain efficient results in an acceptable runtime, which allows the Architecture to be used in a wide range of applications, including those whom may benefit from autonomous learning, as drones, self-driving cars and brain-computer interfaces.
The proposed Architecture is also in accordance with other works present in the literature, aligning several theories in an unique framework and shows a satisfying implementation, being potentially competitive with the existing architectures. So, it is appropriate to say that the Architecture is suitable to be used to design controllers for smart systems.

References:
[1] SUTTON, R. S.; BARTO, A. G. Introduction to Reinforcement Learning. MIT Press, 1998;
[2] TULVING, E., & THOMSON, D. M. (1973). Encoding specificity and retrieval processes in episodic memory. *Psychological Review, 80*(5), 352-373. doi:10.1037/h0020071
[3] JOHNSON, M. et al., The malmo platform for artificial intelligence experimentation. In: Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, p. 4246–424, 2016.