HTMRL: Biologically Plausible Reinforcement Learning with Hierarchical Temporal Memory*

Jakob Struye[0000−0003−1360−7672], Kevin Mets[0000−0002−4812−4841], and Steven Latré[0000−0003−0351−1714]

University of Antwerp - imec
IDLab - Department of Mathematics and Computer Science
Sint-Pietersvliet 7, 2000 Antwerp, Belgium
firstname.lastname@uantwerpen.be

Objective Hierarchical Temporal Memory (HTM) has been shown to adapt well to changing input patterns, making it a natural building block for a Reinforcement Learning (RL) agent that adapts to evolving tasks quickly. In this paper, we design, implement and experimentally evaluate HTMRL: the first such agent, built with nothing but HTM.

Background In RL, an agent learns how to behave in an environment [2]. This is usually modelled as a Markov Decision Process (MDP): the 4-tuple

$$(S, A, P, R) \tag{1}$$

where S is the set of states, A is the set of actions, P models which state will be reached after taking an action, and R defines the reward after such a transition. The main goal of an agent is then to define a policy π, deciding which action to take in each state.

HTM is a computational model of the human neocortex, originally proposed by Jeff Hawkins [1]. One of the core aspects of HTM is its strict biological plausibility, meaning that any feature that could not plausibly be implemented in the neocortex will not be allowed in the HTM model. All data within the model is encoded as Sparse Distributed Representations (SDRs), binary data structures with only a limited number of *enabled bits* (i.e., with value 1). One of the main components of HTM is the spatial pooler, which reorganises incoming SDRs to produce outputs of always the same size and sparsity, and to ensure the available capacity is optimally utilised. To achieve this, the spatial pooler maintains *synapses* between some pairs of input and output bits. These synapses can carry a 1 signal from input bit to output bit, with output bits receiving relatively many signals adopting this 1 value. Synapses carrying such signals to these activated output bits grow stronger while others grow weaker, eventually becoming unable to carry a signal, ensuring that similar inputs will lead to similar outputs.

* Supported by the Research Foundation - Flanders (fwo): PhD Fellowship 1SB0719N.

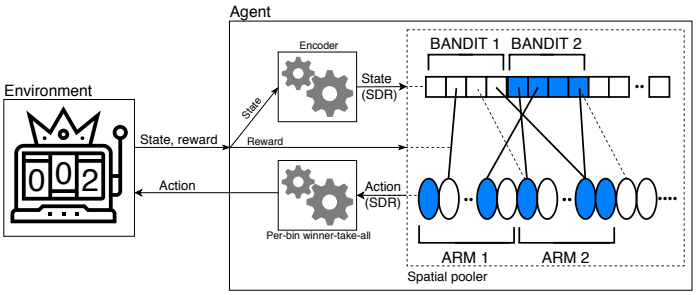

Fig. 1. Overview of the HTMRL system, with coloured bits/cells being enabled/active, illustrated with a contextual bandit environment where the state is the bandit's index.

HTMRL Our proposed HTMRL algorithm, summarised in Figure 1, uses only a spatial pooler, with minimal modifications. HTMRL converts any input into an SDR of, by default, 2048 bits. To map this to one of $|A|$ actions, we simply subdivide the SDR into $|A|$ equally sized *bins*, selecting the bin containing the most 1 bits as the action to take. Synapse growth is then scaled with the reward signal. As such, actions leading to poor rewards become less likely to be selected on future similar inputs, and vice versa. HTMRL can explore rarely selected actions through the *boosting* mechanism also present in core HTM theory, which amplifies incoming signals of rarely activated output bits.

Experiments and evaluation We evaluate our HTMRL implementation in an environment with many multi-armed bandits, each giving a positive reward on only a single arm. One randomly selected bandit is presented to the algorithm at every step. With 4-armed bandits, the algorithm could learn over 1000 bandits before becoming prohibitively slow in our implementation. With a fixed set of 20 bandits, HTMRL could learn up to 1024 arms per bandit, which is the maximum number of actions representable in its base configuration. To evaluate performance in changing environments, we experiment with a bandit whose arms' rewards are sampled from normal distributions. HTMRL reaches near-optimal performance in a number of steps similar to an ϵ-greedy learner. The distributions are then randomly reinitialised, after which HTMRL reaches near-optimal performance just as fast as before, while ϵ-greedy learners struggle to adapt. We also show that by *shuffling* the arms instead of fully reinitialising them, HTMRL adapts even faster, meaning it can leverage its knowledge of previous phases, making it a promising approach for Meta-RL applications.

References

1. Hawkins, J., Ahmad, S., Purdy, S., Lavin, A.: Biological and machine intelligence (2016), https://numenta.com/resources/biological-and-machine-intelligence/
2. Sutton, R.A., Barto, A.G.: Reinforcement Learning: An Introduction. MIT press, 2 edn. (2018)