Mimicking the Bat Brain

Master MVA - Reinforcement Learning

Dorian Desblancs

February 10, 2021

école normale supérieure paris—saclay

université PARIS-SACLAY

Outline

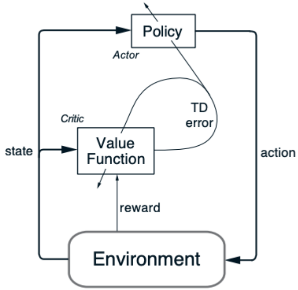
Background Information

Temporal Difference Learning

- Idea: update value function based on predicted and actual reward of next step
- TD Target: reward + estimate of the return in the next state
  - $V(s_t) = R_{t+1} + \gamma V(s_{t+1})$
- TD error: $\delta_t = R_{t+1} + \gamma V(s_{t+1}) - V(s_t)$
- $TD(0)$ algorithm: take action a, observe R, s_{t+1}, update value function using:
  - $V(s_t) = V(s_t) + \alpha[R_{t+1} + \gamma V(s_{t+1}) - V(s_t)]$

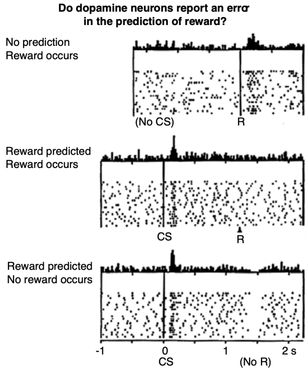
Actor-Critic Architecture

- TD methods with separate memory structure to represent the policy independently of the value function



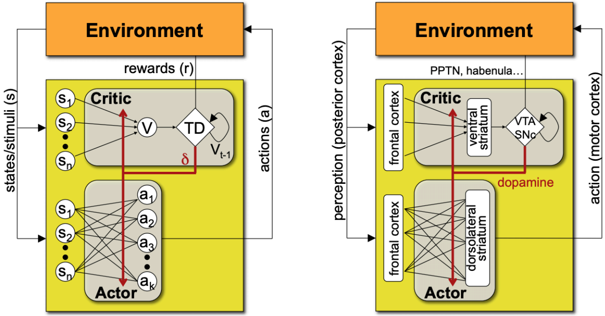
The Actor-Critic Architecture

Temporal-Difference Learning in Neuroscience



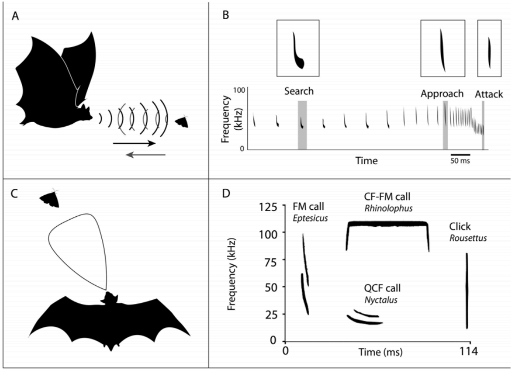
Reward Prediction Error in the Brain

The Actor-Critic Architecture in Neuroscience



The Actor-Critic Architecture in the Brain

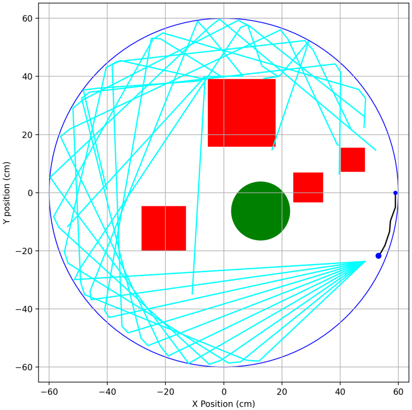
Bat Echolocation



The Echolocation Mechanism in Bats

Methodology

The Environment



Experimental Setup

The Foster Model (Part 1)

- Hippocampal Place Cell activations:

$$\text{for } i \in 1, ..., N, f_i(p) = exp\left(-\frac{||p - s_i||^2}{2\sigma^2}\right)$$

The Foster Model (Part 2)

- The Critic: vector w of length N
- Value function: $C(p) = \sum_i w_i f_i(p)$
- TD prediction error: $\delta_t = R_{t+1} + \gamma C(p_{t+1}) - C(p_t)$
- Weight update: $w_i = w_i + \eta \delta_t f_i(p_t)$
- Convergence towards: $C(p_t) = \bar{R}_{t+1} + \gamma C(p_{t+1})$

The Foster Model (Part 3)

- The Actor: matrix z of dimension $8 \times N$ or $16 \times N$
- Action vector: $a_j(p) = \sum_i z_{ji} f_i(p)$
- Action probabilities: $P_j = \frac{exp(2a_j)}{\sum_k exp(2a_k)}$
- Weight update: $z_{ji} = z_{ji} + \eta \delta_t f_i(p_t)$ (only for selected action row)

The Foster Model (Algorithm)

At each step:

- Determine action from actor probabilities
- Determine reward
- Compute TD error
- Update critic weights
- Update actor weights

The Foster Model (Add-on)

- The Coordinate System: separate networks for X and Y coordinates
- Two vectors of length N

$$X(p) = \sum_i w_i^X f_i(p)$$

$$Y(p) = \sum_i w_i^Y f_i(p)$$

- Weight update:

$$w_i^X = w_i^X + \eta(\Delta x_t - (X(p_{t+1}) - X(p_t))) \sum_k^t \lambda^{t-k} f_i(p_k)$$

$$w_i^Y = w_i^Y + \eta(\Delta y_t - (Y(p_{t+1}) - Y(p_t))) \sum_k^t \lambda^{t-k} f_i(p_k)$$

The Foster Model (Add-on Algorithm)

When coordinate system is chosen:

- Compute $X(p)$ and $Y(p)$
- Use direction $[d_X, d_Y] = [X' - X(p), Y' - Y(p)]$ to determine next move (X' and Y' are goal coordinates)
- Compute TD error
- Update Coordinate system weights
- Update actor ($z_{ji} = z_{ji} + \eta\delta_t$ only here)

Incorporating Echolocation

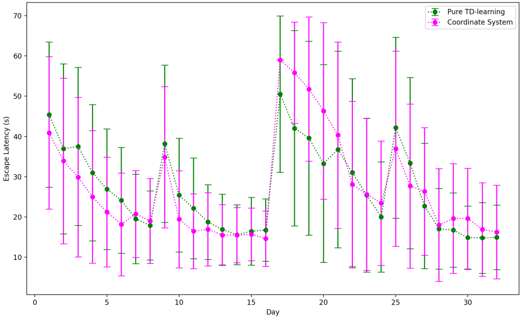
- Idea: at the beginning of each time step, generate sound waves
- Re-balance action probabilities according to the waves that are reflected upon agent

$\Rightarrow$ Ex: goal platform reflects wave on agent implies $P(\neg\text{direction}) = 0$

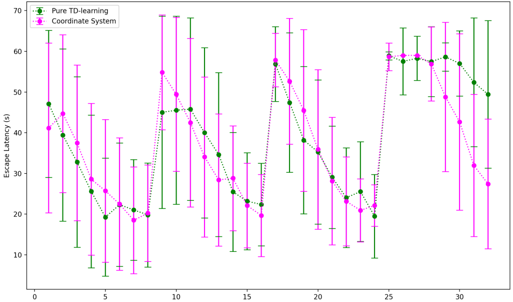
$\Rightarrow$ Ex: obstacle or wall reflects wave on agent implies $P(\text{direction}) = 0$

Results

Pure TD Learning vs Coordinate System Add-On

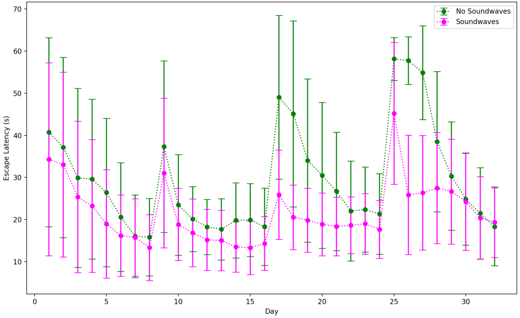


Regular Watermaze

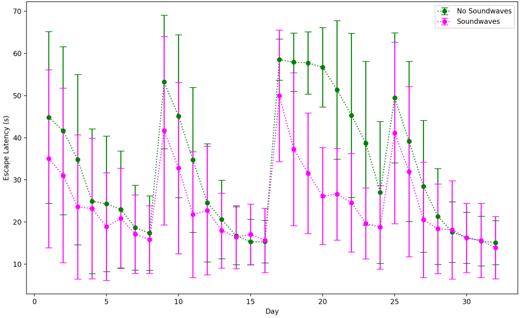


Obtacle Watermaze

Pure TD Learning and Echolocation

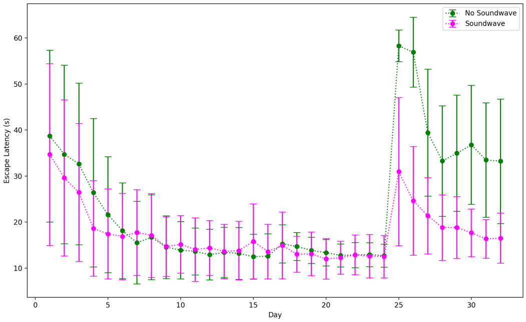


Regular Watermaze

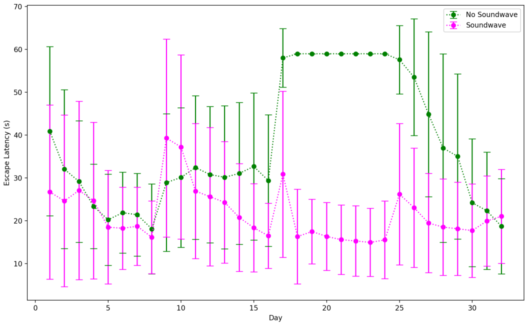


Obtacle Watermaze

Coordinate System and Echolocation



Regular Watermaze



Obtacle Watermaze

Wrap-up

Significance and Future Work

Significance:

- Learning speed-up that could be applied in the real world
- New frontier: incorporating perception in RL decision-making
- Baseline for more thorough understanding of mammalian brains

Future Work:

- More complex environment
- Use signal processing research for sound wave classification
- More complex Actor-Critic architectures

Thank You

Questions?

Bibliography

- David J Foster, Richard GM Morris, and Peter Dayan. A model of hippocampally dependent navigation, using the tem- poral difference learning rule. Hippocampus, 10(1):1–16, 2000.
- James R Hinman, Holger Dannenberg, Andrew S Alexander, and Michael E Hasselmo. Neural mechanisms of navigation involving interactions of cortical and subcortical structures. Journal of neurophysiology, 119(6):2007–2029, 2018.
- P Read Montague, Peter Dayan, and Terrence J Sejnowski. A framework for mesencephalic dopamine systems based on predictive hebbian learning. Journal of neuroscience, 16(5):1936–1947, 1996.
- R.G.M.Morris.Morriswatermaze.Scholarpedia,3(8):6315, 2008. revision 91529.
- Yael Niv. Reinforcement learning in the brain. Journal of Mathematical Psychology, 53(3):139–154, 2009.

Bibliography

- Wolfram Schultz, Peter Dayan, and P Read Montague. A neural substrate of prediction and reward. Science, 275(5306):1593–1599, 1997.
- Richard S Sutton and Andrew G Barto. Reinforcement learn- ing: An introduction. MIT press, 2018.
- YossiYovelandStefanGreif.Bats–usingsoundtorevealcog- nition. Field and Laboratory Methods in Animal Cognition: A Comparative Guide, pages 31–59, 2018.