

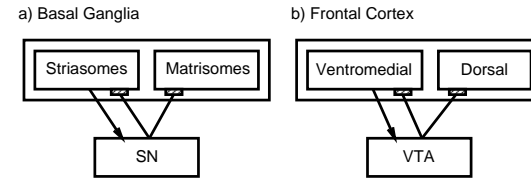
1 Temporally-delayed Learning & Reinforcement

Reinforcement often delayed from the action(s) that lead to it: need to “span the gap”.

Key ideas: We want to predict rewards consistently over time. This process leads us to learn what events are associated with rewards, earlier and earlier back in time.

We use the Temporal Differences (TD) algorithm (Sutton).

2 Reinforcement Biology

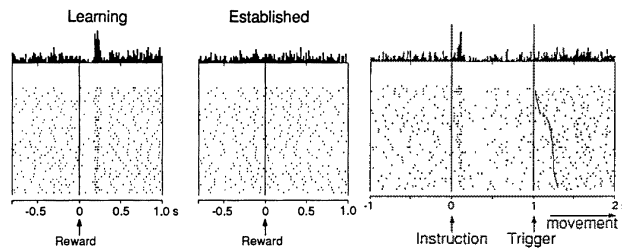


Midbrain dopaminergic (DA) systems modulate cortex & basal ganglia: Substantia Nigra (SN) & Ventral-Tegmental Area (VTA).

SN & VTA are controlled by other cortical/BG areas.

These other areas are like an “Adaptive Critic” (AC), which evaluates stimuli & actions for their rewarding value.

3 Recordings from Actual Neurons



VTA firing moves from responding to reward to *anticipating* it at the instruction.

4 The Equations

Value function, sum of discounted future rewards:

$$V(t) = \langle \gamma^0 r(t) + \gamma^1 r(t+1) + \gamma^2 r(t+2) \dots \rangle \quad (1)$$

Recursive definition:

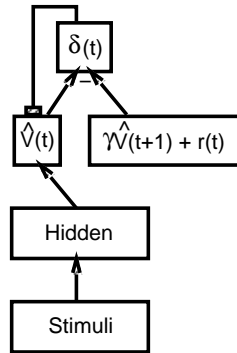
$$V(t) = \langle r(t) + \gamma V(t+1) \rangle \quad (2)$$

Error in predicted reward:

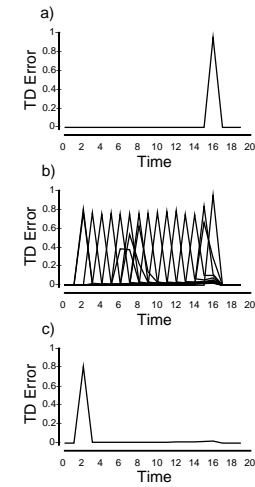
$$\delta(t) = (r(t) + \gamma \hat{V}(t+1)) - \hat{V}(t) \quad (3)$$

5

Network Implementation

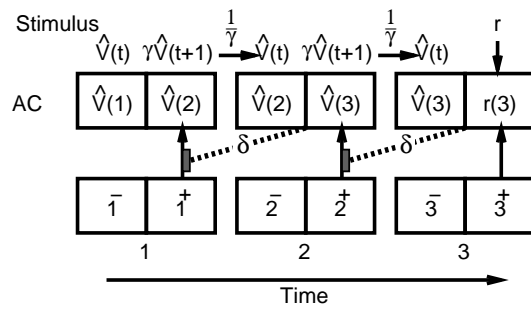


6

Model: CS at $t=2$, US at $t=16$ 

7

Phase-based Implementation



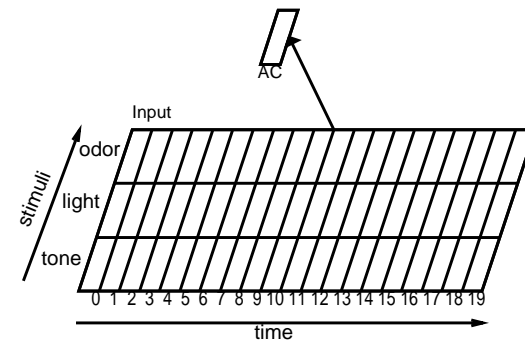
Plus phase: AC settles via weights = expected reward at $t+1$ (or r).

Minus phase: AC clamped to previous plus phase value (0 at start).

Learning goes "backwards in time" to affect previous time step..

8

Exploration



CSC input (Complete Serial Compound): unique unit for each stimulus at each time point.

Good for demonstrating ideas (more realistic versions in Ch 9).

9

Combining Error-driven + Hebbian

Get benefits of both: Solve tasks, learn systematic representations, generalize to new stimuli.

What's left?... Time!

Currently: networks learn *immediate* consequence of a given input.

- What if current input only makes sense as part of a *sequence* of inputs (e.g., language, social interactions)? We represent the *context*, not just the current input.
- What if the consequence of this input comes *later* (e.g., school/work, life)? Reinforcement learning tries to predict rewards consistently over time, learns events earlier in time associated with later rewards.