Reinforcement Learning

Quentin Huys

Division of Psychiatry and Max Planck UCL Centre for Computational Psychiatry and Ageing Research, UCL
Complex Depression, Anxiety and Trauma Service, Camden and Islington NHS Foundation Trust

Systems and Theoretical Neuroscience Course, 4/12/18

Setup

Sutton and Barto 2017

Setup

$$\{a_t\} \leftarrow \underset{\{a_t\}}{\operatorname{argmax}} \sum_{t=1}^{\infty} r_t$$

Sutton and Barto 2017

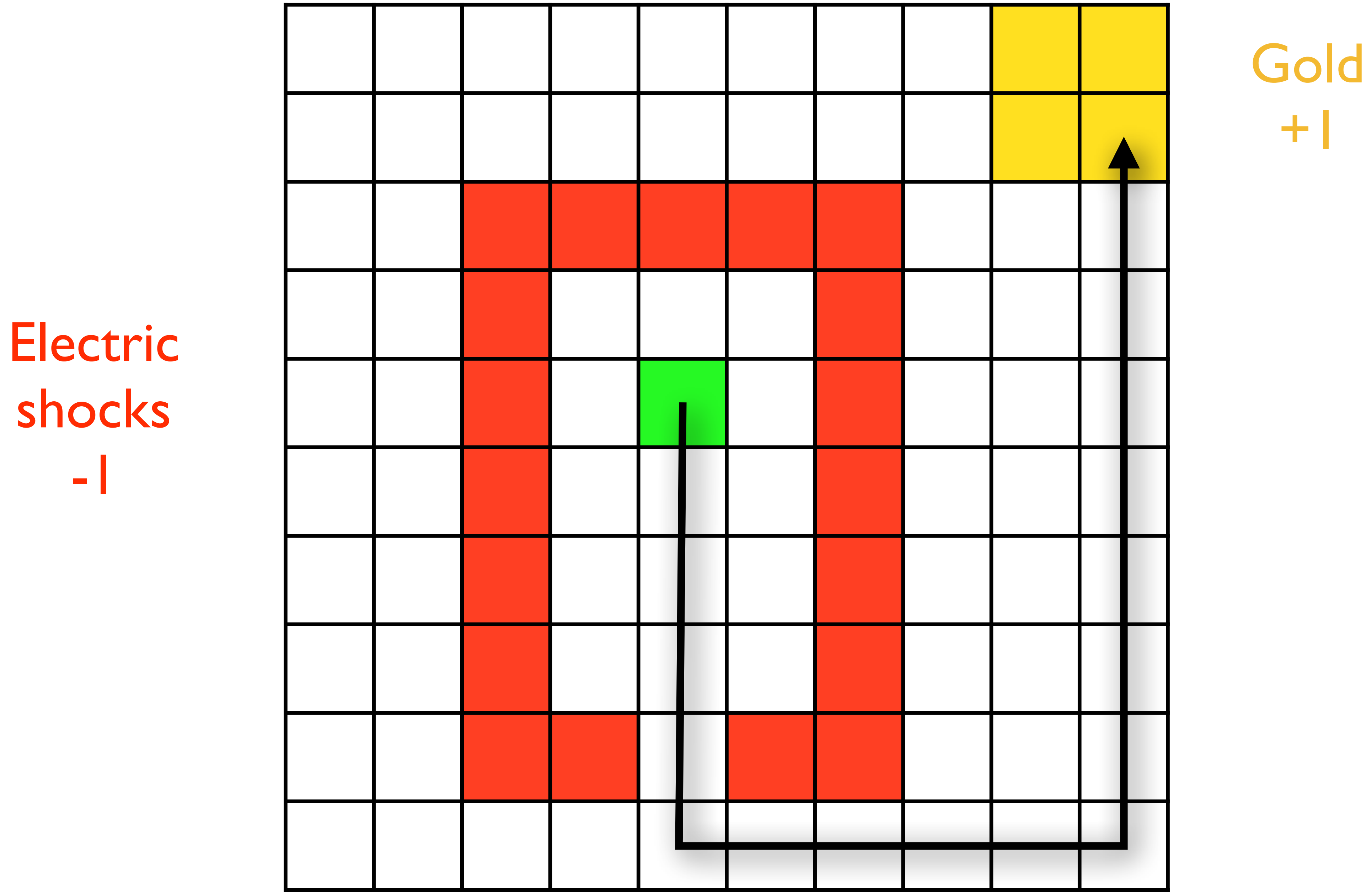
Setup

$$\{a_t\} \leftarrow \underset{\{a_t\}}{\operatorname{argmax}} \sum_{t=1}^{\infty} r_t$$

Minimizing Loss = Maximizing Reward

Sutton and Barto 2017

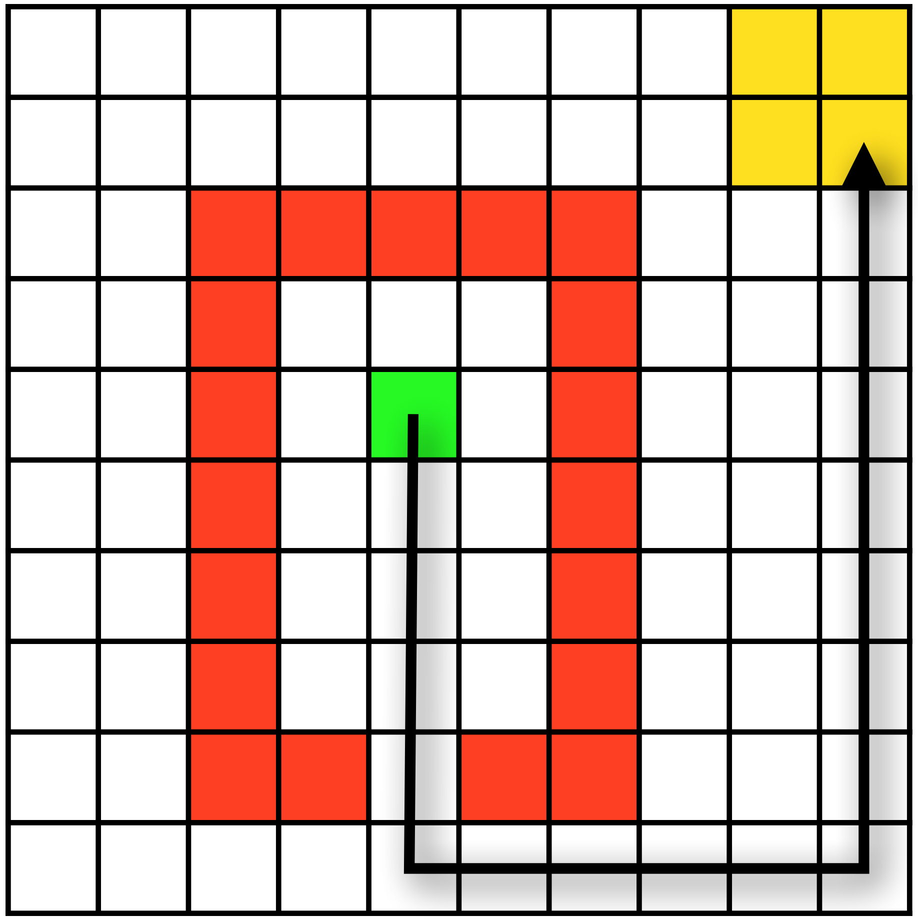
State space



A Markov Decision Problem

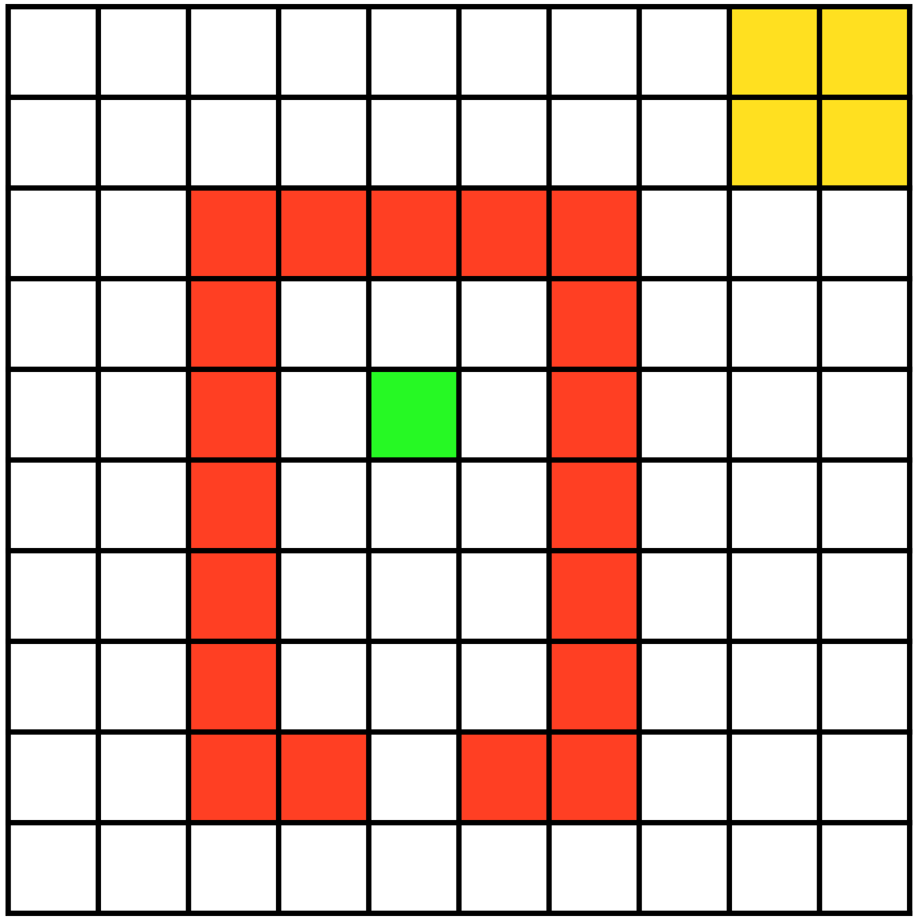
$$\boxed{s_t \in \mathcal{S}}$$

$$\begin{aligned} a_t &\in \mathcal{A} \\ \mathcal{T}^a_{ss'} &= p(s_{t+1}|s_t, a_t) \\ r_t &\sim \mathcal{R}(s_{t+1}, a_t, s_t) \\ \pi(a|s) &= p(a|s) \end{aligned}$$



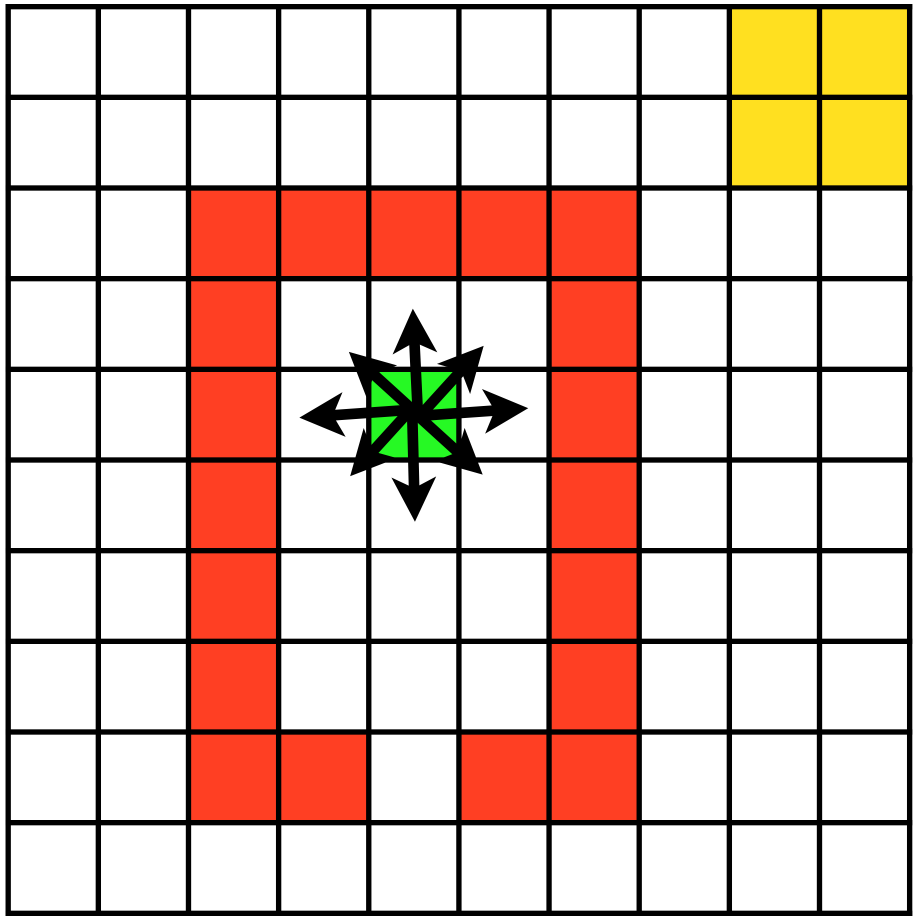
A Markov Decision Problem

$$
\begin{aligned}
s_t &\in \mathcal{S} \\
\boxed{\begin{aligned} a_t &\in \mathcal{A} \\ \mathcal{T}^a_{ss'} &= p(s_{t+1}|s_t, a_t) \end{aligned}} \\
r_t &\sim \mathcal{R}(s_{t+1}, a_t, s_t) \\
\pi(a|s) &= p(a|s)
\end{aligned}
$$

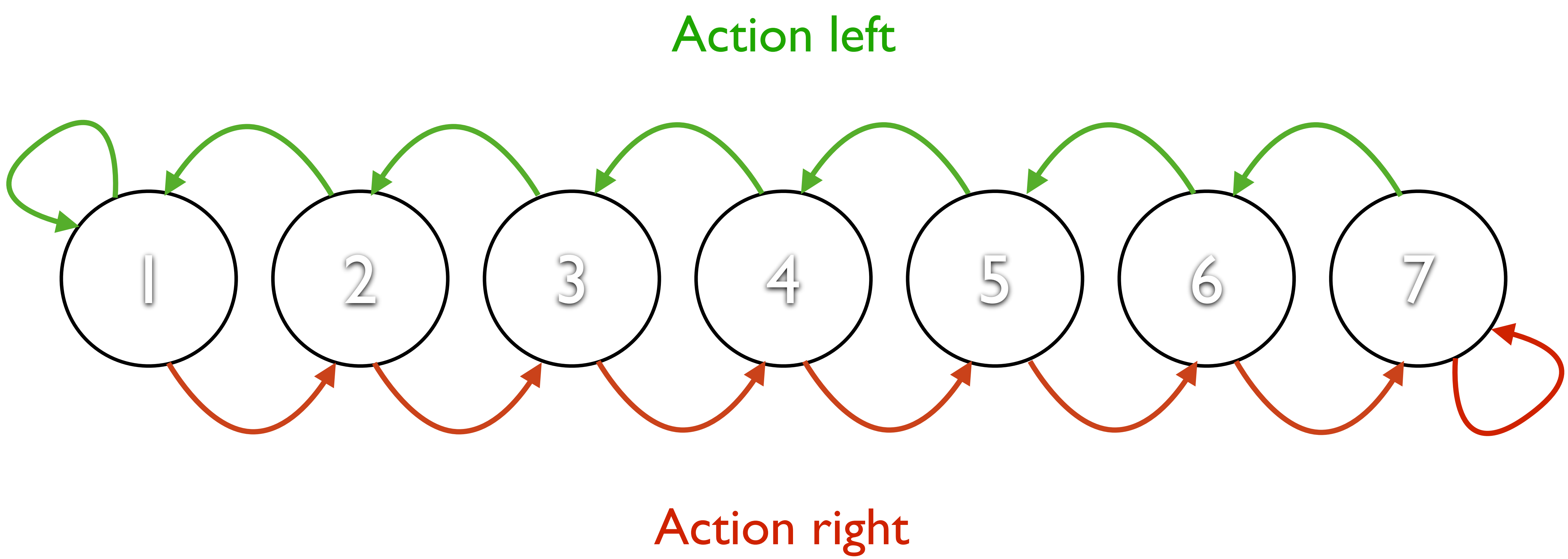


A Markov Decision Problem

$$\begin{aligned} s_t &\in \mathcal{S} \\ \boxed{a_t \in \mathcal{A}} \\ \mathcal{T}^a_{ss'} &= p(s_{t+1}|s_t, a_t) \\ r_t &\sim \mathcal{R}(s_{t+1}, a_t, s_t) \\ \pi(a|s) &= p(a|s) \end{aligned}$$

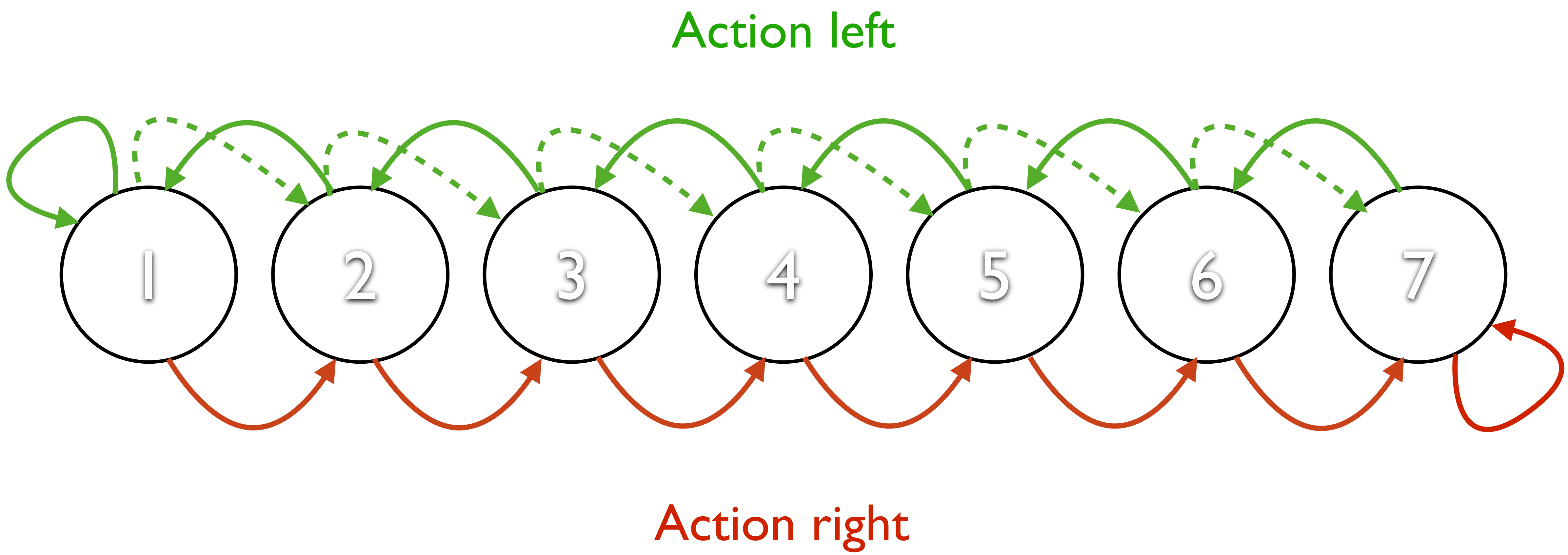


Actions



$$T^{\text{left}} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \qquad T^{\text{right}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

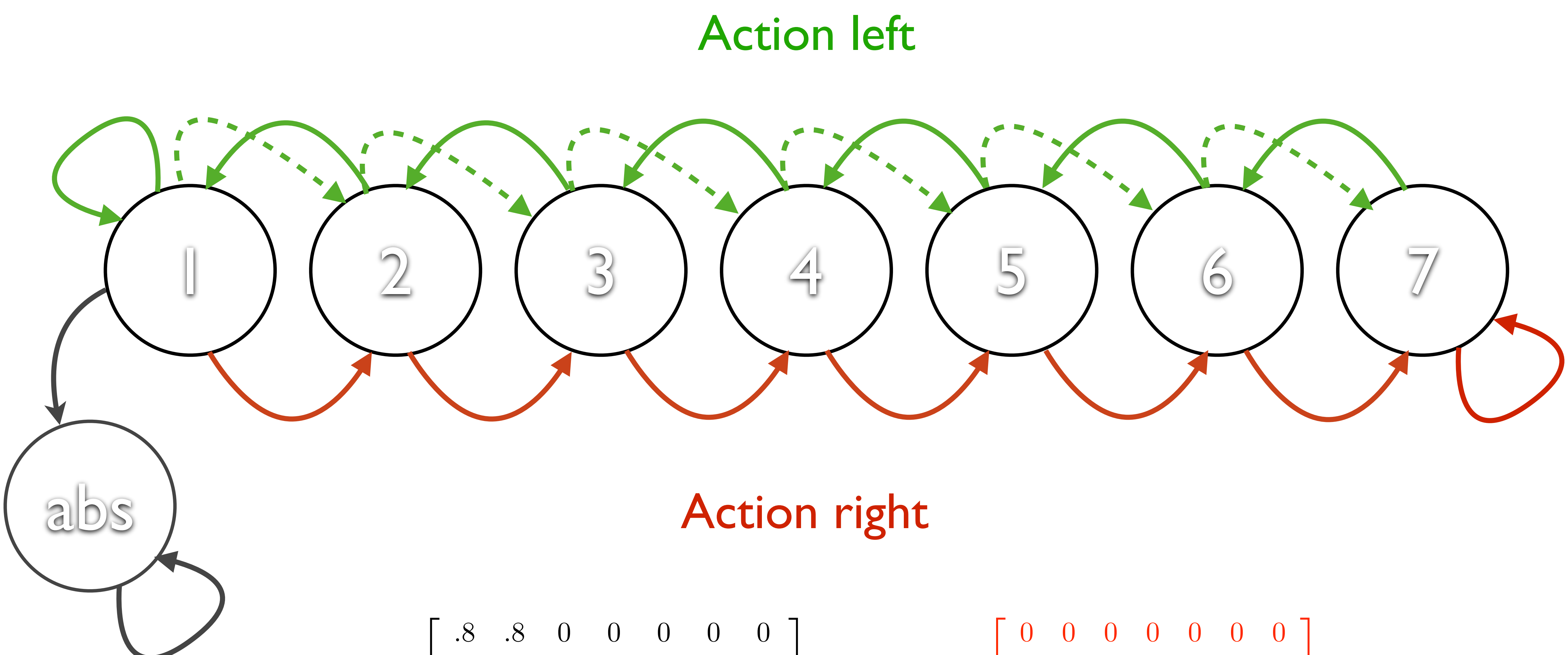
Actions



$$T^{\text{left}} = \begin{bmatrix} .8 & .8 & 0 & 0 & 0 & 0 & 0 \\ .2 & .2 & .8 & 0 & 0 & 0 & 0 \\ 0 & 0 & .2 & .8 & 0 & 0 & 0 \\ 0 & 0 & 0 & .2 & .8 & 0 & 0 \\ 0 & 0 & 0 & 0 & .2 & .8 & 0 \\ 0 & 0 & 0 & 0 & 0 & .2 & .8 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \qquad T^{\text{right}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

Noisy: plants, environments, agent

Actions



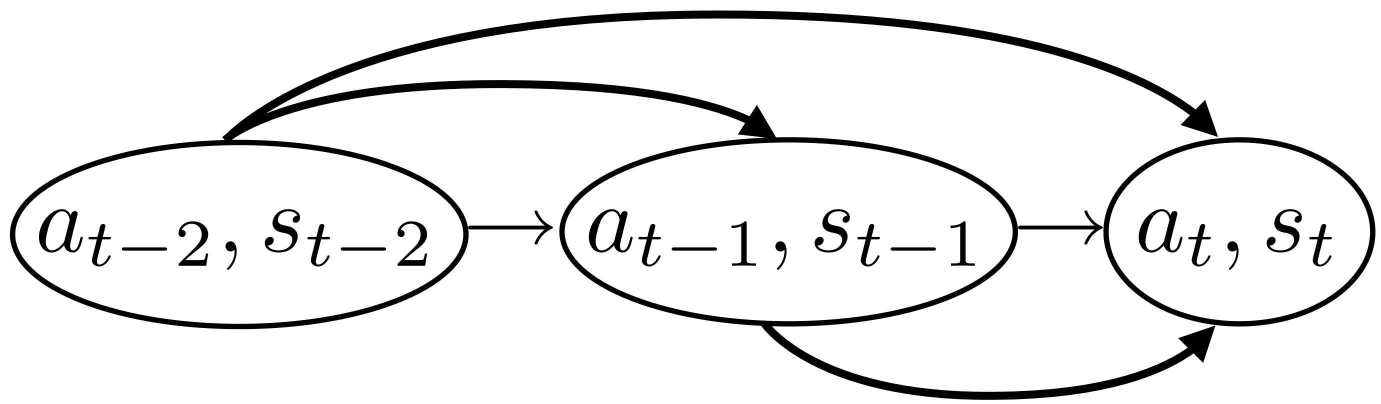
$$T^{\text{left}} = \begin{bmatrix} .8 & .8 & 0 & 0 & 0 & 0 & 0 \\ .2 & .2 & .8 & 0 & 0 & 0 & 0 \\ 0 & 0 & .2 & .8 & 0 & 0 & 0 \\ 0 & 0 & 0 & .2 & .8 & 0 & 0 \\ 0 & 0 & 0 & 0 & .2 & .8 & 0 \\ 0 & 0 & 0 & 0 & 0 & .2 & .8 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \qquad T^{\text{right}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

Noisy: plants, environments, agent

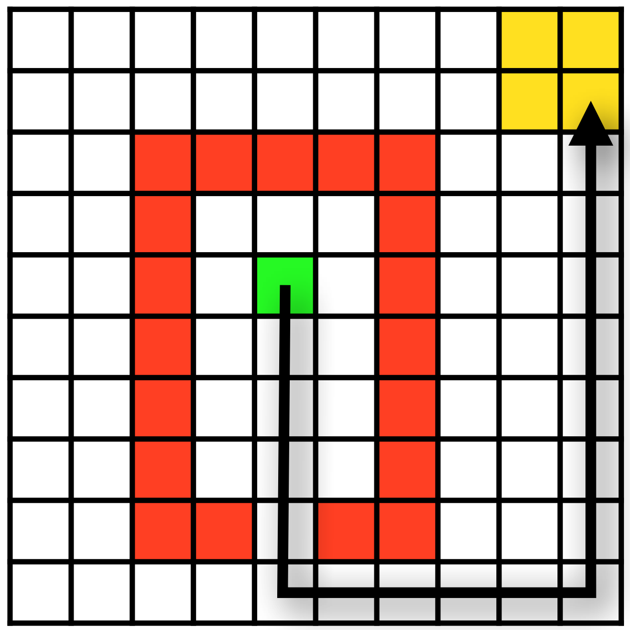
Absorbing state -> max eigenvalue < 1

Markovian dynamics

$$p(s_{t+1}|a_t, s_t, a_{t-1}, s_{t-1}, a_{t-2}, s_{t-2}, \cdots) = p(s_{t+1}|a_t, s_t)$$



Velocity

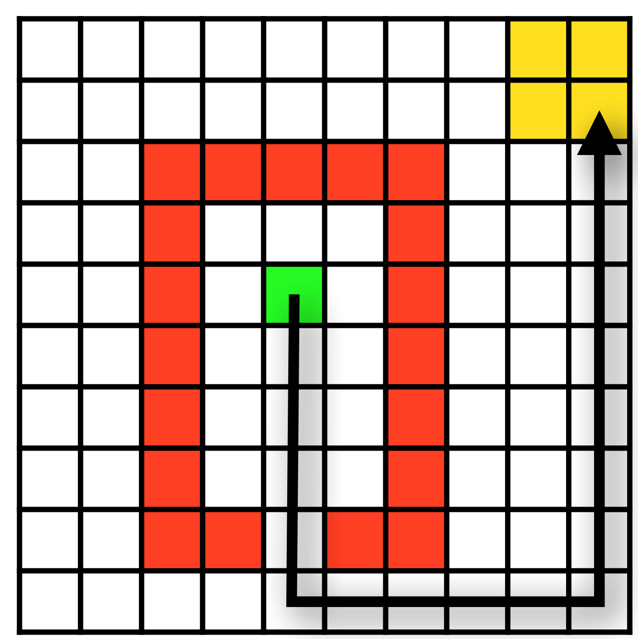


Markovian dynamics

$$p(s_{t+1}|a_t, s_t, a_{t-1}, s_{t-1}, a_{t-2}, s_{t-2}, \cdots) = p(s_{t+1}|a_t, s_t)$$

$$a_{t-2}, s_{t-2} \longrightarrow a_{t-1}, s_{t-1} \longrightarrow a_t, s_t$$

Velocity



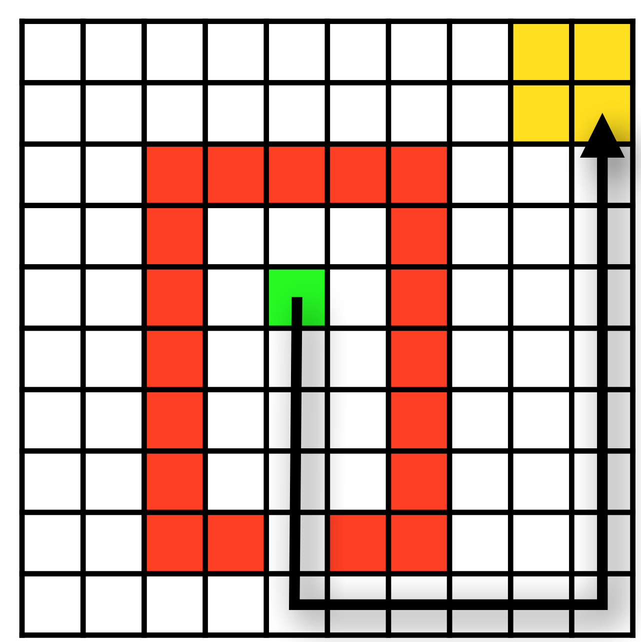
Markovian dynamics

$$p(s_{t+1}|a_t, s_t, a_{t-1}, s_{t-1}, a_{t-2}, s_{t-2}, \cdots) = p(s_{t+1}|a_t, s_t)$$

$$(a_{t-2}, s_{t-2}) \longrightarrow (a_{t-1}, s_{t-1}) \longrightarrow (a_t, s_t)$$

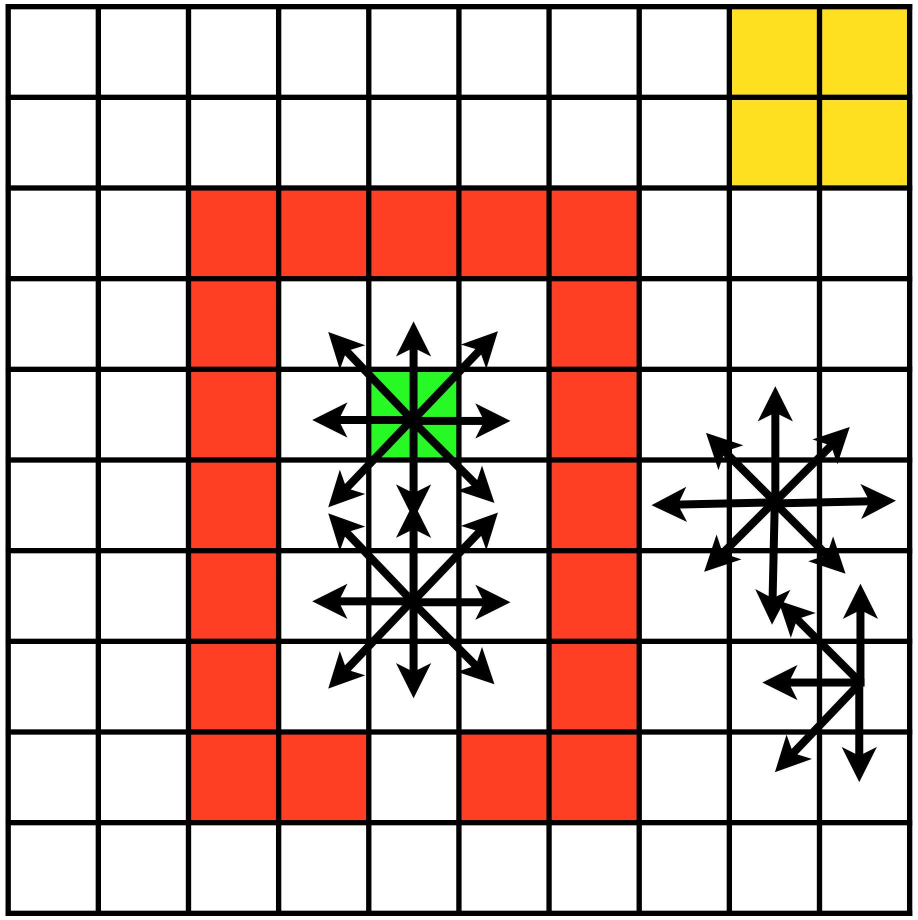
Velocity

$$s' = [\text{position}] \rightarrow s' = \begin{bmatrix} \text{position} \\ \text{velocity} \end{bmatrix}$$



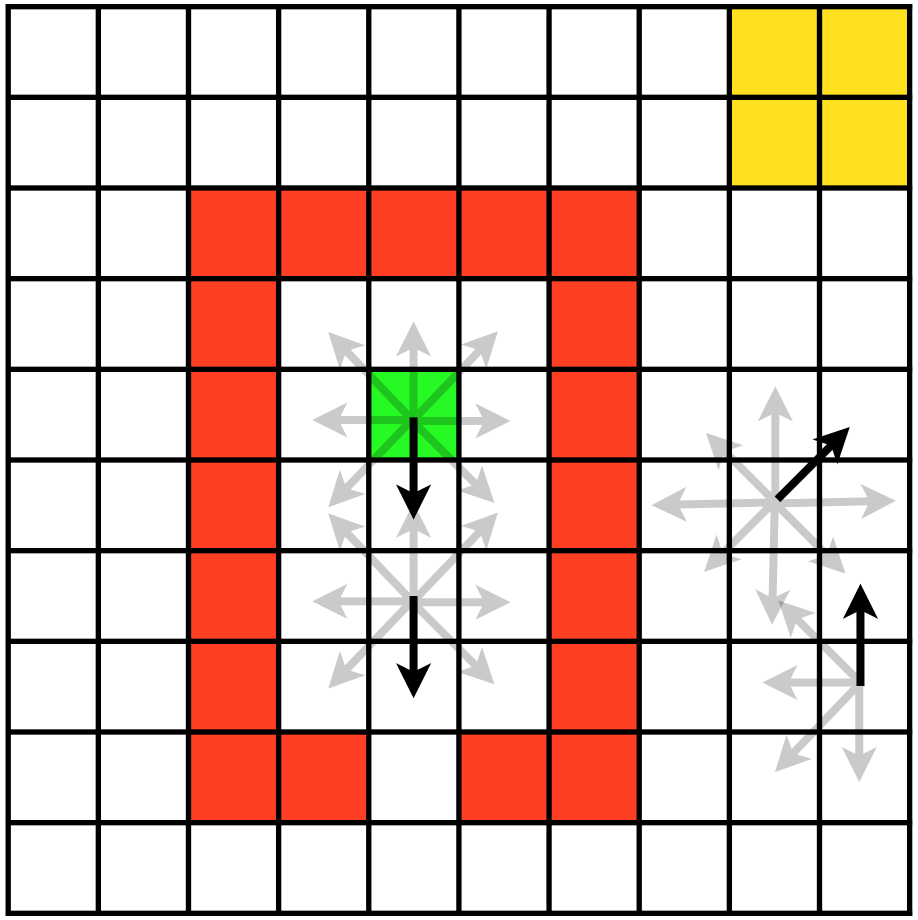
A Markov Decision Problem

$$
\begin{aligned}
s_t &\in \mathcal{S} \\
a_t &\in \mathcal{A} \\
\mathcal{T}^a_{ss'} &= p(s_{t+1}|s_t, a_t) \\
r_t &\sim \mathcal{R}(s_{t+1}, a_t, s_t) \\
\pi(a|s) &= p(a|s)
\end{aligned}
$$



A Markov Decision Problem

$$
\begin{aligned}
s_t &\in \mathcal{S} \\
a_t &\in \mathcal{A} \\
\mathcal{T}^a_{ss'} &= p(s_{t+1}|s_t, a_t) \\
r_t &\sim \mathcal{R}(s_{t+1}, a_t, s_t) \\
\boxed{\pi(a|s) = p(a|s)}
\end{aligned}
$$



A Markov Decision Problem

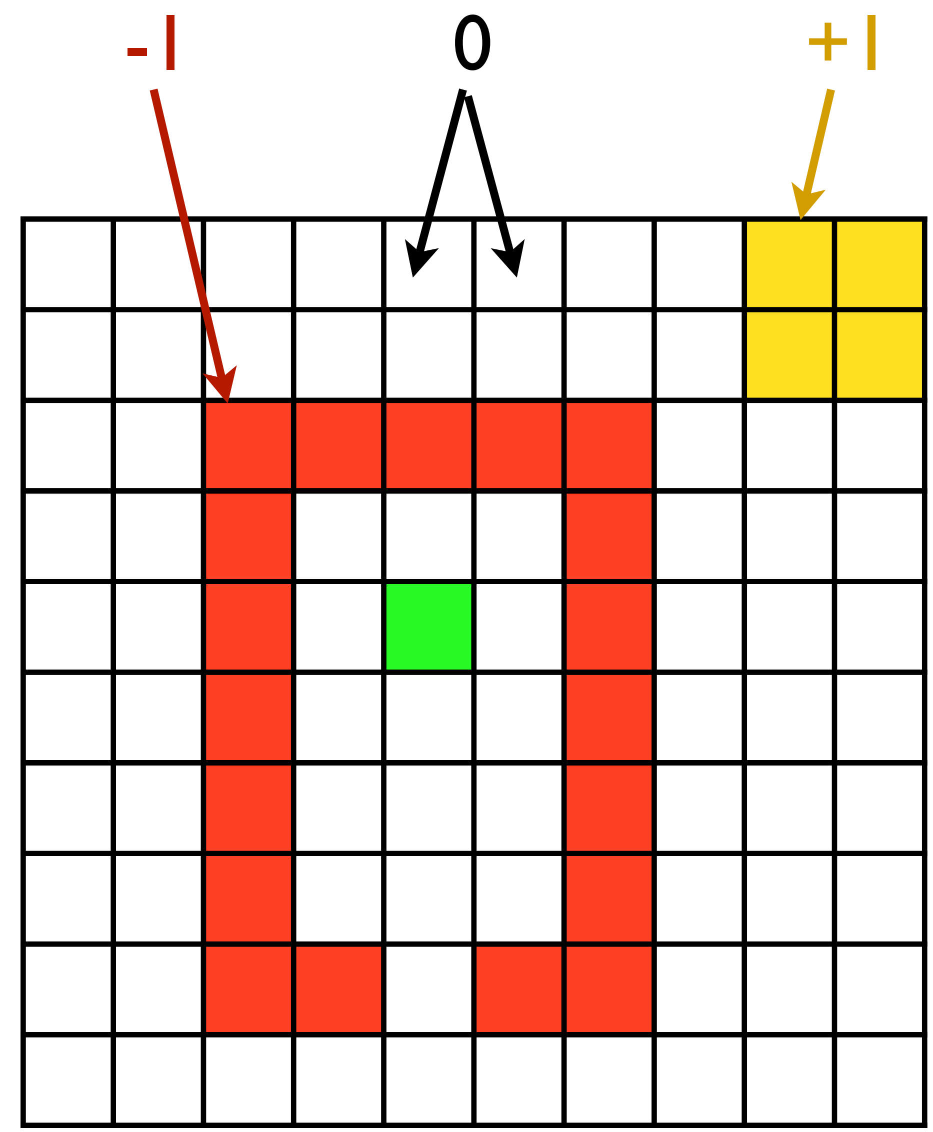
$$s_t \in \mathcal{S}$$

$$a_t \in \mathcal{A}$$

$$\mathcal{T}^a_{ss'} = p(s_{t+1}|s_t, a_t)$$

$$\boxed{r_t \sim \mathcal{R}(s_{t+1}, a_t, s_t)}$$

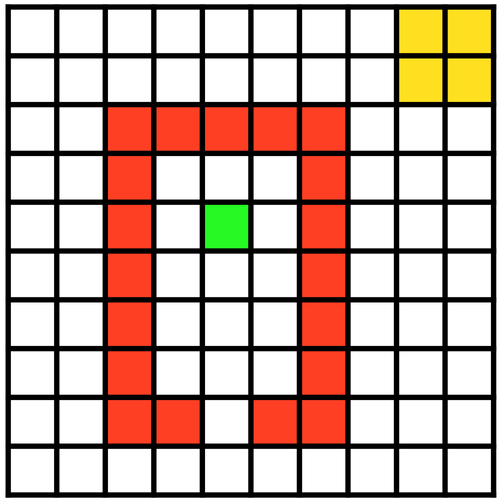
$$\pi(a|s) = p(a|s)$$



Tall orders

- Aim: maximise total future reward

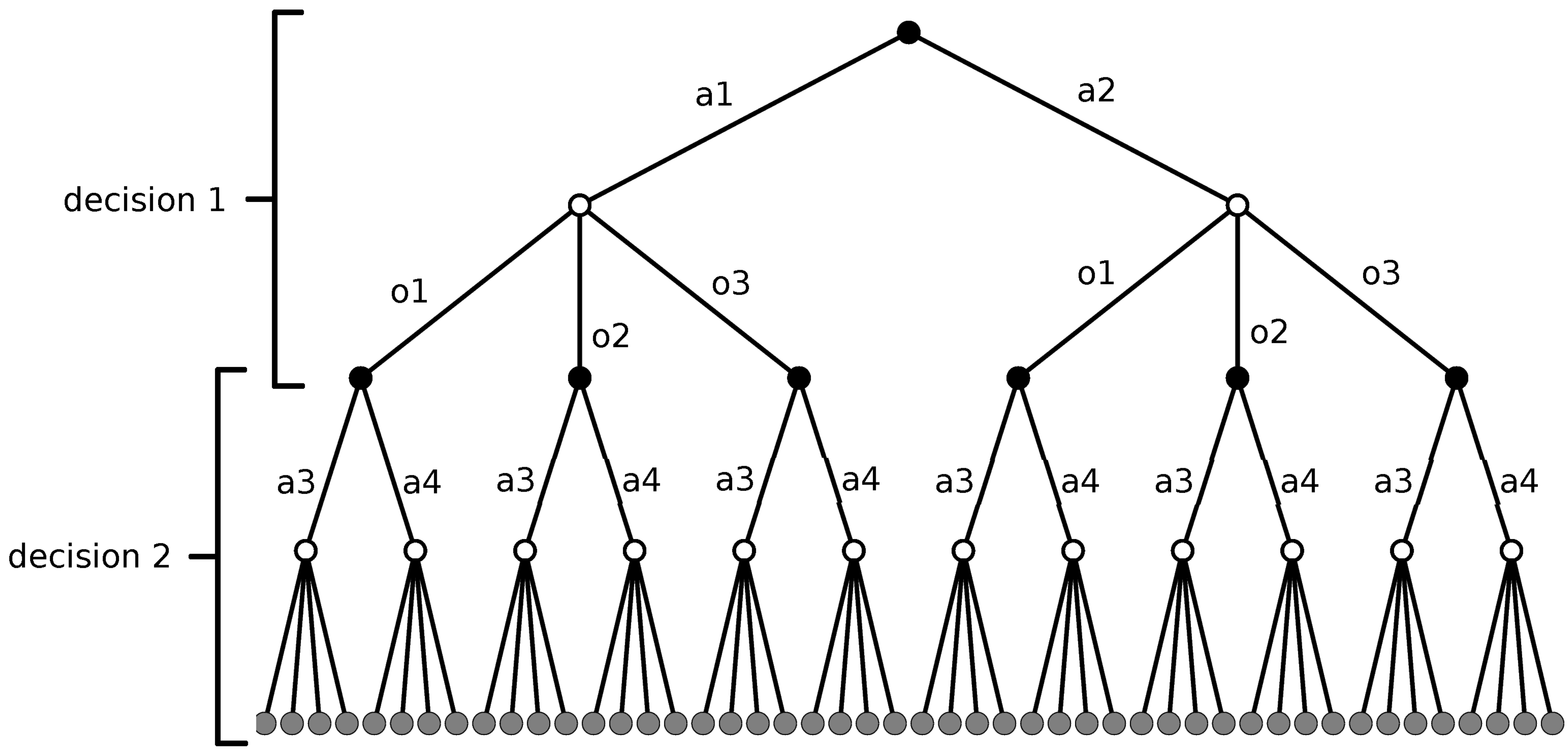
$$\sum_{t=1}^{\infty} r_t$$



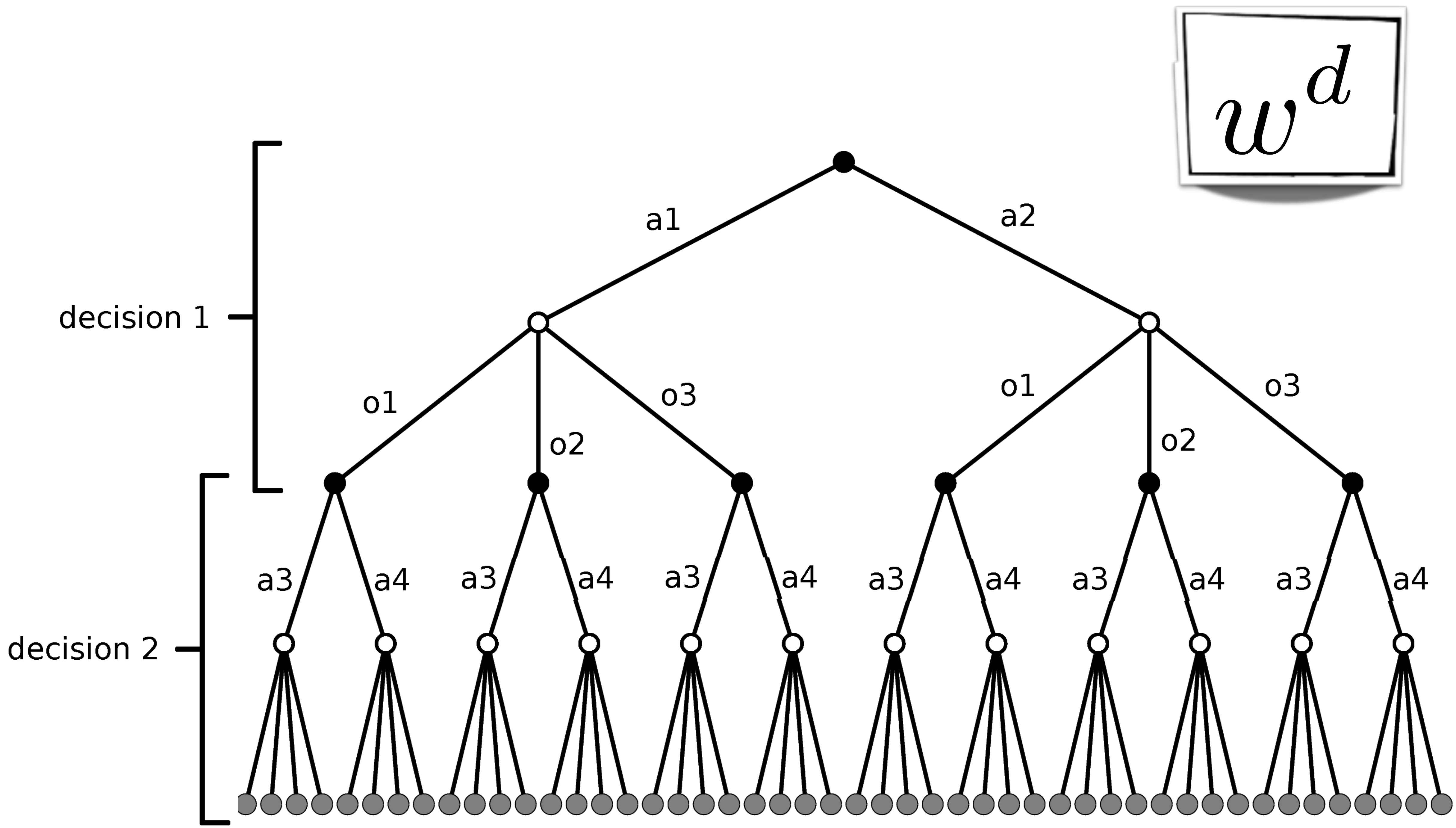
- i.e. we have to sum over paths through the future and weigh each by its probability

- Best policy achieves best long-term reward

Exhaustive tree search

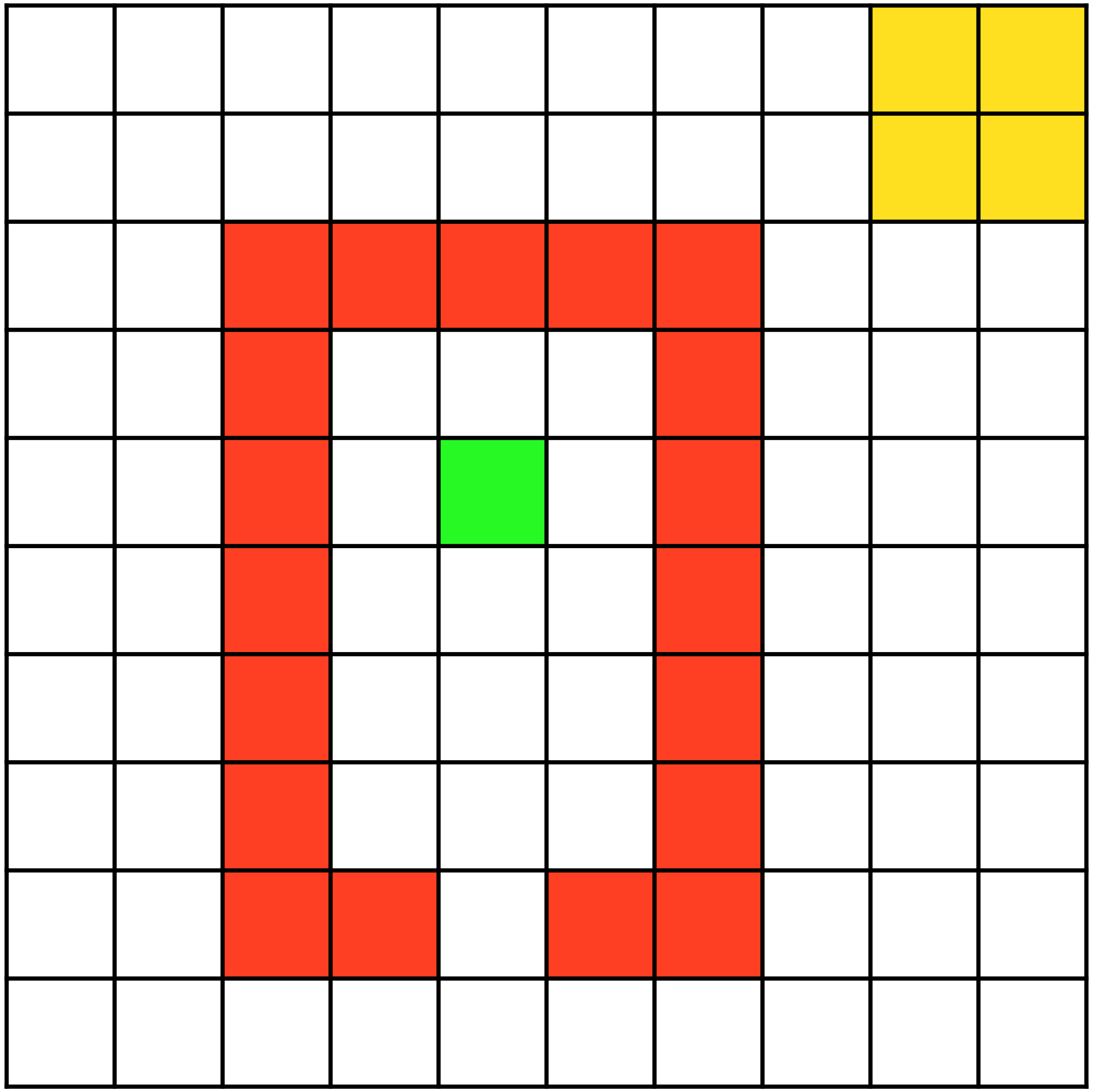


Exhaustive tree search



Decision tree

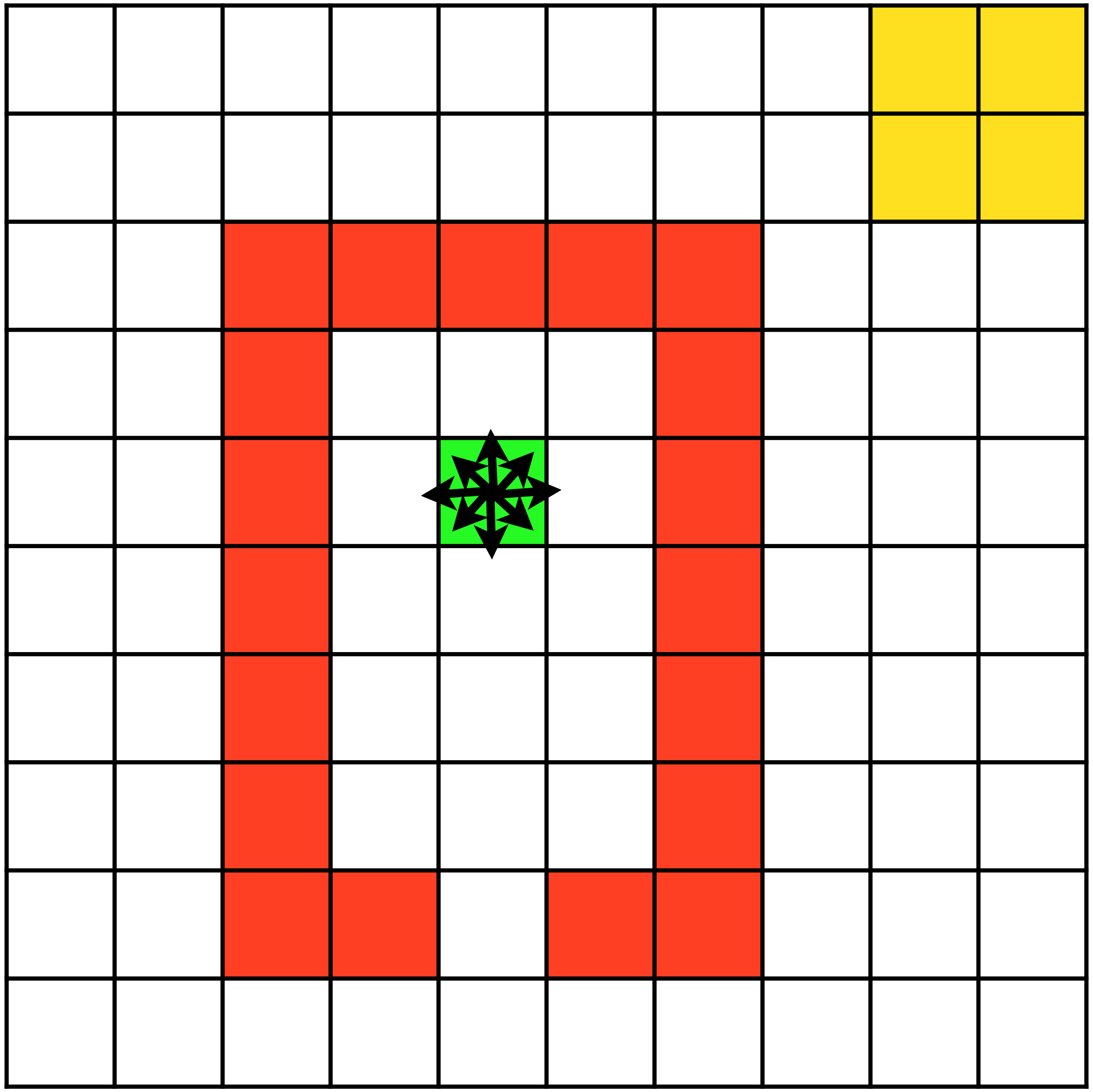
$$\sum_{t=1}^{\infty} r_t$$



Decision tree

$$\sum_{t=1}^{\infty} r_t$$

8

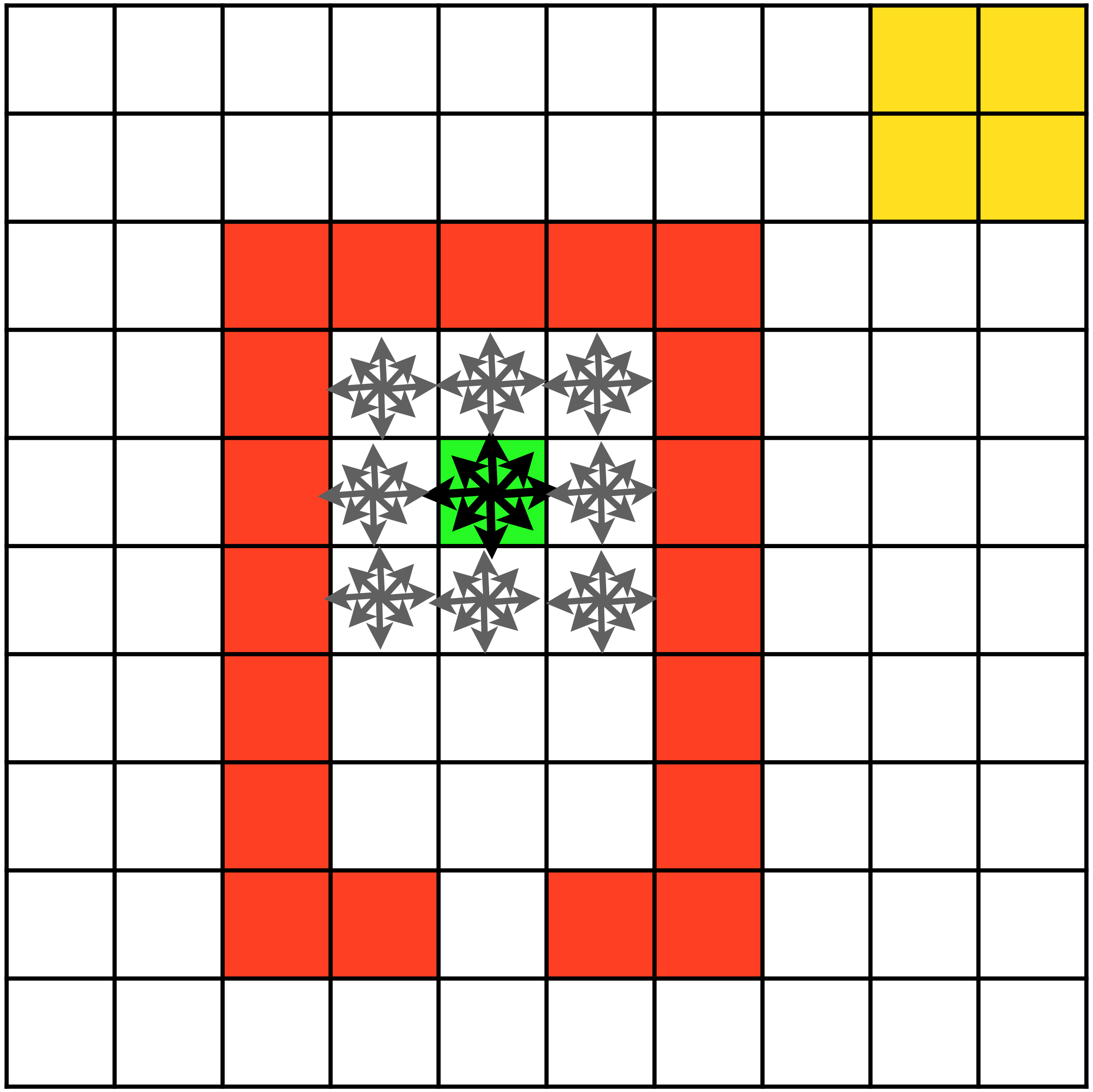


Decision tree

$$\sum_{t=1}^{\infty} r_t$$

8

64



Decision tree

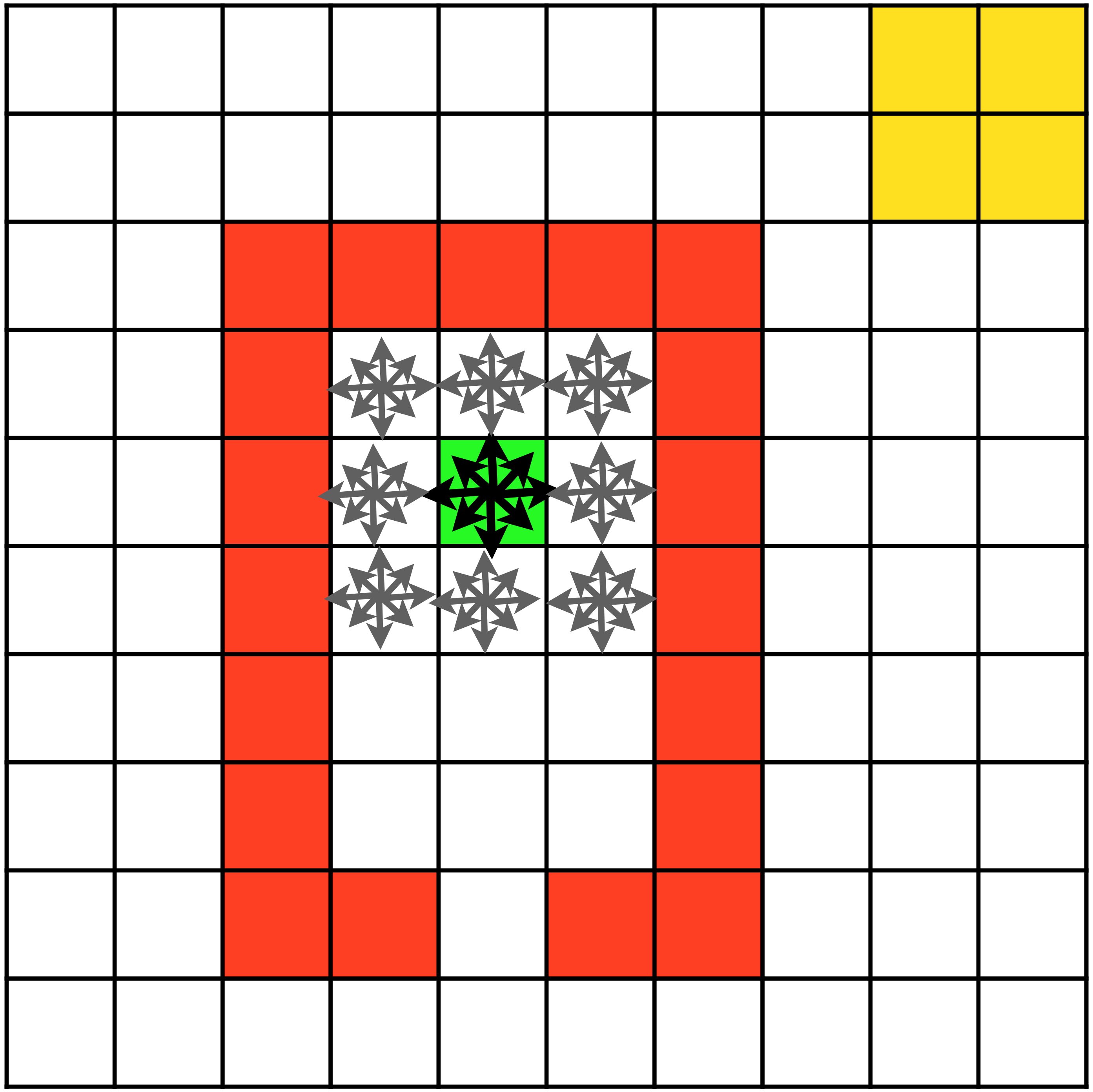
$$\sum_{t=1}^{\infty} r_t$$

8

64

512

...



Policy for this talk

- Pose the problem mathematically
- Policy evaluation
- Policy iteration
- Monte Carlo techniques: experience samples
- TD learning

Policy

Evaluate $\rightleftarrows$ Update

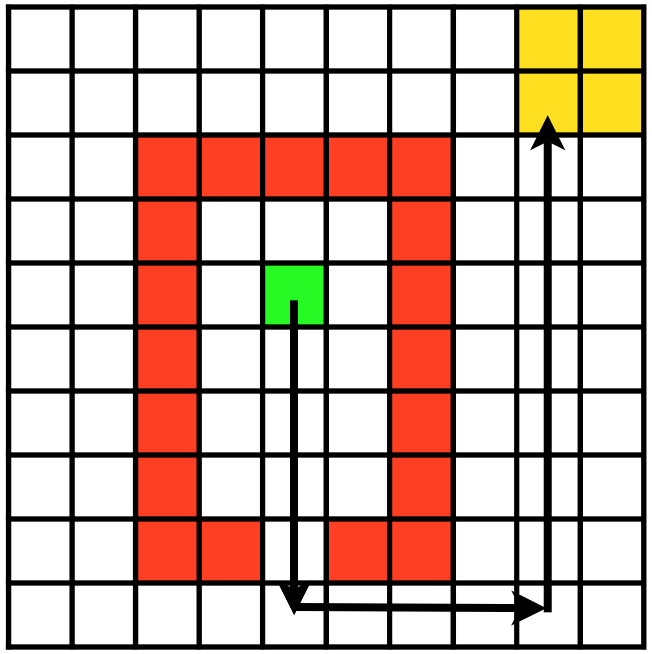
Evaluating a policy

- Aim: maximise total future reward

$$\sum_{t=1}^{\infty} r_t$$

- To know which is best, evaluate it first
- The policy determines the expected reward from each state

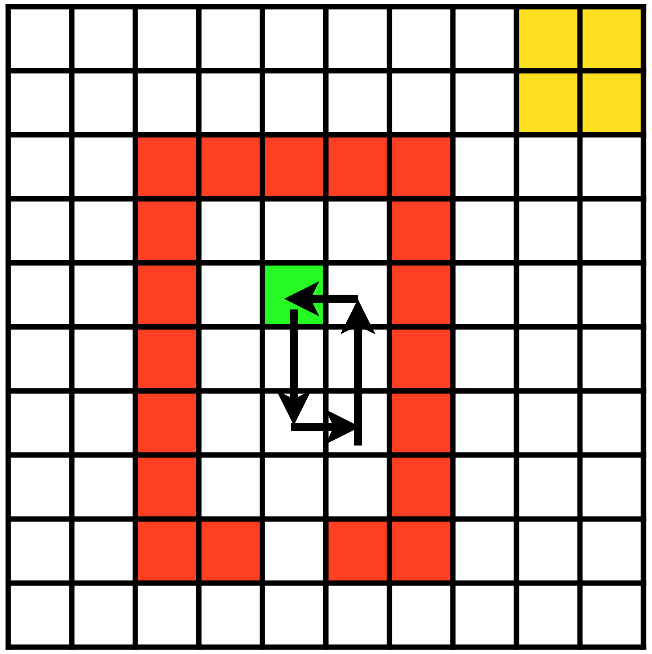
$$\mathcal{V}^\pi(s_1) \quad = \quad \mathbb{E}\left[\sum_{t=1}^{\infty} r_t | s_1 = 1, a_t \sim \pi\right]$$



Discounting

- Given a policy, each state has an expected value

$$\mathcal{V}^{\pi}(s_1) = \mathbb{E}\left[\sum_{t=1}^{\infty} r_t | s_1 = 1, a_t \sim \pi\right]$$



- But: $\sum_{t=0}^{\infty} r_t = \infty$

- Episodic $\sum_{t=0}^{T} r_t < \infty$

- Discounted $\sum_{t=0}^{\infty} \gamma^t r_t < \infty$
  - infinite horizons
  - finite, exponentially distributed horizons

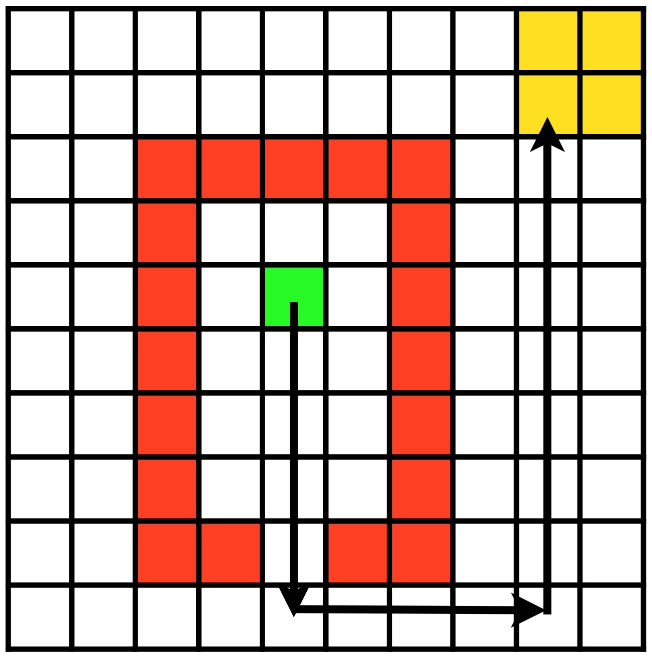
$$\sum_{t=0}^{T} \gamma^t r_t$$

$$T \sim \frac{1}{\tau} e^{t/\tau}$$

Discounting

- Given a policy, each state has an expected value

$$\mathcal{V}^{\pi}(s_1) = \mathbb{E}\left[\sum_{t=1}^{\infty} r_t | s_1 = 1, a_t \sim \pi\right]$$



- But: $\sum_{t=0}^{\infty} r_t = \infty$
- Episodic $\sum_{t=0}^{T} r_t < \infty$
- Discounted $\sum_{t=0}^{\infty} \gamma^t r_t < \infty$
  - infinite horizons
  - finite, exponentially distributed horizons

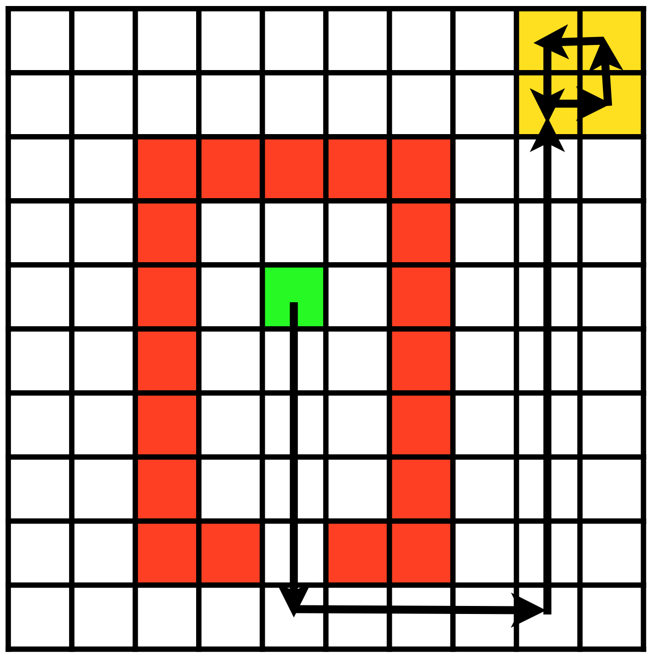
$$\sum_{t=0}^{T} \gamma^t r_t$$

$$T \sim \frac{1}{\tau} e^{t/\tau}$$

Discounting

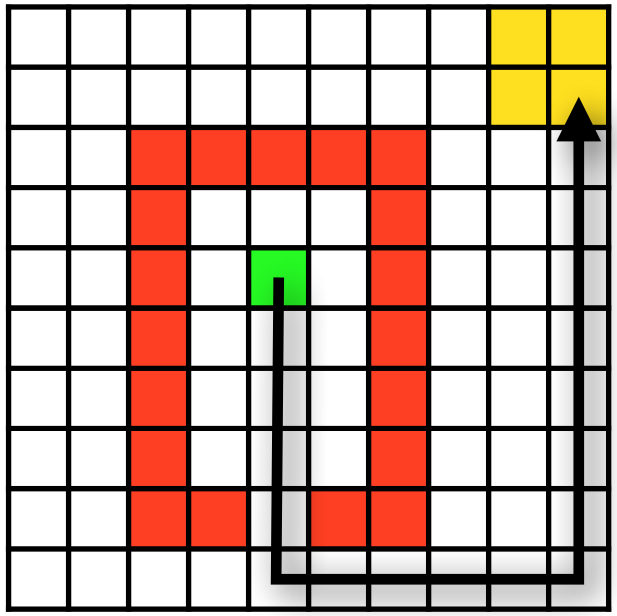
- Given a policy, each state has an expected value

$$\mathcal{V}^{\pi}(s_1) = \mathbb{E}\left[\sum_{t=1}^{\infty} r_t | s_1 = 1, a_t \sim \pi\right]$$



- But: $\sum_{t=0}^{\infty} r_t = \infty$
- Episodic $\sum_{t=0}^{T} r_t < \infty$
- Discounted
  - infinite horizons $\sum_{t=0}^{\infty} \gamma^t r_t < \infty$ $\quad \sum_{t=0}^{T} \gamma^t r_t$
  - finite, exponentially distributed horizons $T \sim \frac{1}{\tau} e^{t/\tau}$

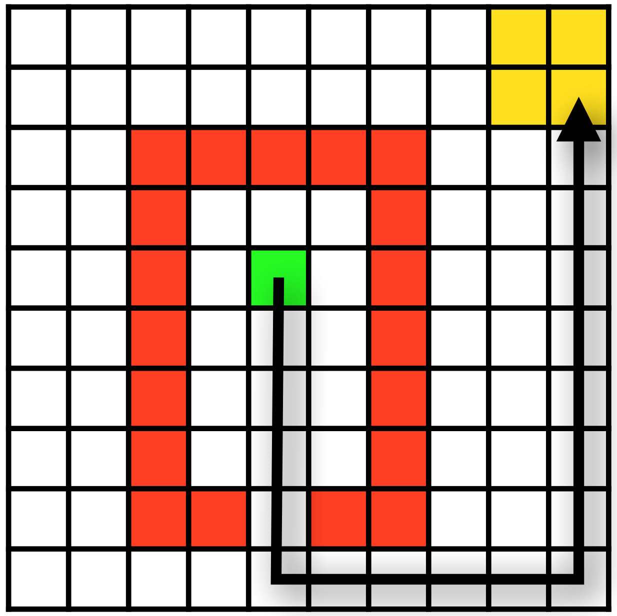
Markov Decision Problems



$$
\begin{aligned}
V^\pi(s_t) &= \mathbb{E}\left[\sum_{t'=1}^{\infty} r_{t'} \middle| s_t = s, \pi\right] \\
&= \mathbb{E}\left[r_1 \middle| s_t = s, \pi\right] + \mathbb{E}\left[\sum_{t=2}^{\infty} r_t \middle| s_t = s, \pi\right] \\
&= \mathbb{E}\left[r_1 \middle| s_t = s, \pi\right] + \mathbb{E}\left[V^\pi(s_{t+1}) \middle| s_t = s, \pi\right]
\end{aligned}
$$

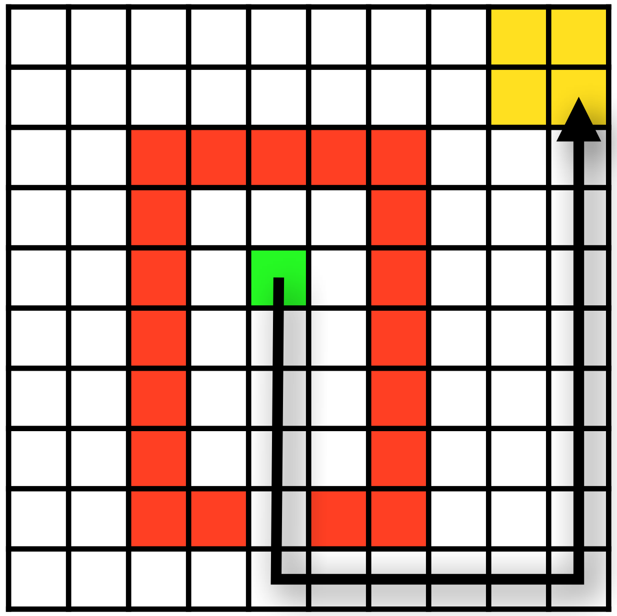
This dynamic consistency is key to many solution approaches.
It states that the value of a state s is related to
the values of its successor states s'.

Markov Decision Problems



$$
\begin{aligned}
V^\pi(s_t) &= \boxed{\mathbb{E}\left[r_1 | \, s_t = s, \pi\right]} + \mathbb{E}\left[V(s_{t+1}), \pi\right] \\
r_1 &\sim \mathcal{R}(s_2, a_1, s_1) \\
\mathbb{E}\left[r_1 | s_t = s, \pi\right] &= \mathbb{E}\left[\sum_{s_{t+1}} p(s_{t+1}|s_t, a_t)\mathcal{R}(s_{t+1}, a_t, s_t)\right] \\
&= \sum_{a_t} p(a_t|s_t)\left[\sum_{s_{t+1}} p(s_{t+1}|s_t, a_t)\mathcal{R}(s_{t+1}, a_t, s_t)\right] \\
&= \sum_{a_t} \pi(a_t, s_t)\left[\sum_{s_{t+1}} \mathcal{T}^{a_t}_{s_t s_{t+1}} \mathcal{R}(s_{t+1}, a_t, s_t)\right]
\end{aligned}
$$

Bellman equation



$$
\begin{aligned}
V^{\pi}(s_t) &= \mathbb{E}\left[r_1 \middle| s_t = s, \pi\right] + \mathbb{E}\left[V(s_{t+1}), \pi\right] \\
\mathbb{E}\left[r_1|s_t, \pi\right] &= \sum_a \pi(a, s_t) \left[\sum_{s_{t+1}} \mathcal{T}^a_{s_t s_{t+1}} \mathcal{R}(s_{t+1}, a, s_t)\right] \\
\mathbb{E}\left[V^{\pi}(s_{t+1}), \pi, s_t\right] &= \sum_a \pi(a, s_t) \left[\sum_{s_{t+1}} \mathcal{T}^a_{s_t s_{t+1}} V^{\pi}(s_{t+1})\right]
\end{aligned}
$$

$$
V^{\pi}(s) = \sum_a \pi(a|s) \left[\sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}(s', a, s) + V^{\pi}(s')\right]\right]
$$

Bellman Equation

$$V^\pi(s) \quad = \quad \sum_a \pi(a|s) \left[\sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}(s',a,s) + V^\pi(s')\right]\right]$$

Bellman Equation

All future reward from state s = E [Immediate reward + All future reward from next state s']

$$V^\pi(s) = \sum_a \pi(a|s) \left[\sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}(s',a,s) + V^\pi(s') \right] \right]$$

Bellman Equation

$$V^\pi(s) \quad = \quad \sum_a \pi(a|s) \left[\sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}(s', a, s) + V^\pi(s') \right] \right]$$

$$\text{All future reward from state s} \quad = \quad \text{E} \left[\text{Immediate reward} \quad + \quad \text{All future reward from next state s'} \right]$$

Q values = state-action values

$$V^\pi(s) = \sum_a \pi(a|s) \underbrace{\left[\sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}(s',a,s) + V^\pi(s')\right]\right]}_{\mathcal{Q}^\pi(s,a)}$$

- so we can define state-action values as:

$$\begin{aligned} \mathcal{Q}(s,a) &= \sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}(s',a,s) + V(s')\right] \\ &= \mathbb{E}\left[\sum_{t=1}^{\infty} r_t | s, a\right] \end{aligned}$$

- and state values are average state-action values:

$$V(s) = \sum_a \pi(a|s) \mathcal{Q}(s,a)$$

Bellman Equation

$$V^\pi(s) \quad = \quad \sum_a \pi(a|s) \left[\sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}(s', a, s) + V^\pi(s') \right] \right]$$

- to evaluate a policy, we need to solve the above equation, i.e. find the self-consistent state values

- options for policy evaluation
  - exhaustive tree search - outwards, inwards, depth-first
  - value iteration: iterative updates
  - linear solution in 1 step
  - sampling

Solving the Bellman Equation

Option 1: turn it into update equation

Option 2: linear solution (w/ absorbing states)

$$
\begin{aligned}
V(s) &= \sum_a \pi(a, s_t) \left[\sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}(s', a, s) + V(s') \right] \right] \\
\Rightarrow \mathbf{v} &= \mathbf{R}^\pi + \mathbf{T}^\pi \mathbf{v} \\
\Rightarrow \mathbf{v}^\pi &= (\mathbf{I} - \mathbf{T}^\pi)^{-1} \mathbf{R}^\pi \qquad \mathcal{O}(|\mathcal{S}|^3)
\end{aligned}
$$

Solving the Bellman Equation

Option 1: turn it into update equation

$$V^{k+1}(s) = \sum_a \pi(a, s_t) \left[\sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}(s', a, s) + V^k(s') \right] \right]$$

Option 2: linear solution (w/ absorbing states)

$$\begin{aligned} V(s) &= \sum_a \pi(a, s_t) \left[\sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}(s', a, s) + V(s') \right] \right] \\ \Rightarrow \mathbf{v} &= \mathbf{R}^\pi + \mathbf{T}^\pi \mathbf{v} \\ \Rightarrow \mathbf{v}^\pi &= (\mathbf{I} - \mathbf{T}^\pi)^{-1} \mathbf{R}^\pi \qquad \mathcal{O}(|\mathcal{S}|^3) \end{aligned}$$

Policy update

Given the value function for a policy, say via linear solution

$$V^\pi(s) \quad = \quad \sum_a \pi(a|s) \underbrace{\left[\sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}(s',a,s) + V^\pi(s')\right]\right]}_{\mathcal{Q}^\pi(s,a)}$$

Given the values V for the policy, we can improve the policy by always choosing the best action:

$$\pi'(a|s) = \begin{cases} 1 \,\text{if}\, a = \operatorname{argmax}_a \mathcal{Q}^\pi(s,a) \\ 0 \,\text{else} \end{cases}$$

It is guaranteed to improve:

for deterministic policy

$$\mathcal{Q}^\pi(s,\pi'(s)) = \max_a \mathcal{Q}^\pi(s,a) \geq \mathcal{Q}^\pi(s,\pi(s)) = \mathcal{V}^\pi(s)$$

Policy iteration

Policy evaluation

$$\mathbf{v}^\pi \quad = \quad (\mathbf{I} - \mathbf{T}^\pi)^{-1}\mathbf{R}^\pi$$

$$\pi(a|s) = \begin{cases} 1 \,\text{if}\, a = \text{argmax}_a \sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}^a_{ss} + V^{pi}(s')\right] \\ 0 \,\text{else} \end{cases}$$

Policy iteration

Policy evaluation

$$\mathbf{v}^{\pi} \quad = \quad (\mathbf{I} - \mathbf{T}^{\pi})^{-1}\mathbf{R}^{\pi}$$

greedy policy improvement

$$\pi(a|s) = \begin{cases} 1 \, \text{if} \, a = \text{argmax}_a \sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}^a_{ss} + V^{pi}(s')\right] \\ 0 \, \text{else} \end{cases}$$

Policy iteration

Policy evaluation

$$\mathbf{v}^{\pi} = (\mathbf{I} - \mathbf{T}^{\pi})^{-1}\mathbf{R}^{\pi}$$

Value iteration

$$V^*(s) = \max_a \sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}^a_{ss} + V^*(s')\right]$$

greedy policy improvement

$$\pi(a|s) = \begin{cases} 1 \,\text{if}\, a = \text{argmax}_a \sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}^a_{ss} + V^{pi}(s')\right] \\ 0 \,\text{else} \end{cases}$$

Model-free solutions

- So far we have assumed knowledge of R and T
  - R and T are the 'model' of the world, so we assume full knowledge of the dynamics and rewards in the environment
- What if we don't know them?
- We can still learn from state-action-reward samples
  - we can learn R and T from them, and use our estimates to solve as above
  - alternatively, we can directly estimate V or Q

Solving the Bellman Equation

Option 3: sampling

$$V(s) = \sum_a \pi(a, s_t) \left[\sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}(s', a, s) + V(s') \right] \right]$$

So we can just draw some samples from the policy and the transitions and average over them:

$$a = \sum_k f(x_k) p(x_k)$$

$$x^{(i)} \sim p(x) \rightarrow \hat{a} = \frac{1}{N} \sum_i f(x^{(i)})$$

Solving the Bellman Equation

Option 3: sampling

So we can just draw some samples from the policy and the transitions and average over them:

$$a = \sum_k f(x_k)p(x_k)$$

$$x^{(i)} \sim p(x) \to \hat{a} = \frac{1}{N}\sum_i f(x^{(i)})$$

Solving the Bellman Equation

Option 3: sampling

this is an expectation over policy and transition samples.

So we can just draw some samples from the policy and the transitions and average over them:

$$\begin{aligned} a &= \sum_k f(x_k)p(x_k) \\ x^{(i)} \sim p(x) \rightarrow \hat{a} &= \frac{1}{N}\sum_i f(x^{(i)}) \end{aligned}$$

Solving the Bellman Equation

Option 3: sampling

this is an expectation over policy and transition samples.

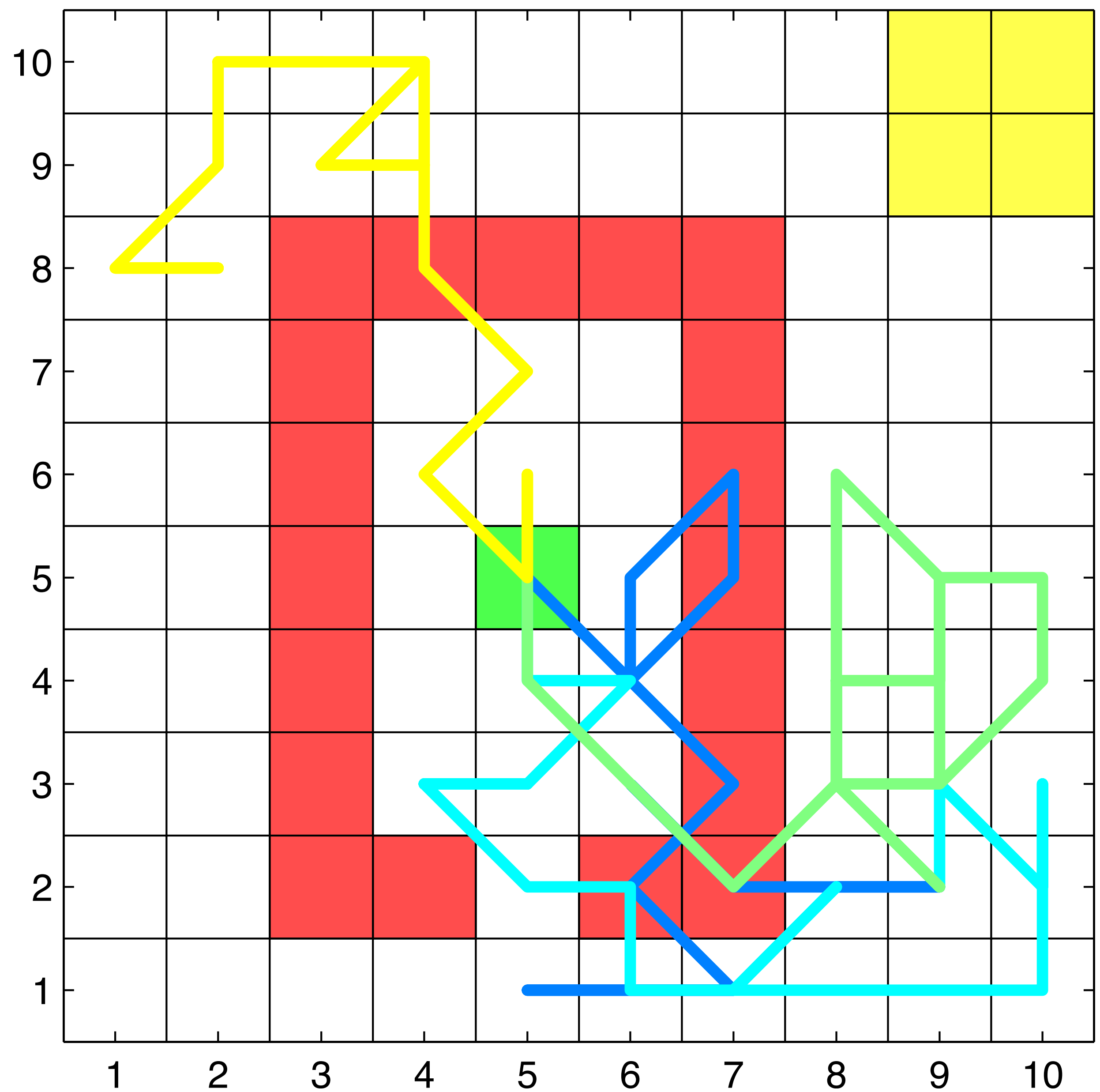
So we can just draw some samples from the policy and the transitions and average over them:

$$a = \sum_k f(x_k)p(x_k)$$

$$x^{(i)} \sim p(x) \rightarrow \hat{a} = \frac{1}{N}\sum_i f(x^{(i)})$$

more about this later...

Learning from samples



A new problem: exploration versus exploitation

The effect of bootstrapping

B1
B1
B1
B1
B1
B1
B0
A0 B0

Markov (every visit)
V(B)=3/4
V(A)=0

TD
V(B)=3/4
V(A)=~3/4

- Average over various bootstrappings: TD(λ)

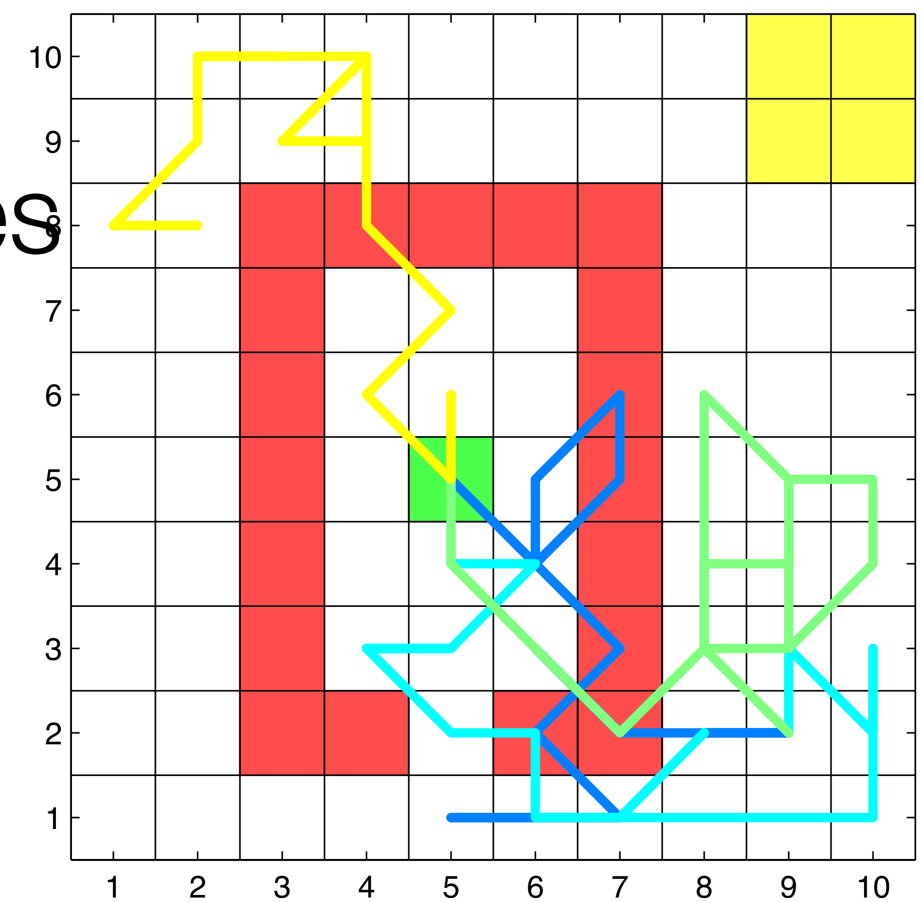
after Sutton and Barto 1998

Monte Carlo

- First visit MC
  - randomly start in all states, generate paths, average for starting state only

$$\mathcal{V}(s) = \frac{1}{N}\sum_i \left\{ \sum_{t'=1}^{T} r^i_{t'} | s_0 = s \right\}$$

- More efficient use of samples
  - Every visit MC
  - Bootstrap: TD
  - Dyna
- Better samples
  - on policy versus off policy
  - Stochastic search, UCT...



Update equation: towards TD

Bellman equation

$$V(s) \quad = \quad \sum_a \pi(a,s) \left[\sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}(s',a,s) + V(s') \right] \right]$$

Not yet converged, so it doesn't hold:

$$dV(s) = -V(s) + \sum_a \pi(a,s) \left[\sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}(s',a,s) + V(s') \right] \right]$$

And then use this to update

$$V^{i+1}(s) = V^i(s) + dV(s)$$

TD learning

$$dV(s) = -V(s) + \sum_a \pi(a,s) \left[\sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}(s',a,s) + V(s') \right] \right]$$

TD learning

$$dV(s) = -V(s) + \sum_a \pi(a,s) \left[\sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}(s',a,s) + V(s') \right] \right]$$

Sample

$$\begin{aligned} a_t &\sim \pi(a|s_t) \\ s_{t+1} &\sim \mathcal{T}^{a_t}_{s_t,s_{t+1}} \\ r_t &= \mathcal{R}(s_{t+1},a_t,s_t) \end{aligned}$$

TD learning

$$dV(s) = -V(s) + \sum_a \pi(a,s)\left[\sum_{s'} \mathcal{T}^a_{ss'}\left[\mathcal{R}(s',a,s) + V(s')\right]\right]$$

Sample

$$\begin{aligned} a_t &\sim \pi(a|s_t) \\ s_{t+1} &\sim \mathcal{T}^{a_t}_{s_t,s_{t+1}} \\ r_t &= \mathcal{R}(s_{t+1},a_t,s_t) \end{aligned}$$

$$\delta_t = -V_{t-1}(s_t) + r_t + V_{t-1}(s_{t+1})$$

TD learning

$$dV(s) = -V(s) + \sum_a \pi(a, s) \left[\sum_{s'} \mathcal{T}^a_{ss'} \left[\mathcal{R}(s', a, s) + V(s') \right] \right]$$

Sample

$$\begin{aligned} a_t &\sim \pi(a|s_t) \\ s_{t+1} &\sim \mathcal{T}^{a_t}_{s_t, s_{t+1}} \\ r_t &= \mathcal{R}(s_{t+1}, a_t, s_t) \end{aligned}$$

$$\delta_t = -V_{t-1}(s_t) + r_t + V_{t-1}(s_{t+1})$$

$$V^{i+1}(s) = V^i(s) + dV(s)$$

$$V_t(s_t) = V_{t-1}(s_t) + \alpha \delta_t$$

TD learning

$$\begin{aligned} a_t &\sim \pi(a|s_t) \\ s_{t+1} &\sim \mathcal{T}^{a_t}_{s_t,s_{t+1}} \\ r_t &= \mathcal{R}(s_{t+1},a_t,s_t) \end{aligned}$$

$$\delta_t = -V_t(s_t) + r_t + V_t(s_{t+1})$$

$$V_{t+1}(s_t) = V_t(s_t) + \alpha\delta_t$$

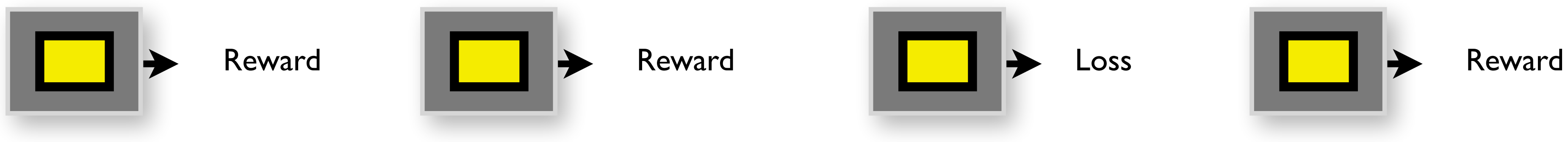
Phasic dopamine neurone firing in

- Pavlovian conditioning

Montague et al., 1996, Schultz et al., 1997

Phasic dopamine neurone firing in

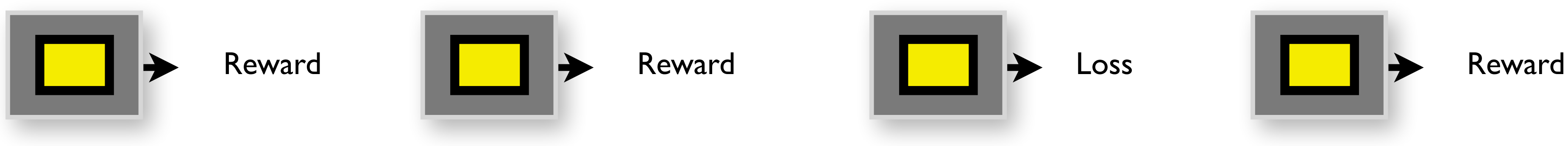
- Pavlovian conditioning



Montague et al., 1996, Schultz et al., 1997

Phasic dopamine neurone firing in

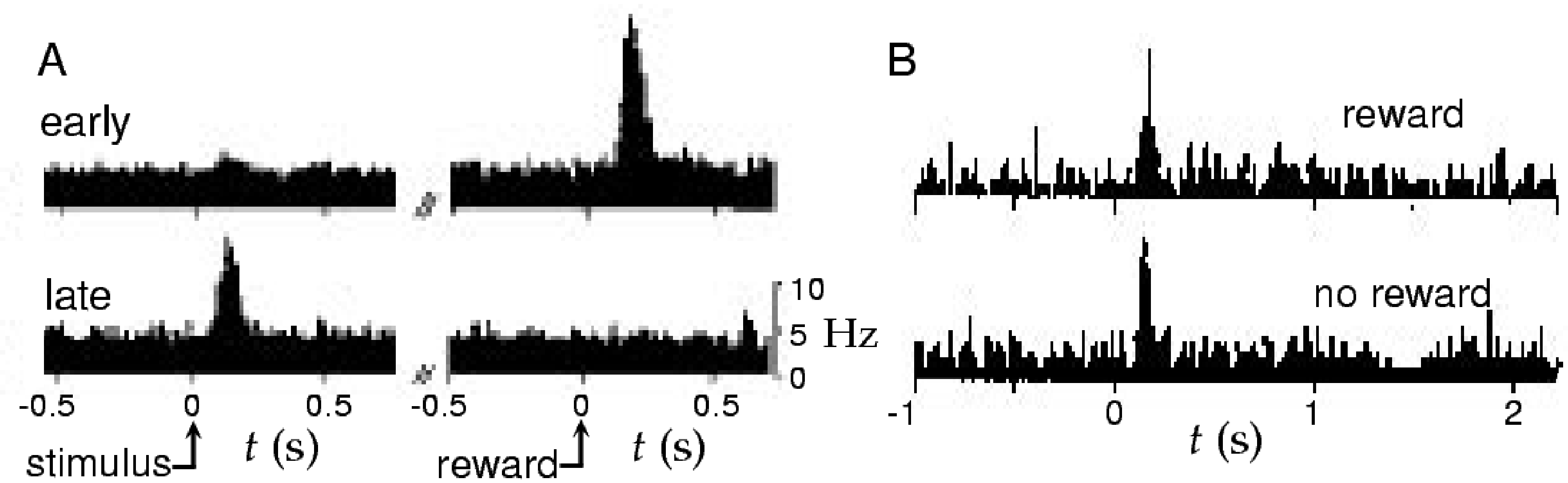
- Pavlovian conditioning



$$\mathcal{V}_{t+1}(s) = \mathcal{V}_t(s) + \epsilon \underbrace{\left(\mathcal{R}_t - \mathcal{V}_t(s)\right)}_{=\ \text{Prediction error}}$$

Montague et al., 1996, Schultz et al., 1997

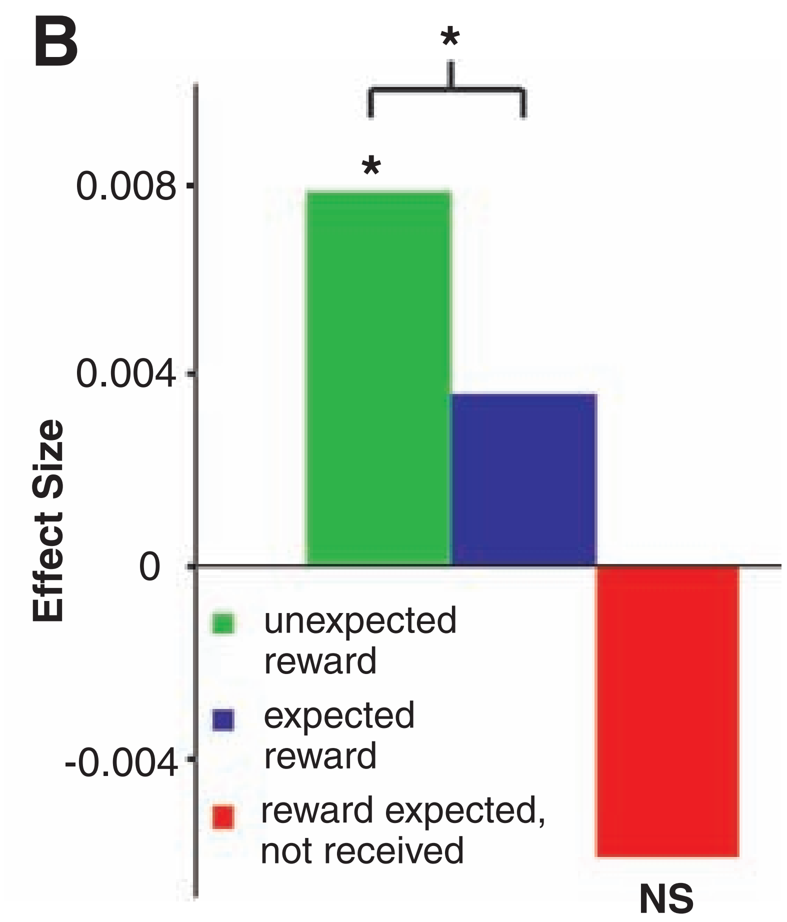
- Pavlovian conditioning

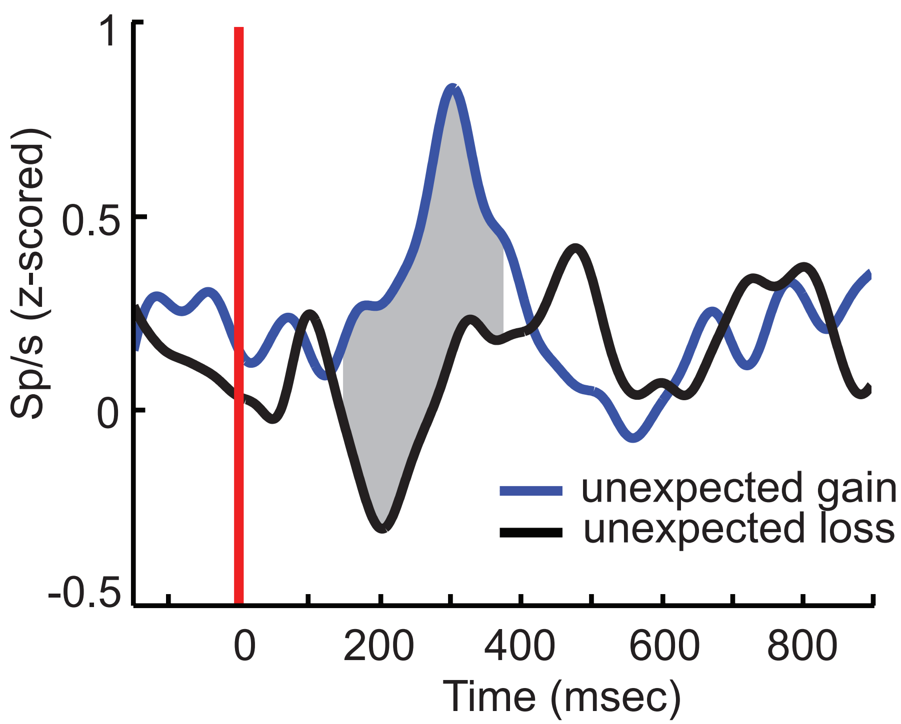


$$\mathcal{V}_{t+1}(s) = \mathcal{V}_t(s) + \epsilon \underbrace{(\mathcal{R}_t - \mathcal{V}_t(s))}_{= \text{Prediction error}}$$

Montague et al., 1996, Schultz et al., 1997

Phasic signals in humans



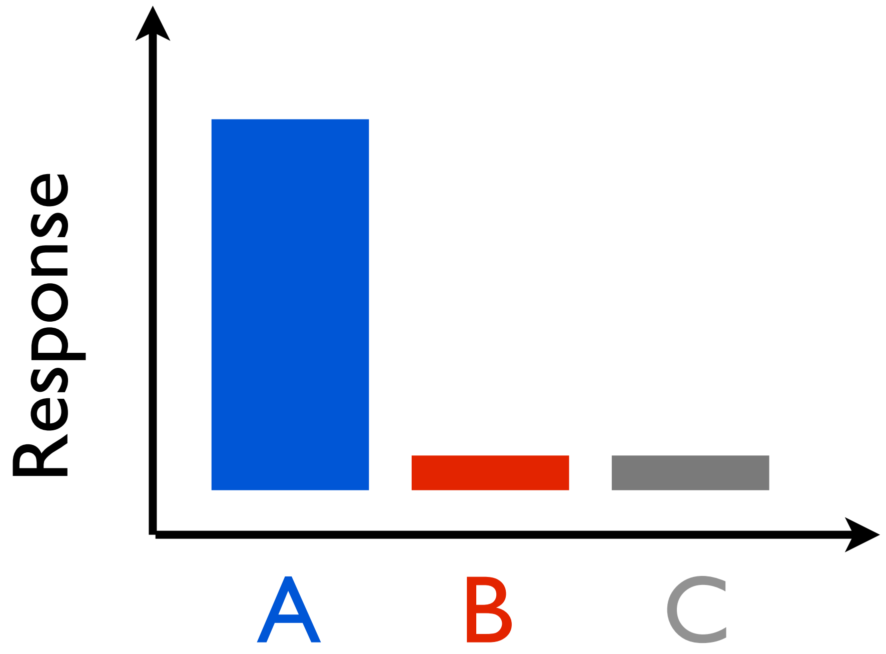


D'Ardenne et al., 2008 Science; Zaghloul et al., 2009 Science

Blocking

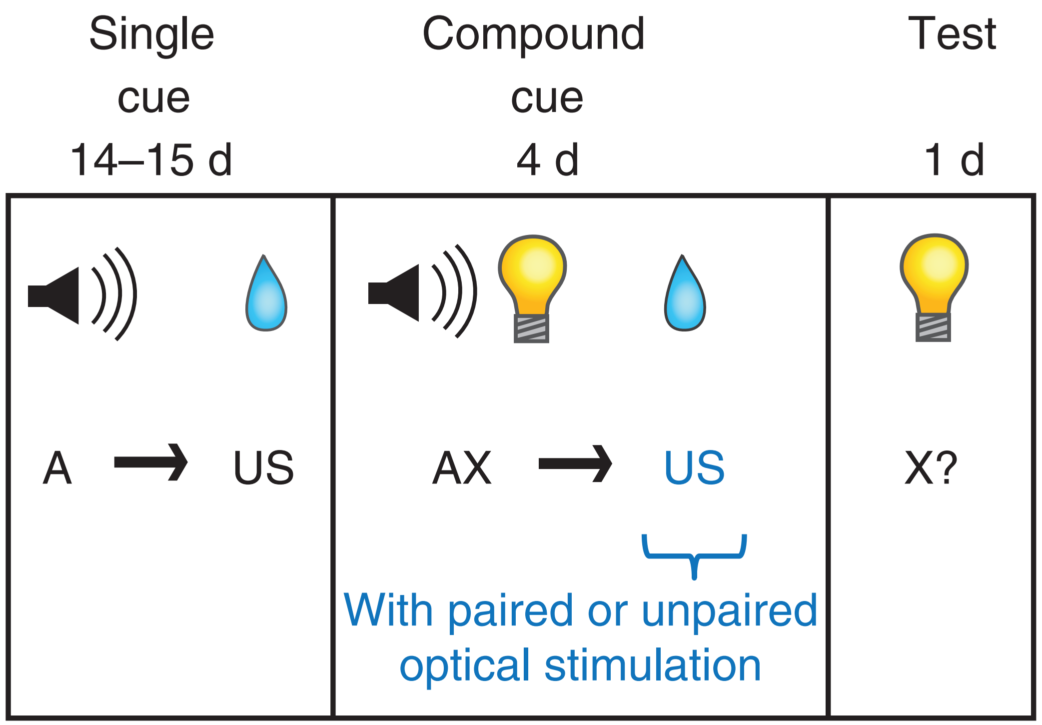
- Are predictions and prediction errors really causally important in learning?

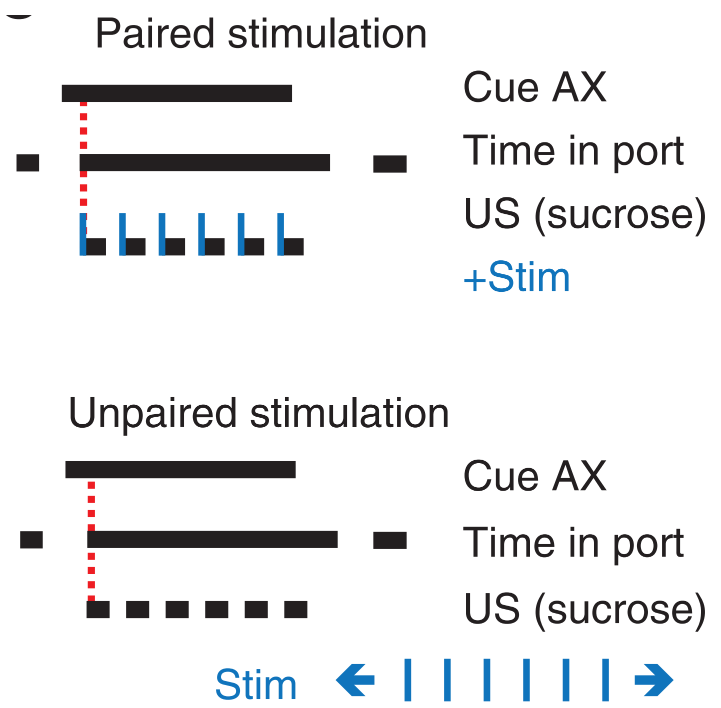
- 1: A -> Reward
- 2: A+B -> Reward
- 3: A -> ? approach
- B -> ? approach

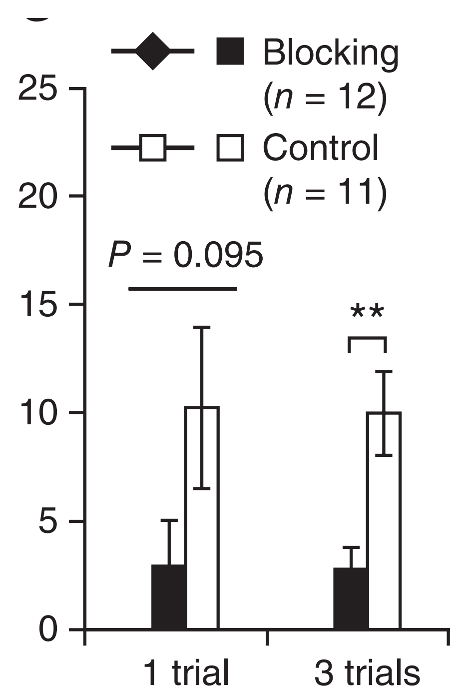


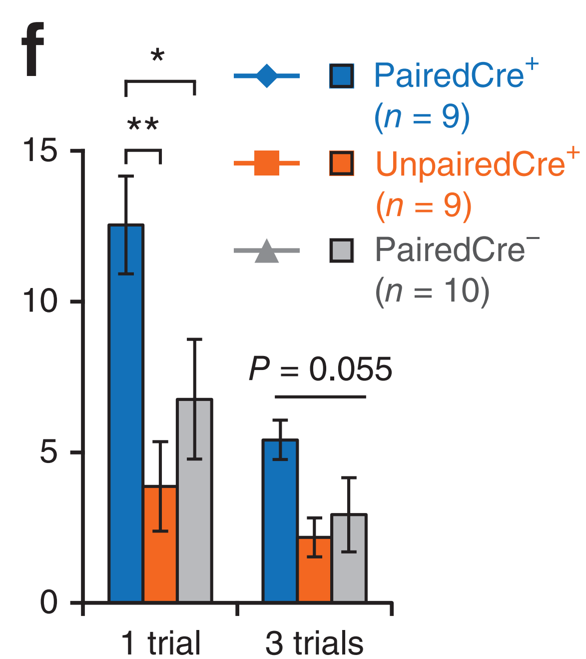
Kamin 1968

Causal role of phasic DA in learning



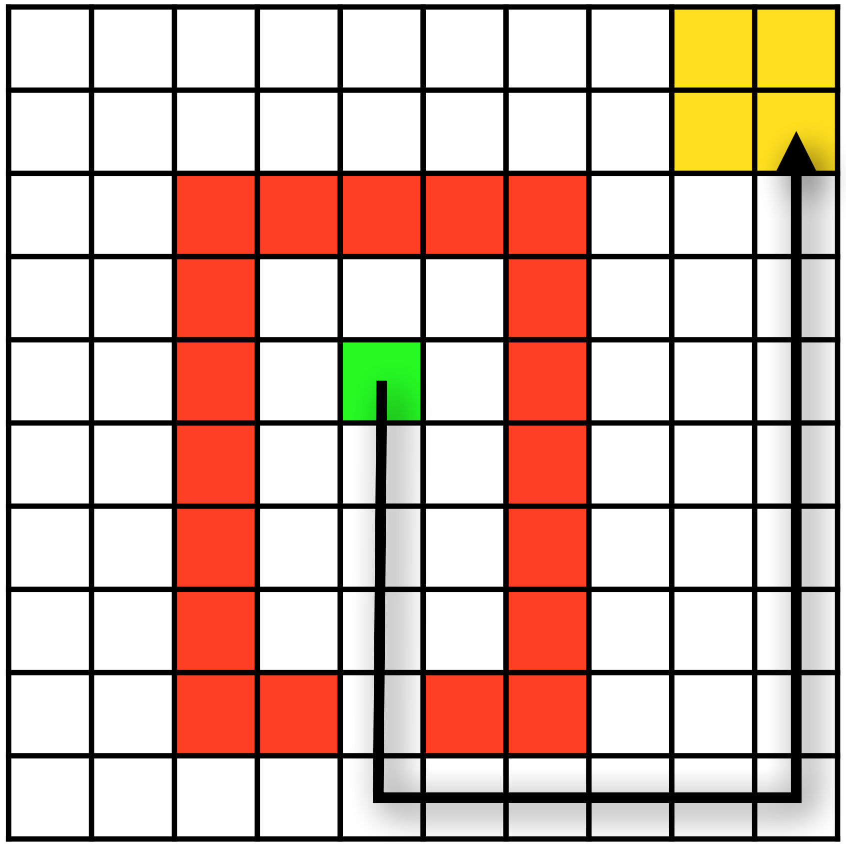






Steinberg et al., 2013 Nat. Neurosci.

Markov Decision Problems



$$V(s_t) = \mathbb{E}[r_t + r_{t+1} + r_{t+2} + \ldots]$$
$$= \mathbb{E}[r_t] + \mathbb{E}[r_{t+1} + r_{t+2} + r_{t+3} \ldots]$$
$$\Rightarrow V(s_t) = \mathbb{E}[r_t] + V(s_{t+1})$$

“Cached” solutions to MDPs

"Cached" solutions to MDPs

- Learn from experience
- If we have true values V, then this is true every trial:

$$V(s_t) = \mathbb{E}[r_t] + V(s_{t+1})$$

- If it is not true (we don't know true V), then we get an error:

$$\delta = (\mathbb{E}[r_t] + V(s_{t+1})) - V(s_t) \neq 0$$

- So now we can update with our experience

$$V(s_t) \leftarrow V(s_t) + \epsilon\delta$$

- This is an average over past experience

SARSA

- Do TD for state-action values instead:

$$\mathcal{Q}(s_t, a_t) \leftarrow \mathcal{Q}(s_t, a_t) + \alpha[r_t + \gamma \mathcal{Q}(s_{t+1}, a_{t+1}) - \mathcal{Q}(s_t, a_t)]$$

$$s_t, a_t, r_t, s_{t+1}, a_{t+1}$$

- convergence guarantees - will estimate $\mathcal{Q}^\pi(s, a)$

Q learning: off-policy

- Learn off-policy
  - draw from some policy
  - “only” require extensive sampling

$$\mathcal{Q}(s_t, a_t) \leftarrow \mathcal{Q}(s_t, a_t) + \alpha \left[\underbrace{r_t + \gamma \max_a \mathcal{Q}(s_{t+1}, a)}_{\text{update towards optimum}} - \mathcal{Q}(s_t, a_t) \right]$$

- will estimate $\mathcal{Q}^*(s, a)$

MF and MB learning of V and Q values

	Model-free
Pavlovian (state) values	$\mathcal{V}^{\text{MF}}(s)$
Instrumental (state-action) values	$\mathcal{Q}^{\text{MF}}(s,a)$

There are both Pavlovian state and instrumental state-action values, and both of these can be either model-free (cached) or model-based.

Solutions

- “Cached” learning
  - average experience
  - do again what worked in the past
  - averages are cheap to compute - no computational curse
  - averages move slowly

If you have an average over large number of subjects, it won’t move much if you add one more.

Solutions

- "Cached" learning
  - average experience
  - do again what worked in the past
  - averages are cheap to compute - no computational curse
  - averages move slowly

If you have an average over large number of subjects, it won't move much if you add one more.

- "Goal-directed" or "Model-based" decisions
  - Think through possible options and choose the best
  - Requires detailed model of the world
  - Requires huge computational resources
  - Learning = building the model, extracting structure

MF and MB learning of V and Q values

	Model-free	**Model-based**
Pavlovian (state) values	$\mathcal{V}^{\text{MF}}(s)$	$\mathcal{V}^{\text{MB}}(s)$
Instrumental (state-action) values	$\mathcal{Q}^{\text{MF}}(s,a)$	$\mathcal{Q}^{\text{MB}}(s,a)$

There are both Pavlovian state and instrumental state-action values, and both of these can be either model-free (cached) or model-based.

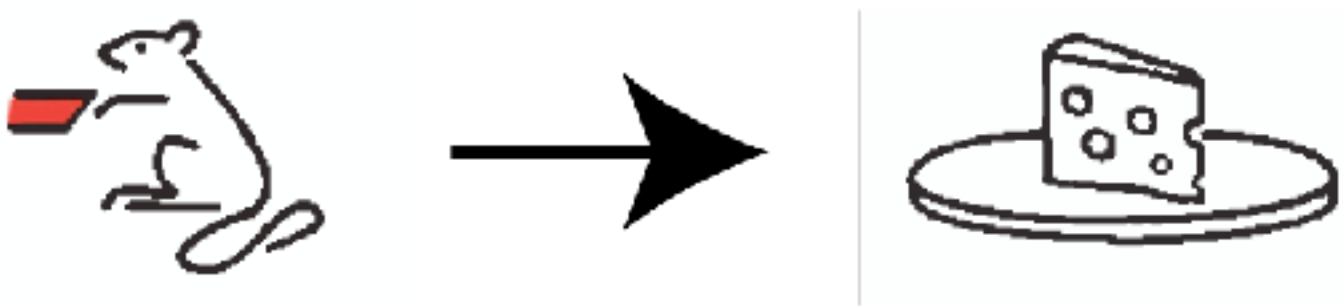
Pavlovian and instrumental

- Pavlovian model-free learning:

$$\mathcal{V}_t(s) = \mathcal{V}_{t-1}(s) + \epsilon(r_t - \mathcal{V}_{t-1}(s))$$

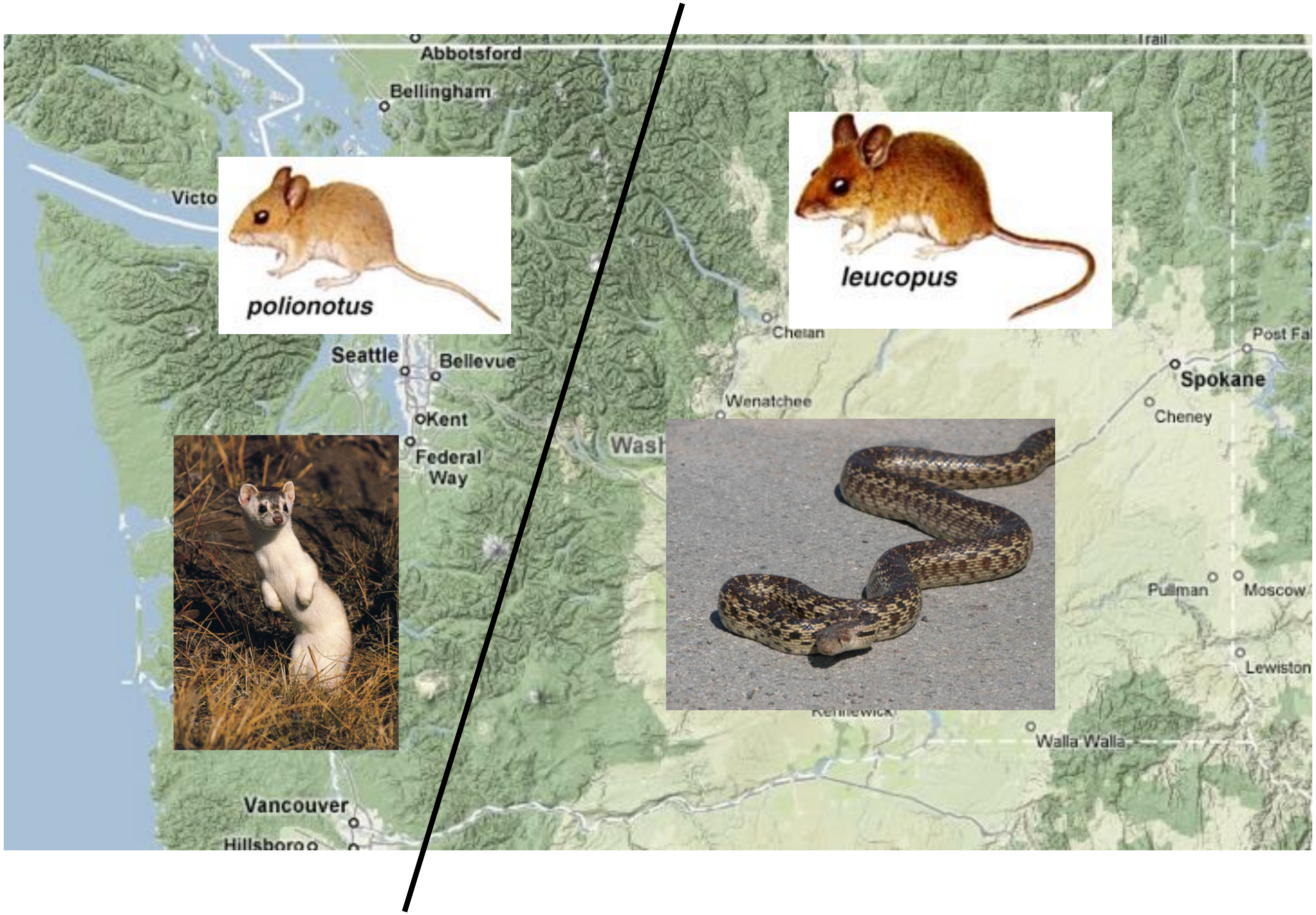
$$p(a|s,\mathcal{V}) \propto f(a,\mathcal{V}(s))\, p(a|s)$$

- Instrumental model-free learning:



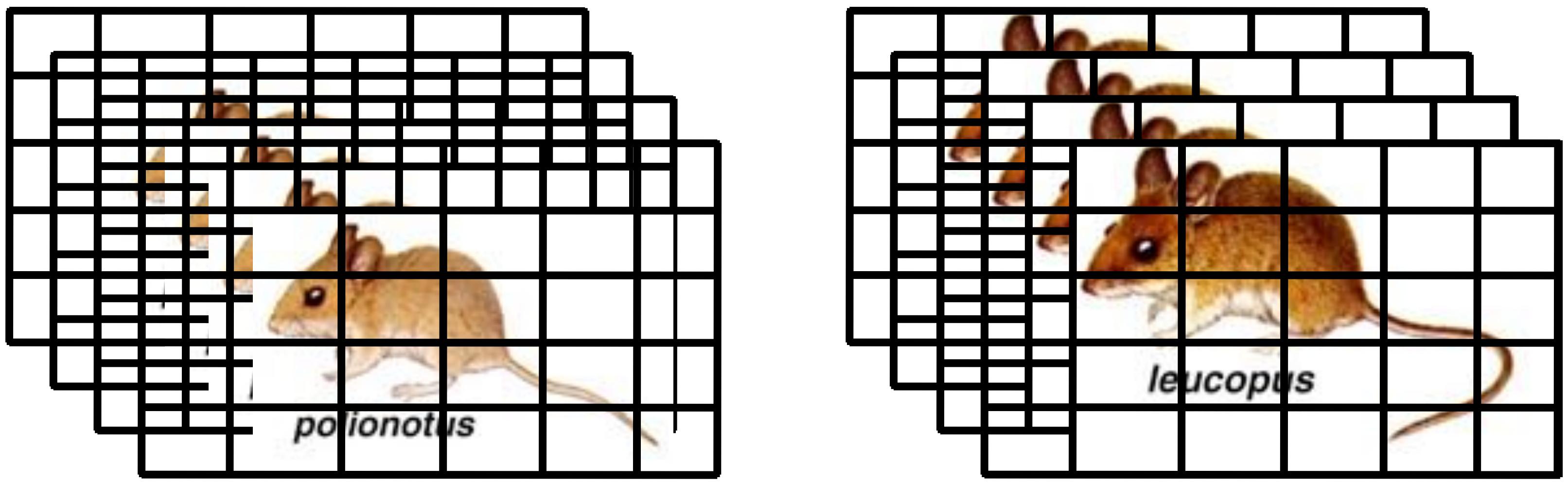
$$\mathcal{Q}_t(a,s) = \mathcal{Q}_{t-1}(a,s) + \epsilon(r_t - \mathcal{Q}_{t-1}(a,s))$$

Innate evolutionary strategies



Hirsch and Bolles 1980

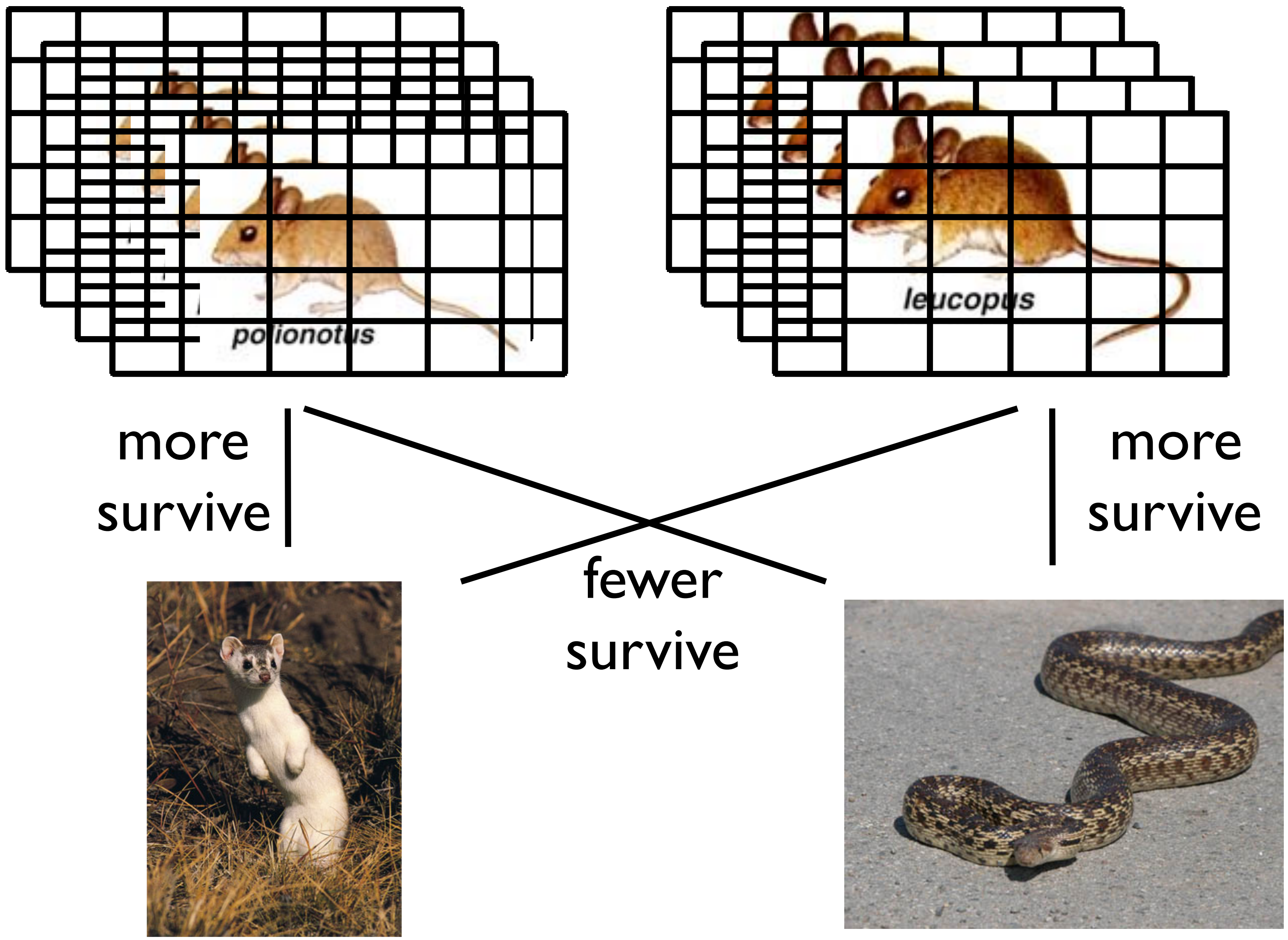
Innate evolutionary strategies



Hirsch and Bolles 1980

Innate evolutionary strategies

are quite sophisticated...



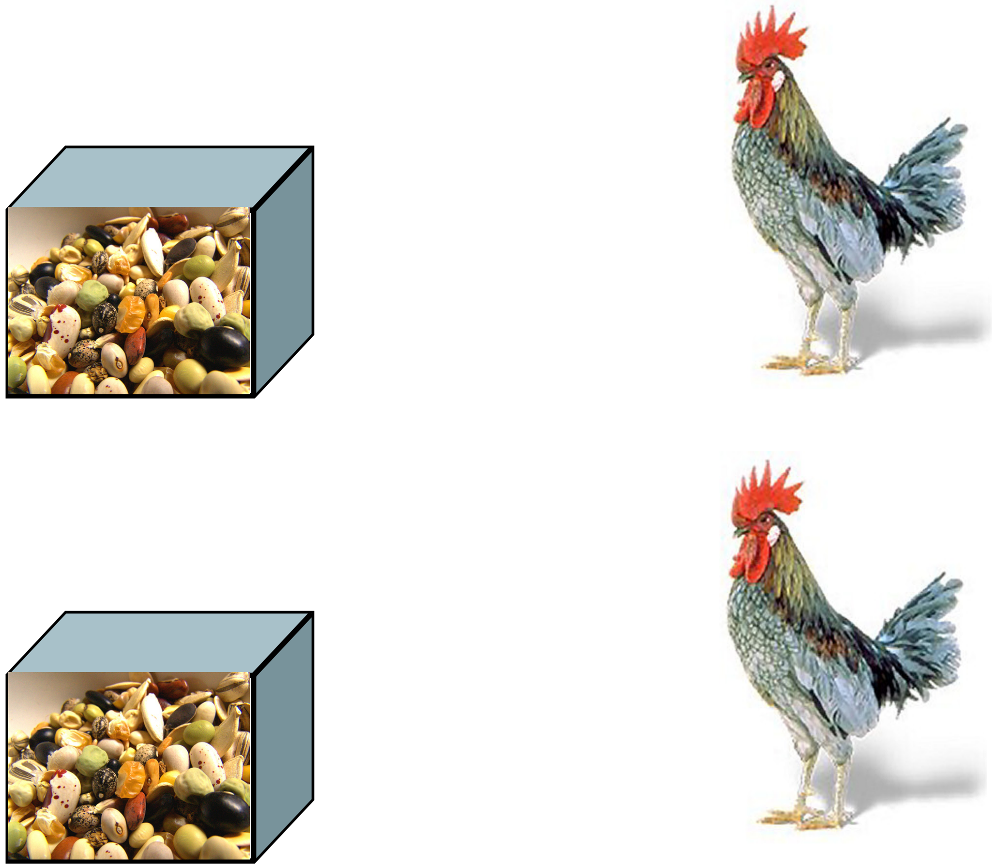
Hirsch and Bolles 1980

Unconditioned responses

- powerful
- inflexible over short timescale
- adaptive on evolutionary scale

Hershberger 1986

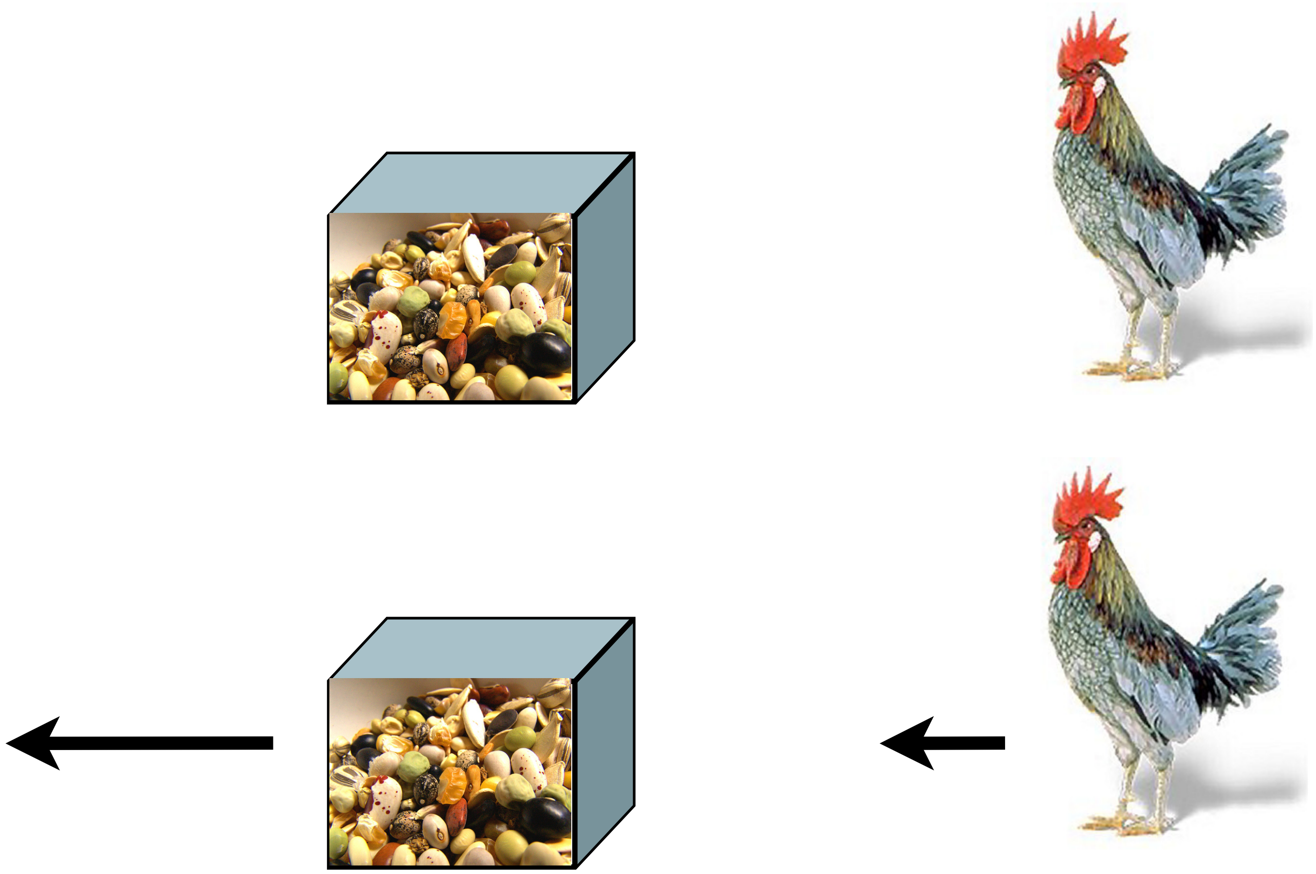
Unconditioned responses



- powerful
- inflexible over short timescale
- adaptive on evolutionary scale

Hershberger 1986

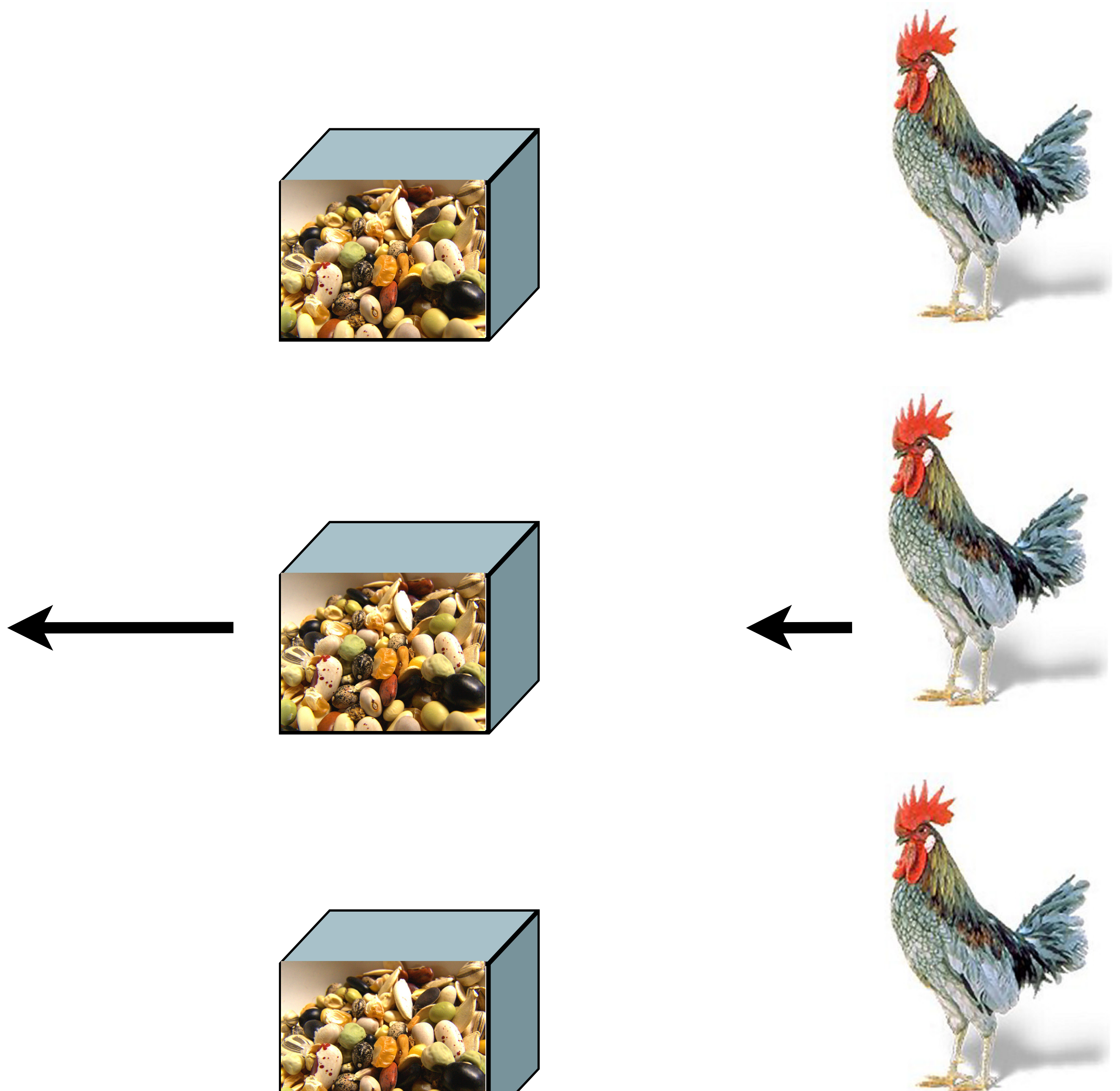
Unconditioned responses



- •powerful
- •inflexible over short timescale
- •adaptive on evolutionary scale

Hershberger 1986

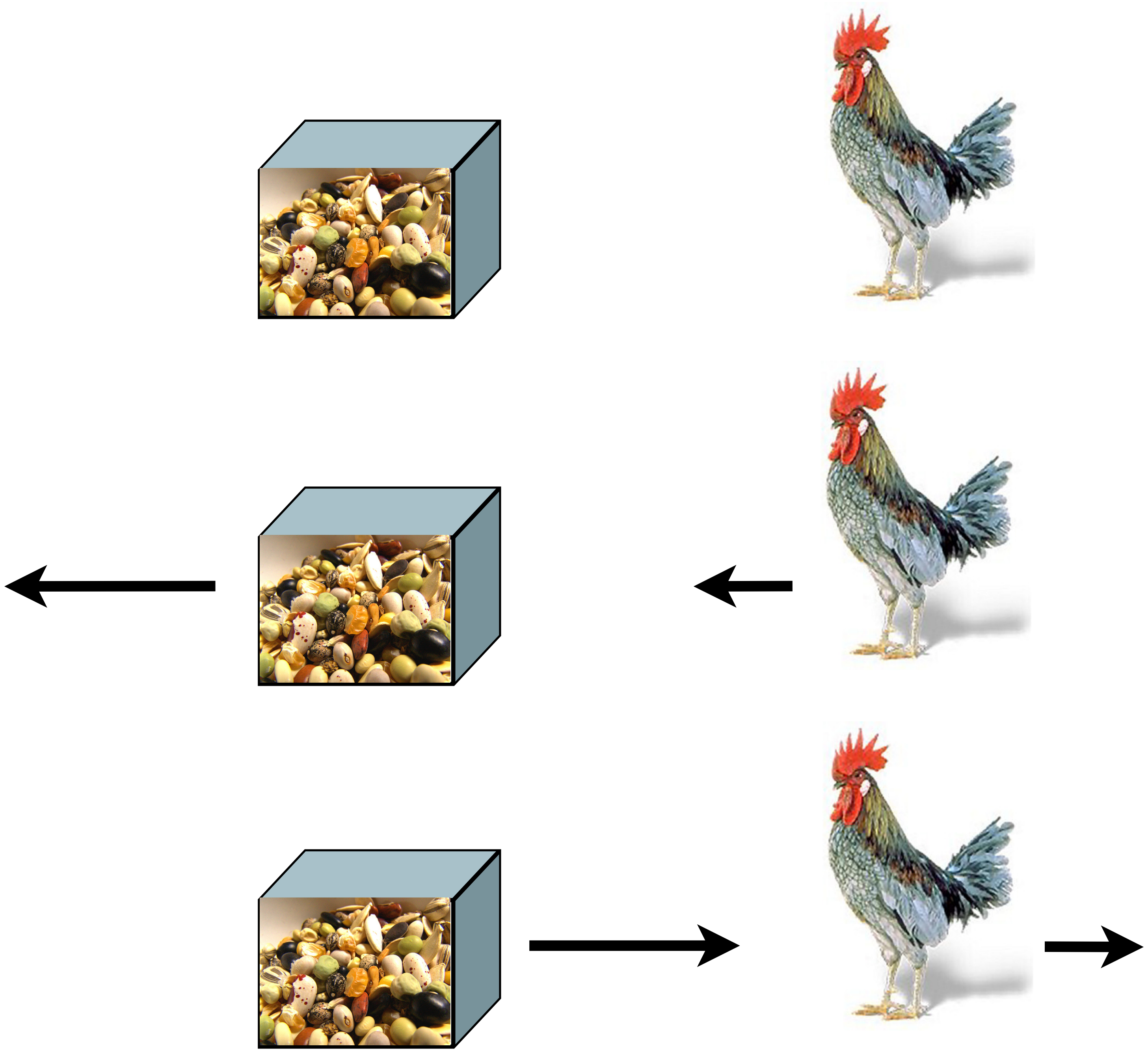
Unconditioned responses



- powerful
- inflexible over short timescale
- adaptive on evolutionary scale

Hershberger 1986

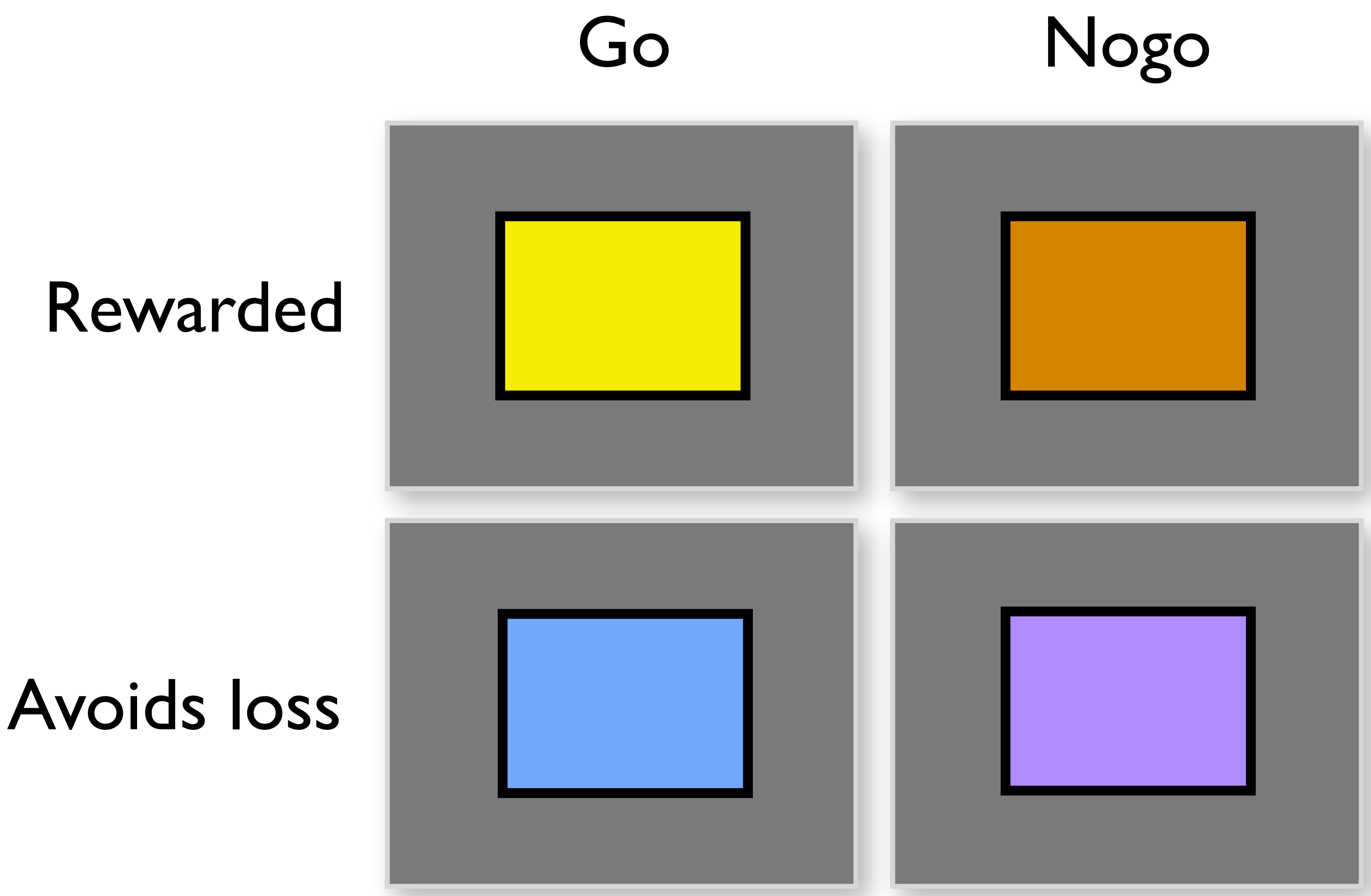
Unconditioned responses



Hershberger 1986

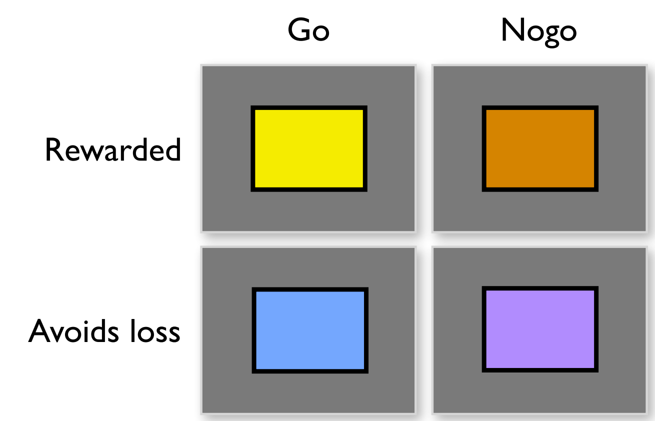
- powerful
- inflexible over short timescale
- adaptive on evolutionary scale

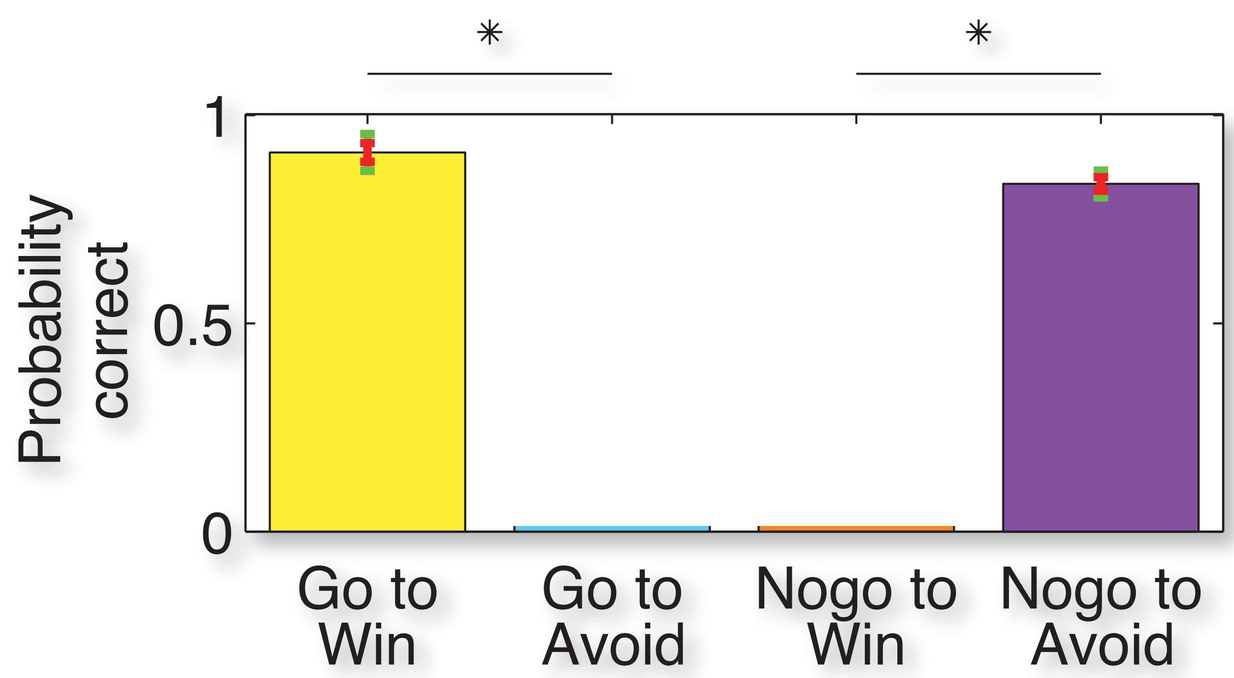
Affective go / nogo task

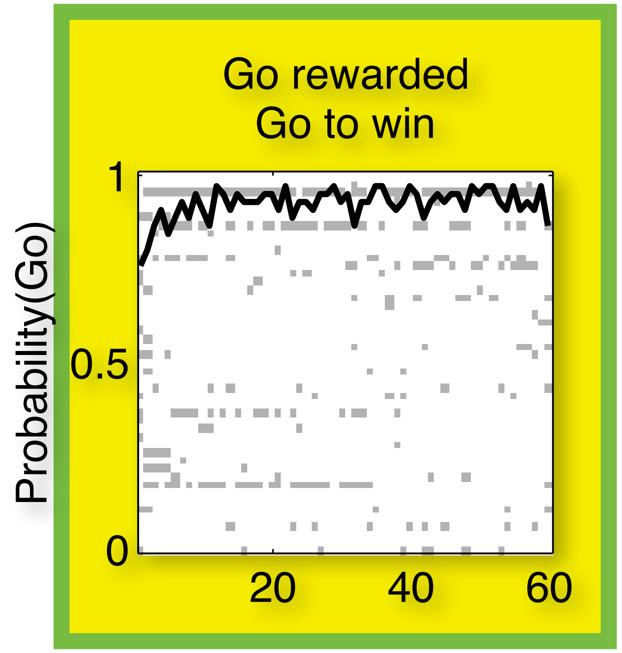


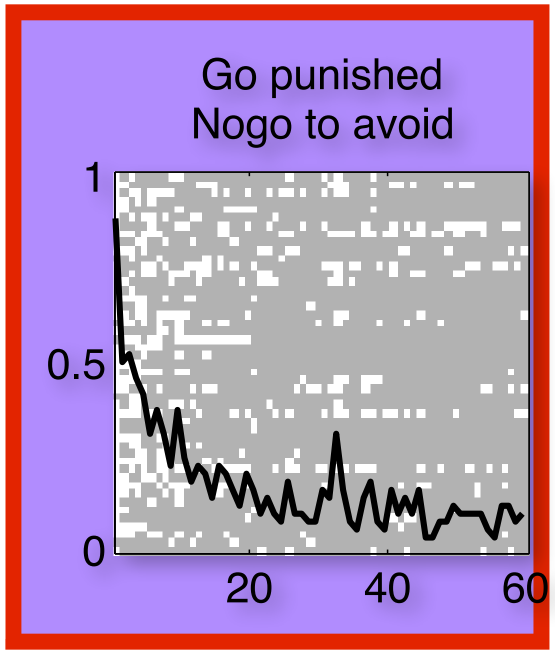
Guitart-Masip et al., 2012 J Neurosci

Affective go / nogo task



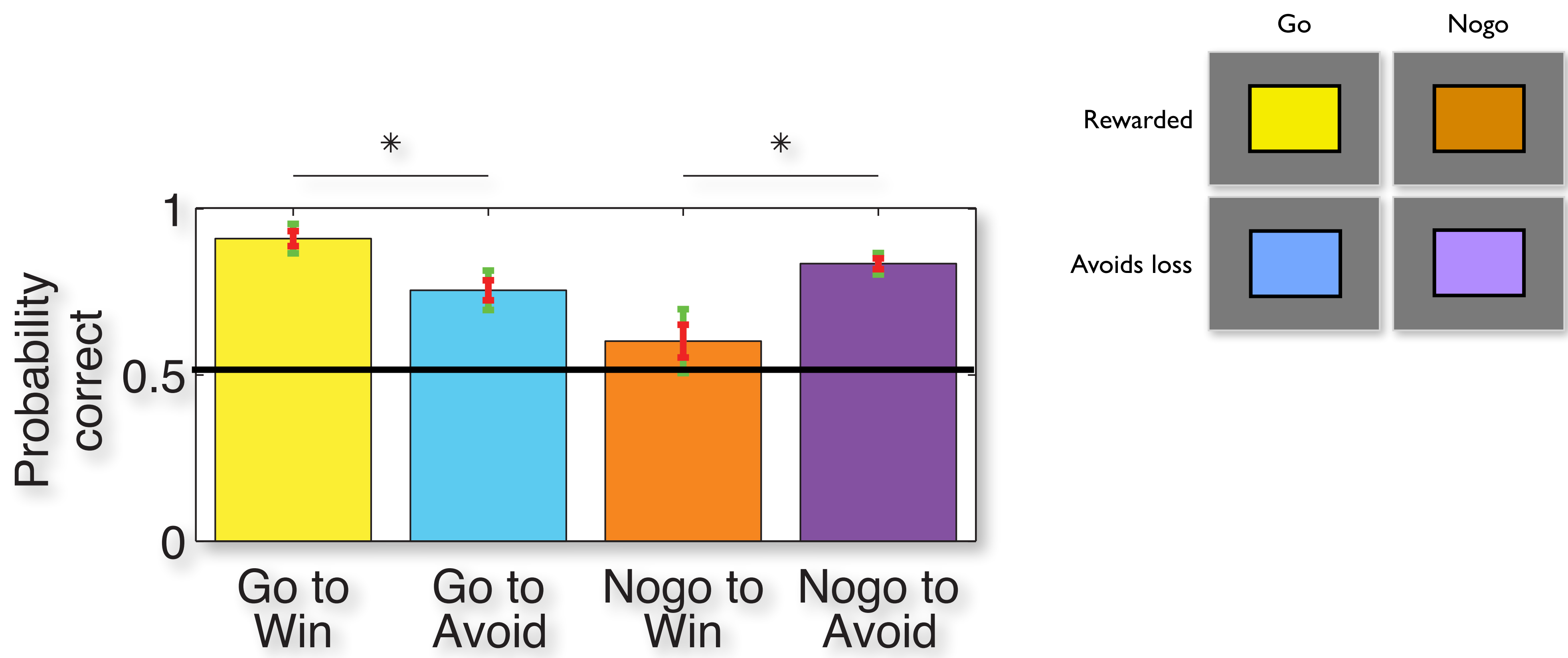


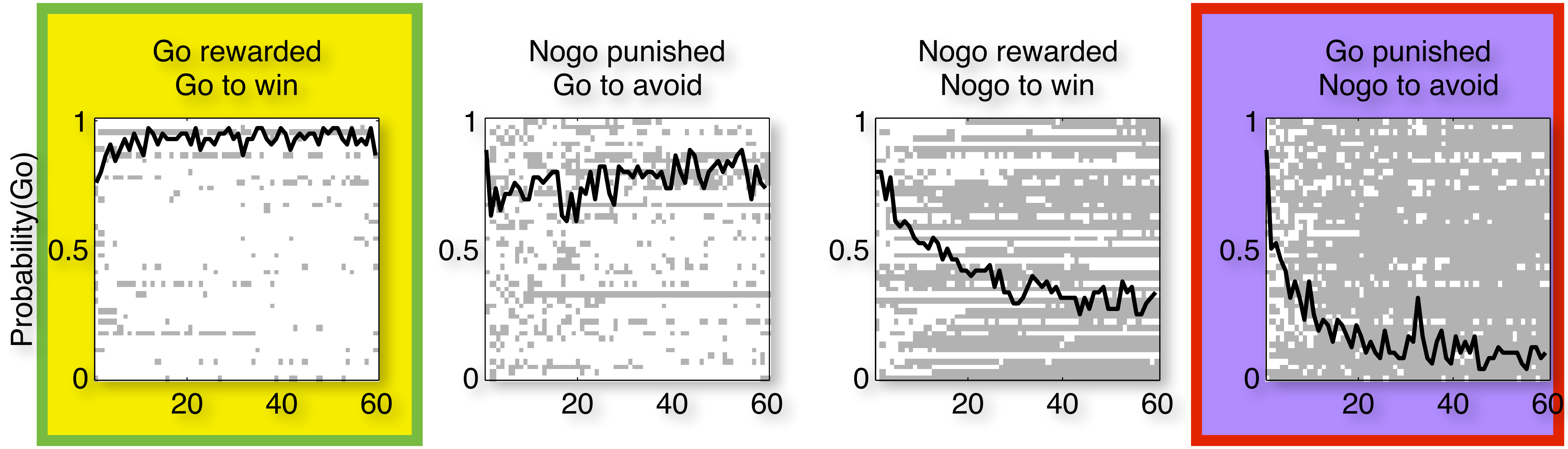




Guitart-Masip et al., 2012 J Neurosci

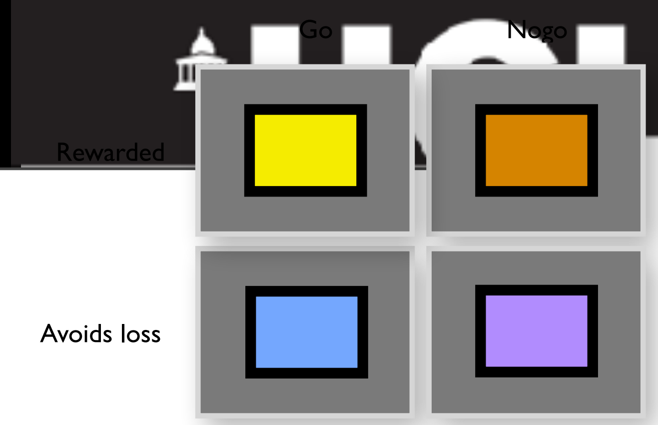
Affective go / nogo task





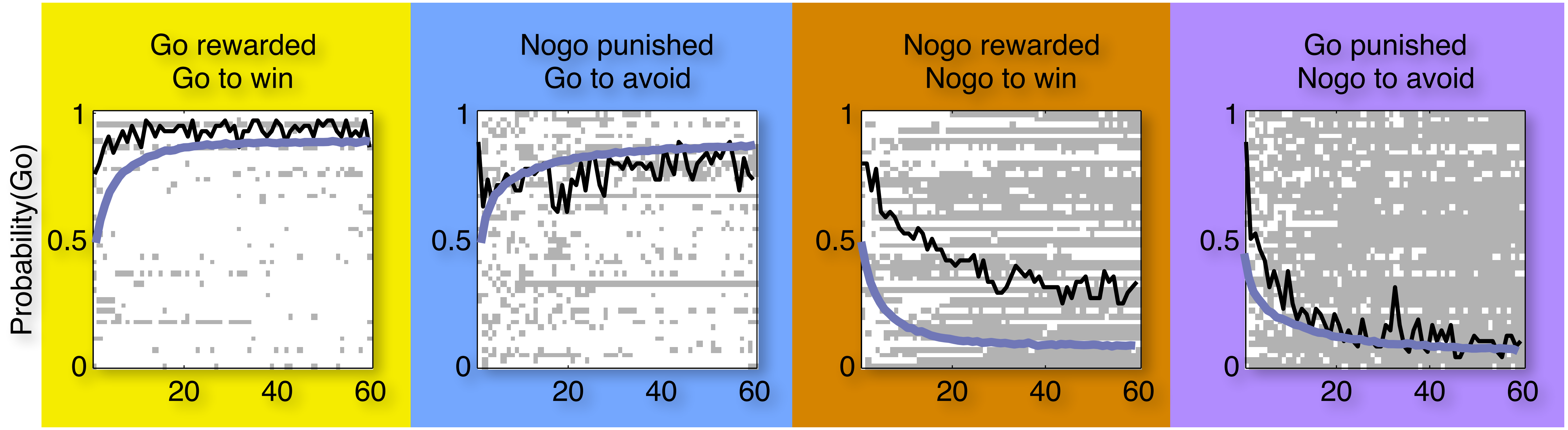
Guitart-Masip et al., 2012 J Neurosci

Models



- Instrumental

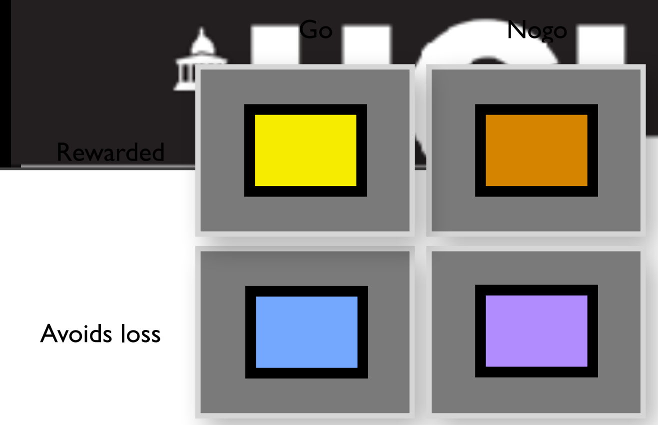
$$p_t(a|s) \propto \mathcal{Q}_t(s,a)$$
$$\mathcal{Q}_{t+1}(s,a) = \mathcal{Q}_t(s,a) + \alpha(r_t - \mathcal{Q}_t(s,a))$$

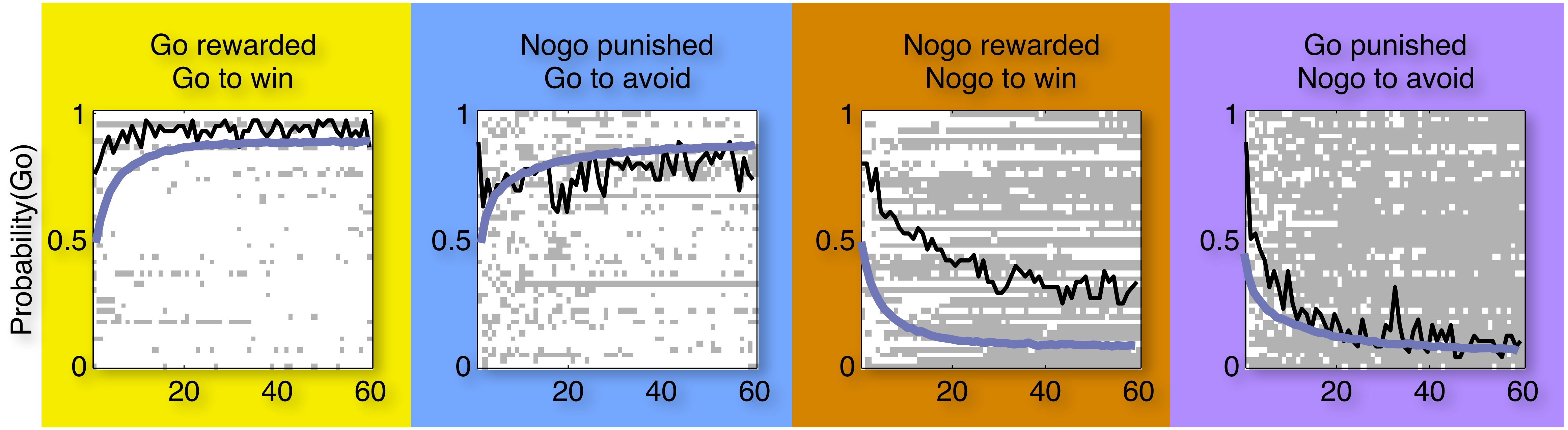


Guitart-Masip et al., 2012 J Neurosci

Models

- Instrumental + bias

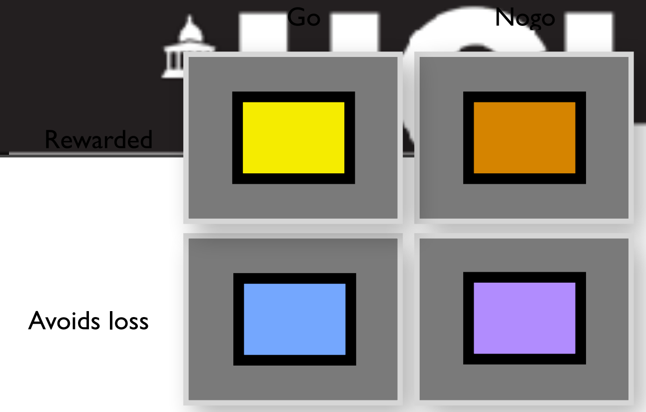


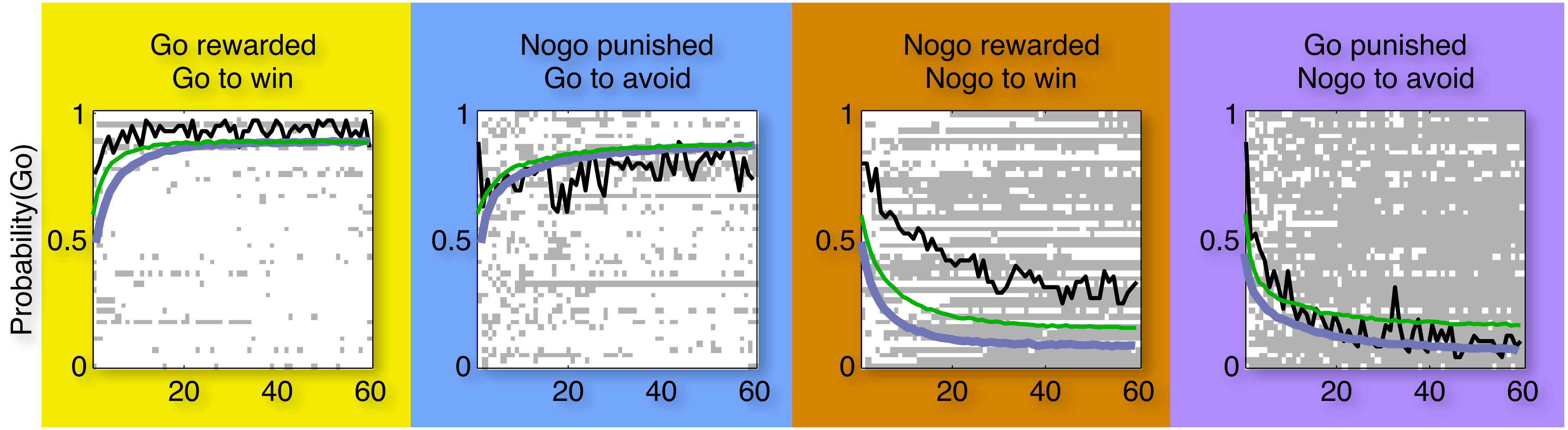


Guitart-Masip et al., 2012 J Neurosci

Models

- Instrumental + bias

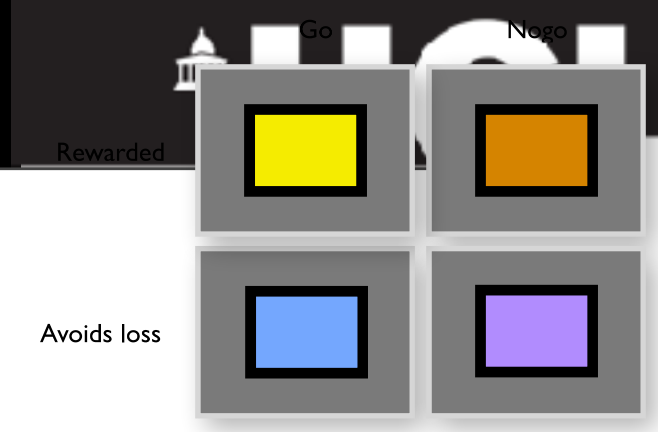


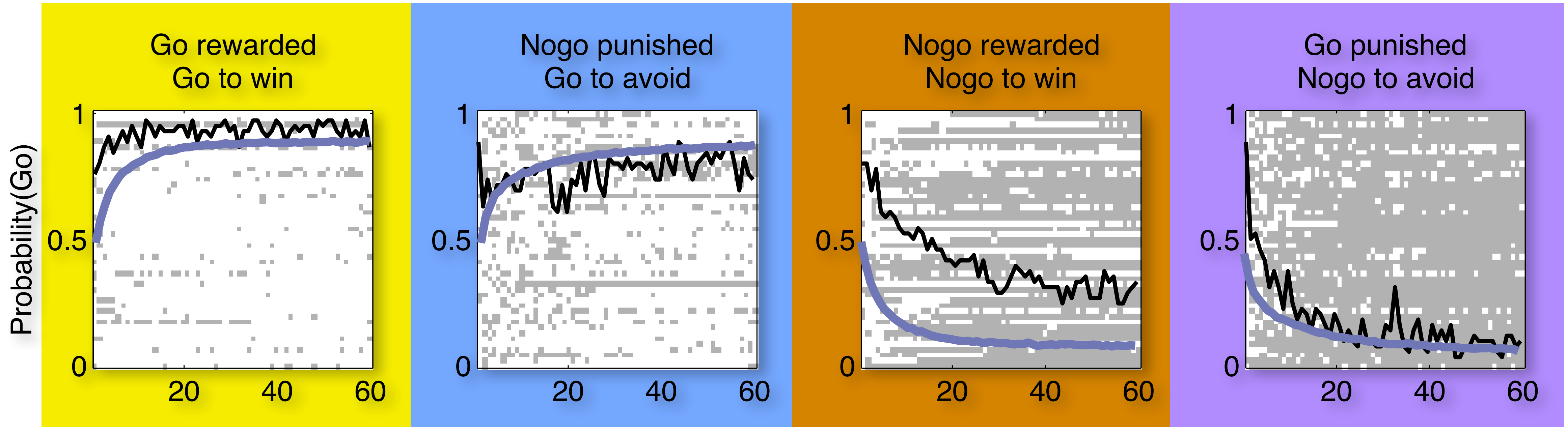


Guitart-Masip et al., 2012 J Neurosci

Models

- Instrumental + bias + Pavlovian

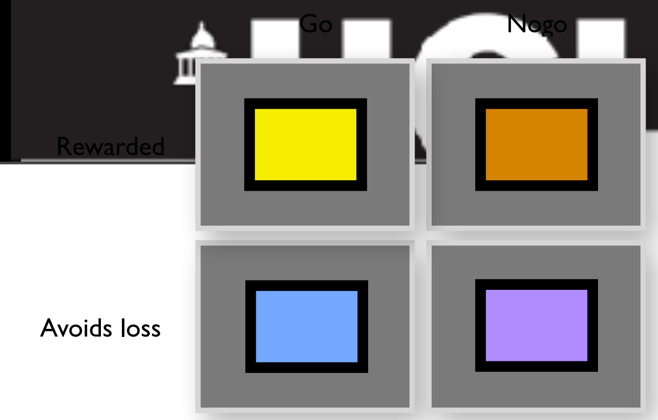


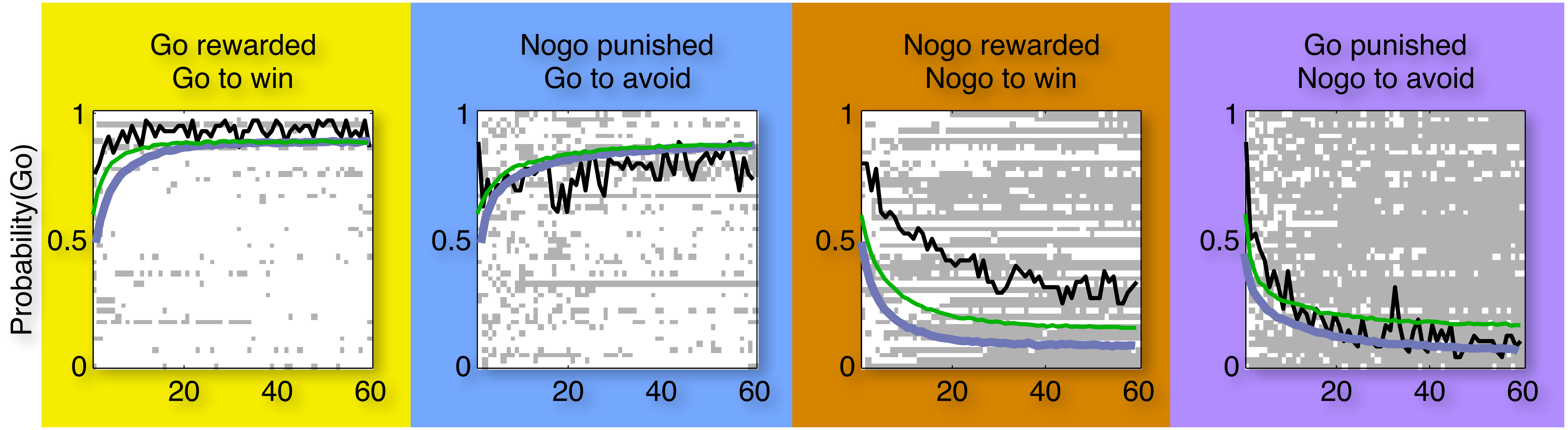


Guitart-Masip et al., 2012 J Neurosci

Models

- Instrumental + bias + Pavlovian

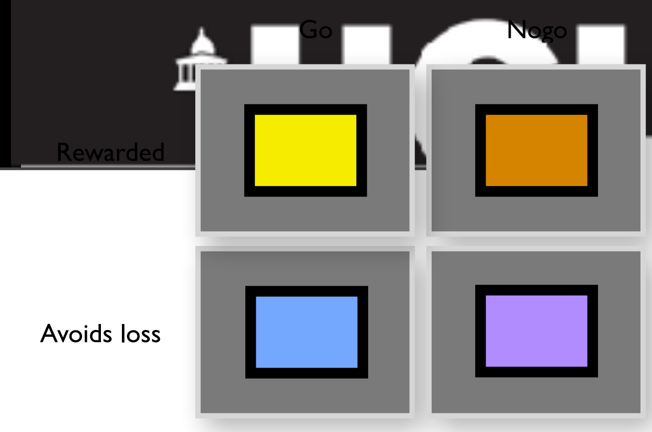


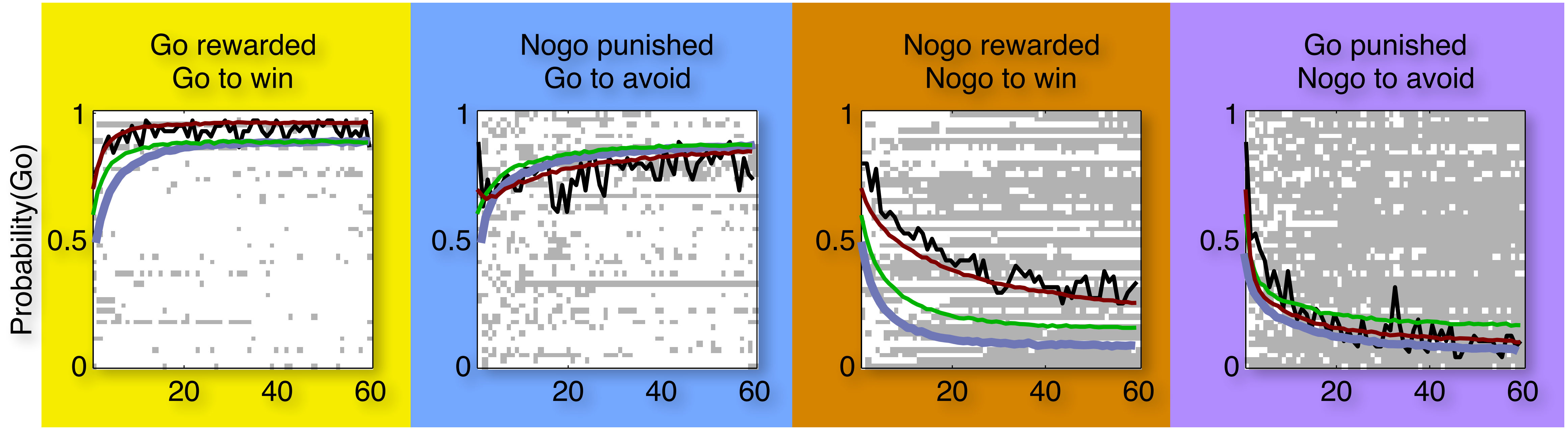


Guitart-Masip et al., 2012 J Neurosci

Models

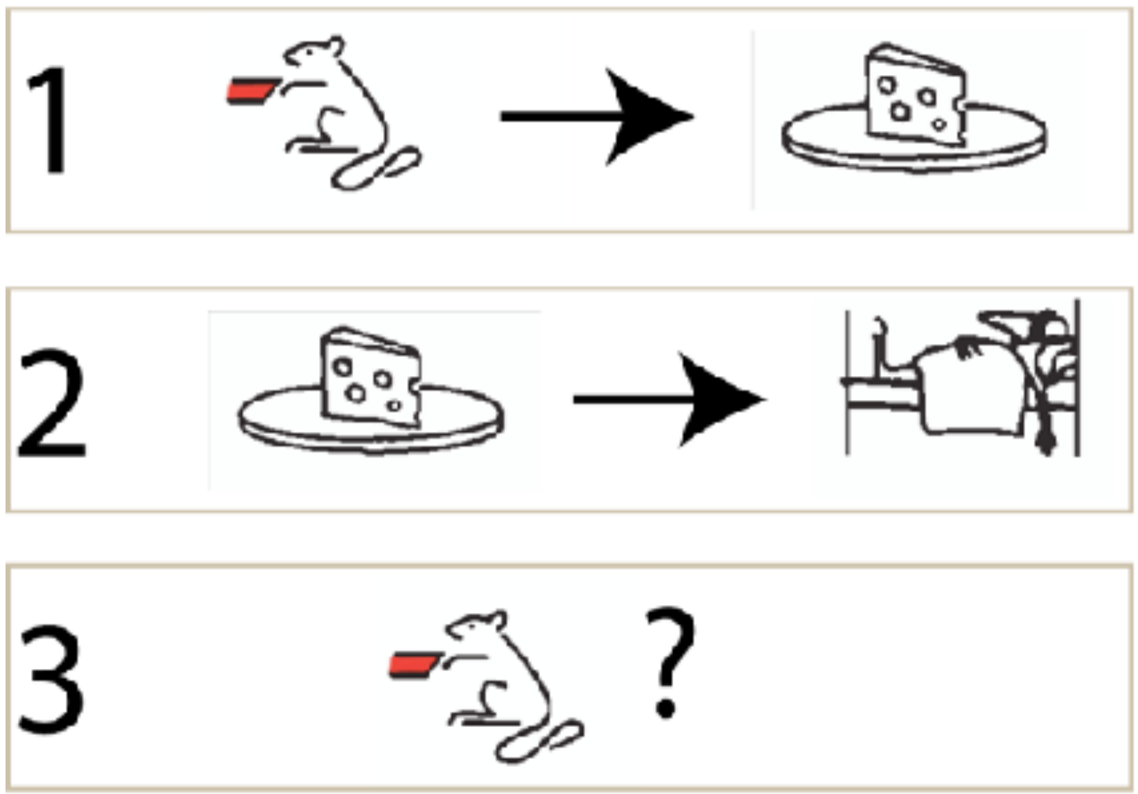
- Instrumental + bias + Pavlovian

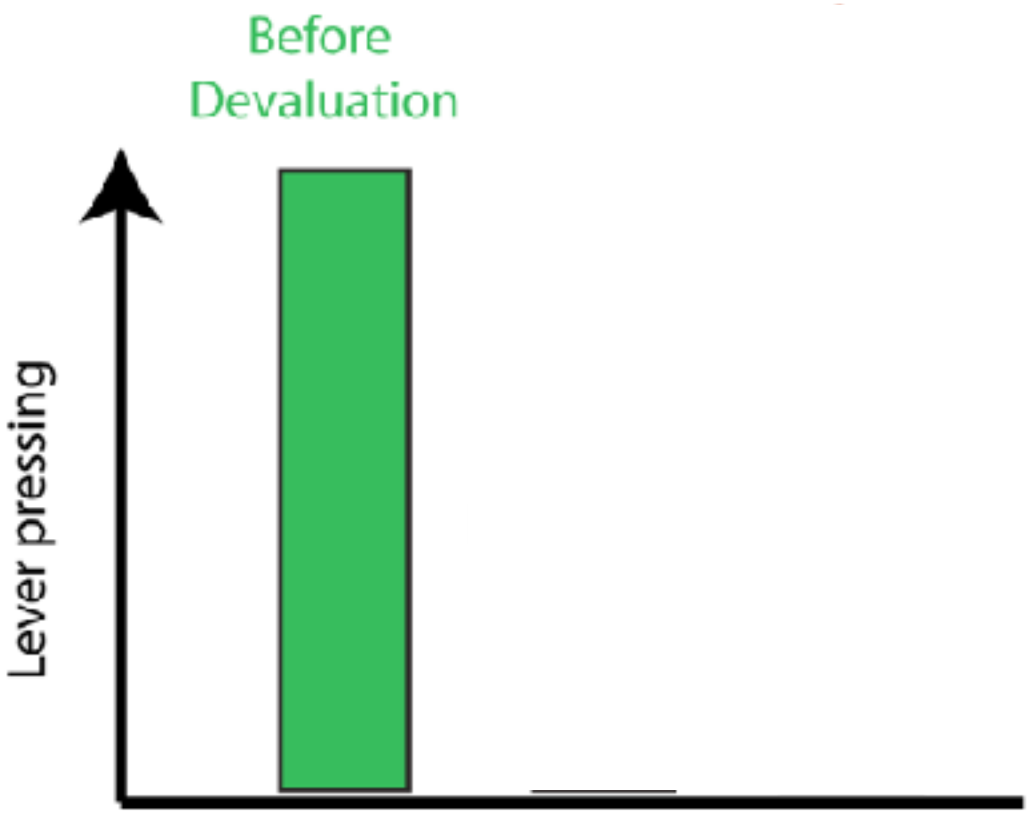




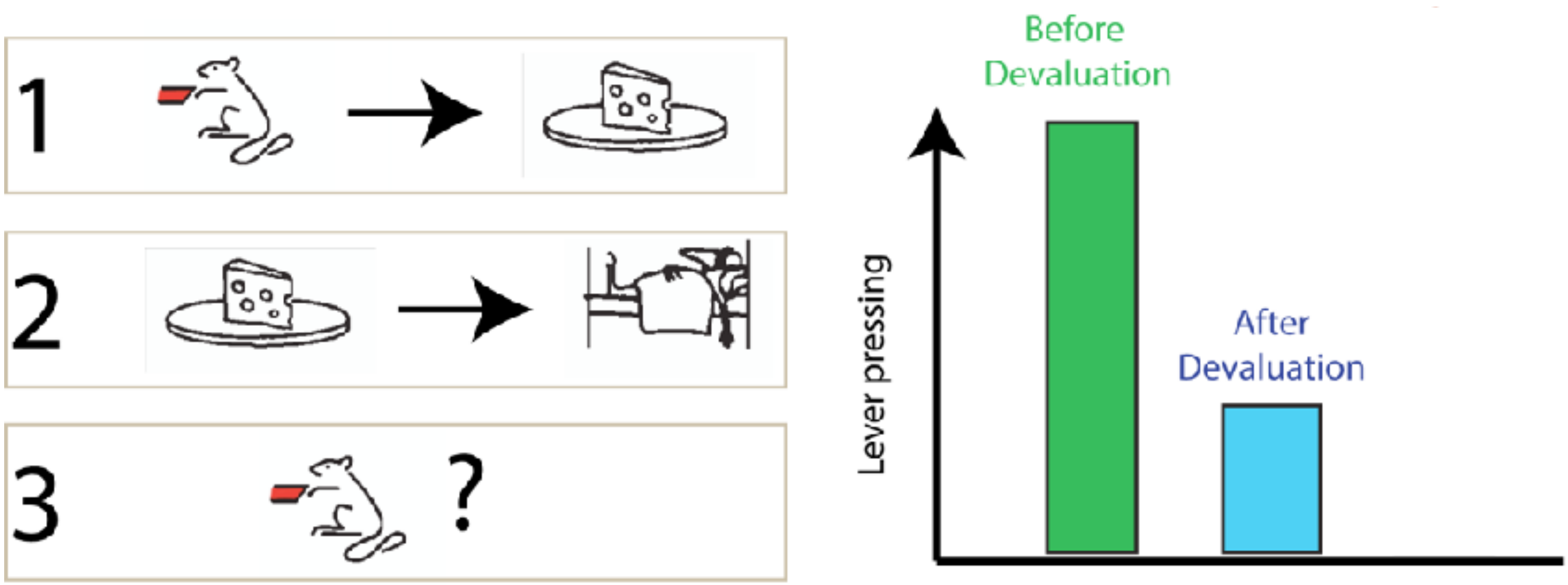
Guitart-Masip et al., 2012 J Neurosci

Habitization

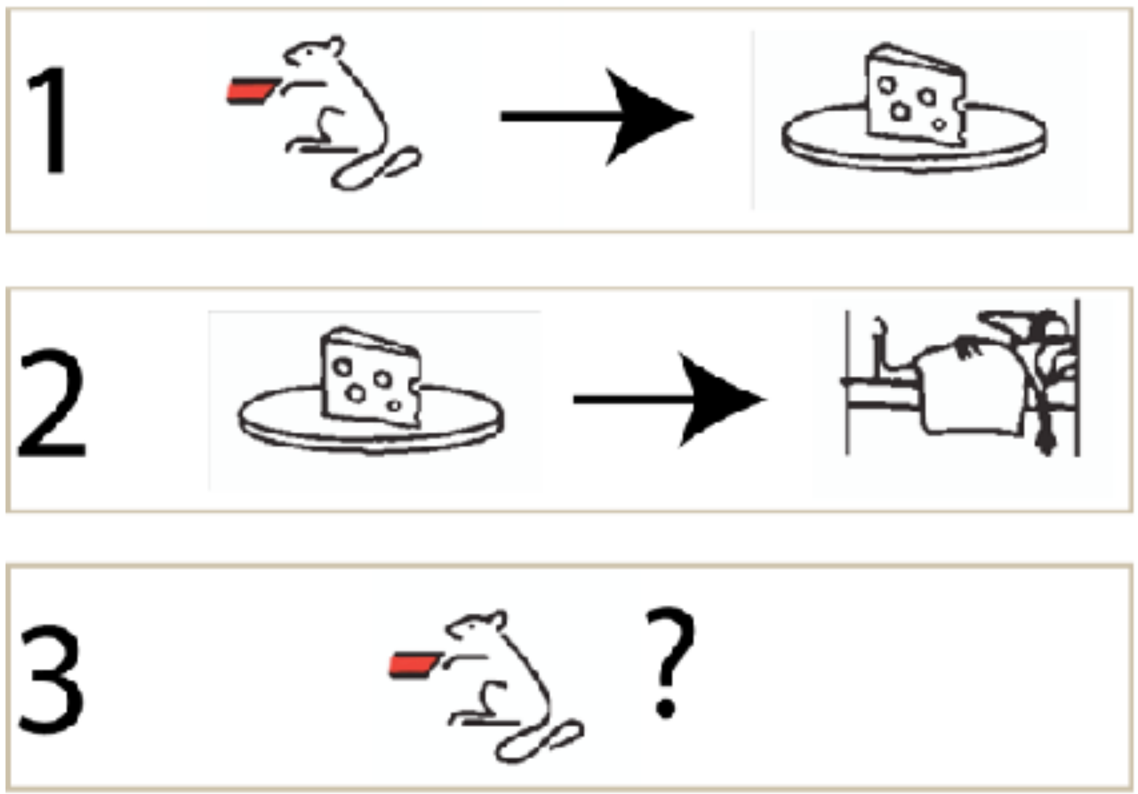


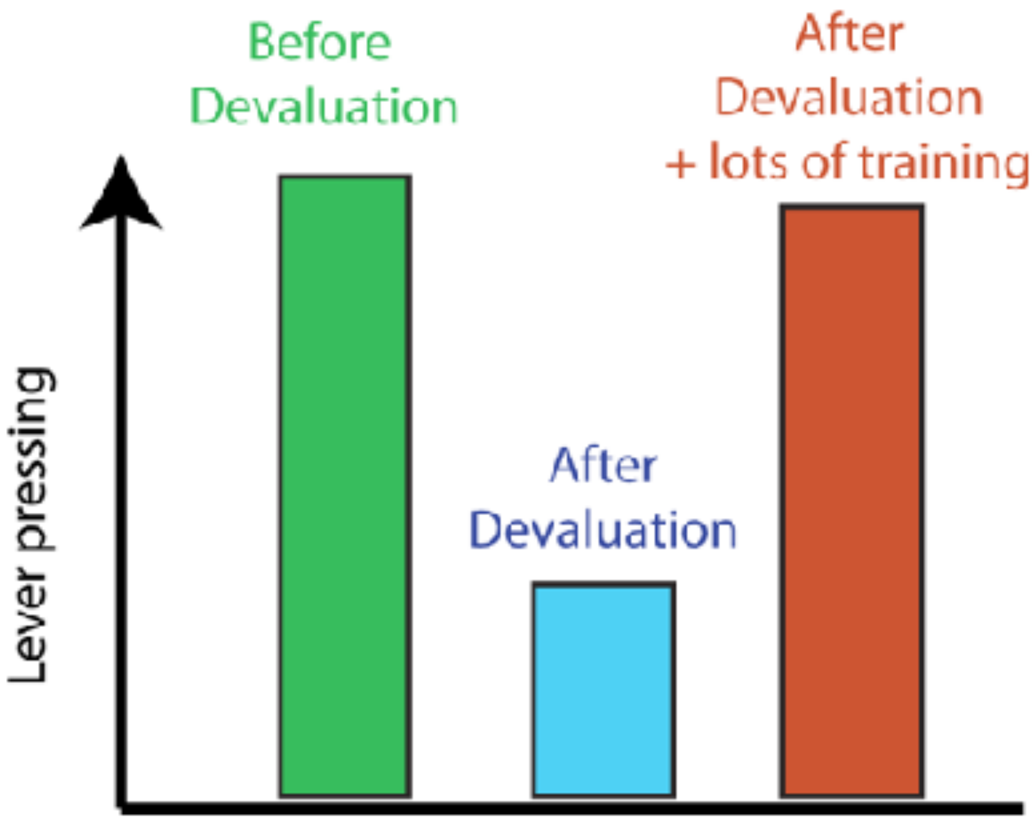


Habitization

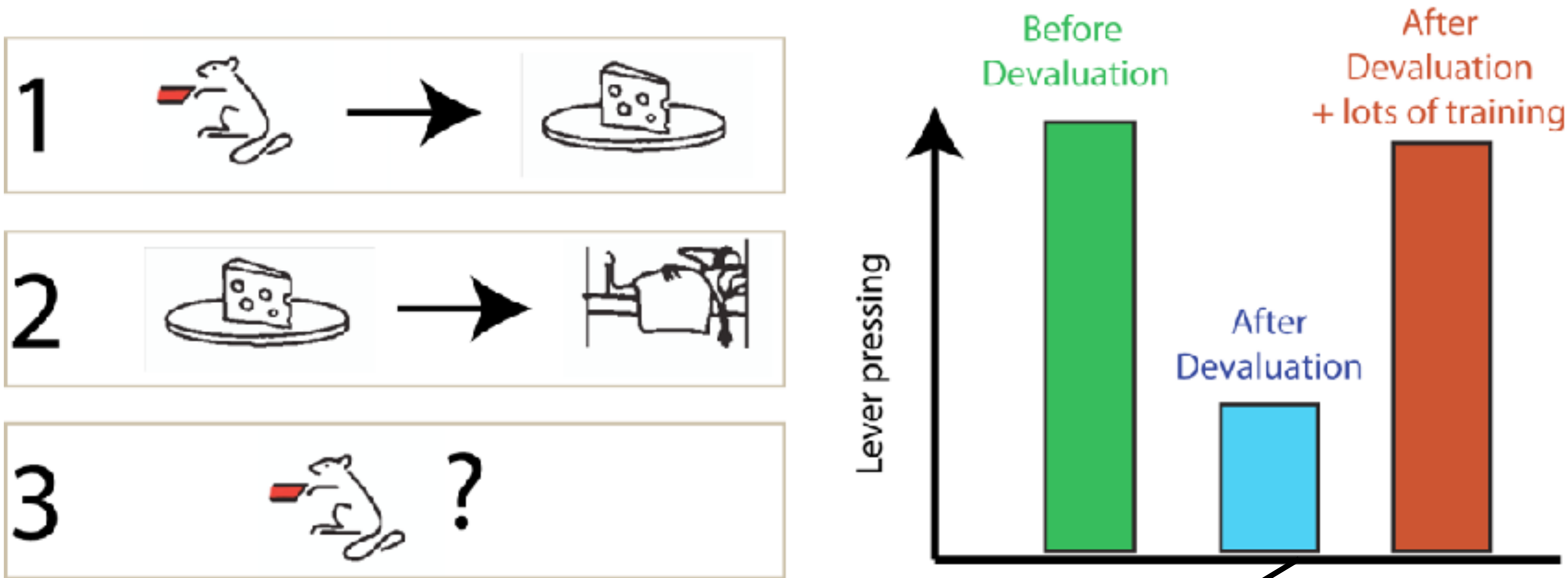


Habitization



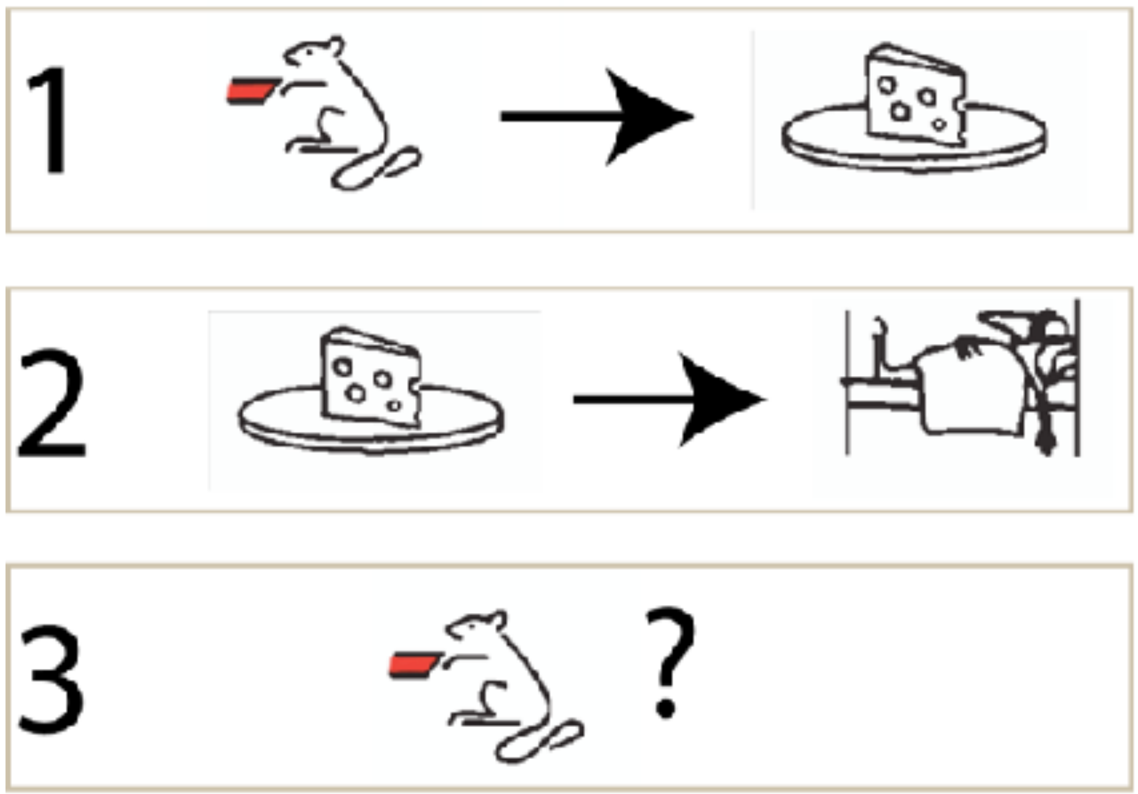


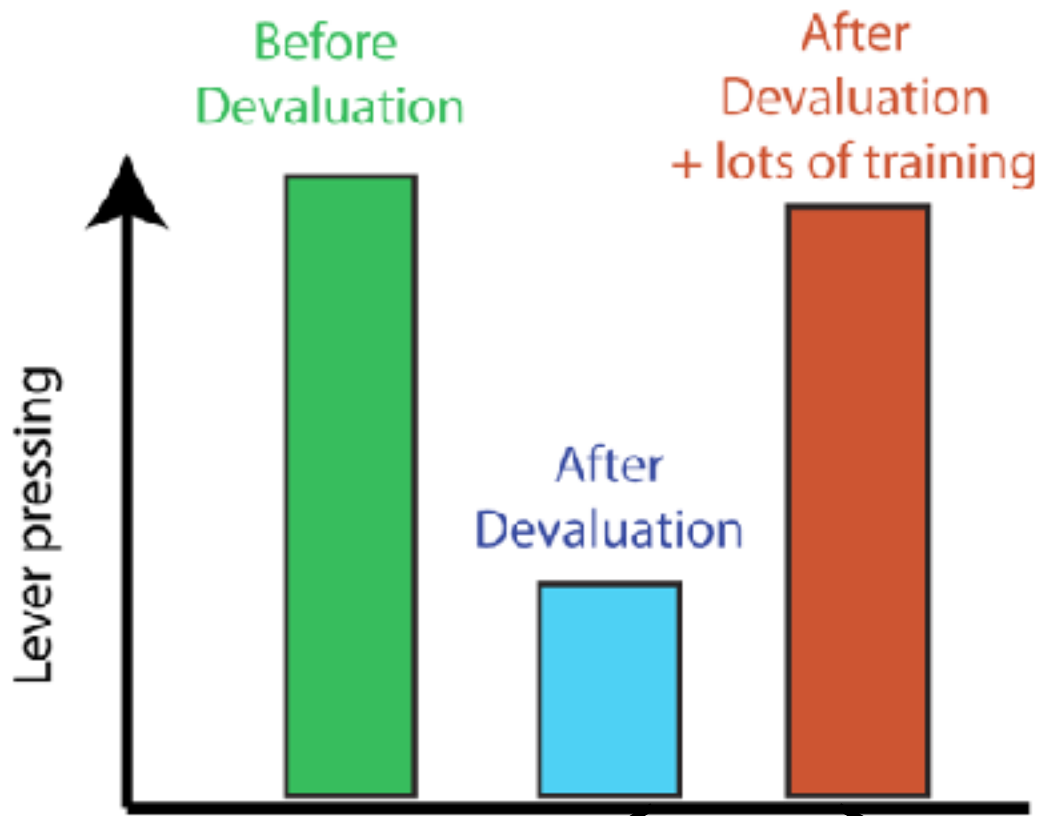
Habitization



Get this pattern late if lesion infralimbic cortex

Habitization

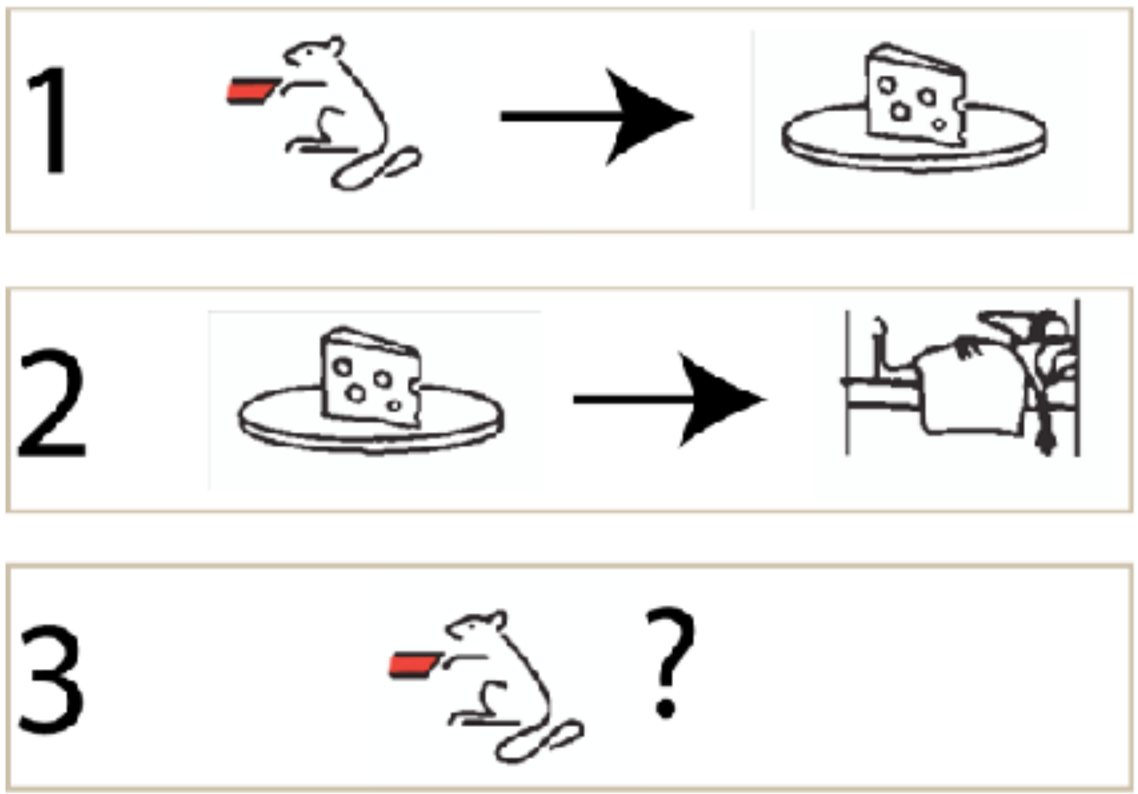


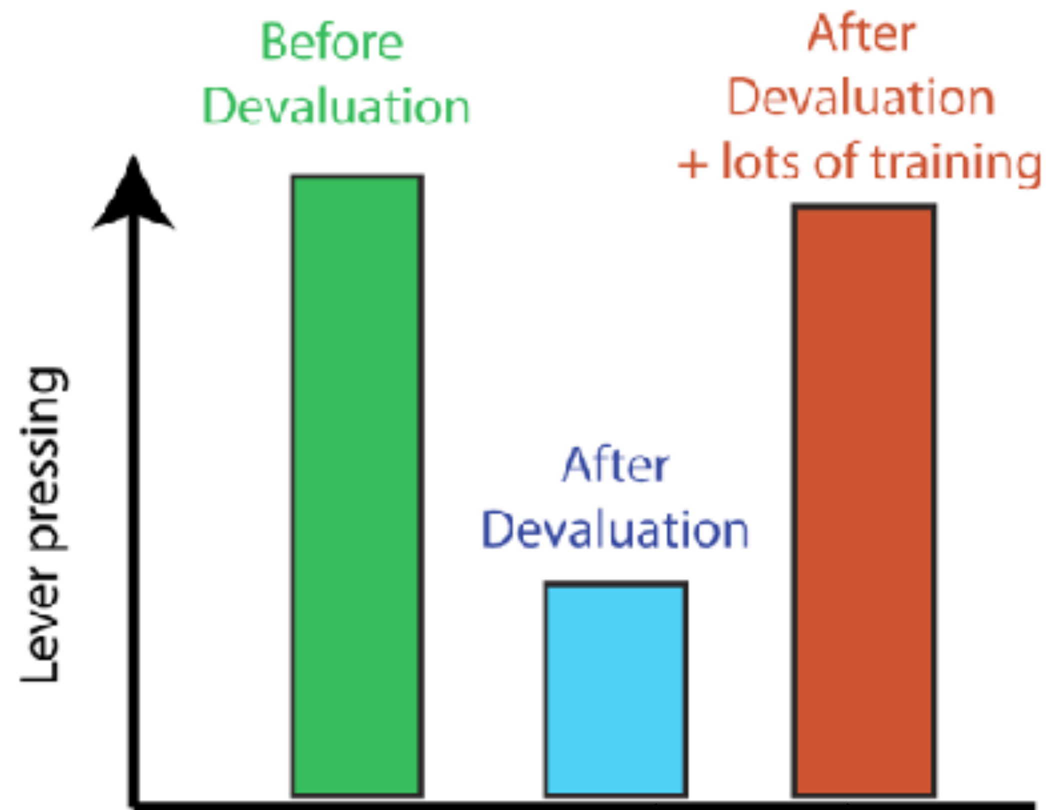


Get this pattern late if lesion infralimbic cortex

Get this pattern early if lesion prelimbic cortex

Habitization



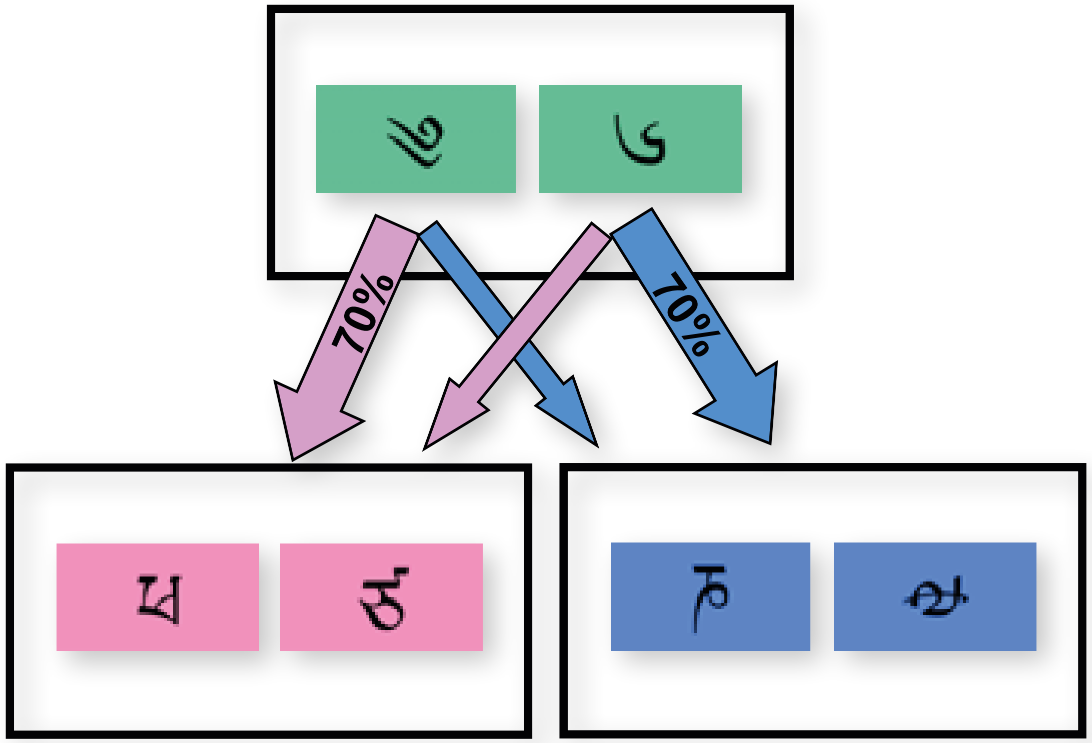


Get this pattern late if lesion infralimbic cortex

Get this pattern early if lesion prelimbic cortex

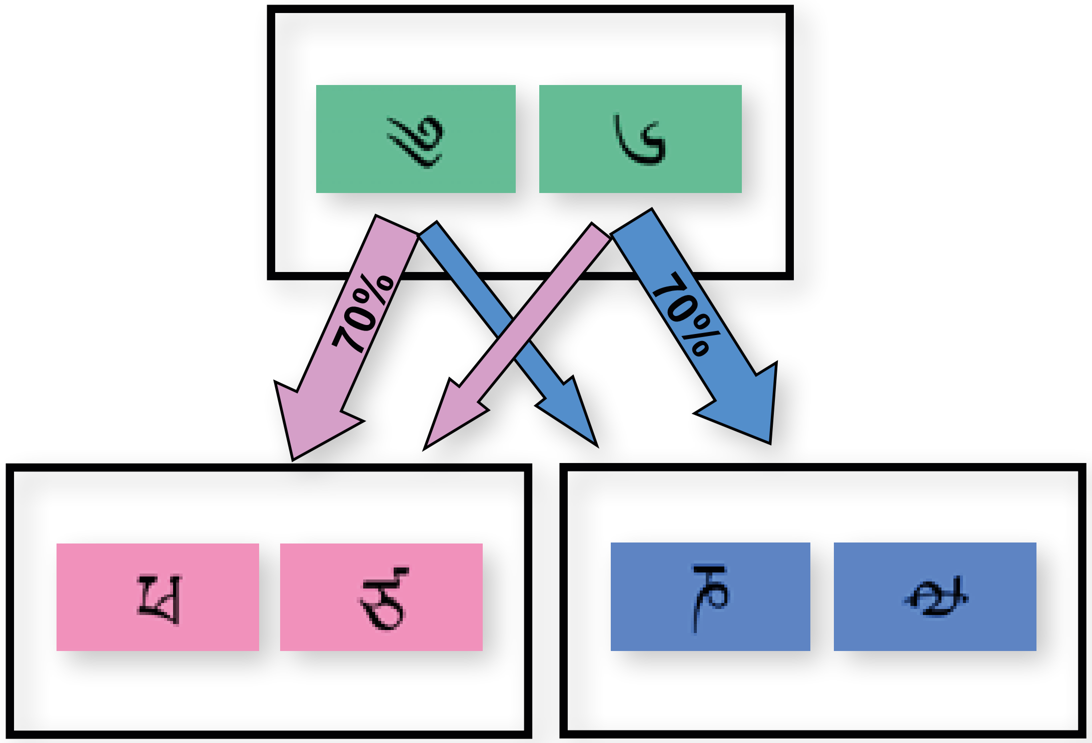
PEL ETOH

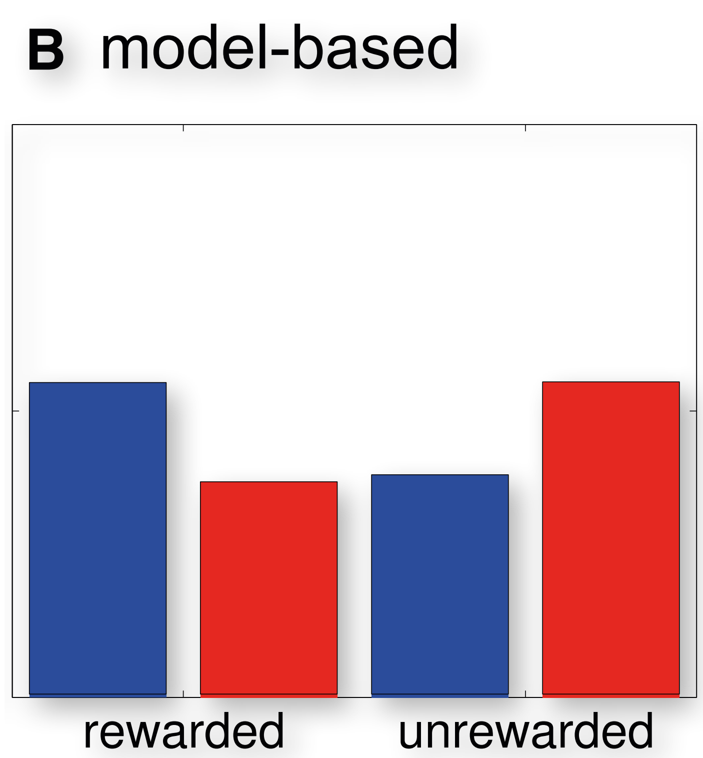
Two-step task



Daw et al. 2011, Neuron

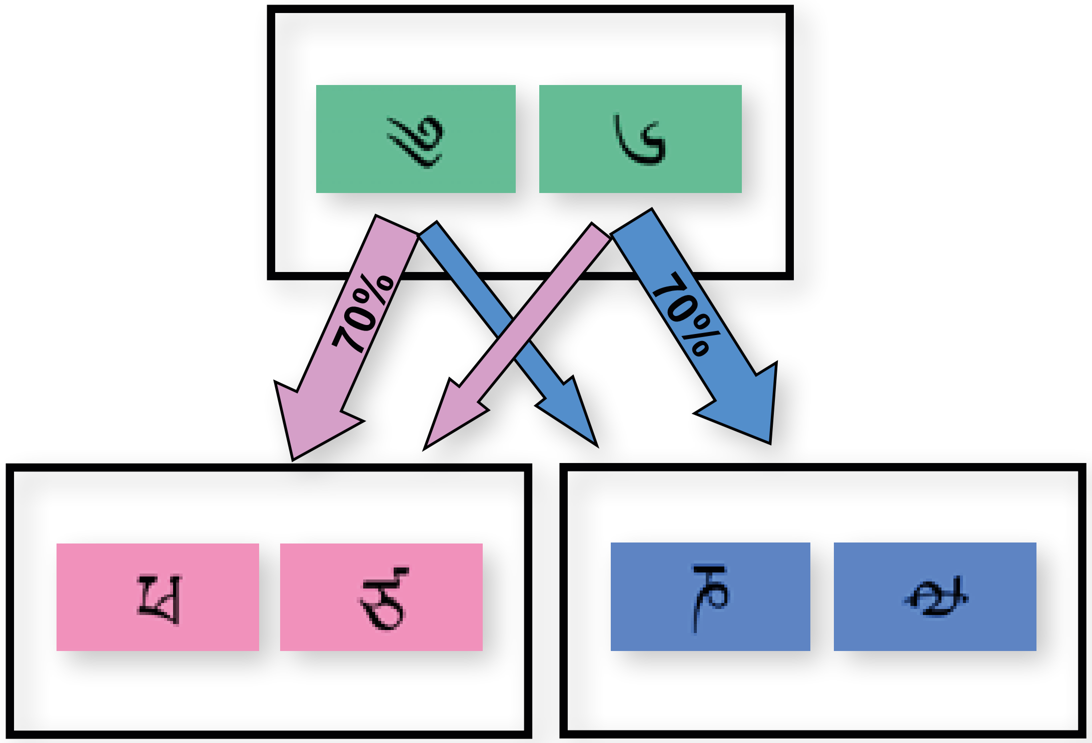
Two-step task

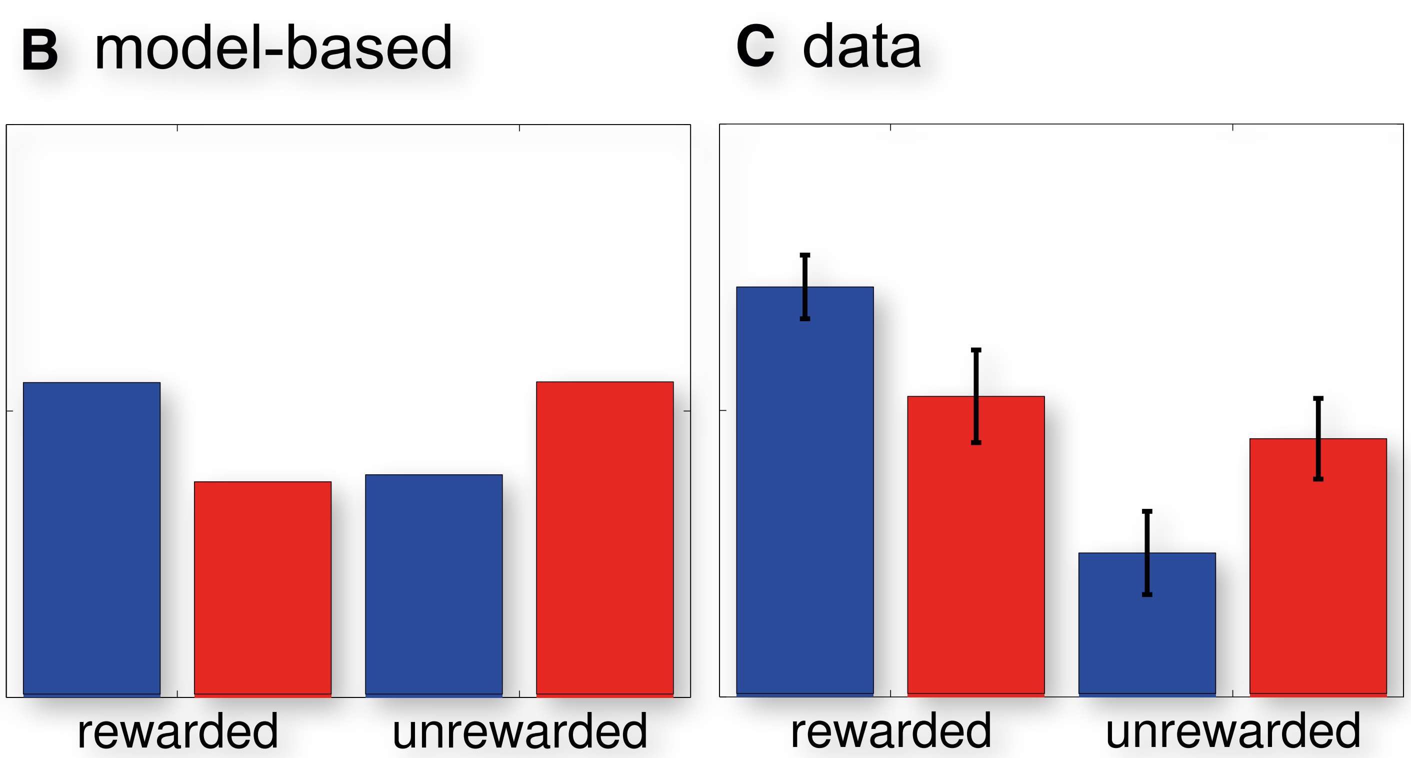




Daw et al. 2011, Neuron

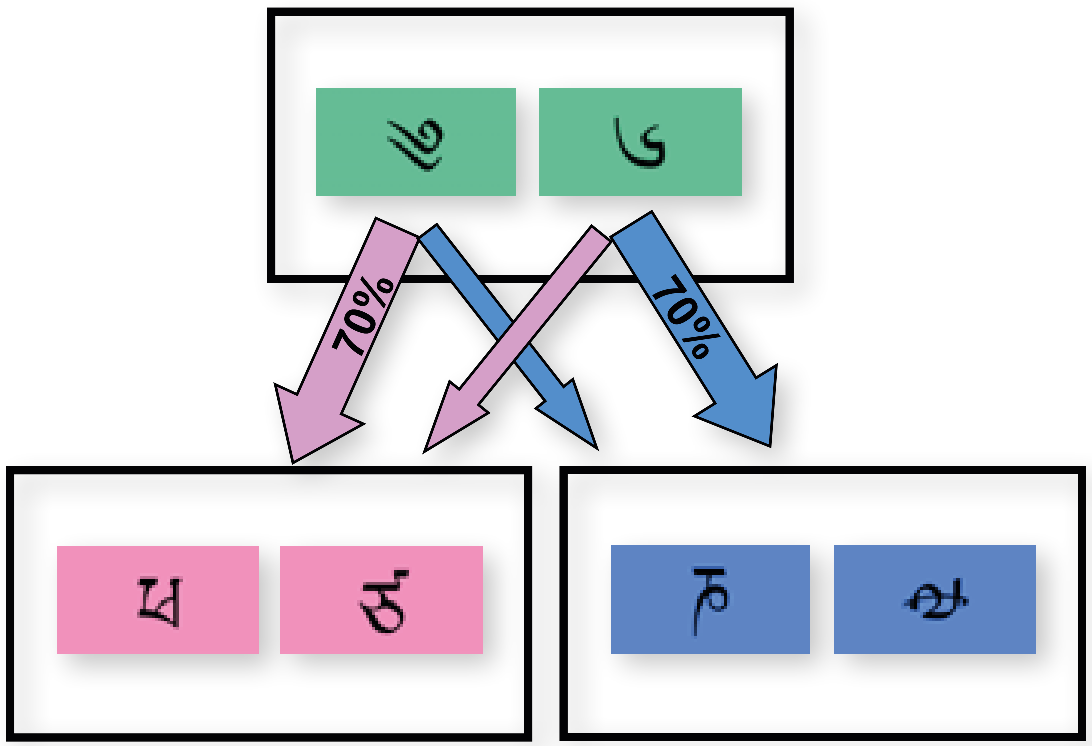
Two-step task

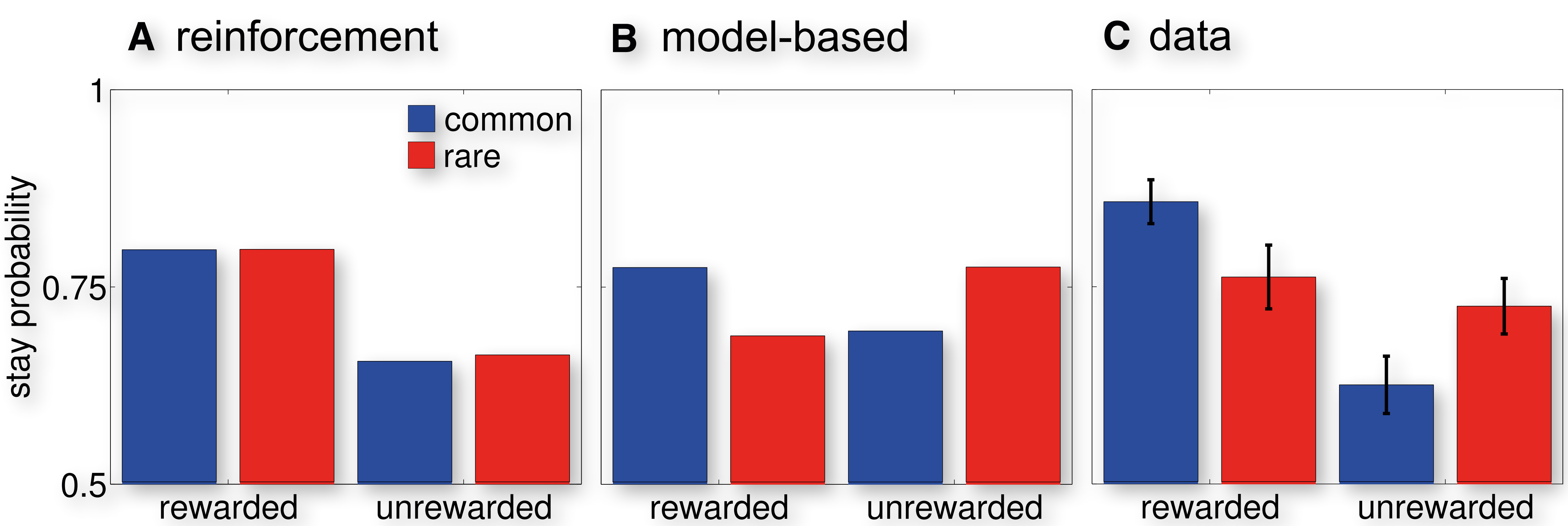




Daw et al. 2011, Neuron

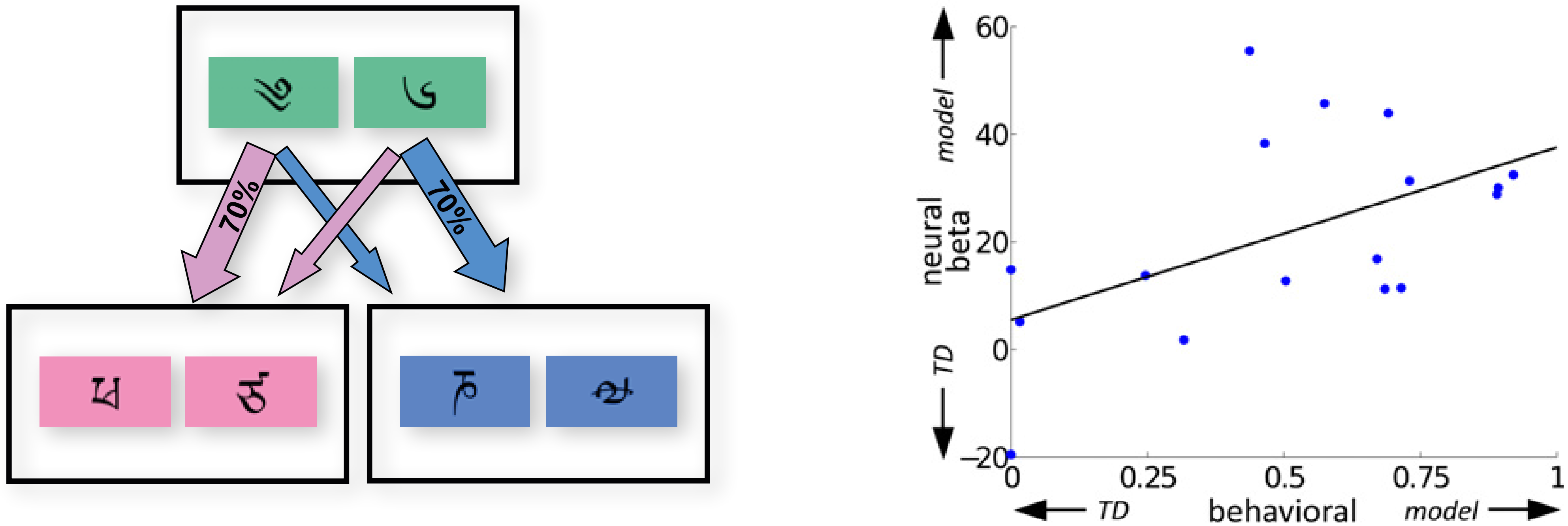
Two-step task

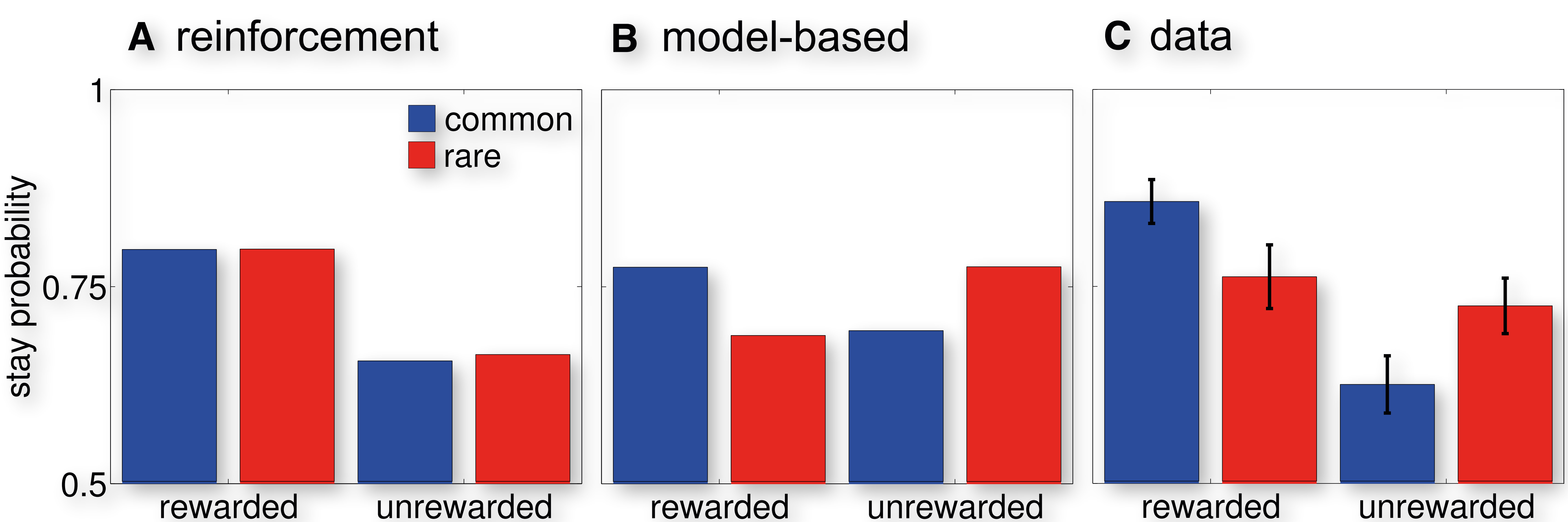




Daw et al. 2011, Neuron

Fault line 1: Balance of cached and goal



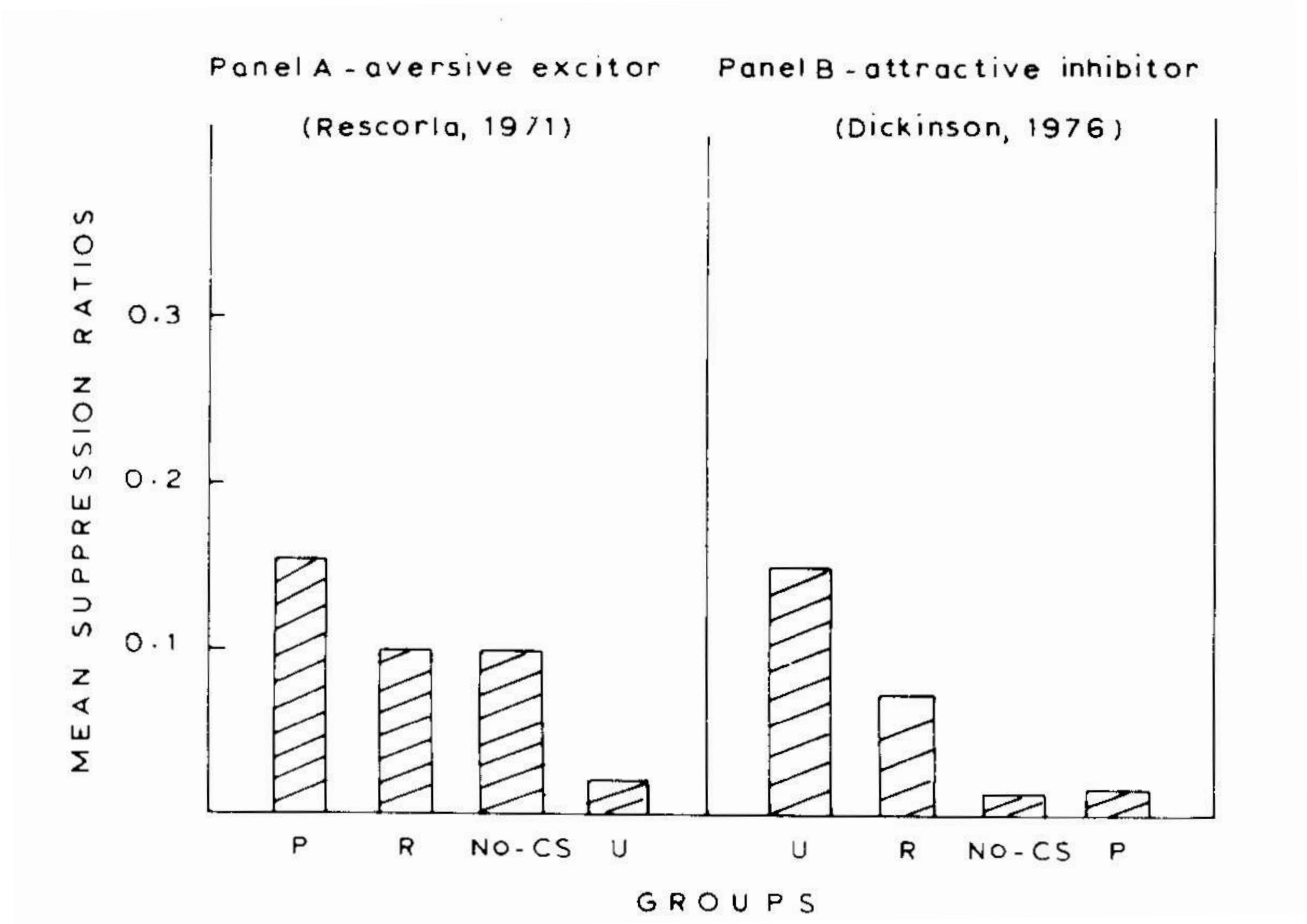


Daw et al. 2011, Neuron

Value matters - transreinforcer blocking

TABLE 8.4
Blocking of aversive conditioning

Conditions	*Stage 1*	*Stage 2*	*test*
1	A → shock	AX → shock	X
2	control treatments	AX → shock	X
3	A → food ommission	AX → shock	X

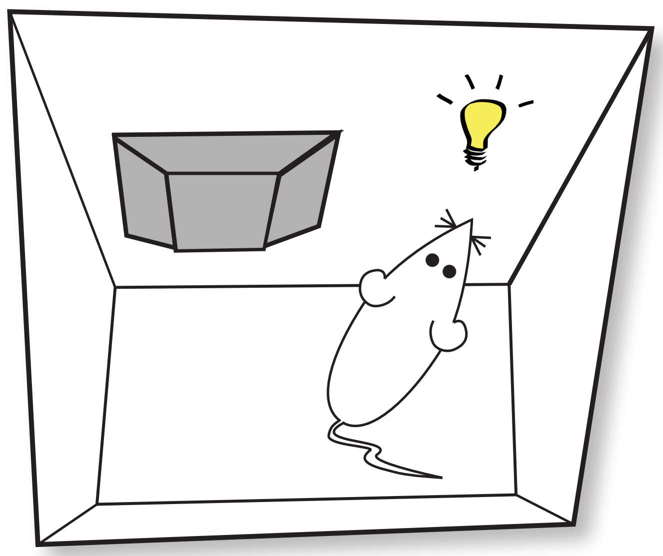


"bad"
vs
"good"

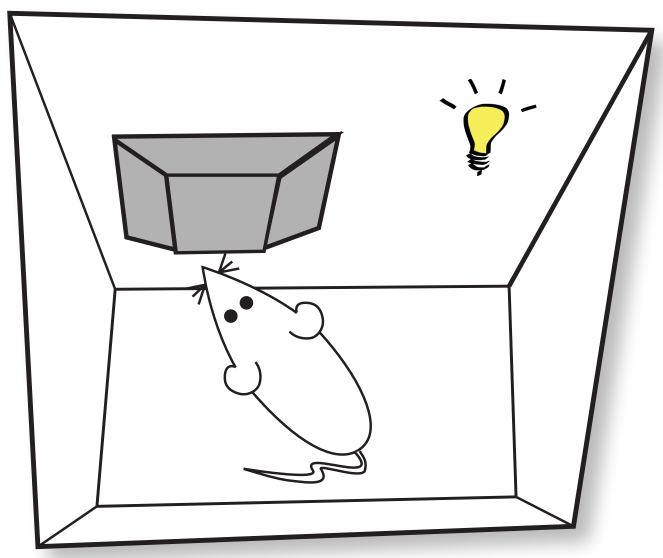
Dickinson and Dearing 1979

Signtracking

A

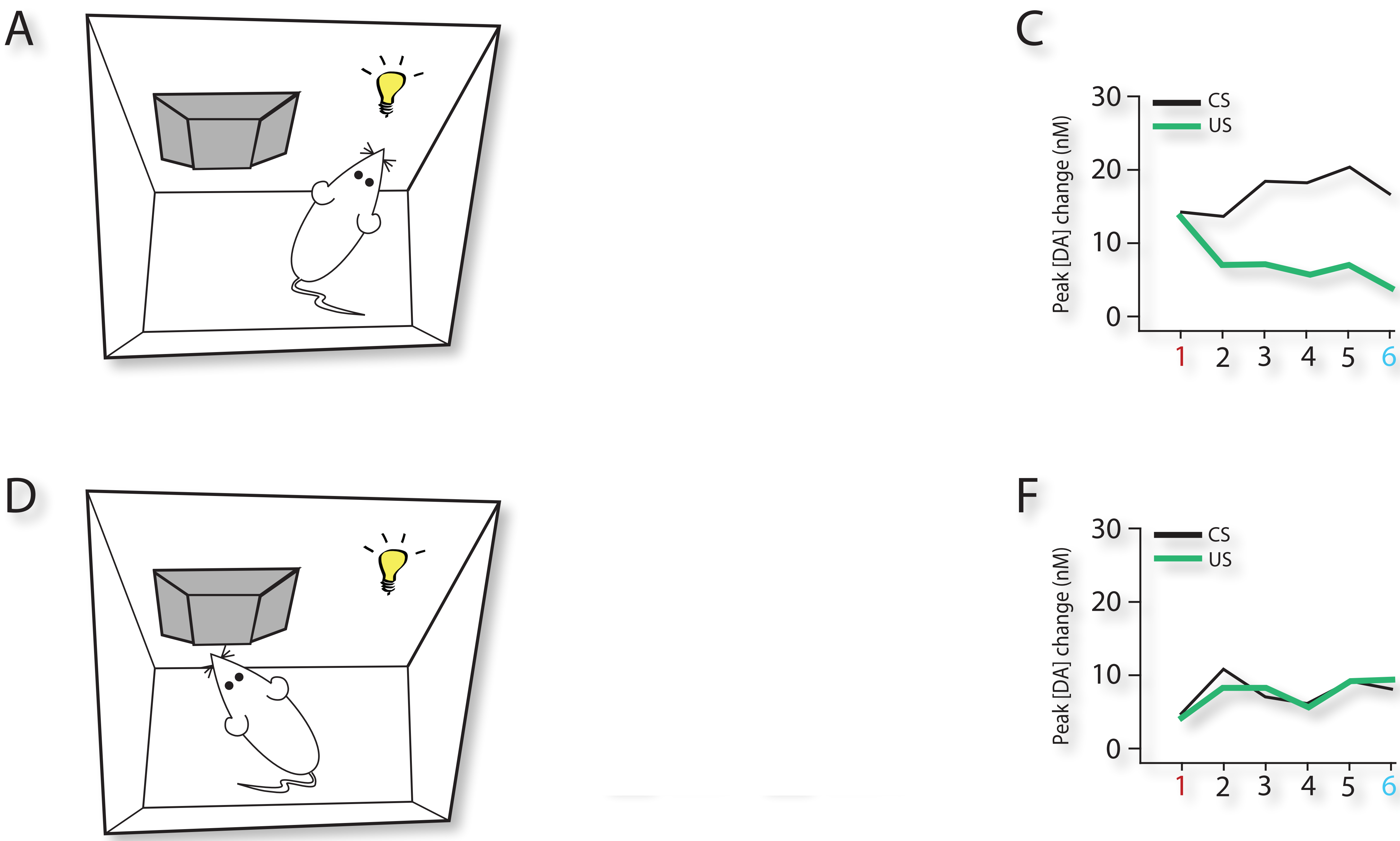


D



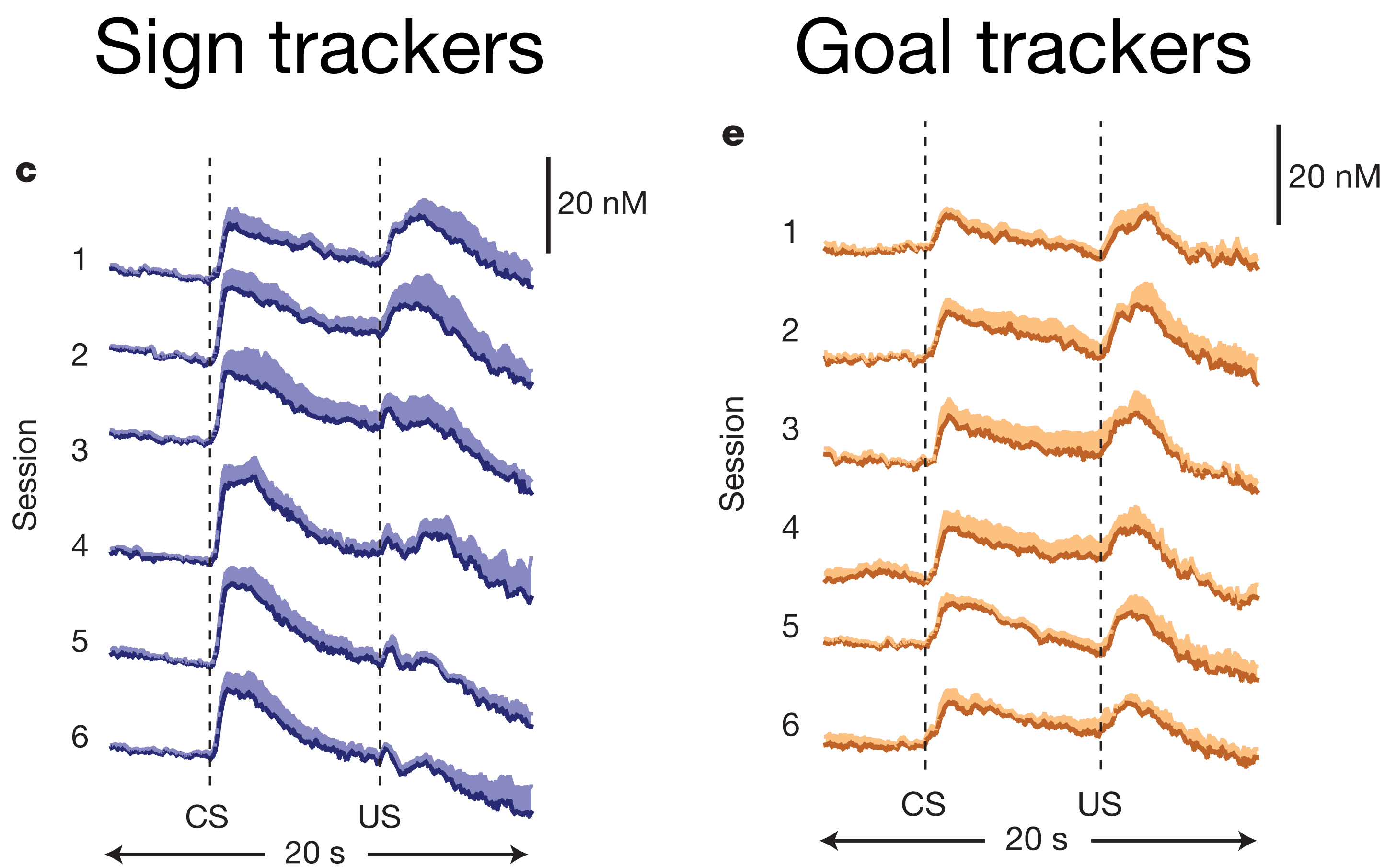
Flagel et al., 2011 Nature, Huys et al., 2014 Prog. Neurobiol.

Signtracking



Flagel et al., 2011 Nature, Huys et al., 2014 Prog. Neurobiol.

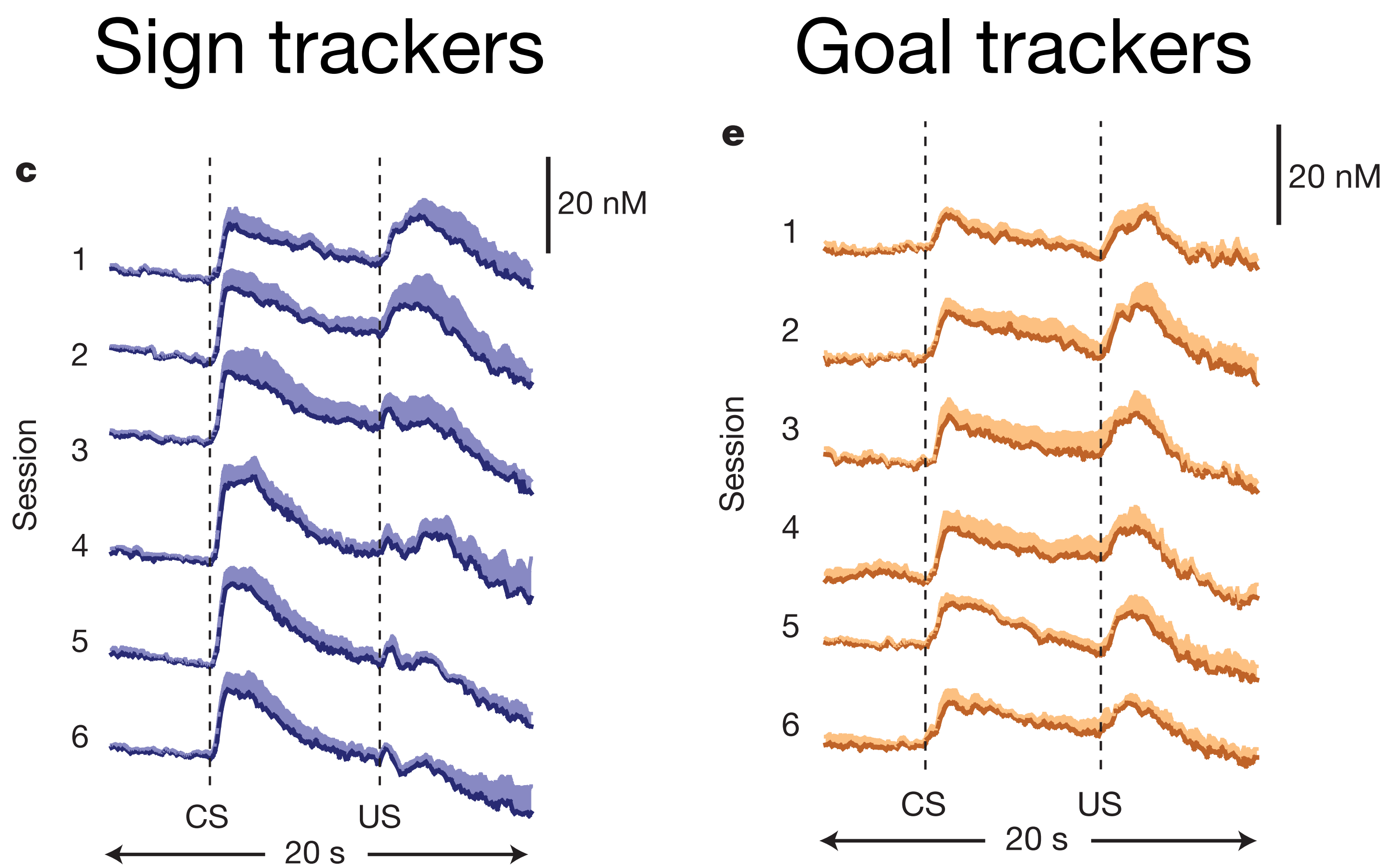
Absent model?



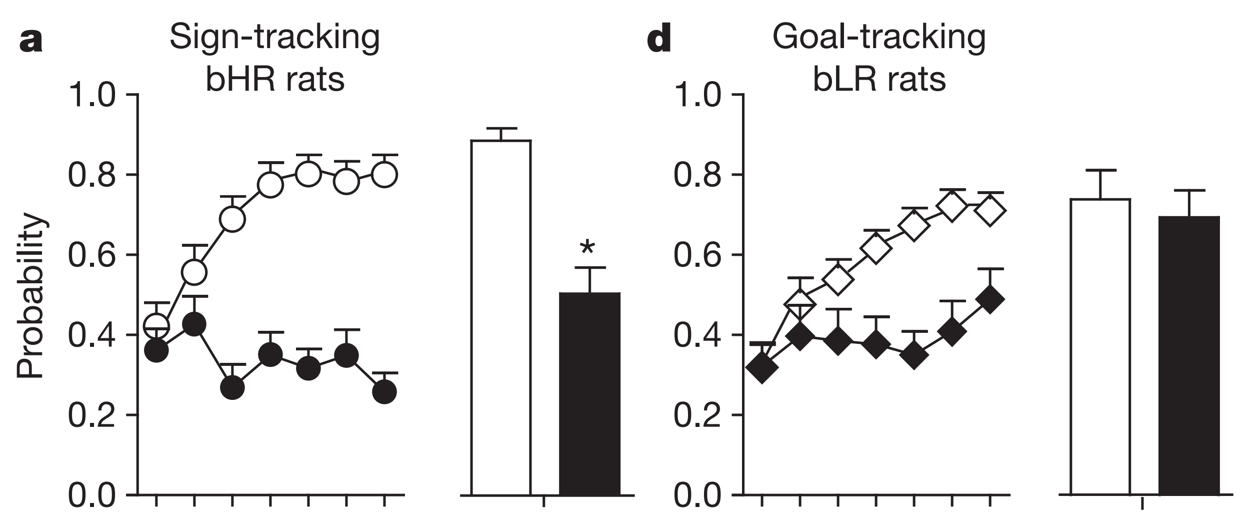
$$\delta = r - \mathcal{Q}$$

Flagel et al., 2011 Nature

Absent model?



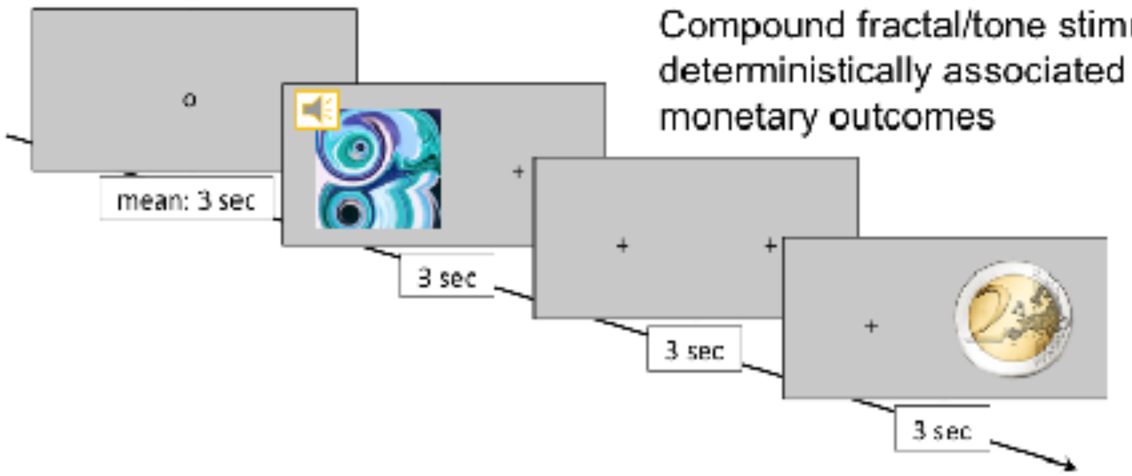
$$\delta = r - Q$$



Flagel et al., 2011 Nature

Sign-tracking in humans?

Experimental Paradigm
Pavlovian Conditioning

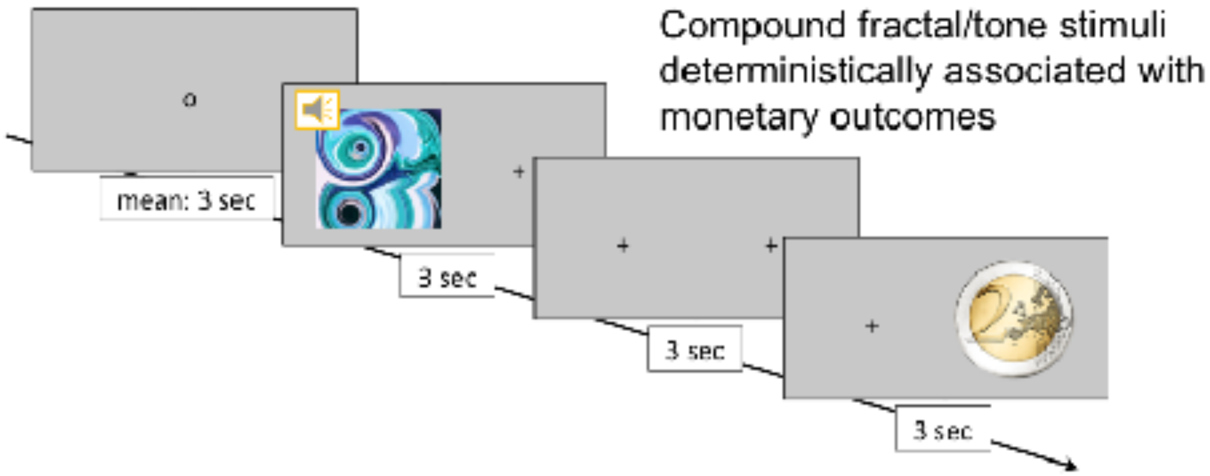


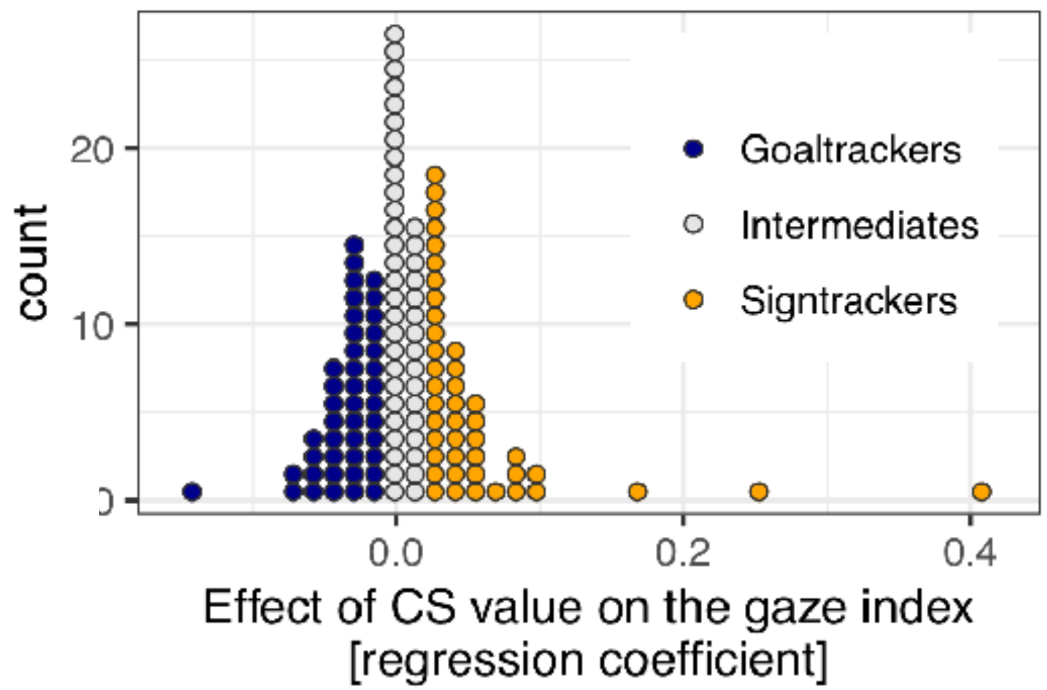
Compound fractal/tone stimuli deterministically associated with monetary outcomes

Schad et al., in prep

Sign-tracking in humans?

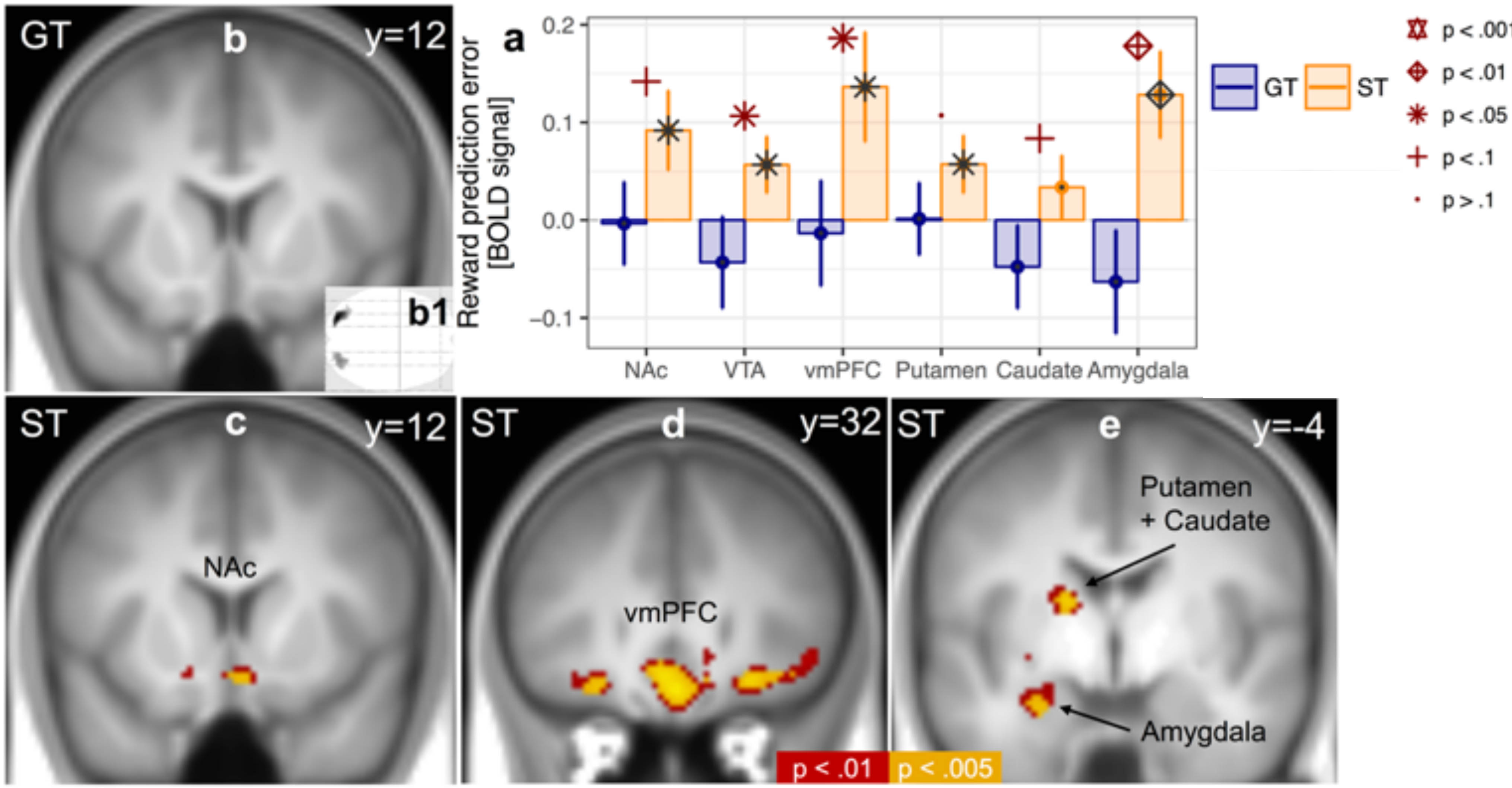
Experimental Paradigm
Pavlovian Conditioning





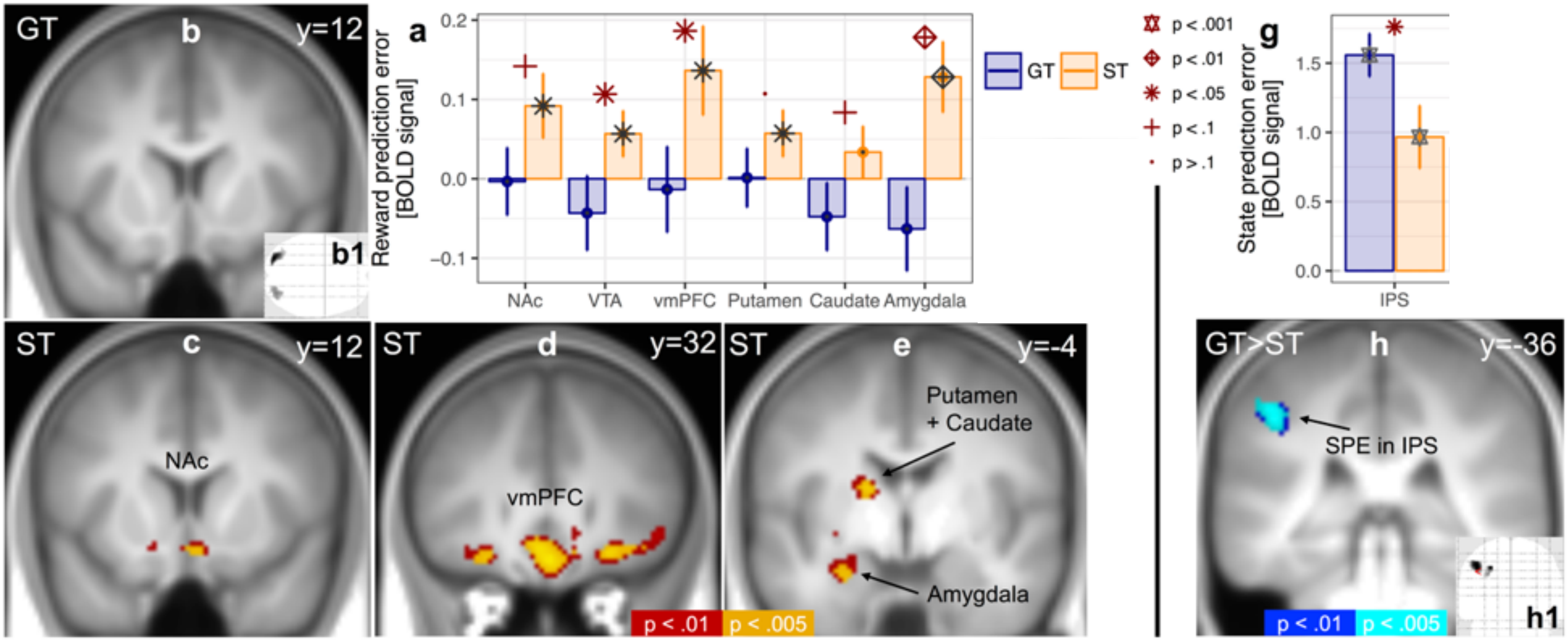
Schad et al., in prep

Double dissociation between ST and GT



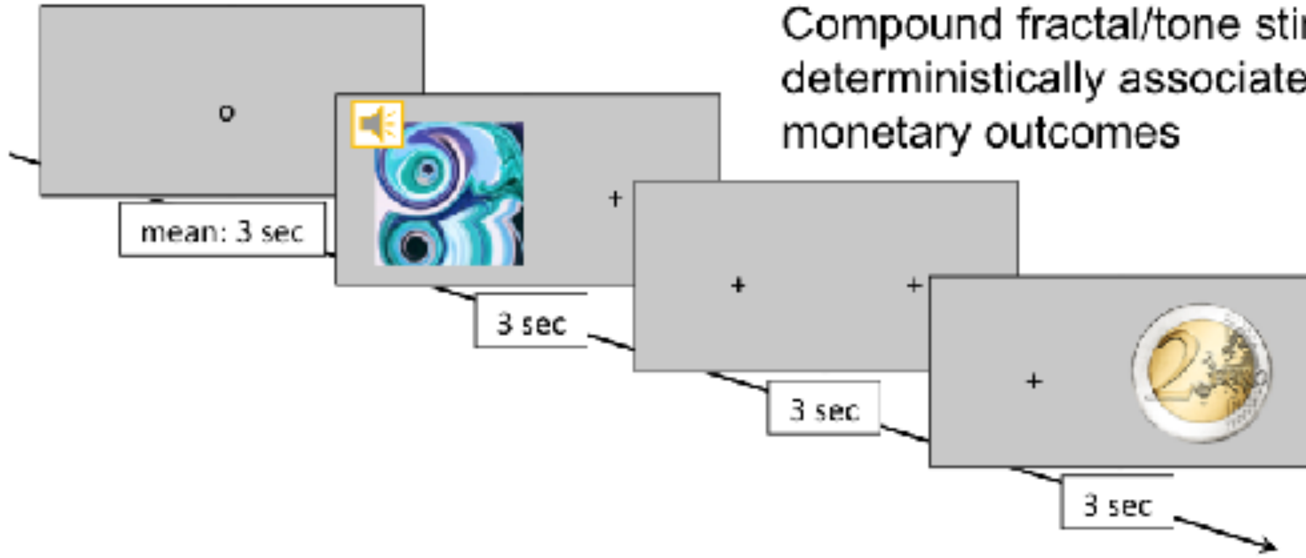
Schad et al., in prep

Double dissociation between ST and GT



Schad et al., in prep

Goal-tracking in humans?



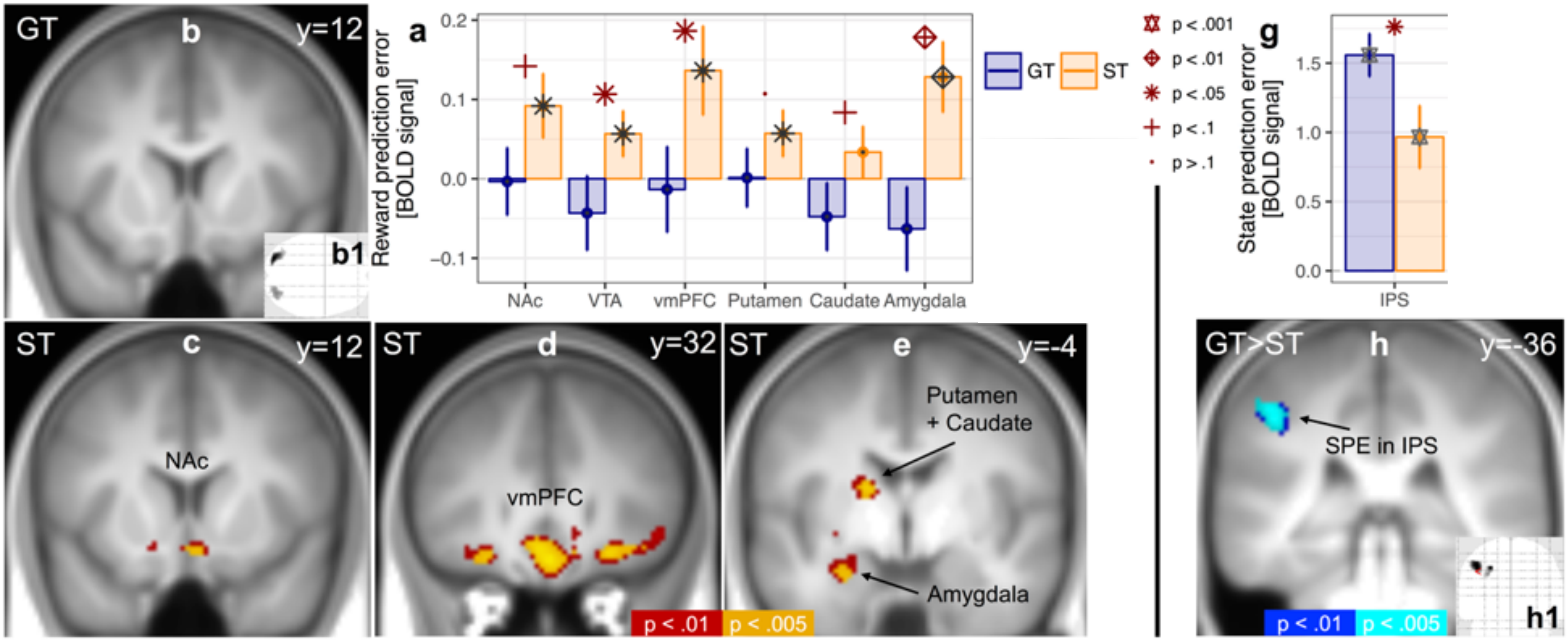
Compound fractal/tone stimuli deterministically associated with monetary outcomes

ST: learn expected value V
GT: learn mappings T from CS to US identity

$$\mathcal{V}(s) = \sum_a \pi(a;s) \sum_{s'} \mathcal{T}(s'|s,a)[\mathcal{R}(s',a,s) + \mathcal{V}(s')]$$

Schad et al., in prep

Double dissociation between ST and GT



Schad et al., in prep

Successor representation

$$V^\pi(s) = \sum_a \pi(a|s) \sum_{s'} \mathcal{T}^a_{ss'}[\mathcal{R}^a_{ss'} + V(s')]$$

$$\mathbf{v}^\pi = \mathbf{R}^\pi + \mathbf{T}^\pi \mathbf{v}^\pi$$

$$\mathbf{v}^\pi = (\mathbf{I} - \mathbf{T}^\pi)^{-1}\mathbf{R}^\pi$$

$$\hat{\mathbf{v}} = \mathbf{M}\mathbf{w}$$

Learning a successor representation

$$M^\pi(s,:) = \mathbf{1}_s + \gamma \sum_{s'} T^\pi(s,s') M^\pi(s',:),$$

$$M^\pi(s,:) \leftarrow M^\pi(s,:) + \alpha_{SR}[\mathbf{1}_s + \gamma M^\pi(s',:) - M^\pi(s,:)],$$

$$M^\pi = (I - \gamma T^\pi)^{-1}$$

Russek et al., 2017 PLoS Biol

Learning a successor representation

- "Model-free learning"

$$M^\pi(s,:) = \mathbf{1}_s + \gamma \sum_{s'} T^\pi(s,s') M^\pi(s',:),$$

$$M^\pi(s,:) \leftarrow M^\pi(s,:) + \alpha_{SR}[\mathbf{1}_s + \gamma M^\pi(s',:) - M^\pi(s,:)],$$

$$M^\pi = (I - \gamma T^\pi)^{-1}$$

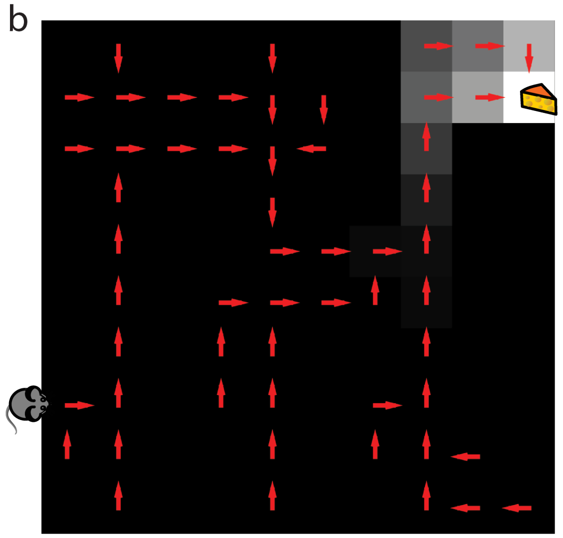
Russek et al., 2017 PLoS Biol

Learning a successor representation

- “Model-free learning”

$$M^\pi(s,:) = \mathbf{1}_s + \gamma \sum_{s'} T^\pi(s,s') M^\pi(s',:),$$

$$M^\pi(s,:) \leftarrow M^\pi(s,:) + \alpha_{SR}[\mathbf{1}_s + \gamma M^\pi(s',:) - M^\pi(s,:)],$$



$$M^\pi = (I - \gamma T^\pi)^{-1}$$

Russek et al., 2017 PLoS Biol

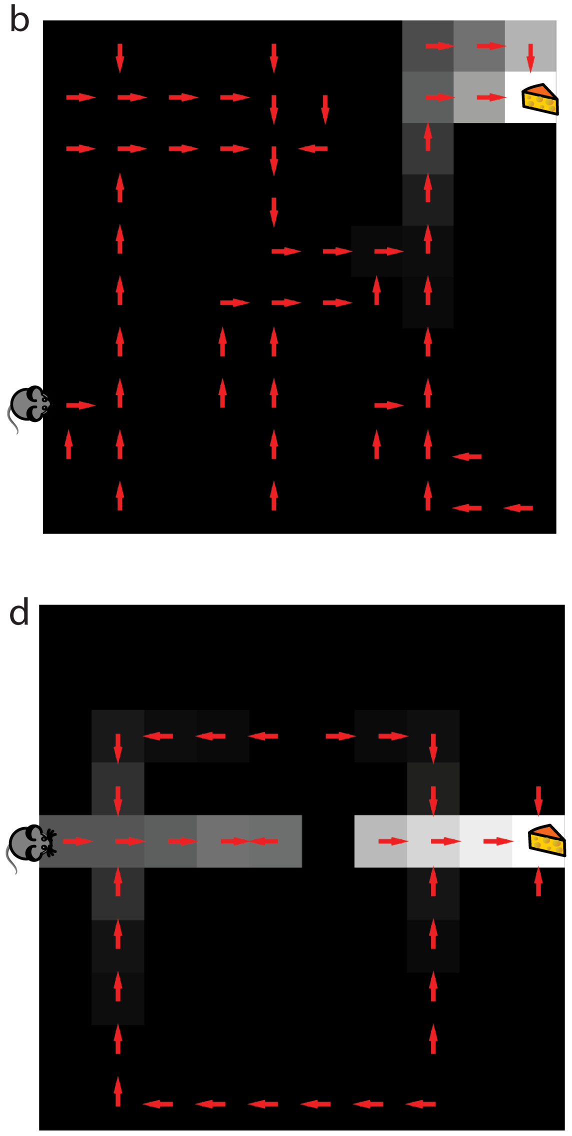
Learning a successor representation

- “Model-free learning”

$$M^\pi(s,:) = \mathbf{1}_s + \gamma \sum_{s'} T^\pi(s,s') M^\pi(s',:),$$

$$M^\pi(s,:) \leftarrow M^\pi(s,:) + \alpha_{SR}[\mathbf{1}_s + \gamma M^\pi(s',:) - M^\pi(s,:)],$$

$$M^\pi = (I - \gamma T^\pi)^{-1}$$



Russek et al., 2017 PLoS Biol

Learning a successor representation

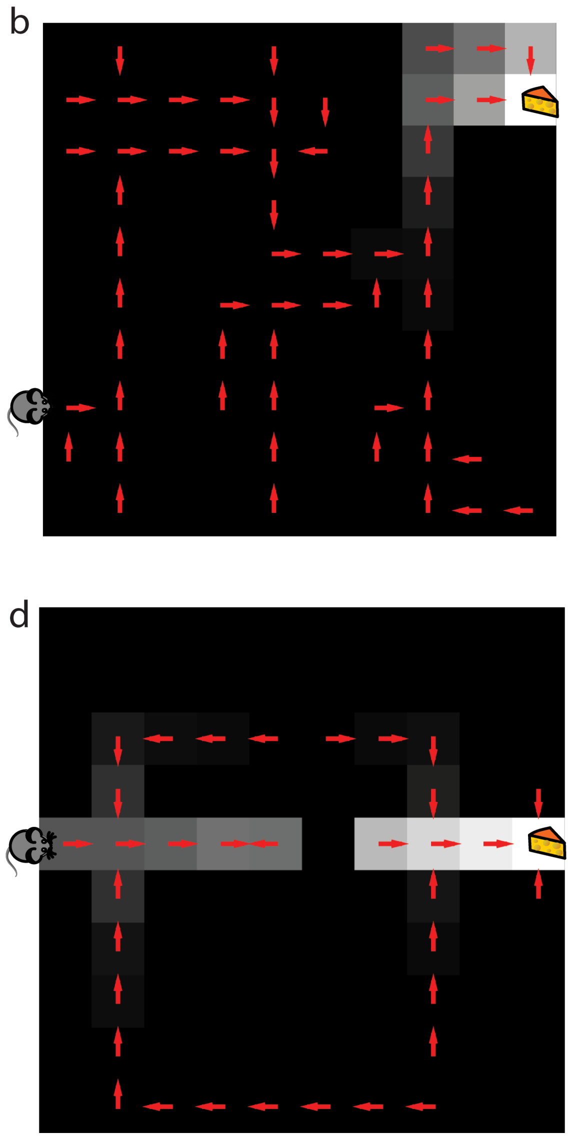
- “Model-free learning”

$$M^{\pi}(s,:) = \mathbf{1}_s + \gamma \sum_{s'} T^{\pi}(s,s') M^{\pi}(s',:),$$

$$M^{\pi}(s,:) \leftarrow M^{\pi}(s,:) + \alpha_{SR}[\mathbf{1}_s + \gamma M^{\pi}(s',:) - M^{\pi}(s,:)],$$

- “Model-based learning”
  - Estimate transition and compute

$$M^{\pi} = (I - \gamma T^{\pi})^{-1}$$



Russek et al., 2017 PLoS Biol

Learning a successor representation

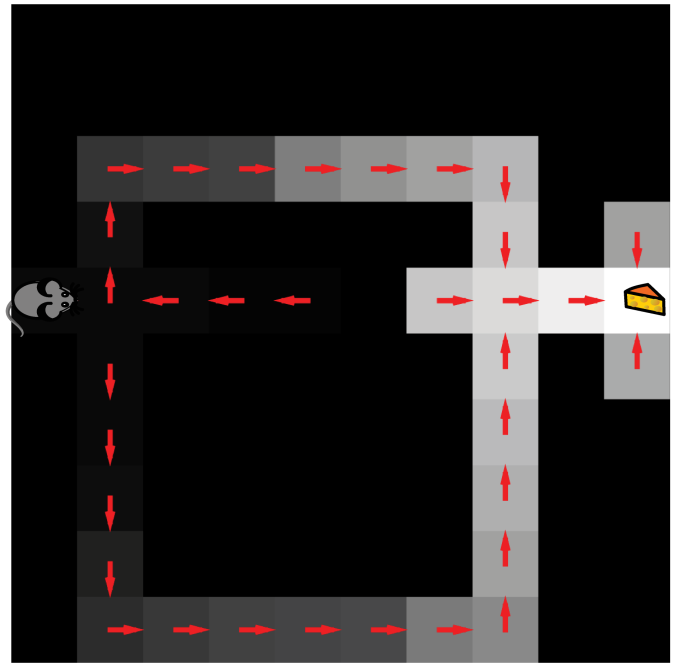
- “Model-free learning”

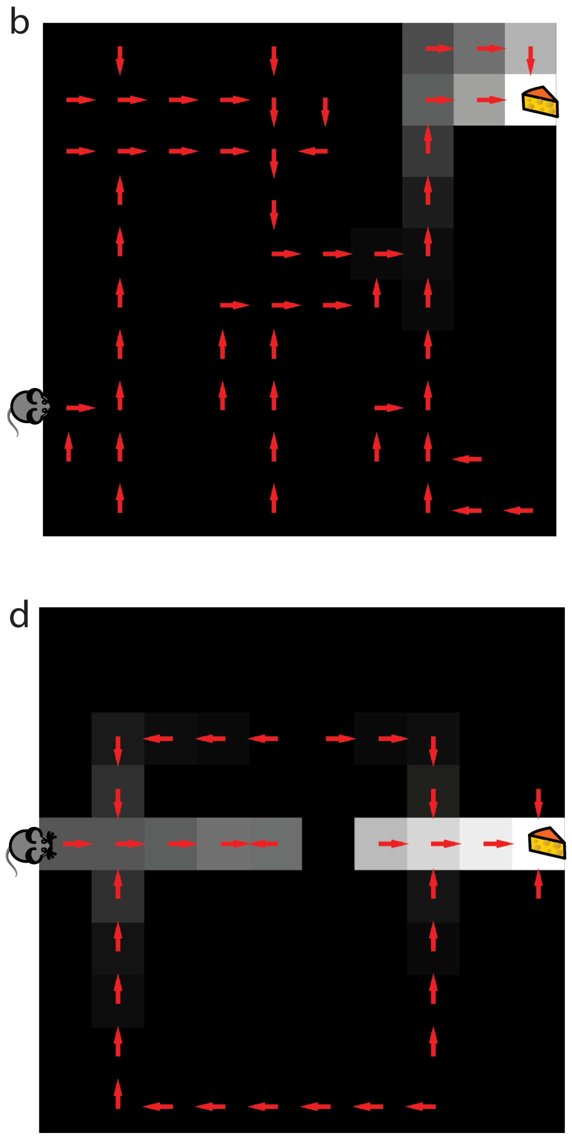
$$M^\pi(s,:) = \mathbf{1}_s + \gamma \sum_{s'} T^\pi(s,s') M^\pi(s',:),$$

$$M^\pi(s,:) \leftarrow M^\pi(s,:) + \alpha_{SR}[\mathbf{1}_s + \gamma M^\pi(s',:) - M^\pi(s,:)],$$

- “Model-based learning”
  - Estimate transition and compute

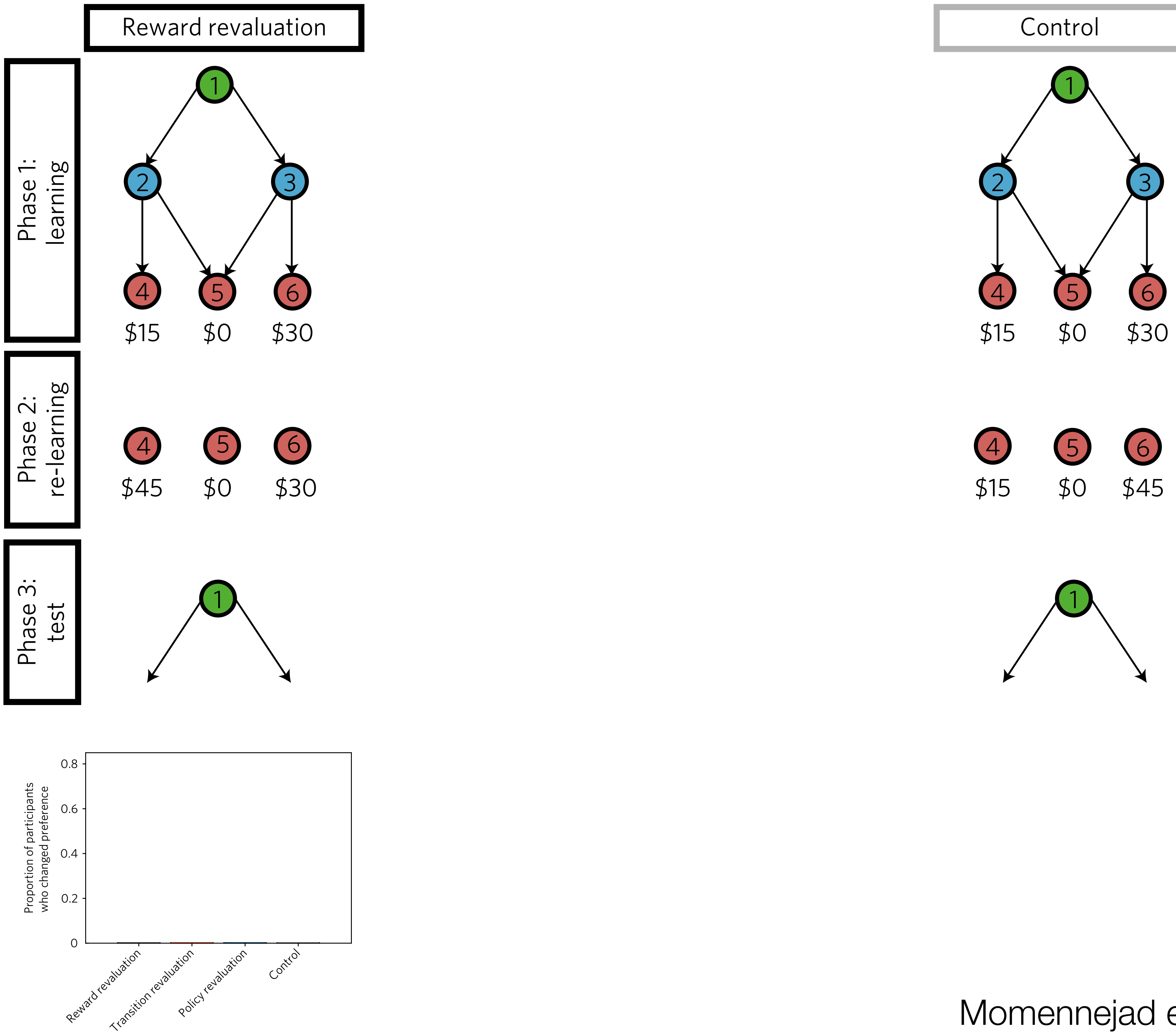
$$M^\pi = (I - \gamma T^\pi)^{-1}$$



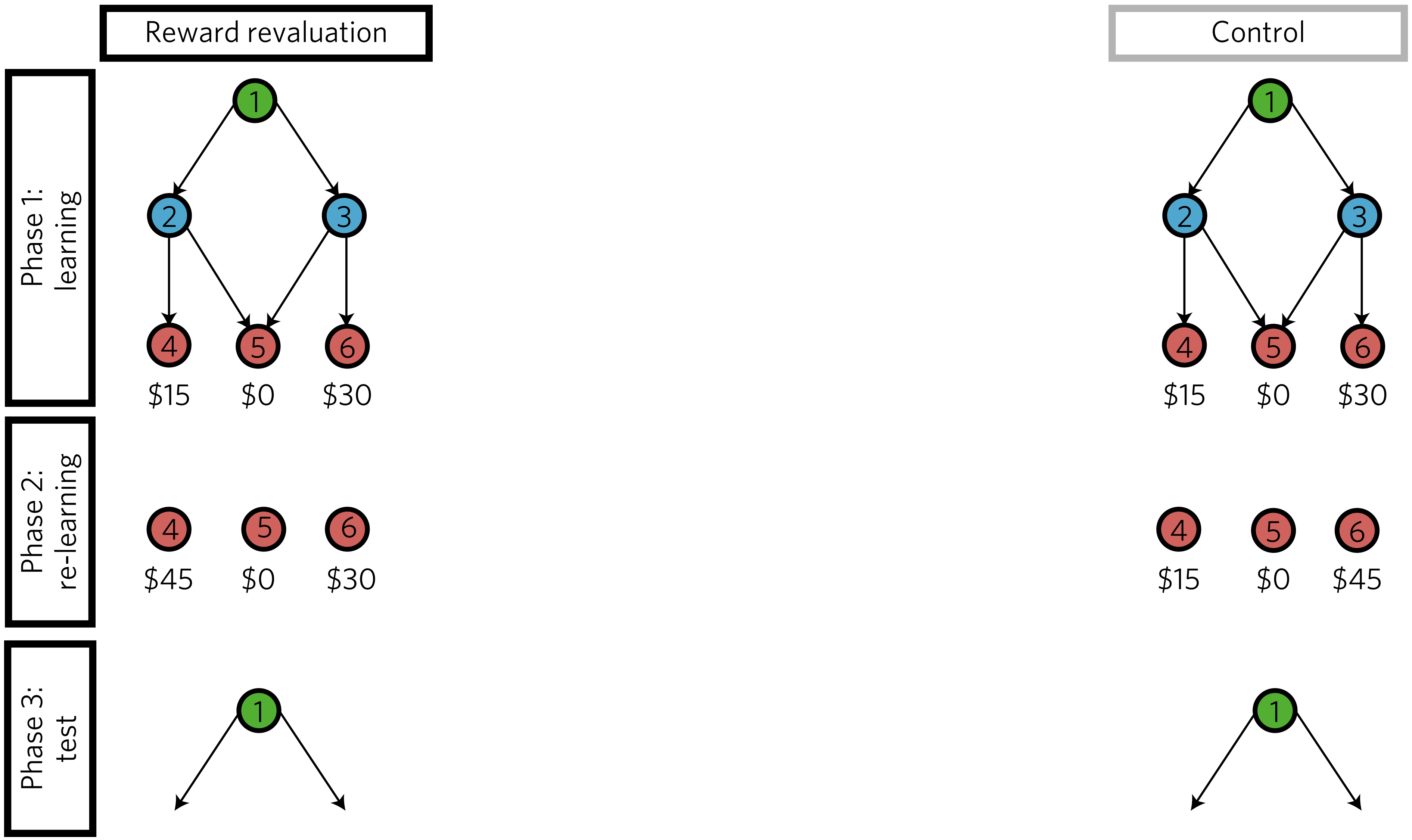


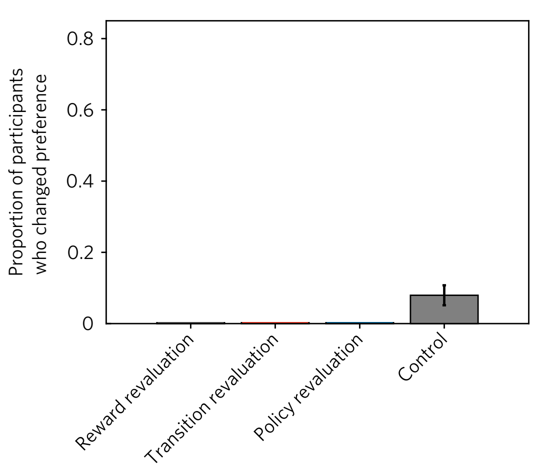
Russek et al., 2017 PLoS Biol

Human successor learning



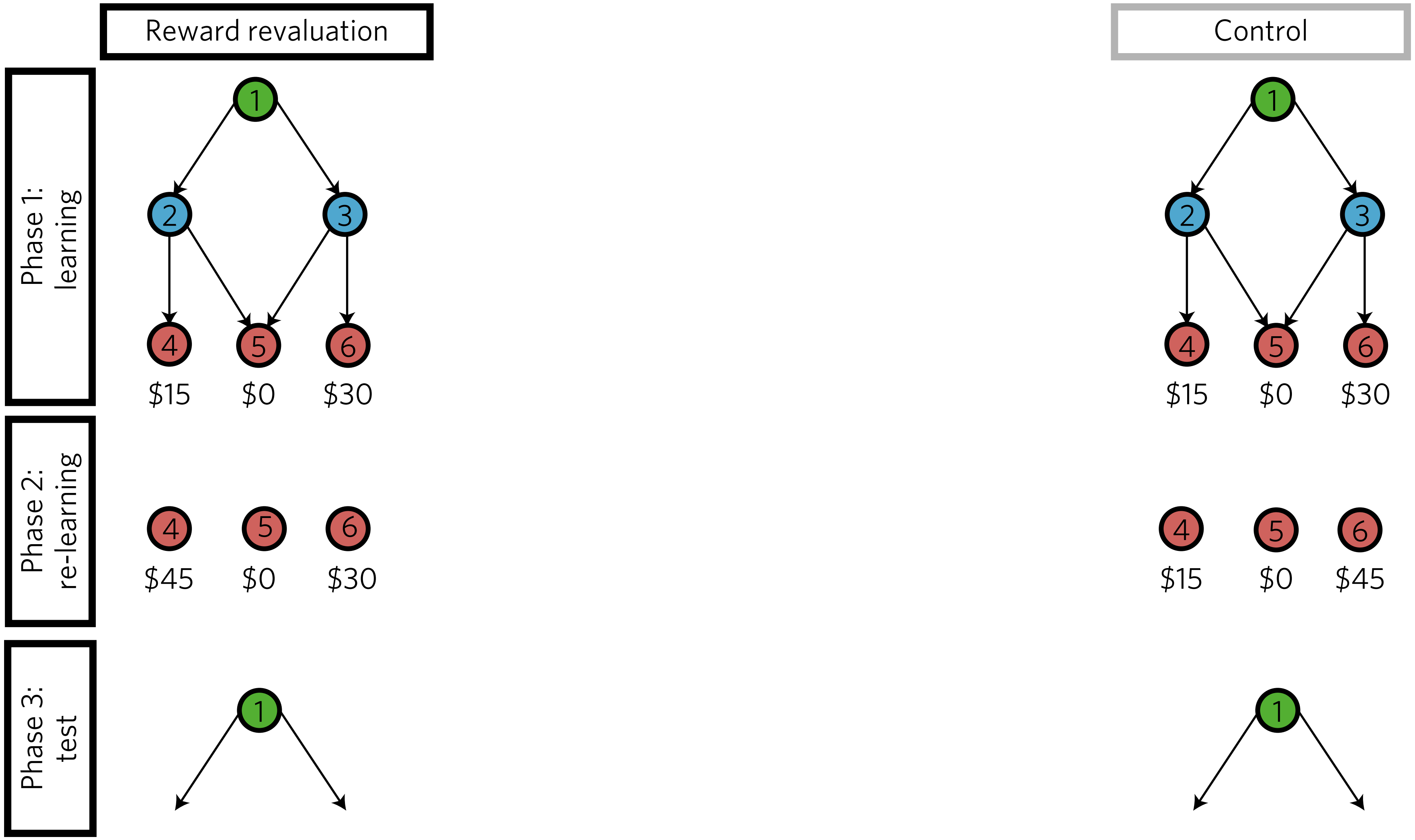
Momennejad et al., 2017 Nat. Hum. Beh.

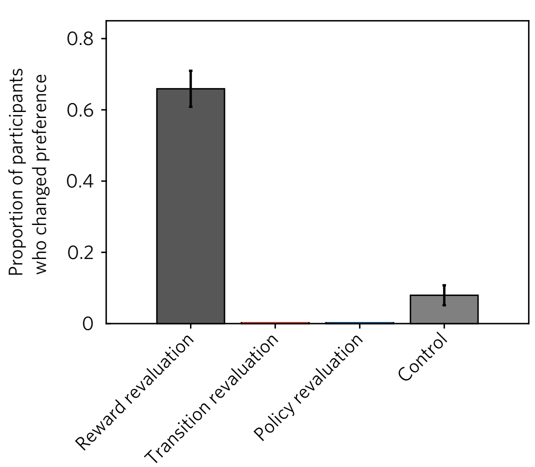




Momennejad et al., 2017 Nat. Hum. Beh.

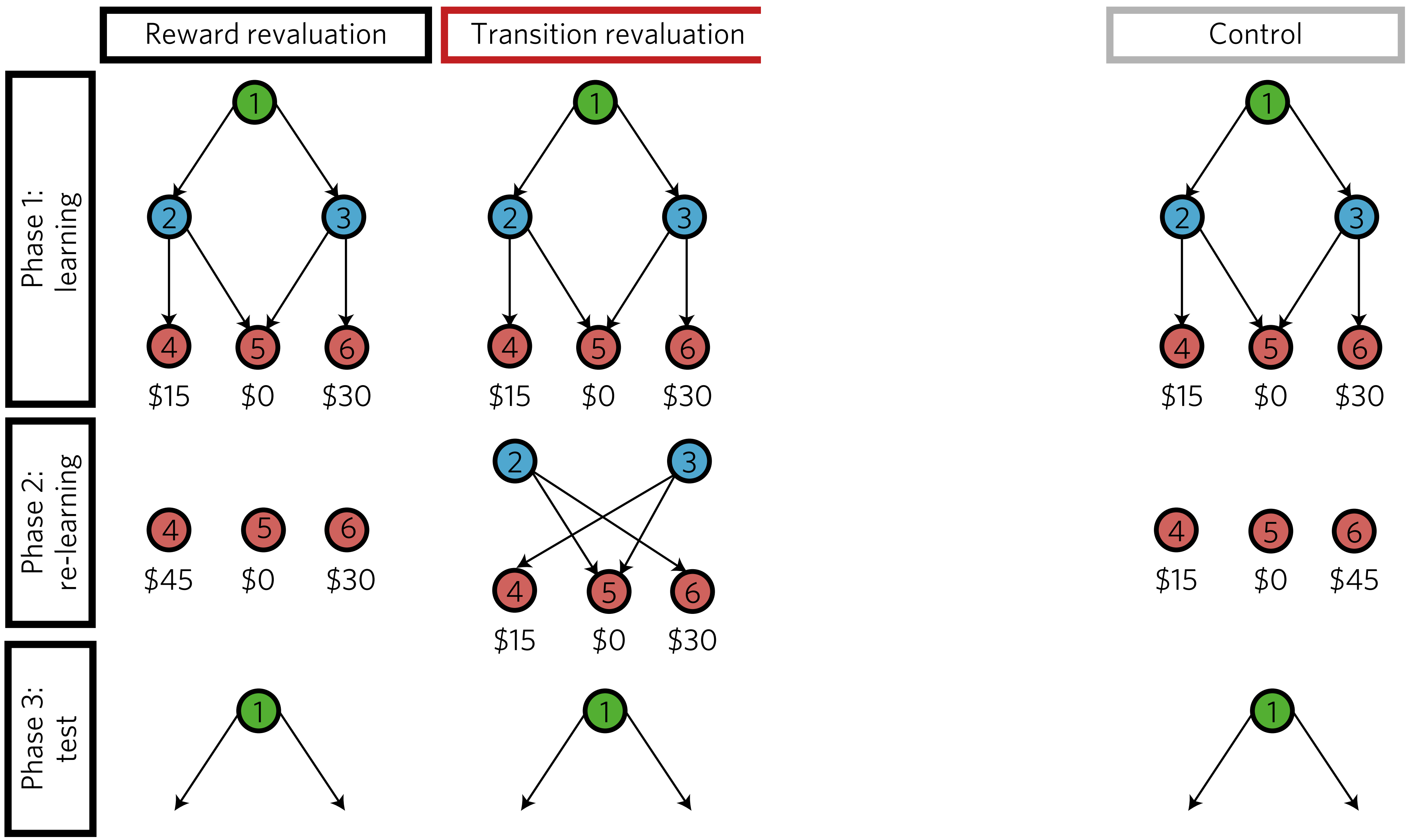
Human successor learning

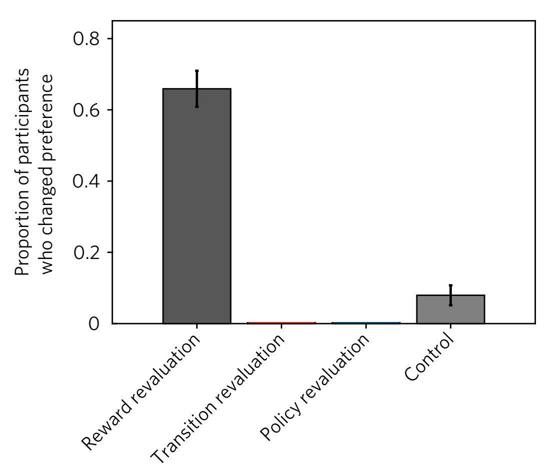




Momennejad et al., 2017 Nat. Hum. Beh.

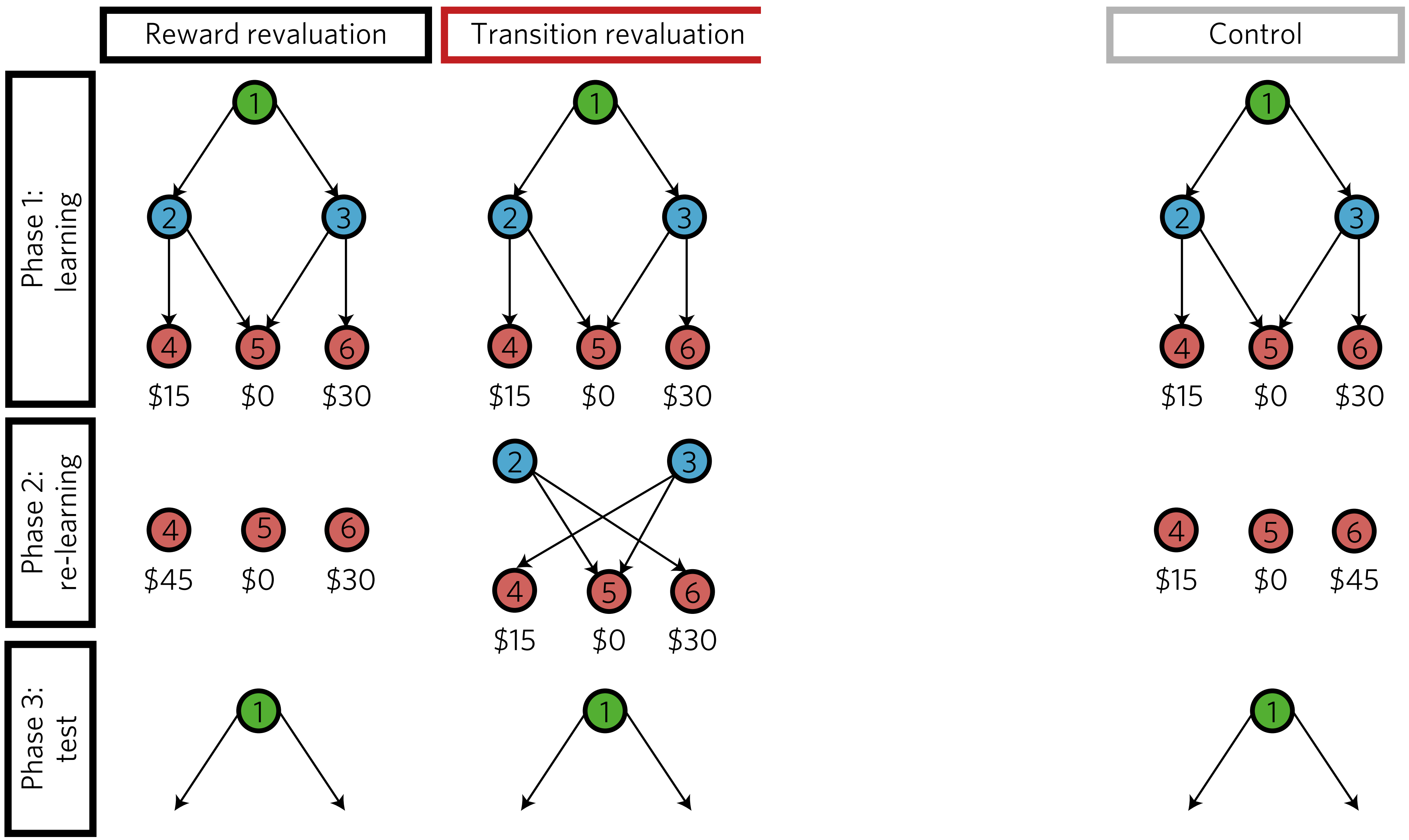
Human successor learning

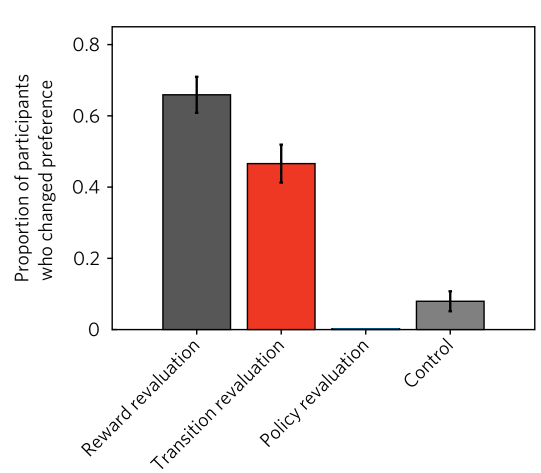




Momennejad et al., 2017 Nat. Hum. Beh.

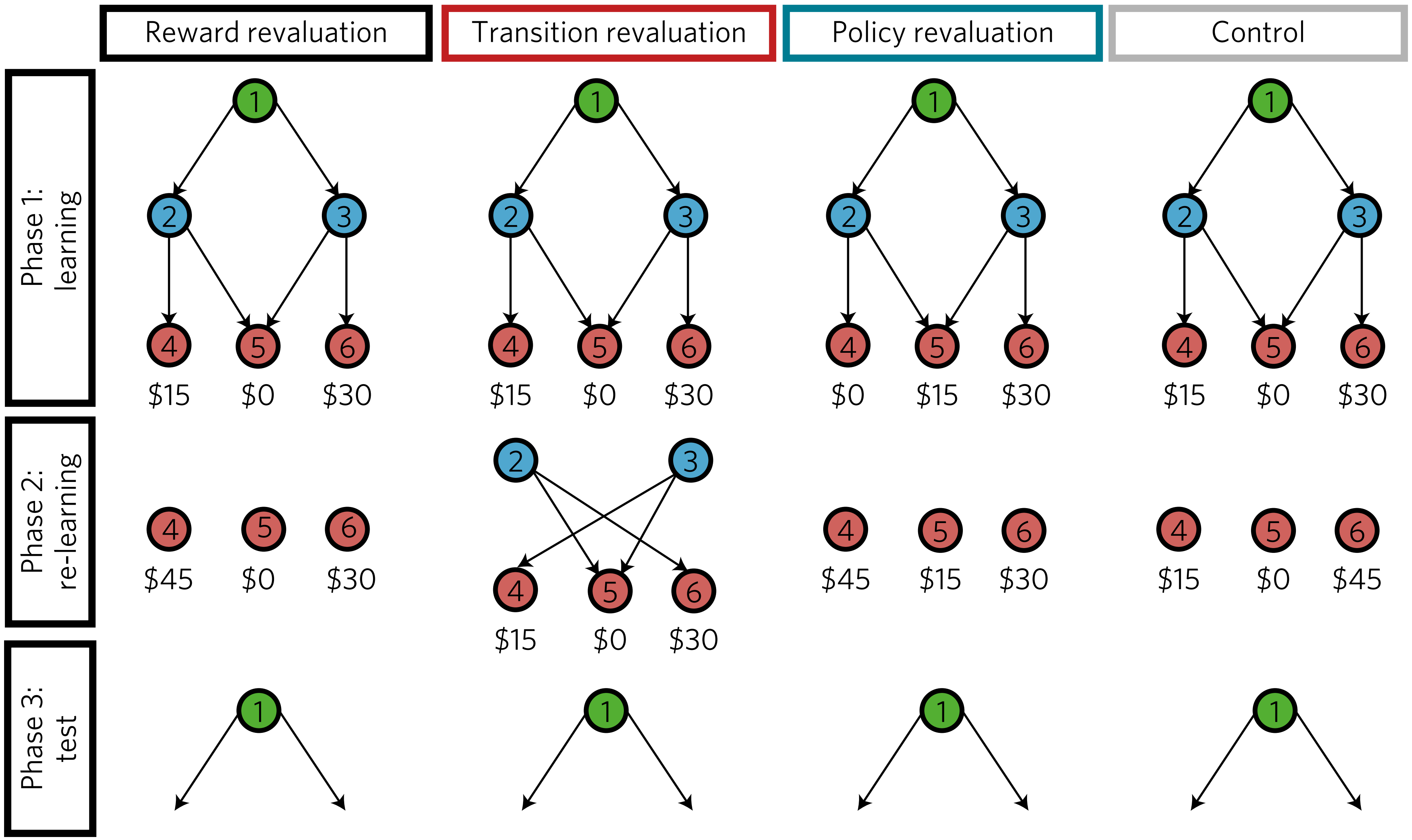
Human successor learning

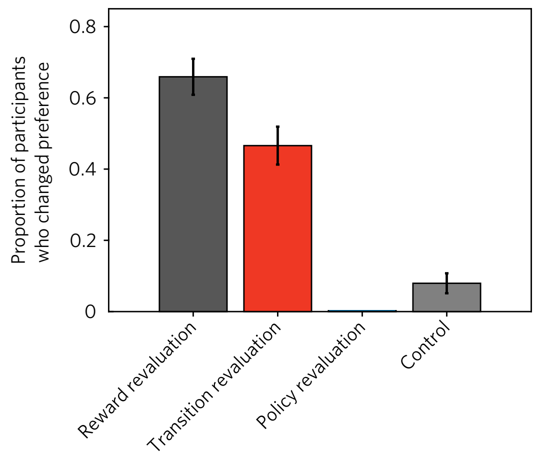




Momennejad et al., 2017 Nat. Hum. Beh.

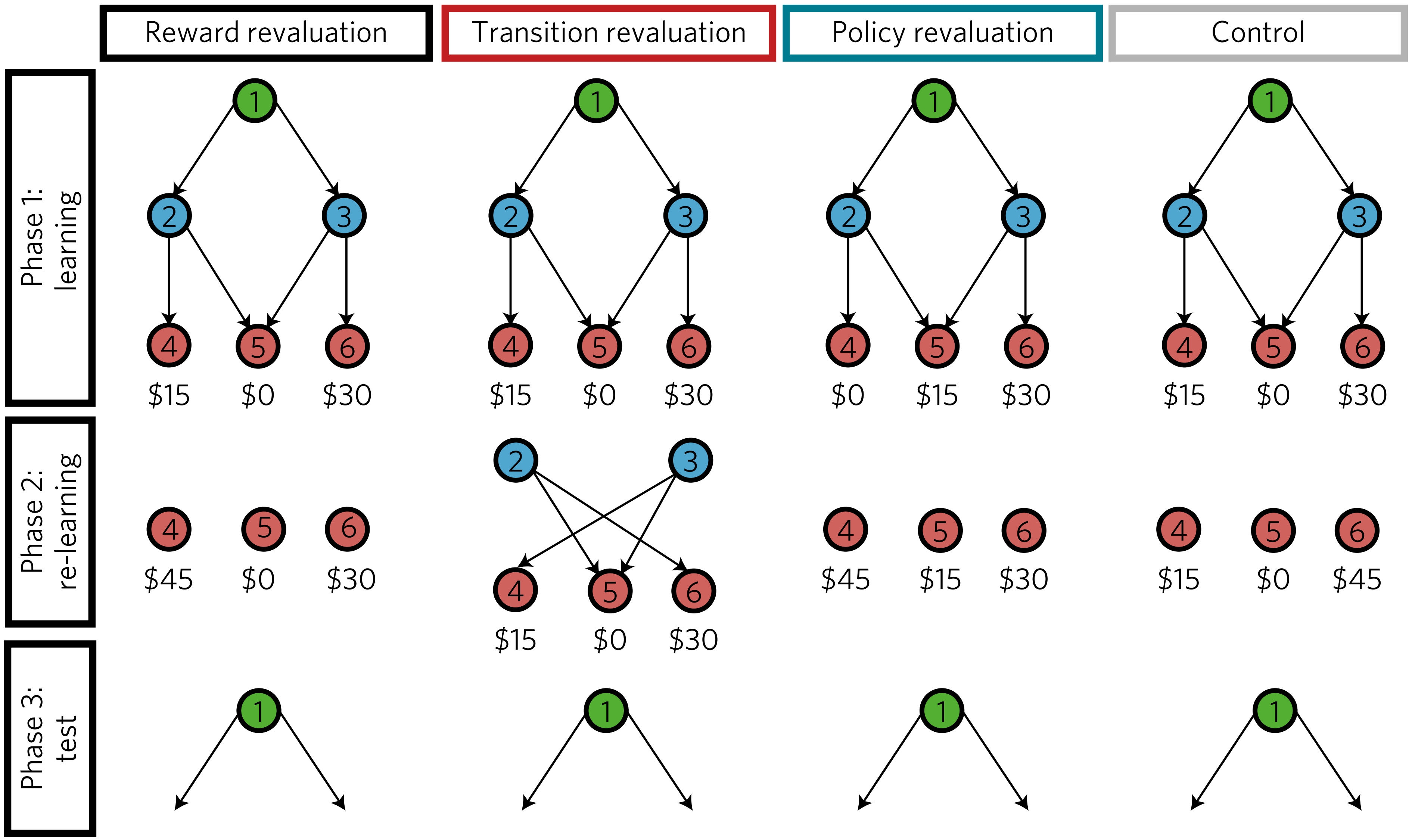
Human successor learning

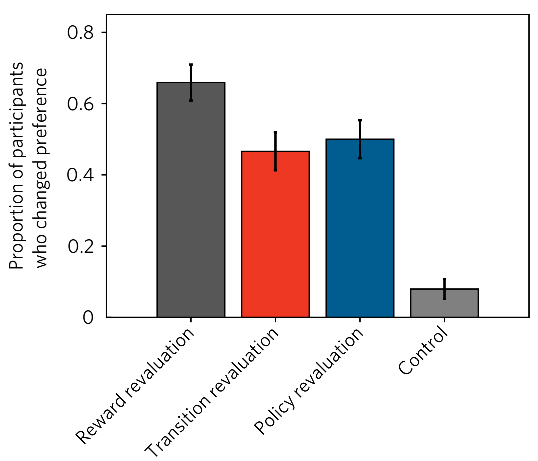




Momennejad et al., 2017 Nat. Hum. Beh.

Human successor learning





Momennejad et al., 2017 Nat. Hum. Beh.