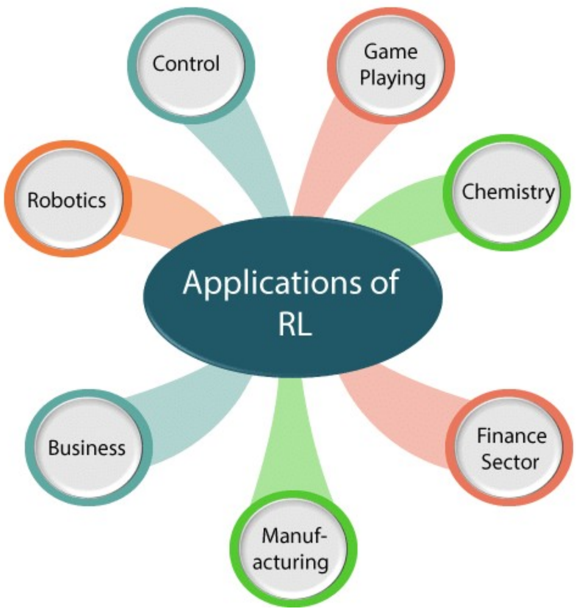
# TEK 5040/9040 Reinforcement Learning (basics)

Narada Warakagoda

# What is Reinforcement Learning (RL)

- How the agents learn by trial and error
- Reward/punish a certain types of behavior




Applications of Reinforcement Learning (Shastha et al., 2019)

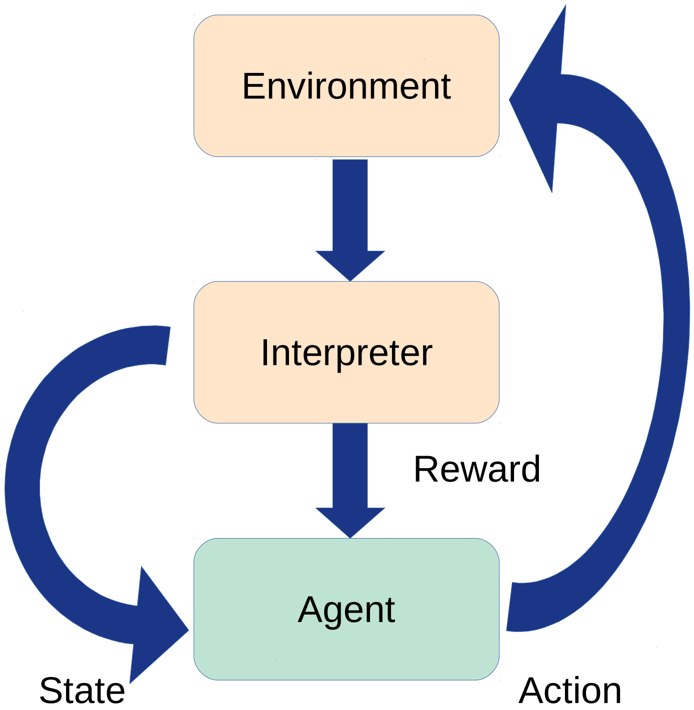
# Why RL

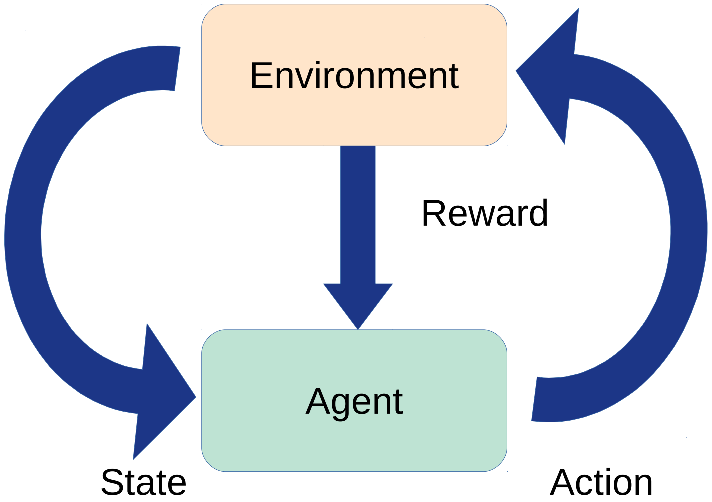
- Less detailed instructions/supervision/annotations
- Learning *optimal behavior* rather than *imitation*

# Basic concepts of RL

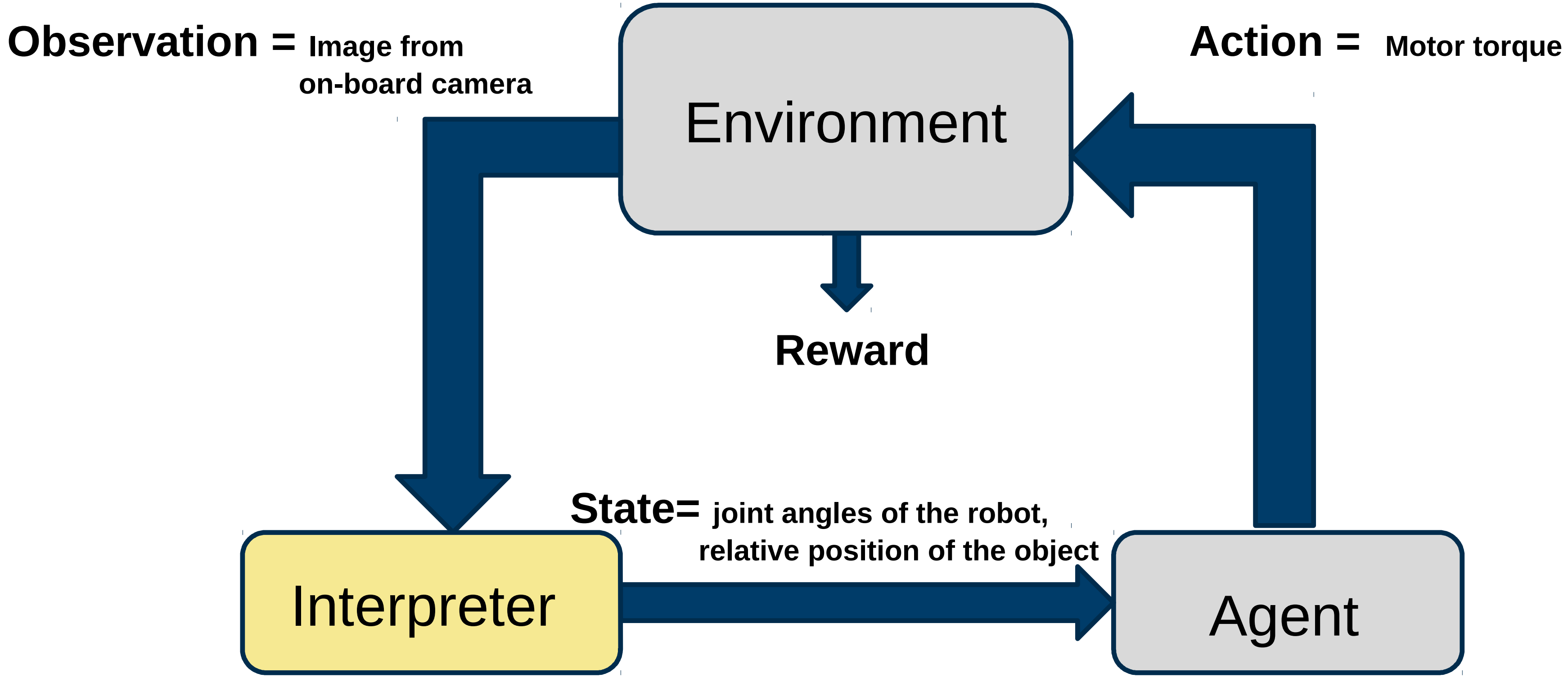
- Agent-Environment interaction

$$\mathbf{s}_t = \text{state at time } t$$
$$\mathbf{a}_t = \text{action at time } t$$
$$\mathbf{r}_t = \text{reward at time } t$$

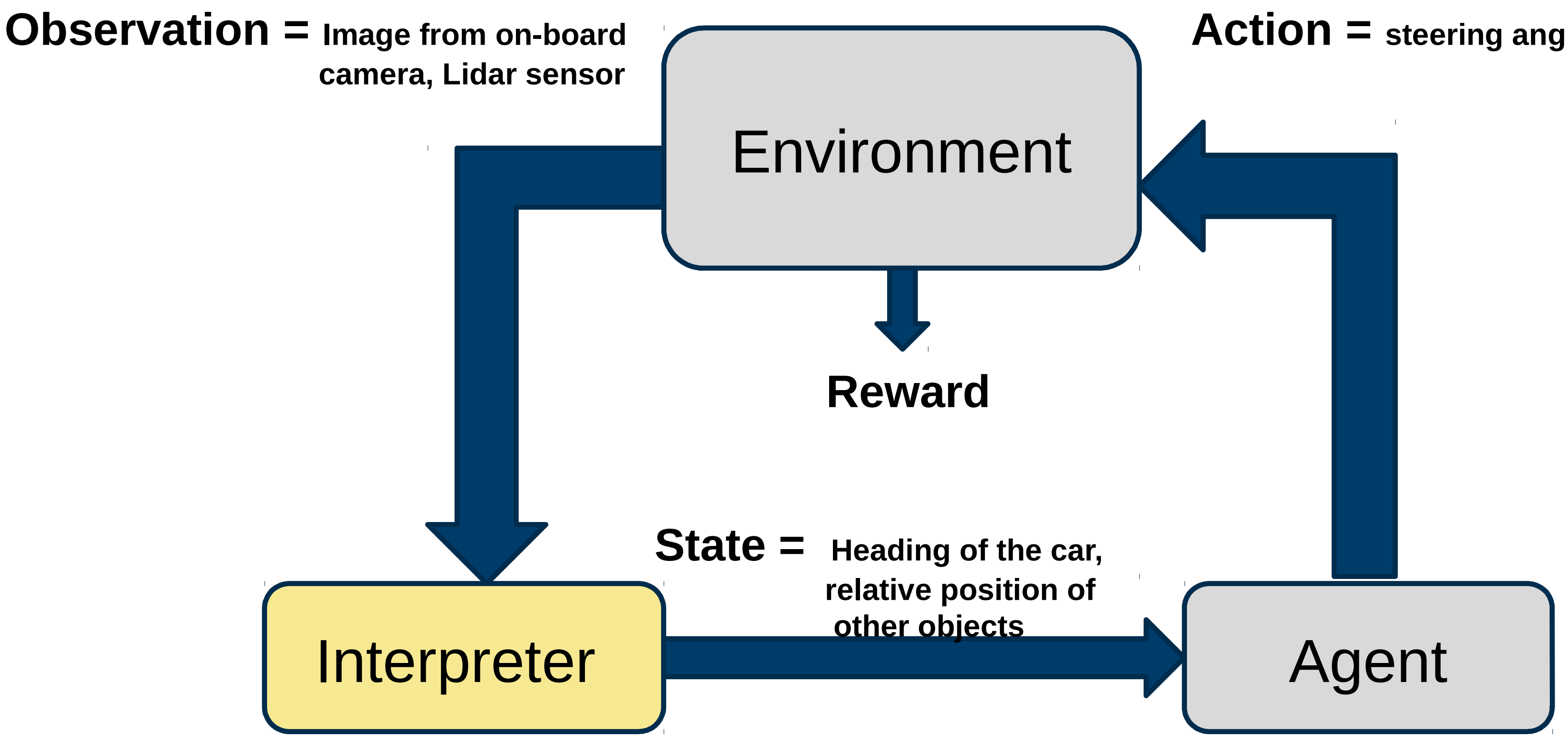


# Example 1 (manipulation robot)



# Example 2 (mobile robot)



# Environment

- Let the current state of the environment is $\mathrm{s}_t$
- When the agent performs an action $\mathrm{a}_t$ , the environment
  - changes its state to $\mathrm{s}_{t+1}$
    - Deterministic ***state transition*** rule

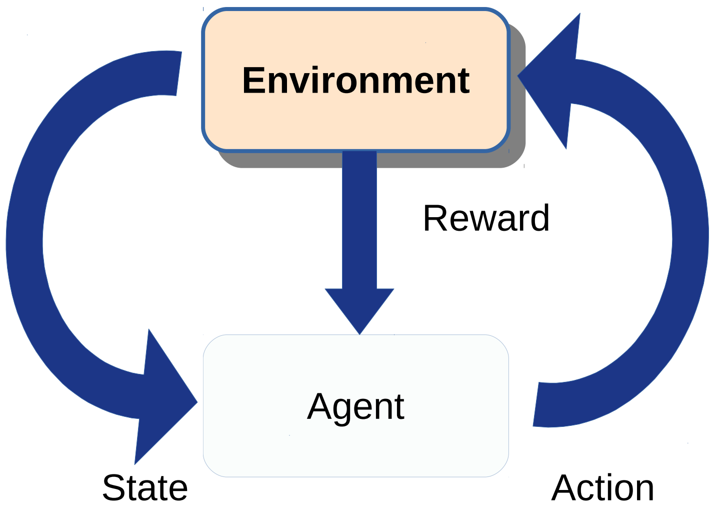
$$\mathrm{s}_{t+1} = f(\mathbf{s}_t, \mathbf{a}_t)$$

    - Stochastic ***state transition*** rule

$$\mathrm{s}_{t+1} \sim P(\cdot|\mathbf{s}_t, \mathbf{a}_t)$$

  - generates a reward $\mathrm{r}_t$

$$\mathrm{r}_t = R(\mathbf{s}_t, \mathbf{a}_t)$$



# State space

- Discrete state spaces
  - Set of possible states is finite
  - Eg: Board state of  game Go

- Continuous state spaces
  - Set of possible states is infinite
  - Eg: Angle of robotic arm

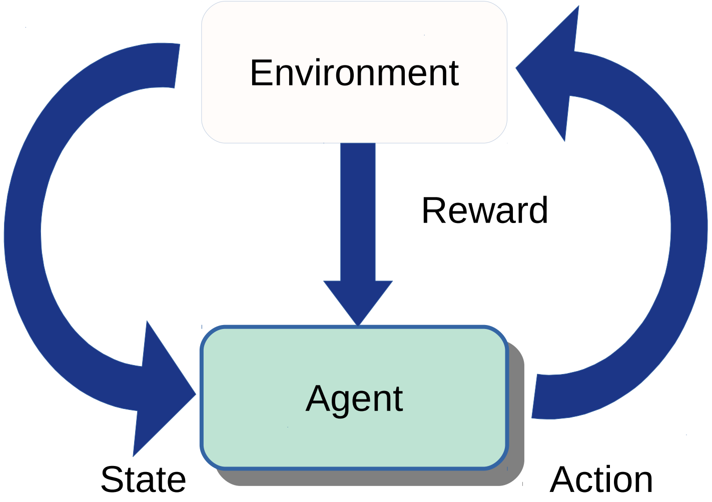
# Agent/Policy

- When the current state of the environment is $\mathrm{s}_t$, the agent generates an action $\mathrm{a}_t$
  - Deterministic ***policy***

$$\mathrm{a}_t = \mu(\mathbf{s}_t)$$

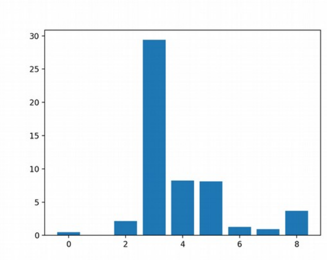
  - Stochastic ***policy***

$$\mathrm{a}_t \sim \pi(\cdot|\mathbf{s}_t)$$

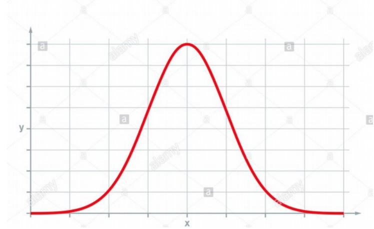


# Action spaces

- Action space = set of all possible actions
- Discrete action space
  - Set of actions is finite
  - Discrete valued vectors
  - Stochastic policy is a categorical distribution
  - Eg: possible moves in game playing such as Go, Atari




- Continuous action space
  - Set of actions is infinite
  - Continuous valued vectors
  - Stochastic policy is a continuous distribution such as Gaussian
  - Eg: Steering angle of self-driving car

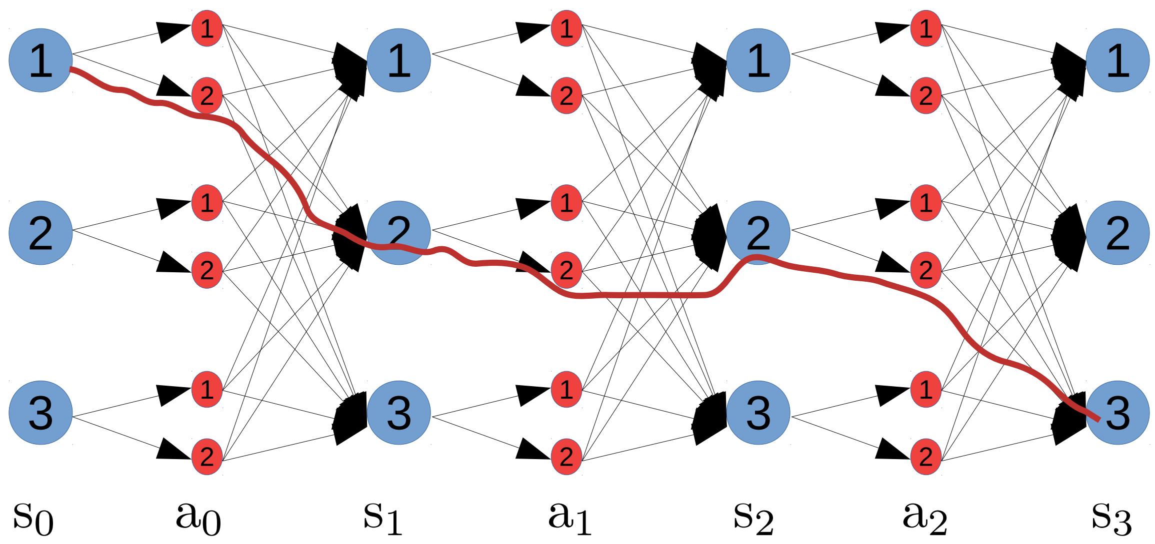


# Trajectories

- A trajectory (also called ***rollout*** or ***episode***) is a sequence of state-action pairs

$$\tau = (\mathbf{s}_0, \mathbf{a}_0, \mathbf{s}_1, \mathbf{a}_1, \mathbf{s}_2, \mathbf{a}_2, \cdots)$$

where development of this sequence is governed by state-transition function of the environment and agent's policy.



# Reward and Return

- At a given state $\mathrm{s}_t$, when the action is $\mathrm{a}_t$, the environment generates a reward $\mathrm{r}_t$ using a reward function $R(\cdot,\cdot)$

$$\mathrm{r}_t = R(\mathbf{s}_t, \mathbf{a}_t)$$

- Total reward of a finite length trajectory, ***finite horizon undiscounted return***

$$\mathrm{R}(\tau) = \sum_{t=0}^{T} \mathbf{r}_t$$

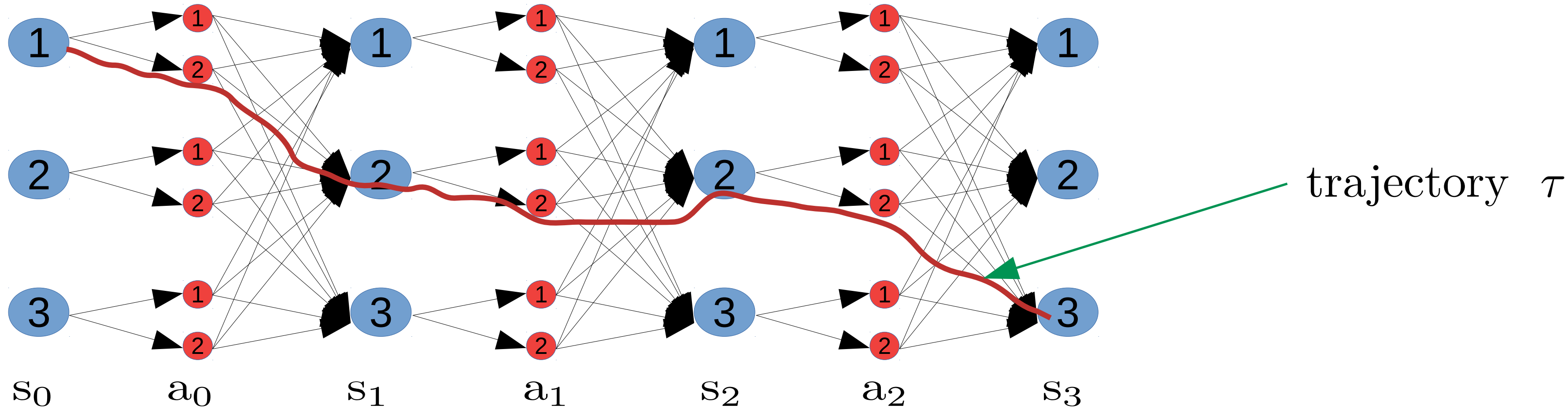
- Total reward of an infinite length trajectory, ***infinite-horizon discounted return***

$$\mathrm{R}(\tau) = \sum_{t=0}^{\infty} \gamma^t \mathbf{r}_t$$

where $\gamma \in (0,1)$ is called a ***discount factor***

# The RL Problem

- Given an environment and agent, find a policy $\pi$ which ***maximizes the expected return*** $\mathrm{J}(\pi)$ when the agent acts according to it.




$\rho(\mathbf{s}_0)$ $\pi(\mathbf{a}_0|\mathbf{s}_0)$ $\mathrm{P}(\mathrm{s}_1|\mathbf{a}_0,\mathbf{s}_0)$ $\pi(\mathbf{a}_0|\mathbf{s}_0)$ $\mathrm{P}(\mathrm{s}_1|\mathbf{a}_0,\mathbf{s}_0)$ $\pi(\mathbf{a}_0|\mathbf{s}_0)$ $\mathrm{P}(\mathrm{s}_1|\mathbf{a}_0,\mathbf{s}_0)$

$$\mathrm{P}_\tau(\tau|\pi) = \rho_0(\mathbf{s}_0 \prod_{t=0}^{T} P(\mathbf{s}_{t+1}|\mathbf{s}_t,\mathbf{a}_t)\pi(\mathbf{a}_t|\mathbf{s}_t)$$

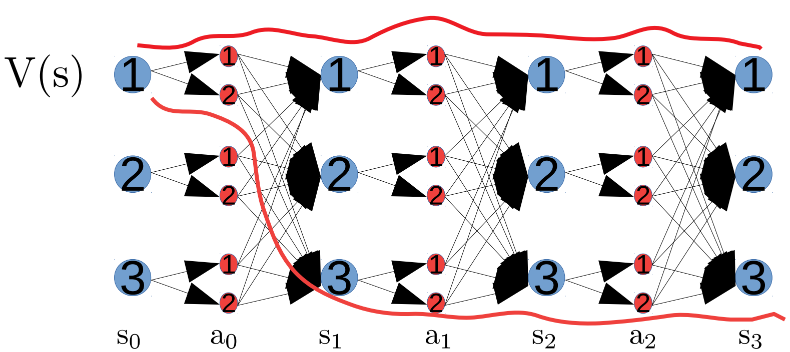
$$\mathrm{R}(\tau) = \sum_t \gamma^t \mathbf{r}_t$$

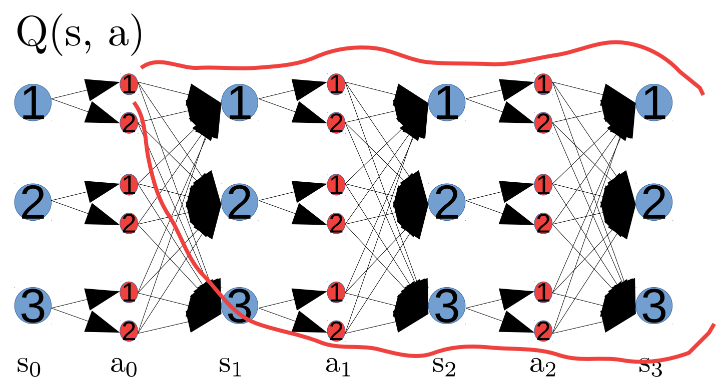
$$\mathrm{J}(\pi) = \sum_\tau P_\tau(\tau|\pi) R(\tau)$$

$$\pi^\star = \arg\max_\pi J(\pi)$$

# Value Functions (I)

- RL problem optimizes the expected(average) return over all trajectories.
- However, sometimes we are interested in the expected return over
  - All trajectories start at a given state (state value function or ***value function*** $V(s)$ )
  - All trajectories start at a given state and taking a given action (state-action value function or ***action value function*** $Q(s, a)$ )



# Value functions (II)

- On-policy value function
  - Expected return when acting according to the policy $\pi$ starting at state $\mathrm{s}$

$$V^{\pi}(\mathbf{s}) = \underset{\tau\sim\pi}{E}[R(\tau)|\mathbf{s_0}=\mathbf{s}]$$

- On-policy action value function
  - Expected return, when starting at $\mathrm{s}$ and taking an action $\mathrm{a}$ and thereafter acting according to the policy $\pi$

$$Q^{\pi}(\mathbf{s},\mathbf{a}) = \underset{\tau\sim\pi}{E}[R(\tau)|\mathbf{s_0}=\mathbf{s},\mathbf{a}_0=\mathbf{a}]$$

- Optimal value function
  - Expected return when acting according to the optimal policy starting at state $\mathrm{s}$

$$V^{\star}(\mathbf{s}) = \max_{\pi}\underset{\tau\sim\pi}{E}[R(\tau)|\mathbf{s_0}=\mathbf{s}]$$

- Optimal action value function
  - Optimal policy version of $Q^{\pi}(\mathbf{s},\mathbf{a})$

$$Q^{\star}(\mathbf{s},\mathbf{a}) = \max_{\pi}\underset{\tau\sim\pi}{E}[R(\tau)|\mathbf{s_0}=\mathbf{s},\mathbf{a}_0=\mathbf{a}]$$

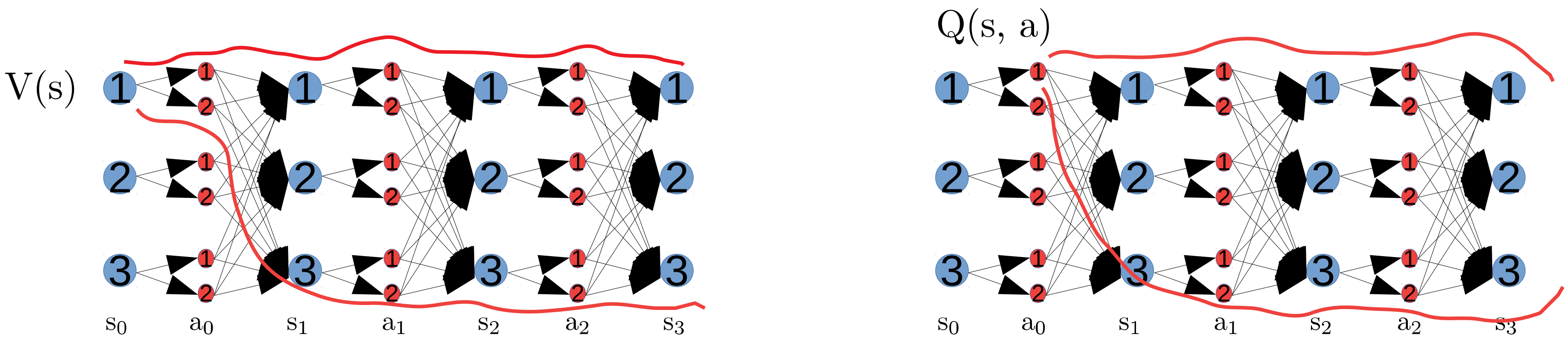
# Relationship between value functions

- On-policy versions

$$V^{\pi}(\mathbf{s}) = \underset{a\sim\pi}{E}[Q^{\pi}(s,\mathbf{a})]$$

- Optimal versions

$$V^{\star}(\mathbf{s}) = \max_{\mathbf{a}} Q^{\star}(s,\mathbf{a})$$



# Value function estimation

- How to calculate the value function?
  - We can apply the definition

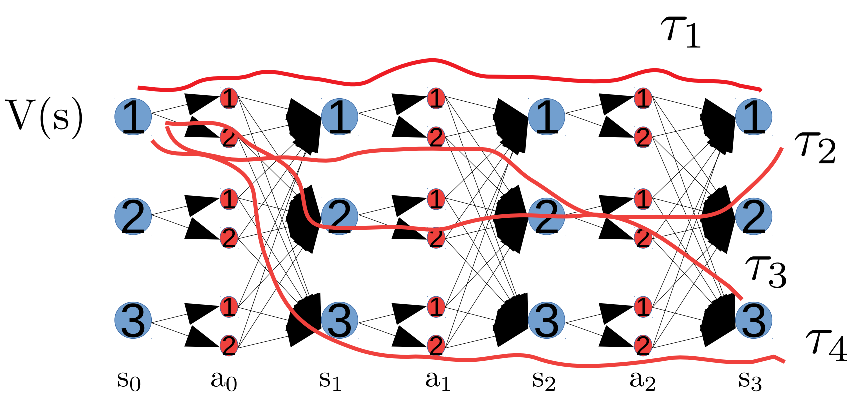
$$
\begin{aligned}
V^{\pi}(\mathbf{s}) &= \underset{\tau\sim\pi}{E}[R(\tau)|\mathbf{s_0}=\mathbf{s}] \\
&= \sum_{\tau\sim\pi} P_{\tau}(\tau)R(\tau|\mathbf{s}_0=\mathbf{s}) \\
&= \sum_{\mathbf{a}_0}\sum_{\mathbf{s}_1}\sum_{\mathbf{a}_1}\cdots\sum_{\mathbf{s}_T}\rho(\mathbf{s}_0)\pi(\mathbf{a}_0|\mathbf{s}_0)P(\mathbf{s}_1|\mathbf{s}_0,\mathbf{a}_0)\cdots P(\mathbf{s}_T|\mathbf{s}_{T-1},\mathbf{a}_{T-1})[r_0+\gamma r_1+\gamma^2 r_2+\cdots+\gamma^T r_T]
\end{aligned}
$$

- We may encounter two problems
  - Summation becomes intractable
  - We do not know the environment model $\mathrm{s}_{t+1} \sim P(\cdot|\mathbf{s}_t,\mathbf{a}_t)$
- Then we can use an approximate method
  - Monte Carlo method
  - Temporal difference method

# Monte Carlo (MC) method

- Value function is an expectation $V^\pi(\mathbf{s}) = \underset{\tau\sim\pi}{E}[R(\tau)|\mathbf{s_0}=\mathbf{s}]$
- Sampling and calculate the sample average
  - Generate sample trajectories $\mathcal{D} = \{\tau_i,\ i=1,2,\cdots,N\}$
  - Find trajectory segments starting at the desired state $\mathbf{s}$
  - Calculate the return for each trajectory segment
  - Find the average of  all such returns.

- Different strategies
  - First visit
  - All visit
  - Incremental update

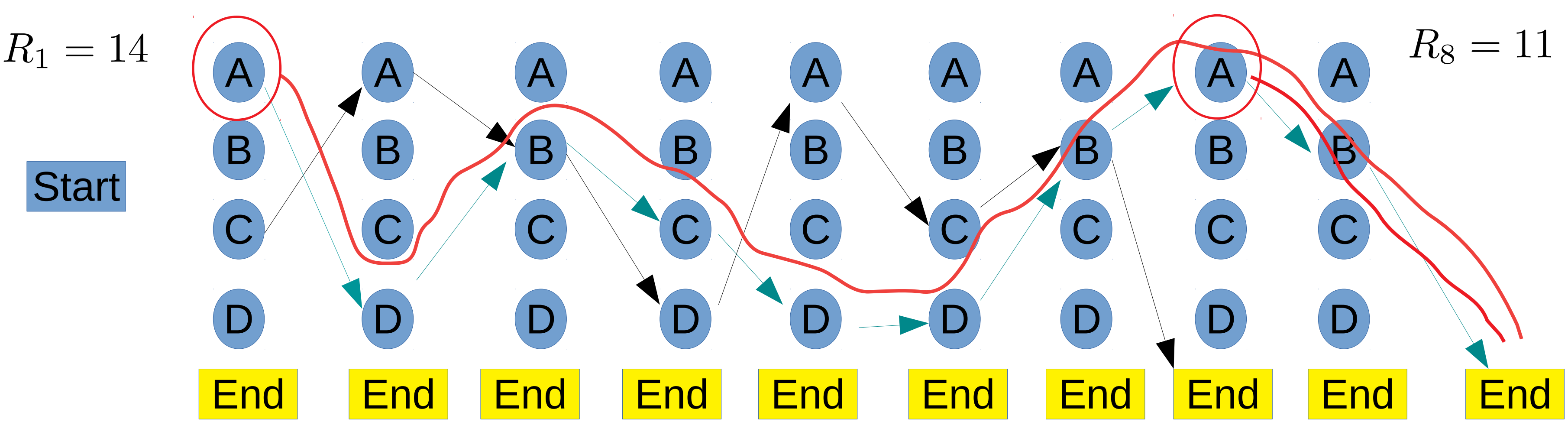


# Monte Carlo example

- Consider a state space consisting of four states A,B,C,D
- Let us assume that we have generated two trajectories (Actions have been omitted and numbers over the arrows are rewards)

$$(1): A \xrightarrow{-4} D \xrightarrow{5} B \xrightarrow{2} C \xrightarrow{0} D \xrightarrow{1} D \xrightarrow{-1} B \xrightarrow{0} A \xrightarrow{3} B \xrightarrow{8} END$$

$$(2): C \xrightarrow{2} A \xrightarrow{-3} B \xrightarrow{0} D \xrightarrow{-1} A \xrightarrow{2} C \xrightarrow{1} B \xrightarrow{0} END$$



# First Visit

- For each trajectory
  - Accumulate the return $\mathrm{R}_t$ for the first visit of the concerned state
- Take the average $\frac{1}{N}\sum R_t$

$$(1): A \xrightarrow{-4} D \xrightarrow{5} B \xrightarrow{2} C \xrightarrow{0} D \xrightarrow{1} D \xrightarrow{-1} B \xrightarrow{0} A \xrightarrow{3} B \xrightarrow{8} END$$
$$(2): C \xrightarrow{2} A \xrightarrow{-3} B \xrightarrow{0} D \xrightarrow{-1} A \xrightarrow{2} C \xrightarrow{1} B \xrightarrow{0} END$$

| $N_t$ | Trajectory 1 | | | | | | | | | | Trajectory 2 | | | | | | | | Total |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Time | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
| A | 0 | 1 | | | | | | | | | | | 2 | | | | | | 2 |
| B | 0 | | | 1 | | | | | | | | | | 2 | | | | | 2 |
| C | 0 | | | | 1 | | | | | | | 2 | | | | | | | 2 |
| D | 0 | | 1 | | | | | | | | | | | | 2 | | | | 2 |

| $R_t$ | Trajectory 1 | | | | | | | | | | Trajectory 2 | | | | | | | | Total | V |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Time | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | | |
| A | 0 | 14 | | | | | | | | | | | -1+14 | | | | | | 13 | 13/2 |
| B | 0 | | | 13 | | | | | | | | | 2+13 | | | | | | 15 | 15/2 |
| C | 0 | | | | 11 | | | | | | | 1+11 | | | | | | | 12 | 12/2 |
| D | 0 | | 18 | | | | | | | | | | | | 2+18 | | | | 20 | 20/2 |

# Every Visit

- For each trajectory
  - Accumulate the return $\mathrm{R}_t$ for <u>every</u> visit of the concerned state
- Take the average $\frac{1}{N}\sum R_t$

$$(1): A \xrightarrow{-4} D \xrightarrow{5} B \xrightarrow{2} C \xrightarrow{0} D \xrightarrow{1} D \xrightarrow{-1} B \xrightarrow{0} A \xrightarrow{3} B \xrightarrow{8} END$$
$$(2): C \xrightarrow{2} A \xrightarrow{-3} B \xrightarrow{0} D \xrightarrow{-1} A \xrightarrow{2} C \xrightarrow{1} B \xrightarrow{0} END$$

| $N_t$ | Trajectory 1 | | | | | | | | | | Trajectory 2 | | | | | | | | Total |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Time | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
| A | 0 | 1 | | | | | | | 2 | | | | 3 | | | 4 | | | 4 |
| B | 0 | | | 1 | | | | 2 | | 3 | | | | 4 | | | | 5 | 5 |
| C | 0 | | | | 1 | | | | | | | 2 | | | | | 3 | | 3 |
| D | 0 | | 1 | | | 2 | 3 | | | | | | | | 4 | | | | 4 |

| $R_t$ | Trajectory 1 | | | | | | | | | | Trajectory 2 | | | | | | | | Total | V |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Time | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | | |
| A | 0 | 14 | | | | | | | 11+14 | | | | -1+25 | | | 3+24 | | | 27 | 27/4 |
| B | 0 | | | 13 | | | | 11+13 | | 8+24 | | | 2+32 | | | | | 0+34 | 34 | 34/5 |
| C | 0 | | | | 11 | | | | | | | 1+11 | | | | | 1+12 | | 13 | 13/3 |
| D | 0 | | 18 | | | 11+18 | 10+29 | | | | | | | | 2+39 | | | | 41 | 41/4 |

# Incremental Update

- Instead of taking average at the end, calculate a running average
- For each trajectory
  - Update the value $V(\mathbf{s})$ every time the concerned state $\mathbf{s}$ is visited.

$$V_t(\mathbf{s}) \leftarrow V_{t-1}(\mathbf{s}) + \frac{1}{t}(R_t - V_{t-1}(\mathbf{s}))$$

$V_t(\mathbf{s}) = \text{Value at the } t^{\text{th}} \text{ visit},\quad V_{t-1}(\mathbf{s}) = \text{Value at the previous visit},\quad R_t(\mathbf{s}) = \text{Return after the } t^{\text{th}} \text{ visit}$

$$(1): A \xrightarrow{-4} D \xrightarrow{5} B \xrightarrow{2} C \xrightarrow{0} D \xrightarrow{1} D \xrightarrow{-1} B \xrightarrow{0} A \xrightarrow{3} B \xrightarrow{8} END$$
$$(2): C \xrightarrow{2} A \xrightarrow{-3} B \xrightarrow{0} D \xrightarrow{-1} A \xrightarrow{2} C \xrightarrow{1} B \xrightarrow{0} END$$

| | Trajectory 1 | | | | | | | | | | Trajectory 2 | | | | | | | | V |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Time | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
| A | 0 | (14-0)/1 | | | | | | | 14+(11-14)/2 | | | | 12.5+(-1-12.5)/3 | | | 8+(3-8)/4 | | | 6,75 |
| B | 0 | | | (13-0)/1 | | | | 13+(11-13)/2 | | 12+(8-12)/3 | | | 10.66+(2-10.66)/4 | | | | | 8,495+(0-8.495)/5 | 6.8 |
| C | 0 | | | | (11-0)/1 | | | | | | | 11+(1-11)/2 | | | | | 6+(1-6)/3 | | 4.3 |
| D | 0 | | (18-0)/1 | | | 18+(11-18)/2 | 14.5+(10-14.5)/3 | | | | | | | | 13+(2-13)/4 | | | | 10.25 |