Speech and Language Processing

Lecture 5
Reinforcement Learning (1)
Value Functions

Information and Communications Engineering Course

Takahiro Shinozaki

Manabu Okumura

2025/10/10

Reinforcement Learning

In some applications, AI has become stronger than human by reinforcement learning breaking away from supervised learning

Figure is cited from [1]

Dilaudid, CC BY-SA 3.0, via Wikimedia Commons

Figure is cited from [3]

[1] V. Mnih+, "Playing Atari with Deep Reinforcement Learning," NIPS Deep Learning Workshop, 2013

[2] D. Silver+, "Mastering the game of Go without human knowledge," Nature 2017

[3] N. Bhonker+, "Playing SNES in the Retro Learning Environment," arXiv, 2018

Dialogue System Training

- By using RL, the agent can learn rational behavior flexibly

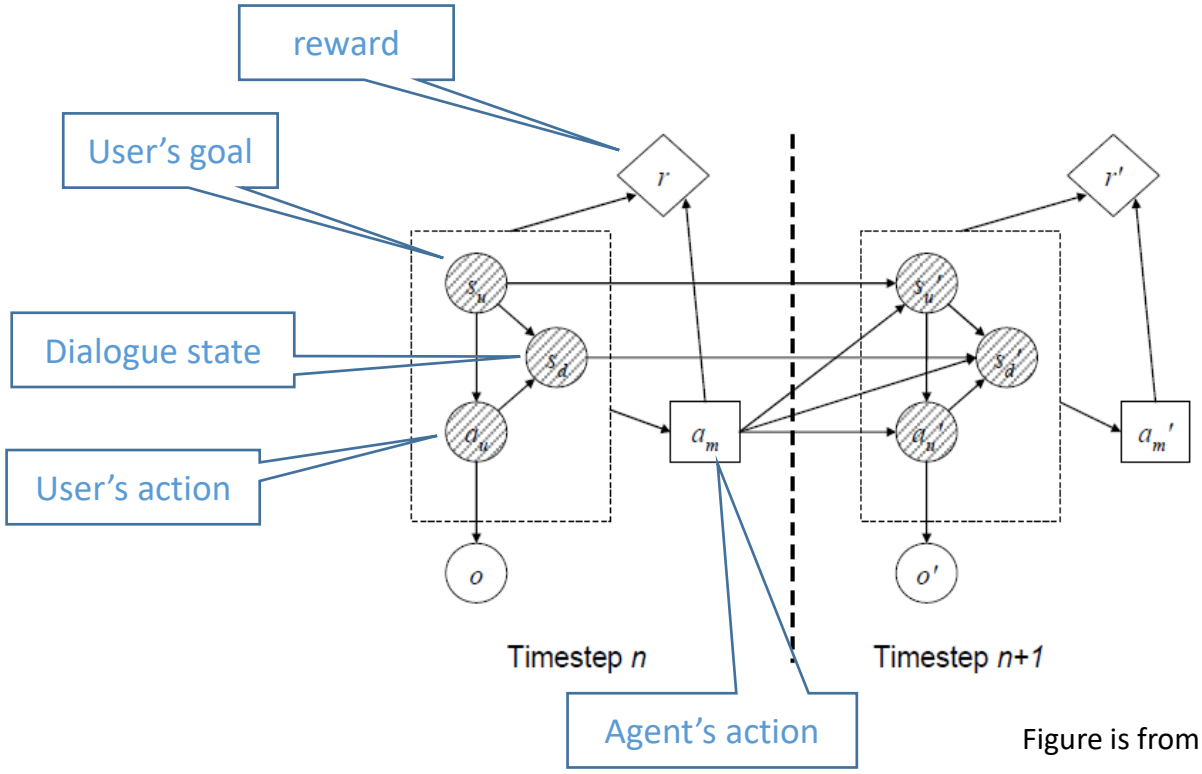


Figure is from [2]

[1] E. Levin+, "A Stochastic Model of Computer-Human Interaction For Learning Dialogue Strategies," Eurospeech, 1997

[2] J. Williams+, "Factored Partially Observable Markov Decision Processes for Dialogue Management," Proc. Knowledge and Reasoning in Practical Dialog Systems, 2005

Human Language Acquisition

B.F. Skinner

Silly rabbit, CC BY 3.0, via Wikimedia Commons

Reinforcement Learning

Language acquisition could be explained by mechanisms of operant conditioning (OC)

[1] B. F. Skinner. "Verbal behavior," New York: Appleton-Century-Crofts, 1957.

Difference From Supervised Learning

Supervised Learning

At each time step, the correct action to take is explicitly provided as labeled data, and the robot (agent) learns based on these pre-taught patterns of behavior.

Reinforcement Learning

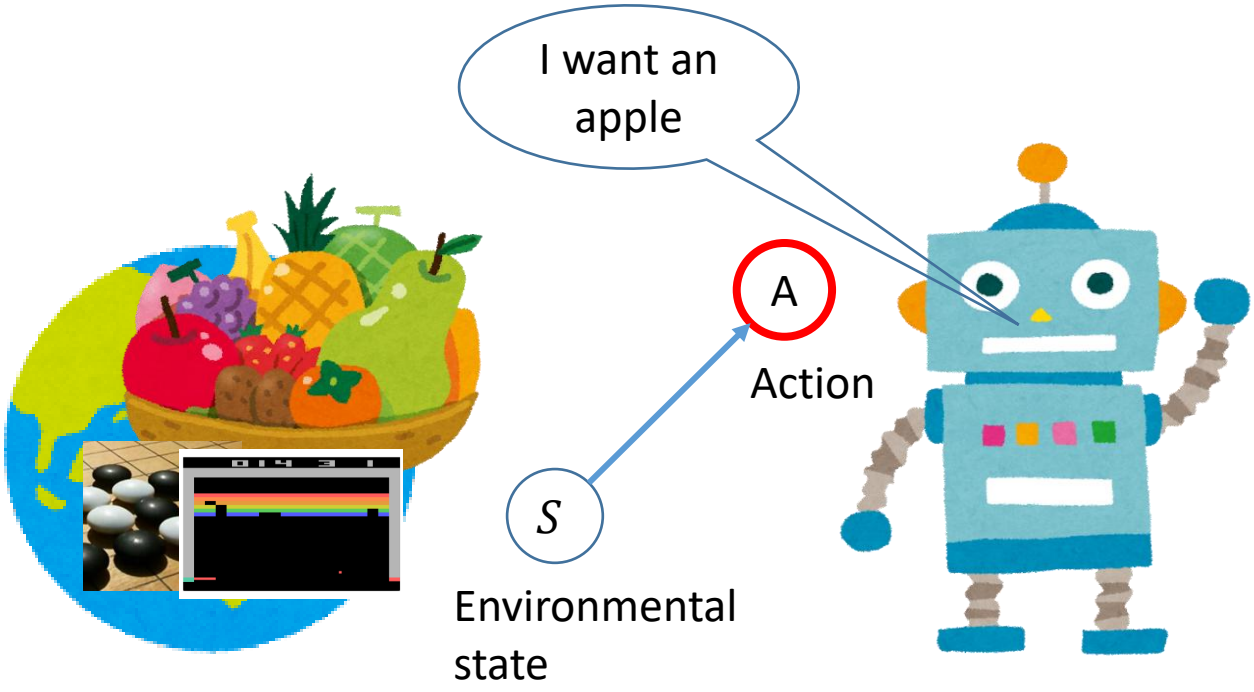
The robot (agent) interacts with the environment and receives rewards. It learns autonomously through experience, developing a strategy (policy) to maximize the cumulative reward over time.

Demo video of automatic game playing (last visited 2024/10/2)
https://www.bing.com/videos/riverview/relatedvideo?q=Playing%20Atari%20with%20Deep%20Reinforcement%20Learning&mid=8680C3FA93A1F93F02FF8680C3FA93A1F93F02FF&ajaxhist=0

Agents in Reinforcement Learning

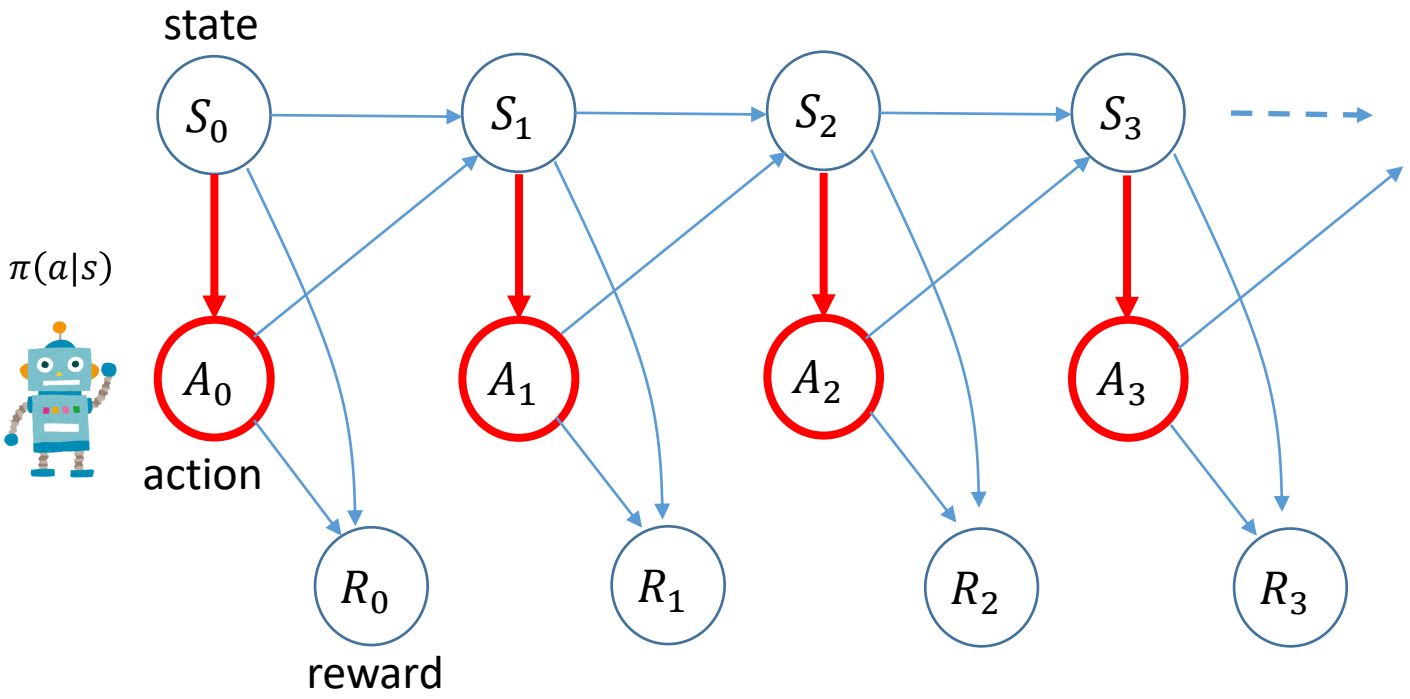
At each time frame, agents observe the environmental state and perform an action. An agent's behavior can be described by a policy function π that takes the environmental state and returns an action.

The policy function can be either stochastic, $\pi(a|s)$, where actions are chosen based on a probability distribution, or deterministic, where the action is directly determined by the state, a=$\pi(s)$.



Agent-Environment Interaction

- A policy π defines the behavior of an agent
- The ultimate goal of reinforcement learning is for the agent to discover the optimal policy π that maximizes cumulative rewards through interaction with the environment



Basic Formulations and Value-Functions

Markov Process (MP)

A Markov Process is a tuple $\langle S, I, T \rangle$

- S: set of states
- I: initial distribution $I(s) = P(S_0 = s)$
- T: state transition $T_t(s, s') = P(S_{t+1} = s' | S_t = s)$

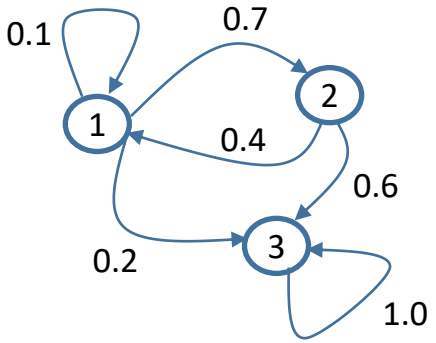
S_t is a random variable that represent the state at time t. The probability of the next state is solely determined by the current state.

When the state is discrete $S = \{1,2, \cdots, N\}$, and the transition is time invariant $T_t(s, s') = T_{t'}(s, s') = T(s, s')$, we can represent the initial distribution I by a N-dimensional vector whose s -th element is $I(s)$ and the state transition T by a $N \times N$ matrix whose $\langle s, s' \rangle$ element is $T(s, s')$. In the followings, we assume the transition is time invariant.

Example: $S = \{1,2,3\}$

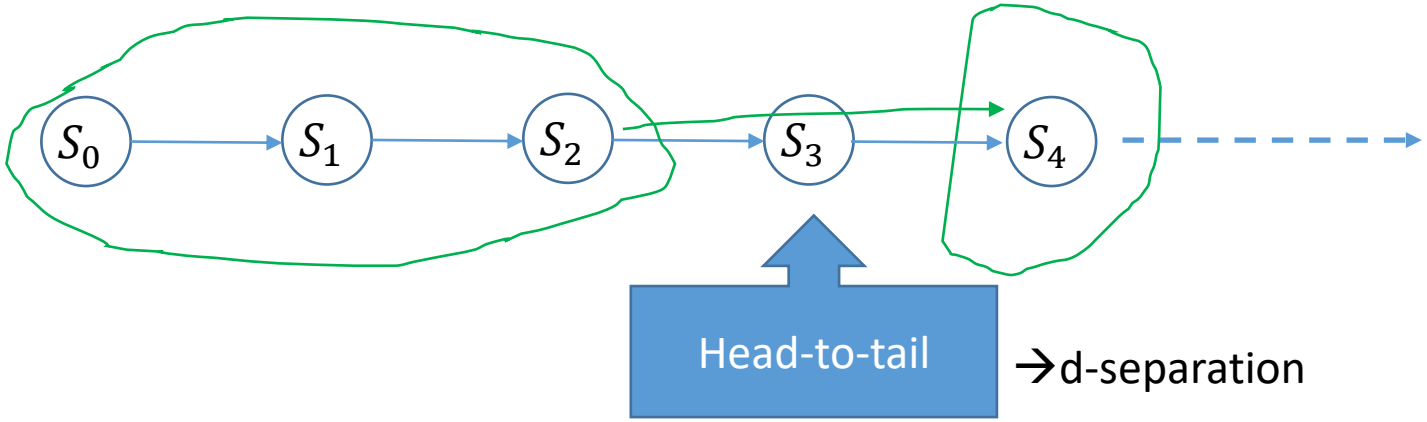
$$I = [1.0 \quad 0.0 \quad 0.0]^T$$

$$T = \begin{bmatrix} 0.1 & 0.7 & 0.2 \\ 0.4 & 0 & 0.6 \\ 0 & 0 & 1.0 \end{bmatrix}$$



BN representation and Markov Property

Because the probability of the next state is determined by the current state, Bayesian network (BN) representation has a linear structure.



→d-separation

$$\{S_{t+1}\} \perp \{S_0, S_1, \cdots, S_{t-1}\} | \{S_t\}$$

For arbitral t,

$$P(S_{t+1}|S_0, S_1, S_2, \cdots, S_t) = P(S_{t+1}|S_t)$$

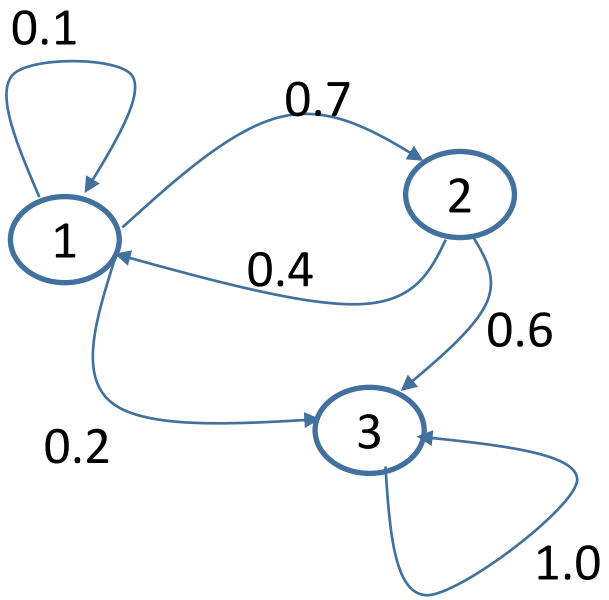
Exercise 5.1

Obtain $P(S_0 = 1, S_1 = 2, S_2 = 3)$ with the following Markov Process

$S = \{1,2,3\}$

$I = [1.0 \quad 0.0 \quad 0.0]^T$

$$T = \begin{bmatrix} 0.1 & 0.7 & 0.2 \\ 0.4 & 0 & 0.6 \\ 0 & 0 & 1.0 \end{bmatrix}$$



Hint:

$P(S_0 = 1, S_1 = 2, S_2 = 3)$

$= P(S_0 = 1)\, P(S_1 = 2|S_0 = 1\,)\, P(S_2 = 3|S_0 = 1, S_1 = 2)$

$= P(S_0 = 1)\, P(S_1 = 2|S_0 = 1\,)\, P(S_2 = 3|S_1 = 2)$

Probability of State in Sequence

$$P(S_0, S_1, \cdots, S_t) = P(S_0)P(S_1|S_0)P(S_2|S_0, S_1)\cdots P(S_t|S_0, S_1, \cdots, S_{t-1}) = P(S_0)P(S_1|S_0)\cdots P(S_t|S_{t-1})$$

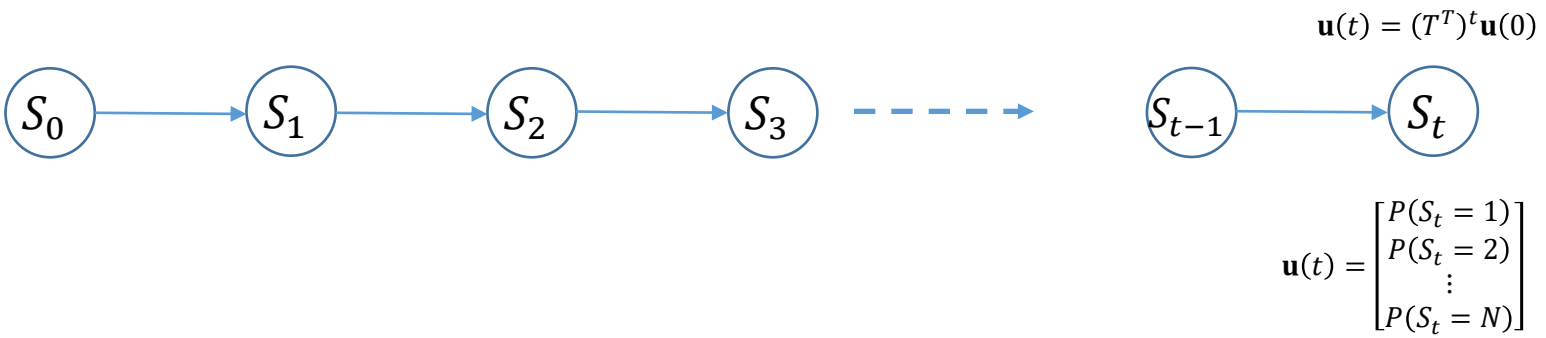
Let $\mathbf{u}(0) = I,\ \mathbf{u}(t) = \begin{bmatrix} P(S_t = 1) \\ P(S_t = 2) \\ \vdots \\ P(S_t = N) \end{bmatrix}$

Then because $P(S_t) = \sum_{S_{t-1}} P(S_{t-1}, S_t) = \sum_{S_{t-1}} P(S_t|S_{t-1})P(S_{t-1})$, we have:

$$\mathbf{u}(t) = T^T\mathbf{u}(t-1) = (T^T)^k\mathbf{u}(t-k) = \left(T^k\right)^T\mathbf{u}(t-k) = (T^t)^T\mathbf{u}(0)$$

T^k is a transition matrix from the current to k-th future.

$T^k_{ss'} = P(S_{t+k} = s'|S_t = s)$



Markov Reward Process (MRP)

A Markov Reward Process is a tuple $\langle S, I, T, R, \gamma \rangle$

- S: set of states
- I: initial distribution $I(s) = P(S_0 = s)$
- T: state transition $T(s, s') = P(S_{t+1} = s' | S_t = s)$
- R: reward distribution $R(s) = P(R_t = r | S_t = s)$
- γ: discount factor $\gamma \in [0, 1]$

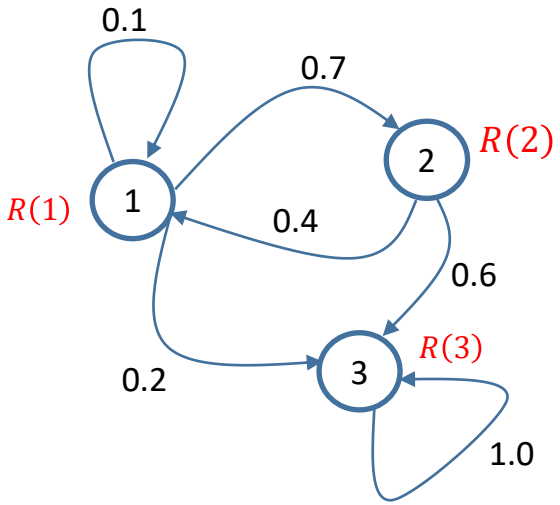
(We assume R is time invariant $R(s) = P(R_t = r | S_t = s) = P(R_u | S_u = s)$)

Example:

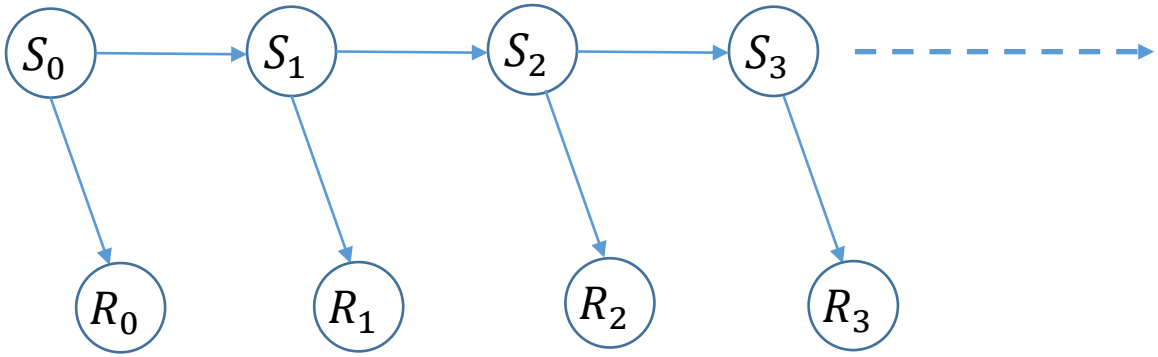
$$S = \{1,2,3\}$$

$$I = [1.0 \quad 0.0 \quad 0.0]$$

$$T = \begin{bmatrix} 0.1 & 0.7 & 0.2 \\ 0.4 & 0 & 0.6 \\ 0 & 0 & 1.0 \end{bmatrix}$$



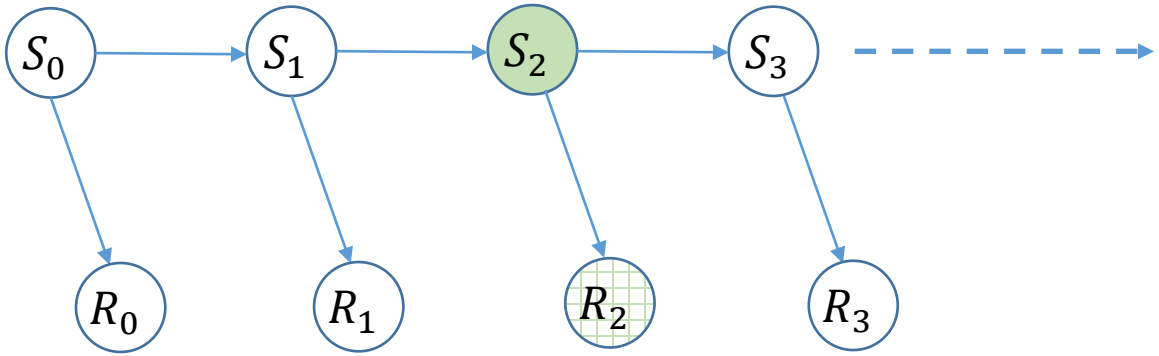
Bayesian Network (BN) Representation



Reward Function

- A reward function is an expectation of reward given a state
- It is independent of time t and is a function of state s

$$\bar{r}(s) = \bar{r}_t(s) = E[R_t = r | S_t = s] = \sum_r r\, P(R_t = r | S_t = s)$$



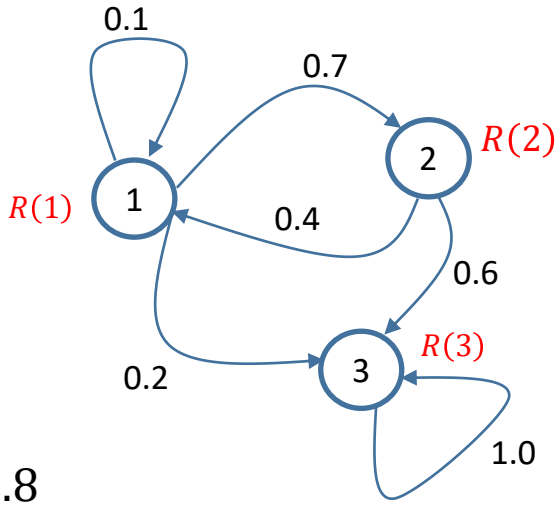
Example Question

Obtain $P(S_0 = 1, R_0 = 0, S_1 = 2, R_1 = 0, S_2 = 3, R_2 = 1.0)$ with the following Markov Reward Process

$$S = \{1,2,3\}$$

$$I = [1.0 \quad 0.0 \quad 0.0]$$

$$T = \begin{bmatrix} 0.1 & 0.7 & 0.2 \\ 0.4 & 0 & 0.6 \\ 0 & 0 & 1.0 \end{bmatrix}$$



$$P(R_t = 0|S_t = 1) = 0.2, \qquad P(R_t = 1|S_t = 1) = 0.8$$

$$P(R_t = 0|S_t = 2) = 0.2, \qquad P(R_t = 1|S_t = 2) = 0.8$$

$$P(R_t = 0|S_t = 3) = 0.7, \qquad P(R_t = 1|S_t = 3) = 0.3$$

$$\gamma = 0.8$$

Answer

$P(S_0 = 1, R_0 = 0, S_1 = 2, R_1 = 0, S_2 = 3, R_2 = 1.0)$

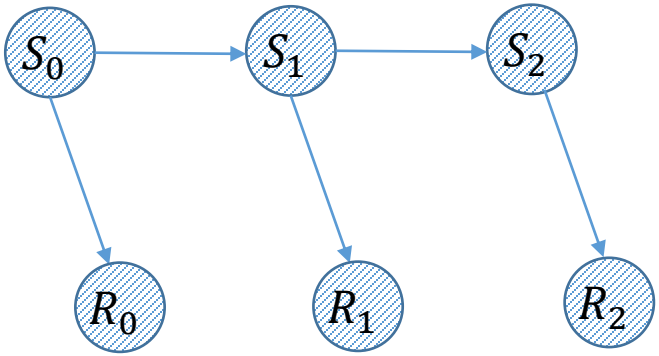
$= P(S_0 = 1)P(R_0 = 0|S_0 = 1)P(S_1 = 2|S_0 = 1, R_0 = 0)\ P(R_1 = 0|S_0 = 1, R_0 = 0, S_1 = 2)$

$P(S_2 = 3|S_0 = 1, R_0 = 0, S_1 = 2, R_1 = 0)\ P(R_2 = 1.0|S_0 = 1, R_0 = 0, S_1 = 2, R_1 = 0, S_2 = 3)$

$= P(S_0 = 1)P(R_0 = 0|S_0 = 1)P(S_1 = 2|S_0 = 1)\ P(R_1 = 0|S_1 = 2)\ P(S_2 = 3|S_1 = 2)\ P(R_2 = 1.0|S_2 = 3)$

$= 1.0 \times 0.2 \times 0.7 \times 0.2 \times 0.6 \times 0.3$

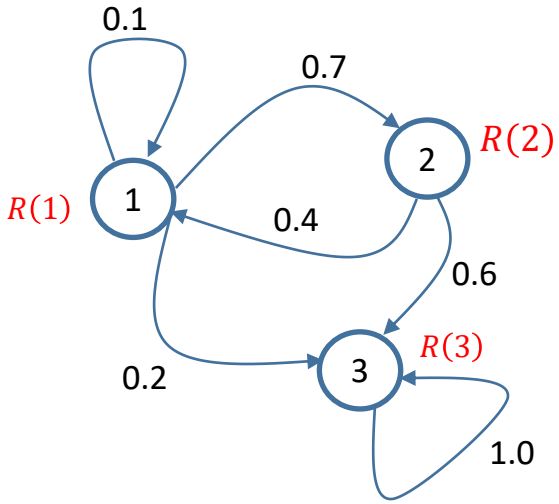
$= 0.00504$



$P(R_t = 0|S_t = 1) = 0.2,$ $P(R_t = 1|S_t = 1) = 0.8$

$P(R_t = 0|S_t = 2) = 0.2,$ $P(R_t = 1|S_t = 2) = 0.8$

$P(R_t = 0|S_t = 3) = 0.7,$ $P(R_t = 1|S_t = 3) = 0.3$

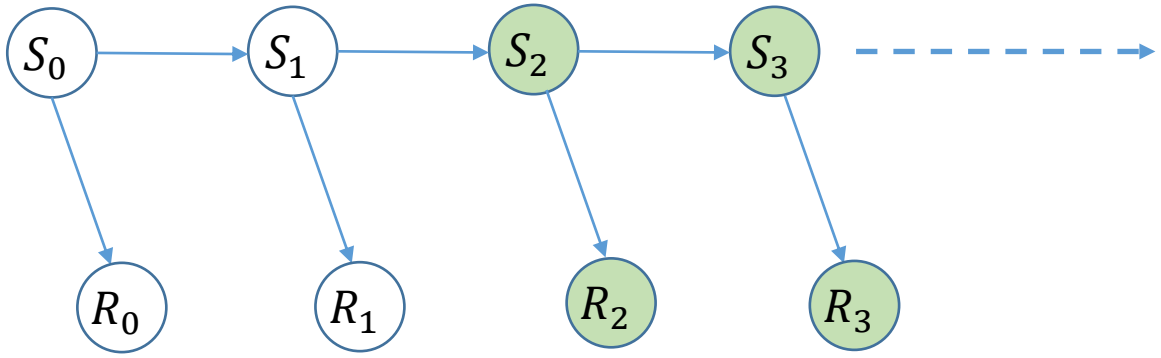


Return

For a trajectory $(S_t = s_t, R_t = r_t, S_{t+1} = s_{t+1}, R_{t+1} = r_{t+1}, \cdots)$ starting from time-step t, return G_t is a sum of discounted rewards

$$G_t = r_t + \gamma r_{t+1} + \cdots = \sum_{k=0}^{\infty} \gamma^k r_{t+k}$$

e.g. G_2

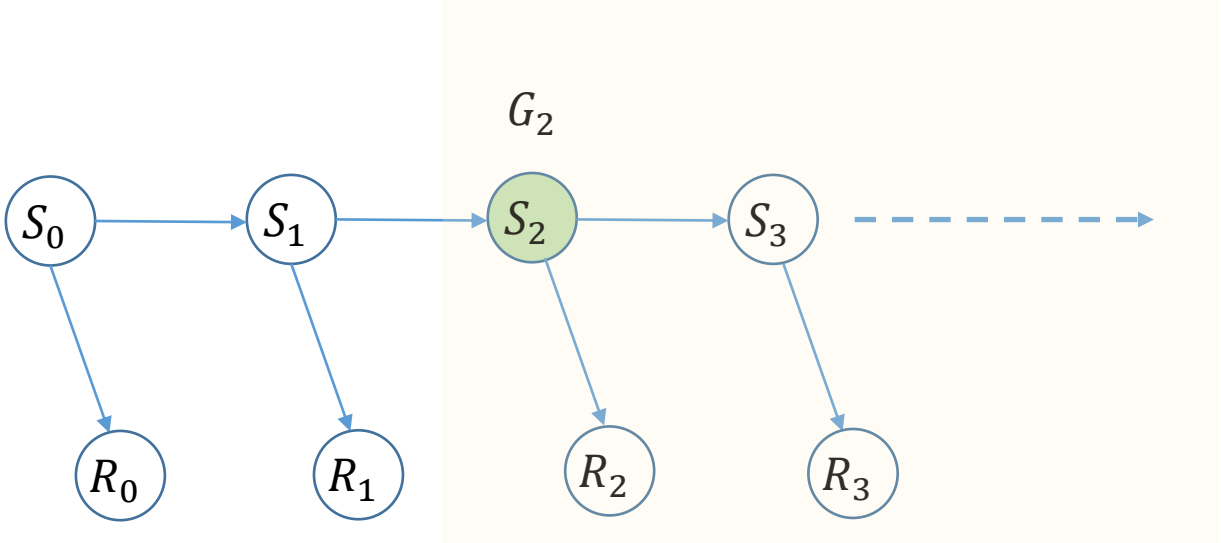


State-Value Function

State-value function $v(s)$ is the expected return starting from state s

$$v_t(s) = \sum_{r_t, s_{t+1}, r_{t+1}, s_{t+2}, \cdots, s_\infty} P(r_t, s_{t+1}, r_{t+1}, s_{t+2}, \cdots, s_\infty | s_t = s)\, G_t = E[G_t | S_t = s]$$

$v(s) = v_t(s)$ for any t



e.g. $v_2(s) = E[G_2 | S_2 = s]$

Markov Decision Process (MDP) and Policy

- A Markov Decision Process is a tuple $\langle S, A, I, T, R, \gamma \rangle$
  - S: set of states
  - A: set of actions (action space)
  - I: initial distribution $I(s) = P(S_0 = s)$
  - T: state transition $T(a, s, s') = P(S_{t+1} = s' | S_t = s, A_t = a)$
  - R: reward distribution $R(a, s) = P(R_t = r | S_t = s, A_t = a)$
  - γ: discount factor $\gamma \in [0, 1]$
- A policy π is a distribution over actions given states
  - $\pi(a|s) = P(A_t = a | S_t = s)$

Example

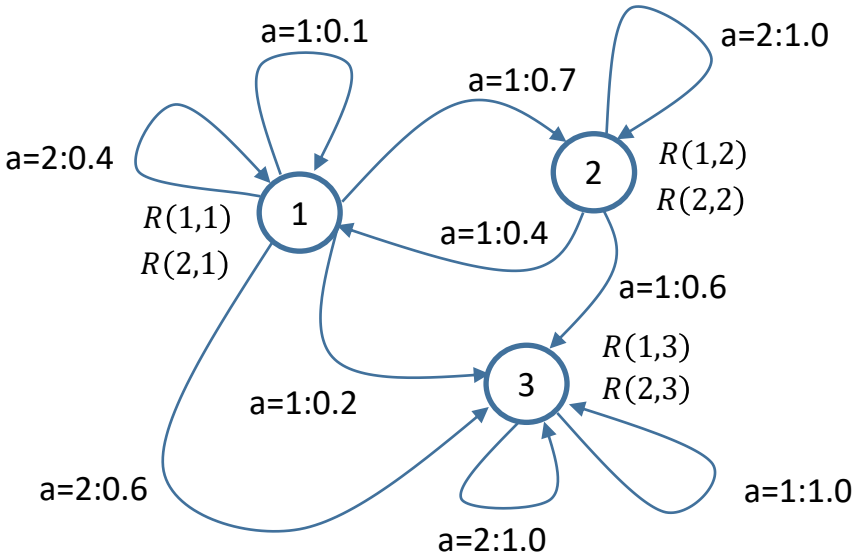
$S = \{1,2,3\}$

$A = \{1,2\}$

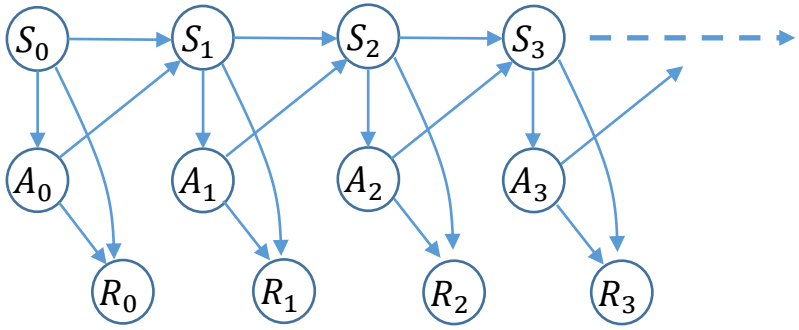
$I = [1.0 \quad 0.0 \quad 0.0]$

$$T_{a=1} = \begin{bmatrix} 0.1 & 0.7 & 0.2 \\ 0.4 & 0 & 0.6 \\ 0 & 0 & 1.0 \end{bmatrix}$$

$$T_{a=2} = \begin{bmatrix} 0.4 & 0 & 0.6 \\ 0 & 1.0 & 0 \\ 0 & 0 & 1.0 \end{bmatrix}$$

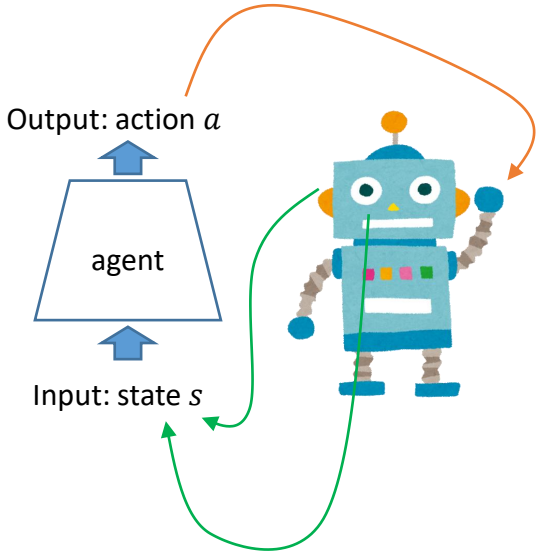


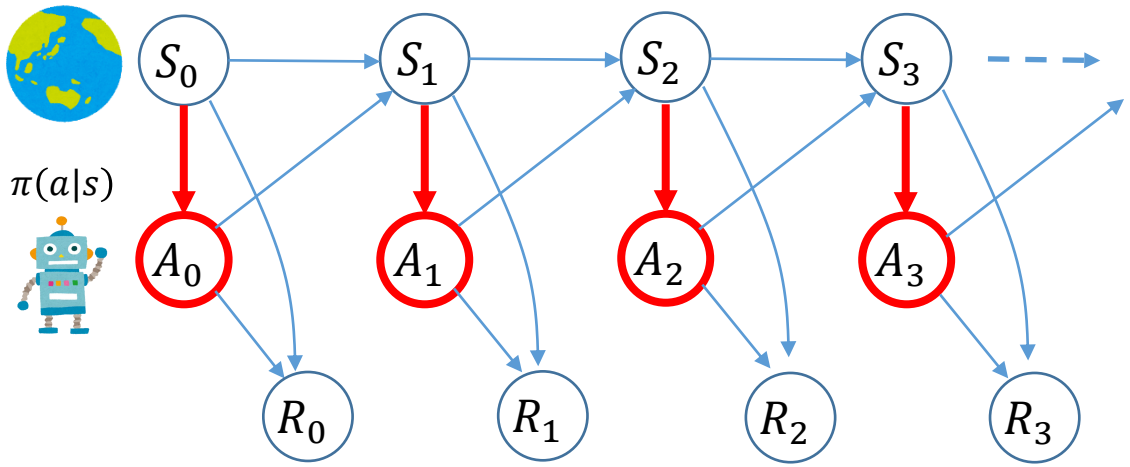
$R(1,1)$	$P(R_t = 0 \mid S_t = 1, a = 1) = 0.2,$	$P(R_t = 1 \mid S_t = 1, a = 1) = 0.8$
$R(2,1)$	$P(R_t = 0 \mid S_t = 1, a = 2) = 0.6,$	$P(R_t = 1 \mid S_t = 1, a = 2) = 0.4$
$R(1,2)$	$P(R_t = 0 \mid S_t = 2, a = 1) = 0.3,$	$P(R_t = 1 \mid S_t = 2, a = 1) = 0.7$
$R(2,2)$	$P(R_t = 0 \mid S_t = 2, a = 2) = 0.2,$	$P(R_t = 1 \mid S_t = 2, a = 2) = 0.8$
$R(1,3)$	$P(R_t = 0 \mid S_t = 3, a = 1) = 0.3,$	$P(R_t = 1 \mid S_t = 3, a = 1) = 0.7$
$R(2,3)$	$P(R_t = 0 \mid S_t = 3, a = 2) = 0.7,$	$P(R_t = 1 \mid S_t = 3, a = 2) = 0.3$



Policy and Environment

- A policy π defines the behavior of an agent
- The goal of the agent learning is to find optimal π to obtain the maximum return





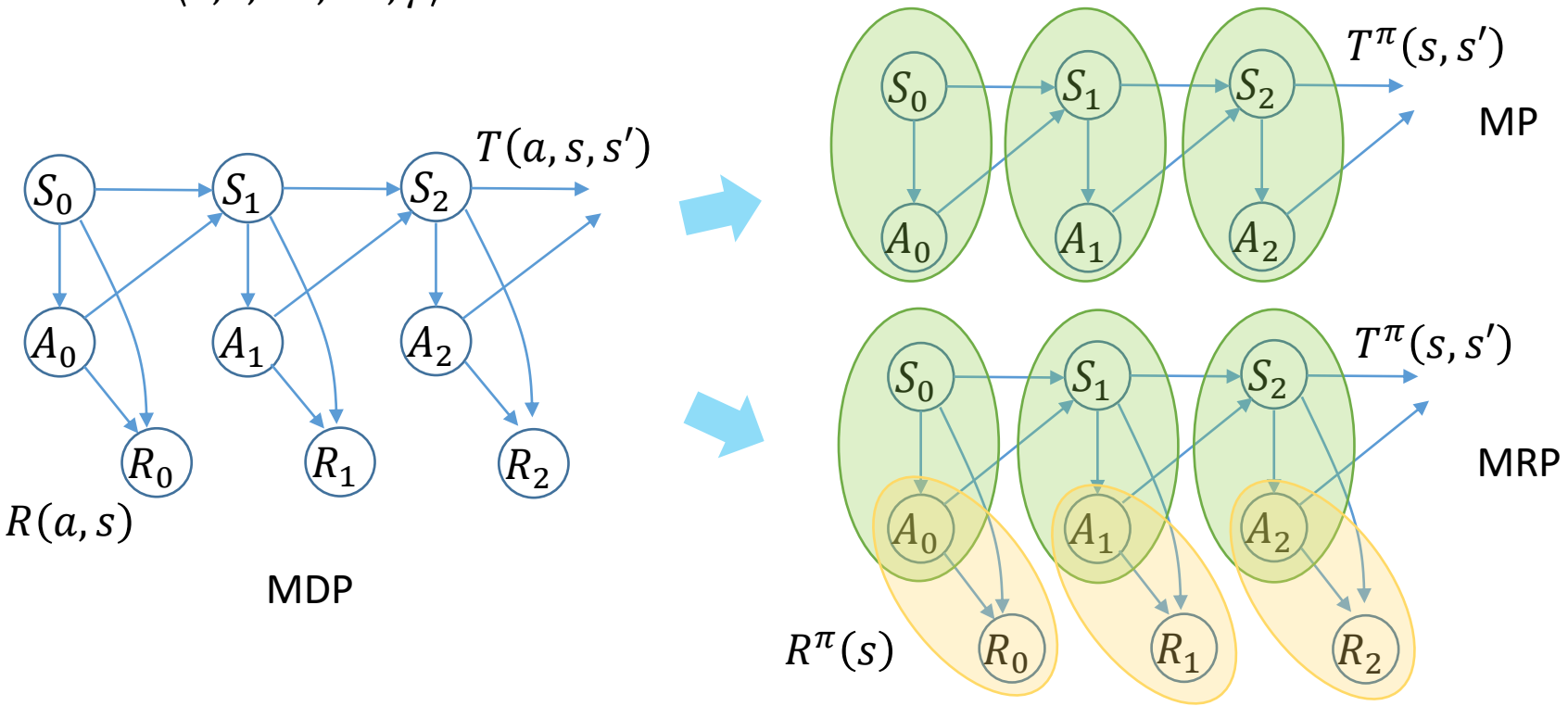
Model

- For a Markov Decision Process $\langle S, A, I, T, R, \gamma \rangle$, a model (in the context of reinforcement learning) is a parametrized representation of $\langle S, A, I, T, R \rangle$
- Model based reinforcement learning
  - Learning algorithms that directly access I, T and R
- Model free reinforcement learning
  - Learning algorithms that do not directly access I, T and R
  - Instead, gets samples of actions and rewards by interacting with the environment

Induced MP and MRP from MDP

For an MDP $\langle S, A, I, T, R, \gamma \rangle$ and a policy π,

- Let $T^{\pi}(s, s') = \sum_{a \in A} \pi(a|s) T(a, s, s') = \sum_{a \in A} \pi(a|s) P(S_{t+1} = s' | S_t = s, A_t = a)$.
  Then $\langle S, I, T^{\pi} \rangle$ is a MP
- Let $R^{\pi}(s) = \sum_{a \in A} \pi(a|s) R(a, s) = \sum_{a \in A} \pi(a|s) P(R_t | S_t = s, A_t = a)$.
  Then $\langle S, I, T^{\pi}, R^{\pi}, \gamma \rangle$ is a MRP



Exercise 5.2

$S = \{1,2,3\}$

$A = \{1,2\}$

$I = [1.0 \quad 0.0 \quad 0.0]$

$$T_{a=1} = \begin{bmatrix} 0.1 & 0.7 & 0.2 \\ 0.4 & 0 & 0.6 \\ 0 & 0 & 1.0 \end{bmatrix}$$

$$T_{a=2} = \begin{bmatrix} 0.4 & 0 & 0.6 \\ 0 & 1.0 & 0 \\ 0 & 0 & 1.0 \end{bmatrix}$$

$\pi(1|1) = 0.4$

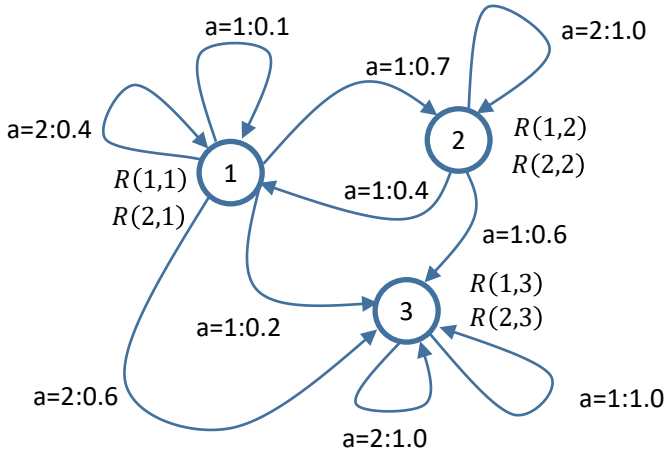
$\pi(2|1) = 0.6$

$\pi(1|2) = 0.8$

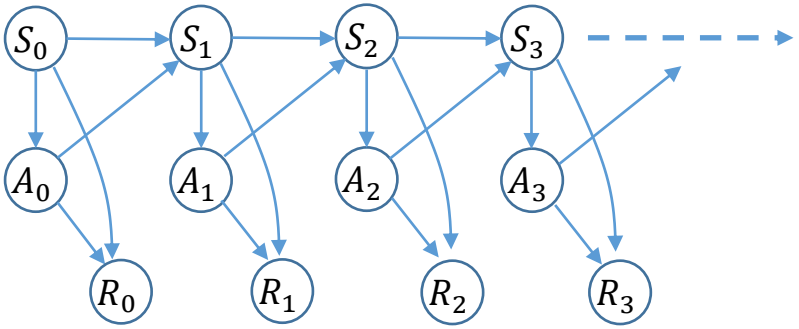
$\pi(2|2) = 0.2$

$\pi(1|3) = 0.3$

$\pi(2|3) = 0.7$



$R(1,1)$	$P(R_t = 0 \mid S_t = 1, a = 1) = 0.2,$	$P(R_t = 1 \mid S_t = 1, a = 1) = 0.8$
$R(2,1)$	$P(R_t = 0 \mid S_t = 1, a = 2) = 0.6,$	$P(R_t = 1 \mid S_t = 1, a = 2) = 0.4$
$R(1,2)$	$P(R_t = 0 \mid S_t = 2, a = 1) = 0.3,$	$P(R_t = 1 \mid S_t = 2, a = 1) = 0.7$
$R(2,2)$	$P(R_t = 0 \mid S_t = 2, a = 2) = 0.2,$	$P(R_t = 1 \mid S_t = 2, a = 2) = 0.8$
$R(1,3)$	$P(R_t = 0 \mid S_t = 3, a = 1) = 0.3,$	$P(R_t = 1 \mid S_t = 3, a = 1) = 0.7$
$R(2,3)$	$P(R_t = 0 \mid S_t = 3, a = 2) = 0.7,$	$P(R_t = 1 \mid S_t = 3, a = 2) = 0.3$



5.2) Obtain $T^{\pi}(1,2)$

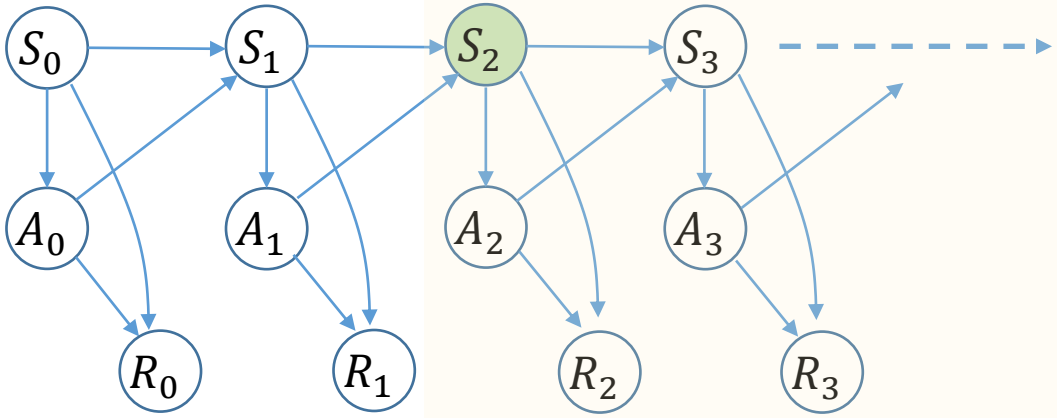
State-Value Function of MDP

State-value function $v^{\pi}(s)$ of MDP is an expected return starting from state s and following the policy π

$$v_t^{\pi}(s) = \sum_{a_t, r_t, s_{t+1}, a_{t+1}, r_{t+1}, s_{t+2}, \cdots, s_{\infty}} P(a_t, r_t, s_{t+1}, a_{t+1}, r_{t+1}, s_{t+2}, \cdots, s_{\infty} | s_t = s)\, G_t$$

$$= E[G_t | S_t = s]$$

$v_t^{\pi}(s) = v^{\pi}(s)$ for any t



e.g. $E[G_2 | S_2 = s]$

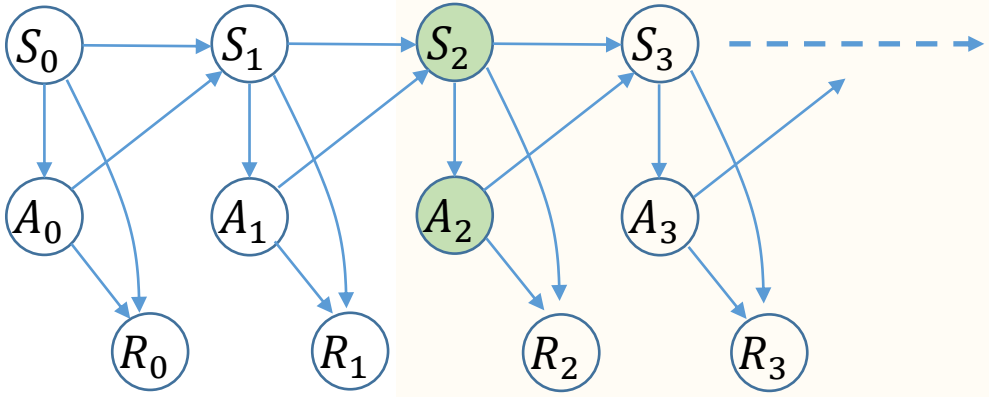
Action-Value Function of MDP

Action-value function $q_\pi(s,a)$ is an expected return starting from state s, taking action a, and then following policy π

$$q_t^\pi(s,a)$$

$$= \sum_{r_t, s_{t+1}, a_{t+1}, r_{t+1}, s_{t+2}, \cdots, s_\infty} P(r_t, s_{t+1}, a_{t+1}, r_{t+1}, s_{t+2}, \cdots, s_\infty | S_t = s, A_t = a)\, G_t$$

$$= E[G_t | S_t = s, A_t = a] \qquad q_t^\pi(s,a) = q^\pi(s,a) \text{ for any } t$$



e.g. $E[G_2 | S_2 = s, A_2 = a]$

Appendix

A Variation of Bellman Equation

Multiply at both sides by $\gamma^2 T^2$, and then subtract

$$\boldsymbol{V} = \boldsymbol{R} + \gamma T \boldsymbol{R} + \gamma^2 T^2 \boldsymbol{R} + \gamma^3 T^3 \boldsymbol{R} \cdots + \gamma^K T^K \boldsymbol{R}$$

$$\gamma^2 T^2 \boldsymbol{V} = \gamma^2 T^2 \boldsymbol{R} + \gamma^3 T^3 \boldsymbol{R} \cdots + \gamma^K T^K \boldsymbol{R} + \gamma^{K+1} T^{K+1} \boldsymbol{R} + \gamma^{K+2} T^{K+2} \boldsymbol{R}$$

$K \to \infty$

$$\boldsymbol{V} = \boldsymbol{R} + \gamma T \boldsymbol{R} + \gamma^2 T^2 \boldsymbol{V}$$

Bellman Equation
(in matrix form)

Derivation of Bellman Equation for $q^{\pi}(s,a)$

$$q^{\pi}(s,a) = \sum_{r_t, s_{t+1}, a_{t+1}, r_{t+1}, s_{t+2}, \cdots, s_{\infty}} P(r_t, s_{t+1}, a_{t+1}, r_{t+1}, s_{t+2}, \cdots, s_{\infty} | S_t = s, A_t = a) \sum_{k=0}^{\infty} \gamma^k r_{t+k}$$

$$= \sum_{r_t} P(r_t | S_t = s, A_t = a)\, r_t + \sum_{s_{t+1}, a_{t+1}, r_{t+1}, s_{t+2}, \cdots, s_{\infty}} P(s_{t+1}, a_{t+1}, r_{t+1}, s_{t+2}, \cdots, s_{\infty} | S_t = s, A_t = a) \sum_{k=1}^{\infty} \gamma^k r_{t+k}$$

$$= \bar{r}(a,s) + \sum_{s_{t+1}, a_{t+1}, r_{t+1}, s_{t+2}, \cdots, s_{\infty}} P(s_{t+1} | S_t = s, A_t = a) P(a_{t+1} | S_t = s, A_t = a, s_{t+1}) P(r_{t+1}, s_{t+2}, \cdots, s_{\infty} | S_t = s, A_t = a, s_{t+1}, a_{t+1}) \sum_{k=1}^{\infty} \gamma^k r_{t+k}$$

Use the conditional independence structure of MDP

$$= \bar{r}(a,s) + \sum_{s_{t+1}, a_{t+1}, r_{t+1}, s_{t+2}, \cdots, s_{\infty}} P(s_{t+1} | S_t = s, A_t = a) P(a_{t+1} | s_{t+1}) P(r_{t+1}, s_{t+2}, \cdots, s_{\infty} | s_{t+1}, a_{t+1}) \sum_{k=1}^{\infty} \gamma^k r_{t+k}$$

$$= \bar{r}(a,s) + \gamma \sum_{s_{t+1}} P(s_{t+1} | S_t = s, A_t = a) \sum_{a_{t+1}} P(a_{t+1} | s_{t+1}) \sum_{r_{t+1}, s_{t+2}, a_{t+2} \cdots, s_{\infty}} P(r_{t+1}, s_{t+2}, \cdots, s_{\infty} | s_{t+1}, a_{t+1}) \sum_{k=0}^{\infty} \gamma^k r_{(t+1)+k}$$

Therefore:

$$q^{\pi}(s,a) = \bar{r}(s,a) + \gamma \sum_{s'} T(a,s,s') \sum_{a'} \pi(a'|s')\, q^{\pi}(s',a')$$