ECE 590D-001, Reinforcement Learning at Scale

Jay Hineman, Ph.D.

Geometric Data Analytics

2020

Sequential decision (intuition)

- MDP: *Markov decision process*
- *MDPs are a classical formalization of sequential decision making, where actions influence not just immediate rewards, but also subsequent situations, or states, and through those future rewards.*

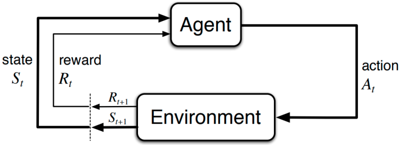

Figure: Agent-environment loop diagramming a Markov decision process.

Recycling robot *Example 3.3* [?]

- States: {low, high}
- Actions: {wait, search, recharge}
- Transitions: ¡¡board!!

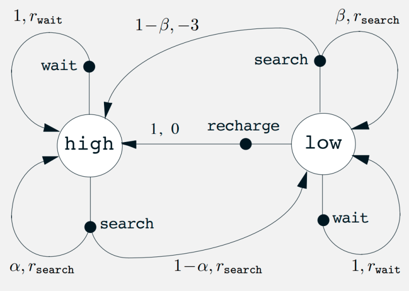

Figure: Diagram of MDP for [?] recycling robot example (page 52, Example 3.3).

Pole Balancing (cart-pole) *Example 3.4* [?]

- States: $\phi \in [0, \pi]$
- Actions: accelerate cart left or right
- Transitions: an ODE

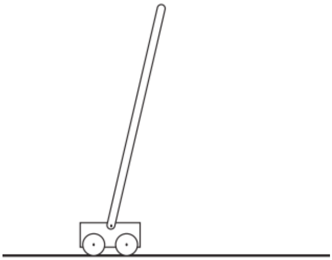

Figure: A *classic* problem in control: cart-pole/pole balancer/broom balancer.

Gridworld *Example 3.5* [?]

- States: $\{(x, y) : x = 0, 1, \ldots, m,\ \ y = 0, 1, , \ldots n\}$
- Actions: $\{\mathrm{N}, \mathrm{E}, \mathrm{S}, \mathrm{W}\}$
- Transitions: ¡¡board!!

3.3	8.8	4.4	5.3	1.5
1.5	3.0	2.3	1.9	0.5
0.1	0.7	0.7	0.4	-0.4
-1.0	-0.4	-0.4	-0.6	-1.2
-1.9	-1.3	-1.2	-1.4	-2.0

Figure: Actions representation and State-Value function for uniform random policy.

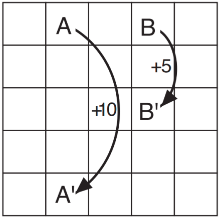

Figure: 5-by-5 gridworld with 2 distinguished states A and B.

Mathematical details [?]

Following Sutton and Barto [?]

- $t =$ time/iteration, $s, S_t \in \mathcal{S} =$ states, $a, A_t \in \mathcal{A}(s) =$ actions at state, $r, R_{t+1} \in \mathbb{R} =$ rewards, $S_0, A_0, R_1, S_1, A_1, R_2, S_2, A_2, R_3, \ldots =$ trajectory
- transition function $= p(s', r'|s, a) := \mathrm{Prob}\{S_t = s', R_t = r|S_{t-1} = s, A_{t-1} = a\}$
- **probability fact:** $\sum_{s',r} p(s', r|s, a) = 1$ for all $s \in \mathcal{S}, a \in \mathcal{A}(s)$
- **return and discounted return:**
  $G_t = R_{t+1} + R_{t+2} + R_{t+3} + \cdots + R_T$ and
  $G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots$
- **((board work))** from [?]

Policy, value, and action value functions

- **value and action-value functions:** functions of states (or of state–action pairs) that estimate how good it is for the agent to be in a given state (or how good it is to perform a given action in a given state).
- **policy:** a policy is a mapping from states to probabilities of selecting each possible action.
- **((board work))** from [**?**]

Mathematical specification of value and action-value

$$
\begin{aligned}
q_\pi(s) &= \mathbb{E}_\pi[G_t | S_t = s, A_t = a] \\
&= \mathbb{E}_\pi[\sum_{k=0} \gamma^k R_{(t+1)+k} | S_t = s, A_t = a], s \in \mathcal{S} \\
&= \text{Value of } s \text{ given the policy } \pi \\
v_\pi(s) &= \mathbb{E}_\pi[G_t | S_t = s] \\
&= \mathbb{E}_\pi[\sum_{k=0} \gamma^k R_{(t+1)+k} | S_t = s] \\
&= \text{action-value or } q\text{-value of } s, a \text{ given the policy } \pi
\end{aligned} \tag{1}
$$

- **((board work))** on *Bellman equation* for v_π from [**?**]

Spinning up's Taxonomy

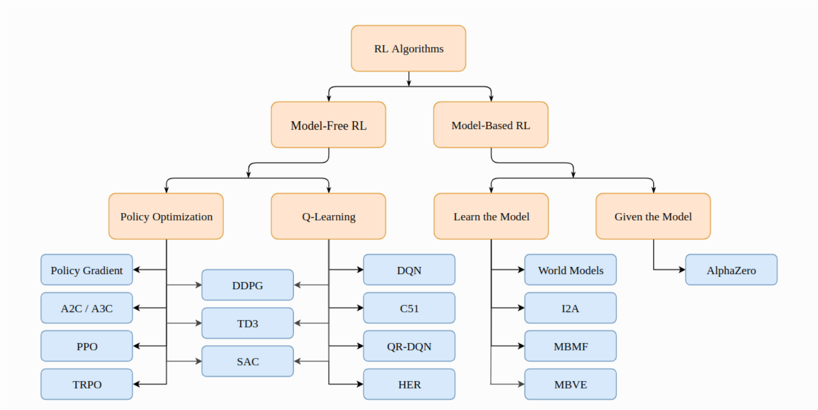

Figure: Non-exhaustive, but nice starting Taxonomy of (deep) RL methods. See also: citations.