CS 175: Project in Artificial Intelligence

Winter 2020

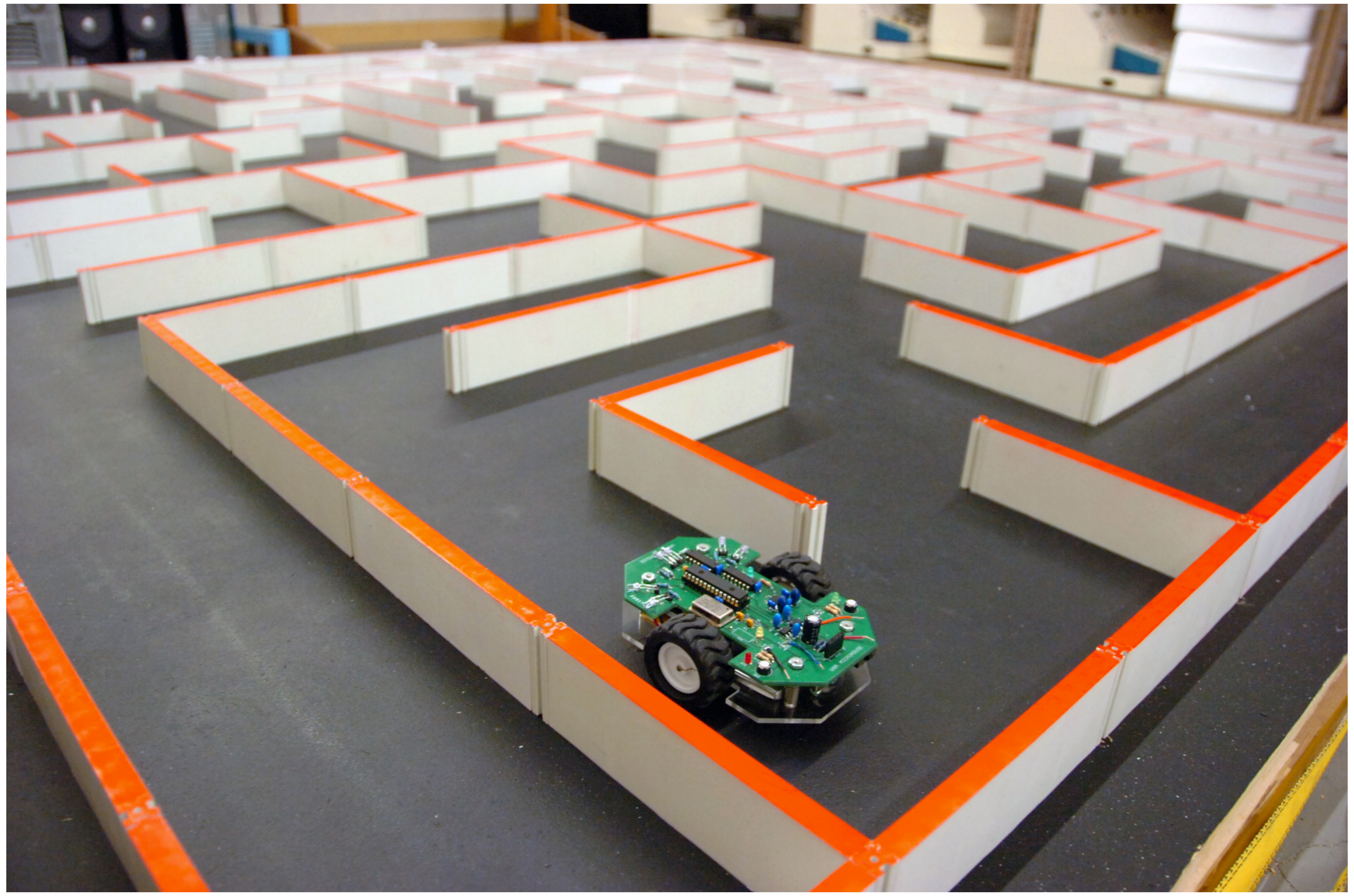
Lecture 3: Reinforcement Learning

Roy Fox
Department of Computer Science
Bren School of Information and Computer Sciences
University of California, Irvine

Today's lecture

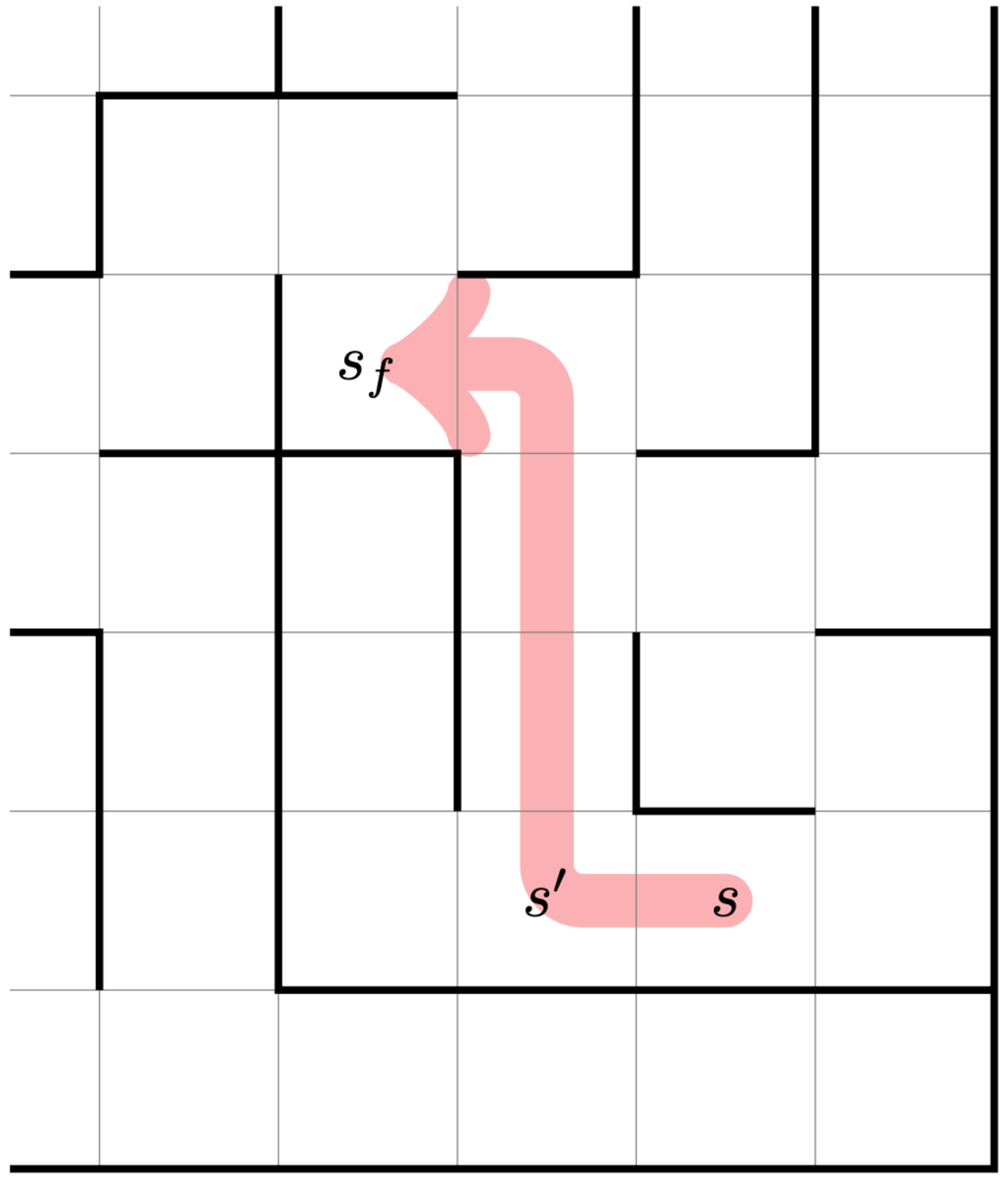
- Policy evaluation + improvement = RL
- Value Iteration, Generalized Policy Iteration
- Model-free RL
- Exploration

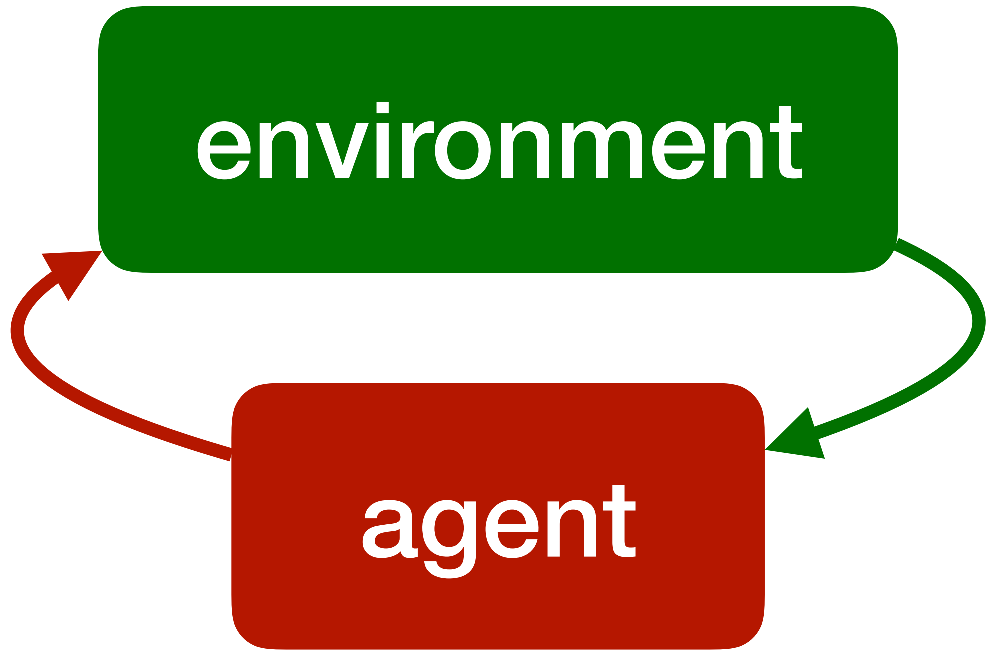
System state

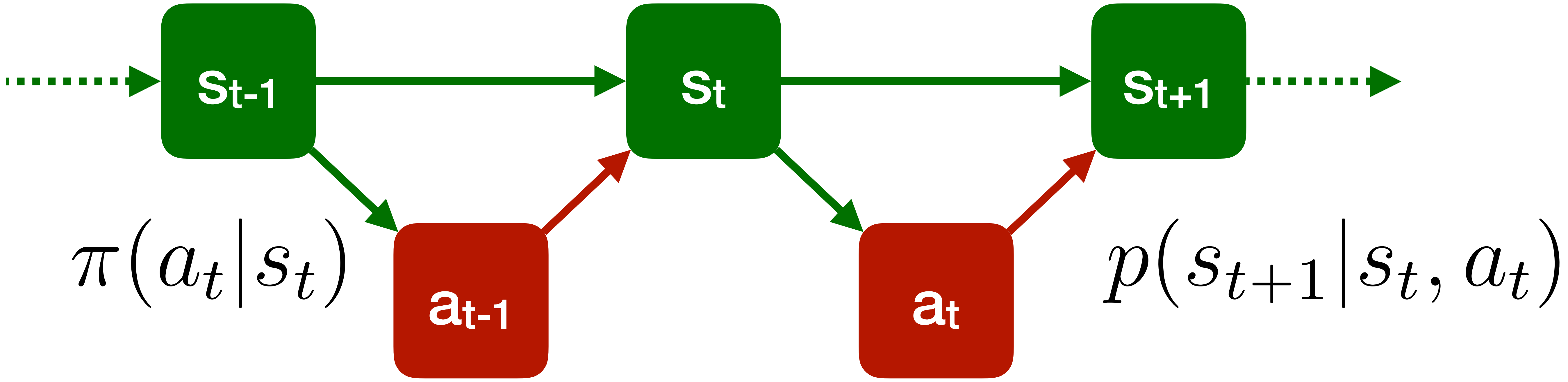


System = agent + environment

- Markov Decision Process (MDP)
  - State?
  - Action?
  - Reward?
  - Value?







Optimality principle

- **Proposition:** If ξ is a shortest path from s to s_f that goes through s', then a suffix of ξ is a shortest path from s' to s_f

- It follows that for all $s \neq s_f$

$$V(s) = \min_a \{1 + V(f(s,a))\}$$

- The optimal policy is

$$\pi(s) = \operatorname*{argmin}_a \{1 + V(f(s,a))\}$$

Algorithm 1 Bellman-Ford

$V(s_f) \leftarrow 0$
$V(s) \leftarrow \infty \qquad \forall s \in S \setminus \{s_f\}$
for ℓ from 1 to $|S| - 1$ **do**
    $V(s) \leftarrow \min_{a \in A}\{1 + V(f(s,a))\} \qquad \forall s \in S \setminus \{s_f\}$

Horizon classes

How to trade off short-term and long-term rewards?

- Finite: $$R = \sum_{t=0}^{T-1} r(s_t, a_t)$$
- Infinite: $$R = \lim_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T-1} r(s_t, a_t)$$
- Discounted: $$R = \sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \qquad 0 \leqslant \gamma < 1$$
- Episodic: $$R = \sum_{t=0}^{T-1} r(s_t, a_t) \qquad \text{s.t. } s_T = s_f$$

Learning setting is usually episodic, but return is usually discounted

Policy evaluation

- Distribution over trajectories:

$$p_\pi(\xi) = p(s_0) \prod_t \pi(a_t|s_t) p(s_{t+1}|s_t, a_t)$$

- Expected return: $\mathbb{E}_{\xi \sim p_\pi}[R]$

- State value function: $V_\pi(s) = \mathbb{E}_{\xi \sim p_\pi}[R|s_0 = s]$

- Recursively:

$$V_\pi(s) = \mathbb{E}_{a|s \sim \pi}[r(s,a) + \gamma\, \mathbb{E}_{\boxed{s'|s,a \sim p}}[V_\pi(s')]]$$

Model-free policy evaluation

- Monte Carlo (MC) evaluation:

$$\xi_i | s \sim p_\pi \qquad V(s) = \frac{1}{N} \sum_i R_i$$

- Temporal-Difference (TD) evaluation:

$$\text{for each } (\mathrm{s_i}, \mathrm{a_i}, \mathrm{r_i}, \mathrm{s_i'}) : \quad \Delta V(s_i) \leftarrow \alpha(r_i + \gamma V(s_i') - V(s_i))$$

  - Only works on-policy $a_i | s_i \sim \pi$

- Off-policy version:

$$Q_\pi(s, a) = \mathbb{E}_{\xi \sim p_\pi}[R | s_0 = s, a_0 = a]$$

$$\text{for each } (\mathrm{s_i}, \mathrm{a_i}, \mathrm{r_i}, \mathrm{s_i'}) : \quad \Delta Q(s_i, a_i) \leftarrow \alpha(r_i + \gamma\, \mathbb{E}_{a' | s_i' \sim \pi}[Q(s_i', a')] - Q(s_i, a_i))$$

Policy improvement

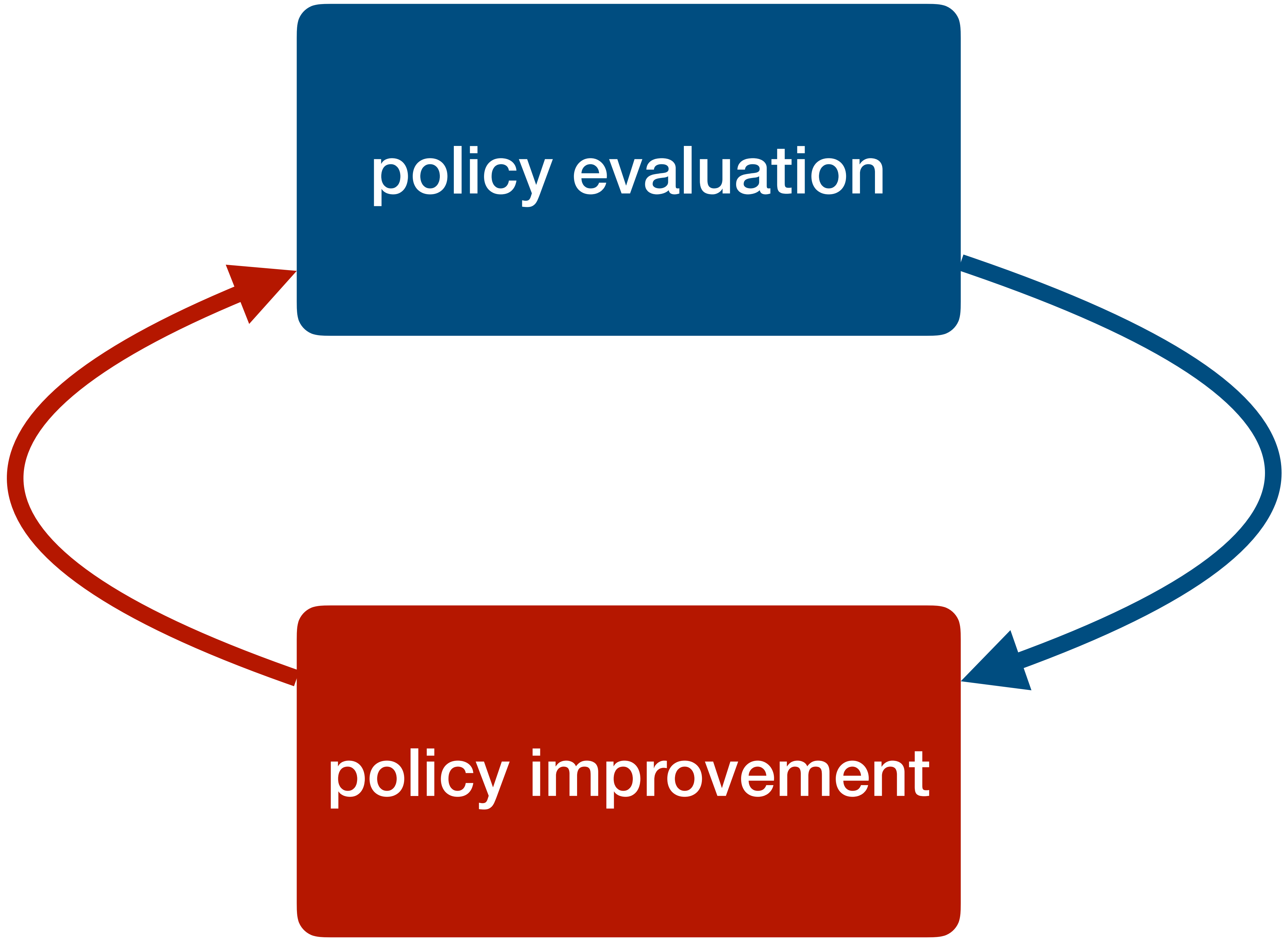
- A value function suggests the greedy policy:

$$\pi(s) = \operatorname*{argmax}_a Q(s,a) = \operatorname*{argmax}_a (r(s,a) + \gamma\, \mathbb{E}_{s'|s,a\sim p}[V(s')])$$

- Proposition: the greedy policy for Q_π is never worse than π
  - Generally: the greedy policy for $\max(Q_{\pi_1}, Q_{\pi_2})$ is never worse than π_1 or π_2
- Corollary 1: the optimal policy π^* is greedy for $Q^* = Q_{\pi^*}$
- Corollary 2: all fixed points of $\pi(s) = \operatorname*{argmax}_a Q_\pi(s,a)$ have $Q_\pi = Q^*$

Bellman optimality

The RL scheme



Value Iteration

- Repeat:

$$V(s_i) \leftarrow \max_a (r(s_i, a) + \gamma \, \mathbb{E}_{s'|s_i,a \sim p}[V(s')])$$

  - Must update each state repeatedly until convergence

Generalized Policy Iteration

- Alternate by some schedule:

$$V(s_i) \leftarrow \mathbb{E}_{a|s_i \sim \pi}[r(s_i, a) + \gamma\, \mathbb{E}_{s'|s_i,a \sim p}[V(s')]]$$

$$\pi(s_i) \leftarrow \operatorname*{argmax}_a (r(s_i, a) + \gamma\, \mathbb{E}_{s'|s_i,a \sim p}[V(s')])$$

Model-free reinforcement learning

- MC:

$$\xi_i | s, a \sim p_\pi \qquad Q(s,a) \leftarrow \frac{1}{N} \sum_i R_i$$

$$\pi \leftarrow \operatorname{argmax} Q$$

- Q-learning (TD):

$$\Delta Q(s_i, a_i) \leftarrow \alpha(r_i + \gamma \max_{a'} Q(s'_i, a') - Q(s_i, a_i))$$

Interaction policy

- In model-free RL, we often get data by interaction with the environment
  - How should we interact?
- On-policy methods (e.g. MC): must use current policy
- Off-policy methods: can use different policy — but not too different!
  - Otherwise may have train–test distribution mismatch (with Deep RL)
- In either case, must make sure interaction policy **explores** well enough

Exploration policies

- ε-greedy exploration: select uniform action w.p. ε, otherwise greedy
- Boltzmann exploration:

$$\pi(a|s) = \underset{a}{\mathrm{sm}}(Q(s,a);\beta) = \frac{\exp(\beta Q(s,a))}{\sum_{a'} \exp(\beta Q(s,a'))}$$

  - Becomes uniform as $\beta \to 0$, greedy as $\beta \to \infty$

Partial observability

- Need to infer something about the state from observations
- Optimal inference is Bayesian, maintain belief $b(s_t|\text{observable history})$
- Can define MDP over belief space
  - But it's very large!
- Many methods and tricks: PBVI, PSR, etc.
- This is one topic Deep RL makes conceptually much easier

Recap

- Bellman optimality = policy is greedy for its own value
- Can optimize by iterating policy evaluation ↔ policy improvement
- On-policy (e.g. MC) vs. off-policy (e.g. TD / Q-learning)
- Exploration should reach all states often enough