Markov Reward Processes

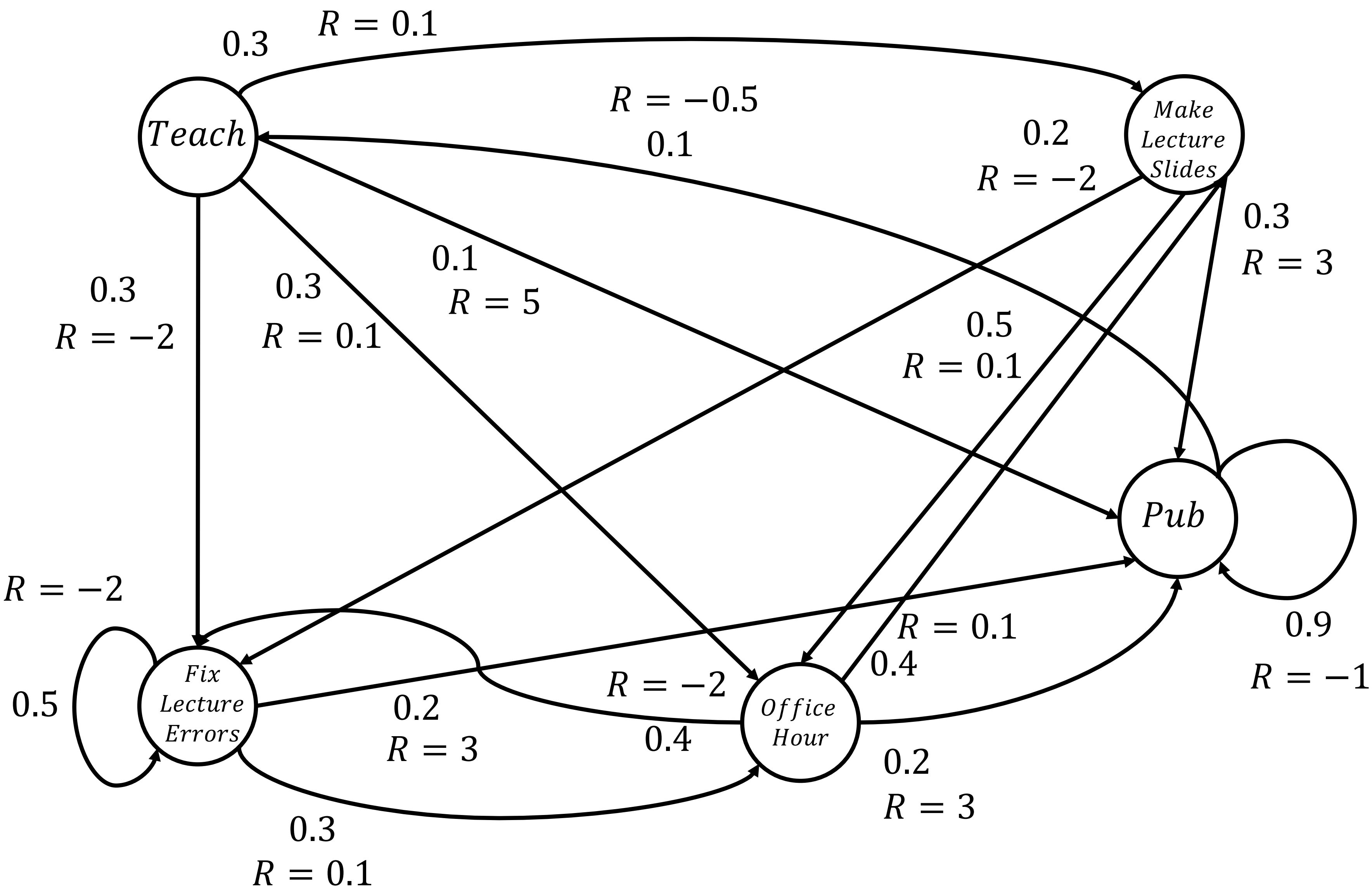
Reading

- Sutton, Richard S., and Barto, Andrew G. Reinforcement learning: An introduction. MIT press, 2018.
  - http://www.incompleteideas.net/book/the-book-2nd.html
  - Chapter 3
- Puterman, Martin L. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.
  - Chapters 2, 3, 4
- David Silver lecture on Markov Reward Processes
  - https://www.youtube.com/watch?v=lfHX2hHRMVQ
  - Overall good, but with a bias for MRPs with a terminal state
- MRP/MDP formalization
  - We’ll only talk about MRP in these slides

Overview

- Markov reward processes (MRPs) are an extension of Markov chains
  - You get a reward after each state transition
  - You can calculate your expected reward over time
- Markov decision processes (MDPs) are an extension of MRPs
  - Add actions to influence the transition probabilities
  - Model the control problem
- Both models lead to classical recursive equalities known as the Bellman equations

MRP for Workday Example



Questions

- What is the expected reward in $Teach$ after one step?

$$-2 * 0.3 + 0.1 * 0.3 + 0.1 * 0.3 + 5 * 0.1 = -0.04$$

- Ignoring the probabilities, which path maximizes the reward in the long run?
  - Trick question
  - Over a finite horizon, the path $Teach - Pub - Teach$ … brings the highest reward (4.5 every two hops)
  - Over an infinite horizon, any cycle with positive rewards will result in an infinite reward
    - E.g., $Make\ Lecture\ Slides - Office\ Hour - \cdots$

Probability Aside: Conditional Expectation

- Given two random variables X and Y, the conditional expectation of X given Y is defined as:

$$\mathbb{E}[X|Y=y] = \sum_{x \in \mathcal{X}} x \mathbb{P}[X=x|Y=y]$$

  - where $\mathcal{X}$ is the (discrete) set of all values X can take
- For a specific value of Y, what is the distribution of X
  - E.g., given that it is raining, what is the distribution of traffic
- Technically, the conditional expectation is a random variable
  - Takes on different values for different realizations of Y
- Similarly, for any function f:

$$\mathbb{E}[f(X)|Y=y] = \sum_{x \in \mathcal{X}} f(x) \mathbb{P}[X=x|Y=y]$$

MRP Formalization

Rensselaer

- An MRP is a 4-tuple (S, P, R, η) where
  - S is the set of states (aka the state space)
  - $P: S \times S \rightarrow \mathbb{R}$ is the probabilistic transition function
    - $\mathbb{P}[S_t | S_{t-1}] = P(S_{t-1}, S_t)$
  - $R: S \times S \rightarrow \mathbb{R}$ is the reward function
    - $R(S_{t-1}, S_t)$ is the reward received when following transition from S_{t-1} to S_t
    - Can also derive expected reward from s: $R_e(s) = \mathbb{E}[R_{t+1} | S_t = s]$
    - By convention, the reward associated with some transition is actually received on the next step
      - We use R_t to denote the reward we get at time t
    - The reward is typically determined by which state you land in
  - $\eta: S \rightarrow \mathbb{R}$ is the initial state distribution

A MRP Trace/Episode/Run/Trajectory

- Each MRP run is also called a trace/episode in different fields
  - Could be finite or infinite
- An example finite run:
  $S_0 = Teach, S_1 = Make\ Lecture\ Slides, S_2 = Fix\ Lecture\ Errors,$
  $S_3 = Office\ Hour$
- Corresponding rewards are:
  $$R_1 = 0.1, R_2 = -2, R_3 = 0.1$$
  - Total reward is -1.8
- In trace notation, the trajectory is:
  $$S_0, R_1, S_1, R_2, S_2, R_3, S_3$$
- What is the probability of this run:
  $$0.3 * 0.2 * 0.3 = 0.018$$

A MRP Trace/Episode/Run/Trajectory

- An example infinite run:

$$S_0 = Teach, S_1 = Pub, S_2 = Teach, S_3 = Pub, \ldots$$

- Corresponding rewards are:

$$R_1 = 5, R_2 = -0.5, R_3 = 5, \ldots$$

  – Total reward is infinite

- What is the probability of this trajectory?

0!

  – Multiplying infinitely many numbers less than 1

Goals and Rewards

- The reward is typically specified by the user to achieve a conceptual goal
  - E.g., avoid crashes, compute an optimal trajectory
- On the one hand, this works very well since the reward function can be arbitrarily specific and complex
- On the other, it is quite hard because sometimes the reward encourages unexpected behaviors
  - E.g., alternate between $Teach$ and Pub without making slides
  - E.g., go through walls in (imperfect physics) simulators

Finite vs Infinite Horizon

- An MRP can produce finite or infinite traces/episodes
  - Both settings are valid (also in the MDP case)
  - Note: book tries to combine them by assuming the system always has a sink goal state (not true for all MRPs/MDPs)
- In both cases, one can look at the total reward per trace
  - In the finite case (with T steps), total reward is:

$$R_1 + R_2 + \cdots + R_T$$

  - In the infinite case, the total reward is:

$$R_1 + R_2 + \cdots = \sum_{t=1}^{\infty} R_t$$

- What is a potential issue in the second case?
  - Total reward can be infinite

Discounted Reward

- Typically, we consider a **discounted** future reward:

$$\begin{aligned} G_t &= R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots \\ &= R_{t+1} + \gamma(R_{t+2} + \gamma R_{t+3} + \cdots) \\ &= R_{t+1} + \gamma G_{t+1} \end{aligned}$$

  - Discount factor $\gamma \in (0,1)$
- Why?
  - Future rewards less important than current ones
  - Mathematical convenience: don't want infinite rewards
- Note that sum is finite if R_t is bounded by some M for all t:

$$G_t \leq M \sum_{k=0}^{\infty} \gamma^k = \frac{M}{1-\gamma}$$

Value Function

- Intuitively, how *good* is your current state
- In the finite-horizon case, the value function is

$$\begin{aligned} v^t(s) &\coloneqq \mathbb{E}\left[R_{t+1} + \gamma R_{t+2} + \cdots + \gamma^{T-t+1} R_T \middle| S_t = s\right] \\ &= \mathbb{E}[G_t | S_t = s] \end{aligned}$$

- In the infinite-horizon case, it is

$$\begin{aligned} v^t(s) &\coloneqq \mathbb{E}[R_{t+1} + \gamma R_{t+2} + \cdots | S_t = s] \\ &= \mathbb{E}[G_t | S_t = s] \end{aligned}$$

- In both cases, it is the expected discounted reward
- Value function may be **time-dependent**
  - Book omits this important difference
    - Value functions are time-independent for MRPs/MDPs with a terminal state
    - Assuming terminal state doesn't depend on time

Value Function Example

Rensselaer

- Let $T = 2$

$$v^1(Teach) = \mathbb{E}[R_2 | S_1 = Teach]$$
$$= -2 * 0.3 + 0.1 * 0.3 + 0.1 * 0.3 + 5 * 0.1 = -0.04$$

- But

$$v^0(Teach) =$$
$$= \mathbb{E}[R_1 + \gamma R_2 | S_0 = Teach]$$

  - Note that $\mathbb{E}[R_1 | S_0 = Teach] = \mathbb{E}[R_2 | S_1 = Teach] = -0.04$
  - What about $\mathbb{E}[\gamma R_2 | S_0 = Teach]$?

$$\mathbb{E}[\gamma R_2 | S_0 = Teach] =$$
$$= \gamma \sum_r r \mathbb{P}[R_2 = r | S_0 = Teach]$$
$$= \gamma \sum_r r \sum_s \mathbb{P}[R_2 = r, S_1 = s | S_0 = Teach]$$

Value Function Example, cont'd

$$\mathbb{E}[\gamma R_2 | S_0 = Teach] =$$

$$= \gamma \sum_r r \sum_s \mathbb{P}[R_2 = r, S_1 = s | S_0 = Teach]$$

$$= \gamma \sum_r r \sum_s \mathbb{P}[R_2 = r | S_1 = s, S_0 = Teach]\, \mathbb{P}[S_1 = s | S_0 = Teach]$$

$$= \gamma \sum_r r \sum_s \mathbb{P}[R_2 = r | S_1 = s]\, \mathbb{P}[S_1 = s | S_0 = Teach]$$

$$= \gamma \sum_s \mathbb{P}[S_1 = s | S_0 = Teach] \sum_r r \mathbb{P}[R_2 = r | S_1 = s]$$

$$= \gamma \sum_s \mathbb{P}[S_1 = s | S_0 = Teach] \mathbb{E}[R_2 | S_1 = s]$$

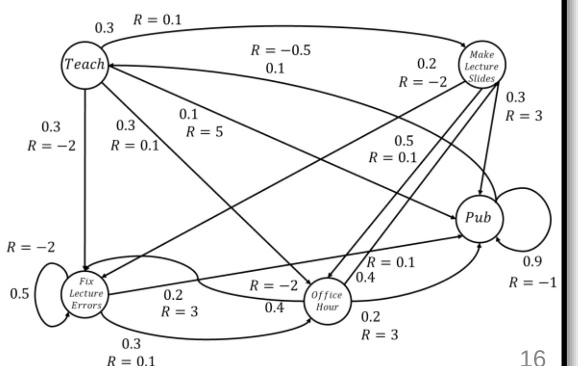
Value Function Example, cont'd

$$\mathbb{E}[\gamma R_2 | S_0 = Teach] = \gamma \sum_s \mathbb{P}[S_1 = s | S_0 = Teach] \mathbb{E}[R_2 | S_1 = s]$$

$$= \gamma \sum_s \mathbb{P}[S_1 = s | S_0 = Teach] v^1(s)$$

- We already know $v^1(Teach) = -0.04$
  - But this is not used since $\mathbb{P}[S_1 = Teach | S_0 = Teach] = 0$
    - $v^1(OH) = 3 * 0.2 + 0.1 * 0.4 - 2 * 0.4 = -0.16$
    - $v^1(Pub) = -1 * 0.9 - 0.5 * 0.1 = -0.95$
    - $v^1(MLS) = -2 * 0.2 + 0.1 * 0.5 + 3 * 0.3 = 0.55$
    - $v^1(FLE) = -2 * 0.5 + 3 * 0.2 + 0.1 * 0.3 = -0.37$
- So finally

$$\mathbb{E}[\gamma R_2 | S_0 = Teach] =$$
$$= \gamma(-0.16 * 0.3 - 0.95 * 0.1 + 0.55 * 0.3 - 0.37 * 0.3)$$



Value Function Example, cont'd

- Finally,

$$v^0(Teach) = \mathbb{E}[R_1 + \gamma R_2 | S_0 = Teach]$$
$$= -0.04 + \gamma(-0.089)$$

  - For $\gamma = 0.9, v^0(Teach) = -0.1201$

- So, for $T = 2, v^0(Teach) < v^1(Teach)$
- What about larger T?

Finite Horizon Bellman Equation

- We derived a recursive definition of v for the case $T = 2$:

$$v^0(s) = \mathbb{E}[R_1|S_0 = s] + \gamma \sum_{s'} \mathbb{P}[S_1 = s'|S_0 = s]v^1(s')$$
$$= \mathbb{E}\big[R_1 + \gamma v^1(S_1)\big|S_0 = s\big]$$

- This recursion applies for all t

$$v^t(s) = \mathbb{E}\big[R_{t+1} + \gamma R_{t+2} + \cdots + \gamma^{T-t+1} R_T\big|S_t = s\big]$$
$$= \mathbb{E}\big[R_{t+1} + \gamma(R_{t+2} + \cdots + \gamma^{T-t} R_T)\big|S_t = s\big]$$
$$= \mathbb{E}[R_{t+1} + \gamma G_{t+1}|S_t = s]$$

- Note that

$$\mathbb{E}[G_{t+1}|S_t = s] = \sum_g g\mathbb{P}[G_{t+1} = g|S_t = s]$$
$$= \sum_g g \sum_{s'} \mathbb{P}[G_{t+1} = g, S_{t+1} = s'|S_t = s]$$

  - Where g loops through all (finitely many) values of G_{t+1}

Finite Horizon Bellman Equation, cont'd

- This recursion applies for all t

$$\begin{aligned} v^t(s) &= \mathbb{E}\left[R_{t+1} + \gamma R_{t+2} + \cdots + \gamma^{T-2} R_T \middle| S_t = s\right] \\ &= \mathbb{E}\left[R_{t+1} + \gamma\left(R_{t+2} + \cdots + \gamma^{T-3} R_T\right) \middle| S_t = s\right] \\ &= \mathbb{E}[R_{t+1} + \gamma G_{t+1} | S_t = s] \end{aligned}$$

- Note that

$$\begin{aligned} \mathbb{E}[G_{t+1} | S_t = s] &= \sum_g g \sum_{s'} \mathbb{P}[G_{t+1} = g, S_{t+1} = s' | S_t = s] \\ &= \sum_g g \sum_{s'} \mathbb{P}[G_{t+1} = g | S_{t+1} = s', S_t = s] \mathbb{P}[S_{t+1} = s' | S_t = s] \\ &= \sum_{s'} \mathbb{P}[S_{t+1} = s' | S_t = s] \sum_g g \mathbb{P}[G_{t+1} = g | S_{t+1} = s'] \\ &= \sum_{s'} \mathbb{P}[S_{t+1} = s' | S_t = s] v^{t+1}(s') = \mathbb{E}\left[v^{t+1}(S_{t+1}) \middle| S_t = s\right] \end{aligned}$$

Finite Horizon Bellman Equation, cont'd

- This recursion applies for all t

$$v^t(s) = \mathbb{E}\left[R_{t+1} + \gamma R_{t+2} + \cdots + \gamma^{T-2} R_T \middle| S_t = s\right]$$
$$= \mathbb{E}\left[R_{t+1} + \gamma (R_{t+2} + \cdots + \gamma^{T-3} R_T) \middle| S_t = s\right]$$
$$= \mathbb{E}[R_{t+1} + \gamma G_{t+1} | S_t = s]$$

- Note that

$$\mathbb{E}[G_{t+1} | S_t = s] = \mathbb{E}\left[v^{t+1}(S_{t+1}) \middle| S_t = s\right]$$

- So, the (finite-horizon) Bellman equation is

$$v^t(s) = \mathbb{E}\left[R_{t+1} + \gamma v^{t+1}(S_{t+1}) \middle| S_t = s\right]$$

Infinite-Horizon MRPs

- Recall the definition of the value function

$$\begin{aligned} v^t(s) &:= \mathbb{E}[R_{t+1} + \gamma R_{t+2} + \cdots | S_t = s] \\ &= \mathbb{E}[G_t | S_t = s] \end{aligned}$$

- Sum (and expectation) is finite when R_t are bounded
- It turns out also that v does not depend on time, i.e.,

$$v^t(s) = v^{t+k}(s)$$

  – for any integer k

  – This is only true for stationary MDP/MRP

    - i.e., probabilities don't change over time

  – We will drop the superscript in the infinite-horizon case

Infinite-horizon Bellman Equation

- The Bellman equation in the infinite-horizon case is similar
$$v(s) = \mathbb{E}[R_{t+1} + \gamma v(S_{t+1}) | S_t = s]$$
  - The time t here is implicit
    - Only need it to distinguish the previous from the next state/reward
  - But the function v is the same
  - Proof is quite involved (proof in book is incomplete)
  - The discounted reward G_t no longer takes on finitely many values

Bellman Equation Matrix Form

- The Bellman equation in the infinite-horizon case is

$$v(s) = \mathbb{E}[R_{t+1} + \gamma v(S_{t+1}) | S_t = s]$$

- If we expand the expectation, we get:

$$v(s) = R_e(s) + \gamma \sum_{s'} \mathbb{P}[S_{t+1} = s' | S_t = s] v(s')$$

$$= R_e(s) + \gamma \sum_{s'} P(s, s') v(s')$$

- Let $\boldsymbol{s}$ be the vector of all states
  - E.g., $\boldsymbol{s} = [Teach, MLS, FLE, OH, Pub]$
- We can write the Bellman equation in matrix form

$$v(\boldsymbol{s}) = R_e(\boldsymbol{s}) + \gamma \boldsymbol{P} v(\boldsymbol{s})$$

Bellman Equation Matrix Form, cont'd

- We can write the Bellman equation in matrix form

$$v(\boldsymbol{s}) = R_e(\boldsymbol{s}) + \gamma \boldsymbol{P} v(\boldsymbol{s})$$

- How do we solve for $v(\boldsymbol{s})$?
  - Note that

$$(\boldsymbol{I} - \gamma \boldsymbol{P}) v(\boldsymbol{s}) = R_e(\boldsymbol{s})$$

  - i.e.,

$$v(\boldsymbol{s}) = (\boldsymbol{I} - \gamma \boldsymbol{P})^{-1} R_e(\boldsymbol{s})$$

  - Is $\boldsymbol{I} - \gamma \boldsymbol{P}$ always invertible?
    - Yes, because $\gamma \boldsymbol{P}$ has a maximum eigenvalue of $\gamma < 1$
    - If eigenvalues of $\boldsymbol{P}$ are λ_i, the eigenvalues of $\boldsymbol{I} - \gamma \boldsymbol{P}$ are $1 - \gamma \lambda_i$
    - For any eigenvector $\boldsymbol{v}_i$ of $\boldsymbol{P}$:

$$(\boldsymbol{I} - \gamma \boldsymbol{P}) \boldsymbol{v}_i = (1 - \gamma \lambda_i) \boldsymbol{v}_i$$

Workday Example, Infinite Horizon

- Recall that

$$\boldsymbol{P} = \begin{bmatrix} 0 & 0.3 & 0.3 & 0.3 & 0.1 \\ 0 & 0 & 0.4 & 0.4 & 0.2 \\ 0 & 0.5 & 0 & 0.2 & 0.3 \\ 0 & 0.3 & 0 & 0.5 & 0.2 \\ 0.1 & 0 & 0 & 0 & 0.9 \end{bmatrix}, R_e(\boldsymbol{s}) = \begin{bmatrix} -0.04 \\ -0.95 \\ 0.55 \\ -0.37 \\ -0.14 \end{bmatrix}$$

- For $\gamma = 0.9$,
$$(\boldsymbol{I} - \gamma\boldsymbol{P})^{-1} R_e(\boldsymbol{s}) = [-2.10 \quad -2.79 \quad -1.64 \quad -2.16 \quad -1.73]^T$$
- For $\gamma = 0.5$,
$$(\boldsymbol{I} - \gamma\boldsymbol{P})^{-1} R_e(\boldsymbol{s}) = [-0.31 \quad -1.10 \quad 0.16 \quad -0.75 \quad -0.28]^T$$
- Higher γ's generate lower state values. Why?
  - If you get stuck in Pub or FLE, self-transitions with negative rewards count for more

Finite vs Infinite Horizon

- Most of RL algorithms are built assuming infinite horizons
  – Theory is cleaner
  – Stronger claims (e.g., deterministic policies are sufficient)
- Most RL in practice is used in finite-horizon scenarios
  – Games, control tasks, protein folding
- What gives?
  – Practitioners are somewhat lucky
  – Either end time is conditioned on reaching a specific state
    - E.g., when we want to reach a goal or win a game
  – Or the same state is rarely visited at different times
    - E.g., when you are driving, you don’t usually go in circles

Finite vs Infinite Horizon, cont’d

- Whenever you have a finite horizon, you need to be careful
  - Is it possible to visit the same state multiple times?
    - If so, is the value different?
  - Is it possible to get stuck in some weird behavior
    - E.g., maybe we can’t reach the goal in time, so we just stay put in order to not crash
- We’ll discuss more when we get to MDPs