Computation Limits and The LP Formulation

Sham Kakade and Wen Sun

CS 6789: Foundations of Reinforcement Learning

Today:

- Next two lectures:
  - Today: computational complexity & the linear programming approach
  - Next lecture: statistical complexity (when the MDP is not known)

Today:

- Next two lectures:
  - Today: computational complexity & the linear programming approach
  - Next lecture: statistical complexity (when the MDP is not known)

- Today: Given an MDP $\mathscr{M} = (S, A, P, r, \gamma)$ can we exactly compute $Q^\star$ (or find $\pi^\star$) in polynomial time?

Today:

- Next two lectures:
  - Today: computational complexity & the linear programming approach
  - Next lecture: statistical complexity (when the MDP is not known)
- Today: Given an MDP $\mathscr{M} = (S, A, P, r, \gamma)$ can we exactly compute $Q^\star$ (or find $\pi^\star$) in polynomial time?
- But first, our recap:
  - value/policy iteration + contraction

Recap

Define Bellman Operator $\mathcal{T}$:

Given a function $f : S \times A \mapsto \mathbb{R}$,

$$\mathcal{T}f : S \times A \mapsto \mathbb{R},$$

$$\left(\mathcal{T}f\right)(s, a) := r(s, a) + \gamma \mathbb{E}_{s' \sim P(\cdot|s,a)} \max_{a' \in A} f(s', a'), \forall s, a \in S \times A$$

Value Iteration Algorithm:

1. Initialization: $Q^0 : \|Q^0\|_\infty \in (0, \frac{1}{1-\gamma})$

2. Iterate until convergence: $Q^{t+1} = \mathcal{T} Q^t$

Policy Iteration Algorithm:

Closed-form for PE
(see AJKS)

1. Initialization: $\pi^0 : S \mapsto \Delta(A)$

2. Policy Evaluation: $Q^{\pi^t}(s, a), \forall s, a$

3. Policy Improvement $\pi^{t+1}(s) = \arg\max_a Q^{\pi^t}(s, a), \forall s$

Final Quality of the Policy (for VI):

- $\pi^t : \pi^t(s) = \arg\max_a Q^t(s, a)$

  Theorem: $V^{\pi^t}(s) \geq V^\star(s) - \dfrac{2\gamma^t}{1-\gamma} \|Q^0 - Q^\star\|_\infty \forall s \in S$

Final Quality of the Policy (for VI):

- $\pi^t : \pi^t(s) = \arg\max_a Q^t(s,a)$

  Theorem: $V^{\pi^t}(s) \geq V^\star(s) - \frac{2\gamma^t}{1-\gamma} \|Q^0 - Q^\star\|_\infty \forall s \in S$

- Corollary: Set $Q^0 = 0$. After $t \geq \frac{\log \frac{2}{\epsilon(1-\gamma)^2}}{1-\gamma}$ iterations, we have:

  $V^{\pi^t}(s) \geq V^\star(s) - \epsilon \quad \forall s \in S$

Final Quality of the Policy (for VI):

- $\pi^t : \pi^t(s) = \arg\max_a Q^t(s, a)$

  Theorem: $V^{\pi^t}(s) \geq V^\star(s) - \frac{2\gamma^t}{1-\gamma} \| Q^0 - Q^\star \|_\infty \forall s \in S$

- Corollary: Set $Q^0 = 0$. After $t \geq \frac{\log \frac{2}{\epsilon(1-\gamma)^2}}{1-\gamma}$ iterations, we have:

  $V^{\pi^t}(s) \geq V^\star(s) - \epsilon \quad \forall s \in S$

- Same rate for PI.

Today

Polynomial Time & Strongly Polynomial Complexity

- Complexity to compute an **exact solution** given $\mathscr{M}$. (Aside: why?)
  - Assume basic arithmetic operations (+,-,x,÷) take unit time.

Polynomial Time & Strongly Polynomial Complexity

- Complexity to compute an **exact solution** given $\mathscr{M}$. (Aside: why?)
  - Assume basic arithmetic operations (+,-,x,÷) take unit time.
- Polytime computation: Suppose that (P, r, γ) in our MDP $\mathscr{M}$ is specified with rational entries, where $L(P, r, \gamma)$ is total bit-size required to specify (P, r, γ).
  - Can we (exactly) compute $Q^\star$ in time poly$\big(S, A, L(P, r, \gamma)\big)$?

Polynomial Time & Strongly Polynomial Complexity

- Complexity to compute an **exact solution** given $\mathcal{M}$. (Aside: why?)
  - Assume basic arithmetic operations (+,-,x,÷) take unit time.
- Polytime computation: Suppose that (P, r, γ) in our MDP $\mathcal{M}$ is specified with rational entries, where $L(P, r, \gamma)$ is total bit-size required to specify (P, r, γ).
  - Can we (exactly) compute $Q^\star$ in time poly$\big(S, A, L(P, r, \gamma)\big)$?
- (nearly) strongly polynomial time: Suppose (P, r) is specified with real numbers. Can we compute $Q^\star$ in poly$\big(S, A, \log(1/(1-\gamma))\big)$, with no dependence on $L(P, r, \gamma)$?

Polynomial Time & Strongly Polynomial Complexity

- Complexity to compute an **exact solution** given $\mathcal{M}$. (Aside: why?)
  - Assume basic arithmetic operations (+,-,x,÷) take unit time.
- Polytime computation: Suppose that (P, r, γ) in our MDP $\mathcal{M}$ is specified with rational entries, where $L(P, r, \gamma)$ is total bit-size required to specify (P, r, γ).
  - Can we (exactly) compute $Q^\star$ in time poly$\big(S, A, L(P, r, \gamma)\big)$?
- (nearly) strongly polynomial time: Suppose (P, r) is specified with real numbers. Can we compute $Q^\star$ in poly$\big(S, A, \log(1/(1-\gamma))\big)$, with no dependence on $L(P, r, \gamma)$?

- Scalings:
  - How does the complexity scale with the "horizon" $1/(1-\gamma)$? With $L(P, r, \gamma)$?

Computational Complexities of our Iterative Algorithms

Value Iteration

- When the sub-optimality gap between Q^t and $Q^\star$ is less than $2^{-L(P,r,\gamma)}$, than the greedy policy will be optimal.
  (by a standard argument in optimization)

Value Iteration

- When the sub-optimality gap between Q^t and $Q^\star$ is less than $2^{-L(P,r,\gamma)}$, than the greedy policy will be optimal.
  (by a standard argument in optimization)
- VI:
  - $O(\ \log\big(1/(\epsilon(1-\gamma))\big)/(1-\gamma)\ )$ iterations suffice for an ϵ-accurate solution.
  - Per iteration complexity: S^2A

Value Iteration

- When the sub-optimality gap between Q^t and $Q^\star$ is less than $2^{-L(P,r,\gamma)}$, than the greedy policy will be optimal.
  (by a standard argument in optimization)
- VI:
  - $O(\ \log\left(1/(\epsilon(1-\gamma)\right)/(1-\gamma)\ )$ iterations suffice for an ϵ-accurate solution.
  - Per iteration complexity: S^2A
- Poly runtime? For fixed γ, YES
  - VI is poly with $O\left(S^2A \cdot \dfrac{L(P,r,\gamma)\ \log(1/(1-\gamma))}{1-\gamma}\right)$ complexity.

Value Iteration

- When the sub-optimality gap between Q^t and $Q^\star$ is less than $2^{-L(P,r,\gamma)}$, than the greedy policy will be optimal.
  (by a standard argument in optimization)
- VI:
  - $O\left(\log\left(1/(\epsilon(1-\gamma)\right)/(1-\gamma) \right)$ iterations suffice for an ϵ-accurate solution.
  - Per iteration complexity: S^2A
- Poly runtime? For fixed γ, YES
  - VI is poly with $O\left(S^2A \cdot \dfrac{L(P,r,\gamma)\log(1/(1-\gamma))}{1-\gamma}\right)$ complexity.
- Strongly poly? NO
  (There are counterexamples)

Policy Iteration

- PI Per iteration complexity: $S^3 + S^2 A$ (why?)
  - PI is more costly than VI per iteration.
  - PI is observed to be much faster than VI to obtain an exact opt policy.

Policy Iteration

- PI Per iteration complexity: $S^3 + S^2 A$ (why?)
    - PI is more costly than VI per iteration.
    - PI is observed to be much faster than VI to obtain an exact opt policy.
- Poly runtime?
    - For fixed γ, YES
    - PI is $O\left((S^3 + S^2 A)\frac{L(P, r, \gamma)\log(1/(1-\gamma))}{1-\gamma}\right)$

Policy Iteration

- PI Per iteration complexity: $S^3 + S^2 A$ (why?)
  - PI is more costly than VI per iteration.
  - PI is observed to be much faster than VI to obtain an exact opt policy.
- Poly runtime?
  - For fixed γ, YES
  - PI is $O\left((S^3 + S^2 A) \frac{L(P, r, \gamma) \log(1/(1-\gamma))}{1-\gamma} \right)$
- Strongly poly?
  - Does PI compute an optimal policy in time independent of $L(P, r, \gamma)$? (ignoring other dependencies?)

Is PI a strongly poly algo?

- Does PI compute an optimal policy in time independent of $L(P, r, \gamma)$?
  - Yes: after A^S iterations

    Why? There are at most A^S policies, and PI is monotonic.
  - Refinement: [Mansour & Singh '99] PI halts after A^S/S iterations.

Is PI a strongly poly algo?

- Does PI compute an optimal policy in time independent of $L(P, r, \gamma)$?
  - Yes: after A^S iterations

    Why? There are at most A^S policies, and PI is monotonic.
  - Refinement: [Mansour & Singh '99] PI halts after A^S/S iterations.

- Is PI strongly polynomial?

  For fixed γ, yes:

  [Ye '12] PI halts after $\dfrac{S^2 A \log(S^2/(1-\gamma))}{1-\gamma}$ iterations.

Summary Table

	Value Iteration	Policy Iteration	LP-based Algorithms
Poly.	$S^2 A \frac{L(P,r,\gamma)\ \log \frac{1}{1-\gamma}}{1-\gamma}$	$(S^3 + S^2 A) \frac{L(P,r,\gamma) \log \frac{1}{1-\gamma}}{1-\gamma}$	?
Strongly Poly.	X	$(S^3 + S^2 A) \cdot \min \left\{ \frac{A^S}{S}, \frac{S^2 A \log \frac{S^2}{1-\gamma}}{1-\gamma} \right\}$	?

- VI Per iteration complexity: $S^2 A$
- PI Per iteration complexity: $S^3 + S^2 A$

Are VI and PI Polynomial Time algorithms?
(technically, no)

Is there a poly (and strongly poly) time algo for an MDP?
YES! Linear Programming

The Primal Linear Program

The Primal Linear Program

- We can write the Bellman equations with values rather than Q-values:

$$V(s) = \max_a \left\{ r(s, a) + \gamma \mathbb{E}_{s' \sim P(\cdot|s,a)} \left[V(s) \right] \right\}$$

The Primal Linear Program

- We can write the Bellman equations with values rather than Q-values:

$$V(s) = \max_a \left\{ r(s,a) + \gamma \mathbb{E}_{s' \sim P(\cdot|s,a)} \left[V(s) \right] \right\}$$

- An equivalent way to write the Bellman equations is as a linear program.

The Primal Linear Program

- We can write the Bellman equations with values rather than Q-values:
$$V(s) = \max_a \left\{ r(s,a) + \gamma \mathbb{E}_{s' \sim P(\cdot|s,a)} \left[V(s) \right] \right\}$$

- An equivalent way to write the Bellman equations is as a linear program.

- With variables $V \in \mathbb{R}^S$, the LP is:
$$\min V(s_0)$$
$$\text{s.t.} \quad V(s) \geq r(s,a) + \gamma \mathbb{E}_{s' \sim P(\cdot|s,a)} V(s') \quad \forall s, a \in S \times A$$

LP Runtimes and Comments

- An LP solver gives us a poly time algorithm.

LP Runtimes and Comments

- An LP solver gives us a poly time algorithm.
- [Ye, ’05]: there is an interior point algorithm (CIPA) which is (“nearly”) strongly polynomial.

LP Runtimes and Comments

- An LP solver gives us a poly time algorithm.
- [Ye, '05]: there is an interior point algorithm (CIPA) which is ("nearly") strongly polynomial.

- Comments:
  - VI is best thought of as a fixed point algorithm
  - PI is equivalent to a (block) simplex algorithm
    (Recall the simplex algo, in general, could be exp time.
    But not for MDPS, at least for fixed γ.)

Summary Table

	Value Iteration	Policy Iteration	LP-based Algorithms
Poly.	$S^2 A \frac{L(P,r,\gamma)\,\log\frac{1}{1-\gamma}}{1-\gamma}$	$(S^3 + S^2 A)\frac{L(P,r,\gamma)\,\log\frac{1}{1-\gamma}}{1-\gamma}$	$S^3 A\, L(P,r,\gamma)$
Strongly Poly.	X	$(S^3 + S^2 A)\cdot\min\left\{\frac{A^S}{S}, \frac{S^2 A \log\frac{S^2}{1-\gamma}}{1-\gamma}\right\}$	$S^4 A^4 \log\frac{S}{1-\gamma}$

- VI Per iteration complexity: $S^2 A$
- PI Per iteration complexity: $S^3 + S^2 A$
- The LP approach is only logarithmic in $1-\gamma$

- The linear programming is helpful in understanding the problem. (even though it is not used often)

What about the Dual LP?

What about the Dual LP?

- Let us now consider the dual LP.
  - It is also very helpful conceptually.
  - In some cases, it also provides a reasonable algorithmic approach

What about the Dual LP?

- Let us now consider the dual LP.
  - It is also very helpful conceptually.
  - In some cases, it also provides a reasonable algorithmic approach

- Let us start by understanding the dual variables
  and the “state-action polytope”

State-Action Visitation Measures

- For a fixed (possibly stochastic) policy π, define the state-action visitation distribution $d_{s_0}^{\pi}$ as:

$$d_{s_0}^{\pi}(s, a) = (1 - \gamma) \sum_{t=0}^{\infty} \gamma^t \mathrm{Pr}^{\pi}(s_t = s, a_t = a \mid s_0)$$

where $\mathrm{Pr}^{\pi}(s_t = s, a_t = a \mid s_0)$ is the state-action visitation probability when we execute π starting at state s_0.

State-Action Visitation Measures

- For a fixed (possibly stochastic) policy π, define the state-action visitation distribution $d^{\pi}_{s_0}$ as:

$$d^{\pi}_{s_0}(s, a) = (1 - \gamma) \sum_{t=0}^{\infty} \gamma^t \mathrm{Pr}^{\pi}(s_t = s, a_t = a \,|\, s_0)$$

where $\mathrm{Pr}^{\pi}(s_t = s, a_t = a \,|\, s_0)$ is the state-action visitation probability when we execute π starting at state s_0.

- We can verify that have $d^{\pi}_{s_0}$ satisfies, for all states $s \in S$:

$$\sum_{a} d^{\pi}_{s_0}(s, a) = (1 - \gamma) I(s = s_0) + \gamma \sum_{s', a'} P(s \,|\, s', a') d^{\pi}_{s_0}(s', a')$$

The “State-Action” Polytope

- Let us define the state-action polytope K as follows:

$$K_{s_0} := \left\{ d \,|\, d \geq 0 \ \text{ and } \right.$$

$$\left. \sum_a d(s,a) = (1-\gamma)I(s = s_0) + \gamma \sum_{s',a'} P(s \,|\, s',a')d(s',a') \right\}$$

The "State-Action" Polytope

- Let us define the state-action polytope K as follows:

$$K_{s_0} := \left\{ d \,|\, d \geq 0 \;\; \text{and} \;\; \sum_a d(s,a) = (1-\gamma) I(s = s_0) + \gamma \sum_{s',a'} P(s \,|\, s', a') d(s', a') \right\}$$

- This set precisely characterizes all state-action visitation distributions:

The "State-Action" Polytope

- Let us define the state-action polytope K as follows:

$$K_{s_0} := \left\{ d \,|\, d \geq 0 \ \ \text{and} \right.$$

$$\left. \sum_a d(s,a) = (1-\gamma)I(s=s_0) + \gamma \sum_{s',a'} P(s\,|\,s',a')d(s',a') \right\}$$

- This set precisely characterizes all state-action visitation distributions:

Lemma: $d \in K$ if and only if there exists a (possibly randomized) policy π s.t. $d^{\pi}_{s_0} = d$

The Dual LP

$$\max \sum_{s,a} d(s,a) r(s,a)$$

$$\text{s.t.} \quad d \in K$$

- One can verify that this is the dual of the primal LP.
- Note that K is a polytope