Reinforcement Learning without Intractable Oracles?

Ankur Moitra (MIT)

April 7th, ONR Program Meeting
OUTLINE

Part I: Introduction

• Models and Problems
• Hardness and Beyond Worst-Case Analysis
• Our Results

Part II: Planning

Part III: Learning

• Approximate MDPs via Barycentric Spanners
OUTLINE

Part I: Introduction

• Models and Problems
• Hardness and Beyond Worst-Case Analysis
• Our Results

Part II: Planning

Part III: Learning

• Approximate MDPs via Barycentric Spanners
REINFORCEMENT LEARNING (RL)

**Goal:** Agent learns by interacting with the environment

Diagram: Agent interacts with the environment to learn through actions, states, and rewards.
REINFORCEMENT LEARNING (RL)

**Goal:** Agent learns by interacting with the environment

Standard model is a **Markov Decision Process**
• State Space $\mathcal{S}$, start at $s_0$
MARKOV DECISION PROCESSES

- State Space $\mathcal{S}$, start at $s_0$
- Action Space $\mathcal{A}$
MARKOV DECISION PROCESSES

- State Space $\mathcal{S}$, start at $s_0$
- Action Space $\mathcal{A}$
- Rewards $R_h(s, a)$
MARKOV DECISION PROCESSES

- State Space $\mathcal{S}$, start at $s_0$
- Action Space $\mathcal{A}$
- Rewards $R_h(s, a)$
- Transition Probabilities $\mathbb{T}_h(s' | s, a)$
MARKOV DECISION PROCESSES

- State Space $\mathcal{S}$, start at $s_0$
- Action Space $\mathcal{A}$
- Rewards $R_h(s, a)$
- Transition Probabilities $\mathbb{T}_h(s'|s, a)$
- Horizon $H$
MARKOV DECISION PROCESSES

- State Space $\mathcal{S}$, start at $s_0$
- Action Space $\mathcal{A}$
- Rewards $R_h(s, a)$
- Transition Probabilities $\mathbb{T}_h(s' | s, a)$
- Horizon $H$

Goal: Find a policy $\pi : \mathcal{S} \rightarrow \mathcal{A}$ that maximizes expected reward
MARKOV DECISION PROCESSES

Main problems:

(1) **Planning:** Given a full description of the MDP, compute an optimal policy
MARKOV DECISION PROCESSES

Main problems:

(1) **Planning**: Given a full description of the MDP, compute an optimal policy

e.g. value iteration, policy iteration, linear programming
MARKOV DECISION PROCESSES

Main problems:

(1) **Planning**: Given a full description of the MDP, **compute** an optimal policy
   e.g. value iteration, policy iteration, linear programming

(2) **Learning**: Given budget of iterations with the environment (e.g. simulator, episodic), **learn** an optimal policy
MARKOV DECISION PROCESSES

Main problems:

(1) **Planning**: Given a full description of the MDP, compute an optimal policy
   
   e.g. value iteration, policy iteration, linear programming

(2) **Learning**: Given budget of iterations with the environment (e.g. simulator, episodic), learn an optimal policy

   e.g. model based, q-learning, actor-critic, policy gradient
And yet, for many applications tabular MDPs are insufficient
And yet, for many applications tabular MDPs are *insufficient*

Too many states to write down or visit?
And yet, for many applications tabular MDPs are insufficient.

Too many states to write down or visit?

function approximation, block MDPs, etc
And yet, for many applications tabular MDPs are **insufficient**

Too many states to write down or visit?

**function approximation, block MDPs, etc**

Cannot directly observe the full state?
And yet, for many applications tabular MDPs are insufficient.

Too many states to write down or visit?

function approximation, block MDPs, etc

Cannot directly observe the full state?

Partially observable MDPs (POMDPs)
And yet, for many applications tabular MDPs are insufficient

Too many states to write down or visit?

function approximation, block MDPs, etc

Cannot directly observe the full state?

Partially observable MDPs (POMDPs)

There is a rich understanding of how to augment the model, and still be able to bound sample complexity
WHAT ABOUT COMPUTATIONAL COMPLEXITY?

Returning to our earlier picture

Modern RL is generally built on computationally intractable oracles
WHAT ABOUT COMPUTATIONAL COMPLEXITY?

Returning to our earlier picture

Modern RL is generally built on computationally intractable oracles
WHAT ABOUT COMPUTATIONAL COMPLEXITY?

Returning to our earlier picture

Modern RL is generally built on computationally intractable oracles

Are there computationally efficient algorithms with strong end-to-end provable guarantees?
MARKOV DECISION PROCESSES

- State Space $\mathcal{S}$, start at $s_0$
- Action Space $\mathcal{A}$
- Rewards $R_h(s, a)$
- Transition Probabilities $\mathbb{T}_h(s'|s, a)$
- Horizon $H$
PARTIALLY OBSERVABLE MDPS (POMDPS)

- State Space $S$, start at $s_0$
- Action Space $A$
- Rewards $R_h(s, a)$
- Transition Probabilities $T_h(s' | s, a)$
- Horizon $H$
- Observation Space $O$ and probabilities $O_h(o | s)$
OUTLINE

Part I: Introduction

• Models and Problems
• Hardness and Beyond Worst-Case Analysis
• Our Results

Part II: Planning

Part III: Learning

• Approximate MDPs via Barycentric Spanners
Part I: Introduction

- Models and Problems
- **Hardness and Beyond Worst-Case Analysis**
- Our Results

Part II: Planning

Part III: Learning

- Approximate MDPs via Barycentric Spanners
PLANNING IS HARD

Classic lower bound:

**Theorem [Papadimitriou, Tsitsiklis]:** Optimal planning in a POMDP is PSPACE hard
THE CURSE OF HISTORY

Can you succinctly represent an optimal policy?
THE CURSE OF HISTORY

Can you succinctly represent an optimal policy?

<table>
<thead>
<tr>
<th>MDPs</th>
<th>POMDPs</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Optimal action only depends on current state</strong></td>
<td></td>
</tr>
</tbody>
</table>

$$\pi : S \rightarrow A$$
THE CURSE OF HISTORY

Can you succinctly represent an optimal policy?

<table>
<thead>
<tr>
<th>MDPs</th>
<th>POMDPs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optimal action only</td>
<td>Optimal action depends on</td>
</tr>
<tr>
<td>depends on current state</td>
<td>action/observation history</td>
</tr>
</tbody>
</table>

\[
\pi : S \rightarrow A
\]

\[
\pi : A \times O \times \cdots \times O \rightarrow A
\]
## THE CURSE OF HISTORY

Can you succinctly represent an optimal policy?

<table>
<thead>
<tr>
<th>MDPs</th>
<th>POMDPs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optimal action only</td>
<td>Optimal action depends on action/observation history</td>
</tr>
<tr>
<td>depends on current state</td>
<td>Alternatively, it depends on the current belief</td>
</tr>
</tbody>
</table>

\[
\pi : S \rightarrow A
\]

\[
\pi : A \times O \times O \rightarrow A
\]
# THE CURSE OF HISTORY

Can you succinctly represent an optimal policy?

<table>
<thead>
<tr>
<th>MDPs</th>
<th>POMDPs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optimal action only depends on current state</td>
<td>Optimal action depends on action/observation history</td>
</tr>
<tr>
<td>$\pi : S \rightarrow A$</td>
<td>$\pi : A \times O \cdots \times O \rightarrow A$</td>
</tr>
</tbody>
</table>

Alternatively, it depends on the current belief

$\pi : \Delta^S \rightarrow A$

Natural approaches use exponential space $(|A||O|)^H$ or $C^{|S|}$
Even worse news:

**Theorem [Golowich, Moitra, Rohatgi]:** Unless the exponential time hierarchy collapses, there is no polynomial sized description of an approximately optimal policy.
PLANNING IS EVEN HARDER

Even worse news:

**Theorem [Golowich, Moitra, Rohatgi]:** Unless the exponential time hierarchy collapses, there is no polynomial sized description of an approximately optimal policy.

Why should real-world POMDPs have succinct descriptions of good policies?
BEYOND WORST-CASE ANALYSIS

The hard instances have a curious feature:

“The observations don’t tell you anything about the state”
BEYOND WORST-CASE ANALYSIS

The hard instances have a curious feature:

“The observations don’t tell you anything about the state”

But what if they are at least somewhat informative?

“The observations leak some information about the state”
BEYOND WORST-CASE ANALYSIS

The hard instances have a curious feature:

“The observations don’t tell you anything about the state”

But what if they are at least somewhat informative?

“The observations leak some information about the state”

Could this enable tractable planning/learning?
BEYOND WORST-CASE ANALYSIS

**Definition:** We say the POMDP is $\gamma$-observable if for all $h$ and all distributions $b, b'$ on states we have

$$||\mathcal{O}_h b - \mathcal{O}_h b'||_1 \geq \gamma ||b - b'||_1$$

i.e. well-separated distributions on states lead to well-separated distributions on observations
BEYOND WORST-CASE ANALYSIS

**Definition:** We say the POMDP is $\gamma$-observable if for all $\mathcal{H}$ and all distributions $b, b'$ on states we have

\[ \| Q^\mathcal{H} b - Q^\mathcal{H} b' \|_1 \geq \gamma \| b - b' \|_1 \]

i.e. well-separated distributions on states lead to well-separated distributions on observations

Introduced by [Even-Dar, Kakade, Mansour] for understanding stability of beliefs in HMMs under misspecification
BEYOND WORST-CASE ANALYSIS

**Definition:** We say the POMDP is $\gamma$-observable if for all $h$ and all distributions $b, b'$ on states we have

$$\|\mathcal{O}_h b - \mathcal{O}_h b'\|_1 \geq \gamma \|b - b'\|_1$$

i.e. well-separated distributions on states lead to well-separated distributions on observations

Introduced by [Even-Dar, Kakade, Mansour] for understanding stability of beliefs in HMMs under misspecification

**Key Point:** No assumption on transition dynamics like e.g. deterministic transitions or mixing (under every possible policy)
OUTLINE

Part I: Introduction

• Models and Problems
• Hardness and Beyond Worst-Case Analysis
• Our Results

Part II: Planning

Part III: Learning

• Approximate MDPs via Barycentric Spanners
OUTLINE

Part I: Introduction
  • Models and Problems
  • Hardness and Beyond Worst-Case Analysis
  • Our Results

Part II: Planning

Part III: Learning
  • Approximate MDPs via Barycentric Spanners
MAIN RESULTS (PLANNING)

There is a quasi-polynomial time algorithm for planning under observability:

**Theorem [Golowich, Moitra, Rohatgi]:** Given the description of a $\gamma$-observable POMDP there is an algorithm running in time

$$H(\|\mathcal{O}\|, |\mathcal{A}|)^C \log(|S|H/\epsilon)/\gamma^4$$

that outputs an $\epsilon$-suboptimal policy.
MAIN RESULTS (PLANNING)

There is a quasi-polynomial time algorithm for planning under observability:

**Theorem [Golowich, Moitra, Rohatgi]:** Given the description of a $\gamma$-observable POMDP there is an algorithm running in time

$$H(||O||A)C\log(|S|H/\epsilon)/\gamma^4$$

that outputs an $\epsilon$-suboptimal policy

**Key Idea:** The Bayes filter is exponentially stable

\[\checkmark\] compute posterior on states, given actions/observations
MAIN RESULTS (PLANNING), CONTINUED

Moreover these results are tight

**Theorem [Golowich, Moitra, Rohatgi]:** Under the Exponential Time Hypothesis, there is no algorithm running in time

\[
(|S| |A| H |O|)^{O(\log(|S| |A| H |O|/\epsilon)/\gamma)}
\]

for finding an \(\epsilon\)-suboptimal policy in a \(\gamma\)-observable POMDP
MAIN RESULTS (PLANNING), CONTINUED

Moreover these results are tight

**Theorem [Golowich, Moitra, Rohatgi]:** Under the Exponential Time Hypothesis, there is no algorithm running in time

\[
(|S| |A| H|O|)^{o(\log(|S| |A| H|O|/\epsilon)/\gamma)}
\]

for finding an \(\epsilon\)-suboptimal policy in a \(\gamma\)-observable POMDP

It’s hard even in the **lossy case**, where you observe the state with probability \(\gamma\) independently at each step
WHAT ABOUT LEARNING?

In observable POMDPs:

- **Agent**
- **Environment**
- **Reward**
- **Observation**
- **Action**

- **PSPACE**
- **NP**
- **coNP**
- **P**
WHAT ABOUT LEARNING?

In observable POMDPs:

- Agent
- Environment
- Action
- Reward
- Observation

Complexity classes:
- P
- NP
- coNP
- PSPACE

Quasi-polynomial time planning
WHAT ABOUT LEARNING?

In observable POMDPs:

Can we build a learning algorithm on top of this primitive?
SAMPLE EFFICIENT LEARNING?

Assumption 1: The POMDP is undercomplete, i.e. \( |S| \leq |O| \)
And moreover \( \sigma_{min}(O_h) \geq \alpha \) for all \( h \)
SAMPLE EFFICIENT LEARNING?

Assumption 1: The POMDP is undercomplete, i.e. $|S| \leq |O|$

And moreover $\sigma_{min}(\mathbb{O}_h) \geq \alpha$ for all $h$

Theorem [Jin, Kakade, Krishnamurthy, Liu]: Given access to an optimistic planning oracle, there is an algorithm that uses

$$\text{poly}(|S|, |A|, H, |O|, 1/\alpha)$$

samples and finds an $\varepsilon$-suboptimal policy under Assumption 1.
SAMPLE EFFICIENT LEARNING?

**Assumption 1:** The POMDP is undercomplete, i.e. \(|S| \leq |O|\)
And moreover \(\sigma_{\min}(\mathbb{O}_h) \geq \alpha\) for all \(h\)

**Theorem [Jin, Kakade, Krishnamurthy, Liu]:** Given access to an optimistic planning oracle, there is an algorithm that uses
\[\text{poly}(|S|, |A|, H, |O|, 1/\alpha)\]
samples and finds an \(\varepsilon\)-suboptimal policy under **Assumption 1**

i.e. given a constrained, non-convex set of POMDPs, find the maximum value achievable by any policy in the set
SAMPLE EFFICIENT LEARNING?

Assumption 1: The POMDP is undercomplete, i.e. $|\mathcal{S}| \leq |\mathcal{O}|$

And moreover $\sigma_{min}(\Omega_h) \geq \alpha$ for all $h$

Theorem [Jin, Kakade, Krishnamurthy, Liu]: Given access to an optimistic planning oracle, there is an algorithm that uses

$$\text{poly}(|\mathcal{S}|, |\mathcal{A}|, H, |\mathcal{O}|, 1/\alpha)$$

samples and finds an $\varepsilon$-suboptimal policy under Assumption 1

i.e. given a constrained, non-convex set of POMDPs, find the maximum value achievable by any policy in the set

But optimism is very hard!
MAIN RESULTS (LEARNING)

We show how to solve learning by using barycentric spanners to construct a policy cover. As a result:

**Theorem [Golowich, Moitra, Rohatgi]:** There is an algorithm with running time and sample complexity

$$(|\mathcal{O}| |\mathcal{A}|)C \log(H|\mathcal{S}| |\mathcal{O}|/\epsilon \gamma) / \gamma^4$$

that outputs an $\epsilon$-suboptimal policy in a $\gamma$-observable POMDP
MAIN RESULTS (LEARNING)

We show how to solve learning by using barycentric spanners to construct a policy cover. As a result:

**Theorem [Golowich, Moitra, Rohatgi]:** There is an algorithm with running time and sample complexity

\[
(|O| |A|)^C \log(H|S||O|/\epsilon \gamma)/\gamma^4
\]

that outputs an \( \epsilon \)-suboptimal policy in a \( \gamma \)-observable POMDP

These are the first end-to-end algorithmic guarantees for learning POMDPs, without oracles or strong assumptions about the model dynamics.
OUTLINE

Part I: Introduction

• Models and Problems
• Hardness and Beyond Worst-Case Analysis
• Our Results

Part II: Planning

Part III: Learning

• Approximate MDPs via Barycentric Spanners
OUTLINE

Part I: Introduction

• Models and Problems
• Hardness and Beyond Worst-Case Analysis
• Our Results

Part II: Planning

Part III: Learning

• Approximate MDPs via Barycentric Spanners
BELIEF CONTRACTION

Theorem: Fix any $\gamma$-observable POMDP and policy $\pi$. Then

$$E_\tau[||b_t - b'_t||_1] \leq (1 - \gamma^4)^t|S|$$

posterior, starting from arbitrary belief state

posterior, starting from uniform belief state

where $\tau$ is the trajectory from the POMDP by playing $\pi$
BELIEF CONTRACTION

Theorem: Fix any $\gamma$-observable POMDP and policy $\pi$. Then

$$\mathbb{E}_\tau[\|b_t - b'_t\|_1] \leq (1 - \gamma^4)^t |\mathcal{S}|$$

posterior, starting from arbitrary belief state

posterior, starting from uniform belief state

where $\tau$ is the trajectory from the POMDP by playing $\pi$

Parallels well-known stability results for Kalman filtering
BELLMAN UPDATES FOR POMDPS

Can find an optimal policy through:

\[ \text{Value}(x) = \mathbb{E} \left[ \max_{\text{actions } a} \text{Reward}(a) + \text{Value}(x') \right] \]

- current action/obs. sequence
- new action/obs. sequence
- latent state sampled from current belief
TRUNCATED BELLMAN UPDATES

Belief contraction allows us to truncate
Belief contraction allows us to truncate

$$\text{Value}(x) = \mathbb{E} \left[ \max_{\text{actions } a} \text{Reward}(a) + \text{Value}(x') \right]$$

latent state sampled from truncated belief, with uniform prior
TRUNCATED BELLMAN UPDATES

Belief contraction allows us to **truncate**

\[
\text{Value}(x) = \mathbb{E} \left[ \max \text{ Reward}(a) + \text{Value}(x') \right]
\]

length t window

latent state sampled from truncated belief, with uniform prior

We only need a quasi-polynomial number of belief states
OUTLINE

Part I: Introduction

• Models and Problems

• Hardness and Beyond Worst-Case Analysis

• Our Results

Part II: Planning

Part III: Learning

• Approximate MDPs via Barycentric Spanners
OUTLINE

Part I: Introduction

• Models and Problems
• Hardness and Beyond Worst-Case Analysis
• Our Results

Part II: Planning

Part III: Learning

• Approximate MDPs via Barycentric Spanners
**APPROMATION BY MDPS**

**Corollary:** Any $\gamma$-observable POMDP $P$ can be approximated by an MDP $M$ with a quasi-polynomial number of states.
APPROXIMATION BY MDPS

Corollary: Any $\gamma$-observable POMDP $P$ can be approximated by an MDP $M$ with a quasi-polynomial number of states

(1) $P$ can be thought of as an MDP on belief states
Corollary: Any \( \gamma \)-observable POMDP \( P \) can be approximated by an MDP \( M \) with a quasi-polynomial number of states

(1) \( P \) can be thought of as an MDP on belief states

(2) Construct \( M \) as follows:

\[
\text{states} = \text{length } L \text{ sequences of actions/observations}
\]
Corollary: Any $\gamma$-observable POMDP $P$ can be approximated by an MDP $M$ with a quasi-polynomial number of states

1. $P$ can be thought of as an MDP on belief states
2. Construct $M$ as follows:
   - states = length $L$ sequences of actions/observations
   - transitions = shift in/out the newest/oldest actions/obs.
**APPROXIMATION BY MDPS**

**Corollary:** Any $\gamma$-observable POMDP $P$ can be approximated by an MDP $M$ with a quasi-polynomial number of states.

1. $P$ can be thought of as an MDP on belief states.
2. Construct $M$ as follows:
   - **states** = length $L$ sequences of actions/observations
   - **transitions** = shift in/out the newest/oldest actions/obs.
3. States in $M$ can be mapped to beliefs (using a uniform prior).
   
   **By belief contraction, $M$ and $P$ approximate each other.**
APPROXIMATION BY MDPS

Corollary: Any $\gamma$-observable POMDP $P$ can be approximated by an MDP $M$ with a quasi-polynomial number of states

Can we learn $M$ efficiently?
**Corollary:** Any $\gamma$-observable POMDP $P$ can be approximated by an MDP $M$ with a quasi-polynomial number of states

Can we learn $M$ efficiently?

**Simplification:** For any latent state $x$ in $P$, and any timestep $h$, there is some policy $\pi$ that visits $x$ at $h$ with nonnegligible probability
APPROXIMATION BY MDPS

**Corollary:** Any $\gamma$-observable POMDP $P$ can be approximated by an MDP $M$ with a quasi-polynomial number of states.

Can we learn $M$ efficiently?

**Simplification:** For any latent state $x$ in $P$, and any timestep $h$, there is some policy $\pi$ that visits $x$ at $h$ with nonnegligible probability.

How can we find a mixture of policies that visits all latent states?
BARYCENTRIC SPANNERS

Definition: Given a set $\mathcal{X} \subseteq \mathbb{R}^d$, a $\lambda$-approximate barycentric spanner is a set $\mathcal{C} \subseteq \mathcal{X}$ of size $d$ such that every point in $\mathcal{X}$ can be expressed as a linear combination of points in $\mathcal{C}$ with coefficients in the range $[-\lambda, \lambda]$.
BARYCENTRIC SPANNERS

Definition: Given a set $\mathcal{X} \subseteq \mathbb{R}^d$, a $\lambda$-approximate barycentric spanner is a set $\mathcal{C} \subseteq \mathcal{X}$ of size $d$ such that every point in $\mathcal{X}$ can be expressed as a linear combination of points in $\mathcal{C}$ with coefficients in the range $[-\lambda, \lambda]$.

Theorem [Awerbuch, Kleinberg ‘04]: Given an oracle for optimizing linear functions over $\mathcal{X}$, there is a polynomial time algorithm for constructing a $\lambda$-approximate barycentric spanner with

$$O(d^2 \log \lambda d)$$

calls to the optimization oracle (assuming $\mathcal{X}$ is compact).
POLICY COVERS

Now let

\[ \mathcal{X} = \text{set of all distributions on observations at step h that can be obtained by a policy} \]
POLICY COVERS

Now let

\[ \mathcal{X} = \text{set of all distributions on observations at step } h \text{ that can be obtained by a policy} \]

**Claim:** By observability, if we can construct policies

\[ \pi_1, \pi_2, \ldots, \pi_{|O|} \]

whose induced distributions on observations at step \( h \) are an approximate barycentric spanner
POLICY COVERS

Now let

\[ \mathcal{X} = \text{set of all distributions on observations at step } h \text{ that can be obtained by a policy} \]

**Claim:** By observability, if we can construct policies

\[ \pi_1, \pi_2, \ldots, \pi_{|O|} \]

whose induced distributions on observations at step \( h \) are an approximate barycentric spanner, **we must visit each latent state with nonnegligible probability**
ITERATIVE EXPLORATION

Our approach is:

$M_h$

MDP that approximates $P$ up to step $h$
ITERATIVE EXPLORATION

Our approach is:

\[ M_h \xrightarrow{\text{Barycentric spanner}} X_h \]

- \( M_h \): MDP that approximates \( P \) up to step \( h \)
- \( X_h \): Barycentric spanner for observation distributions at step \( h \)
ITERATIVE EXPLORATION

Our approach is:

\[ \mathbb{M}_h \rightarrow \mathbb{X}_h \]

- \( \mathbb{M}_h \): MDP that approximates \( P \) up to step \( h \)
- \( \mathbb{X}_h \): Barycentric spanner for observation distributions at step \( h \)

Reaches all latent states

“explorability”
ITERATIVE EXPLORATION

Our approach is:

\[ \mathbf{M}_h \rightarrow \mathbf{X}_h \]

- \( \mathbf{M}_h \): MDP that approximates \( P \) up to step \( h \)
- \( \mathbf{X}_h \): Barycentric spanner for observation distributions at step \( h \)

"explorability"

- Estimate next layer of transitions
- Reaches all latent states
Our approach is:

$M_{h+1}$: MDP that approximates $P$ up to step $h+1$

$X_h$: Barycentric spanner for observation distributions at step $h$

Estimate next layer of transitions

Reaches all latent states

“explorability”
ITERATIVE EXPLORATION

Our approach is:

\[ M_{h+1} \rightarrow X_h \]

- MDP that approximates \( P \) up to step \( h+1 \)
- Barycentric spanner for observation distributions at step \( h \)

“explorability”

- Estimate next layer of transitions
- Reaches all latent states

Without explorability, need more complex measure of progress
LOOKING FORWARD

To get end-to-end algorithmic guarantees, we need to explore new assumptions and frameworks
LOOKING FORWARD

To get end-to-end algorithmic guarantees, we need to explore new assumptions and frameworks.

In [Golowich, Moitra], we took a learning-augmented algorithms approach:

“Can you improve Q-learning with advice?”
LOOKING FORWARD

To get end-to-end algorithmic guarantees, we need to explore new assumptions and frameworks.

In [Golowich, Moitra], we took a learning-augmented algorithms approach:

“Can you improve Q-learning with advice?”

**Takeaway:** Improved regret bounds, where you only need to explore state-action pairs with substantially inaccurate predictions, 
**even without knowing which ones are accurate in advance**
Summary:

- Modern RL is built on computationally intractable oracles. **Are there end-to-end guarantees?**
- Quasi-polynomial time algorithm for planning in **observable** POMDPs, no assumption on dynamics
- New framework for learning without optimism
Summary:

• Modern RL is built on computationally intractable oracles. **Are there end-to-end guarantees?**

• Quasi-polynomial time algorithm for planning in **observable** POMDPs, no assumption on dynamics

• New framework for learning without optimism

Thanks! Any Questions?