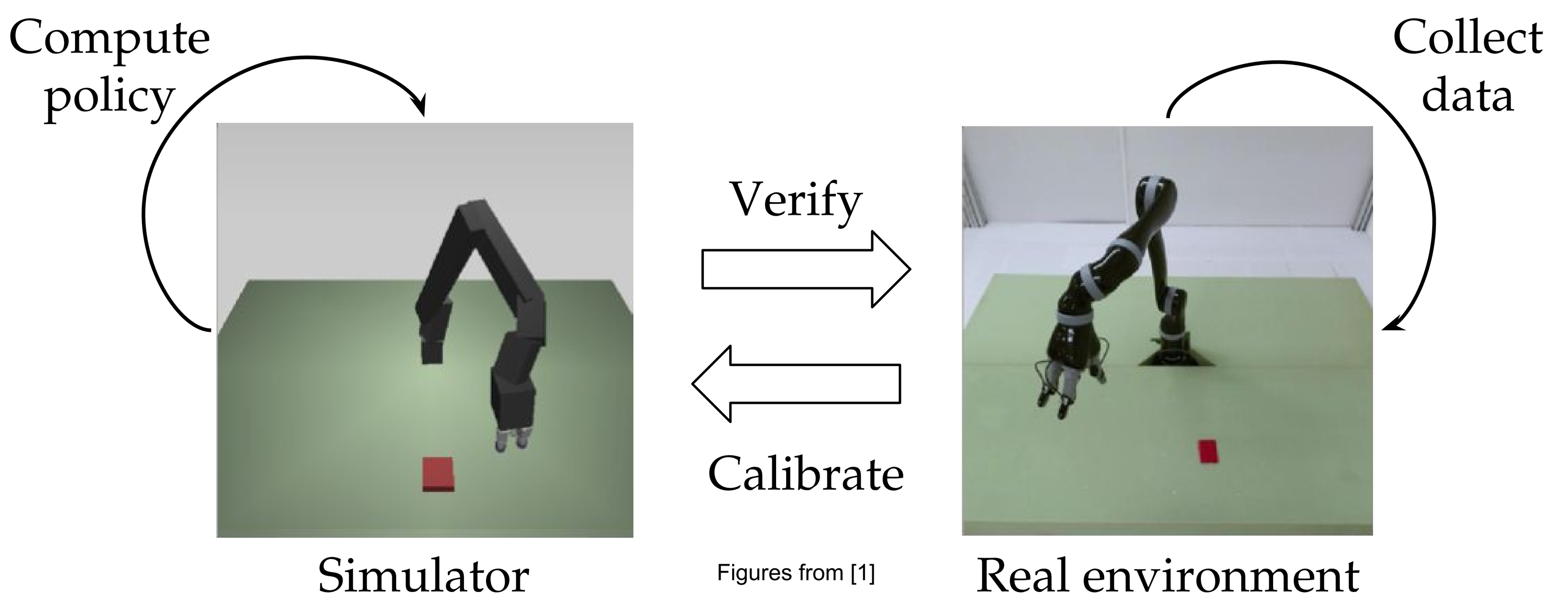
PAC Reinforcement Learning with an Imperfect Model

Nan Jiang
Microsoft Research, NYC

Microsoft Research

Motivation: sim2real transfer for RL

- Empirical success of deep RL (Atari games, MuJoCo, Go, etc.)
- Popular algorithms are **sample-intensive** for real-world applications
- Sim2real approach: (1) train in a **simulator**, (2) transfer to real world
- Hope: **reduce** sample complexity with a **high-fidelity** simulator



A simple theoretical question:

If the simulator is only wrong in a small number of state-action pairs, can we substantially reduce **#real** trajectories needed?

Answer: **No**! Further conditions are needed…

Deeper thoughts: many scenarios in sim2real transfer

- What to transfer: policy, features, skills, etc. (we focus on policy)
- How to quantify fidelity
  - Prior theories (e.g., [2]) focus on global error (worst over all states)
  - Local errors (#states with large errors)?
- Is interactive protocol really better than non-interactive? Answer: **Yes**! (non-interactive: collect real data, calibrate the model, done)

Setup

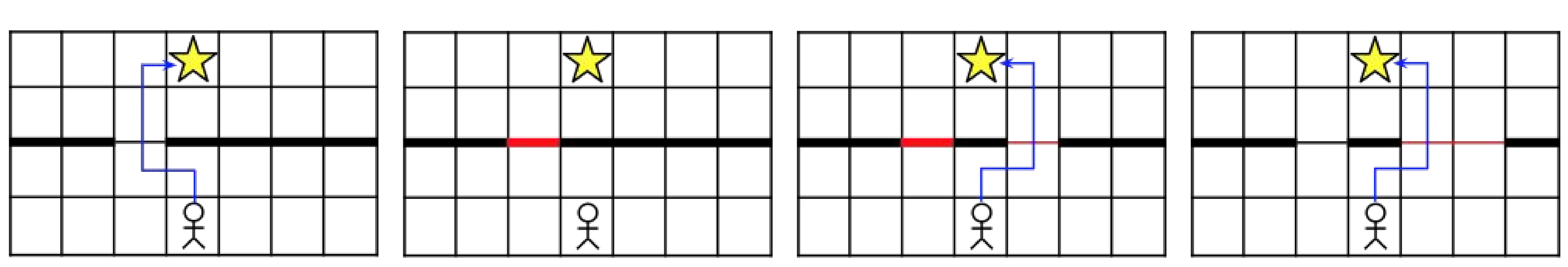
- Real environment: episodic MDP $M = (S, A, P, R, H, s_0)$.
- Simulator: $\widehat{M} = (S, A, \widehat{P}, R, H, s_0)$.
- Define $X_{\xi\text{-}inc}$ as the set of "wrong" (s, a) pairs where
$$\|P(s,a) - \widehat{P}(s,a)\|_{TV} > \xi.$$
- Goal: learn a policy π such that $V^*(s_0) - V^\pi(s_0) \le \varepsilon$, using only
$$poly(|X_{\xi\text{-}inc}|, H, 1/\varepsilon, 1/\delta) \text{ real trajectories.}$$

No dependence on $|S|$ or $|A|$; instead, **adapt** to the simulator's quality.

- This is impossible without further assumptions...

Lower bound and hard instances

- Lower bound: $\Omega(|S \times A|/\varepsilon^2)$, even when $|X_{0\text{-}inc}|$ = constant!
- Proof sketch:
  - Bandit hard instance: M = all arms Ber(½), except one w/ Ber(½+ ε)
  - Approximate model: $\widehat{M}$ = all arms Ber(½) --- $|X_{0\text{-}inc}|$=1 but useless
- Illustration:



Real environment — Model 1 (hard instance) — Model 2 (hard instance) — Model 3 (good cases)

  - Issue with Model 1: too pessimistic
  - Issue with Model 2: initially optimistic; pessimistic once error fixed
  - Good property of Model 3: always optimistic

Sufficient conditions and algorithms

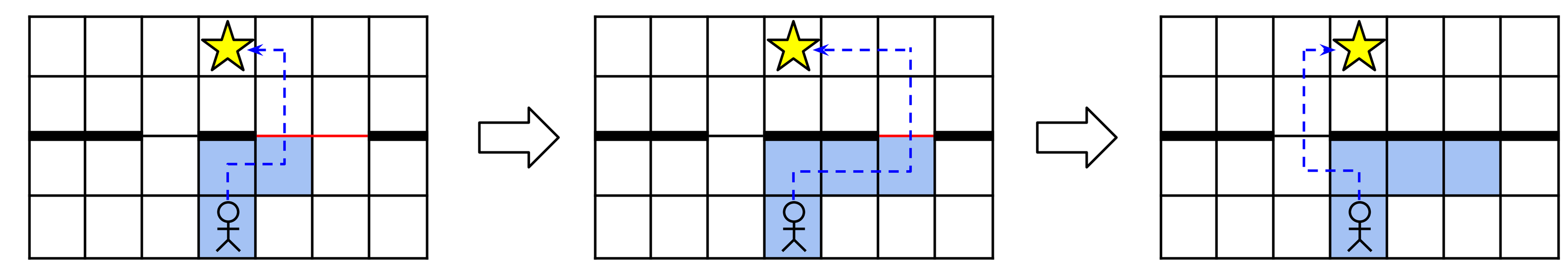
Definition 1: A partially corrected model $\widehat{M}_X$ is one whose dynamics are the same as M on X, and the same as $\widehat{M}$ otherwise.

Condition 1: $V^*(s_0)$ is always higher in $\widehat{M}_X$ than in M for all $X \subseteq X_{\xi\text{-}inc}$.

(see the agnostic version of the conditions in the paper.)

Theorem 1: Under Condition 1, there exists an algorithm that achieves $O(|X_{\xi\text{-}inc}|^2\, H^4 \log(1/\delta)/\varepsilon^3)$ sample complexity for $\xi = O(\varepsilon/H^2)$.

Algorithm 1: illustration on previous example, Model 3.



- Collect data using optimal policy in simulator.
- Blue cells: plug in estimated dynamics along states w/ enough samples.

Key steps in analysis:

- Accurate estimation of transition may require $O(|S|)$ samples per (s, a).
- Incur dependence on $|S|$… need to avoid.
- Workaround: union bound over V^* of all partially corrected models, which only incurs $\log(2^{|X_{\xi\text{-}inc}|})$.

What if we cannot change the model?

Basic idea:

- Identify the wrong states as necessary.
- Terminate a simulated episode when running into wrong (s, a).
  = penalize a wrong (s, a) by fixing $Q(s, a) = 0$ ($V_{\min}$) in planning.

Definition 2: A partially penalized model $M_{\setminus X}$ is one that terminates on X, and have the same dynamics as $\widehat{M}$ otherwise.

Condition 2: $V^*(s_0)$ is always higher in $\widehat{M}_{\setminus X}$ than in M for all $X \subseteq X_{\xi\text{-}inc}$.

Theorem 2: Under Condition 2, there exists an algorithm that achieves $O(|X_{\xi\text{-}inc}|^2\, H^2 \log(1/\delta)/\varepsilon^3)$ sample complexity for $\xi = O(\varepsilon/H)$.

Algorithm 2: $M_0 \leftarrow \widehat{M}$, $X_0 \leftarrow \{\}$.

For $t = 0, 1, 2, \ldots$

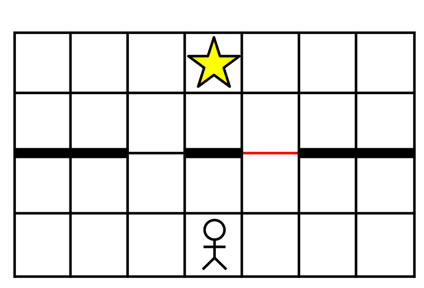
- Let π_t be the optimal policy of M_t. Monte-Carlo evaluate π_t.
- Return if $V^{\pi_t}(s_0)$ in M is close to $V^*(s_0)$ in M_t.
- Sample real trajectories using π_t.
- Once #samples from some (s, a) reaches threshold, compute
$$\left|\mathbb{E}_{s'\sim P(s,a)}[V^\star_{M_t}(s')] - \mathbb{E}_{s'\sim D_{s,a}}[V^\star_{M_t}(s')]\right|$$
- If large, $X_{t+1} \leftarrow X_t \cup \{(s, a)\}$, $M_{t+1} \leftarrow M_{\setminus X_{t+1}}$.

Non-interactive protocol is inefficient

Theorem 3: "Collect data, calibrate, done" style algorithms cannot have $poly(|X_{\xi\text{-}inc}|, H, 1/\varepsilon, 1/\delta)$ sample complexity, even with Conditions 1 & 2.

Proof sketch: assume such an algorithm exists. Then,

- The same dataset can calibrate multiple models.
- Consider the hard instance in bandit. Design $|A|^2$ models: $\forall\, a, a' \in A$,
$$\widehat{M}_{a,a'} = \text{all arms Ber(½), except } a \;\&\; a' \text{ w/ Ber(½+ } \varepsilon).$$



- When $a = a^*$, both Conditions 1 & 2 are met and $|X_{0\text{-}inc}| = 1$.
- Hypothetical algorithm prefers a^* to a' with ⅔ prob. using a dataset of constant size.

- Majority vote from $O(\log|A|)$ datasets: boost success prob. to 1 - $O(1/|A|)$.
- Solve bandit hard instance w/ polylog($|A|$), against $\Omega(|A|)$ lower bound.

[1] Rusu et al. Sim-to-real robot learning from pixels with progressive nets. CoRL 2017.
[2] Cutler et al. 2015. Real-world reinforcement learning via multifidelity simulators. IEEE Transaction on Robotics, 2015.