Representation Learning and Exploration in Reinforcement Learning

Akshay Krishnamurthy

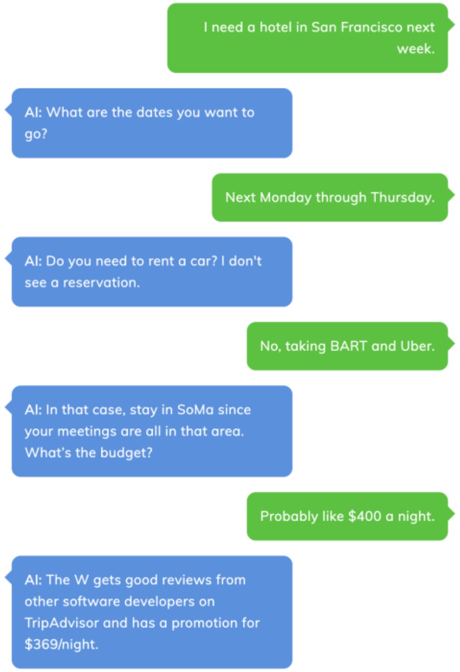
akshaykr@microsoft.com

Back in 2015

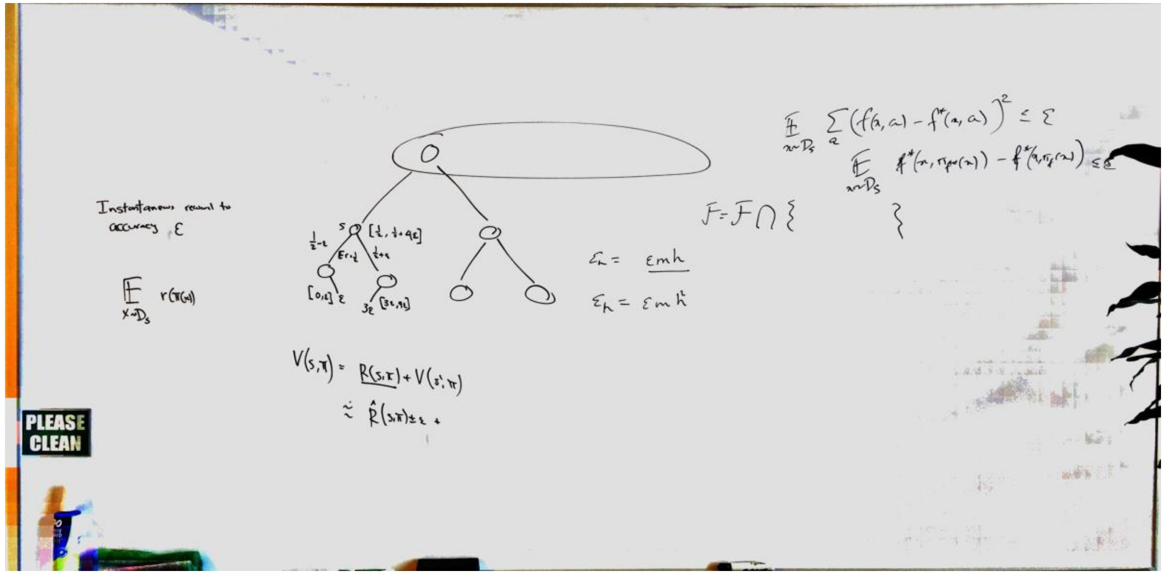
Goal: provably efficient sequential decision making methods that scale to complex domains

Robotics

Gaming

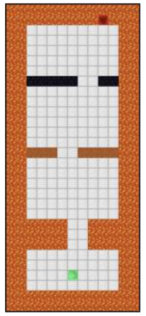


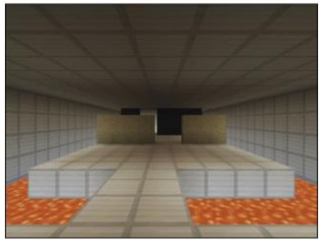
Dialogue

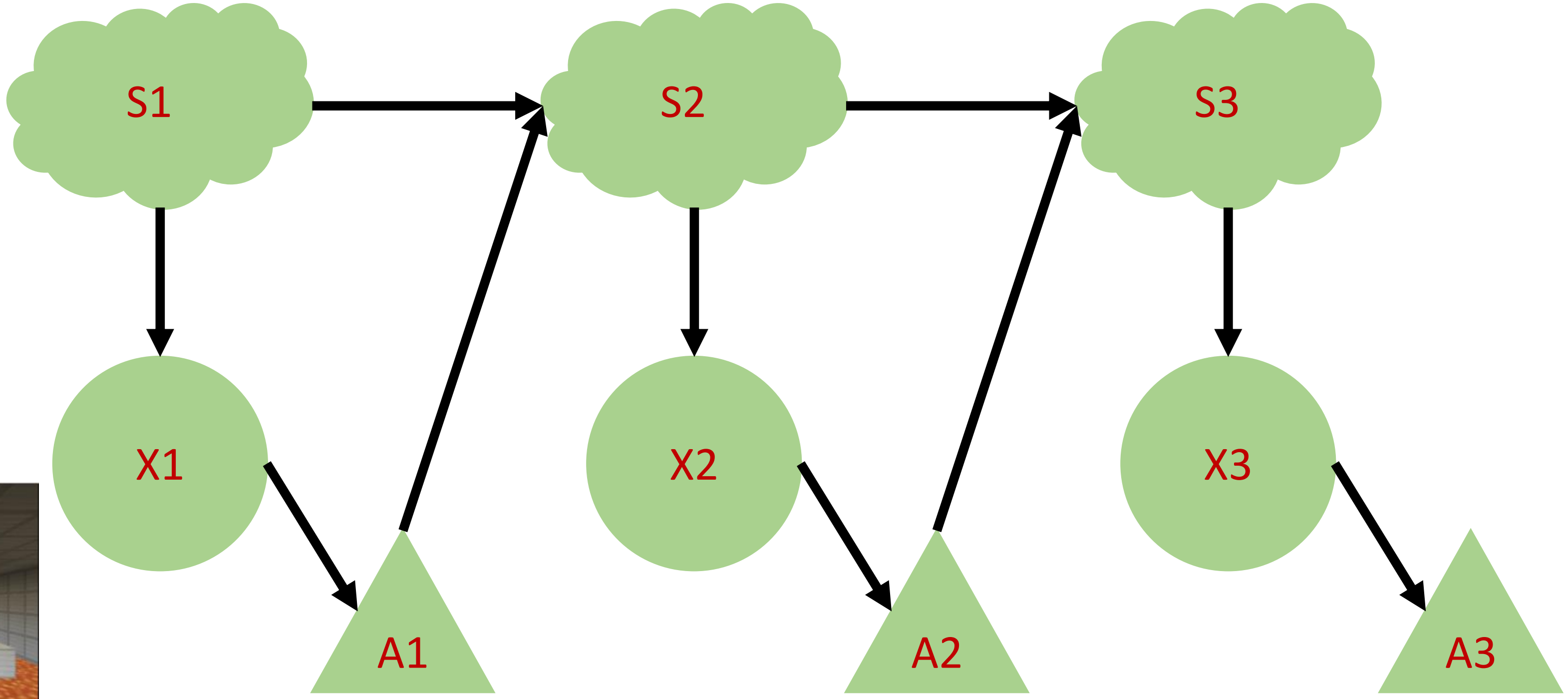


Information theory: [KAL 16] [JKALS 17] [SJKAL 19] [DPWZ 20]
Algorithms for Block MDPs: [DKJADL 19] [FWYDY 20] [FRS-LX 20]

A latent state model: The block MDP







Rich observation problem with discrete latent state space
Agent operates on rich observations
Latent states are decodable from observations, so no partial observability

Nonlinear function approximation

Main guarantee

Assumptions:

1. **Function class:** We have a class of decoders Φ containing the true decoder $\phi^\star$.
2. **Reachability**: Latent states are reachable with probability at least η

Theorem [MHKL19]: Homer covers the states and finds an ϵ-optimal policy using

$$poly(|S|, |A|, H, \frac{1}{\eta}, \frac{1}{\epsilon}, \log(|\Phi|/\delta)) \text{ trajectories}$$

Homer runs in polynomial time assuming supervised learning problems are tractable.

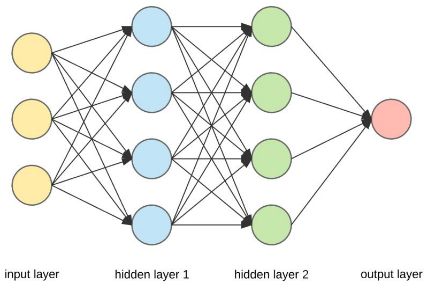
Empirical Results

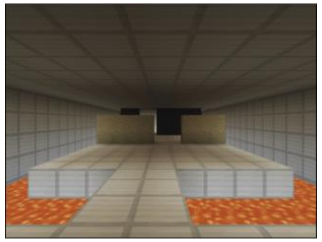
PPO (fails to explore from time step 5)

Methods run for ~1m episodes

Block MDP pros and cons

+ Accommodates nonlinear function approximation
+ Can model many rich observation RL settings
+ Statistically and algorithmically tractable



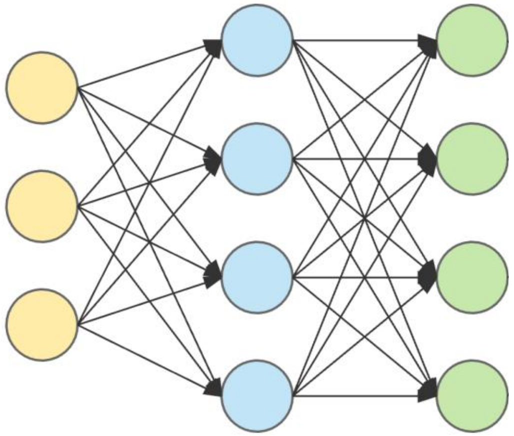


- Discrete/finite latent state space
- Perfect decodability

Meanwhile

Flurry of activity around linear function approximation

- Classical results: [G95] [BB96] [TvR97] [SMcASM00] [PSD01] [LP03] [SSM08] [SMPBSS08] …
- Modern results
    - Exploration [YW19] **[JYWJ19]** [ZBBPL20] [AJSWY20] **[AHKS20]** [WDYS20] [NP-B20]
    - Representation quality + approximation [DKWY19] [LS19] [vRD19]
    - Batch RL **[DW20]**[WFK20]
    - Weaker assumptions **[LSSS20]** [DLMW20] **[ZLKB20]** [WAS20]
    - Infinite horizon **[WJLJ20]**
    - Adversarial losses **[CYJW20]** [NO20]



But where do the features come from?

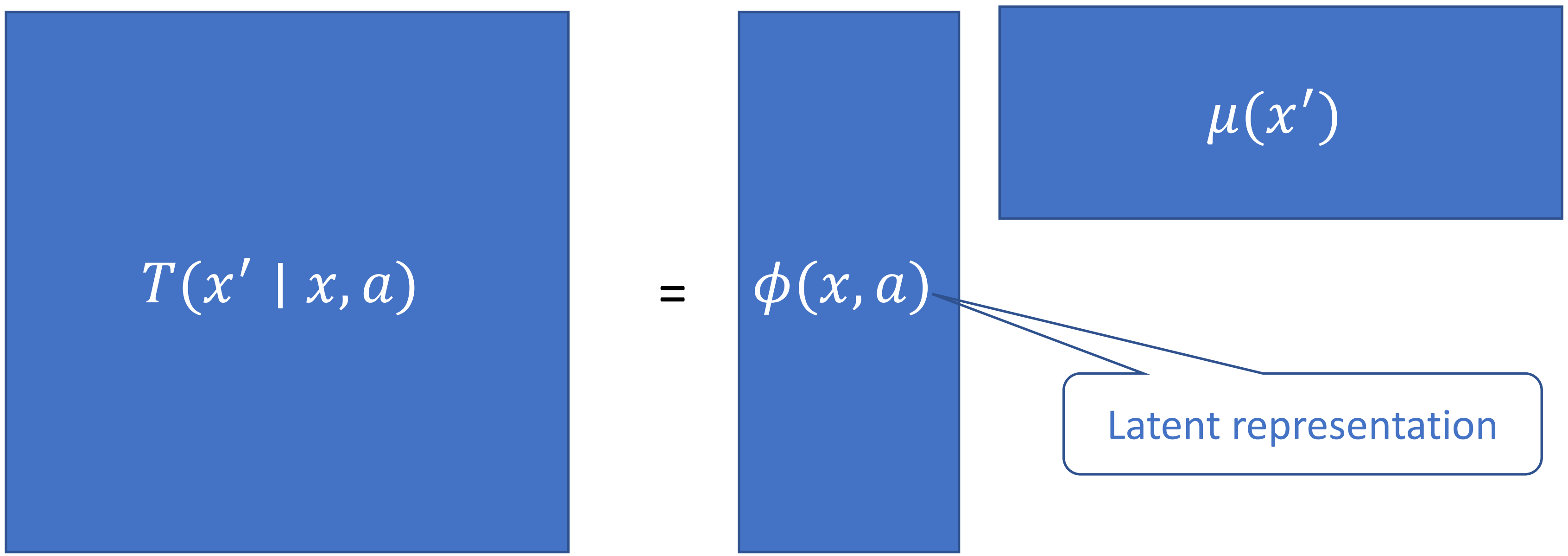
This Talk

Provably efficient representation/feature learning in low rank MDPs

- Non-linear function approximation beyond Block MDPs
- Allows us to apply linear RL methods afterwards

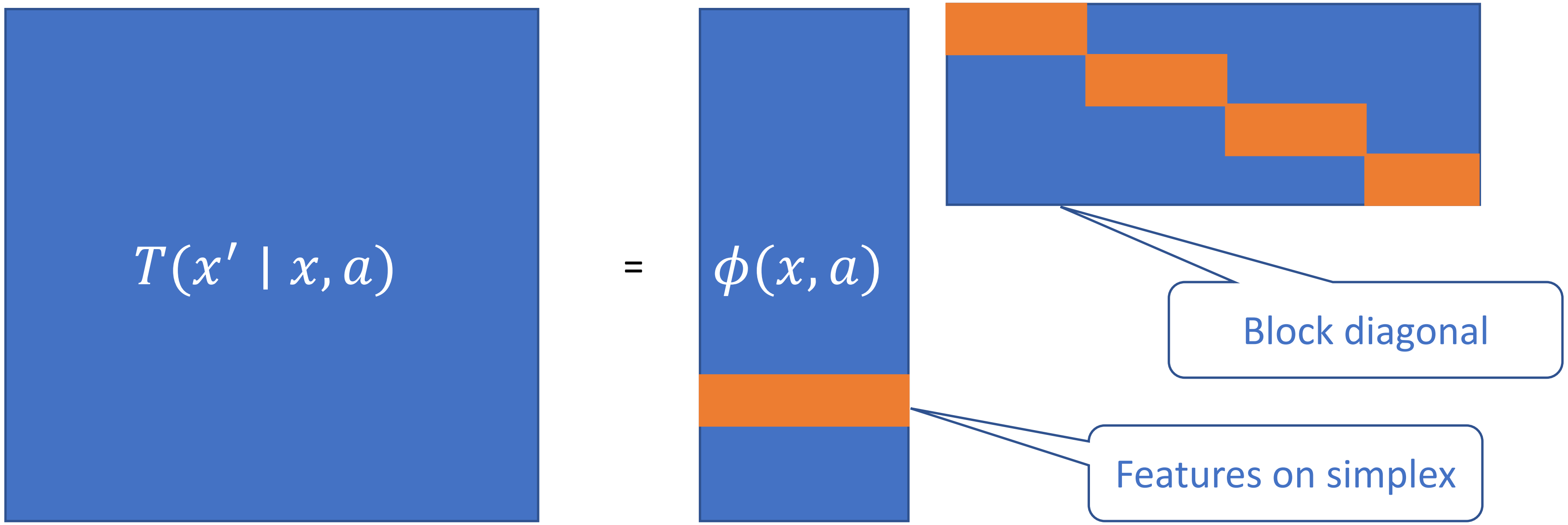
Challenge: Feature learning and exploration are intertwined!

The low rank MDP



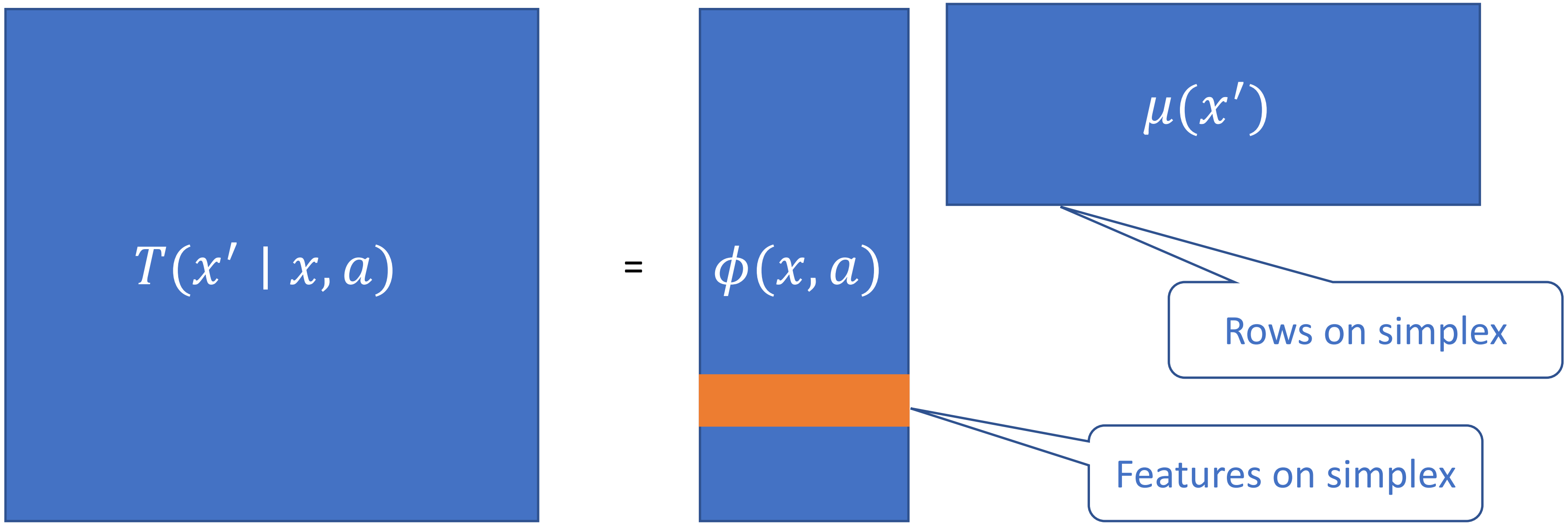
Embedding dimension d $\ll$ size of observation space

Block MDPs are low rank



Proposition: There exist transition operators over N observations with rank 2 that require N latent states in block factorization.

Tangent: beyond decodability



Simplex representation: sample *latent variable* $z \sim \phi(x, a)$ and next state $x \sim \mu(\cdot \mid z)$.
Latent variables not decodable, but not an HMM.
Studied in [BPP11], mentioned in [JYWJ19]

Why study low rank MDPs?

Tractable if feature map is known

Theorem [JYWJ19]: Optimistic LSVI has regret $\tilde{O}(\sqrt{d^3 H^3 T})$ when ϕ is known

Statistically tractable even without

Proposition [JKALS17]**:**
- Low rank MDPs have Bellman rank d for *any* function class
- With class Φ of embeddings and realizability, OLIVE has sample complexity:

$$\tilde{O}(d^2 H^3 |A|(d + \log|\Phi|)/\epsilon^2)$$

But OLIVE is not computationally efficient

Main guarantee

System Identification

Assume function class realizability: Φ, Υ contain the true dynamics
Assume oracle computation model: Can optimize/sample from Φ, Υ

Theorem [AKKS20]: FLAMBE learns a low rank MDP model such that

$$\forall\ \pi, h: \quad \mathbb{E}_\pi \left\| \langle \hat{\phi}_h(x_h, a_h), \hat{\mu}_h(\cdot) \rangle - T_h(\cdot \mid x_h, a_h) \right\|_{\mathrm{TV}} \leq \varepsilon$$

With sample complexity:

$$poly(d, |A|, H, \frac{1}{\varepsilon}, \log(|\Phi||\Upsilon|/\delta))$$

No reachability required!

FLAMBE runs in polynomial time in oracle model.

Potpourri

Representation learning:

For any reward, optimal policy (and Q function) for $\widehat{M}$ are linear in $\hat{\phi}_{1:H}$

$\Rightarrow$ near-optimal policy (and Q function) for M are linear in $\hat{\phi}_{1:H}$

Reward-free learning:

Can efficiently optimize any reward function with no further experience

Real-world planning:

Can replace model-based planning with real world planning in FLAMBE

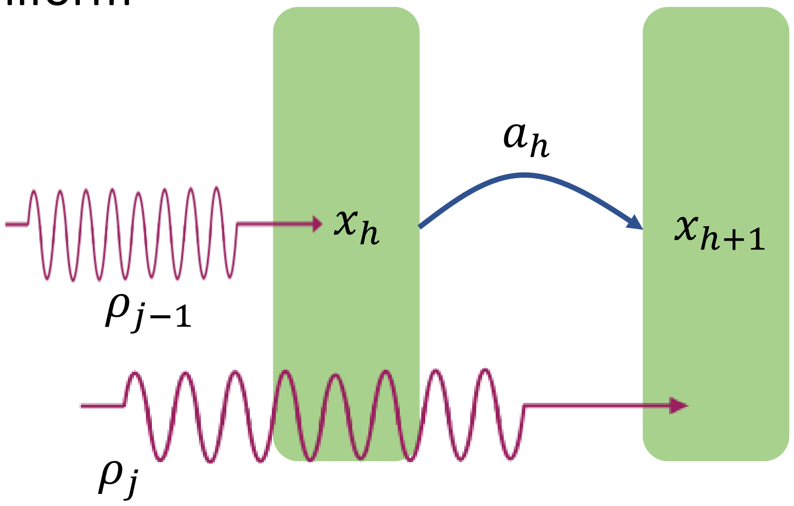
- No need for sampling from models
- But requires a reachability assumption

A model-based algorithm

Ideally ensures good data coverage at time h

ρ_0 = random policy
For j $= 1, \dots, J_{\max}$:

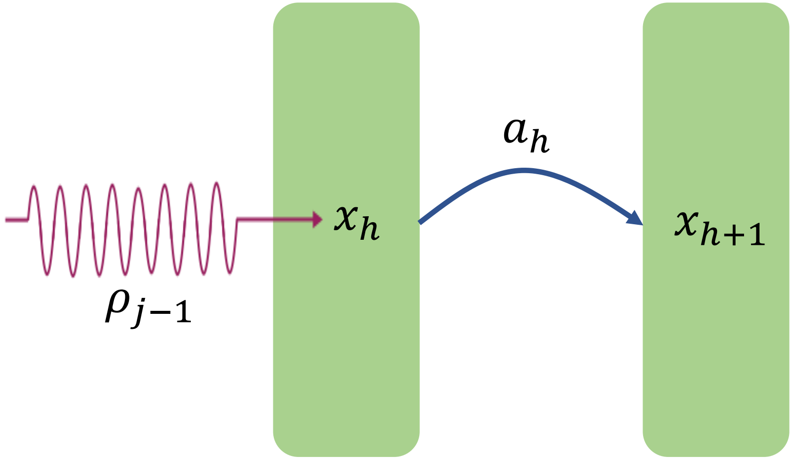
  For each h use ρ_{j-1} to collect data with a_h uniform
  For each h learn dynamics $\hat{T}_h$ using all data
  Compute exploratory policy ρ_j



Questions

1. How to learn dynamics?
2. How to compute exploratory policy?

Learning one-step model



Collect n triples (x_h, a_h, x_{h+1}) from $\rho_{j-1} \circ unif(A)$

Solve MLE problem

$$(\hat{\phi}_h, \hat{\mu}_h) = \underset{\phi,\mu}{\operatorname{argmax}} \sum_{x_h, a_h, x_{h+1}} \log\langle \phi(x_h, a_h), \mu(x_{h+1}) \rangle$$

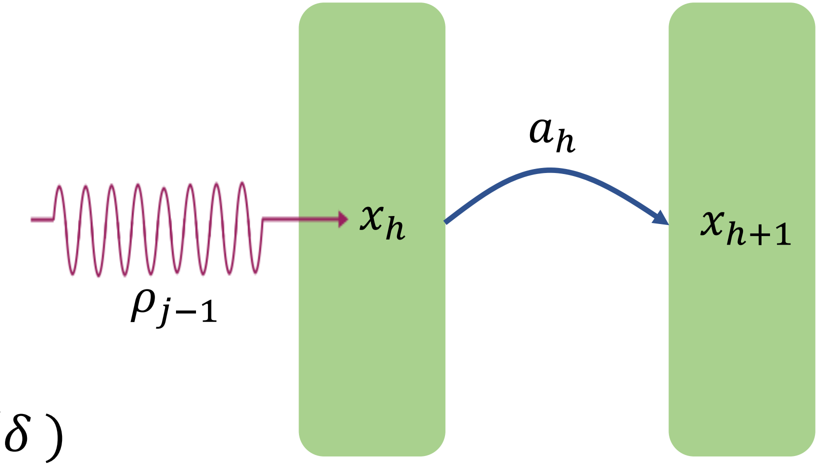
Function classes:
$\phi \in \Phi, \mu \in \Upsilon$

Theorem [Z07]: With realizability, can guarantee:

$$\mathbb{E}_{x_h, a_h \sim \rho_{j-1} \circ \mathrm{unif}(A)} \left\| \langle \hat{\phi}_h(x_h, a_h), \hat{\mu}_h(\cdot) \rangle - T(\cdot \mid x_h, a_H) \right\|_{\mathrm{TV}}^2 \leq \frac{2 \log(|\Phi||\Upsilon|/\delta)}{n}$$

$\mathrm{err}(x_h, a_h)$

Learning one-step model



Martingale version:

$$\sum_{i=0}^{j-1} \mathbb{E}_{x_h, a_h \sim \rho_i \circ \mathrm{unif}(A)} \mathrm{err}(x_h, a_h) \leq \frac{2 \log(|\Phi||\Upsilon|/\delta\,)}{n}$$

Error transfer: Define $\Sigma_{h,j} = \lambda I + \sum_{i=0}^{j-1} \mathbb{E}_{\rho_i} \phi(x_h, a_h) \phi(x_h, a_h)^\top$

$\mathrm{Err}(\Sigma_{h-1,j})$

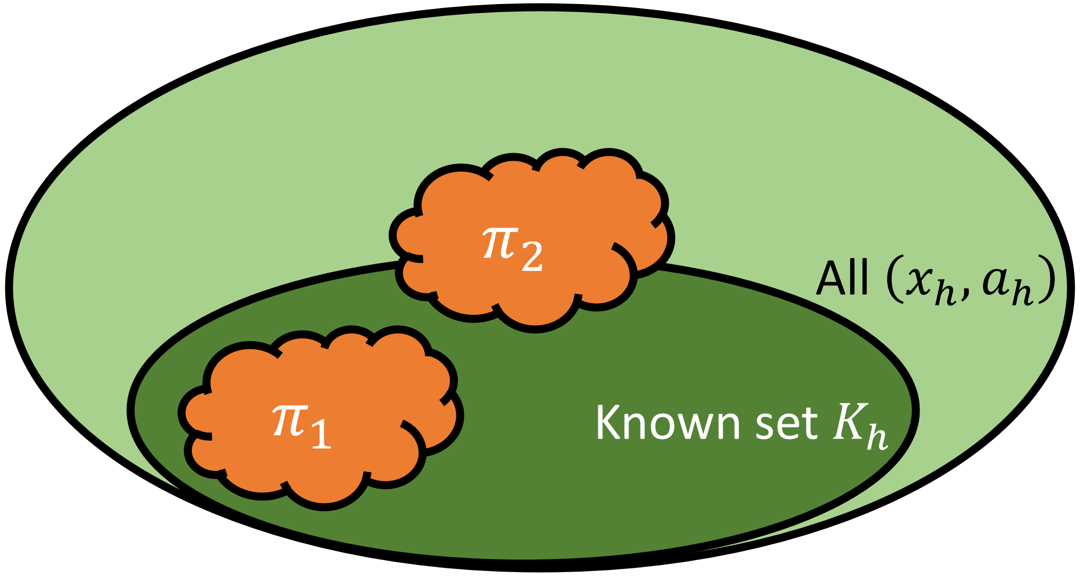
$$\| \, \Sigma_{h-1,j}^{1/2} \cdot \int \mu(x_h) \mathrm{unif}(a_h) \cdot \sqrt{err(x_h, a_h)} \, \|^2 \leq \lambda d + \frac{2 \log(|\Phi||\Upsilon|/\delta\,)}{n}$$

Independent of π

Key property of low rank MDPs: For any function f and any policy π

$$\mathbb{E}_\pi f(x_h) = \langle \mathbb{E}_\pi \phi(x_{h-1}, a_{h-1}), \int \mu(x_h) f(x_h) \rangle$$

Simulation Lemma



We have $\Sigma_{h,j}$ for each h

Define known set $K_h = \{\|\Sigma_{h,j}^{-1/2}\phi(x_h, a_h)\|_2 \leq 1\}$

Define absorbing MDP M_K where unknown (x_h, a_h) transit to absorbing state.

Simulation lemma: For any function f with range [0,1] and any policy π

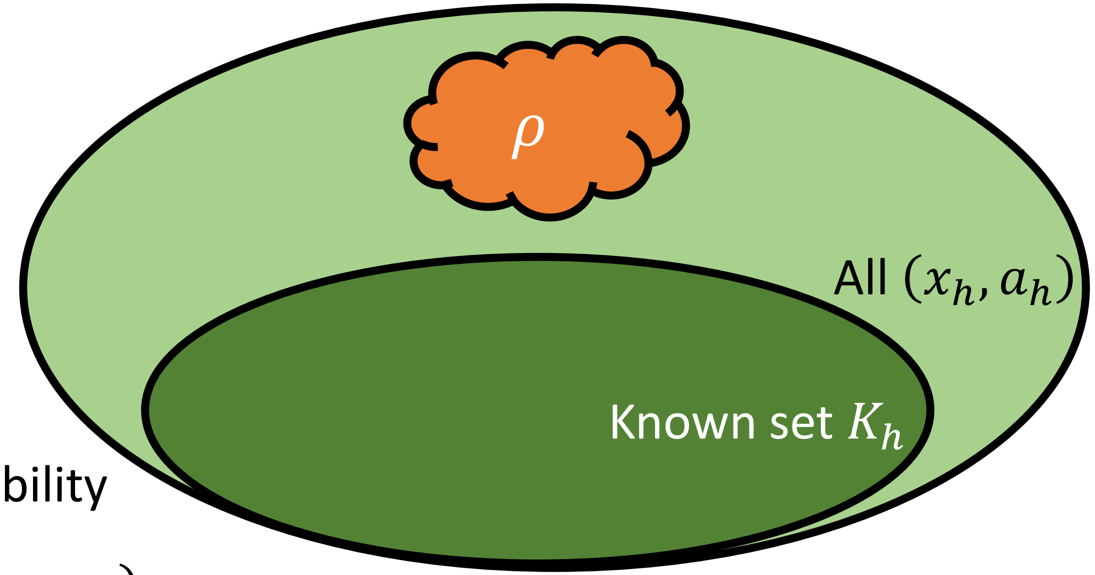
$$\mathbb{E}_\pi[f(x_h, a_h) \mid M_K] \leq \mathbb{E}_\pi\left[f(x_h, a_h) \mid \widehat{M}\right] + |A| \cdot \sum_{h'} Err\left(\Sigma_{h',j-1}\right)$$

$$\mathbb{E}_\pi\left[f(x_h, a_h) \mid \widehat{M}\right] \leq \mathbb{E}_\pi[f(x_h, a_h) \mid M_K] + |A| \cdot \sum_{h'} Err\left(\Sigma_{h',j-1}\right) + \sum_{h'} \mathbb{P}_\pi[(x_{h'}, a_{h'}) \notin K_{h'} \mid M]$$

Small by MLE argument

“escape” probability

Planning



We want exploratory policy ρ to have large escape probability

$$\Delta \leq \mathbb{P}_\rho[(x_h, a_h) \notin K_h] \leq \mathbb{E}_\rho \phi(x_h, a_h)^\top \Sigma_{h,j}^{-1} \phi(x_h, a_h)$$

This can only happen $\sim d \,/\, \Delta$ times.

Elliptical potential using ϕ

Challenge: We do not know K_h as it depends on true features ϕ

Solution: We plan to visit all directions of our learned features $\hat{\phi}$ *at the previous time*

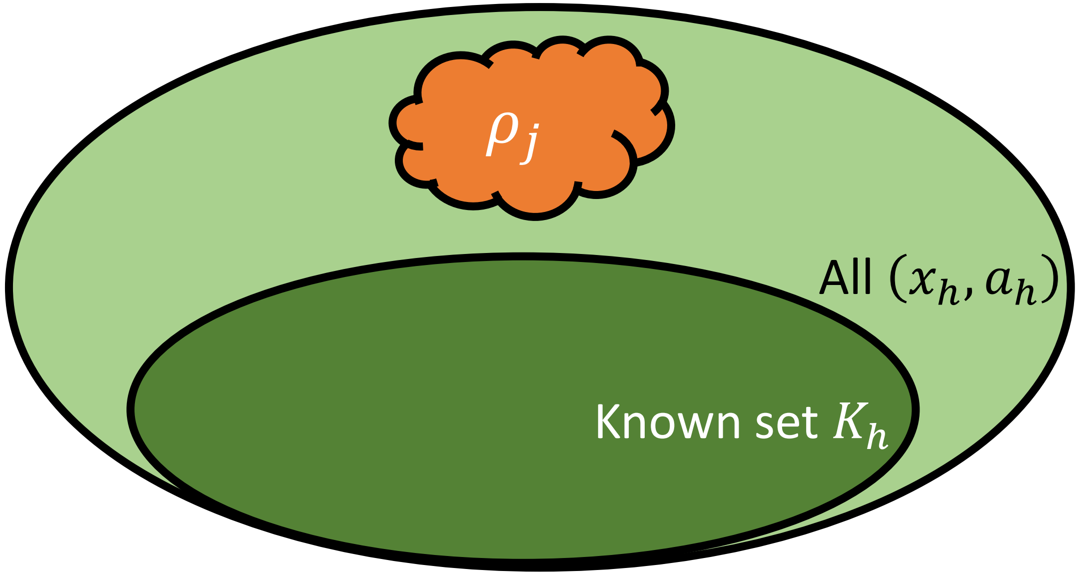
By iteratively maximizing quadratic forms, ρ guarantees that

$$\max_\pi \mathbb{E}_\pi\left[\, \hat{\phi}_{h-1} \, \hat{\Sigma}_\rho^{-1} \, \hat{\phi}_{h-1} \;\middle|\; \widehat{M} \,\right] \leq O(d)$$

Elliptical potential using $\hat{\phi}$

By simulation lemma, either ρ escapes earlier or $\rho \circ \mathrm{unif}\,(A)$ has large escape probability at h.

Final steps



Ingredients:

- Simulation lemma with escaping
- ρ_j approximately maximizes escaping

Case analysis for iteration j:

- If ρ_j escapes with high prob, then we learn a lot: $\Sigma_{h,j} \ll \Sigma_{h,j+1}$.
  - Can only happen in polynomially many iterations.
- If ρ_j escapes with low prob, then no other policy can escape $\Rightarrow$ we are done!
  - No policy can escape and $\widehat{M} \approx M$ in the known set

The landscape

Block MDPs

Known representation
[KS02] [AOM17] [DLB17], etc.

Reachable latent variables

Unknown representation
[DKJADL19] [FWYDY19][FR-SLX20]
Homer [MHKL19]

Low rank MDPs

Known representation
[JYWJ19] [YW19], etc.

Unknown representation
FLAMBE [AKKS20]

Bellman/ Witness rank

Computationally intractable
[JKALS17]
[SJKAL19]

Discussion

- Our approach decouples dynamics assumptions from observations
  - Allow expressive non-linear function approximation, yet tractable
- *Dependence on |A|?* Seems necessary here without further assumptions
- *Sharp rates and regret?*
- *Does it actually work?* We are trying

Homer: https://arxiv.org/abs/1911.05815
FLAMBE: https://arxiv.org/abs/2006.10814