Reinforcement Learning on MDPs

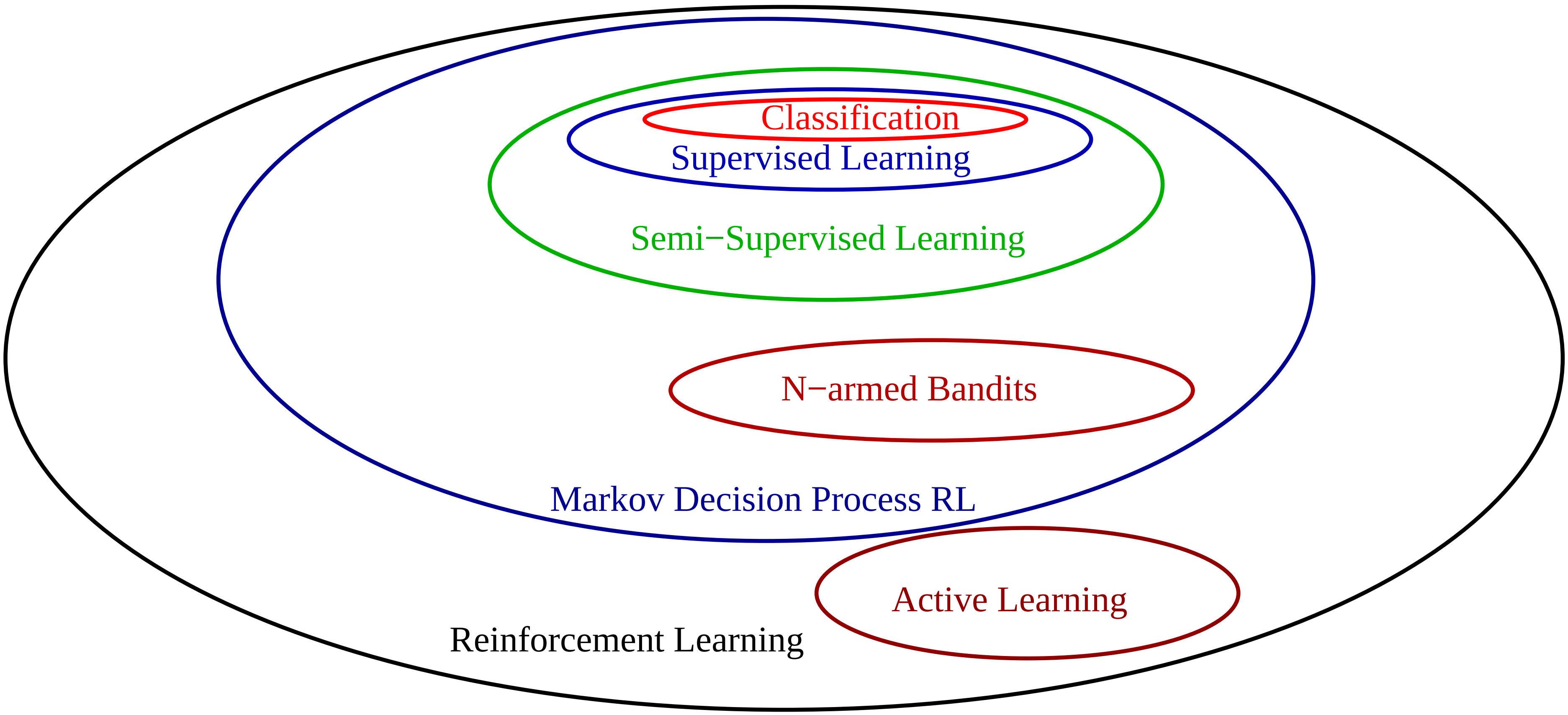
John Langford

Yahoo Research

Backing Material: http://hunch.net/~jl/tutorial/RL.html

COMS-4771, Columbia

Reinforcement Learning is Always Relevant



The answer to: "Is this an RL problem?" is always "yes".

The implication: RL theory is broadly applicable.

The other implication: RL theory is often only weakly relevant. (breadth+relevance=hard.)

Understanding a problem as an RL problem is the *beginning* to solving it. Whenever possible, you want to understand how the problem is special.

Outline

1. Sample Complexity Results

2. Limitations of Sample Complexity

Markov Decision Process (MDP)

1. S = the number of states in an MDP

2. A = the number of actions/state in an MDP

3. T = the horizon time you care about (or γ = discount factor)

4. O = number of observations

5. ϵ = precision parameter

Important Derived Quantities

$$V_t^\pi(s) = E_{(s,a,r)^t \sim \pi, \mathsf{MDP}_s}\left[\sum_{t'=1}^{t} r_{t'}\right]$$

= the value of being in state s and acting according to π for t timesteps.

$$Q_t^\pi(s,a) = E_{(s,a,r)^t \sim \pi, \mathsf{MDP}_{sa}}\left[\sum_{t'=1}^{t} r_{t'}\right]$$

= the value of being in state s, acting with a, and then acting according to π for t timesteps.

$$\pi^*(s) = \arg\max_a Q_t^{\pi^*}(s,a)$$

= recursive definition of optimal policy.

$$Q_t^*(s,a) = Q_t^{\pi^*}(s,a)$$

= short hand for optimal policy Q values.

The E^3 Guarantee

Trace Model = ability to read current state s, take action a, observe next state s' and reward r. Notation: $TM : A \rightarrow S \times [0, 1]$.

Assume $\text{MDP}(S, A, p(s'|s, a))$ with horizon T

1. Original: +assume mixing time $\tau \Rightarrow \text{Poly}(S, A, \tau, \frac{1}{\epsilon})$ samples implies ability to act ϵ optimal for $T > \tau$.

2. Modified: $\text{Poly}(S, A, T, \frac{1}{\epsilon})$ samples implies ability to act ϵ optimal for T timesteps.

(2) + mixing assumption implies (1). (2) holds even for deterministic worlds. We'll go through (2).

E^3 Theorem Statement

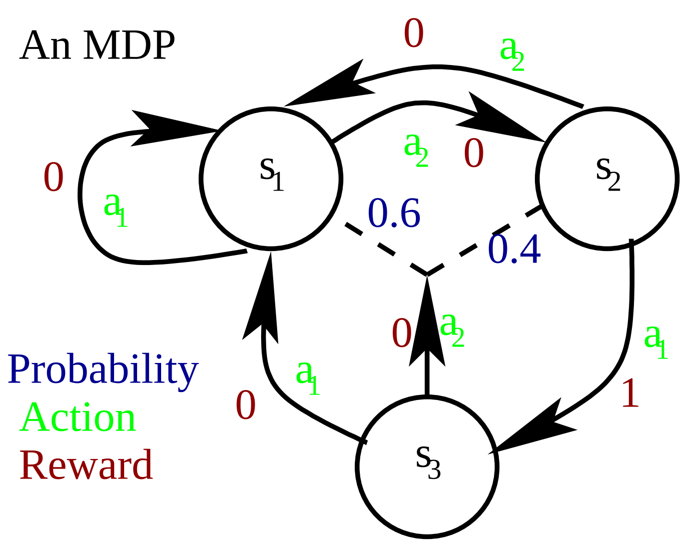
Theorem: There exist an algorithm E^3 such that for all MDP $(S, A, T, p(s'|a,s))$ with rewards $r \in [0,1]$, with probability $1-\delta$, for all except $\mathrm{Poly}\left(S, A, T, \frac{1}{\epsilon}, \ln\frac{1}{\delta}\right)$ steps $Q^{E^3}_{T-t \bmod T}(s, E^3(h)) \geq V^*_{T-t \bmod T}(s) - \epsilon$ where h is the history of observations.

Suboutline:

1. The Algorithm
2. The Proof

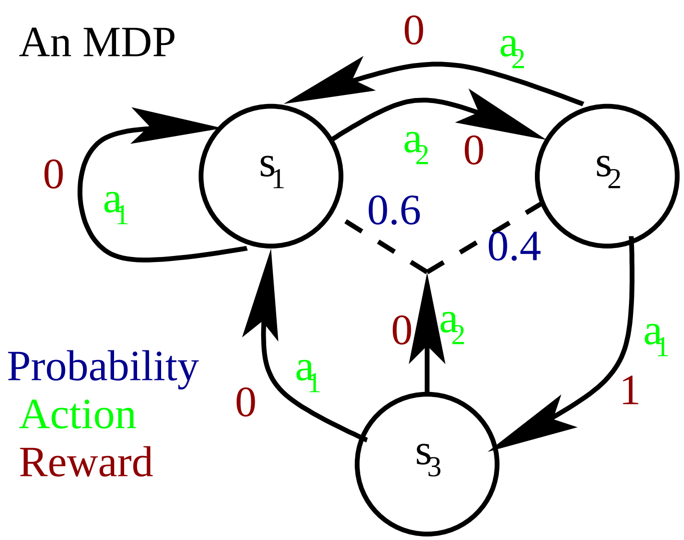
The Known(h) MDP

A state s, all actions a leaving s and the probability of their outcomes is known if all actions a leaving s have been executed at least n times.

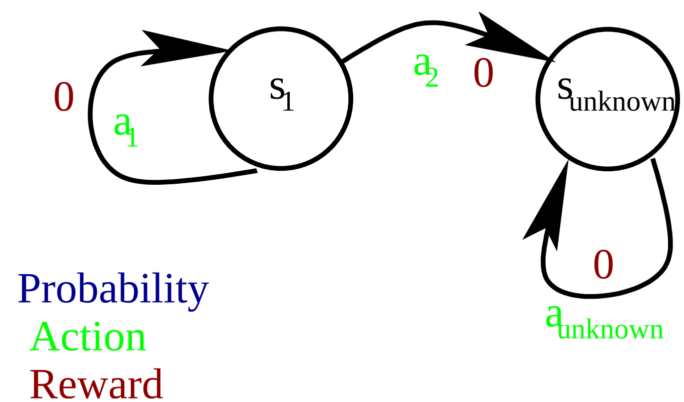


Initially: known MDP = nothing

The Known(h) MDP

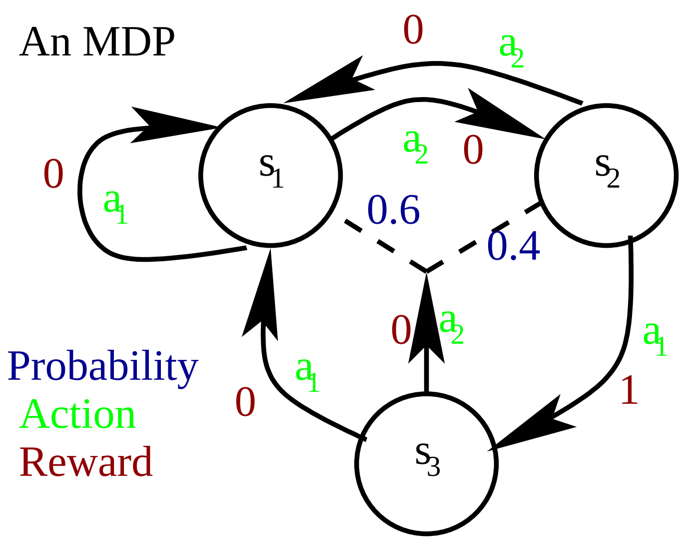


Then:

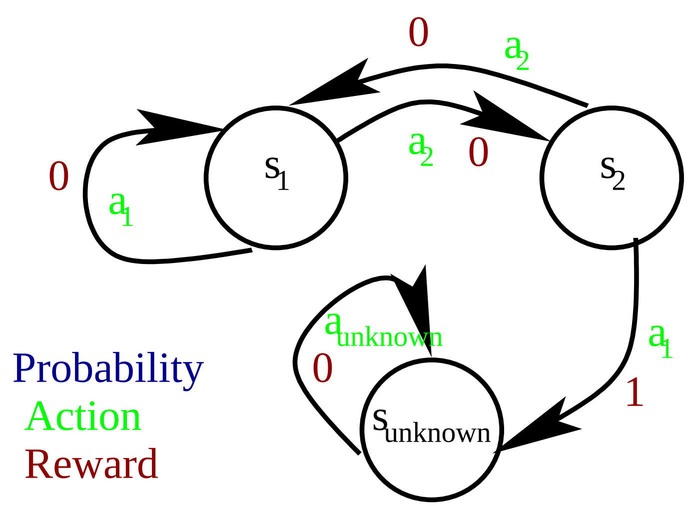


Complete dangling action(s) with one state that always has reward 0.

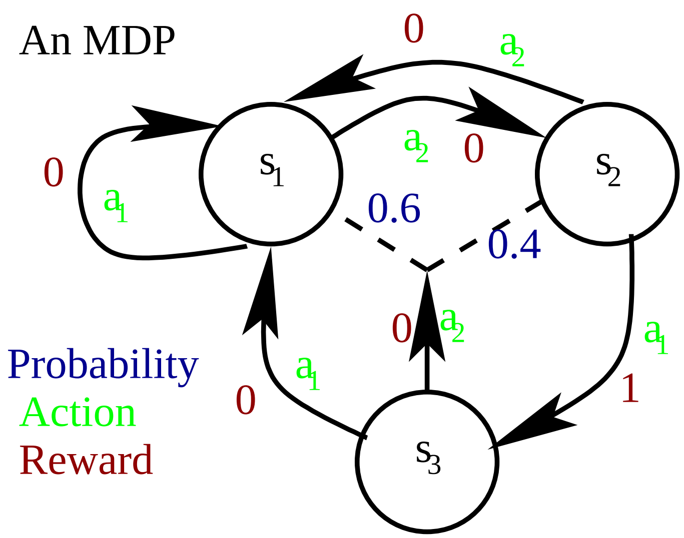
The Known(h) MDP



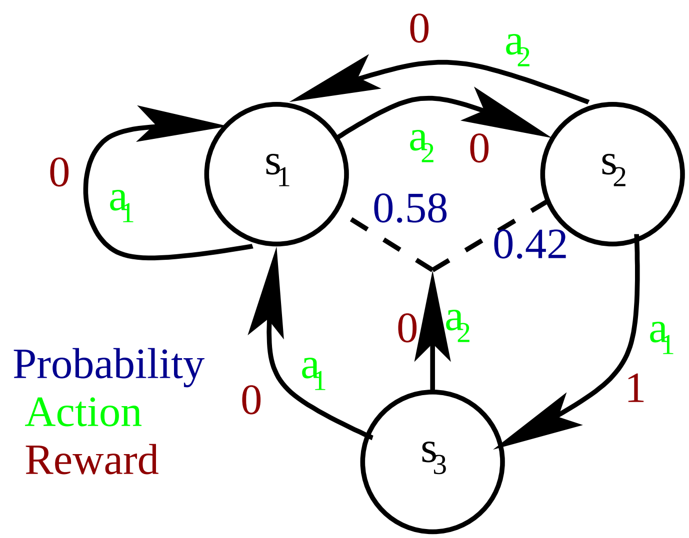
Then:



The Known(h) MDP



Finally:



(note: the probabilities are empirical counts)

The Unknown(h) MDP

Unknown(h) $=$ Known(h) except the reward is 1 for actions which leave the known states and 0 otherwise.

Dynamic Program

Fundamental operation: Given MDP M and state s,

$$\mathsf{DP}(M, s, t) = a, v$$

where v = the maximum expected $T - (t \bmod T)$ reward sum and a= action achieving it.

Computation:

$$\mathsf{DP}(M, s, t) = \max_a E_{s', r \sim M(s,a)} r + \mathsf{DP}(M, s', t + 1)$$

$$\mathsf{DP}(M, s, nT) = 0$$

$E^3(h)$ Explicit Explore or Exploit Algorithm

1. If last s not in Known(h): choose the least previously used action

2. Else:

   (a) If DP(Unknown(h)) $> \epsilon'$ then act according DP(Unknown(h)) until state is unknown or t mod $T = 0$ then go to (1).

   (b) else act according to DP(Known(h)).

The proof uses 5(!) MDPs

1. MDP — the true MDP (Imposed by world)

2. Known(h) = known MDP (Known by E^3 algorithm)

3. Unknown(h) = unknown MDP (Known by E^3 algorithm)

4. $\mathrm{MDP}_{\mathrm{K}(h)}$ = MDP restricted to the known states (exists only in proof)

5. $\mathrm{MDP}_{\mathrm{U}(h)}$ = MDP restricted to the known states with rewards set to 0 except for escaping rewards. (exists only in proof)

Proof Sketch:

Simulation Lemma:

$$|\text{DP}(\text{MDP}_{\text{U/K}(h)}) - \text{DP}((\text{Un})\text{Known}(h))| \leq \frac{1}{\text{Poly}(S, A, T, \ln \frac{1}{\delta})}$$

Explore/Exploit Lemma:

$$\text{DP}(\text{MDP}_{\text{K}(h)}) + T\text{DP}(\text{MDP}_{\text{U}(h)}) \geq \text{DP}(\text{MDP})$$

So $n = \text{Poly}(S, A, T, \ln \frac{1}{\delta})$ implies ability to simulate on known states to precision $\frac{1}{\text{Poly}(S,A,T,\ln \frac{1}{\delta})} << \epsilon$. $\Rightarrow$ Explore/Exploit Lemma implies $\text{DP}(\text{MDP}) - \text{DP}(\text{MDP}_{\text{K}(h)}) > \epsilon \Rightarrow \text{DP}(\text{MDP}_{\text{U}(h)}) > \frac{\epsilon}{T}$ $\Rightarrow$probability about $\frac{\epsilon}{T}$ of encountering new state if exploring. This can happen only $O(\frac{nSAT}{\epsilon})$ times (Using the Chernoff bound). Each exploration uses at most T steps $\Rightarrow$ proof.

Delayed Q-learning

The theorem can be tightened from Poly(S, A) to $\tilde{O}(SA)$ using the Delayed Q-learning algorithm.

Outline

1. Sample Complexity Results

2. Limitations of Sample Complexity

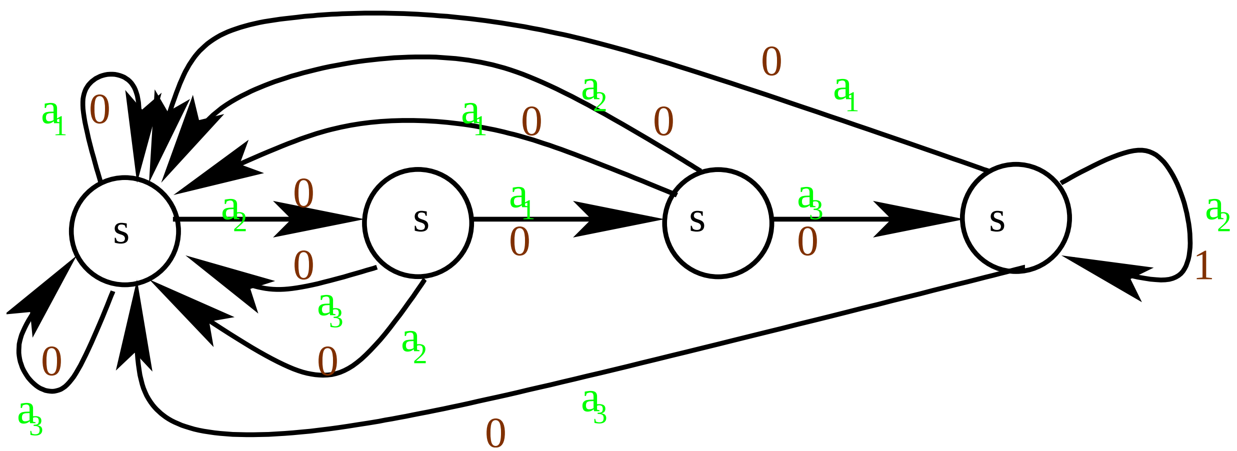
The Limits of Sample Complexity: A lower bound

Theorem: Any algorithm A satisfying the E^3 statement must use at least $\Omega(TSA)$ actions to explore.

(There are stronger lower bounds, but this is sufficient.)

Proof

A "Key lock" MDP



States in a chain. One action leads to next state, all the rest lead to the beginning. The final state has an action with reward 1.

Implications

Lower bound $\Rightarrow$ the really big problems can't be solved.

But the problems *are* solvable: we solve them every day.

$\Rightarrow$More or different assumptions are required.

Related Reading

[E^3] Michael Kearns and Satinder Singh, "Near Optimal Reinforcement Learning in Polynomial Time", ICML 1998.

[Delayed Q-learning] Alexander Strehl et al, "PAC Model-Free Reinforcement Learning", ICML 2006.

[Sparse Sampling] Michael Kearns, Yishay Mansour, and Andrew Ng, "A Sparse Sampling Algorithm for Near-Optimal Planning in Large Markov Decision Processes", IJCAI 1999.

[Many things] Sham Kakade, On the Sample Complexity of Reinforcement Learning Thesis Gatsby, UCL, 2003.