Large-Scale Markov Decision Problems with KL Control Cost and their Application to Crowdsourcing

Yasin Abbasi-Yadkori ¹, Peter Bartlett ¹², Xi Chen ³, Alan Malek ²

¹Queensland University of Technology
²University of California, Berkeley
³NYU Stern School of Business

July 7th, 2015
Problem: MDP planning problem with large state space
Goal: find near-optimal policy in low dimensional family of policies
Novel framework for linearly solvable MDPs
Also: Algorithm with complexity that scales with dimension of family
First theoretical bounds for approximate solutions in linearly solvable MDPs
Demonstrate on practical example


Solving LMDPs (with no theoretical guarantees): [Todorov, 2009] and [Zhong and Todorov, 2011a,b]

Approximate policy iteration (e.g. least squares policy iteration)
1 Motivation

2 Linearly Solvable MDPs

3 Extending to large dimensions

4 Experiments
Large Scale MDPs

- Markov decision process: modeling sequential decisions
- E.g. queueing network, robot planning
- Can solve for small state spaces
- Applications have large state spaces
A Markov Decision Process is specified by:
- State space $X = \{1, \ldots, X\}$
- Action space $A$
- Transition Kernel $K : X \times A \rightarrow \Delta X$
- Loss function $\ell : X \times A \rightarrow \mathbb{R}^+$

Problem:
- Policy $\pi : X \rightarrow \Delta A$
- Find policy to minimize value function

$$
J_\pi(x) = \mathbb{E} \left[ \sum_{t=0}^{\infty} \ell(X_t, \pi) \middle| X_0 = x \right]
$$

Aim for optimality within a restricted family of policies.
Large state space

- Parametric class of value functions $J_\theta$ for $\theta \in \Theta \subset \mathbb{R}^d$
- Bellman operator:

$$\left(LJ\right)(x) = \min_{a \in \mathcal{A}} \{ \ell(x, a) + \mathbb{E}_{x' \sim P_0(x,a)} J(x') \}$$

- Optimal policy $J^*$ is a fixed point: $LJ^* = J^*$
- Greedy policy: $\pi_{J_\theta}$ (the argmin)
- Ultimate goal: find a $\theta$ to minimize

$$J_{\pi_{J_\theta}} ,$$

the actual value of the greedy policy of the approximate optimal value
Approximate solutions

Consider the unconstrained surrogate

\[
\min_{\theta} c^T J_\theta + \| LJ_\theta - J_\theta \|
\]

Can we solve this with algorithms that scale with \( d \) but not \( X \)?
1 Motivation

2 Linearly Solvable MDPs

3 Extending to large dimensions

4 Experiments
KL-cost

- Introduced in [Todorov, 2006]
- $A = \Delta x$
- Loss: $\ell(x, P) = q(x) + D_{KL}(P || P_0(\cdot | x))$
  - state loss $q(x)$, base dynamics $P_0$
  - infinite loss unless $P \ll P_0$
- Terminal state $z$
- Total cost of policy $P$

$$J_P(x) = \mathbb{E} \left[ \sum_{t=0}^{\infty} \ell(X_t, P) \right| X_0 = x$$
Greedy action is:

\[ P_J(\cdot|\mathbf{x}) = \operatorname{arg\ min}_{P \in \Delta_X} \mathbb{E}_{y \sim P(\cdot|x)}[q(y) + J_P(y)] \propto P_0(\cdot|x) e^{-J_P(\cdot)} \]

Bellman’s operator becomes linear in \( g(x) = e^{-J(x)} \):

\[ e^{-LJ(x)} = e^{-q(x)} \sum_{x'} P_0(x, x') e^{-J(x')} \]

Bellman’s optimality equation:

\[ LJ = J \iff e^{-q} P_0 e^{-J} = e^{-J} \]
1 Motivation

2 Linearly Solvable MDPs

3 Extending to large dimensions

4 Experiments
Parameterizing $J_\theta$

- Previous ADP techniques used $J_\theta = \Psi_\theta$
- Intuition: take $J_\theta = -\log(\Psi_\theta)$ so $e^{-LJ_\theta}$ is linear in $\theta$
- Surrogate optimization:

$$\min_\theta c^\top J_\theta + \|LJ_\theta - J_\theta\|$$

Bellman error

- $\|LJ_\theta - J_\theta\|$ not convex in $\theta$, but

$$e^{-\max\{LJ_\theta,J_\theta\}} \|LJ_\theta - J_\theta\| \leq \|e^{-LJ_\theta} - e^{-J_\theta}\|$$

- Plugging $\Psi_\theta = e^{-J_\theta}$ into (1):

$$\min_\theta -c^\top \log(\Psi_\theta) + \|e^{-qP_0}\Psi_\theta - \Psi_\theta\|$$
Parameterizing $J_\theta$

- Previous ADP techniques used $J_\theta = \psi_\theta$
- Intuition: take $J_\theta = -\log(\psi_\theta)$ so $e^{-LJ_\theta}$ is linear in $\theta$
- Surrogate optimization:

$$
\min_{\theta} c^T J_\theta + \left\| LJ_\theta - J_\theta \right\|
$$

Bellman error

- $\left\| LJ_\theta - J_\theta \right\|$ not convex in $\theta$, but

$$
e^{-\max\{LJ_\theta, J_\theta\}} \left\| LJ_\theta - J_\theta \right\| \leq \left\| e^{-LJ_\theta} - e^{-J_\theta} \right\|
$$

- Plugging $\psi_\theta = e^{-J_\theta}$ into (1):

$$
\min_{\theta} -c^T \log(\psi_\theta) + \left\| e^{-qP_0\psi_\theta} - \psi_\theta \right\|
$$

Bellman operator
Our algorithm

- Recall relaxed optimization:

\[
\min_{\theta} -c^\top \log(\psi_\theta) + \| e^{-q} P_0 \psi_\theta - \psi_\theta \|_Q
\]

- Let \( \mathcal{T} \) be the set of trajectories with \( x_1 \sim c \) with distribution \( Q(\cdot) \)

- Optimization is equal to:

\[
\min_{\theta} -c^\top \log(\psi_\theta) + \sum_{T \in \mathcal{T}} Q(T) \sum_{x \in T} \left| e^{-q(x)} P_0 \psi_\theta(x) - \psi_\theta(x) \right|
\]

- Use stochastic gradient descent by sampling trajectories
Theorem

Let $\hat{\theta}$ be an $\epsilon$-optimal solution returned by SGD. Then,

$$J_{P_{J_{\hat{\theta}}}}(x_1) \leq \inf_{\theta \in \Theta} \left\{ J_{P_{J_{\theta}}}(x_1) + \mathcal{E}(J_{\theta}) \right\} + \epsilon$$

$$+ \left\| P_{J_{\hat{\theta}}} - Q \right\|_1 \max_{T \in T} \sum_{x \in T} \left| J_{\hat{\theta}}(x) - LJ_{\hat{\theta}}(x) \right|$$

Off-policy error

Penalty function:

$$\mathcal{E}(J_{\theta}) = \sum_{T \in T} \sum_{x \in T} \left( Q(T) e^{-\min(J_{\theta}, LJ_{\theta})} + P_{J_{\theta}}(T) \right) \left| J_{\theta}(x) - LJ_{\theta}(x) \right|$$

Small if $J_{\theta}$ is close to the optimal value
1 Motivation

2 Linearly Solvable MDPs

3 Extending to large dimensions

4 Experiments
Crowdsourcing

- Need to label $A$ items.
- Each item has soft label $\mu_i \in [0, 1]$
- Guess if $\mu_i \geq \frac{1}{2}$ for as many $i$ as we can
- For $t = 1, \ldots, T$:
  - Pick $i \in \{1, \ldots, A\}$
  - Receive $X_t \sim \text{Bern}(\mu_i)$
- Use Beta prior $\Rightarrow$ MDP dynamics equivalent to Bayesian updates
- $P_0$ limits transitions
- $q(x)$ rewards correct labels
- Average error of three policies
- Our method requires 10% fewer samples for same accuracy

- Portion of budget vs. soft label
- Harder soft labels receive more budget
- Larger difference as $B$ grows
Conclusion

- Novel framework for low dimensional policies for linearly solvable MDPs
- Algorithm for policy optimization with complexity that scales with dimension of subspace
- First theoretical bounds for approximate linearly solvable MDP solutions
- Demonstrate on practical example
Thanks!
Proof outline of main theorem

\[ \left| J_{P_{J^*}}(x_1) - J^*(x_1) \right| = O(\| LJ^* - J^* \|) \]

Similarly bounding \[ \left| J_{P_{\hat{J}}}(x_1) - \hat{J}(x_1) \right| = O(\| LJ_{\hat{J}} - \hat{J} \|) \]

\( J^* \) and \( \hat{J} \) are close by the optimization