**Motivation**

- Markov decision process: modeling sequential decisions
- E.g. queueing network, robot planning
- Dynamic Programming can solve for small state problems
- Applications can have large state spaces
- Here: in the KL-cost setting, can efficiently do large state spaces

**Notation**

An MDP is defined by:

- State space \( X = \{1, \ldots, X\} \)
- Action space \( \mathcal{A} \)
- Transition Kernel \( K : X \times \mathcal{A} \rightarrow \Delta_X \)
- Loss function \( \ell : X \times \mathcal{A} \rightarrow [0, 1] \)

The problem is to:

- Policy \( \pi : \mathcal{A} \rightarrow \Delta_A \)
- Find policy to minimize

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

**LINEARLY SOLVABLE MDPs FROM [TODOROV]**

- \( \mathcal{A} = \Delta_X \)
- Loss: \( \ell(x, P(\cdot|x)) = q(x) + D_{KL}(P(\cdot|x)||P_0(\cdot|x)) \)
  - state loss \( q(x) \), base dynamics \( P_0 \)
  - infinite loss unless \( P \ll P_0 \)
- Terminal state: \( q(x) = 0 \) and \( P_0(z|x) = 1 \)
- Total cost of policy \( P \)

\[
J_p(x) = \mathbb{E} \left[ \sum_{t=0}^{\infty} \ell(X_t, P) | X_0 = x \right]
\]

- Greedy action is:

\[
P_T(\cdot|x) = \arg \min_{p \in \Delta_X} J_p(x) \propto \frac{1}{Z(x)} R_0(x') e^{-J_p(x')}
\]

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

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

**LARGE STATE SPACES FOR LMDPs**

- Intuition: take \( J_\theta = -\log P(\Psi) \) so \( e^{-J_\theta} \) is linear in \( \theta \)
- Approximate unconstrained optimization:

\[
\min_{\theta} e^{-J_\theta} + H \| L J_\theta - J_\theta \|
\]

- \( \| L J_\theta - J_\theta \| \) not convex in \( \theta \), but

\[
e^{-\max\{L J_\theta, J_\theta\}} \| L J_\theta - J_\theta \| \leq \| e^{-L J_\theta} - e^{-J_\theta} \|
\]

- Optimization relaxed to:

\[
\min_{\theta} -e^T (\Psi \theta) + H \| e^{-L} P_0 \Psi \theta - \Psi \theta \|
\]

**OUR ALGORITHM FOR TOTAL COST**

**Input**: \( x_1, N, H, \) step sizes \((\eta_t), v.\)

- Initialize \( \theta_1 = 0 \)

**for** \( t = 1, 2, \ldots, N \)**

- Sample trajectory \( (x_1, a_1, \ldots, x_v) \sim v. \)
- Compute the stochastic subgradient \( r_1. \)
- Update \( \theta_{t+1} = P_0(\theta_t - \eta_t r_t) \)

**end for**

- \( \hat{\theta}_T = \frac{1}{T} \sum_{t=1}^{T} \theta_t . \)
- Return policy \( P_{\hat{\theta}_T} \)

**EXTENDING TO LARGE STATE SPACES**

- Parametric class of policies \( \pi_\theta \) for \( \theta \in \Theta \) with losses \( J_\theta \)
- Bellman operator:

\[
(L J_\theta)(x) = \min_{a \in A} \{ \ell(x, a) + E_{K(x,a)}(J_\theta(x')) | x \}
\]

- Optimal policy has \( L J_\theta = J_\theta \)
- Linear Programming formulation:

\[
\min_{\theta} e^T J_\theta, \quad \text{(low cost)} \quad \text{s.t. } \quad L J_\theta \leq J_\theta \quad \text{(\( J_\theta \) is feasible)}
\]

- Look for efficient relaxations, e.g.

\[
\min_{\theta} e^T J_\theta + \| L J_\theta - J_\theta \|
\]

- Previous ADP techniques used \( J_\theta = \Psi \theta \)

**PERFORMANCE BOUND**

**Theorem 1.** Choose \( H \geq e^{-\theta_0(t+1)\gamma/\|\Psi\|^2} \). Let \( \hat{\theta} \) be an \( \epsilon \)-optimal solution.

Then, for any \( \theta \in \Theta \) with \( \theta_0 = \min(J_\theta, L J_\theta) \),

\[
J_{\hat{\theta}}(x) \leq \min_{\theta \in \Theta} \left\{ J_{\theta}(x) + \mathcal{E}(\theta) \right\} + \epsilon + \|P_0 - Q\|_1 \max_{t \in T} \sum_{\gamma \in T} |J_\gamma(x) - LJ_\gamma(x)|
\]

The penalty function

\[
\mathcal{E}(\theta) = \sum_{t \in T} \sum_{\gamma \in T} \left( HQ(\gamma)e^{-I(\gamma)} + P_0(\gamma) \right) |J_\theta(x) - LJ_\theta(x)|
\]

is related to how far \( J_\theta \) is from the optimal value function.

**CROWDSOURCING**

- Need to label \( A \) items.
- Each item has soft label \( \mu_i \in [0, 1] \)
- Guess if \( \hat{\mu}_i \geq \frac{1}{2} \) for as many \( i \) as we can
- For \( t = 1, \ldots, T \):
  - Pick \( i \in \{1, \ldots, A\} \)
  - Recieve \( X_i \sim \text{Bern}(\mu_i) \)
- Use Beta prior \( \Rightarrow \text{MDP dynamics equivalent to Bayesian updates} \)
- \( P_0 \) limits prior
- Need to label \( \mu_i \) fewer samples for same accuracy

**EXPERIMENTAL RESULTS**

*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, and the difference grows with \( B \).*