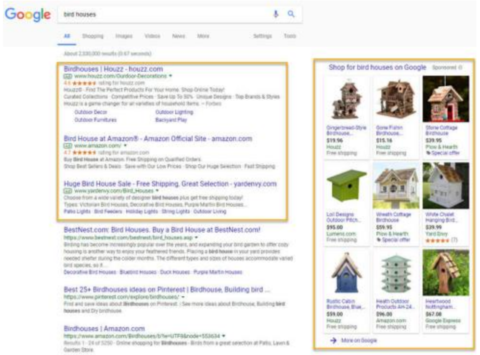
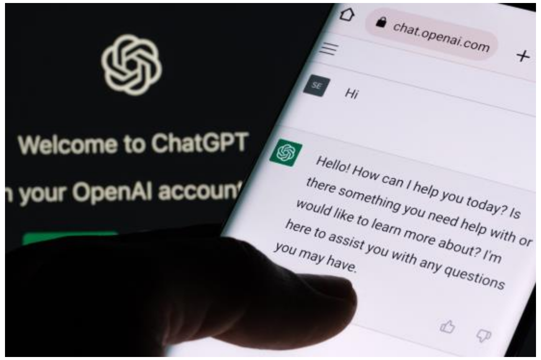
Posterior Sampling with Delayed Feedback for Reinforcement Learning with Linear Function Approximation

Nikki Lijing Kuang, Ming Yin*, Mengdi Wang, Yu-Xiang Wang, Yi-An Ma*

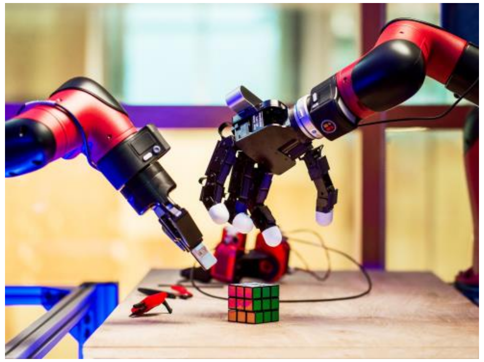
Sequential Decision Making



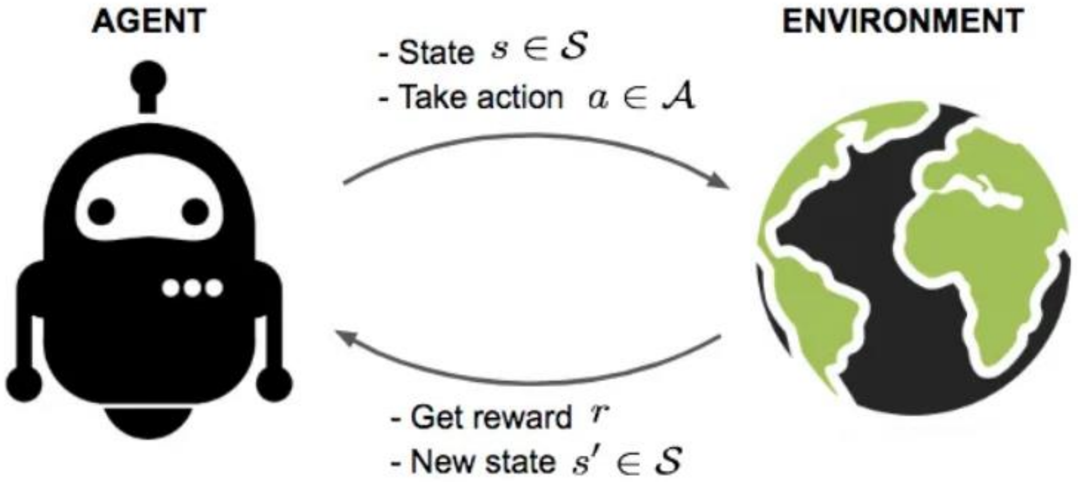
Web search



LLMs with RLHF


Robotics

Online recommendation



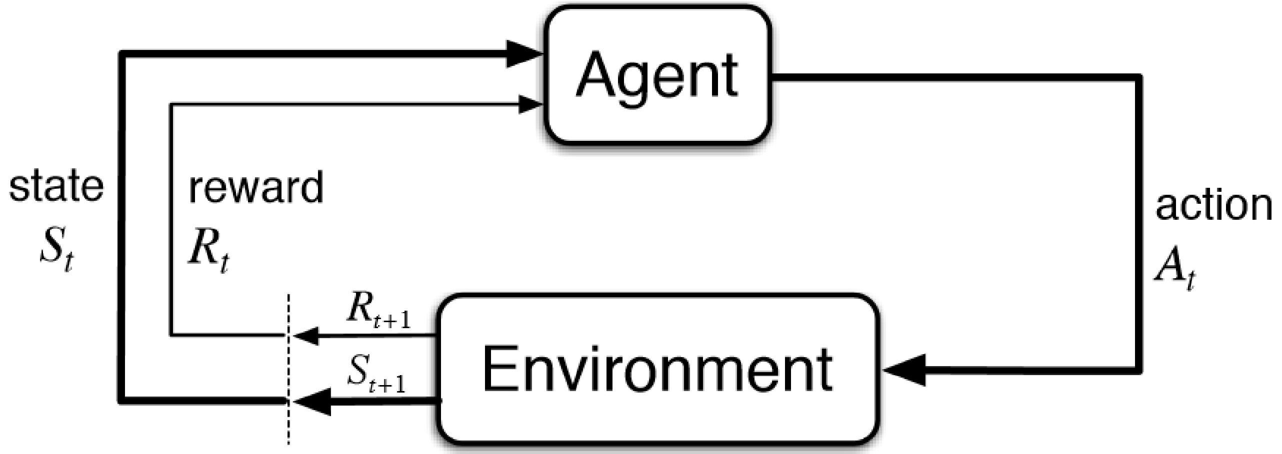

Autonomous Vehicles

RL with Function Approximation

- Empirical success of RL requires function approximation to handle high-dimensional spaces
- Collecting real-world data can be expensive
- Sample-efficient algorithms for the agent to learn using limited amount of samples

Limited Feedback Availability

- **Common assumptions**
  - Real-time communication
  - Feedback is observed immediately upon taking an action
  - **Unrealistic!**



Limited Feedback Availability

- **Reality**
  - Delayed Feedback
    - Robot teleoperation: delay due to signal transmission
    - Clinical trails: effectiveness of treatments can only be determined at a deferred time frame

Clinical trials

Robot teleoperation

Practical Requirement

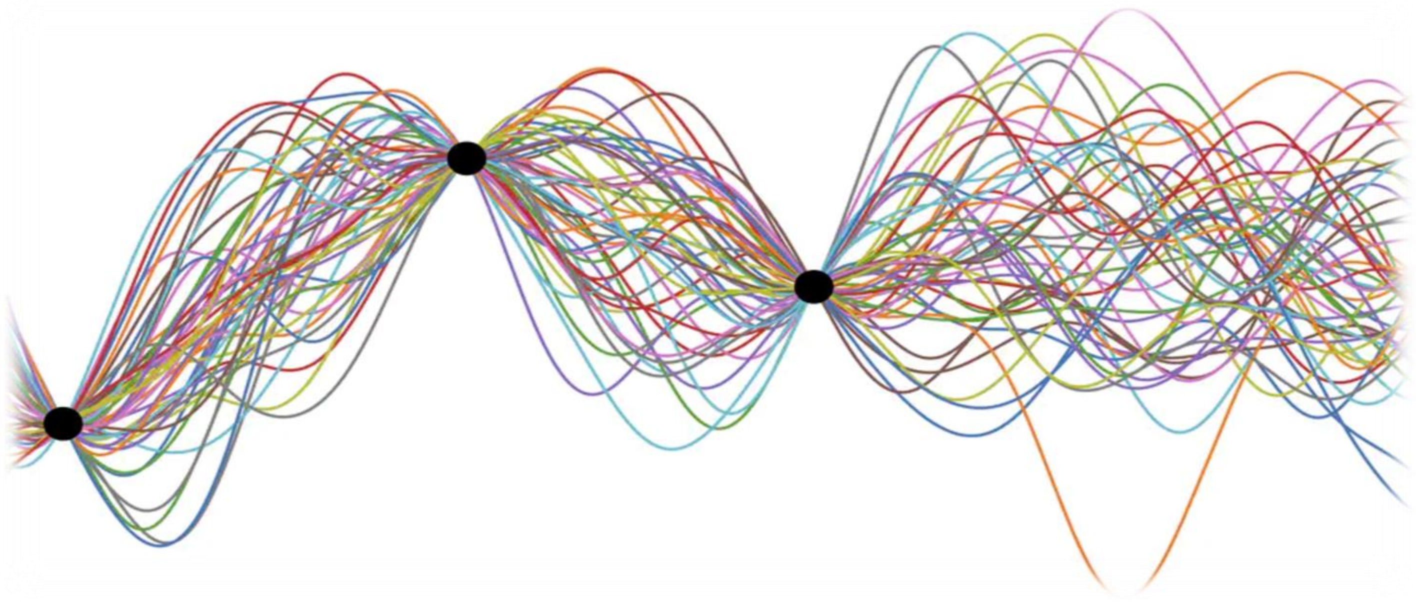
- Computationally efficient algorithms
- Statistically efficient algorithms
- Easy to deploy
- Resilient to delays
- Effective learning with least communication

Computation efficiency problem:
Can we design computationally efficient and practical algorithms?

Sample efficiency problem:
How to obtain statistically accurate algorithms with the least number of samples?

Posterior Sampling (PS)

- A randomized Bayesian algorithm
- Extends Thompson sampling (TS) to RL
- Selects an action according to its posterior probability of being the best
- Bears greater robustness in the presence of delays



Overview

- **TLDR**
  - Provide the first analysis for the class of PS algorithms to handle delayed feedback in RL
- **Contributions**
  - Introduce two novel value-based algorithms for linear MDPs under unknown stochastic delayed feedback
    - Delayed Posterior Sampling Value Iteration (Delayed-PSVI)
    - Delayed Langevin Posterior Sampling Value Iteration (Delayed-LPSVI)
    - Both algorithms achieve a high-probability worst-case regret of $O(\sqrt{d^3H^3T} + d^2H^2\mathbb{E}[\tau])$
    - Delayed-LPSVI reduces the computational complexity of Delayed-PSVI from $\tilde{O}(d^3HK)$ to $\tilde{O}(dHK)$

Comparison

- **Contributions**
  - Regret bounds in linear bandits and episodic MDPs under stochastic delay
  - Our algorithms
    - Achieve the optimal dependence on the parameters d and T under the class of PS algorithms
    - Recover the best-available frequentist regret as in non-delayed settings

Algorithms	Setting	Exploration	Worst-case Regret	Computation
[28]	Linear Bandits	UCB	$\widetilde{O}(d\sqrt{T} + d^{3/2}\mathbb{E}[\tau])$	Confidence set optimization
[29]	Tabular MDPs	UCB	$\widetilde{O}(\sqrt{SAH^3T} + S^2AH^3\mathbb{E}[\tau])$	Active update
[68]	Linear MDPs	UCB	$\widetilde{O}(\sqrt{d^3H^3T} + dH^2\mathbb{E}[\tau])$	Multi-batch reduction
[40]	Adversarial MDPs	UCB	$\widetilde{O}(H^2S\sqrt{AK} + H^{3/2}\sqrt{S\sum_{k=1}^{K}\tau_k})$	Confidence set optimization
Delayed-PSVI (Thm 1)	Linear MDPs	PS	$\widetilde{O}(\sqrt{d^3H^3T} + d^2H^2\mathbb{E}[\tau])$	$O((d^3 + Md)HK)$
Delayed-LPSVI (Thm 2)	Linear MDPs	PS	$\widetilde{O}(\sqrt{d^3H^3T} + d^2H^2\mathbb{E}[\tau])$	$O((N + d)MHK)$
Delayed-PSLB (Cor 2)	Linear Bandits	PS	$\widetilde{O}(\sqrt{d^3T} + d^2\mathbb{E}[\tau])$	$O((N + d)MK)$
UCB Lower bound [27]	Linear MDPs	UCB	$\Omega(dH\sqrt{T})$	—
PS Lower bound [24]	Linear Bandits	PS	$\Omega(\sqrt{d^3T})$	—

RL with Linear Function Approximation

- Finite-horizon episodic setting, time-inhomogeneous
- Both the transition dynamics P and reward function are linear in the feature map
- Action-value functions are always linear in the feature map

Definition 1 (Linear MDPs [66, 35]). *Suppose there exists a known feature map $\phi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^d$ that encodes each state-action pair into a d-dimensional feature vector. An MDP is a linear MDP*[3] *if for any time step $h \in [H]$, $\forall (s,a) \in \mathcal{S} \times \mathcal{A}$, both the transition dynamics $\mathbb{P}$ and reward function r are linear in ϕ:*

$$\mathbb{P}_h(\cdot|s,a) = \phi(s,a)^{\mathrm{T}} \mu_h(\cdot), \qquad r_h(s,a) = \phi(s,a)^{\mathrm{T}} \theta_h, \tag{1}$$

where $\mu_h : \mathcal{S} \to \mathbb{R}^d$ contains d unknown probability measures over $\mathcal{S}$, and $\theta_h \in \mathbb{R}^d$. Furthermore, we assume that $\forall (s,a) \in \mathcal{S} \times \mathcal{A}$, $\|\phi(s,a)\| \leq 1$, and $\forall h \in [H]$, $\|\theta_h\| \leq \sqrt{d}$, $\left\|\int_{\mathcal{S}} \mathrm{d}\mu_h(s')\right\| \leq \sqrt{d}$, where $\|\cdot\|$ denotes the Euclidean norm.

Performance Metric: worst-case Regret

- The goal of the learner: maximize the cumulative rewards / minimize the worst-case regret
- Worst-case regret:

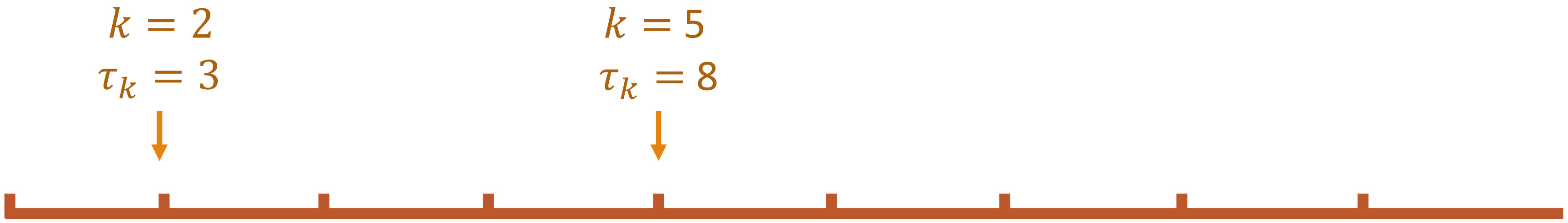
$$R(T) = \sum_{k=1}^{K} V_1^*(s_1^k) - V_1^{\pi_k}(s_1^k).$$

Episodic Delayed Feedback Model

- Consider stochastic delays across episodes
- Trajectory of each episode is not immediately observable

Definition 2 (Episodic Delayed Feedback). *In each episode $k \in [K]$, the execution of a fixed policy π^k generates a trajectory $\{s_h^k, a_h^k, r_h^k, s_{h+1}^k\}_{h \in [H]}$. Such trajectory information is called the feedback of episode k. Let τ_k represent the random delay between the rollout completion of episode k and the time point at which its feedback becomes observable.*

- Feedback of episode k becomes observable at the onset of the $(k + \tau_k)$-th episode
- Assumption: sub-exponential delays



Noisy Value Iteration

- Noisy value iteration via posterior sampling
- Consider a probability model $p(x \mid \theta)$ with a d-dimensional latent variable θ.
- The goal is to estimate the latent variable θ by inferring its posterior:

$$\underbrace{\boxed{p(\theta \mid x)}}_{\text{Posterior}} = \frac{\lambda(\theta) \cdot p(x \mid \theta)}{p(x)}$$

$$\propto \underbrace{\boxed{\lambda(\theta)}}_{\text{Prior}} \underbrace{\boxed{p(x \mid \theta)}}_{\text{Likelihood}}$$

- Posterior is often computationally intractable: $p(x) = \int \lambda(\theta)p(x \mid \theta)d\theta$

Delayed Posterior Sampling Value Iteration

- Not to maintain an exact posterior, but to inject randomness for efficient exploration
- Parameterize Q-function with parameter $w \in \mathbb{R}^d$:

$$\widetilde{Q}(s,a) = \phi(s,a)^{\mathrm{T}} w$$

$$p(w|\mathcal{D}, \boldsymbol{y}) \propto \exp(-L(w, \boldsymbol{y}, \mathcal{D})) p_0(w)$$

- Posteriors:

$$p(w_h^k|\mathcal{D}_h, \boldsymbol{y}_h) \propto \mathcal{N}\left((\Omega_h^k)^{-1}\Phi_h \boldsymbol{y}_h^{\mathrm{T}}, (\Omega_h^k)^{-1}\right)$$

$$\Omega_h^k := \Phi_h \Phi_h^{\mathrm{T}} + \lambda I_d \text{ and } \Phi_h = [\phi(s_h^1, a_h^1), \phi(s_h^2, a_h^2), \ldots, \phi(s_h^{k-1}, a_h^{k-1})]$$

- Approximates the solution of Bellman optimality equation via the least-square ridge regression

$$\widehat{w}_h^k = \mathrm{argmin}_w \sum_{\tau=1}^{k-1} (\phi(s_h^\tau, a_h^\tau)^{\mathrm{T}} w - (r + \max \bar{Q}_h^k))^2 + \lambda I_d$$

Delayed Posterior Sampling Value Iteration

Algorithm 1: Delayed Posterior Sampling Value Iteration (Delayed-PSVI)

Input: priors $p_0(w_h^k) \leftarrow \mathcal{N}(0, \lambda I)$, scaling factor ν, multi-round paramter M, hyper parameters λ and σ^2.

Initialization: $\forall k, h$, $\widetilde{Q}_{H+1}^k(\cdot,\cdot), \widetilde{V}_{H+1}(\cdot,\cdot), \widetilde{V}_h(\cdot,\cdot) \leftarrow 0$, $\mathcal{D}_h \leftarrow \emptyset$.

for *episode* $k = 1, \ldots, K$ **do**
- Sample initial state s_1^k
- **for** *time step* $h = H, \ldots, 1$ **do**
  - $\boldsymbol{y_h} \leftarrow [y_h^1, \ldots, y_h^{k-1}]$, with $y_h^\tau \leftarrow \mathbb{1}_{\tau,k-1} \cdot [r_h^\tau + \widetilde{V}_{h+1}(s_{h+1}^\tau)]$
  - $\Phi_h \leftarrow [\phi^1, \phi^2, \ldots, \phi^{k-1}]$ with $\phi^\tau = \mathbb{1}_{\tau,k-1} \cdot \phi(s_h^\tau, a_h^\tau)$
  - $\Omega_h^k \leftarrow \sigma^{-2}\Phi_h\Phi_h^{\mathrm{T}} + \lambda I$, $\widehat{w}_h^k \leftarrow \sigma^{-2}(\Omega_h^k)^{-1}\Phi_h \boldsymbol{y_h}^{\mathrm{T}}$
  - $p(w_h^k \mid \mathcal{D}_h, \boldsymbol{y_h}) \leftarrow \mathcal{N}(\widehat{w}_h^k, \nu^2 \cdot (\Omega_h^k)^{-1})$
  - **for** $m = 1, \ldots, M$ **do**
    - Sample $\widetilde{w}_h^{k,m} \sim p(w_h^k \mid \mathcal{D}_h, \boldsymbol{y_h})$
    - $\widetilde{Q}_h^{k,m}(\cdot,\cdot) \leftarrow \phi(\cdot,\cdot)^{\mathrm{T}}\widetilde{w}_h^{k,m}$
  - Update $\widetilde{Q}_h^k(\cdot,\cdot) \leftarrow \max_m \widetilde{Q}_h^{k,m}$
  - $\widetilde{V}_h(\cdot,\cdot) \leftarrow \max_a \min\{\widetilde{Q}_h^k(\cdot,a), H-h+1\}$
  - Update $\pi_h^k(\cdot) \leftarrow \mathrm{argmax}_{a\in\mathcal{A}} \min\{\widetilde{Q}_h^k(\cdot,a), H-h+1\}$
- **for** *time step* $h = 1, \ldots, H$ **do**
  - Choose action $a_h^k = \pi_h^k(s_h^k)$
  - Collect trajectory observations $\mathcal{D}_h \leftarrow \mathcal{D}_h \cup \{(s_h^k, a_h^k, r_h^k, s_{h+1}^k)\}$
- /* Feedback generated in episode k cannot be immediately observed in the presence of delay */

Noisy value iteration

Optimism: multi-round sampling

Kuang, Yin, Wang, Wang, and Ma, "Posterior Sampling with Delayed Feedback for Reinforcement Learning with Linear Function Approximation." NeurIPS 2023

Performance Guarantee

- **Worst-case regret guarantee**
- **Recover the best-available frequentist regret** $O(\sqrt{d^3H^3T})$ **as in non-delayed linear MDPs**
- **Computational complexity:** $O((d^3 + M\,d)HK)$

Theorem 1. *Suppose delays satisfy Assumption 1. In any episodic linear MDP with time horizon* $T = KH$*, where* K *is the total number of episodes, for any* $0 < \delta < 1$*, let* $\lambda = 1$*,* $\sigma^2 = 1$*,* $M = \log(4HK/\delta)/\log(64/63)$ *and* $\nu = C_{\delta/4} \approx \widetilde{O}(\sqrt{dMH^2})$ *(*$C_{\delta/4}$ *in Lemma B.10). Then with probability at least* $1-\delta$*, there exists some absolute constants* $c, c', c'' > 0$ *such that the regret of Delayed-PSVI (Algorithm 1) satisfies:*

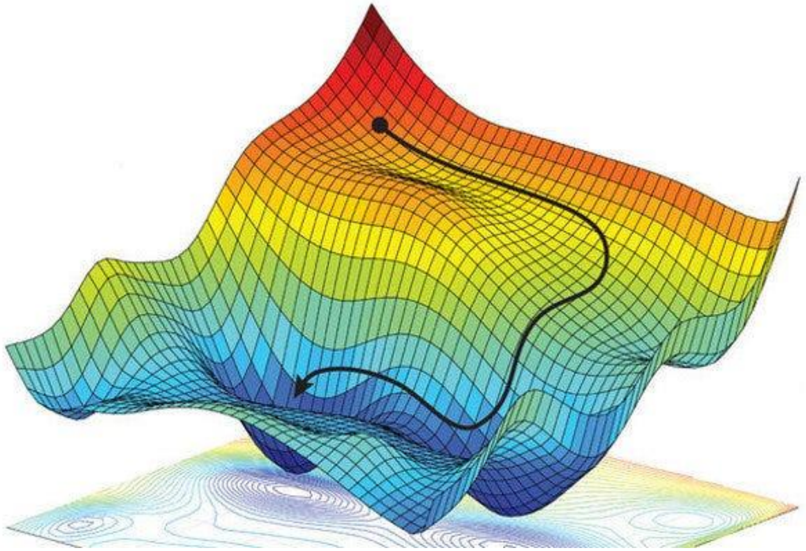
$$R(T) \leq c\sqrt{d^3H^3T\iota} + c'd^2H^2\mathbb{E}[\tau]\iota + c''\iota.$$

Here ι *is a Polylog term of* H, d, K, δ.

Kuang, Yin, Wang, Wang, and Ma, "Posterior Sampling with Delayed Feedback for Reinforcement Learning with Linear Function Approximation." NeurIPS 2023

Estimation of Complex Probabilistic Model

- Posteriors are often computationally intractable
- Delayed-PSVI is not sufficiently efficient in high-dimensional settings
- Resort to approximate Bayesian inference methods



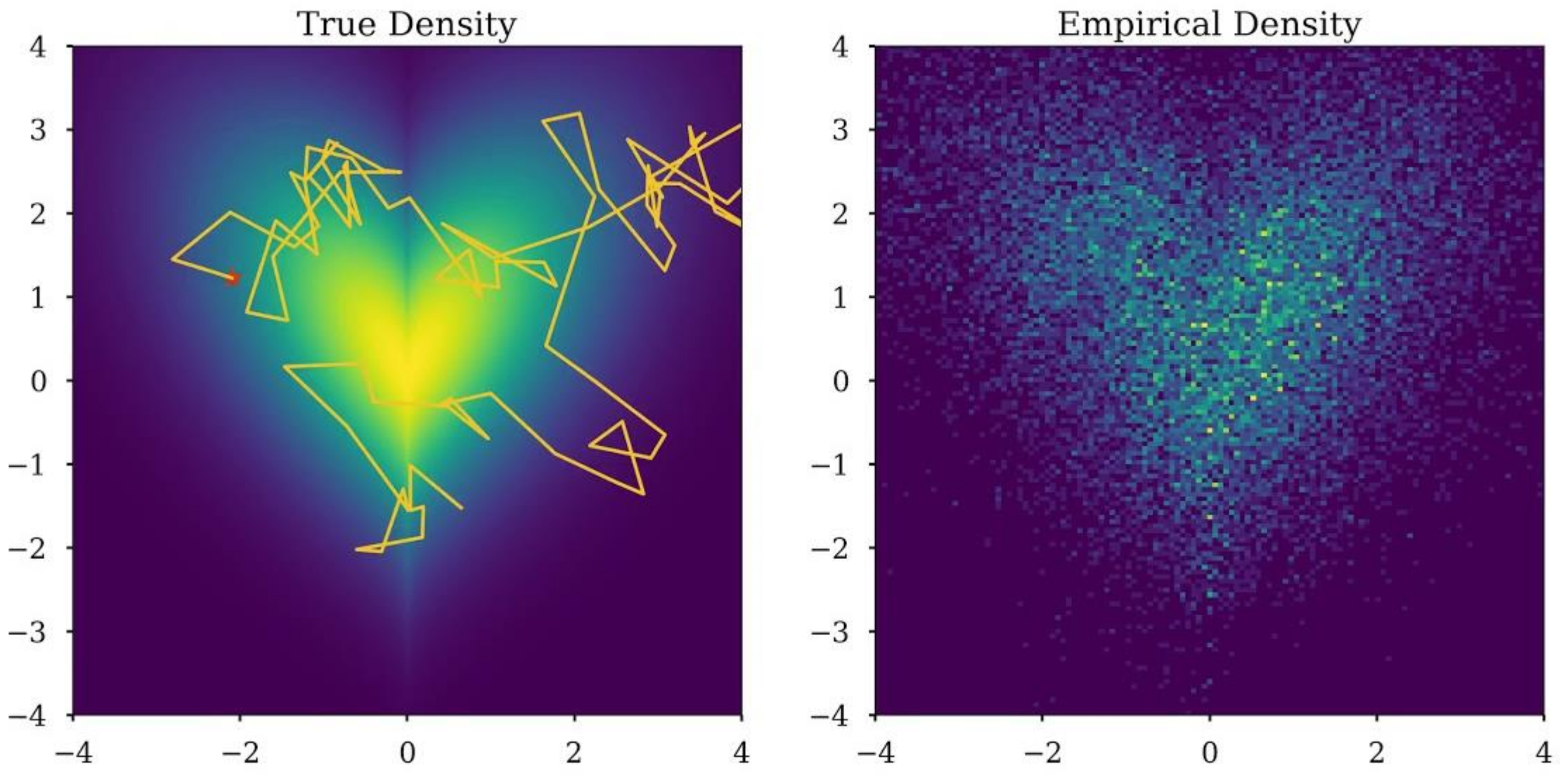
How to sample from unknown non-conjugate distributions?

Approximate Bayesian Inference

- Bootstrapping
- Ensemble Methods
- Variational Inference (VI)
- Markov Chain Monte Carlo (MCMC)

Langevin Monte Carlo

- A class of gradient-based MCMC methods, tailored for large-scale online learning



Langevin Monte Carlo

- Efficient in large-scale online learning
- Perform gradient optimization over data D
- Euler discretization of the Langevin stochastic differential equation (SDE):

$$\mathrm{d}\boldsymbol{w}(t) = -\nabla L(\boldsymbol{w}(t))\mathrm{d}t + \sqrt{2\beta^{-1}}\,\mathrm{d}\boldsymbol{B}(t)$$

- Update rule: noisy gradient update

$$\theta_t \leftarrow \theta_{t-1} - \eta \nabla U(\theta_{t-1}) + \sqrt{2\eta\gamma}\varepsilon_t, \qquad where\ \ \varepsilon_t \sim \mathcal{N}(0, I_d)$$

Delayed Langevin PSVI

- Noisy value iteration via Langevin posterior sampling

Algorithm 2: Delayed Langevin Posterior Sampling Value Iteration (Delayed-LPSVI)

Input: w_0, η_k, N_k, γ and rounds M, λ. Delayed loss L_h^k as (5).

```
Initialization: ∀k ∈ [K], h ∈ [H], Q̃^k_{H+1}(·,·) ← 0, Ṽ^k_{H+1}(·,·) ← 0, Ṽ^0_h(·,·) ← 0
for episode k = 1, ..., K do
    Sample initial state s^k_1
    for time step h = H, ..., 1 do
        for m = 1, ..., M do
            w̃^{k,m}_h ← LMC(L^k_h, w_0, η_k, N_k, γ)        //LMC is given by Algorithm 3
            Q̃^{k,m}_h(·,·) ← φ(·)^T w̃^{k,m}_h
        Update Q̃^k_h(·,·) ← max_m Q̃^{k,m}_h
        Ṽ^k_h(·,·) ← max_a min{Q̃^k_h(·,a), H − h + 1}
        Update policy π^k_h(·) ← argmax_{a∈A} min{Q̃^k_h(·,a), H − h + 1}
    for time step h = 1, ..., H do
        Choose action a^k_h = π^k_h(s^k_h)
        Collect trajectory observations D_h ← D_h ∪ {(s^k_h, a^k_h, r^k_h, s^k_{h+1})}
    /* Feedback generated in episode k cannot be immediately observed in the presence of delay */
```

Optimism: multi-round sampling

Algorithm 3: Langevin Monte Carlo $LMC(\mathcal{L}, w_0, \eta, N, \gamma)$

```
for t = 1 ... N − 1 do
    Draw ε_t ~ N(0, I_d)
    w_t ← w_{t−1} − η∇L(w_{t−1}) + sqrt(2ηγ) ε_t
Output: w_N
```

Worst-case Regret Guarantee

- **Worst-case regret guarantee**
- **Recover the best-available frequentist regret $O(\sqrt{d^3H^3T})$ as in non-delayed linear MDPs**
- **Computational complexity: $O((N+d)HK)$**

Theorem 2. *Suppose delays satisfy Assumption 1. In any episodic linear MDP with time horizon $T = KH$, where K is the total number of episodes and H is the fixed episode length, for any $0 < \delta < 1$, let $\lambda = 1$, $N_k = \max\{\log(\frac{32H^2(K+\lambda)dk}{\gamma\lambda} + 1)/[2\log(1/(1-\frac{1}{2\kappa_h}))],$ $\frac{\log 2}{2\log(1/(1-\frac{1}{2\kappa_h}))}, \log(\frac{4HK^3}{\sqrt{\lambda/dK}})/\log(1/(1-\frac{1}{2\kappa_h}))\}$, $\eta_k = \frac{1}{4\lambda_{\max}(\Omega_h^k)}$, $\gamma = 16C^2_{\delta/4} \approx \widetilde{O}(dMH^2)$,*

$w_0 = \mathbf{0}$ and $M = \log(4HK/\delta)/\log(64/63)$. Then with probability at least $1-\delta$, there exists some absolute constants $c, c', c'' > 0$ such that the regret of Algorithm 2 satisfies:

$$R(T) \le c\sqrt{d^3H^3T\iota} + c'd^2H^2\mathbb{E}[\tau]\iota + c''\iota.$$

Here ι is a Polylog term of H, d, K, δ and C_δ is defined in Lemma C.9.

Kuang, Yin, Wang, Wang, and Ma, "Posterior Sampling with Delayed Feedback for Reinforcement Learning with Linear Function Approximation." NeurIPS 2023

Experiments

- **Sub-exponential delays and long-tail delays:**
  - Multinomial delay
  - Poisson delay
  - Long-tail Pareto delay

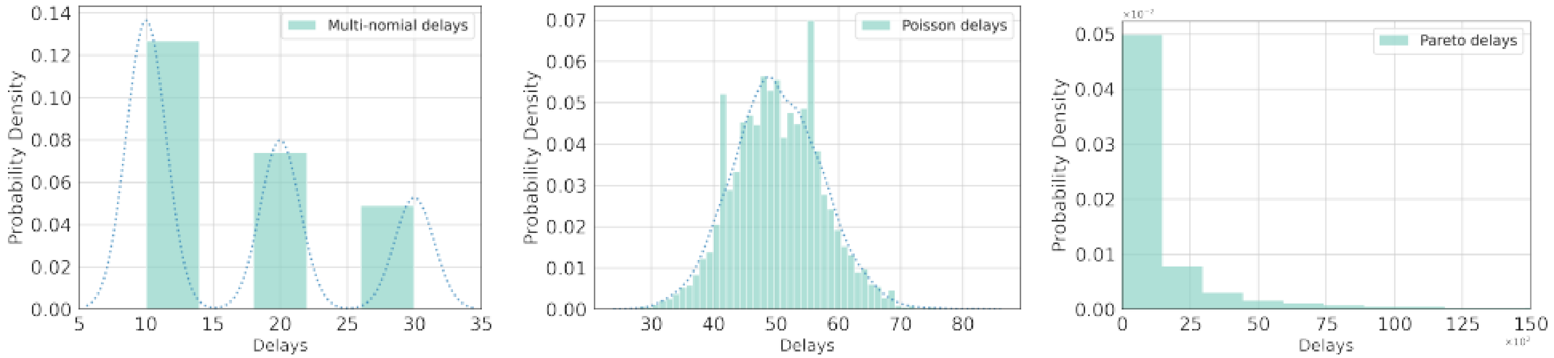


Figure 2: Empirical distributions of three types of delays. (a) Multinomial delays with delay categories $\{10, 20, 30\}$. (b) Poisson delays with rate $\mathbb{E}[\tau] = 50$. (c) Long-tail Pareto delays with shape 1.0, scale 500. The first two types of delays are well-behaved and decay exponentially fast, while pareto delays are heavy-tailed.

Kuang, Yin, Wang, Wang, and Ma, "Posterior Sampling with Delayed Feedback for Reinforcement Learning with Linear Function Approximation." NeurIPS 2023

Experiments

- **Performance Comparison**

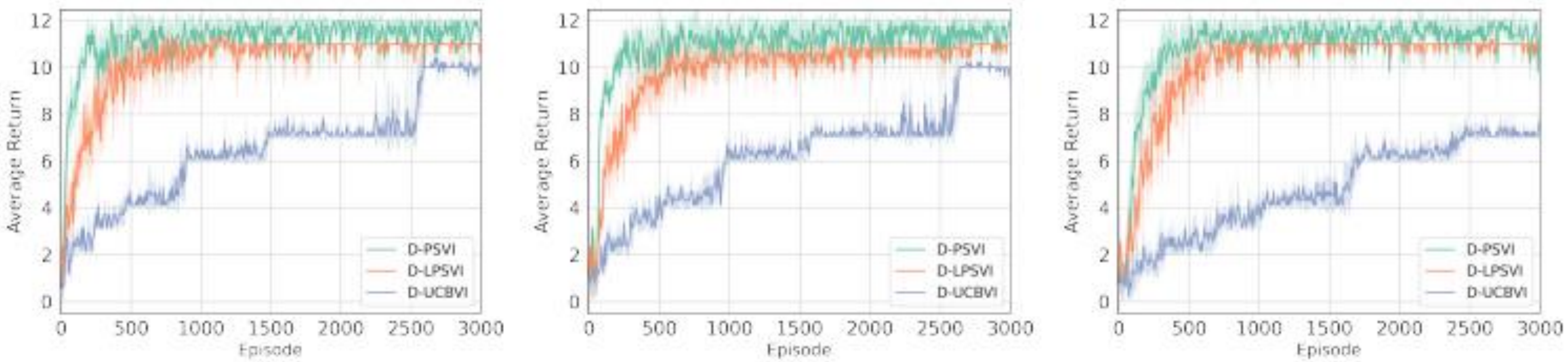


Figure 1: Left:(a) Multinomial delay with delay categories $\{10, 20, 30\}$. (b) Poisson delay with rate $\mathbb{E}[\tau] = 50$. (c) Long-tail Pareto delay with shape 1.0, scale 500. Results are reported over 10 experiments. Delayed-PSVI and Delayed-LPSVI demonstrate robust performance under both well-behaved and long-tail delays.

	Multinomial Delay $(10, 20, 30)$	Poisson Delay $(\mathbb{E}[\tau] = 50)$	Pareto Delay (Shape 1.0, Scale 500)
Delayed-PSVI ($\sigma = 0.1$)	11.53 ± 0.76	11.48 ± 0.81	11.53 ± 0.74
Delayed-LPSVI ($c_\eta = 0.5$)	11.56 ± 0.48	11.37 ± 0.48	10.98 ± 0.40
Delayed-UCBVI ($c_\beta = 0.1$)	10.61 ± 0.76	10.54 ± 0.81	7.20 ± 0.38

Table 2: Average return achieved by Delayed-PSVI, Delayed-LPSVI and Delayed-UCBVI upon convergence under different delays. Environment setup: $|\mathcal{S}| = 2$, $|\mathcal{A}| = 20$, $d = 10$, $H = 20$. Optimal average return is $V_1^*(s_1) = 11.96$. Results are obtained over 10 experiments.

Experiments

- **Computational overhead**
- **Measured by number of episodes to converge**

	$\lvert\mathcal{S}\rvert\lvert\mathcal{A}\rvert = 20$	$\lvert\mathcal{S}\rvert\lvert\mathcal{A}\rvert = 40$	$\lvert\mathcal{S}\rvert\lvert\mathcal{A}\rvert = 100$	$\lvert\mathcal{S}\rvert\lvert\mathcal{A}\rvert = 200$
Delayed-PSVI ($\sigma = 0.3$)	1418	1290	1669	2633
Delayed-PSVI ($\sigma = 0.2$)	531	1114	1323	826
Delayed-PSVI ($\sigma = 0.1$)	391	571	650	709
Delayed-LPSVI ($c_\eta = 0.5$)	293	246	517	566
Delayed-UCBVI ($c_\beta = 0.1$)	3205	2713	3351	3694

Table 3: Number of episodes for each method to achieve its highest expected return. Different synthetic environments are examined with varied $\lvert\mathcal{S}\rvert$ and $\lvert\mathcal{A}\rvert$. Optimal average return is $V_1^*(s_1) = 11.96$ for all environments ($d = 10$, $H = 20$). Results are obtained over 10 experiments with Poisson delays ($\mathbb{E}[\tau] = 50$).

Kuang, Yin, Wang, Wang, and Ma, "Posterior Sampling with Delayed Feedback for Reinforcement Learning with Linear Function Approximation." NeurIPS 2023

Conclusions

- Study posterior sampling with episodic delayed feedback in linear MDPs
- Introduce two novel value-based algorithms: Delayed-PSVI and Delayed-LPSVI
- Both algorithms achieve a high-probability worst-case regret of $O(\sqrt{d^3H^3T} + d^2H^2\mathbb{E}[\tau])$
- By incorporating LMC for approximate sampling, Delayed-LPSVI reduces the computational cost by $\tilde{O}(d^2)$ while maintaining the same order of regret
- Empirical evaluation demonstrates the effectiveness of our algorithms over UCB-based methods

Thank you!