# A Safe Bayesian Learning Algorithm for Constrained MDPs with Bounded Constraint Violation

**Krishna C Kalagarla**
Amazon

**Rahul Jain**
USC and Google Deepmind

**Pierluigi Nuzzo**
UC Berkeley

## Abstract

Constrained Markov decision process (CMDP) models are increasingly important in many applications with multiple objectives. When the model is unknown and must be learned online, it is desirable to ensure that the constraint is met, or at least the violation is bounded with time. In the recent literature, progress has been made on this very challenging problem, but with either unsatisfactory assumptions, such as the knowledge of a safe policy, or high cumulative regret. We propose the Safe-PSRL (posterior sampling-based RL) algorithm that does not need such assumptions and yet performs very well, both in terms of theoretical regret bounds as well as empirically. The algorithm efficiently trades-off exploration and exploitation using posterior sampling-based exploration, and yet provably suffers only bounded constraint violation using carefully-crafted pessimism. We establish a sub-linear $\tilde{\mathcal{O}}\left(H^{2.5}\sqrt{|\mathcal{S}|^2|\mathcal{A}|K}\right)$ upper bound on the Bayesian objective regret along with a *bounded*, i.e., $\tilde{\mathcal{O}}\left(1\right)$ constraint-violation regret over $K$ episodes for an $|\mathcal{S}|$-state, $|\mathcal{A}|$-action, and horizon $H$ CMDP, which improves over state-of-the-art algorithms for the same setting.

## 1 INTRODUCTION

In many settings, active or online learning is an important way to learn effectively and efficiently. For example, fine-tuning in large language models (LLMs) or diffusion models is often done with human feedback (HF) which is slow and expensive. Incorporation of active or online learning methods could significantly improve the effectiveness and cost-efficiency of HF collection. Furthermore, the generated outputs are required to be "safe", i.e., non-offensive, non-vulgar, etc. at inference time. Even during HF data collection for fine-tuning, it is desirable not to expose raters to offensive language or images any more than necessary. The need for safety-constrained online reinforcement learning is also well motivated for many cyber-physical applications, e.g., when a robot is learning to navigate an office environment as efficiently and safely as possible upon deployment.

Constrained Markov decision process (CMDP) models [Altman, 1999] provide a suitable modeling framework to incorporate multi-objective sequential decision-making problems. Such models, however, do not satisfy Bellman's principle of optimality, and alternative approaches, e.g., convex analytic methods [Hordijk and Kallenberg, 1979, Borkar, 1988] are used instead. And while online reinforcement learning is well-developed for unconstrained MDPs, similar developments for CMDPs are mostly lacking. In fact, online learning when the model is unknown, while maintaining safety, i.e., satisfying some constraints, has long seemed to be a near-impossible challenge. And yet, the need for such algorithms has never been greater.

Significant advancements have been made toward addressing this issue in the past few years. Early results in this area [Efroni et al., 2020, Brantley et al., 2020] include regret-optimal online RL algorithms. These algorithms were demonstrably capable of achieving sublinear bounds in both objective regret and constraint violations. Concurrently, or as a complement, the first PAC (probably approximately correct) algorithms [Kalagarla et al., 2021, HasanzadeZonuzy et al., 2021], which offer sample complexity bounds for safe learning, were also developed. Following these, a series of online algorithms [Bura et al., 2022, Wei et al., 2022, Liu et al., 2021a, Ghosh et al., 2022] not only reached theoretical sublinear objective regret but also demonstrated provably bounded constraint violations. Regrettably, despite their impressive theoretical bounds, the practi-

Proceedings of the 28th International Conference on Artificial Intelligence and Statistics (AISTATS) 2025, Mai Khao, Thailand. PMLR: Volume 258. 

cal performance of these algorithms often falls below expectations. This paper seeks to narrow the divide between theoretical guarantees and empirical performance.

We consider the online learning problem for an episodic Constrained MDP model. The transition model is unknown to the learning agent. We consider a tabular setting for the sake of simplicity. The learning agent's performance, as is usual in such settings, is measured in terms of *cumulative expected regret*, a measure of the difference between the cumulative reward of the learning agent and that of the optimal policy. This online learning problem thus leads to the well-known *exploration-exploitation* tradeoff. There are two main approaches to balance this tradeoff in such a way that the learning regret is minimized. The first is a (non-Bayesian) optimism (OPT)-based approaches [Lai and Robbins, 1985, Jaksch et al., 2010, Azar et al., 2017, Jin et al., 2018] that roughly work by introducing an exploration bonus term based on optimistic (or upper bound) on an optimality term. In RL settings, this often translates to optimizing over uncertainty sets for the unknown transition model, a computationally challenging task. An alternative (Bayesian) approach is based on Thompson [Thompson, 1933], or posterior sampling (PS) [Russo et al., 2018] that maintains a posterior distribution over the unknown model parameters by using the observations during learning. The procedure then is pretty general and straightforward: the learning agent samples a model from the posterior distribution, and computes an optimal policy for it, which is then used for decision making in the next episode. Remarkably, such a procedure is able to efficiently balance the *exploration-exploitation* tradeoff. While OPT-based and PS algorithms often have similar theoretical performance bounds (albeit for slightly different notions of regret), PS algorithms usually demonstrate superior empirical performance [Osband et al., 2013, Ouyang et al., 2017]. This makes such an approach more desirable for designing online learning algorithms in safety-constrained settings.

In this paper, we introduce the `Safe-PSRL` algorithm for online learning in finite-horizon CMDPs. Our algorithm uses the primal-dual approach wherein the primal part performs unconstrained MDP planning with a sampled transition probability, and the dual part updates the Lagrangian variable to track the constraint violation. The algorithm is much simpler than some recent state-of-the-art (optimism-based) algorithms and yet has better theoretical bounds, and demonstrably superior empirical performance in both objective regret minimization as well as constraint-violation bounds.

We achieve bounded constraint-violation regret by using the idea of *pessimism* [Liu et al., 2021b]. "Pessimism" is achieved by tightening the constraint bound in a systematic manner. By appropriately balancing *posterior sampling-based exploration* and *pessimism-based safe learning*, we show that the `Safe-PSRL` algorithm achieves sub-linear $\tilde{\mathcal{O}}\left(H^{2.5}\sqrt{|\mathcal{S}|^2|\mathcal{A}|K}\right)$ reward regret while achieving bounded, i.e., $\tilde{\mathcal{O}}(1)$-constraint violation regret for an $|\mathcal{S}|$-state, $|\mathcal{A}|$-action, and an $H$ episode length CMDP over $K$ number of episodes. In this paper, we use the notion of Bayesian regret, which applies naturally to a Bayesian algorithm. Importantly, however, our algorithm exhibits a superior empirical performance in the frequentist sense when contrasted with comparable optimism-type algorithms with frequentist regret bounds.

The **main contributions** of this paper are as follows. *Algorithmic:* We introduce the first PS-based safe online learning algorithm for episodc CMDPs that achieves $\tilde{\mathcal{O}}\left(\sqrt{K}\right)$ objective reward regret with $\tilde{\mathcal{O}}(1)$ constraint violation regret. A strength of our algorithm is its simplicity, with a key novelty being a carefully-crafted pessimism-term that helps ensure bounded constraint-violation while not overly constraining exploration. We also do not need assumption of a known safe policy unlike in `DOPE` algorithm [Bura et al., 2022].

*Technical:* Our algorithm achieves an $O(S^{\frac{1}{2}})$-better regret compared to the SOTA `OptPess-PrimalDual` algorithm [Liu et al., 2021a] and similar regret to that of the `DOPE` algorithm (which assumes a known safe policy). Our theoretical analysis involves a novel decomposition which allows us to leverage posterior sampling regret analysis and Lyapunov-drift analysis for the dual variables. Since we address the same problem, some of our analysis is similar in spirit to that in [Liu et al., 2021a, Ouyang et al., 2017], and yet since our algorithm is of a different nature, several details are different: The parameter choices are different in our algorithm to leverage the lower regret bounds of posterior sampling. There is also a non-trivial difference in the decomposition of the reward regret terms (e.g., in Lemma 6) and the analysis of Lemma 4, due to the use of the posterior sampling property.

*Empirical:* Our `Safe-PSRL` algorithm is simpler and yet outperforms multiple state-of-the-art (SOTA) algorithms such as `OptPess-PrimalDual` and `DOPE` in empirical performance by a wide margin. Additionally, the `DOPE` algorithm needs knowledge of safe baseline policy, which we do not. We conjecture that the superior empirical performance of our algorithm vis-a-vis SOTA optimism-based algorithms is due to its use of posterior sampling, which wouldn't be surprising. We do use a weaker notion of (Bayesian) objective regret, as well as cumulative cost in constraint-violation (a notion used in most prior related work including the

seminal paper [Efroni et al., 2020]). However, our empirical results report that our algorithm has both better (non-Bayesian) frequentist regret and better cumulative constraint-violation regret.

**Related Work.** Posterior (or Thompson) sampling goes back to the work of [Thompson, 1933], but attracted less attention for several decades until empirical evidence [Chapelle and Li, 2011] showed its superior performance for online learning. Recently, it has been widely applied to various settings like multi-armed bandits [Kaufmann et al., 2012, Agrawal and Goyal, 2012, Agrawal and Goyal, 2013], MDPs [Osband et al., 2013, Osband and Van Roy, 2017, Ouyang et al., 2017] and POMDPs [Jafarnia-Jahromi et al., 2021c].

In the CMDP setting, several existing works [Efroni et al., 2020, Qiu et al., 2020] leverage optimism or posterior sampling to provide $\tilde{\mathcal{O}}(\sqrt{K})$ regret for the reward as well as the constraint objective, where $K$ is the number of episodes. Such an approach, however, can lead to a large number of constraint violations during learning, which is unacceptable during various safety-critical tasks such as driving or power distribution. Thus, the problem of an online RL algorithm with sublinear (ideally, $\tilde{\mathcal{O}}(\sqrt{K})$) reward regret and *bounded* constraint violation regret, while also demonstrating strong empirical performance, remains open.

Optimism-based algorithms have been widely used for efficient learning in CMDPs, e.g., in the setting of PAC performance guarantees for finite-horizon CMDPs [HasanzadeZonuzy et al., 2021, Kalagarla et al., 2021], or to provide regret bounds for CMDPs in the finite-horizon setting [Efroni et al., 2020, Brantley et al., 2020, Müller et al., 2024] and infinite-horizon average cost setting [Singh et al., 2020]. Policy gradient algorithms for CMDPs [Ding et al., 2020, Ding et al., 2021] have also been studied. However, these algorithms do not provide bounded or zero constraint violation guarantees.

Recently, some optimism-based approaches for *safe* learning with bounded or zero constraint violation guarantees have been proposed [Zheng and Ratliff, 2020, Chen et al., 2022, Liu et al., 2021a, Ghosh et al., 2022, Wei et al., 2022]. However, these approaches differ for at least one of the following reasons: they assume that the transition model is known, they only satisfy the constraint with high probability, they assume that a safe policy is available to the algorithm, e.g., in [Liu et al., 2021a, Bura et al., 2022], or they achieve 0 constraint regret for a large enough value of $K$ [Ghosh et al., 2022, Wei et al., 2022] without a constant bound independent of $K$.

The `OptPess-PrimalDual` algorithm in [Liu et al., 2021a] is the closest comparable algorithm to our `Safe-PSRL` algorithm. While the use of the posterior sampling principle for constrained RL problems is under-explored (despite the promise of better empirical performance), [Provodin et al., 2023, Agarwal et al., 2022] indeed introduced PSRL algorithms for CMDPs but for the average setting. Moreover, they only achieve a $\tilde{\mathcal{O}}(\sqrt{K})$ constraint violation regret which is worse than our $\tilde{\mathcal{O}}(1)$ bound.

## 2 PRELIMINARIES

**Finite-Horizon MDPs.** An episodic finite-horizon MDP [Puterman, 1994] can be formally defined by a tuple $\mathcal{M} = (\mathcal{S}, \mathcal{A}, H, s_1, p, r)$, where $\mathcal{S}$ and $\mathcal{A}$ denote the state and action spaces, respectively. In this setting, the agent interacts with the environment in episodes of fixed length $H$, with each episode starting with a random initial state denoted $s_1$. The non-stationary transition probability $p_h(s'|s,a)$ is the the probability of transitioning to state $s'$ on taking action $a$ at state $s$ at time $h \in [1:H]$ of the episode. The non-stationary reward obtained on taking action $a$ in state $s$ at time $h$ of an episode is denoted by a random variable $R_h(s,a) \in [0,1]$, with mean $r_h(s,a)$. We use $r$ as a shorthand to denote the mean reward vector $r_1, \ldots, r_H$. A non-stationary randomized policy $\pi = (\pi_1, \ldots, \pi_H) \in \Pi$ where $\pi_i : \mathcal{S} \to \Delta_{\mathcal{A}}$, maps a state to a probability simplex over the action space $\mathcal{A}$. The action $a_h$ at time $h$ at state $s_h$ is taken according to the policy $\pi$, $a_h \sim \pi_h(s_h)$. The value function of a non-stationary randomized policy $\pi$, $V_h^\pi(s; r, p)$ (when clear, $s, r$, and $p$ are omitted) at a state $s$ and time $h$ is defined as $V_h^\pi(s; r, p) := \mathbb{E}_\pi \left[\sum_{i=h}^{H} r_i(s_i, a_i) | s_h = s, p\right]$. We can always find an optimal non-stationary deterministic policy $\tilde{\pi}$ [Puterman, 1994] such that $V_h^{\tilde{\pi}}(s) = \tilde{V}_h(s) = \sup_\pi V_h^\pi(s)$. The optimal policy can be computed by using backward induction on the Bellman optimality equations [Puterman, 1994].

**Finite-Horizon Constrained MDPs.** A finite-horizon constrained MDP (CMDP) [Altman, 1999] is a finite-horizon MDP with a required upper bound on the expectation of a cost function, $\{c, \tau \in (0, H]\}$. The non-stationary cost obtained on taking action $a$ in state $s$ at time $h$ with respect to the constraint cost function is denoted by a random variable $C_h(s,a) \in [0,1]$, with mean $c_h(s,a)$. The total expected reward (cost) of an episode under policy $\pi$ with respect to the reward (cost) function $r$ ($c$) is the respective value function from the initial state $s_1$, i.e., $V_1^\pi(s_1; r, p)(V_1^\pi(s_1; c, p))$ (by definition). Our objective is to find a policy which maximizes the total expected objective reward under the constraint that the total expected constraint cost

is below a desired threshold.

$$\pi^* \in \underset{\pi\in\Pi}{\operatorname{argmax}} \quad V_1^{\pi}(s_1; r, p) \quad \text{s.t.} \quad V_1^{\pi}(c, p) \leq \tau. \tag{1}$$

The optimal value is denoted by $V^*(s_1; r, p) = V_1^{\pi^*}(s_1; r, p)$. Since a deterministic optimal policy may not exist, we consider $\Pi$, the class of all randomized policies [Altman, 1999]. Since the Bellman optimality equation does not hold due to the constraints, dynamic programming-based backward induction algorithms cannot be used to find an optimal policy. However, a linear programming approach can be given that will find an optimal policy [Altman, 1999].

## 3 THE LEARNING PROBLEM

We consider the setting where an agent interacts with a finite-horizon CMDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, H, s_1, p, r, \{c, \tau\})$ over multiple episodes of fixed length $H$, starting each episode from initial state $s_1$ and with stationary transition probability (i.e., $p_h = p, \forall h$). We employ the Bayesian framework and regard the transition probability $p$ as random with a prior distribution $\mu_1$. The transition probability is unknown to the learning agent. We consider finite-horizon CMDP with transition probability in the set $\Theta_{c_0}$ with the following property:

**Assumption 1.** *For all $\hat{p} \in \Theta_{c_0}$, there exists a policy $\pi_0^{\hat{p}}$ such that $V_1^{\pi_0^{\hat{p}}}(c, \hat{p}) \leq c_0 < \tau$.*

Moreover, we assume that the support of the prior distribution $\mu_1$ is a subset of $\Theta_{c_0}$ and $c_0$ is known. Assumption 1 is not only reasonable but also necessary to ensure that the problem is feasible. Without loss of generality, we assume that the reward and cost functions $r$ and $c$ are known to the learning agent.

The agent interacts with the environment for $K$ episodes, each of length $H$. In each episode, the agent starts from a state $s_1$ and chooses a Markov policy $\pi_k$ determined by the information gathered until that episode. This policy is then executed in the next episode, while collecting the rewards and costs. The main objectives of the learning agent are to:

(1) Maximize the expected cumulative reward or equivalently, minimize the Bayesian regret with respect to the reward function defined as: $\mathfrak{BR}(K; r) := \mathbb{E}\left[\sum_{k=1}^{K}\left(V_1^{\pi^*}(s_1; r, p) - V_1^{\pi^k}(s_1; r, p)\right)\right]$. (2) Minimize the constraint violation or equivalently, minimize the Bayesian regret with respect to the constraint defined as: $\mathfrak{BR}(K; c) := \mathbb{E}\left[\sum_{k=1}^{K}\left(V_1^{\pi^k}(s_1; c, p) - \tau\right)\right]$.

The above notion of constraint violation regret may be regarded as weaker as it allows for error cancellations. Nevertheless, it's still widely used due to its utility in applications where average constraint satisfaction is sufficient (e.g., LLM or image safety).

## 4 THE SAFE-PSRL ALGORITHM

We propose the Safe Posterior Sampling-based Reinforcement Learning (`Safe-PSRL`) algorithm for the finite-horizon CMDP model. This algorithm leverages the idea of posterior sampling to balance exploration and exploitation. It also takes a primal-dual approach to handle the constraint cost objective along with reward maximization objective. We further introduce the idea of pessimism [Liu et al., 2021b] to ensure that the cost regret is bounded. This "pessimism" is achieved by considering a "more constrained" CMDP problem as compared to the original problem. This is done by decreasing the threshold by $\epsilon_k$ in each episode $k$. Formally, we consider the objective:

$$\max \quad V_1^{\pi}(r, p) \quad \text{s.t.} \quad V_1^{\pi}(c, p) \leq \tau - \epsilon_k. \tag{2}$$

This pessimistic term $\epsilon_k$ ensures bounded cost regret and it decreases as the episode count increases. The algorithm starts with the prior distribution $\mu_1$ on the transition probability. Then, at every time $t$, the learning agent maintains a posterior distribution $\mu_t$ on the unknown transition probability $p$ given by $\mu_t(\Theta) = \mathbb{P}(p \in \Theta | \mathcal{F}_t)$ for any set $\Theta \subseteq \Theta_{c_0}$. Here $\mathcal{F}_t$ is the information available at time $t$, i.e., the sigma algebra generated by encountered states and actions up to time $t$. On observing the next state $s_{t+1}$ by taking action $a_t$ at state $s_t$, the posterior is updated according to Bayes's rule:

$$\mu_{t+1}(dp) = \frac{p_t(s_{t+1}|s_t, a_t)\mu_t(dp)}{\int p_t'(s_{t+1}|s_t, a_t)\mu_t(dp')}. \tag{3}$$

In parallel, at the beginning of each episode $k$, transition probability $\hat{p}_k$ is sampled from the posterior distribution $\mu_{t_k}$ (where $t_k$ is the time corresponding to beginning of episode $k$). We then consider the Lagrangian defined as $L_k(\pi, \lambda) = V_1^{\pi}(r, \hat{p}_k) + \frac{\lambda_k}{\eta_k}\left(\tau - \epsilon_k - V_1^{\pi}(c, \hat{p}_k)\right)$.

The learning agent then chooses a Markov policy $\pi_k$ (primal update) which maximizes the above Lagrangian. We can find such a policy by applying standard dynamic programming with respect to the reward function $r - \frac{\lambda_k}{\eta_k}c$. The (dual) parameter $\lambda_k$ is updated according to the sub-gradient algorithm as: $\lambda_{k+1} = \left(\lambda_k + V_1^{\pi_k}(c, \hat{p}_k) + \epsilon_k - \tau\right)_+$. The agent then applies the policy $\pi_k$ for the $H$ steps of episode $k$. Despite policy $\pi_k$ being deterministic and not randomized, we will show that a sequence of distinct deterministic policies will do well with respect to both objective and constraint objectives when we consider the cumulative performance over multiple episodes.

**Algorithm 1** `Safe-PSRL for Episodic CMDPs`

**Input**: $K, \mu_1, c_0, \tau$
**Initialization**: $\lambda^1 \leftarrow 0$
**for** episodes $k = 1, \ldots, K$ **do**
  $K_\epsilon \leftarrow 5$
  $\epsilon_k \leftarrow \frac{K_\epsilon |H|^{1.5}\sqrt{|\mathcal{S}|^2|\mathcal{A}|}(\log k|\mathcal{S}||\mathcal{A}|H+1)}{\sqrt{k \log k|\mathcal{S}||\mathcal{A}|H}}$
  $\eta_k \leftarrow (\tau - c_0)H\sqrt{k}$
  $t_k = (k-1)H + 1$
  Generate $\hat{p}_k \sim \mu_{t_k}(.)$
  Compute $\pi_k \in \arg\max_\pi V_1^\pi(r - \frac{\lambda_k}{\eta_k}c, \hat{p}_k)$
  (Policy Update)
  $\lambda_{k+1} \leftarrow \max(0, \lambda_k + V_1^{\pi_k}(c, \hat{p}_k) + \epsilon_k - \tau)$
  (Dual Update)
  **for** $t = (k-1)H + 1, \ldots, kH$ **do**
    Choose action $a_t \sim \pi_k(s_t)$
    Observe $s_{t+1} \sim p(.|s_t, a_t)$
    Update the posterior distribution $\mu_{t+1}$ according to (3)
  **end for**
**end for**

We note that while some of the details of the algorithm are natural (as they are common to PSRL algorithms for various settings) [Ouyang et al., 2017, Jafarnia-Jahromi et al., 2021c, Jafarnia-Jahromi et al., 2021a, Jafarnia-Jahromi et al., 2021b], the key novelty in the design are the $\epsilon_k$ and $\eta_k$ parameters to be used in conjunction with a primal-dual approach. Their choice is guided by the regret analysis presented in Section 5.

The following theorem establishes that the `Safe- PSRL` algorithm achieves sub-linear $\tilde{\mathcal{O}}(\sqrt{K})$ reward regret while achieving bounded constraint violation regret.

**Theorem 1.** *Suppose Assumption 1 holds, then the reward and cost regret of the `Safe PSRL` algorithm is upper bounded as:*

$$\mathfrak{BR}(K; r) = \tilde{\mathcal{O}}\left(\frac{H^{2.5}}{\tau - c^0}\sqrt{|\mathcal{S}|^2|\mathcal{A}|K}\right)$$
$$\mathfrak{BR}(K; c) = \tilde{\mathcal{O}}\left(C''(H-\tau) + H^{1.5}\sqrt{|\mathcal{S}|^2|\mathcal{A}|C''}\right) = \mathcal{O}(1),$$

*where* $C'' = \mathcal{O}(\frac{H^3|\mathcal{S}|^2|\mathcal{A}|}{(\tau - c^0)^2})$ *is independent of* $K$.

*Remark* 1. (i) The upper bound on $\mathfrak{BR}(K; r)$ of the `OptPess-PrimalDual` algorithm [Liu et al., 2021a] is $\tilde{\mathcal{O}}\left(H^{2.5}\sqrt{|\mathcal{S}|^3|\mathcal{A}|K}\right)$ when we consider a stationary transition probability setting. Thus, our upper bound is the same in terms of $|\mathcal{A}|$,$K$,$H$, and better in terms of $|\mathcal{S}|$. This improvement is achieved by leveraging the tighter regret bounds of posterior sampling algorithms versus optimism-based algorithms for MDPs. Also, both `OptPess-PrimalDual` and `Safe-PSRL` achieve $\tilde{\mathcal{O}}(1)$ upper bounds on $\mathfrak{BR}(K; c)$.

(ii) The upper bound on $\mathfrak{BR}(K; r)$ of the `DOPE` algorithm [Bura et al., 2022] is also $\tilde{\mathcal{O}}\left(H^{2.5}\sqrt{|\mathcal{S}|^2|\mathcal{A}|K}\right)$ when we consider a stationary transition probability setting. Although our regret bounds are comparable to those of `DOPE`, we shall see that the numerical performance is much better. The `DOPE` algorithm guarantees zero constraint violations with high probability. However, this requires a strong assumption, i.e, the knowledge of a safe policy that can satisfy the constraint.

(iii) The CMDP-PSRL algorithm [Agarwal et al., 2022] uses posterior sampling in the average CMDP setting and achieves $\tilde{\mathcal{O}}\left(T_M|\mathcal{S}|\sqrt{|A|K}\right)$ reward objective and the same constraint violation regret, where $T_M$ is the mixing time. In comparison, we are able to achieve bounded constraint violation regret (albiet, in a different setting i.e., finite horizon).

(iv) Note that we only assume that there exists a policy which satisfies the constraint threshold $c_0$ i.e., we do not need to know (and the algorithm does not use) a policy that satisfies it. This is not an uncommon assumption in the literature and is justified since one can choose a smaller support for the transition probabilities. Crucially, we do not need a fallback policy.

## 5 REGRET ANALYSIS

We now provide theoretical analysis of the `Safe-PSRL` algorithm. We first state some relevant results from the literature on posterior sampling for RL.

A key property of posterior sampling [Osband et al., 2013] is the posterior sampling lemma, i.e., the transition probability $\hat{p}_t$ sampled from the posterior distribution at time $t$ and transition probability $p$ have the same distribution.

**Lemma 1.** *For any function* $f$*, we have* $\mathbb{E}[f(\hat{p}_t)] = \mathbb{E}[f(p)]$ *where* $p$ *is the transition probability and* $\hat{p}_t$ *is the sampled transition probability from the posterior distribution* $\mu_t$ *at time* $t$.

The following is a restatement [Osband et al., 2013] of the sub-linear regret bound achieved when using posterior sampling for unconstrained finite horizon MDPs.

**Lemma 2.** *The Bayesian regret of the PSRL algorithm for unconstrained MDPs is given by* $\sum_{k=1}^K \mathbb{E}\left[V_1^{\pi^k}(c, p) - V_1^{\pi^k}(c, \hat{p}_k)\right] \leq H^{1.5}\sqrt{30|\mathcal{S}|^2|\mathcal{A}|K\log(|\mathcal{S}||\mathcal{A}|KH)} + 2H$.

The above lemma holds for both the objective reward function $r$ and constraint cost function $c$.

**Cost Constraint Violation Analysis.** We present analysis of the cost constraint violation. We can de-

compose the constraint violation regret $\mathfrak{BR}(K;c)$ as:

$$
\begin{aligned}
&= \sum_{k=1}^{K} \mathbb{E}\left[V_1^{\pi^k}(c,p) - V_1^{\pi^k}(c,\hat{p}_k)\right] \\
&+ \sum_{k=1}^{K} \mathbb{E}\left[V_1^{\pi^k}(c,\hat{p}_k) - \tau\right] \\
&\leq \sum_{k=1}^{K} \mathbb{E}\left[V_1^{\pi^k}(c,p) - V_1^{\pi^k}(c,\hat{p}_k)\right] \\
&+ \sum_{k=1}^{K} \mathbb{E}\left[\lambda_{k+1} - \lambda_k - \epsilon_k\right] \\
&\text{(by dual update rule of algorithm)} \\
&= \sum_{k=1}^{K} \mathbb{E}\left[V_1^{\pi^k}(c,p) - V_1^{\pi^k}(c,\hat{p}_k)\right] \\
&+ \mathbb{E}\left[\lambda_{K+1}\right] - \sum_{k=1}^{K} \epsilon_k \qquad (4) \\
&\leq H^{1.5}\sqrt{30|\mathcal{S}|^2|\mathcal{A}|K\log(|\mathcal{S}||\mathcal{A}|KH)} + 2H \\
&+ \mathbb{E}\left[\lambda_{K+1}\right] - \sum_{k=1}^{K} \epsilon_k \qquad (5)
\end{aligned}
$$

where the last upper bound follows by use of Lemma 2 to upper bound the first term in (4). We next show that the dual parameter $\mathbb{E}\left[\lambda_{K+1}\right]$ can be upper bounded by use of Lyapunov-drift analysis. To that end, we restate the following lemma [Liu et al., 2021b] which states the Lyapunov-drift conditions for the boundedness of a random process.

**Lemma 3.** *[Liu et al., 2021b] Consider a random process $S(t)$ with a Lyapunov function $\Phi(k)$ such that $\Phi(0) = \Phi_0$ and $\Delta(k) = \Phi(k+1) - \Phi(k)$ is the Lyapunov drift. Given an increasing sequence $\{\varphi_k\}$ and constants $\rho$ and $\nu_{max}$ with $0 < \rho \leq \nu_{max}$, if the expected drift $\mathbb{E}\left[\Delta(k)|S(k) = s\right]$ satisfies the following conditions:*
*(i) There exists constants $\rho > 0$ and $\varphi_k > 0$ s.t. $\mathbb{E}\left[\Delta(k)|S(k) = s\right] \leq -\rho$ when $\Phi(k) \geq \varphi_k$, and*
*(ii) $|\Phi(k+1) - \Phi(k)| \leq \nu_{\max}$ with probability 1, then*

$$
\mathbb{E}\left[e^{\zeta\Phi(t)}\right] \leq \mathbb{E}\left[e^{\zeta\Phi_0}\right] + \frac{2e^{\zeta(\nu_{\max}+\varphi_t)}}{\zeta\rho},
$$

*where $\zeta = \rho/(\nu_{\max}^2 + \nu_{\max}\rho/3)$.*

We divide the episodes into two parts, i.e. $k < C''$ and $k \geq C''$ where $C'' = \frac{80H^3|\mathcal{S}|^2|\mathcal{A}|}{(\tau - c_0)^2}$. We can clearly see that for $k \geq C''$, we have $\epsilon_k \leq \frac{\tau - c_0}{2}$. Thus, for $k \geq C''$, Problem (2) is feasible for all $\hat{p_k} \in \Theta_{c_0}$ by Assumption 1. For $k \geq C''$, we show that the Lyapunov function $\Phi(\lambda) = \lambda$ satisfies the conditions of Lemma 3 and thus provide a bound on the exponential moment of the dual variable $\lambda$.

**Lemma 4.** *For $k \geq C''$, when $\lambda \geq \varphi_k$, we have, $\mathbb{E}\left[\lambda_{k+1} - \lambda_k | \lambda_k = \lambda\right] \leq \rho$ and $|\lambda_{k+1} - \lambda_k| \leq H$ with probability 1, where $\varphi_k := 4(H^2 + \epsilon_k^2 + \eta_k H)/(\tau - c^0)$ and $\rho := (\tau - c_0)/4$. Thus, we have,*

$$
\mathbb{E}\left[e^{\zeta\lambda_{K+1}}\right] \leq \mathbb{E}\left[e^{\zeta\lambda_{C''}}\right] + \frac{2e^{\zeta(H+\varphi_{K+1})}}{\zeta\rho}, \qquad (6)
$$

where $\zeta = \rho/(H^2 + H\rho/3)$. The above inequality (6) can be simplified to

$$
\begin{aligned}
\mathbb{E}\left[\lambda_{K+1}\right] \leq \frac{1}{\zeta}\log\frac{11H^2}{3\rho^2} + H + \sum_{1}^{C''}\epsilon_k + C''(H-\tau) \\
+ \frac{4(H^2 + \epsilon_{K+1}^2 + \eta_{K+1}H)}{(\tau - c^0)}. \qquad (7)
\end{aligned}
$$

Next, we bound the $\sum_k \epsilon_k$ term:

$$
\begin{aligned}
\sum_{k=1}^{K}\epsilon_k &\geq \int_1^{K+1}\epsilon_u \mathrm{d}u \qquad (8) \\
&\geq 10H^{1.5}\sqrt{|\mathcal{S}|^2|\mathcal{A}|Klog|\mathcal{S}||\mathcal{A}|HK} \qquad (9) \\
&- 10H^{1.5}\sqrt{|\mathcal{S}|^2|\mathcal{A}|log|\mathcal{S}||\mathcal{A}|H}.
\end{aligned}
$$

Thus, putting together (5), (7) and (8), the leading terms of $\tilde{\mathcal{O}}(\sqrt{K})$ cancel out and we get $\mathfrak{BR}(K;c) = \tilde{\mathcal{O}}\left(C''(H-\tau) + H^{1.5}\sqrt{|\mathcal{S}|^2|\mathcal{A}|C''}\right) = \tilde{\mathcal{O}}(1)$ i.e., constraint violation regret is a constant, and does not grow with $K$.

**Reward Objective Regret Analysis.** We next provide regret analysis of the reward objective. Let $\pi^{\epsilon_k,*}$ be the optimal policy for the pessimistic optimization problem (where $p$ is MDP's true transition probability):

$$
\max \quad V_1^{\pi}(r,p) \quad \text{s.t.} \quad V_1^{\pi}(c,p) \leq \tau - \epsilon_k.
$$

Let $\pi^{\epsilon_k,\hat{p}_k}$ be the optimal policy for the pessimistic optimization problem (where $\hat{p}_k$ is a sampled transition probability):

$$
\max \quad V_1^{\pi}(r,\hat{p}_k) \quad \text{s.t.} \quad V_1^{\pi}(c,\hat{p}_k) \leq \tau - \epsilon_k.
$$

We decompose the reward regret term $\mathfrak{BR}(K;r)$ different from that for Optimism-type algorithms. It is decomposed to utilize the posterior sampling property as follows:

$$
\begin{aligned}
&\sum_{k=1}^{C''-1}\mathbb{E}\left[V_1^{\pi^*}(r,p) - V_1^{\pi^k}(r,p)\right] \\
&+ \sum_{k=C''}^{K}\mathbb{E}\left[V_1^{\pi^*}(r,p) - V_1^{\pi^k}(r,p)\right] \\
&\leq C'' H + \sum_{k=C''}^{K}\mathbb{E}\left[V_1^{\pi^*}(r,p) - V_1^{\pi^{\epsilon_k,*}}(r,p)\right]
\end{aligned}
$$

$$+\sum_{k=C''}^{K}\mathbb{E}\left[V_1^{\pi^{\epsilon_k,*}}(r,p)-V_1^{\pi^{\epsilon_k,\hat{p}_k}}(r,\hat{p}_k)\right]$$
$$+\sum_{k=C''}^{K}\mathbb{E}\left[V_1^{\pi^{\epsilon_k,\hat{p}_k}}(r,\hat{p}_k)-V_1^{\pi_k}(r,\hat{p}_k)\right]$$
$$+\sum_{k=C''}^{K}\mathbb{E}\left[V_1^{\pi_k}(r,\hat{p}_k)-V_1^{\pi_k}(r,p)\right]$$
$$\leq C^{''}H+\sum_{k=C''}^{K}\mathbb{E}\left[V_1^{\pi^*}(r,p)-V_1^{\pi^{\epsilon_k,*}}(r,p)\right]+0$$

(by the posterior sampling property in Lemma 1)

$$+\sum_{k=C''}^{K}\mathbb{E}\left[V_1^{\pi^{\epsilon_k,\hat{p}_k}}(r,\hat{p}_k)-V_1^{\pi_k}(r,\hat{p}_k)\right]$$
$$+\sum_{k=C''}^{K}\mathbb{E}\left[V_1^{\pi_k}(r,\hat{p}_k)-V_1^{\pi_k}(r,p)\right]$$
$$\leq C^{''}H+\sum_{k=C''}^{K}\mathbb{E}\left[V_1^{\pi^*}(r,p)-V_1^{\pi^{\epsilon_k,*}}(r,p)\right]$$
$$+\sum_{k=C''}^{K}\mathbb{E}\left[V_1^{\pi^{\epsilon_k,\hat{p}_k}}(r,\hat{p}_k)-V_1^{\pi_k}(r,\hat{p}_k)\right]$$
$$+H^{1.5}\sqrt{30|\mathcal{S}|^2|\mathcal{A}|K\log(|\mathcal{S}||\mathcal{A}|KH)}+2H$$

( by the regret bound in Lemma 2) (10)

The other terms are bounded as follows. Similar to Lemma 5.7 in [Liu et al., 2021a], we can define a probabilistic mixed policy of $\pi^*$ and $\pi_0^p$ to prove the following:

**Lemma 5.** *The first summation term above can be bounded as* $\sum_{k=C''}^{K}\mathbb{E}\left[V_1^{\pi^*}(r,p)-V_1^{\pi^{\epsilon_k,*}}(r,p)\right]\leq\sum_{k=C''}^{K}\frac{\epsilon_k H}{\tau-c^0}=\tilde{\mathcal{O}}\left(\frac{H^{2.5}}{\tau-c^0}\sqrt{|\mathcal{S}|^2|\mathcal{A}|K}\right)$.

By optimality of $\pi_k$ and the nature of the update of the dual parameter $\lambda_k$, we can prove the following:

**Lemma 6.** $\sum_{k=C''}^{K}\mathbb{E}\left[V_1^{\pi^{\epsilon_k,\hat{p}_k}}(r,\hat{p}_k)-V_1^{\pi_k}(r,\hat{p}_k)\right]=\tilde{\mathcal{O}}\left(\frac{H}{\tau-c^0}\sqrt{K}\right)$

The proof of this lemma can be found in the Appendix. Now, putting together (10), Lemma 5 and Lemma 6, we get that $\mathfrak{BR}(K;r)=\tilde{\mathcal{O}}\left(\frac{H^{2.5}}{\tau-c^0}\sqrt{|\mathcal{S}|^2|\mathcal{A}|K}\right)$.

*Remark* 2. We note that we can improve the upper bound on $\mathfrak{BR}(K;r)$ to $\tilde{\mathcal{O}}\left(H^{2.5}\sqrt{|\mathcal{S}||\mathcal{A}|K}\right)$ by leveraging an improved regret bound [Osband and Van Roy, 2017], i.e., $\tilde{\mathcal{O}}\left(H^{1.5}\sqrt{|\mathcal{S}||\mathcal{A}|K}\right)$ for the PSRL algorithm and appropriate scaling of the $\epsilon_k$ terms. But, this would require an assumption that the transition probability has an independent Dirichlet prior.

# 6 EMPIRICAL PERFORMANCE

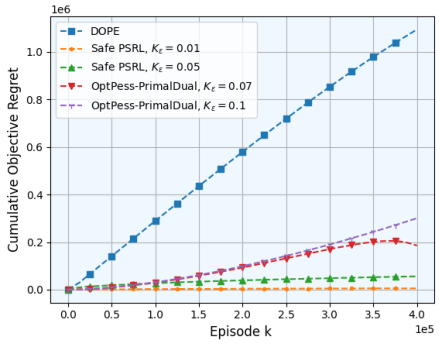
We now evaluate the empirical objective and constraint regret performance. Although our theoretical results are on Bayesian regret, our empirical evaluation is for frequentist (non-Bayesian) performance of the `Safe-PSRL` algorithm, benchmarking it against the state-of-the-art `DOPE` algorithm [Bura et al., 2022], which has been shown to perform better than other comparable algorithms (e.g., `OptPess-LP` in [Liu et al., 2021a]) and `OptPess-PrimalDual` algorithm [Liu et al., 2021a], which is actually the closest comparable algorithm to ours.

**Environment:** We consider media streaming [Bura et al., 2022] from a wireless base station offering two speeds. Data packets are stored in a buffer and transmitted based on a Bernoulli process. The objective is to minimize the cost of packet shortages while limiting the use of the faster service. We model this scenario as a finite-horizon CMDP, with the state representing the buffer size and actions $\{1,2\}$ representing speed choices.

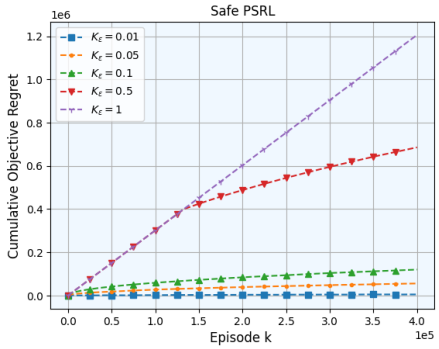
**Experimental Setup:** We evaluate the cumulative regret for the `Safe-PSRL`, `DOPE`, and `OptPess-PrimalDual` algorithms. The transition probability is fixed and not sampled from a prior distribution (i.e., the evaluation is frequentist). For the `Safe-PSRL` algorithm, we consider a Dirichlet prior for the posterior distribution, with parameters $[0.1,\ldots,0.1]$, a convenient choice, since it is a conjugate prior for the multinomial and categorical distributions. We further scale the $\epsilon_k$ parameters of the `Safe-PSRL` and `OptPess-PrimalDual` algorithm, by varying the coefficient of the $\epsilon_k$ parameters denoted by $K_\epsilon$, to control the pessimism. We report the empirical performance for various values of $K_\epsilon$. This is in line with the practical use of various state-of-the-art and time-tested algorithms like UCB [Auer, 2002] etc. The performance of our algorithm is also compared against the `DOPE` algorithm, which requires a known safe policy.

We choose the optimal policy of the given CMDP with $c_0=1$ as the safe policy. The same $c_0$ is also used in the `Safe-PSRL` and `OptPess-PrimalDual` algorithms as the satisfiable constraint threshold. Our experiments are repeated 10 times and averaged to obtain the regret plots. Additional details on the environment setup and experiments are provided in Appendix B.

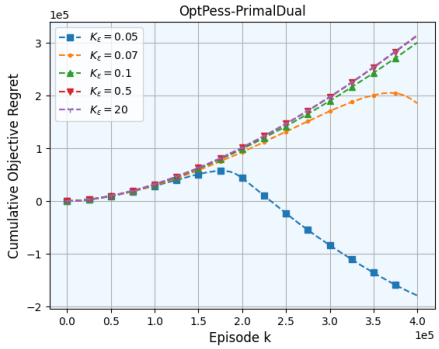
Figure 1(a) shows that the `Safe-PSRL` algorithm significantly outperforms the `DOPE` and `OptPess-PrimalDual` algorithms in terms of objective regret. It also ensures that the constraint regret is negative for almost all of the episodes, as shown by Fig. 2(a). We also see that the constraint is satisfied in almost all of the episodes,




(a)

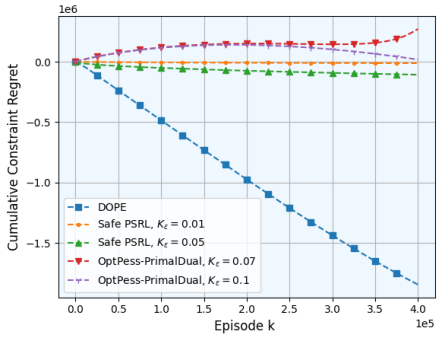



(b)

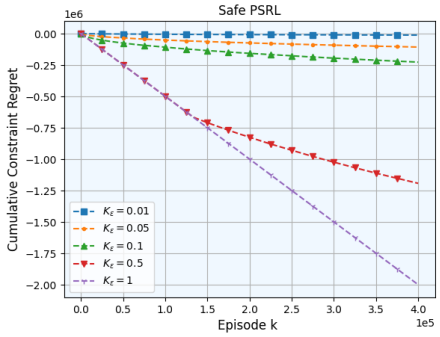



(c)

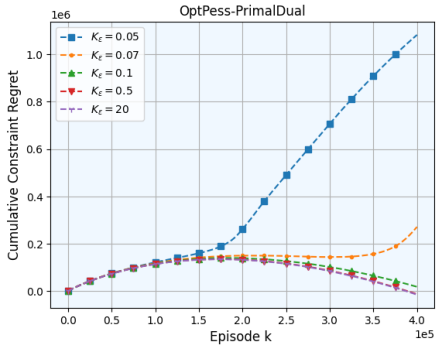
Figure 1: Cumulative objective regret for (a) different algorithms; (b) `Safe-PSRL` with different values of $K_\epsilon$; (c) `OptPess-PrimalDual` with different values of $K_\epsilon$.




(a)




(b)




(c)

Figure 2: Cumulative constraint regret for (a) different algorithms; (b) `Safe-PSRL` with different values of $K_\epsilon$; (c) `OptPess-PrimalDual` with different values of $K_\epsilon$.

much better than our theoretical guarantee. Further, though the `OptPess-PrimalDual` algorithm appears to perform better than the `DOPE` algorithm in Fig. 1(a) in terms of objective regret, it has very high constraint regret, as shown in Fig. 2(a). On the other hand, `DOPE` satisfies the constraint in every episode.

We further evaluate `Safe-PSRL` for various values of $K_\epsilon$ and note that, in all instances, the constraint regret is negative for almost all of the episodes, as shown by Fig. 2(b). Moreover, the objective regret in Fig. 1(b) increases as the levels of *pessimism* expressed by $K_\epsilon$ increase. Therefore, for suitable levels of pessimism, `Safe-PSRL` algorithm ensures low objective regret while satisfying the constraint objective. Differently, Fig. 2(c) shows that the `OptPess-PrimalDual` algorithm is unable to achieve low regret even at high levels of pessimism. Considering Fig. 1(c) and Fig. 2(c) together, we see that the algorithm achieves low objective regret at the expense of exploding constraint regret. Overall, `Safe-PSRL` is able to achieve superior objective regret performance while satisfying the constraint for almost all the episodes without the knowledge of a safe policy.

**Additional Experiments:** We conduct additional

experiments on an inventory control problem. The `Safe-PSRL` algorithm again significantly outperforms `DOPE` and `OptPess-PrimalDual`. Further details on the setup and experiments are in Appendix C.

**Clipped Regret Comparison:** We also evaluate our algorithm using clipped regret, a stricter regret definition where negative terms are clipped to 0. Our algorithm demonstrates superior objective regret performance. While `DOPE` achieves near-zero constraint regret, this relies on knowledge of a safe policy, which we do not assume. Our algorithm outperforms `OptPess-PrimalDual` in empirical constraint regret, despite similar assumptions. See Appendix D for plots.

## 7 CONCLUSIONS

We address safe online learning for episodic CMDPs with unknown transition probabilities. `Safe-PSRL` is the first posterior sampling algorithm to guarantee bounded constraint violation regret while achieving near-optimal reward regret. It outperforms state-of-the-art algorithms (e.g., `DOPE` and `OptPess-PrimalDual`) without assuming a known safe policy.

## 8 ACKNOWLEDGMENTS

This work was supported in part under NSF grants EECS 2514683, CNS 2514748 and ECCS 2025732.

### References

[Agarwal et al., 2022] Agarwal, M., Bai, Q., and Aggarwal, V. (2022). Regret guarantees for model-based reinforcement learning with long-term average constraints. In *The 38th Conference on Uncertainty in Artificial Intelligence*.

[Agrawal and Goyal, 2012] Agrawal, S. and Goyal, N. (2012). Analysis of thompson sampling for the multi-armed bandit problem. In *Conference on learning theory*, pages 39–1. JMLR Workshop and Conference Proceedings.

[Agrawal and Goyal, 2013] Agrawal, S. and Goyal, N. (2013). Thompson sampling for contextual bandits with linear payoffs. In *International Conference on Machine Learning*, pages 127–135. PMLR.

[Altman, 1999] Altman, E. (1999). *Constrained Markov Decision Processes*, volume 7. CRC Press.

[Auer, 2002] Auer, P. (2002). Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research*, 3(Nov):397–422.

[Azar et al., 2017] Azar, M. G., Osband, I., and Munos, R. (2017). Minimax regret bounds for reinforcement learning. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 263–272. JMLR. org.

[Bertsekas, 2015] Bertsekas, D. P. (2015). Dynamic programming and optimal control 4th edition, volume ii. *Athena Scientific*.

[Borkar, 1988] Borkar, V. S. (1988). A convex analytic approach to markov decision processes. *Probability Theory and Related Fields*, 78(4):583–602.

[Brantley et al., 2020] Brantley, K., Dudik, M., Lykouris, T., Miryoosefi, S., Simchowitz, M., Slivkins, A., and Sun, W. (2020). Constrained episodic reinforcement learning in concave-convex and knapsack settings. *Advances in Neural Information Processing Systems*, 33:16315–16326.

[Bura et al., 2022] Bura, A., Hasanzadezonuzy, A., Kalathil, D., Shakkottai, S., and Chamberland, J.-F. (2022). Dope: Doubly optimistic and pessimistic exploration for safe reinforcement learning. In *Advances in Neural Information Processing Systems*.

[Chapelle and Li, 2011] Chapelle, O. and Li, L. (2011). An empirical evaluation of thompson sampling. *Advances in neural information processing systems*, 24:2249–2257.

[Chen et al., 2022] Chen, L., Jain, R., and Luo, H. (2022). Learning infinite-horizon average-reward markov decision processes with constraints. *arXiv preprint arXiv:2202.00150*.

[Ding et al., 2021] Ding, D., Wei, X., Yang, Z., Wang, Z., and Jovanovic, M. (2021). Provably efficient safe exploration via primal-dual policy optimization. In *International Conference on Artificial Intelligence and Statistics*, pages 3304–3312. PMLR.

[Ding et al., 2020] Ding, D., Zhang, K., Basar, T., and Jovanovic, M. (2020). Natural policy gradient primal-dual method for constrained markov decision processes. *Advances in Neural Information Processing Systems*, 33:8378–8390.

[Efroni et al., 2020] Efroni, Y., Mannor, S., and Pirotta, M. (2020). Exploration-exploitation in constrained mdps. *arXiv preprint arXiv:2003.02189*.

[Ghosh et al., 2022] Ghosh, A., Zhou, X., and Shroff, N. (2022). Provably efficient model-free constrained rl with linear function approximation. *Advances in Neural Information Processing Systems*, 35:13303–13315.

[HasanzadeZonuzy et al., 2021] HasanzadeZonuzy, A., Bura, A., Kalathil, D., and Shakkottai, S. (2021). Learning with safety constraints: Sample complexity of reinforcement learning for constrained mdps. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 7667–7674.

[Hordijk and Kallenberg, 1979] Hordijk, A. and Kallenberg, L. (1979). Linear programming and markov decision chains. *Management Science*, 25(4):352–362.

[Jafarnia-Jahromi et al., 2021a] Jafarnia-Jahromi, M., Chen, L., Jain, R., and Luo, H. (2021a). Online learning for stochastic shortest path model via posterior sampling. *arXiv preprint arXiv:2106.05335*.

[Jafarnia-Jahromi et al., 2021b] Jafarnia-Jahromi, M., Jain, R., and Nayyar, A. (2021b). Learning zero-sum stochastic games with posterior sampling. *arXiv preprint arXiv:2109.03396*.

[Jafarnia-Jahromi et al., 2021c] Jafarnia-Jahromi, M., Jain, R., and Nayyar, A. (2021c). Online learning for unknown partially observable mdps. *arXiv preprint arXiv:2102.12661*.

[Jaksch et al., 2010] Jaksch, T., Ortner, R., and Auer, P. (2010). Near-optimal regret bounds for reinforcement learning. *Journal of Machine Learning Research*, 11(Apr):1563–1600.

[Jin et al., 2018] Jin, C., Allen-Zhu, Z., Bubeck, S., and Jordan, M. I. (2018). Is Q-learning provably efficient? In *Advances in Neural Information Processing Systems*, pages 4863–4873.

[Kalagarla, 2025] Kalagarla, K. C. (2025). Safe psrl. `https://github.com/kalagarl/Safe_PSRL`.

[Kalagarla et al., 2021] Kalagarla, K. C., Jain, R., and Nuzzo, P. (2021). A Sample-Efficient Algorithm for Episodic Finite-Horizon MDP with Constraints. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 8030–8037.

[Kaufmann et al., 2012] Kaufmann, E., Korda, N., and Munos, R. (2012). Thompson sampling: An asymptotically optimal finite-time analysis. In *International conference on algorithmic learning theory*, pages 199–213. Springer.

[Lai and Robbins, 1985] Lai, T. L. and Robbins, H. (1985). Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1):4–22.

[Liu et al., 2021a] Liu, T., Zhou, R., Kalathil, D., Kumar, P., and Tian, C. (2021a). Learning policies with zero or bounded constraint violation for constrained mdps. *Advances in Neural Information Processing Systems*, 34:17183–17193.

[Liu et al., 2021b] Liu, X., Li, B., Shi, P., and Ying, L. (2021b). An efficient pessimistic-optimistic algorithm for stochastic linear bandits with general constraints. *Advances in Neural Information Processing Systems*, 34:24075–24086.

[Müller et al., 2024] Müller, A., Alatur, P., Cevher, V., Ramponi, G., and He, N. (2024). Truly no-regret learning in constrained mdps. *arXiv preprint arXiv:2402.15776*.

[Osband et al., 2013] Osband, I., Russo, D., and Van Roy, B. (2013). (more) efficient reinforcement learning via posterior sampling. *Advances in Neural Information Processing Systems*, 26.

[Osband and Van Roy, 2017] Osband, I. and Van Roy, B. (2017). Why is posterior sampling better than optimism for reinforcement learning? In *International Conference on Machine Learning*, pages 2701–2710. PMLR.

[Ouyang et al., 2017] Ouyang, Y., Gagrani, M., Nayyar, A., and Jain, R. (2017). Learning unknown markov decision processes: A thompson sampling approach. In *Advances in Neural Information Processing Systems*, pages 1333–1342.

[Provodin et al., 2023] Provodin, D., Gajane, P., Pechenizkiy, M., and Kaptein, M. (2023). Provably efficient exploration in constrained reinforcement learning: Posterior sampling is all you need. *arXiv preprint arXiv:2309.15737*.

[Puterman, 1994] Puterman, M. L. (1994). *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc., New York, NY, USA, 1st edition.

[Qiu et al., 2020] Qiu, S., Wei, X., Yang, Z., Ye, J., and Wang, Z. (2020). Upper confidence primal-dual reinforcement learning for cmdp with adversarial loss. *Advances in Neural Information Processing Systems*, 33:15277–15287.

[Russo et al., 2018] Russo, D. J., Van Roy, B., Kazerouni, A., Osband, I., Wen, Z., et al. (2018). A tutorial on thompson sampling. *Foundations and Trends® in Machine Learning*, 11(1):1–96.

[Singh et al., 2020] Singh, R., Gupta, A., and Shroff, N. B. (2020). Learning in markov decision processes under constraints. *arXiv preprint arXiv:2002.12435*.

[Thompson, 1933] Thompson, W. R. (1933). On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294.

[Wei et al., 2022] Wei, H., Liu, X., and Ying, L. (2022). Triple-q: A model-free algorithm for constrained reinforcement learning with sublinear regret and zero constraint violation. In *International Conference on Artificial Intelligence and Statistics*, pages 3274–3307. PMLR.

[Zheng and Ratliff, 2020] Zheng, L. and Ratliff, L. (2020). Constrained upper confidence reinforcement learning. In *Learning for Dynamics and Control*, pages 620–629. PMLR.

## Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes]
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes]

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes]
   (b) Complete proofs of all theoretical results. [Yes]
   (c) Clear explanations of any assumptions. [Yes]

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes]
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes]
   (b) The license information of the assets, if applicable. [Not Applicable]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Not Applicable]
   (d) Information about consent from data providers/curators. [Not Applicable]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Not Applicable]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

# A Proofs

## A.1 Proof of Lemma 4

*Proof.* Now for $k \geq C''$, consider:

$$\begin{aligned}
\frac{{\lambda_{k+1}}^2}{2} - \frac{{\lambda_k}^2}{2} &= \lambda_k(\lambda_{k+1} - \lambda_k) + \frac{1}{2}(\lambda_{k+1} - \lambda_k)^2 \\
&= \lambda_k(V_1^{\pi_k}(c, \hat{p}_k) + \epsilon_k - \tau) + \frac{1}{2}(V_1^{\pi_k}(c, \hat{p}_k) + \epsilon_k - \tau)^2 \\
&= \lambda_k(V_1^{\pi_k}(c, \hat{p}_k) + \epsilon_k - \tau) - \eta_k V_1^{\pi_k}(r, \hat{p}_k) + \eta_k V_1^{\pi_k}(r, \hat{p}_k) + \frac{1}{2}(V_1^{\pi_k}(c, \hat{p}_k) + \epsilon_k - \tau)^2 \\
&\leq \lambda_k(V_1^{\pi_k}(c, \hat{p}_k) + \epsilon_k - \tau) - \eta_k V_1^{\pi_k}(r, \hat{p}_k) + \eta_k H + \frac{1}{2}(V_1^{\pi_k}(c, \hat{p}_k) + \epsilon_k - \tau)^2 \\
&\leq \lambda_k(V_1^{\pi_k}(c, \hat{p}_k) + \epsilon_k - \tau) - \eta_k V_1^{\pi_k}(r, \hat{p}_k) + \eta_k H + (V_1^{\pi_k}(c, \hat{p}_k) - \tau)^2 + \epsilon_k^2 \\
&\text{(Using } \frac{(a+b)^2}{2} \leq a^2 + b^2\text{)} \\
&\leq \lambda_k(V_1^{\pi_k}(c, \hat{p}_k) + \epsilon_k - \tau) - \eta_k V_1^{\pi_k}(r, \hat{p}_k) + \eta_k H + H^2 + \epsilon_k^2 \\
&\leq \lambda_k(V_1^{\pi_0^{\hat{p}_k}}(c, \hat{p}_k) + \epsilon_k - \tau) - \eta_k V_1^{\pi_0^{\hat{p}_k}}(r, \hat{p}_k) + \eta_k H + H^2 + \epsilon_k^2 \\
&\text{( By optimality of } \pi_k \text{ in primal update )} \\
&\leq \lambda_k(c_0 + \epsilon_k - \tau) + \eta_k H + H^2 + \epsilon_k^2 \\
&\leq -\frac{\lambda_k(\tau - c_0)}{2} + \eta_k H + H^2 + \epsilon_k^2 \text{( as for } k \geq C'', \epsilon_k \leq \frac{(\tau - c_0)}{2}\text{)}
\end{aligned}$$

Now for $\lambda \geq \varphi_k$ where $\varphi_k := 4(H^2 + \epsilon_k^2 + \eta_k H)/(\tau - c^0)$, we have:

$$\begin{aligned}
\mathbb{E}\left[\lambda_{k+1} - \lambda_k | \lambda_k = \lambda\right] &\leq \mathbb{E}\left[\frac{\lambda_{k+1}^2 - \lambda_k^2}{2\lambda_k} | \lambda_k = \lambda\right] \text{(Using } x - y \leq \frac{x^2 - y^2}{2y}, \text{for } y > 0\text{)} \\
&= \frac{1}{\lambda}\mathbb{E}\left[\frac{\lambda_{k+1}^2 - \lambda_k^2}{2} | \lambda_k = \lambda\right] \\
&\leq \frac{1}{\lambda}\mathbb{E}\left[-\frac{\lambda_k(\tau - c_0)}{2} + \eta_k H + H^2 + \epsilon_k^2 | \lambda_k = \lambda\right] \\
&= -\frac{(\tau - c_0)}{2} + \frac{\eta_k H + H^2 + \epsilon_k^2}{\lambda} \\
&\leq -\frac{(\tau - c_0)}{2} + \frac{(\tau - c_0)}{4} \\
&= -\frac{(\tau - c_0)}{4} := -\rho
\end{aligned}$$

Further, $|\lambda_{k+1} - \lambda_k| = |V_1^{\pi_k}(c, \hat{p}_k) + \epsilon_k - \tau| \leq H$ with probability 1. Thus, by lemma 3, we have :

$$\mathbb{E}\left[e^{\zeta \lambda_{K+1}}\right] \leq \mathbb{E}\left[e^{\zeta \lambda_{C''}}\right] + \frac{2e^{\zeta(H + \varphi_{K+1})}}{\zeta\rho},$$

where $\zeta = \rho/(H^2 + H\rho/3)$.

$$\begin{aligned}
&\implies e^{\zeta \mathbb{E}[\lambda_{K+1}]} \leq \mathbb{E}\left[e^{\zeta \lambda_{C''}}\right] + \frac{2e^{\zeta(H + \varphi_{K+1})}}{\zeta\rho} \text{ (By Jensen's inequality)} \\
&\implies \mathbb{E}\left[\lambda_{K+1}\right] \leq \frac{1}{\zeta}\log\left[\mathbb{E}\left[e^{\zeta \lambda_{C''}}\right] + \frac{2e^{\zeta(H + \varphi_{K+1})}}{\zeta\rho}\right]
\end{aligned}$$

Further,

$$\begin{aligned}\lambda_{C''} &\leq \lambda_1 + \sum_{1}^{C''-1} (V_1^{\pi^k}(c,\hat{p}_k) + \epsilon_k - \tau)_+ \\ &\leq \sum_{1}^{C''} \epsilon_k + C''(H-\tau) := \lambda_{C''}^{\max}\end{aligned}$$

Continuing,

$$\begin{aligned}\mathbb{E}\left[\lambda_{K+1}\right] &\leq \frac{1}{\zeta}\log\left[e^{\zeta\lambda_{C''}^{\max}} + \frac{2e^{\zeta(H+\varphi_{K+1})}}{\zeta\rho}\right] \\ &\leq \frac{1}{\zeta}\log\left[e^{\zeta\lambda_{C''}^{\max}} + \frac{8H^2e^{\zeta(H+\varphi_{K+1})}}{3\rho^2}\right] (\text{ Using } \zeta \geq \frac{3(\tau-c_0)}{13H^2}) \\ &\leq \frac{1}{\zeta}\log\left[\frac{11H^2}{3\rho^2}e^{\zeta\left(H+\varphi_{K+1}+\lambda_{C''}^{\max}\right)}\right] \\ &= \frac{1}{\zeta}\log\frac{11H^2}{3\rho^2} + H + \varphi_{K+1} + \lambda_{C''}^{\max} \\ &= \frac{1}{\zeta}\log\frac{11H^2}{3\rho^2} + H + \sum_{1}^{C''}\epsilon_k + C''(H-\tau) + \frac{4(H^2+\epsilon_{K+1}^2+\eta_{K+1}H)}{(\tau-c^0)}\end{aligned}$$

□

### A.2 Proof of Lemma 6

*Proof.*

$$\begin{aligned}&\sum_{k=C''}^{K}\mathbb{E}\left[V_1^{\pi^{\epsilon_k,\hat{p}_k}}(r,\hat{p}_k) - V_1^{\pi_k}(r,\hat{p}_k)\right] = \sum_{k=C''}^{K}\mathbb{E}\left[\frac{\lambda_k}{\eta_k}\left(V_1^{\pi^{\epsilon_k,\hat{p}_k}}(c,\hat{p}_k) - V_1^{\pi_k}(c,\hat{p}_k)\right)\right] \\ &+ \sum_{k=C''}^{K}\mathbb{E}\left[\left(V_1^{\pi^{\epsilon_k,\hat{p}_k}}(r,\hat{p}_k) - \frac{\lambda_k}{\eta_k}V_1^{\pi^{\epsilon_k,\hat{p}_k}}(c,\hat{p}_k)\right)\right] - \sum_{k=C''}^{K}\mathbb{E}\left[\left(V_1^{\pi^k}(r,\hat{p}_k) - \frac{\lambda_k}{\eta_k}V_1^{\pi^k}(c,\hat{p}_k)\right)\right] \\ &\leq \sum_{k=C''}^{K}\mathbb{E}\left[\frac{\lambda_k}{\eta_k}\left(V_1^{\pi^{\epsilon_k,\hat{p}_k}}(c,\hat{p}_k) - V_1^{\pi_k}(c,\hat{p}_k)\right)\right] + 0 \quad (\text{By optimality of } \pi_k \text{ in primal update }) \\ &\leq \sum_{k=C''}^{K}\mathbb{E}\left[\frac{\lambda_k}{\eta_k}\left(\tau - \epsilon_k - V_1^{\pi_k}(c,\hat{p}_k)\right)\right] \\ &\leq \sum_{k=C''}^{K}\mathbb{E}\left[\frac{1}{\eta_k}((\lambda_k(\lambda_{k+1}-\lambda_k)+\tau^2)\right] (\text{By update rule for } \lambda_k) \\ &\leq \mathbb{E}\left[\sum_{k=C''}^{K}\frac{1}{\eta_k}(\frac{\lambda_k^2}{2} - \frac{\lambda_{k+1}^2}{2}) + \sum_{k=C''}^{K}\frac{1}{2\eta_k}(\lambda_{k+1}-\lambda_k)^2 + \sum_{k=C''}^{K}\frac{\tau^2}{\eta_k}\right] \\ &\leq \mathbb{E}\left[\frac{(\lambda_{C''})^2}{2\eta_{C''}}\right] + \sum_{k=C''}^{K}\frac{H^2}{2\eta_k} + \sum_{k=C''}^{K}\frac{H^2}{\eta_k} (\text{As } \eta_k \text{ increases with } k) \\ &\leq \frac{(\sum_{k=1}^{C''}\epsilon_k + C''(H-\tau))^2}{2\eta_{C''}} + \frac{3H}{2}\sum_{C''}^{K}\frac{1}{(\tau-c_0)\sqrt{k}} \\ &= \tilde{\mathcal{O}}\left(\frac{H}{\tau-c^0}\sqrt{K}\right)\end{aligned}$$

□

## B Experiment Setup

We consider the setting of a media streaming service [Bura et al., 2022] from a wireless base station. The base station provides the streaming service at two different speeds. These speeds follow independent Bernoulli distributions denoted by parameters $\mu_1 = 0.9$ and $\mu_2 = 0.1$, with $\mu_1$ corresponding to the faster service. The data packets arriving at the device are stored in a buffer and sent out according to a Bernoulli random process with mean $\gamma$. The buffer size $s_h$ evolves as $s_{h+1} = \min\left(\max\left(0, s_h + A_h - B_h\right), N\right)$ where $A_h$ is the number of packet arrivals, $B_h$ is the number of packet departures, and $N = 10$ is the maximum size of the buffer. The device desires to minimize the cost of running out of packets, i.e., an empty buffer, while restricting the use of the faster service. We model this scenario as a finite horizon CMDP with the state representing the buffer size and actions $\{1, 2\}$ denoting the choice of speed. We set the objective cost as $r(s, a) = \mathbb{1}\{s = 0\}$ and the constraint cost as $c(s, a) = \mathbb{1}\{a = 1\}$. The episode length $H$ is 10 and the threshold $\tau$ is 5.

The algorithms are evaluated over $K = 400,000$ episodes. All the experiments are performed on a 2019 MacBook Pro with 1.4 GHz Quad-Core Intel Core i5 processor and 16GB RAM.

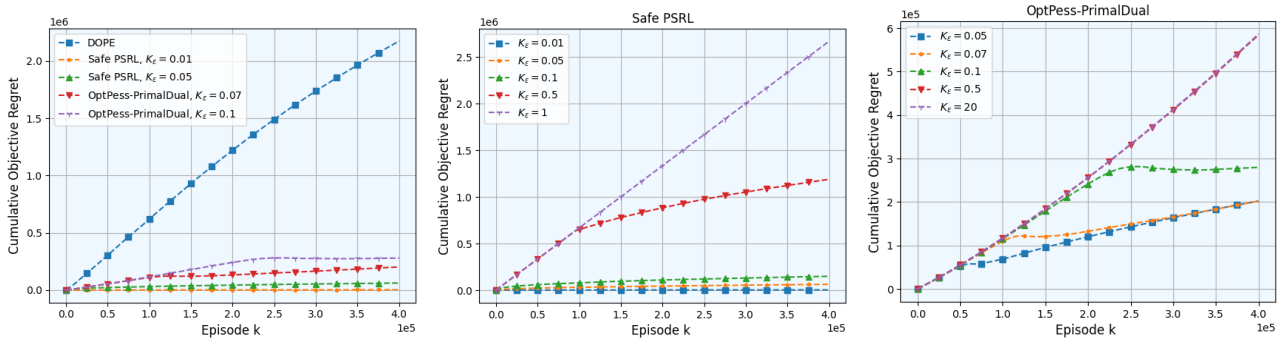
## C Additional Experiments

We consider an inventory control problem [Bertsekas, 2015]. It is modeled as a finite horizon CMDP, with episode length $H = 7$, where each time $h$ represents a day of the week. The goal of the CMDP is to maximize the expected total revenue, while keeping the expected total costs below a desired threshold.

The store has a maximum capacity of $N = 5$. The state of the environment $s_h$ denotes the amount of inventory available on the $h^{th}$ day. The action $a_h \in \{0, 1, \ldots, N - s_h\}$ is the amount of inventory purchased while taking care that the store doesn't overflow. The stochastic exogenous demand on the $h^{th}$ day is represented by a random variable $d_h$. We take $d_h$ to be in $\{0, 1, \ldots, N\}$ with distribution $[0.3, 0.2, 0.2, 0.2, 0.05, 0.05]$. The state then evolves as $s_{h+1} = \max\{0, s_h + a_h - d_h\}$.

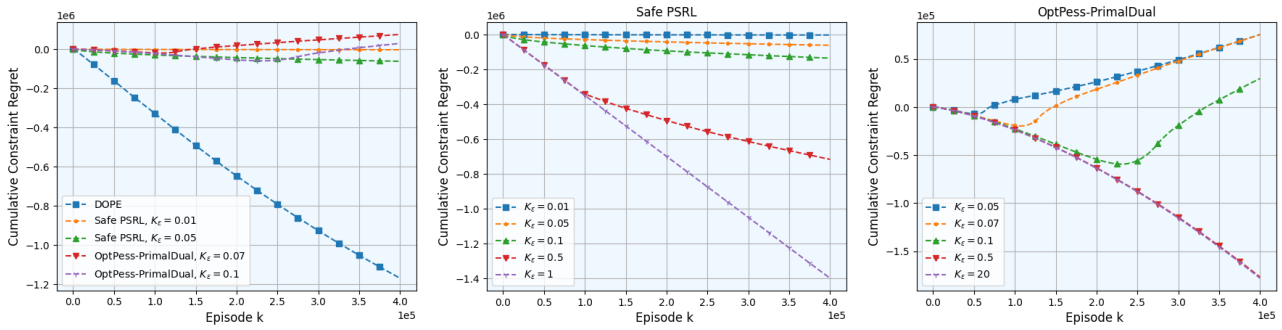
The objective reward and constraint cost functions are defined as follows. The revenue generated is defined as $f(s, a, s') = 8(s + a - s')$, when $s' > 0$ and 0 otherwise. The reward function $r(s, a)$ is the expected revenue over all next states $s'$, i.e., $r(s, a) = \mathbb{E}\left[f(s, a, s')\right]$. The cost function consists of two parts. The first part is the purchase cost when the inventory is bought. It is a fixed cost of 4 units plus a variable cost of $2a$ where $a$ is the amount of purchase. The second part is the cost for storing the inventory and is equal to $s$. Thus, the total cost function $c(s, a)$ is equal to $4 + 2a + s$. These rewards and costs are then normalized to be in the range $[0, 1]$. Finally, the goal is to maximize the expected total revenue over a week, while keeping the expected total costs in that week below a threshold $\tau = \frac{H}{2}$.

The algorithms are evaluated over $K = 400,000$ episodes. All the experiments are performed on a 2019 MacBook Pro with 1.4 GHz Quad-Core Intel Core i5 processor and 16GB RAM.




(a) (b) (c)

Figure 3: Cumulative objective regret for (a) different algorithms; (b) `Safe-PSRL` with different values of $K_\epsilon$; (c) `OptPess-PrimalDual` with different values of $K_\epsilon$.

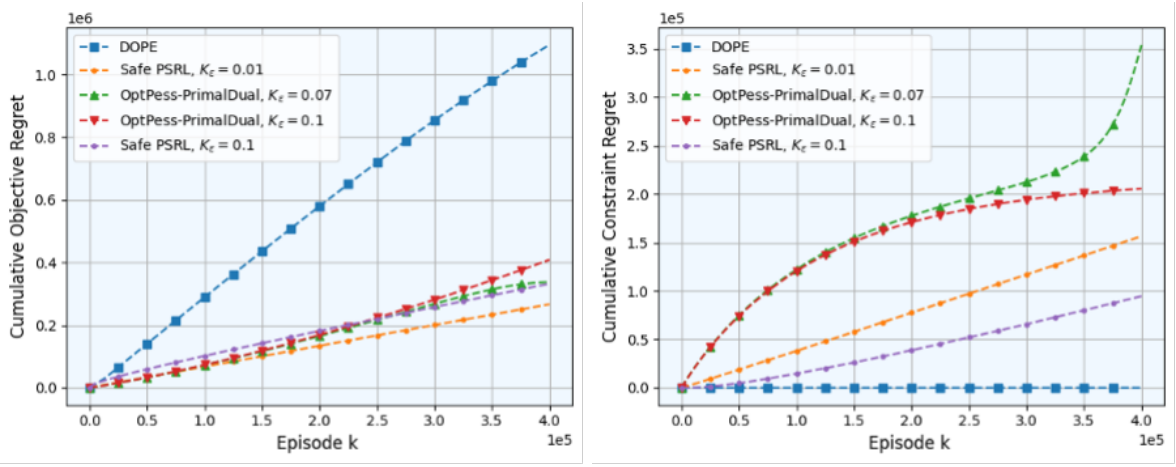



(a) (b) (c)

Figure 4: Cumulative constraint regret for (a) different algorithms; (b) `Safe-PSRL` with different values of $K_\epsilon$; (c) `OptPess-PrimalDual` with different values of $K_\epsilon$.

## D Clipped Regret Comparison

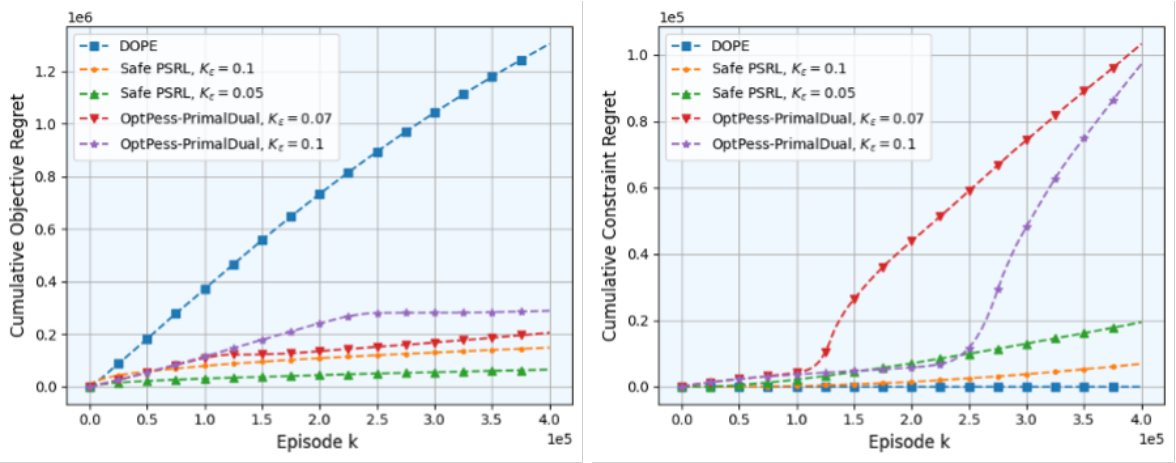
We further evaluate our algorithm using clipped regret, a stricter regret definition where negative regret terms are clipped to 0. Figures 5(a) and 6(a) illustrate that our algorithm exhibits superior performance in terms of objective regret, even under the clipped regret criterion. While `DOPE` achieves near-zero constraint regret, this relies on knowledge of a safe policy, which we do not assume. Further, despite making similar assumptions, as illustrated in figures 5(b) and 6(b), our algorithm demonstrates superior performance in empirical clipped constraint regret compared to `OptPess-PrimalDual`.




(a) (b)

Figure 5: Clipped Regret Comparison for Media streaming service setting

We would like to underscore that the clipped regret plots do not undermine the validity of our theoretical results, nor do they affect the conclusions drawn from our prior empirical experiments, which were based on the classical definition of regret. The clipped regret plots do not imply that the algorithm fails to identify a feasible policy. Had this been the case, the plots based on the classical regret definition would have exhibited an increasing trend with respect to the episode index, rather than the decreasing trend observed. We interpret the clipped regret plots as suggesting that the policy may occasionally oscillate between satisfying the constraint and violating it by a small margin. The positive regret observed in these plots can be attributed to such marginal violations, as negative contributions to regret are clipped, ultimately resulting in a small but positive regret.




(a) (b)

Figure 6: Clipped Regret Comparison for Inventory control setting

## E Code

The code used for the experiments is available at Github [Kalagarla, 2025]. We use the publicly available code provided by the authors for running the `DOPE` algorithm.