A Provably-Efficient Model-Free Algorithm for Infinite-Horizon Average-Reward Constrained Markov Decision Processes

Honghao Wei,[1] **Xin Liu,** [2] **Lei Ying** [1]

[1] University of Michigan, Ann Arbor
[2] ShanghaiTech University
honghaow@umich.com, liuxin7@shanghaitech.edu.cn, leiying@umich.com

Abstract

Model-free reinforcement learning (RL) algorithms are known to be memory and computation-efficient compared with model-based approaches. This paper presents a model-free RL algorithm for infinite-horizon average-reward Constrained Markov Decision Processes (CMDPs), which achieves sub-linear regret and zero constraint violation. To the best of our knowledge, this is the first *model-free* algorithm for general CMDPs in the infinite-horizon average-reward setting with provable guarantees.

Introduction

Reinforcement Learning has drawn significant attention due to its success in board and video games such as Go (Silver et al. 2017) and Starcraft (Vinyals et al. 2019), and in highly-complex robotics systems (Andrychowicz et al. 2020). An agent's objective in a typical RL problem is to maximize a cumulative reward through interacting with an unknown environment. In board games or video games, the consequences of a random action are limited. However, a careless action in the real-world might have catastrophic outcomes such as collisions and fatalities in robotics and autonomous driving (Ono et al. 2015; Garcia and Fernández 2012; Fisac et al. 2018) or surgical robotics (Richter, Orosco, and Yip 2019). Therefore, it is critical to strike a balance between reward maximization and safety in real-world applications. A standard formulation for RL with constraints is the Constrained Markov Decision Processes framework (Altman 1999), in which the agent aims at learning a policy that maximizes the expected cumulative reward under the safety constraints during and after learning.

There are two main classes of RL algorithms: model-based and model-free. Model-based algorithms estimate the transition kernel of the underlying CMDP during the learning process and utilize it to derive the optimal policy. For example, the estimated model can be used to formulate a linear programming (LP) problem for the CMDP (Singh, Gupta, and Shroff 2020; Brantley et al. 2020; Kalagarla, Jain, and Nuzzo 2020; Efroni, Mannor, and Pirotta 2020), or a LP problem as part of a primal-dual algorithm (Qiu et al. 2020; Efroni, Mannor, and Pirotta 2020). (Ding et al. 2020a) proposed a model-based, primal-dual algorithm with linear function approximation dealing with infinite state and action spaces under the assumption that the transition kernel is linear. Model-based RL algorithms are sample efficient and perform well when the model can be estimated precisely. However, model-based algorithms are well known for their memory complexity for storing a large amount of model parameters. Furthermore, building accurate models is very challenging computationally and data-wise (Sutton and Barto 2018).

Model-free algorithms, on the other hand, learn state or action value functions rather than the transition kernel, which require significantly smaller memory. For example, several policy-gradient algorithms (Tessler, Mankowitz, and Mannor 2018; Stooke, Achiam, and Abbeel 2020; Yang et al. 2020) have been proposed and seen successes in practice for solving constraint RL problems, but without regret and constraint violation analysis. (Ding et al. 2020b; Xu, Liang, and Lan 2020; Chen, Dong, and Wang 2021) are some exceptions, but both of these works require a simulator to either simulate the underlying MDP from any given state or to evaluate a policy. Two very recently works (Liu et al. 2021; Wei, Liu, and Ying 2021) are the most related works, they both leverage Lyapunov drift analysis to achieve regret bound and zero violation, but they focus on episodic CMDPs, whereas we are looking at infinite average-reward CMDPs, which is a harder problem. The question we seek to answer in this paper is

Can we design efficient RL algorithms for infinite-horizon, average-reward CMDPs with provably regret guarantees?

We answer this question affirmatively, and present the first model-free RL algorithm under this setting, which achieves sub-linear regret and zero constraint violation. For comparisons with other existing approaches under the same setting, see Table 1[1].

Preliminaries

An infinite-horizon average-reward CMDP can be defined as $(\mathcal{S}, \mathcal{A}, r, g, p)$, where $\mathcal{S}$ is the finite state space, $\mathcal{A}$ is the finite action space, $r(g) : \mathcal{S} \times \mathcal{A} \to [0, 1]$ is the unknown

[1] Throughout the paper, we use the notation $\tilde{\mathcal{O}}$ to suppress log terms.

Table 1: Regret comparisons for RL algorithms in infinite-horizon average-reward CMDPs with S states, A actions and K total steps. D is the diameter of the CMDP. δ is the slackness that is defined later (Eq. (10)).

	Algorithm	**Regret**	**Constraint Violations**
Known Model	C-UCRL(Zheng and Ratliff 2020)	$\tilde{\mathcal{O}}(SA\sqrt{K^{1.5}})$	0
Model-based	UCRL-CMDP (Singh, Gupta, and Shroff 2020)	$\tilde{\mathcal{O}}(S\sqrt{AK^{1.5}})$	$\tilde{\mathcal{O}}(S\sqrt{AK^{1.5}})$
Known Model	CMDP-PSRL (Agarwal, Bai, and Aggarwal 2021)	$\tilde{\mathcal{O}}(poly(SAD)\sqrt{K})$	$\tilde{\mathcal{O}}(poly(SA)\sqrt{K})$
Model-free	**this work**	$\tilde{\mathcal{O}}\left(\frac{\sqrt{SA}}{\delta}K^{\frac{5}{6}}\right)$	0

reward (utility) function, and $p : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ represents the transition probability such that $p(s'|s, a) := \mathbb{P}(s_{k+1} = s'|s_k = s, a_k = a)$ for $s_k \in \mathcal{S}, a_k \in \mathcal{A}$ and time $k = 1, 2, \ldots$. A stationary policy is a mapping $\pi : \mathcal{S} \to \mathcal{A}$, the long-term average reward (reward rate) of a stationary policy π with initial state $s \in \mathcal{S}$ is defined as

$$J_r^\pi(s) := \liminf_{K\to\infty} \frac{1}{K}\mathbb{E}\left[\sum_{k=1}^{K} r(s_k, \pi(s_k))\middle| s_1 = s\right],$$

and the long-term average utility (utility rate) is such policy can be denoted by

$$J_g^\pi(s) := \liminf_{K\to\infty} \frac{1}{K}\mathbb{E}\left[\sum_{k=1}^{K} g(s_k, \pi(s_k))\middle| s_1 = s\right].$$

In this paper, we consider the constrained RL problem in which an agent interacts with the CMDP through K steps, starting from an arbitrary initial state $s_1 \in \mathcal{S}$. At each step k, the agent observes the state s_k, decides an action a_k and receives the reward $r(s_k, a_k)$ and utility $g(s_k, a_k)$. The next state s_{k+1} is then sampled according to the probability distribution $p(\cdot|s_k, a_k)$. The goal of the learning problem is to learn a policy that maximizes the reward rate subject to a constraint on the utility rate:

$$\begin{aligned} &\max_{\pi\in\Pi} \liminf_{K\to\infty} \frac{1}{K}\mathbb{E}_\pi\left[\sum_{k=1}^{K} r(s_k, a_k)\right] \\ &\text{s.t.} \liminf_{K\to\infty} \frac{1}{K}\mathbb{E}_\pi\left[\sum_{k=1}^{K} g(s_k, a_k)\right] \geq \rho, \end{aligned} \tag{1}$$

where Π is set of all possible policies and $\rho \in (0, 1)$ to avoid triviality. Let π^* be the optimal policy of the CMDP problem in (1). The performance of the learning algorithm is evaluated through regret and constraint violation defined below:

$$\text{Regert}(K) = \sum_{k=1}^{K}\left(J_r^{\pi^*} - r(s_k, a_k)\right), \tag{2}$$

$$\text{Violation}(K) = \sum_{k=1}^{K}\left(\rho - g(s_k, a_k)\right). \tag{3}$$

Regret measures the difference between the total reward of the optimal policy and that obtained by a learning algorithm and violation evaluates the difference between the total utility collected by a learning algorithm and the requirement. To analyze the regret and constraint violation, we make the following assumption throughout the paper:

Assumption 1. *The MDP is an unichain MDP, which means for any stationary deterministic policy π, the Markov chain induced by π contains a single (aperiodic) ergodic class.*

Assumption 1 is necessary for ensure that the optimal policy π^* of the CMDP is independent of the state, i.e. $J_r^{\pi^*}(s) = J_r^{\pi^*}, J_g^{\pi^*}(s) = J_g^{\pi^*}$ for all $s \in \mathcal{S}$, and the optimality is achievable by using linear programming (LP) approach (Altman 1999) (defined below). This assumption is commonly used in infinite-horizon average-reward CMDPs (Wei et al. 2020; Ortner 2020; Abbasi-Yadkori et al. 2019).

When the transition kernel $p(s'|s, a)$ is known, an optimal policy can be obtained by solving the following LP problem,

$$\max_{\{q(s,a):(s,a)\in\mathcal{S}\times\mathcal{A}\}} \sum_{s,a} q(s, a)r(s, a) \tag{4}$$

$$\text{s.t.} \sum_{s,a} q(s, a)g(s, a) \geq \rho, \forall s \in \mathcal{S}, \forall a \in \mathcal{A} \tag{5}$$

$$q(s, a) \geq 0, \forall s \in \mathcal{S}, \forall a \in \mathcal{A} \tag{6}$$

$$\sum_{s,a} q(s, a) = 1 \tag{7}$$

$$\sum_{a} q(s, a) = \sum_{s',a'} p(s|s', a')q(s', a'), \tag{8}$$

where the $q(s, a)$ is called the occupancy measure, which is defined as the set of distributions generated by executing the associated induced policy π in the infinite-horizon CMDP. $\sum_a q(s, a)$ represents the probability the system is in state s, and $\frac{q(s,a)}{\sum_{a'} q(s,a')}$ is the probability of taking action a in state s. The utility constraint is defined as (5). More details can be found in (Altman 1999). To analyze the performance of our algorithm, we need to consider a tightened version of the above LP problem, which is defined below:

$$\max_{\{q(s,a):(s,a)\in\mathcal{S}\times\mathcal{A}\}} \sum_{s,a} q(s, a)r(s, a) \tag{9}$$

$$\text{s.t.} \sum_{s,a} q(s, a)g(s, a) \geq \rho + \epsilon, \forall s \in \mathcal{S}, \forall a \in \mathcal{A}$$

$$(6) - (8),$$

where $\epsilon > 0$ is called a tightness constant. As in previous works (Ding et al. 2020a,b; Efroni, Mannor, and Pirotta 2020; Paternain et al. 2019), we make the following standard assumption of Slater's condition.

Assumption 2. *(Slater's Condition). There exist $\delta > 0$ such that*

$$\sum_{s,a} q(s,a)g(s,a) - \rho \geq \delta. \tag{10}$$

It is obvious that when $\epsilon < \delta$ the problem (9) has a feasible solution due to Slater's condition. We remark that this assumption is a noticeable difference from some existing works in which the agent needs to know the value of this constant (e.g. (Ding et al. 2020a)) or alternatively a feasible policy (e.g. (Achiam et al. 2017)).

Let

$$J_r^* = \sum_{s,a} q^*(s,a)r(s,a) \tag{11}$$

$$J_g^* = \sum_{s,a} q^*(s,a)g(s,a). \tag{12}$$

be the optimal reward rate and utility rate, where $q^*(s,a)$ is the optimal solution obtained by solving the LP problem(4). Moreover it is obvious that J_r^*, J_g^* are independent of the start state and we have $J_r^* = J_r^{\pi^*}, J_g^* = J_g^{\pi^*}$.

In the following, we use superscript * to denote the optimal policy achieved by solving the LP (4) of the original CMDP, and superscript $^{\epsilon,*}$ to denote the optimal policy related to the ϵ-tightened version of LP (9).

Primal-Dual-Based, Two-Time-Scale Optimistic SARSA

In this section, we introduce our algorithm (see Algorithm 1 for pseudo-code) which achieves sub-linear regret and zero constraint violation. The algorithm is inspired by (Wei et al. 2020) to solve the average-reward CMDP via designing an algorithm for a discounted CMDP which is defined with the same states, actions, reward/utility function, transition kernels, and an extra discount factor γ. The intuition is that the reward of the discounted problem (scaled by $1-\gamma$)) approaches to the reward of the average reward problem as γ goes to 1.

Under the discounted CMDP setting, given a policy π, the reward value function V_k^π at step k is the expected cumulative rewards from step k under policy π :

$$V_k^\pi(s) = \mathbb{E}\left[\sum_{i=k}^{\infty} \gamma^{i-k} r(s_i, \pi(s_i)) \middle| s_k = s\right].$$

The reward Q-function $Q_k^\pi(s,a)$ at step k is the expected cumulative rewards when agent starts from a state-action pair (s,a) at step k and then follows policy π :

$$Q_k^\pi(s,a) = r(s,a) + \mathbb{E}\left[\sum_{i=k}^{\infty} \gamma^{i-k} r(s_i, \pi(s_i)) \middle| \begin{array}{c} s_k = s \\ a_k = a \end{array}\right].$$

Algorithm 1: Algorithm

Initialize $Q_1(s,a) = \hat{Q}_1(s,a) \leftarrow H$ and $n_1(s,a) \leftarrow 0$ for all $(s,a) \in \mathcal{S} \times \mathcal{A}, \gamma = 1 - \frac{1}{H}, \hat{V}_1(s) = H, \forall s \in \mathcal{S}$
Choose $\chi = K^{\frac{1}{3}}, \eta = K^{\frac{1}{6}}, \iota = 8\log(\sqrt{2}K), \beta = \frac{2}{3}$.
Choose $\epsilon = \frac{9\kappa\sqrt{SA\iota}}{K^{\frac{1}{6}}}, \kappa = \max\{sp(v^{\epsilon,*}), sp(w^{\epsilon,*}), 1\}$.
Initialize $\bar{C} \leftarrow 0, Z_1 \leftarrow 0$.
Define , $\alpha_k = \frac{\chi+1}{\chi+\tau}, b_\tau = \kappa\sqrt{\frac{(\chi+1)\iota}{\chi+\tau}}$.
for episode $k = 1, \ldots, K$ **do**
    Take action
$$a_k = \arg\max_a \left(\hat{Q}_k(s_k,a) + \frac{Z}{\eta}\hat{C}_k(s_k,a)\right).$$
    Observe s_{k+1}.
    $n_{k+1}(s_k,a_k) \leftarrow n_k(s_k,a_k)+1, \tau \leftarrow n_{k+1}(s_k,a_k)$.
    Update
    $Q_{k+1}(s_k,a_k) \leftarrow (1-\alpha_\tau)Q_k(s_k,a_k)$
        $+\alpha_\tau[r(s_k,a_k) + \gamma\hat{V}_k(s_{k+1}) + b_\tau]$,
    $C_{k+1}(s_k,a_k) \leftarrow (1-\alpha_\tau)C_k(s_k,a_k)$
        $+\alpha_\tau[g(s_k,a_k) + \gamma\hat{W}_k(s_{k+1}) + b_\tau]$.
    if $Q_{k+1}(s_k,a_k) \leq \hat{Q}_k(s_k,a_k)$ and $C_{k+1}(s_k,a_k) \leq \hat{C}_k(s_k,a_k)$ **then**
        $\hat{Q}_{k+1}(s_k,a_k) \leftarrow Q_{k+1}(s_k,a_k)$
        $\hat{C}_{k+1}(s_k,a_k) \leftarrow C_{k+1}(s_k,a_k)$
    else
        $\hat{Q}_{k+1}(s_k,a_k) \leftarrow \hat{Q}_k(s_k,a_k)$
        $\hat{C}_{k+1}(s_k,a_k) \leftarrow \hat{C}_k(s_k,a_k)$
    $\bar{C} \leftarrow \bar{C} + (1-\gamma)\hat{C}_k(s_k,a_k)$
    $a' = \arg\max_a \left(\hat{Q}_{k+1}(s_k,a) + \frac{Z}{\eta}\hat{C}_{k+1}(s_k,a)\right)$
    $\hat{V}_{k+1}(s_k) \leftarrow \hat{Q}_{k+1}(s_k,a')$
    $\hat{W}_{k+1}(s_k) \leftarrow \hat{C}_{k+1}(s_k,a')$
    if $t \bmod K^\beta = 0$ **then**
        $Z \leftarrow \left(Z + \rho + \epsilon - \frac{\bar{C}}{K^\beta}\right)$
        Reset $\bar{C} \leftarrow 0, n_t(s,a) \leftarrow 0$.
        $\hat{Q}_{k+1}(s,a) \leftarrow \hat{Q}_{k+1}(s,a) + \frac{4H}{\eta}, \forall(s,a)$
        $Q_{k+1}(s,a) \leftarrow Q_{k+1}(s,a) + \frac{4H}{\eta}, \forall(s,a)$
        if $\hat{Q}_{k+1}(s,a) > H$ or $\hat{C}_{k+1}(s,a) > H$ **then**
            Reset $\hat{Q}_{k+1}(s,a), Q_{k+1}(s,a), V_{k+1}(s)$ to H
            Reset $\hat{C}_{k+1}(s,a), C_{k+1}(s,a), W_{k+1}(s)$ to H

Similarly, we use $W_k^\pi(s) : \mathcal{S} \to \mathbb{R}^+$ and $C_k^\pi(s,a) : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^+$ to denote the utility value function and utility Q-function at step k:

$$W_k^\pi(x) = \mathbb{E}\left[\sum_{i=k}^{\infty} \gamma^{i-k} g(s_i, \pi_i(s_i)) \middle| s_k = s\right],$$

$$C_k^\pi(s,a) = g(s,a) + \mathbb{E}\left[\sum_{i=k}^{\infty} \gamma^{i-k} g(s_i, \pi(s_i)) \middle| \begin{array}{c} s_k = s \\ a_k = a \end{array}\right].$$

It is obvious that all the reward and utility value (Q-value) functions are bounded by $\frac{1}{1-\gamma}$ due to the fact reward and

utility functions are bounded by 1. We denote $H = \frac{1}{1-\gamma}$. Then given a state-action pair (s, a) at step k, our algorithm updates the estimate of reward (utility) $Q-$value functions of the discounted CMDP setting instead.

The design of our algorithm is based on the primal-dual approach for constrained optimization problems. Suppose that $V^\pi(s)(W^\pi(s))$ is a accurate estimate of $\frac{J_r^\pi(s)}{1-\gamma}(\frac{J_g^\pi(s)}{1-\gamma})$ (the formal proof is deferred to next section). Given Lagrangian multiplier μ, we consider the following problem:

$$\begin{aligned}&\max_\pi J_r^\pi(s) + \mu(J_g^\pi(s) - \rho)\\ \approx &\max_\pi (1-\gamma)(V^\pi(s) + \mu W^\pi(s)) - \mu\rho\end{aligned}$$

which can be interpreted as an unconstrained MDP with a modified reward function $(1-\gamma)(r + \mu g)$.

Specifically, the algorithm maintains an estimate $\hat{V}_k(s)(\hat{W}_k(s))$ for the optimal value function $V^*(s)(W^*(s))$ and $\hat{Q}_k(s,a)(\hat{C}_k(s,a))$ for the optimal Q-function $Q^*(s,a)(C^*(s,a))$. At each step k, after observing state s, the agent selects action a_k^* based on the combined Q-value:

$$a_k^* \in \arg\max_a \hat{Q}_k(s,a) + \frac{Z}{\eta}\hat{C}_k(s,a), \tag{13}$$

where $\frac{Z}{\eta}$ can be treated as an estimate of the Lagrange multiplier μ. Although primal-dual is a standard approach and is heavily used in previous works, analyzing the regret or constraint violation is particularly challenging. We need to consider how to carefully choose the Lagrange multiplier and how often it will be updated. Too fast would lead to divergence but too slow would delay the speed of convergence which tends to have a large regret and constraint violation. We address these difficulties by designing our algorithm as a two-time-scale algorithm where Z is updated at a slow timescale, i.e., every K^β steps in line $25-26$ in Algorithm 1. In particular,

$$Z \leftarrow \left(Z + \rho + \epsilon - \frac{\bar{C}}{K^\beta}\right)^+, \tag{14}$$

where $(x)^+ = \max\{x, 0\}$, and $\bar{C}$ is the summation of all $(1-\gamma)\hat{C}_k(s_k, a_k)$ of the steps in the previous frame, where each frame consists of K^β consecutive steps.

On the contrary, it learns the combined Q functions for fixed Z at a fast time scale. The estimates of reward and utility value functions are updated each time after observing a new state-action pair.

It is important to mention that the optimal policy may not satisfy the optimality principle due to the constraints. More specifically,

$$V^*(s) \neq \max_a Q^*(s,a). \tag{15}$$

This means we cannot leverage optimistic Q-learning algorithms for unconstrained MDPs (Jin et al. 2018; Wei et al. 2020; Dong et al. 2019) to estimate the optimal value functions of CMDP. Instead, our algorithm uses a SARSA-type updating rule, as shown in line $11-14$.

We remark that the optimal policy for a CMDP is stochastic in general. The policy under our algorithm is a stochastic policy because the virtual queue Z varies during and after the learning process, which results in a stochastic policy.

We then introduce some additional notations before presenting our main theorem. Let $v^\pi(s), w^\pi(s)$ be the reward and utility relative value functions for state s under average-reward setting, and $q^\pi(s,a), c^\pi(s,a)$ be the reward and utility Q value functions for any state-action pair (s,a). Based on the Bellman equation, we have

$$\begin{aligned}J_r^\pi(s) + q^\pi(s,a) &= r(s,a) + \mathbb{E}_{s'\sim p(\cdot|s,a)}[v^\pi(s')]\\ v^\pi(s) &= \sum_a q^\pi(s,a)\mathbb{P}(\pi(s)=a)\\ J_g^\pi(s) + c^\pi(s,a) &= g(s,a) + \mathbb{E}_{s'\sim p(\cdot|s,a)}[w^\pi(s')]\\ w^\pi(s) &= \sum_a c^\pi(s,a)\mathbb{P}(\pi(s)=a)\end{aligned}$$

Define

$$sp(f) = \max_{s\in\mathcal{S}} f(s) - \min_{s\in\mathcal{S}} f(s) \tag{16}$$

to be the span of the function f. It is well known that the span of the optimal reward relative value function $sp(v^*)$ and utility relative value function $sp(w^*)$ are bounded for weakly communication or ergodic MDPs. In particular, they are bounded by the diameter of the MDP (Lattimore and Szepesvári 2020).

Let $\kappa = \max\{sp(v^{\epsilon,*}), sp(w^{\epsilon,*}), 1\}$ be a upper bound for convenience. We assume that $sp(v^{\epsilon,*}), sp(w^{\epsilon,*})$ which are used in the algorithm are known beforehand as (Wei et al. 2020, 2021) throughout the paper, we can always substitute them with any upper bound (e.g. the diameter) when they are unknown.

We now state the main result guarantee of Algorithm 1.

Theorem 1. *Under assumption* $K \geq \left(\frac{18\kappa\sqrt{SA\iota}}{\delta}\right)^6$. *Let* $\epsilon = \frac{9\kappa\sqrt{SA\iota}}{K^{\frac{1}{6}}}$ *such that* $\epsilon \leq \frac{\delta}{2}$. *By choosing* $m = K^{\frac{1}{6}}\log K, H = K^{\frac{1}{6}}, \eta = K^{\frac{1}{6}}, \chi = K^{\frac{1}{3}}, \beta = \frac{2}{3}$ *we have* $H \geq \frac{6\kappa}{\delta}$, *Then*

$$\begin{aligned}\text{Regret}(K) &\leq \tilde{O}\left(\frac{\sqrt{SA}\kappa}{\delta}K^{\frac{5}{6}}\right)\\ \text{Violation}(K) &\leq \frac{92K^{\frac{2}{3}}}{\delta}\log\left(\frac{24}{\delta}\right) - \sqrt{SA\iota}K^{\frac{5}{6}} = 0,\end{aligned}$$

where $\iota = 32\log(\sqrt{2}K)$.

Proof of the Main Theorem

Notations

Throughout the paper, we use shorthand notation

$$\{f-g\}(x) = f(x) - g(x),$$

where $f(\cdot)$ and $g(\cdot)$ the the same argument value. Similarly,

$$\{(f-g)q\}(x) = (f(x) - g(x))q(x).$$

Due to the page limit, we will only present several key lemmas and the key intuitions in this section. The complete proof can be found in the appendix.

Regret Analysis

We start the proof by adding and subtracting the corresponding terms to the regret defined in (2), we obtain

$$\begin{aligned}
&\text{Regret}(K) = \mathbb{E}\left[\sum_{k=1}^{K}(J_r^* - r(s_k,a_k))\right] \\
=&\mathbb{E}\left[\sum_{k=1}^{K}(J_r^* - J_r^{\epsilon,*})\right] && (17)\\
+&\mathbb{E}\left[\sum_{k=1}^{K}(J_r^{\epsilon,*} - (1-\gamma)V^{\epsilon,*}(s_k))\right] && (18)\\
+&\mathbb{E}\left[\sum_{k=1}^{K}(1-\gamma)\left(V^{\epsilon,*}(s_k) - \hat{Q}_k(s_k,a_k)\right)\right] && (19)\\
+&\mathbb{E}\left[\sum_{k=1}^{K}\left((1-\gamma)\hat{Q}_k(s_k,a_k) - r(s_k,a_k)\right)\right]. && (20)
\end{aligned}$$

We will bound each of the four terms above in the following sequence of lemmas.

Term (17) is the difference between original CMDP and its corresponding $\epsilon-$tighten version which is a perturbation of the original problem. We establish a bound by using the following lemma:

Lemma 2. *Under assumption 2, given $\epsilon \le \delta$, we have*

$$\sum_{t=1}^{K}(J_r^* - J_r^{\epsilon,*}) \le \frac{\epsilon K}{\delta} \tag{21}$$

For the second term (18), we establish a bound by using Lemma 3, which shows the difference of value functions under average-reward setting and discounted setting is small, the proof is based on the Bellman equation under two settings.

Lemma 3. *For an arbitrary policy π, we have*

$$\begin{aligned}
J_r^\pi(s) - (1-\gamma)V^\pi(s) &\le (1-\gamma)sp(v^\pi(s)), && (22)\\
|V^\pi(s_1) - V^\pi(s_2)| &\le 2sp(v^\pi(s)); && (23)\\
J_g^\pi(s) - (1-\gamma)W^\pi(s) &\le (1-\gamma)sp(w^\pi(s)), && (24)\\
|W^\pi(s_1) - W^\pi(s_2)| &\le 2sp(w^\pi(s)), && (25)
\end{aligned}$$

where $V^\pi(s)$ is the value function for the discounted setting under policy π, and $J_r^\pi(J_g^\pi)$ is the reward (utility) rate under policy π.

Then it is easy to obtain

$$J_r^{\epsilon,*} - (1-\gamma)V^{\epsilon,*}(s) \le (1-\gamma)\kappa, \tag{26}$$

Next we establish a bound on term (19) by using Lyapunov-drift analysis. This term is not hard to be addressed in the unconstrained MDPs, because using optimistic Q-learning guarantees that $\hat{Q}_k(s,a)$ is an overestimate of $Q^*(s,a)$. However this inequality does not hold in CMDPs, because the algorithm needs to consider reward and utility simultaneously which makes the analysis difficult. To bound this term, we first add and subtract some addition terms to have

$$\begin{aligned}
&\sum_{k=1}^{K}(1-\gamma)\left(V^{\epsilon,*}(s_k) - \hat{Q}_k(s_k,a_k)\right)\\
=&\sum_{k=1}^{K}(1-\gamma)\sum_a\left\{Q^{\epsilon,*}q^{\epsilon,*} + \frac{Z_k}{\eta}C^{\epsilon,*}q^{\epsilon,*}\right\}(s_k,a) && (27)\\
-&\sum_{k=1}^{K}(1-\gamma)\sum_a\left\{\hat{Q}_k q^{\epsilon,*} + \frac{Z_k}{\eta}\hat{C}_k q^{\epsilon,*}\right\}(s_k,a) && (28)\\
+&\sum_{k=1}^{K}(1-\gamma)\left(\sum_a\left\{\hat{Q}_k q^{\epsilon,*}\right\}(s_k,a) - \hat{Q}_k(s_k,a_k)\right. && (29)\\
&\left.+\frac{Z_k}{\eta}\sum_a\left\{\hat{C}_k q^{\epsilon,*} - C^{\epsilon,*}q^{\epsilon,*}\right\}(s_k,a)\right). && (30)
\end{aligned}$$

We can see (27) + (28) is the difference of two combined Q functions. We will show that $\left\{\hat{Q}_k + \frac{Z_k}{\eta}\hat{C}_{k,h}\right\}(s,a)$ is always an over-estimate of $\left\{Q^{\epsilon,*} + \frac{Z_k}{\eta}C^{\epsilon,*}\right\}(s,a)$ (i.e. (27) + (28) ≤ 0) for all (s,a,k) simultaneously with a high probability using the following Lemma 4. This result further implies an expected upper bound

$$\mathbb{E}\left[(27)+(28)\right] \le (1-\gamma)\frac{3H}{\eta K}. \tag{31}$$

Lemma 4. *With probability at least $1 - \frac{1}{K^3}$, the following inequality holds simultaneously for all $(s,a,k) \in \mathcal{S}\times\mathcal{A}\times[K]$:*

$$\left\{\left(\hat{Q}_k - Q^{\epsilon,*}\right) + \frac{Z_k}{\eta}\left(\hat{C}_k - C^{\epsilon,*}\right)\right\}(s,a) \ge 0, \tag{32}$$

Then for the rest term (29) + (30), we can bound it by using the following lemma:

Lemma 5. *Assuming $\epsilon < \delta$, we have*

$$\begin{aligned}
&\mathbb{E}\left[\sum_{k=1}^{K}(1-\gamma)\left(\sum_a\left\{\hat{Q}_k q^{\epsilon,*}\right\}(s_k,a) - \hat{Q}_k(s_k,a_k)\right.\right.\\
&\left.\left.+\frac{Z_k}{\eta}\sum_a\left\{\hat{C}_k q^{\epsilon,*} - C^{\epsilon,*}q^{\epsilon,*}\right\}(s_k,a)\right)\right]\\
\le&\frac{2K}{\eta} + \sum_{T=1}^{K^{1-\beta}}\mathbb{E}\left[Z_T\right]\frac{(1-\gamma)\kappa}{\eta}. && (33)
\end{aligned}$$

To see the idea behind Lemma 5, we need to consider the Lyapunov function $L_T = \frac{1}{2}Z_T^2$, where T is the frame index and Z_T is the virtual-queue length at the beginning of Tth frame. Recall that each frame contains K^β consecutive steps. In the proof of Lemma 5, we will show that the Lyapunov-drift satisfies

$$\begin{aligned}
&\mathbb{E}[L_{T+1} - L_T] \le \text{a negative drift}\\
&+2 + \mathbb{E}\left[Z_T\right]\frac{(1-\gamma)\kappa}{K^\beta} - \frac{\eta}{K^\beta}\sum_{k=TK^\beta+1}^{(T+1)K^\beta}\Phi_k, && (34)
\end{aligned}$$

where

$$\Phi_k =(1-\gamma)\left(\sum_a \left\{\hat{Q}_k q^{\epsilon,*}\right\}(s_k,a) - \hat{Q}_k(s_k,a_k) + \frac{Z_k}{\eta}\sum_a \left\{\hat{C}_k q^{\epsilon,*} - C^{\epsilon,*} q^{\epsilon,*}\right\}(s_k,a)\right)$$

Then take submission on both sides of equation over all the $K^{1-\beta}$ frames, we can obtain

$$\begin{aligned}&\mathbb{E}[L_1 - L_{K^{1-\beta}+1}]\\ \leq& 2K^{1-\beta} + \sum_{T=1}^{K^{1-\beta}} \mathbb{E}\left[Z_T\right]\frac{(1-\gamma)\kappa}{K^\beta} - \frac{\eta}{K^\beta}\sum_k \Phi_k\end{aligned}$$

Therefore

$$\begin{aligned}&(29)+(30) = \sum_k \Phi_k\\ \leq& \frac{K^\beta \mathbb{E}[L_1 - L_{K^{1-\beta}+1}]}{\eta} + \frac{2K}{\eta} + \sum_{T=1}^{K^{1-\beta}} \mathbb{E}\left[Z_T\right]\frac{(1-\gamma)\kappa}{\eta}\\ \leq& \frac{2K}{\eta} + \sum_{T=1}^{K^{1-\beta}} \mathbb{E}\left[Z_T\right]\frac{(1-\gamma)\kappa}{\eta},\end{aligned}$$

where the last inequality holds because $L_1 = 0$ and $L_T \geq 0$ for all T.

Then combining the result form (31) and Lemma 5 we can obtain

$$\begin{aligned}&\sum_{k=1}^{K}((1-\gamma)\left(V^{\epsilon,*}(s_k) - \hat{Q}_k(s_k,a_k)\right))\\ \leq& \frac{2K}{\eta} + \sum_{T=1}^{K^{1-\beta}} \mathbb{E}\left[Z_T\right]\frac{(1-\gamma)\kappa}{\eta} + \frac{3H}{\eta K}.\end{aligned} \tag{35}$$

The term $\mathbb{E}[Z_T]$ is proved uniformly bounded in the following Lemma 6 by using Lyapunov-drift analysis on the moment generating function of Z i.e. $\mathbb{E}[e^r Z]$ that holds uniformly over the entire learning process. The reason is that our algorithm takes actions to almost greedily reduce the virtual-queue Z when Z is large.

Lemma 6. *Assuming $\epsilon \leq \frac{\delta}{2}, H \geq \frac{6\kappa}{\delta}$, we have for any $1 \leq T \leq K^{1-\beta}$,*

$$\mathbb{E}[Z_T] \leq \frac{92}{\delta}\log\left(\frac{24}{\delta}\right) + \frac{6\eta}{\delta}$$

We apply the following lemma to bound the last term (20).

Lemma 7. *For any $T \in [K^{1-\beta}]$,*

$$\begin{aligned}&\mathbb{E}\left[\sum_{k=(T-1)K^\beta+1}^{TK^\beta}\left((1-\gamma)\hat{Q}_k(s_k,a_k) - r(s_k,a_k)\right)\right]\\ \leq& \gamma^m K^\beta + \frac{K^\beta m}{\chi} + 4(1-\gamma)m\kappa\sqrt{(\chi+1)SAK^\beta\iota} + 2mS\\ &\mathbb{E}\left[\sum_{k=(T-1)K^\beta+1}^{TK^\beta}\left((1-\gamma)\hat{C}_k(s_k,a_k) - g(s_k,a_k)\right)\right]\\ \leq& \gamma^m K^\beta + \frac{K^\beta m}{\chi} + 4(1-\gamma)m\kappa\sqrt{(\chi+1)SAK^\beta\iota} + 2mS,\end{aligned}$$

where m is a positive integer.

This lemma is one of our key technical contributions which show that the cumulative estimation error over one frame (K^β consecutive episodes) between weighted reward(utility) Q-value functions and average reward (utility) is upper bounded. From the lemma above, we can immediately conclude that:

$$\begin{aligned}&\mathbb{E}\left[\sum_{k=1}^{K}\left((1-\gamma)\hat{Q}_t(s_k,a_k) - r(s_k,a_k)\right)\right] \leq \gamma^m K + \frac{Km}{\chi}\\ &+ 4(1-\gamma)m\kappa\sqrt{(\chi+1)SAK^{2-\beta}\iota} + 2mSK^{1-\beta}\end{aligned} \tag{36}$$

To balance the terms in regret, we carefully select that

$$m = H = K^{\frac{1}{6}}\log K, \chi = K^{\frac{1}{3}}, \beta = \frac{2}{3}.$$

Then we have

$$\gamma^m = \left(1 - \frac{1}{H}\right)^{H\log K} \leq \frac{1}{K},$$

and the order of the second and third terms in the above equation (36) is $\tilde{O}(K^{\frac{5}{6}})$, which is also the dominate term in our regret bound.

Then by appropriately choosing other parameters ϵ, ι, η and combing the results from (35), (36), Lemma 2, Lemma 3, and Lemma 6 we finish the proof for regret.

Constraint Violation Analysis

Recall that we use Z_T to denote the value of virtual-queue in frame T. According to the update of virtual-queue length, we have

$$\begin{aligned}Z_{T+1} =& \left(Z_T + \rho + \epsilon - \frac{\bar{C}_T}{T^\beta}\right)^+\\ \geq& Z_T + \rho + \epsilon - \frac{\bar{C}_T}{K^\beta},\end{aligned} \tag{37}$$

which implies that

$$\begin{aligned}&\sum_{k=(T-1)K^\beta+1}^{TK^\beta}(-g(s_k,a_k)+\rho) \leq K^\beta\left(Z_{T+1} - Z_T\right)\\ &+ \sum_{k=(T-1)K^\beta+1}^{TK^\beta}\left((1-\gamma)\hat{C}_k(s_k,a_k) - g(s_k,a_k) - \epsilon\right).\end{aligned}$$

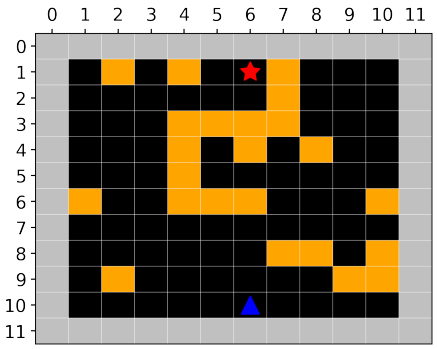

Figure 1: Grid World with Safety Constraints

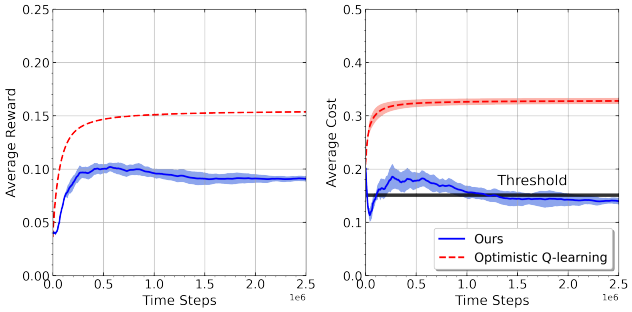

Figure 2: Average reward and cost of our algorithm during training, and the shaded region represents the standard deviations.

Summing the inequality above over all frames and taking expectation on both sides, we obtain the following upper bound on the constraint violation:

$$\mathbb{E}\left[\sum_{t=1}^{T}\rho - g(s_k,a_k)\right] \leq -K\epsilon + K^{\beta}\mathbb{E}\left[Z_{K^{1-\beta}+1}\right] + \mathbb{E}\left[\sum_{k=1}^{K}(1-\gamma)\hat{C}_k(s_k,a_k) - g(s_k,a_k)\right], \quad (38)$$

where we used the fact $Z_1 = 0$. Combining the upper bound on the estimation error of $\hat{C}_k$ in Lemma 7 and the upper bound on $\mathbb{E}[Z_T]$ in Lemma 6 yields the constraint violation bound. Furthermore, under our carefully choices of $m, \gamma, \epsilon, \eta, \alpha, \beta$ and ι, it can be easily verified that $K\epsilon$ dominates the upper bounds in (38), which leads to fact that constraint violation because zero when K is sufficiently large. In particular, under our assumption on K, which implies that $\epsilon \leq \frac{\delta}{2}$, and leads to

$$Violation(K) = 0.$$

Simulation

In this section, we present simulation results that evaluate our algorithm using the 2D safety grid-world exploration problem (Zheng and Ratliff 2020; Leike et al. 2017). Figure 1 shows the map of the 10×10 grid-world with a total of 100 states. We choose a error probability 0.03 which means with probability 0.03 the agent will choose an action randomly to make the environment stochastic. The objective of the agent is to travel to the destination (the red star) from the original position (the blue triangle) as quickly as possible while limiting the number of times hitting the obstacles (the yellow squares). Hitting an obstacle incurs cost 1 and otherwise, there is no cost. The reward for the destination is 1, and for others are the normalized Euclidean distance between them and the destination times a scaled factor 0.1. We set constraint limit as 0.15 through the simulation which means the expected cost rate should below the limit. To account for statistical significance, the results of each experiment are averaged over 5 trials.

We remark that in the simulation we consider the following constraint

$$\liminf_{K\to\infty}\frac{1}{K}\mathbb{E}_{\pi}\left[\sum_{k=1}^{K} g(s_k,a_k)\right] \leq \rho,$$

which is similar to the constraint that the average utility needs to be above a threshold.

Figure 2 shows the performance of our algorithm in terms of average reward and average cost during training compared with the algorithm in (Wei et al. 2020). We can see that our algorithm is able to learn a policy that achieves a high reward while satisfying the safety constraint very quickly.

Conclusion

In this paper, we proposed the first model-free RL algorithm for infinite-horizon average-reward CMDPs. The design of the algorithm is based on the primal-dual approach. By using the Lyapunov drift analysis, we proved that our algorithm achieves sublinear regret and zero constraint violation. Our regret bound scales as $\tilde{O}(K^{\frac{5}{6}})$ and is suboptimal compared to model-based approaches. However, this is the first model-free and simulator-free algorithm with sub-linear regret and optimal constraint violation. It is still an interesting open problem that how to achieve $\tilde{O}(\sqrt{K})$ regret bound via model-free algorithms.

The algorithm is also computationally efficient from algorithmic perspective because it is model-free, which means that it is potential to apply our method for complex and challenging CMDPs in practice. Simulation result also demonstrates the good performance of our algorithm.

References

Abbasi-Yadkori, Y.; Bartlett, P.; Bhatia, K.; Lazic, N.; Szepesvari, C.; and Weisz, G. 2019. POLITEX: Regret Bounds for Policy Iteration using Expert Prediction. In Chaudhuri, K.; and Salakhutdinov, R., eds., *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, 3692–3702. PMLR.

Achiam, J.; Held, D.; Tamar, A.; and Abbeel, P. 2017. Constrained policy optimization. In *International Conference on Machine Learning*, 22–31. PMLR.

Agarwal, M.; Bai, Q.; and Aggarwal, V. 2021. Markov Decision Processes with Long-Term Average Constraints. *arXiv preprint arXiv:2106.06680*.

Altman, E. 1999. *Constrained Markov decision processes*, volume 7. CRC Press.

Andrychowicz, O. M.; Baker, B.; Chociej, M.; Jozefowicz, R.; McGrew, B.; Pachocki, J.; Petron, A.; Plappert, M.; Powell, G.; Ray, A.; et al. 2020. Learning dexterous in-hand manipulation. *The International Journal of Robotics Research*, 39(1): 3–20.

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. *arXiv preprint arXiv:2006.05051*.

Chen, Y.; Dong, J.; and Wang, Z. 2021. A primal-dual approach to constrained Markov decision processes. *arXiv preprint arXiv:2101.10895*.

Ding, D.; Wei, X.; Yang, Z.; Wang, Z.; and Jovanović, M. R. 2020a. Provably efficient safe exploration via primal-dual policy optimization. *arXiv preprint arXiv:2003.00534*.

Ding, D.; Zhang, K.; Basar, T.; and Jovanovic, M. 2020b. Natural Policy Gradient Primal-Dual Method for Constrained Markov Decision Processes. *Advances in Neural Information Processing Systems*, 33.

Dong, K.; Wang, Y.; Chen, X.; and Wang, L. 2019. Q-learning with ucb exploration is sample efficient for infinite-horizon mdp. *arXiv preprint arXiv:1901.09311*.

Efroni, Y.; Mannor, S.; and Pirotta, M. 2020. Exploration-exploitation in constrained MDPs. *arXiv preprint arXiv:2003.02189*.

Fisac, J. F.; Akametalu, A. K.; Zeilinger, M. N.; Kaynama, S.; Gillula, J.; and Tomlin, C. J. 2018. A general safety framework for learning-based control in uncertain robotic systems. *IEEE Transactions on Automatic Control*, 64(7): 2737–2752.

Garcia, J.; and Fernández, F. 2012. Safe exploration of state and action spaces in reinforcement learning. *Journal of Artificial Intelligence Research*, 45: 515–564.

Jin, C.; Allen-Zhu, Z.; Bubeck, S.; and Jordan, M. I. 2018. Is Q-Learning Provably Efficient? In *Advances Neural Information Processing Systems (NeurIPS)*, volume 31, 4863–4873.

Kalagarla, K. C.; Jain, R.; and Nuzzo, P. 2020. A Sample-Efficient Algorithm for Episodic Finite-Horizon MDP with Constraints. *arXiv preprint arXiv:2009.11348*.

Lattimore, T.; and Szepesvári, C. 2020. *Bandit algorithms*. Cambridge University Press.

Leike, J.; Martic, M.; Krakovna, V.; Ortega, P. A.; Everitt, T.; Lefrancq, A.; Orseau, L.; and Legg, S. 2017. AI safety gridworlds. *arXiv preprint arXiv:1711.09883*.

Liu, T.; Zhou, R.; Kalathil, D.; Kumar, P.; and Tian, C. 2021. Learning Policies with Zero or Bounded Constraint Violation for Constrained MDPs. *arXiv preprint arXiv:2106.02684*.

Ono, M.; Pavone, M.; Kuwata, Y.; and Balaram, J. 2015. Chance-constrained dynamic programming with application to risk-aware robotic space exploration. *Autonomous Robots*, 39(4): 555–571.

Ortner, R. 2020. Regret bounds for reinforcement learning via markov chain concentration. *Journal of Artificial Intelligence Research*, 67: 115–128.

Paternain, S.; Chamon, L. F.; Calvo-Fullana, M.; and Ribeiro, A. 2019. Constrained reinforcement learning has zero duality gap. *arXiv preprint arXiv:1910.13393*.

Qiu, S.; Wei, X.; Yang, Z.; Ye, J.; and Wang, Z. 2020. Upper Confidence Primal-Dual Reinforcement Learning for CMDP with Adversarial Loss. *arXiv preprint arXiv:2003.00660*.

Richter, F.; Orosco, R. K.; and Yip, M. C. 2019. Open-sourced reinforcement learning environments for surgical robotics. *arXiv preprint arXiv:1903.02090*.

Silver, D.; Schrittwieser, J.; Simonyan, K.; Antonoglou, I.; Huang, A.; Guez, A.; Hubert, T.; Baker, L.; Lai, M.; Bolton, A.; et al. 2017. Mastering the game of go without human knowledge. *nature*, 550(7676): 354–359.

Singh, R.; Gupta, A.; and Shroff, N. B. 2020. Learning in Markov decision processes under constraints. *arXiv preprint arXiv:2002.12435*.

Stooke, A.; Achiam, J.; and Abbeel, P. 2020. Responsive safety in reinforcement learning by pid lagrangian methods. In *International Conference on Machine Learning*, 9133–9143. PMLR.

Sutton, R. S.; and Barto, A. G. 2018. *Reinforcement learning: An introduction*. MIT press.

Tessler, C.; Mankowitz, D. J.; and Mannor, S. 2018. Reward constrained policy optimization. *arXiv preprint arXiv:1805.11074*.

Vinyals, O.; Babuschkin, I.; Czarnecki, W. M.; Mathieu, M.; Dudzik, A.; Chung, J.; Choi, D. H.; Powell, R.; Ewalds, T.; Georgiev, P.; et al. 2019. Grandmaster level in StarCraft II using multi-agent reinforcement learning. *Nature*, 575(7782): 350–354.

Wei, C.-Y.; Jafarnia Jahromi, M.; Luo, H.; and Jain, R. 2021. Learning Infinite-horizon Average-reward MDPs with Linear Function Approximation. In Banerjee, A.; and Fukumizu, K., eds., *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, 3007–3015. PMLR.

Wei, C.-Y.; Jahromi, M. J.; Luo, H.; Sharma, H.; and Jain, R. 2020. Model-free reinforcement learning in infinite-horizon average-reward markov decision processes. In *International conference on machine learning*, 10170–10180. PMLR.

Wei, H.; Liu, X.; and Ying, L. 2021. A Provably-Efficient Model-Free Algorithm for Constrained Markov Decision Processes. *arXiv preprint arXiv:2106.01577*.

Xu, T.; Liang, Y.; and Lan, G. 2020. A Primal Approach to Constrained Policy Optimization: Global Optimality and Finite-Time Analysis. *arXiv preprint arXiv:2011.05869*.

Yang, T.-Y.; Rosca, J.; Narasimhan, K.; and Ramadge, P. J. 2020. Projection-based constrained policy optimization. *arXiv preprint arXiv:2010.03152*.

Zheng, L.; and Ratliff, L. 2020. Constrained upper confidence reinforcement learning. In *Learning for Dynamics and Control*, 620–629. PMLR.