A Duality Approach for Regret Minimization in Average-Reward Ergodic Markov Decision Processes

Hao Gong HGONG@PRINCETON.EDU
Department of Operations Research and Financial Engineering, Princeton University, NJ 08540

Mengdi Wang MENGDIW@PRINCETON.EDU
Department of Electrical Engineering, Princeton University, NJ 08540

Editors: A. Bayen, A. Jadbabaie, G. J. Pappas, P. Parrilo, B. Recht, C. Tomlin, M.Zeilinger

Abstract

In light of the Bellman duality, we propose a novel value-policy gradient algorithm to explore and act in infinite-horizon Average-reward Markov Decision Process (AMDP) and show that it has sublinear regret. The algorithm is motivated by the Bellman saddle point formulation. It learns the optimal state-action distribution, which encodes a randomized policy, by interacting with the environment along a single trajectory and making primal-dual updates. The key to the analysis is to establish a connection between the min-max duality gap of Bellman saddle point and the cumulative regret of the learning agent. We show that, for ergodic AMDPs with finite state space $\mathcal{S}$ and action space $\mathcal{A}$ and uniformly bounded mixing times, the algorithm's T-time step regret is

$$R(T) = \tilde{\mathcal{O}}\left((t^*_{mix})^2 \tau^{\frac{3}{2}} \sqrt{(\tau^3 + |\mathcal{A}|)|\mathcal{S}|T}\right),$$

where t^*_{mix} is the worst-case mixing time, τ is an ergodicity parameter, T is the number of time steps and $\tilde{\mathcal{O}}$ hides polylog factors.

Keywords: regret analysis, Markov decision process, primal-dual method, saddle point, exponentiated gradient, reinforcement learning, online learning

1. Introduction

Reinforcement learning (RL) addresses the problem of an agent learning to act in an environment in order to optimize long term performance (Bertsekas, 2007; Sutton and Barto, 1998). We consider RL in the infinite-horizon undiscounted *Average-Reward Markov Decision Process* (AMDP). Without knowing the transition model, an agent has to explore continuously and maximize the long-term average-per-time-step reward. We focus on the tabular AMDP with finitely many states and actions, under the assumption that the MDP is ergodic under any policy. Our main interest is to develop an online algorithm that observes state transitions and learns to act along a single trajectory, with provably sublinear regret.

Regret minimization for AMDP has been considered in a number of prior works including Auer et al. (2009); Bartlett and Tewari (2012); Osband and Roy (2016); Ouyang et al. (2017); Agrawal and Jia (2017); Zhang and Ji (2019). The best known regret is $\mathcal{O}(\sqrt{D|\mathcal{S}||\mathcal{A}|T})$, where $\mathcal{S}$, $\mathcal{A}$ are the state space and action space respectively, $D = \max_{i,j} \min_{\pi} \mathbb{E}^{\pi}[\text{HittingTime}(j) \mid i]$ is known as the MDP's diameter. Most of these methods are based on estimating either transition models or Q functions and constructing upper confidence state-action values to encourage exploration.

In this paper, we take a different route towards regret minimization in AMDP, in light of the min-max duality that is intrinsic to Bellman equations. It is known that optimal Bellman equation can be formulated as linear programs (Puterman, 2014). It further implies an equivalent min-max Bellman saddle point (Wang, 2017), which motivates us to adopt a primal-dual optimization approach. This approach was previously considered in the case with a generative model and can learn an approximate-optimal policy by sampling from the oracle (Chen and Wang, 2016; Wang, 2017). In this work we study the more challenging regret minimization problem without assuming a generative model, where the agent has to learn online in an infinitely long process. Our algorithm approximates the value function (primal variable) and the optimal state-action stationary distribution (dual variable) simultaneously by conducting a series of on-policy primal-dual updates, during which actions are picked greedily according to the randomized policy inferred from the dual iterates. A key observation is that the cumulative regret of this learning algorithm can be bounded by a multiple of the averaged duality gap of the primal-dual iterates. This allows us to establish a sublinear regret $C\sqrt{|\mathcal{S}||\mathcal{A}|T}$, where C depends on the worst-case mixing times and an ergodicity parameter that measures the multiplicative range of stationary distributions of the AMDP. In the case where $\mathcal{S}$ contains a single state, the algorithm reduces to EXP3 for multi-arm bandit (Auer et al., 2003) (Freund and Schapire, 1997). In this case we have $C = O(1)$ and the result becomes the standard regret $\tilde{O}(\sqrt{|\mathcal{A}|T})$ for multi-arm bandit.

This paper has several technical novelties:

- Our results appear to be the first *duality-based value-policy gradient method and regret analysis* for infinite-horizon RL. The proof relates the cumulative regret with a Lagrangian duality gap through characterizing the empirical state distribution between consecutive dual updates.
- We do not assume bounded diameter (worst-case hitting time) as needed in most existing analyses. Instead, our analysis depends on an ergodicity parameter that plays an important role in the complexity theory of AMDP, which is an analogy of "diameter in policy space". Our bound can be significantly smaller than diameter-dependent bounds in some cases.
- To analyze the empirical state distribution between updates, we use a change of measure trick and the Aldous's lemma for Martingale stopping times, which associates the cumulative regret with cover time and hitting time of the MDP, and may be of independent interest

2. Preliminaries

2.1. Average-Reward Markov Decision Problem

The environment of an AMDP can be specified by a tuple $\mathcal{M} = (\mathcal{S}, \mathcal{A}, (P^a)_{a\in\mathcal{A}}, (\mathbf{r}^a)_{a\in\mathcal{A}})$, where $\mathcal{S}$ is the set of states, $\mathcal{A}$ is the set of actions, $P^a_{i,j} = P(s_{t+1} = j | s_t = i, a_t = a)$ is the unknown transition matrix, and $r^a_i = \mathbf{E}[r_{t+1}|s_t = i, a_t = a]$ is the unknown reward function. Suppose that the environment is in state i at time t and the agent selects action a, the environment will evolve to the next state j with probability $P^a_{i,j}$ and give the agent a random reward $r_t \in [0, 1]$. A stationary policy π maps each state to a distribution over the action space. It can be represented by a $|\mathcal{S}|$-by-$|\mathcal{A}|$ stochastic matrix, whose (i,a)-th element is the probability of choosing action a at state i. The optimal policy π^* is the policy that maximizes the infinite-horizon expected average reward

$$\bar{v}^\pi = \mathbf{E}^\pi \left[\lim_{t\to\infty} \frac{1}{T} \sum_{r=1}^{T} r_t \right], \tag{1}$$

where $\mathbf{E}^\pi$ denotes expectation over all possible trajectories generated by policy π.

Throughout the paper, we assume that the AMDP is ergodic under any π, i.e., the transition matrix P^π under policy π is aperiodic and positive-recurrent. The above $\bar{v}^\pi$ is thus uniquely defined regardless of the initial state s_1. Ergodicity of AMDP is necessary for an online agent to avoid getting stuck in some state. Without ergodicity, any learning agent may incur linear regret $\Omega(T)$, if she misses choosing the correct action to reach some rewarding state that is not reachable later.

2.2. Min-Max Formulation of Bellman Equation

Under some parameterized ergodicity condition (see Assumption 1 in the next section), for any policy π there exists a unique stationary distribution ν^π such that $\nu_i^\pi = \lim_{t\to\infty} P(s_t = i | a_{1:t-1} \sim \pi)$, which is independent of s_1. It can be shown that maximizing the average reward is equivalent to the following optimization problem (Puterman, 2014)

$$\max_{\xi,\pi} \quad \bar{v}^\pi = \sum_{i\in\mathcal{S}} \xi_i \sum_{a\in\mathcal{A}} \pi_{i,a} r_i^a \qquad \text{subject to} \quad (P^\pi)^\top \xi = \xi,\ \xi \geq \mathbf{0},\ \sum_{i\in\mathcal{S}} \xi_i = 1, \tag{2}$$

where the constraint forces ξ to be the stationary distribution ν^π of policy π. Let $\mu_{i,a} = \xi_i \pi_{i,a}$ denote the joint stationary state-action probability. The above problem is equivalent to a linear program

$$\max_{\mu} \quad \sum_{a\in\mathcal{A}} \mu_a^\top \mathbf{r}^a \qquad \text{subject to} \quad \sum_{a\in\mathcal{A}} \left(I - (P^a)^\top\right) \mu_a = \mathbf{0}, \quad \sum_{a,i} \mu_{i,a} = 1, \quad \mu \geq \mathbf{0}, \tag{3}$$

where the constraint ensures that μ is a stationary joint state-action distribution. It is worth pointing out that the dual problem of (3) is equivalent to the Bellman equation (Puterman, 2014), where the state value function coincides with the Lagrangian multipliers associated with the constraint $\sum_{a\in\mathcal{A}} \left(I - (P^a)^\top\right) \mu_a = \mathbf{0}$. Now we follow the ideas of Chen and Wang (2016); Wang (2017) and formulate the linear system (3) into an equivalent min-max problem

$$\min_{\mathbf{h}\in\mathcal{H}} \max_{\mu\in\mathcal{U}} \quad \sum_{a\in\mathcal{A}} \mu_a^\top \left((P^a - I)\mathbf{h} + \mathbf{r}^a\right) \tag{4}$$

where $\mathcal{H}$ and $\mathcal{U}$ are constraint sets used to regularize primal-dual iterates for fast convergence (see Lemma 3 in Appendix B for details). As shown in Wang (2017), the saddle point $(\mu^*,\ \mathbf{h}^*)$ solution gives the optimal policy by $\pi_{i,a}^* = \frac{\mu_{i,a}^*}{\sum_{a\in\mathcal{A}} \mu_{i,a}^*}$. In the rest of the paper we call $\mathbf{h}$ the primal variable and μ the dual variable.

3. An Online Primal-Dual Algorithm

Bellman duality implies that maximizing the average-reward (1) in AMDP is equivalent to solving the min-max problem (4). Our goal is to construct a reinforcement learning algorithm by taking advantage of the min-max duality to explore and act in the environment. The learning agent will use the current dual variable μ to greedily prescribe actions at each state, using the implied policy π given by $\pi_{i,a} = \frac{\mu_{i,a}}{\sum_{a\in\mathcal{A}} \mu_{i,a}}$.

There are two technical issues: (1) Unbiased gradients cannot be easily obtained. Unlike the case with a generative model, constructing unbiased value and distributional gradient estimates is

challenging in online RL. This is because we only have highly dependent past experiences, and the future state distribution that will generate new samples is unknown. (2) The algorithm needs to balance the exploration-exploitation trade-off. Our primal-dual algorithm does not compute any upper confidence bounds. The μ update needs to automatically encourage exploration in areas of high uncertainty. Having the two questions in mind, let us construct the algorithm.

Constructing partial gradient estimates. Denoting $L(\mathbf{h}, \mu) = \sum_{a\in\mathcal{A}} \mu_a^\top ((P^a - I)\mathbf{h} + \mathbf{r}^a)$ the min-max objective. To solve the optimization problem, we wish to construct unbiased estimates of the primal-dual gradients

$$\frac{\partial L}{\partial \mathbf{h}} = \sum_{a\in\mathcal{A}} (P^a - I)^\top \mu_a, \qquad \frac{\partial L}{\partial \mu_a} = (P^a - I)\mathbf{h} + \mathbf{r}^a, \tag{5}$$

and use them to perform updates. Let us start with an ideal situation, assuming there was a uniform sampler that generates states $i \sim \text{Uniform}(\mathcal{S})$, waits until we pick some action a, and outputs $r \sim r_i^a$ and $j \sim P_{i,\cdot}^a$. Given the current μ, we pick a with probability $\pi_{i,a} = \frac{\mu_{i,a}}{\sum_{a\in\mathcal{A}} \mu_{i,a}}$ at state i. In this case, it is easy to see that $(\mathbf{e}_j - \mathbf{e}_i) \cdot \sum_{a\in\mathcal{A}} \mu_{i,a}$ and $\mathbf{e}_{i,a} \cdot \frac{h_j - h_i + r}{\pi_{i,a}}$ are unbiased estimates of $\frac{\partial L}{\partial \mathbf{h}}$ and $\frac{\partial L}{\partial \mu}$, respectively. Next we will construct unbiased gradient samplings without being able to sample states uniformly.

Making gradient samples unbiased via a change of measure. The empirical distribution of the online samples depends on both the initial state and all actions that have been chosen by the learning agent. The sample transitions depend on one other, making it harder to obtain independent unbiased estimators than in the case where an oracle is available (Chen and Wang, 2016; Wang, 2017). However, the above discussion on hypothetical uniform sampler inspires a way to tackle this problem. We impose a "uniform distribution" on the training samples by constructing batch updates such that each batch contains exactly one sample transition starting from each state. Intuitively, we are selecting a subset of observations to change the empirical measure to a uniform measure.

In preparation for a new update, we initialize an empty set $\mathcal{B}$. The agent then keeps sampling along the trajectory and acting according to the policy defined by $\pi_{i,a} = \frac{\mu_{i,a}}{\sum_{a\in\mathcal{A}} \mu_{i,a}}$ at each state i it faces. Every time it encounters a state i that has not been visited during the current update, the agent adds the its transition and reward (i, a_i, j_i, r_i) to $\mathcal{B}$. When all states have been visited, we have exactly $|\mathcal{S}|$ samples in the current batch $\mathcal{B} = \{(i, a_i, j_i, r_i) \mid i \in \mathcal{S}\}$. Then the agent constructs a pair of unbiased gradient estimators of $-\frac{\partial L}{\partial \mathbf{h}}$ and $\frac{\partial L}{\partial \mu_a}$ as

$$\mathbf{d} = \sum_{i\in\mathcal{S}} (\mathbf{e}_i - \mathbf{e}_{j_i}) \cdot \xi_i,$$
$$\Delta_{i,a} = \mathbb{1}_{\{a_i = a\}} \frac{h_{j_i} - h_i + r_i}{\pi_{i,a}}, \quad \forall\, i, a.$$

where $\xi_i = \sum_{a\in\mathcal{A}} \mu_{i,a}$. The agent then updates $\mathbf{h}$, μ in the way detailed below, resets $\mathcal{B}$ to empty set and starts preparing for the next update batch. The constructed samples can be shown to be conditionally unbiased per batch, where each batch update takes a random number of time steps.

Exploration with exponentiated distributional gradient. In analogy to EXP3, we also use exponentiated updates in μ to encourage exploration. Denoting $\Delta^{(k+1)}$ the unbiased estimator obtained

for $\frac{\partial L}{\partial \mu}$, we conduct the update by

$$\mu^{(k+\frac{1}{2})} \leftarrow \frac{\mu^{(k)} \cdot \exp(\Delta^{(k+1)})}{\|\mu^{(k)} \cdot \exp(\Delta^{(k+1)})\|_{1,1}},\ \mu^{(k+1)} \leftarrow \operatorname*{argmin}_{\mu \in \mathcal{U}} D_{KL}(\mu \parallel \mu^{(k+\frac{1}{2})}).$$

In view of optimization, this exponentiated μ update is a proximal gradient step using projection with respect to the Kullback-Leiber divergence. See Schulman et al. (2017) for similar usage. Similar to the role of exponentiated update in online optimization, the above μ update is the key to exploration. In the case where $\mathcal{S}$ contains only one state, the algorithm degenerates to the EXP3 algorithm for multi-arm bandit.

Algorithm 1 gives the full implementation details.

Algorithm 1: Online Primal-Dual π Learning

Input: Precision level $\epsilon > 0$, $\mathcal{S}$, $\mathcal{A}$, t^*_{mix}, τ
Set α, β, M according to Theorem 4 in Appendix B
Initialize primal-dual variables $\mathbf{h}^{(0)} \leftarrow \mathbf{0} \in \mathbb{R}^{|\mathcal{S}|}$, $\mu^{(0)} \leftarrow \frac{1}{|\mathcal{S}||\mathcal{A}|}\mathbf{1} \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$
Initialize $k \leftarrow 0$, $\Delta \leftarrow \mathbf{0} \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$, $\mathbf{d} \leftarrow \mathbf{0} \in \mathbb{R}^{|\mathcal{S}|}$
Initialize $S' \leftarrow S$, the set of all states that haven't been visited
Initialize environment s_1
Compute current policy $\pi^{(k)}_{i,a} = \frac{\mu^{(k)}_{i,a}}{\sum_{a' \in \mathcal{A}} \mu^{(k)}_{i,a'}}$ and $\xi^{(k)}_i = \sum_{a' \in \mathcal{A}} \mu^{(k)}_{i,a'}$, $\forall i \in \mathcal{S}$, $\forall a \in \mathcal{A}$
for *time step* $t = 1, 2, 3, \cdots$ **do**
  Agent picks action a_t according to $\pi^{(k)}_{s_t,\cdot}$, observes reward r_t and next state s_{t+1}
  if $s_t \in S'$ **then**
    $S' \leftarrow S' \setminus \{s_t\}$
    $\Delta \leftarrow \Delta + \beta \cdot \frac{h^{(k)}_{s_{t+1}} - h^{(k)}_{s_t} + r_t - M}{\pi^{\top}_{s_t,a_t}} \cdot \mathbf{e}_{s_t,a_t}$
    $\mathbf{d} \leftarrow \mathbf{d} + \alpha \cdot \xi^{(k)}_{s_t} \cdot (\mathbf{e}_{s_t} - \mathbf{e}_{s_{t+1}})$
    where $\mathbf{e}_{i,j} \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$, $\mathbf{e}_i \in \mathbb{R}^{|\mathcal{S}|}$ are one-hot vectors
  end
  if $S' = \emptyset$ **then**
    Dual update $\mu^{(k+\frac{1}{2})}_{i,a} \leftarrow \frac{\mu^{(k)}_{i,a} \cdot \exp(\Delta_{i,a})}{\sum_{i',a'} \mu^{(k)}_{i',a'} \exp(\Delta_{i',a'})}$ $\mu^{(k+1)} \leftarrow \operatorname{argmin}_{\mu \in \mathcal{U}} D_{KL}(\mu \parallel \mu^{(k+\frac{1}{2})})$
    Primal update $\mathbf{h}^{(k+1)} \leftarrow \mathbf{Proj}_{\mathcal{H}}(\mathbf{h}^{(k)} + \mathbf{d})$
    Update policy $\pi^{(k+1)}_{i,a} = \frac{\mu^{(k+1)}_{i,a}}{\sum_{a' \in \mathcal{A}} \mu^{(k+1)}_{i,a'}}$ and $\xi^{(k+1)}_i = \sum_{a' \in \mathcal{A}} \mu^{(k+1)}_{i,a'}$, $\forall i \in \mathcal{S}$, $\forall a \in \mathcal{A}$
    Start next batch update $k \leftarrow k+1$, $\Delta \leftarrow \mathbf{0} \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$, $\mathbf{d} \leftarrow \mathbf{0} \in \mathbb{R}^{|\mathcal{S}|}$, $S' \leftarrow S$
  end
end

4. Regret Analysis

In this section we analyze the regret of Algorithm 1.

4.1. Main Result

The regret of RL is the difference between the expected cumulative reward obtained by the algorithm, and that we could have gained if we ran the optimal policy for the same length of time:

$$R(T) := \mathbf{E}^{\pi^*}\left[\sum_{t=1}^{T} r'_t\right] - \mathbf{E}^{alg}\left[\sum_{t=1}^{T} r_t\right], \tag{6}$$

where $\mathbf{E}^{\pi^*}$ and $\mathbf{E}^{alg}$ denote expectations taken over trajectories generated under the optimal policy and the learning algorithm, respectively.

The following assumption and definitions are useful for analyzing the regret.

Assumption 1 (Ergodic Decision Process) *The Markov decision process specified by $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{P} = (P^a)_{a\in\mathcal{A}}, \mathbf{r} = (\mathbf{r}^a)_{a\in\mathcal{A}})$ is τ-stationary in the sense that it is ergodic under any stationary policy π and there exists $\tau > 1$ such that*

$$\frac{1}{\sqrt{\tau}|\mathcal{S}|}\mathbf{1} \le \nu^\pi \le \frac{\sqrt{\tau}}{|\mathcal{S}|}\mathbf{1}.$$

The ergodicity parameter τ measures the range of stationary distribution across possible policies. It is an important quantity that plays a key role in the complexity theory of AMDP. For example, this τ showed up in sample complexity bound for solving AMDP using a generative model Wang (2017). It also relates to the worst-case distributional correction ratio which shows up in off-policy RL Liu et al. (2019). The paper Sidford et al. (2019) proves that the number of policy iterations needed to solve AMDP depends linearly on τ. This ergodicity parameter can be essentially viewed as the "diameter" of policy space.

Definition 1 (Mixing Time, Hitting Time and Cover Time) *For any MDP $\mathcal{M}$, its worst-case mixing time, worst-case hitting time and worst-case cover time are defined respectively, as*

$$\begin{aligned} t^*_{mix} &:= \max_\pi \min\left\{ t \ge 1 \,\middle|\, \|(P^\pi)^\top(i,\cdot) - \nu^\pi\|_{TV} \le \frac{1}{4},\ \forall i \in \mathcal{S} \right\}, \\ t^*_{hit} &:= \max_\pi \max_{s,x\in\mathcal{S}} \mathbf{E}^\pi\left[\tau_{hit}(x) | s_0 = s\right], \quad t^*_{cov} := \max_\pi \max_{s\in\mathcal{S}} \mathbf{E}^\pi\left[\tau_{cov} | s_0 = s\right]. \end{aligned} \tag{7}$$

where $\|\cdot\|_{TV}$ denotes the total variation, $\tau_{hit}(x) = \min\{t > 0 \mid s_t = x\}$ is the first time at which state $x \in \mathcal{S}$ is visited, and $\tau_{cov} = \min\{t > 0 \mid \{s_1, s_2, ..., s_t\} \supset \mathcal{S}\}$ is the first time at which all the states have been visited.

Our main result is given below.

Theorem 2 *Suppose MDP $\mathcal{M}$ satisfies Assumption 1, then the following regret bound holds for Algorithm 1*

$$R(T) = \tilde{\mathcal{O}}\left((t^*_{mix})^2 \tau^{\frac{3}{2}} \sqrt{(\tau^3 + |\mathcal{A}|)|\mathcal{S}|T}\right). \tag{8}$$

Our regret result depends on the worst-case mixing time t^*_{mix} and the ergocity parameter τ that is a worst-case probability correction ratio. The mixing time t^*_{mix} characterize the "transientness" of the AMDP, while τ characterizes only stationary distributions - the two quantities are orthogonal to each other. For comparison, a best known regret upper bound for AMDP is $D\sqrt{|\mathcal{S}||\mathcal{A}|T}$ which

depends on the diameter D. There are cases where our regret bound can be much smaller while D is large. For example consider the extreme case where the $P(\cdot|s,a)$ equals to the uniform distribution for all s, a. Then the AMDP reduces to contextual bandit with $|\mathcal{S}|$ distinct contexts and $|\mathcal{A}|$ distinct arms per context. In this case it is easy to verify that $D = 1/|\mathcal{S}|$, $t^* = 1$ and $\tau = 1$, as a result our regret bound is $\tilde{O}(\sqrt{|\mathcal{S}||\mathcal{A}|T})$ - much smaller than the existing result in this case which is $D\sqrt{|\mathcal{S}||\mathcal{A}|T} = \tilde{O}(\sqrt{|\mathcal{S}|^{3/2}|\mathcal{A}|T})$.

4.2. Sketch of the Proof.

In what follows we outline the key ideas of the regret proof.

Convergence of duality gap From an optimization view, the algorithm makes noisy primal-dual gradient updates. By invoking a primal-dual convergence analysis tailored to the exponentiated dual update, we can show that the averaged duality gap across N batch updates satisfies

$$\text{DualityGap}(N) = \frac{1}{N}\sum_{k=0}^{N-1}\mathbf{E}\left[\sum_{a\in\mathcal{A}}(\mathbf{h}^* - P_a\mathbf{h}^* - \mathbf{r}_a + \bar{v}^*\mathbf{1})^\top \mu_a^{(k)}\right] \leq \tilde{\mathcal{O}}\left(t^*_{mix}\sqrt{\frac{\tau^3+|\mathcal{A}|}{N}}\right). \tag{9}$$

See Theorem 4, Lemmas 4-9 in Appendix B for details of the duality gap analysis.

Relating cumulative regret and duality gap Recall that each batch update takes a variable number of time steps. Now we analyze R_N, the regret accumulated throughout N batch updates, i.e.,

$$R_N = \sum_{k=0}^{N-1}\mathbf{E}^{alg}\left[\sum_{a\in\mathcal{A}}\sum_{i\in\mathcal{S}}\left(\mathbf{h}^* - P^a\mathbf{h}^* - \mathbf{r}^a + \bar{v}^*\mathbf{1}\right)_i \pi_{i,a}^{(k)}\hat{n}_i^{(k)}\right] + \mathcal{O}\left(t^*_{mix}\right), \tag{10}$$

where $\bar{v}^*$ is the optimal average reward and $\pi^{(k)}$ is the behavior policy used by the learning agent between the k-th and the $(k+1)$-th updates. Here $\hat{n}_i^{(k)}$ is the number of visits to state i during the same period, and in our case it can be decomposed as the empirical distribution $\hat{\nu}^{(k)}$ multiplied by a cover time $\tau_{cov}^{(k)}$ needed for finishing the batch. We denote $\hat{n}^{(k)} = \sum_i \hat{n}_i^{(k)}$ be the total number of time steps between the two consecutive updates.

Intuitively, if we can figure out a way to related the empirical distribution $\hat{\nu}^{(k)}$ to the stationary distribution $\nu^{\pi^{(k)}}$ associated with policy $\pi^{(k)}$, we will be able to control $\pi_{i,a}^{(k)}\hat{n}_i^{(k)}$ by $\mathcal{O}\left(\pi_{i,a}^{(k)}\nu^{\pi^{(k)}}t^*_{cov}\right)$, and thus by $\mathcal{O}\left(\tau\mu_{i,a}t^*_{cov}\right)$ according to the ergodicity assumption. This leads to an upper bound of R_N in the form of

$$R_N \leq \mathcal{O}\left(t_{cov}\tau\sum_{k=0}^{N-1}\mathbf{E}^{alg}\left[\sum_{a\in\mathcal{A}}\left(\mathbf{h}^* - P^a\mathbf{h}^* - \mathbf{r}^a + \bar{v}^*\mathbf{1}\right)^\top \mu_a^{(k)}\right]\right) \leq C\cdot\text{DualityGap}(N), \tag{11}$$

which C is a constant depending on the ergodicity parameter τ.

Analyzing the state distribution between consecutive batch updates The problem now reduces to analyzing the empirical distribution of the states visited between two consecutive batch updates, where each update time is a Martingale stopping time. Analyzing the empirical distribution of states between two stopping times is nontrivial in general - the number of states in between is stochastic, as well as the starting state and the end state.

To make it work, we first condition on the σ-algebra generated by the former stopping time - the difference between the two stopping times thus becomes a cover time. We invoke the Aldous Lemma (Levin et al., p. 130) and obtain that the expected empirical distribution covered by a stopping time equals to the stationary distribution, if the stopping time always stops at the same state as starting state, i.e.,

$$\text{if } P(X_\tau = a | X_1 = a) = 1 \text{ then } \mathbf{E}\left[\sum_{t=1}^{\infty} \mathbf{1}_{\{X_t = i, \tau > t\}}\right] = \mathbf{E}[\tau | X_1 = a]\nu(i),\ \forall i,$$

where ν is the stationary distribution. In order to utilize the Aldous Lemma, we append a hitting time to the cover time as if the agent was to wait to see all states and then the original state where the last update happens. The hitting time, on the other hand, can be as long as the cover time in expectation, leading to an upper bound of $\mathbb{E}[\hat{n}_i^{(k)}] \leq 2t^*_{cov}\nu_i^{(k)}$. The detailed proof is provided in Appendix C.

Regret analysis for fixed T The previous analysis is concerned with the cumulative regret over N updates, where each update takes a random number of time steps.

Finally let us analyze the regret as the number of time steps T increases. Note that the T-timestep regret of Algorithm 1 is less than R_N with $N = T/|\mathcal{S}|$, which is the most number of updates that can be conducted within T steps. Appendix D summarizes the theoretical results obtained so far. Putting the analysis together we obtain the following bound on Algorithm 1 with T steps.

$$R_T = \tilde{\mathcal{O}}\left(t^*_{mix}\tau\sqrt{(\tau^3 + |\mathcal{A}|)\frac{T}{|\mathcal{S}|}{t^*_{cov}}^2}\right). \tag{12}$$

Cover time and hitting time analysis The only thing left is to estimate the worst-case cover time of the MDP - if it was too large compared to the state size $|\mathcal{S}|$, the proposed algorithms would be intractable. Matthews proved that the worst-case cover time of an irreducible finite Markov is controlled by its worst-case hitting time multiplied by log state size. Meanwhile, the relationship between the mixing time and the hitting time of a large set has been studied in probability literature (Aldous, 1982; Peres and Sousi, 2011; Oliveira; Anderson et al., 2018) that indicates these two quantities are equal up to some universal multiplicative constant. Although the previous work sheds some light on our case, none of it directly applies to the worst-case hitting time of a single state. Most work also imposed reversibility or some other conditions on the Markov chain. We adjust the idea and provide a simple proof in Appendix E to bound the hitting time of fast-mixing Markov chains under Assumption 1. Combining Mathews' method and the bound on hitting time we get $t^*_{cov} = \tilde{\mathcal{O}}\left(\sqrt{\tau}t^*_{mix}|\mathcal{S}|\right)$. This bound, together with (12), completes the proof.

5. Summary

This paper explored a new approach towards regret minimization in MDP by leveraging the intrinsic min-max duality of Bellman equation. We provided the first duality-based value-policy updating method that is able to learn and explore in undiscounted MDP environment and achieve sublinear regret. The regret analysis combines primal-dual convergence, exponentiated updates and a hitting time analysis, which finds a useful connection between min-max duality gap and the cumulative regret. We hope this new approach and its analysis might pique researchers' interest and motivate more generalizable methods.

References

Shipra Agrawal and Randy Jia. Optimistic posterior sampling for reinforcement learning: worst-case regret bounds. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 1184–1194. Curran Associates, Inc., 2017.

David Aldous and James Allen Fill. Reversible markov chains and random walks on graphs, 2002. Unfinished monograph, recompiled 2014, available at http://www.stat.berkeley.edu/$\sim$aldous/RWG/book.html.

David J. Aldous. Some inequalities for reversible markov chains. *J. London Math. Soc*, pages 564–576, 1982.

Robert M. Anderson, Haosui Duanmu, and Aaron Smith. Mixing times and hitting times for general markov processes, 2018.

Peter Auer, Nicolò Cesa-Bianchi, Yoav Freund, and Robert E. Schapire. The nonstochastic multi-armed bandit problem. *SIAM J. Comput.*, 32(1):48–77, January 2003. ISSN 0097-5397.

Peter Auer, Thomas Jaksch, and Ronald Ortner. Near-optimal regret bounds for reinforcement learning. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, *Advances in Neural Information Processing Systems 21*, pages 89–96. Curran Associates, Inc., 2009.

Peter L. Bartlett and Ambuj Tewari. Regal: A regularization based algorithm for reinforcement learning in weakly communicating mdps, 2012.

Dimitri P Bertsekas. *Dynamic programming and optimal control*. Athena scientific Belmont, MA, 2007.

Yichen Chen and Mengdi Wang. Stochastic Primal-Dual Methods and Sample Complexity of Reinforcement Learning. *arXiv e-prints*, art. arXiv:1612.02516, December 2016.

Yoav Freund and Robert E Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55(1):119–139, 1997.

David A. Levin, Yuval Peres, and Elizabeth L. Wilmer. *Markov chains and mixing times*. American Mathematical Society.

Yao Liu, Adith Swaminathan, Alekh Agarwal, and Emma Brunskill. Off-policy policy gradient with state distribution correction. *arXiv preprint arXiv:1904.08473*, 2019.

Peter Matthews. Covering problems for markov chains. *Ann. Probab.*, (3):1215–1228, 07 . doi: 10.1214/aop/1176991686.

Roberto Oliveira. Mixing and hitting times for finite markov chains. *Electron. J. Probab.*, page 12 pp. doi: 10.1214/EJP.v17-2274.

Ian Osband and Benjamin Van Roy. On lower bounds for regret in reinforcement learning, 2016.

Yi Ouyang, Mukul Gagrani, Ashutosh Nayyar, and Rahul Jain. Learning unknown markov decision processes: A thompson sampling approach, 2017.

Yuval Peres and Perla Sousi. Mixing times are hitting times of large sets, 2011.

Martin L Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017.

Aaron Sidford, Mengdi Wang, Lin Yang, and Yinyu Ye. Improved upper and lower bounds for policy and strategy iteration. *NeurIPS workshop on OptRL*, 2019.

Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*, volume 1. MIT press Cambridge, 1998.

Mengdi Wang. Primal-dual π learning: Sample complexity and sublinear run time for ergodic markov decision problems. *CoRR*, abs/1710.06100, 2017.

Zihan Zhang and Xiangyang Ji. Regret minimization for reinforcement learning by evaluating the optimal bias function, 2019.

Appendix A. Notations

Some additional notations are needed before heading to the regret analysis.

Let $s_1, a_1, r_1, s_2, a_2, r_2, ...$ be the infinite-length trajectory generated by Algorithm 1 as if $T = \infty$. Define $\{\mathcal{F}_t = \sigma(\{s_1, a_1, r_1, s_2, \cdots, s_{t-1}, a_{t-1}, r_{t-1}, s_t\})\}_{t\geq 1}$ be the filtration generated by the trajectory up to time t, not including the terminal action, reward and transition. We define stopping times (the times of batch updates) as

$$\tau^{(0)} = 0,\ \tau^{(k)} = \inf\left\{t > \tau^{(k-1)} | \forall i \in \mathcal{S}, \exists\, t' \in (\tau^{(k-1)}, t] \text{ s.t. } s_{t'} = i\right\},\ k \geq 1.$$

Denote for short $\mathcal{F}_{(k)} := \mathcal{F}_{\tau^{(k)}+1}$, which is a valid σ-algebra since $(\tau^{(k)} + 1)$ is also a stopping time. Furthermore, it follows from $\tau^{(k)} < \tau^{(k+1)}$ that $\mathcal{F}_{(k)} \subset \mathcal{F}_{(k+1)}$.

The trajectory-policy sequence generated by Algorithm 1 with N batch updates is

$$s_1 \xrightarrow{\pi^{(0)}} s_2 \dashrightarrow s_{\tau^{(1)}} \xrightarrow{\pi^{(0)}} s_{\tau^{(1)}+1} \xrightarrow{\pi^{(1)}} s_{\tau^{(1)}+2} \dashrightarrow s_{\tau^{(k)}} \xrightarrow{\pi^{(k-1)}} s_{\tau^{(k)}+1} \xrightarrow{\pi^{(k)}} s_{\tau^{(k)}+2}$$
$$\dashrightarrow \ldots \dashrightarrow s_{\tau^{(N)}} \xrightarrow{\pi^{(N-1)}} s_{\tau^{(N)}+1},$$

where the policy $\pi^{(k)}$ is updated to $\pi^{(k+1)}$ after the agent observes $s_{\tau^{(k+1)}+1}$ and is thus ready for the batch update. It is worth pointing out that if the agent kept acting according to the current policy $\pi^{(k)}$ rather than switching to the new policy, the strong Markov property indicates that the sequence $\left\{s_{\tau^{(k)}+t}\right\}_{t\geq 1}$ would be a Markov chain with transition matrix $P^{\pi^{(k)}}$, and would be independent of the past trajectory when conditioned on $s_{\tau^{(k)}+1}$. Meanwhile, by the construction of Algorithm 1, $(\tau^{(k+1)} - \tau^{(k)})$ is a cover time on this extended Markov chain. Hence we can condition on $\mathcal{F}_{(k)}$ and focus on the $(k+1)$-th sample batch.

Appendix B. Duality Gap

The following lemma shows that the optimal difference-of-value function and stationary state-action distribution $\mathbf{h}^*, \mu^*$ form a saddle point to the min-max problem. (4)

Lemma 3 (Wang (2017)) *Under Assumption 1, the optimal primal and dual solutions $\mathbf{h}^*, \mu^*$ to the min-max problem (4) satisfy $\mathbf{h}^* \in \mathcal{H}$ and $\mu^* \in \mathcal{U}$, where*

$$\mathcal{H} = \left\{\mathbf{h} \in \mathcal{R}^{|\mathcal{S}|} \middle| \|\mathbf{h}\|_\infty \leq 2t^*_{mix}\right\},$$
$$\mathcal{U} = \left\{\mu \in \mathcal{R}^{|\mathcal{S}||\mathcal{A}|} \middle| \mathbf{1}^T\mu = 1, \mu \geq 0, \sum_{a\in\mathcal{A}} \mu_a \geq \frac{1}{\tau|\mathcal{S}|}\mathbf{1}\right\}.$$

In what follows, we analyze the averaged duality gap of our algorithm across N batch updates.

Theorem 4 *Let $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathbf{r})$ be an arbitrary MDP tuple satisfying Assumption 1. t^*_{mix} is its worst-case mixing time. Then the sequence of iterates generated by Algorithm 1 with N batch updates satisfies*

$$\frac{1}{N}\sum_{k=0}^{N-1} \mathbf{E}\left[\sum_{a\in\mathcal{A}} (\mathbf{h}^* - P_a\mathbf{h}^* - \mathbf{r}_a + \bar{v}^*\mathbf{1})^T\mu_a^{(k)}\right] \leq \tilde{\mathcal{O}}\left(t^*_{mix}\sqrt{\frac{\tau^3 + |\mathcal{A}|}{N}}\right), \tag{13}$$

when $\alpha = \frac{|\mathcal{S}|t^*_{mix}}{\tau}\sqrt{\frac{2}{3N}}$, $\beta = \sqrt{\frac{\log(|\mathcal{S}||\mathcal{A}|)}{4|\mathcal{A}|(4t^*_{mix}+1)^2N}}$, $M = 4t^*_{mix} + 1$.

Denote $\left\{ (i, a_i^{(k+1)}, j_i^{(k+1)}, r_i^{(k+1)}) \;\middle|\; i \in \mathcal{S} \right\}$ the $(k+1)$-th update batch gathered during $[\tau^{(k)}+1, \tau^{(k+1)}+1]$. Then the primal and dual increments used in Algorithm 1 are equivalent to

$$\mathbf{d}^{(k+1)} = \sum_{i\in\mathcal{S}} \alpha \xi_i^{(k)} (\mathbf{e}_i - \mathbf{e}_{j_i^{(k+1)}}), \tag{14}$$

$$\Delta^{(k+1)} = \sum_{i\in\mathcal{S}} \beta \frac{h_{j_i^{(k+1)}}^{(k)} - h_i^{(k)} + r_i^{(k+1)} - M}{\pi_{i,a_i^{(k+1)}}^{(k)}} \mathbf{e}_{i,a_i^{(k+1)}}. \tag{15}$$

Theorem 4 is proved based on the following lemmas.

Lemma 5 *Wang (2017) The iterates generated by Algorithm 1 satisfy*

$$\mathbf{E}\left[D_{KL}(\mu^*\|\mu^{(k+1)})\right] - D_{KL}(\mu^*\|\mu^{(k)}) \leq \sum_{i\in\mathcal{S}}\sum_{a\in\mathcal{A}} (\mu_{i,a}^{(k)} - \mu_{i,a}^*) \mathbf{E}\left[\Delta_{i,a}^{(k+1)} \middle| \mathcal{F}_{(k)}\right] + \frac{1}{2}\sum_{i\in\mathcal{S}}\sum_{a\in\mathcal{A}} \mu_{i,a}^{(k)} \mathbf{E}\left[\left(\Delta_{i,a}^{(k+1)}\right)^2 \middle| \mathcal{F}_{(k)}\right],$$

for all k with probability 1.

Lemma 6 *The iterates generated by Algorithm 1 satisfy*

$$\sum_{i\in\mathcal{S}}\sum_{a\in\mathcal{A}} \mu_{i,a}^{(k)} \mathbf{E}\left[\left(\Delta_{i,a}^{(k+1)}\right)^2 \middle| \mathcal{F}_{(k)}\right] \leq 4|\mathcal{A}|(4t_{mix}^*+1)^2\beta^2,$$

for all k with probability 1.

Proof From (15) we have

$$\left(\Delta_{i,a}^{(k+1)}\right)^2 = \left(\beta \frac{h_{j_i^{(k+1)}}^{(k)} - h_i^{(k)} + r_i^{(k+1)} - M}{\pi_{i,a}^{(k+1)}} \mathbb{1}_{\{a_i^{(k+1)}=a\}}\right)^2 \leq \beta^2 \frac{4\left(4t_{mix}^*+1\right)^2}{\left(\pi_{i,a}^{(k)}\right)^2} \mathbb{1}_{\{a_i^{(k+1)}=a\}} \quad \forall i \in \mathcal{S}, a \in \mathcal{A}.$$

Then it follows that

$$\begin{aligned}
\sum_{i\in\mathcal{S}}\sum_{a\in\mathcal{A}} \mu_{i,a}^{(k)} \mathbf{E}\left[\left(\Delta_{i,a}^{(k+1)}\right)^2 \middle| \mathcal{F}_{(k)}\right] &\leq \sum_{i\in\mathcal{S}}\sum_{a\in\mathcal{A}} \mu_{i,a}^{(k)} \mathbf{E}\left[\beta^2 \frac{4\left(4t_{mix}^*+1\right)^2}{(\pi_{i,a}^{(k)})^2} \mathbb{1}_{\{a_i^{(k+1)}=a\}} \middle| \mathcal{F}_{(k)}\right] \\
&= \sum_{i\in\mathcal{S}}\sum_{a\in\mathcal{A}} \xi_i^{(k)} \pi_{i,a}^{(k)} \beta^2 \frac{4\left(4t_{mix}^*+1\right)^2}{(\pi_{i,a}^{(k)})^2} \pi_{i,a}^{(k)} \\
&= 4\left(4t_{mix}^*+1\right)^2 \beta^2 \sum_{i\in\mathcal{S}}\sum_{a\in\mathcal{A}} \xi_i^{(k)} \\
&= 4|\mathcal{A}|(4t_{mix}^*+1)^2\beta^2.
\end{aligned}$$

■

Lemma 7 *The generated dual iterates* $\{\mu^k\}_{k\geq 0}$ *satisfy*

$$\mathbf{E}\left[D_{KL}(\mu^*\|\mu^{(k+1)})\middle|\mathcal{F}_{(k)}\right] \leq D_{KL}\left(\mu^*\|\mu^{(k)}\right)+\beta\sum_{a\in\mathcal{A}}\left(\mu_a^{(k)}-\mu_a^*\right)^T\left((P^a-I)\mathbf{h}^{(k)}+\mathbf{r}^a\right)+2|\mathcal{A}|(4t^*_{mix}+1)^2\beta^2, \tag{16}$$

for all k *with probability 1.*

Proof Again from (15) we have

$$\begin{aligned}
\mathbf{E}\left[\Delta_{i,a}^{(k+1)}\middle|\mathcal{F}_{(k)}\right] &= \mathbf{E}\left[\beta\frac{h_{j_i^{(k+1)}}^{(k)}-h_i^{(k)}+r_i^{(k+1)}-M}{\pi_{i,a}^{(k)}}\mathbb{1}_{\{a_i^{(k+1)}=a\}}\middle|\mathcal{F}_{(k)}\right]\\
&= \frac{\beta}{\pi_{i,a}^{(k)}}\mathbf{E}\left[\sum_{j\in\mathcal{S}}\left(h_j^{(k)}-h_i^{(k)}+r_i^{(k+1)}-M\right)\mathbb{1}_{\{a_i^{(k+1)}=a,j_i^{(k+1)}=j\}}\middle|\mathcal{F}_{(k)}\right]\\
&= \frac{\beta}{\pi_{i,a}^{(k)}}\sum_{j\in\mathcal{S}}\left(h_j^{(k)}-h_i^{(k)}+r_i^a-M\right)\pi_{i,a}^{(k)}P_{i,j}^a\\
&= \beta\sum_{j\in\mathcal{S}}\left(h_j^{(k)}-h_i^{(k)}+r_i^a-M\right)P_{i,j}^a.
\end{aligned}$$

Then

$$\begin{aligned}
\sum_{i\in\mathcal{S}}\sum_{a\in\mathcal{A}}(\mu_{i,a}^{(k)}-\mu_{i,a}^*)\mathbf{E}\left[\Delta_{i,a}^{(k+1)}\middle|\mathcal{F}_{(k)}\right] &= \sum_{i\in\mathcal{S}}\sum_{a\in\mathcal{A}}(\mu_{i,a}^{(k)}-\mu_{i,a}^*)\beta\sum_{j\in\mathcal{S}}\left(h_j^{(k)}-h_i^{(k)}+r_i^a-M\right)P_{i,j}^a\\
&= \beta\sum_{a\in\mathcal{A}}\left(\sum_{i\in\mathcal{S}}\sum_{j\in\mathcal{S}}(\mu_{i,a}^{(k)}-\mu_{i,a}^*)P_{i,j}^a h_j^{(k)}+\sum_{i\in\mathcal{S}}(\mu_{i,a}^{(k)}-\mu_{i,a}^*)(-h_i^{(k)}+r_i^a)\right)\\
&\quad-\beta M\sum_{i\in\mathcal{S}}\sum_{a\in\mathcal{A}}(\mu_{i,a}^{(k)}-\mu_{i,a}^*)\\
&= \beta\sum_{a\in\mathcal{A}}\left(\mu_a^{(k)}-\mu_a^*\right)^T\left(P^a\mathbf{h}^{(k)}-\mathbf{h}^{(k)}+\mathbf{r}^a\right).
\end{aligned}$$

Here we used the fact that $\sum_{i\in\mathcal{S}}\sum_{a\in\mathcal{A}}\mu_{i,a}^{(k)}=\sum_{i\in\mathcal{S}}\sum_{a\in\mathcal{A}}\mu_{i,a}^*=1$. Combining Lemma 5, 6 we get (16). ■

Lemma 8 *The generated primal iterates* $\{\mathbf{h}^k\}_{k\geq 0}$ *satisfy*

$$\mathbf{E}\left[\|\mathbf{h}^{(k+1)}-\mathbf{h}^*\|^2\middle|\mathcal{F}_{(k)}\right] \leq \|\mathbf{h}^{(k)}-\mathbf{h}^*\|^2+2\alpha\left(\mathbf{h}^{(k)}-\mathbf{h}^*\right)^T\left(\sum_{a\in\mathcal{A}}(I-P^a)^T\mu_a^{(k)}\right)+\frac{3\tau^3}{|\mathcal{S}|}\alpha^2, \tag{17}$$

for all k *with probability 1.*

Proof We will compute $\mathbf{E}\left[\mathbf{d}^{(k+1)}|\mathcal{F}_{(k)}\right]$ and $\mathbf{E}\left[\|\mathbf{d}^{(k+1)}\|^2|\mathcal{F}_{(k)}\right]$ in step 1,2, respectively, and combine them in step 3 to complete the proof.

Step 1. From (14) we have

$$\begin{aligned}
\mathbf{E}\left[\mathbf{d}^{(k+1)}|\mathcal{F}_{(k)}\right] &= \mathbf{E}\left[\sum_{i\in\mathcal{S}}\alpha\xi_i^{(k)}(\mathbf{e}_i-\mathbf{e}_{j_i^{(k+1)}})\middle|\mathcal{F}_{(k)}\right] \\
&= \mathbf{E}\left[\sum_{i\in\mathcal{S}}\sum_{a\in\mathcal{A}}\sum_{j\in\mathcal{S}}\mathbb{1}_{\{a_i^{(k+1)}=a,j_i^{(k+1)}=j\}}\alpha\xi_i^{(k)}(\mathbf{e}_i-\mathbf{e}_j)\middle|\mathcal{F}_{(k)}\right] \\
&= \alpha\sum_{i\in\mathcal{S}}\sum_{a\in\mathcal{A}}\sum_{j\in\mathcal{S}}\pi_{i,a}^{(k)}P_{i,j}^a\xi_i^{(k)}(\mathbf{e}_i-\mathbf{e}_j) \\
&= \alpha\sum_{a\in\mathcal{A}}(I-P^a)^T\mu_a^{(k)}.
\end{aligned}$$

Step 2. Recall that $\pi^{(k)}$ is the policy under which the batch were obtained. We denote $P^{\pi^{(k)}}$ the transition matrix under this policy, by Assumption 1 we have

$$(P^{\pi^{(k)}})^T\mathbf{1} \le (P^{\pi^{(k)}})^T(\sqrt{\tau}|\mathcal{S}|\nu^{\pi^{(k)}}) = \sqrt{\tau}|\mathcal{S}|\nu^{\pi^{(k)}} \le \sqrt{\tau}|\mathcal{S}|\cdot\frac{\sqrt{\tau}}{|\mathcal{S}|}\mathbf{1} = \tau\mathbf{1}.$$

Hence it hold that

$$\begin{aligned}
&\mathbf{E}\left[\left\|\mathbf{d}^{(k+1)}\right\|^2\middle|\mathcal{F}_{(k)}\right] \\
&= \mathbf{E}\left[\left\|\sum_{i\in\mathcal{S}}\alpha\xi_i^{(k)}(\mathbf{e}_i-\mathbf{e}_{j_i^{(k+1)}})\right\|^2\middle|\mathcal{F}_{(k)}\right] \\
&\le 2\mathbf{E}\left[\left\|\sum_{i\in\mathcal{S}}\alpha\xi_i^{(k)}\mathbf{e}_i\right\|^2\middle|\mathcal{F}_{(k)}\right] + 2\mathbf{E}\left[\left\|\sum_{i\in\mathcal{S}}\alpha\xi_i^{(k)}\mathbf{e}_{j_i^{(k+1)}}\right\|^2\middle|\mathcal{F}_{(k)}\right] \\
&= 2\alpha^2\left(\|\xi^{(k)}\|^2+\sum_{i\in\mathcal{S}}(\xi_i^{(k)})^2\mathbf{E}\left[\|\mathbf{e}_{j_i^{(k+1)}}\|^2\middle|\mathcal{F}_{(k)}\right]+\sum_{i_1\neq i_2}\xi_{i_1}^{(k)}\xi_{i_2}^{(k)}\mathbf{E}\left[\mathbf{e}_{j_{i_1}^{(k+1)}}^T\mathbf{e}_{j_{i_2}^{(k+1)}}\middle|\mathcal{F}_{(k)}\right]\right) \\
&= 2\alpha^2\left(2\|\xi^{(k)}\|^2+\sum_{i_1\neq i_2}\xi_{i_1}^{(k)}\xi_{i_2}^{(k)}\sum_{j\in\mathcal{S}}\mathbb{P}(j_{i_1}^{(k+1)}=j_{i_2}^{(k+1)}=j|\mathcal{F}_{(k)})\right) \\
&\text{(Since } j_{i_1}^{(k+1)} \text{ and } j_{i_2}^{(k+1)} \text{ are independent conditioned on } \mathcal{F}_{(k)},) \\
&= 2\alpha^2\left(2\|\xi^{(k)}\|^2+\sum_{i_1\neq i_2}\sum_{j\in\mathcal{S}}\xi_{i_1}^{(k)}\xi_{i_2}^{(k)}P_{i_1,j}^{\pi^{(k)}}P_{i_2,j}^{\pi^{(k)}}\right) \\
&\le 2\alpha^2\left(2\|\xi^{(k)}\|^2+(\xi^{(k)})^TP^{\pi^{(k)}}(P^{\pi^{(k)}})^T\xi^{(k)}\right) \\
&\text{(Recall } \mu^{(k)}\in\mathcal{U} \implies \xi^{(k)} \le \frac{\sqrt{\tau}}{|\mathcal{S}|}\mathbf{1},)
\end{aligned}$$

$$
\begin{aligned}
&\leq 2\alpha^2\left(2\frac{\tau}{|\mathcal{S}|^2}\mathbf{1}^T\mathbf{1}+\frac{\tau}{|\mathcal{S}|^2}\mathbf{1}^T P^{\pi^{(k)}}(P^{\pi^{(k)}})^T\mathbf{1}\right)\\
&\leq 2\alpha^2\left(\frac{2\tau}{|\mathcal{S}|}+\frac{\tau}{|\mathcal{S}|^2}\tau^2\mathbf{1}^T\mathbf{1}\right)\\
&\leq \frac{6\tau^3}{|\mathcal{S}|}\alpha^2
\end{aligned}
$$

Step 3. According to Algorithm 1, we have

$$\mathbf{h}^{(k+1)}=\underset{\mathcal{H}}{\mathbf{Proj}}(\mathbf{h}^{(k)}+\mathbf{d}^{(k+1)}),$$

where $\mathbf{Proj}_{\mathcal{H}}$ denotes the Euclidean projection onto $\mathcal{H}=\{\mathbf{h}\,|\,\|\mathbf{h}\|_\infty\leq 2t^*_{mix}\}$. Because $\mathcal{H}$ is nonexpansive and $\mathbf{h}^*\in\mathcal{H}$, we have

$$
\begin{aligned}
\mathbf{E}\left[\|\mathbf{h}^{(k+1)}-\mathbf{h}^*\|^2\middle|\mathcal{F}_{(k)}\right]&=\mathbf{E}\left[\|\underset{\mathcal{H}}{\mathbf{Proj}}(\mathbf{h}^{(k)}+\mathbf{d}^{(k+1)})-\mathbf{h}^*\|^2\middle|\mathcal{F}_{(k)}\right]\\
&\leq \mathbf{E}\left[\|\mathbf{h}^{(k)}+\mathbf{d}^{(k+1)}-\mathbf{h}^*\|^2\middle|\mathcal{F}_{(k)}\right]\\
&\leq \|\mathbf{h}^{(k)}-\mathbf{h}^*\|^2+2(\mathbf{h}^{(k)}-\mathbf{h}^*)^T\mathbf{E}\left[\mathbf{d}^{(k+1)}|\mathcal{F}_{(k)}\right]+\mathbf{E}\left[\|\mathbf{d}^{(k+1)}\|^2\middle|\mathcal{F}_{(k)}\right]\\
&\leq \|\mathbf{h}^{(k)}-\mathbf{h}^*\|^2+2\alpha(\mathbf{h}^{(k)}-\mathbf{h}^*)^T\left(\sum_{a\in\mathcal{A}}(I-P^a)^T\mu_a^{(k)}\right)+\frac{6\tau^3}{|\mathcal{S}|}\alpha^2.
\end{aligned}
$$

■

Lemma 9 *We define*

$$\mathcal{E}^{(k)}=D_{KL}(\mu^*\|\mu^{(k)})+\frac{\beta}{2\alpha}\|\mathbf{h}^{(k)}-\mathbf{h}^*\|^2,\quad \mathcal{G}^{(k)}=\sum_{i\in\mathcal{S}}\sum_{a\in\mathcal{A}}\mu_{i,a}^{(k)}(\mathbf{h}^*-P^a\mathbf{h}^*-\mathbf{r}^a)_i+\bar{v}^*.$$

The iterates generated by Algorithm 1 satisfy for all k with probability 1 that

$$\mathbf{E}\left[\mathcal{E}^{(k+1)}\middle|\mathcal{F}_{(k)}\right]\leq\mathcal{E}^{(k)}-\beta\mathcal{G}^{(k)}+4|\mathcal{A}|(4t^*_{mix}+1)^2\beta^2+\frac{3\tau^3}{|\mathcal{S}|}\alpha\beta.$$

Proof (16) + $\frac{\beta}{2\alpha}*$ (17) and use the fact

$$\sum_{a\in\mathcal{A}}\left(\mu_a^{(k)}-\mu_a^*\right)^T\left((P^a-I)\mathbf{h}^{(k)}+\mathbf{r}^a\right)+\left(\mathbf{h}^{(k)}-\mathbf{h}^*\right)^T\left(\sum_{a\in\mathcal{A}}(I-P^a)^T\mu_a^{(k)}\right)=-\mathcal{G}^{(k)},$$

we obtain the inequality. ■

Proof of Theorem 4

Note that $\mu^{(0)}$ is set to be uniform distribution and $\mathbf{h}^{(0)},\mathbf{h}^*\in\mathcal{H}$, hence

$$\mathcal{E}^{(0)}=D_{KL}(\mu^*\|\mu^1)+\frac{\beta}{2\alpha}\|\mathbf{h}^{(0)}-\mathbf{h}^*\|^2\leq\log\left(|\mathcal{S}||\mathcal{A}|\right)+\frac{\beta}{2\alpha}\cdot 4|\mathcal{S}|(t^*_{mix})^2.$$

It follows from Lemma 9 that

$$\mathcal{G}^{(k)} \leq \left(\mathcal{E}^{(k)} - \mathbf{E}\left[\mathcal{E}^{(k+1)}|\mathcal{F}_{(k)}\right]\right)\frac{1}{\beta} + 4|\mathcal{A}|(4t^*_{mix}+1)^2\beta + \frac{3\tau^3}{|\mathcal{S}|}\alpha.$$

Summing over $k = 0, 1, \cdots, N-1$ and taking unconditional expectation, we get

$$\begin{aligned}\frac{1}{N}\sum_{k=0}^{N-1}\mathbf{E}\left[\mathcal{G}^{(k)}\right] &\leq \frac{\mathbf{E}\left[\mathcal{E}^0\right] - \mathbf{E}\left[\mathcal{E}^N\right]}{N\beta} + 4|\mathcal{A}|(4t^*_{mix}+1)^2\beta + \frac{3\tau^3}{|\mathcal{S}|}\alpha \\ &\leq \frac{\log(|\mathcal{S}||\mathcal{A}|)}{N}\frac{1}{\beta} + \frac{2|\mathcal{S}|(t^*_{mix})^2}{N}\frac{1}{\alpha} + 4|\mathcal{A}|(4t^*_{mix}+1)^2\beta + \frac{3\tau^3}{|\mathcal{S}|}\alpha \\ &= \sqrt{\frac{4|\mathcal{A}|(4t^*_{mix}+1)^2\log(|\mathcal{S}||\mathcal{A}|)}{N} + \frac{6\tau^3(t^*_{mix})^2}{N}},\end{aligned}$$

which gives us 13. ■

Appendix C. Analysis of Cumulative Regret In N Batch Updates

Recall that the regret is defined as the difference between the expected cumulative reward and that we could have gained if we ran the optimal policy for the same length of time.

$$R_N := \mathbf{E}^{\pi^*}\left[\sum_{t=1}^{\tau^{(N)}} r'_t\right] - \mathbf{E}^{alg}\left[\sum_{t=1}^{\tau^{(N)}} r_t\right], \tag{18}$$

where the superscripts π^* and alg denote that the trajectories are generated under optimal policy and the agent, respectively. $\tau^{(N)}$ is the stopping time defined as in previous section so that it only depends on the algorithm trajectory.

Theorem 10 *Let N be the number of batch updates, we have the following regret bound for Algorithm 1*

$$R_N = \tilde{\mathcal{O}}\left(t^*_{mix}\tau\sqrt{(\tau^3 + |\mathcal{A}|)N{t^*_{cov}}^2}\right).$$

The proof of Theorem 10 is based on two lemmas. We first state the following Aldous Lemma without proving.

Lemma 11 (Aldous and Fill (2002)) *If τ is a stopping time for a finite and irreducible Markov chain and satisfies $P(X_\tau = a|X_1 = a) = 1$, then*

$$G_\tau(a, x) = \mathbf{E}[\tau|X_1 = a]\nu(x),$$

where $G_\tau := \mathbf{E}[\sum_{t=1}^{\infty}\mathbf{1}_{\{X_t=x,\tau>t\}}]$ is the Green's function and ν is the stationary distribution.

Lemma 12 *Suppose $\{X_t\}_{t\geq 1}$ is a finite ergodic Markov chain on Ω. Let ν and τ_{cov} be its stationary distribution and cover time. Then*

$$\mathbf{E}\left[\sum_{t=1}^{\tau_{cov}}\mathbb{1}_{\{X_t=x\}}\middle| X_1\right] \leq 2\mathbf{E}\left[\tau_{cov}|\, X_1\right]\nu(x)$$

holds with probability 1 for any initial distribution and $x \in \Omega$.

Proof It suffices to show for any fixed initial state a. Define $\tau = \inf\{t > \tau_{cov} | X_t = a\}$. By the strong Markov property we know $(\tau - \tau_{cov})$ is a hitting time of $\{X_{\tau_{cov}+t}\}_{t\geq 1}$, and thus $\mathbf{E}[\tau - \tau_{cov}] \leq \mathbf{E}\left[\tau_{cov} \middle| X_1\right]$, a.s. We apply Lemma 11 and get

$$\begin{aligned}\mathbf{E}\left[\sum_{t=1}^{\tau_{cov}} \mathbb{1}_{\{X_t=x\}} \middle| X_1 = a\right] &\leq \mathbf{E}\left[\sum_{t=1}^{\tau-1} \mathbb{1}_{\{X_t=x\}} \middle| X_1 = a\right] = G_\tau(a,x) = \mathbf{E}[\tau | X_1 = a]\nu(x) \\ &\leq 2\mathbf{E}\left[\tau_{cov} \middle| X_1\right]\nu(x)\end{aligned}$$

■

Proof of Theorem 10

Step 1. We observe that

$$\begin{aligned}&\mathbf{E}^{\pi^*}\left[\sum_{t=1}^{\tau^{(N)}} (r'_t + h^*_{s_{t+1}} - h^*_{s_t} - \bar{v}^*) \middle| \mathcal{F}_1\right] \\ &= \mathbf{E}^{\pi^*}\left[\sum_{t=1}^{\infty} (r'_t + h^*_{s_{t+1}} - h^*_{s_t} - \bar{v}^*)\mathbb{1}_{\{\tau^{(N)}\geq t\}} \middle| \mathcal{F}_1\right] \\ &= \mathbf{E}^{\pi^*}\left[\sum_{t=1}^{\infty} \mathbf{E}^{\pi^*}\left[r'_t + h^*_{s_{t+1}} - h^*_{s_t} - \bar{v}^* \middle| \mathcal{F}_t\right] \mathbb{1}_{\{\tau^{(N)}\geq t\}} \middle| \mathcal{F}_1\right] \\ &= \mathbf{E}^{\pi^*}\left[\sum_{t=1}^{\infty} \left(r^{\pi^*} + \sum_j P^{\pi^*}_{s_t,j} h^*_j - h^*_{s_t} - \bar{v}^*\right) \mathbb{1}_{\{\tau^{(N)}\geq t\}} \middle| \mathcal{F}_1\right] \\ &= 0.\end{aligned}$$

Hence we can simplify the first term in as D

$$\begin{aligned}&\mathbf{E}^{\pi^*}\left[\sum_{t=1}^{\tau^{(N)}} r'_t \middle| \mathcal{F}_1\right] \\ &= \mathbf{E}^{\pi^*}\left[\sum_{t=1}^{\tau^{(N)}} (r'_t + h^*_{s_{t+1}} - h^*_{s_t} - \bar{v}^*) \middle| \mathcal{F}_1\right] + \mathbf{E}^{\pi^*}\left[h^*_{s_1} - h^*_{\tau^{(N)}+1} \middle| \mathcal{F}_1\right] + \mathbf{E}^{\pi^*}\left[\tau^{(N)}\bar{v}^* \middle| \mathcal{F}_1\right] \\ &= h^*_{s_1} - \mathbf{E}^{\pi^*}\left[h^*_{s_{\tau^{(N)}+1}} \middle| \mathcal{F}_1\right] + \mathbf{E}^{alg}\left[\tau^{(N)}\bar{v}^* \middle| \mathcal{F}_1\right].\end{aligned} \tag{19}$$

Step 2. We turn to the second term in (D). For each update batch we have

$$\begin{aligned}&\mathbf{E}^{alg}\left[\sum_{t=\tau^{(k)}+1}^{\tau^{(k+1)}} r_t \middle| \mathcal{F}_{(k)}\right] \\ &= \mathbf{E}^{\pi^{(k)}}\left[\sum_{t=\tau^{(k)}+1}^{\tau^{(k+1)}} (r_t + h^*_{s_{t+1}} - h^*_{s_t}) \middle| \mathcal{F}_{(k)}\right] + h^*_{s_{\tau^{(k)}+1}} - \mathbf{E}^{alg}\left[h^*_{s_{\tau^{(k+1)}+1}} \middle| \mathcal{F}_{(k)}\right]\end{aligned}$$

$$
\begin{aligned}
&= \mathbf{E}^{\pi^{(k)}}\left[\sum_{t=1}^{\tau^{(k+1)}-\tau^{(k)}}\left(r_{\tau^{(k)}+t}+h^*_{s_{\tau^{(k)}+t+1}}-h^*_{s_{\tau^{(k)}+t}}\right)\middle|\mathcal{F}_{(k)}\right]+h^*_{s_{\tau^{(k)}+1}}-\mathbf{E}^{alg}\left[h^*_{s_{\tau^{(k+1)}+1}}\middle|\mathcal{F}_{(k)}\right]\\
&= \mathbf{E}^{\pi^{(k)}}\left[\sum_{t=1}^{\tau^{(k+1)}-\tau^{(k)}}\left(r^{\pi^{(k)}}_{s_{\tau^{(k)}+t}}+(P^{\pi^{(k)}}h^*)_{s_{\tau^{(k)}+t}}-h^*_{s_{\tau^{(k)}+t}}\right)\middle|\mathcal{F}_{(k)}\right]+h^*_{s_{\tau^{(k)}+1}}-\mathbf{E}^{alg}\left[h^*_{s_{\tau^{(k+1)}+1}}\middle|\mathcal{F}_{(k)}\right]\\
&= \sum_{a\in\mathcal{A}}\sum_{i\in\mathcal{S}}\pi^{(k)}_{i,a}\left(\mathbf{r}^a+P^a\mathbf{h}^*-\mathbf{h}^*\right)_i\mathbf{E}^{\pi^{(k)}}\left[\sum_{t=1}^{\tau^{(k+1)}-\tau^{(k)}}\mathbb{1}_{\{s_{\tau^{(k)}+t}=i\}}\middle|\mathcal{F}_{(k)}\right]\\
&\quad+h^*_{s_{\tau^{(k)}+1}}-\mathbf{E}^{alg}\left[h^*_{s_{\tau^{(k+1)}+1}}\middle|\mathcal{F}_{(k)}\right]\\
&= \sum_{a\in\mathcal{A}}\sum_{i\in\mathcal{S}}\pi^{(k)}_{i,a}\left(\mathbf{r}^a+P^a\mathbf{h}^*-\mathbf{h}^*-\bar{v}^*\mathbf{1}\right)_i\mathbf{E}^{\pi^{(k)}}\left[\sum_{t=1}^{\tau^{(k+1)}-\tau^{(k)}}\mathbb{1}_{\{s_{\tau^{(k)}+t}=i\}}\middle|\mathcal{F}_{(k)}\right]\\
&\quad+h^*_{s_{\tau^{(k)}+1}}-\mathbf{E}^{alg}\left[h^*_{s_{\tau^{(k+1)}+1}}\middle|\mathcal{F}_{(k)}\right]+\bar{v}^*\mathbf{E}^{alg}\left[\tau^{(k+1)}-\tau^{(k)}\middle|\mathcal{F}_{(k)}\right]
\end{aligned}
$$

As shown before, $\tau^{(k+1)}-\tau^{(k)}$ is a cover time of $\left\{s_{\tau^{(k)}+t}\right\}_{t\geq 1}$. Let $\nu^{(k)}$ denote the stationary distribution under policy $\pi^{(k)}$, applying Lemma 12 we get

$$
\mathbf{E}^{\pi^{(k)}}\left[\sum_{t=1}^{\tau^{(k+1)}-\tau^{(k)}}\mathbb{1}_{\{s_{\tau^{(k)}+t}=i\}}\middle|\mathcal{F}_{(k)}\right]\leq 2\mathbf{E}\left[\tau^{(k+1)}-\tau^{(k)}\middle|\mathcal{F}_{(k)}\right]\nu^{(k)}_i\leq 2t^*_{cov}\nu^{(k)}_i,\quad\forall i\in\mathcal{S}.
$$

Recall that $\mathbf{h}^*$ is also the solution of the Bellman equation

$$
h^*_i=\max_{a\in\mathcal{A}}\left\{\sum_{j\in\mathcal{S}}P^a_{i,j}\left(r^a_i-\bar{v}^*+h^*_j\right)\right\},\quad\forall i\in\mathcal{S},
$$

which implies that

$$
\left(\mathbf{r}^a+P^a\mathbf{h}^*-\mathbf{h}^*-\bar{v}^*\mathbf{1}\right)_i\leq 0,\quad\forall i\in\mathcal{S}.
$$

Therefore

$$
\begin{aligned}
&\mathbf{E}^{alg}\left[\sum_{t=1}^{\tau^{(N)}}r_t\middle|\mathcal{F}_1\right]\\
&=\mathbf{E}^{alg}\left[\sum_{k=0}^{N-1}\mathbf{E}^{alg}\left[\sum_{t=\tau^{(k)}+1}^{\tau^{(k+1)}}r_t\middle|\mathcal{F}_{(k)}\right]\middle|\mathcal{F}_1\right]\\
&\geq\sum_{k=0}^{N-1}\mathbf{E}^{alg}\left[\sum_{a\in\mathcal{A}}\sum_{i\in\mathcal{S}}\pi^{(k)}_{i,a}\left(\mathbf{r}^a+P^a\mathbf{h}^*-\mathbf{h}^*-\bar{v}^*\mathbf{1}\right)_i\cdot 2t^*_{cov}\nu^{(k)}_i\middle|\mathcal{F}_1\right] \qquad (20)\\
&\quad+\mathbf{E}^{alg}\left[h^*_{s_{\tau^{(0)}+1}}-h^*_{s_{\tau^{(N)}+1}}\middle|\mathcal{F}_1\right]+\bar{v}^*\mathbf{E}^{alg}\left[\tau^{(N)}-\tau^{(0)}\middle|\mathcal{F}_1\right]\\
&\geq 2t^*_{cov}\tau\sum_{k=0}^{N-1}\mathbf{E}^{alg}\left[\sum_{a\in\mathcal{A}}\sum_{i\in\mathcal{S}}\left(\mathbf{r}^a+P^a\mathbf{h}^*-\mathbf{h}^*-\bar{v}^*\mathbf{1}\right)_i\mu^{(k)}_{i,a}\middle|\mathcal{F}_1\right]\\
&\quad+h^*_{s_1}-\mathbf{E}^{alg}\left[h^*_{s_{\tau^{(N)}+1}}\middle|\mathcal{F}_1\right]+\bar{v}^*\mathbf{E}^{alg}\left[\tau^{(N)}\middle|\mathcal{F}_1\right]
\end{aligned}
$$

where the last inequality comes from the fact that both the stationary state-action distribution $\left(\pi_{i,a}^{(k)}\nu_i^{(k)}\right)_{i\in\mathcal{S},a\in\mathcal{A}}$ and the dual iterate μ belong to the dual feasible set $\mathcal{U}$, and differ at most by a multiplier τ as specified in Assumption 1.

Combining (19) and (20), we obtain

$$R_N \leq 2t_{cov}^*\tau \sum_{k=0}^{N-1} \mathbf{E}^{alg}\left[\sum_{a\in\mathcal{A}} \left(\mathbf{h}^* - P^a\mathbf{h}^* - \mathbf{r}^a + \bar{v}^*\mathbf{1}\right)^T \mu_a^{(k)}\right] + \mathbf{E}^{log}\left[h^*_{s_{\tau^{(N)}+1}}\right] - \mathbf{E}^{\pi^*}\left[h^*_{s_{\tau^{(N)}+1}}\right].$$

Applying Theorem 4 we complete the proof. ■

Appendix D. Analysis of Cumulative Regret In T Time Steps

We now consider the case when Algorithm 1 stops right after T steps, rather than completing a certain number of batch updates. Since every batch update involves at least $|\mathcal{S}|$ steps, we know that the algorithm at most conducts $\left\lfloor \frac{T}{|\mathcal{S}|} \right\rfloor$ updates. Let $N = \left\lfloor \frac{T}{|\mathcal{S}|} \right\rfloor + 1$, we thus have $\tau^{(N)} > T$ following the notation in Appendix A.

Let $K := \min\left\{k \geq 0 | T \leq \tau^{(k)}\right\} \leq N$ denote the update that the algorithm began to collect data for but did not finish. The regret of Algorithm 1 has the following decomposition

$$\begin{aligned} R_T &= \mathbf{E}^{\pi^*}\left[\sum_{t=1}^{T} r'_t\right] - \mathbf{E}^{alg}\left[\sum_{t=1}^{T} r_t\right] \\ &= \left(\mathbf{E}^{\pi^*}\left[\sum_{t=1}^{\tau^{(N)}} r'_t\right] - \mathbf{E}^{alg}\left[\sum_{t=1}^{\tau^{(N)}} r_t\right]\right) \\ &\quad + \left(\mathbf{E}^{alg}\left[\sum_{t=T+1}^{\tau^{(K)}} r_t\right] - \mathbf{E}^{\pi^*}\left[\sum_{t=T+1}^{\tau^{(K)}} r'_t\right]\right) + \left(\mathbf{E}^{alg}\left[\sum_{t=\tau^{(K)}+1}^{\tau^{(N)}} r_t\right] - \mathbf{E}^{\pi^*}\left[\sum_{t=\tau^{(K)}+1}^{\tau^{(N)}} r'_t\right]\right). \end{aligned}$$

The first term has been shown in Appendix C to be bounded by

$$\tilde{\mathcal{O}}\left(t_{mix}^*\tau\sqrt{(\tau^3 + |\mathcal{A}|)N{t_{cov}^*}^2}\right).$$

The second term can be easily bounded by t_{cov}^* due to the fact that $r_t \in [0,1]$.

$$\mathbf{E}^{alg}\left[\sum_{t=T+1}^{\tau^{(K)}} r_t\right] - \mathbf{E}^{\pi^*}\left[\sum_{t=T+1}^{\tau^{(K)}} r'_t\right] \leq \mathbf{E}^{alg}\left[\sum_{t=\tau^{(K-1)}+1}^{\tau^{(K)}} r_t\right] - 0 \leq \mathbf{E}^{alg}\left[\tau^{(K)} - \tau^{(K-1)}\right] \leq t_{cov}^*.$$

The third term is indeed the negative regret of the algorithm between $t = \tau^{(K)} + 1$ and $\tau^{(N)}$, except that the optimal policy starts from a different state at $t = \tau^{(K)} + 1$. We construct a third

"concatenated" policy $\hat{\pi}$ that picks actions according to the algorithm before $t = \tau^{(K)} + 1$, and follows π^* thereafter. Then we have

$$\mathbf{E}^{alg}\left[\sum_{t=\tau^{(K)}+1}^{\tau^{(N)}} r_t\right] - \mathbf{E}^{\pi^*}\left[\sum_{t=\tau^{(K)}+1}^{\tau^{(N)}} r'_t\right]$$
$$= \left(\mathbf{E}^{alg}\left[\sum_{t=\tau^{(K)}+1}^{\tau^{(N)}} r_t\right] - \mathbf{E}^{\hat{\pi}}\left[\sum_{t=\tau^{(K)}+1}^{\tau^{(N)}} r''_t\right]\right) + \left(\mathbf{E}^{\hat{\pi}}\left[\sum_{t=\tau^{(K)}+1}^{\tau^{(N)}} r''_t\right] - \mathbf{E}^{\pi^*}\left[\sum_{t=\tau^{(K)}+1}^{\tau^{(N)}} r'_t\right]\right),$$

where the first term on the right hand side is the negative regret, and thus negative. The second term is the difference of the cumulative returns along two trajectories generated by the optimal policy starting from different states. From the theory of MDP Puterman (2014) we know that it can be controlled by $2\|\mathbf{h}^*\|_\infty = \mathcal{O}(t^*_{mix})$.

From above we have the following bound of the Algorithm 1's regret in T time steps.

$$R_T = \tilde{\mathcal{O}}\left(t^*_{mix}\tau\sqrt{(\tau^3 + |\mathcal{A}|)\frac{T}{|\mathcal{S}|}{t^*_{cov}}^2}\right). \tag{21}$$

Appendix E. Controlling Worst-Case Cover Time

The following theorem says that for fast-mixing MDP, the worst-cast cover time is roughly the same as the size of the state space.

Theorem 13 *Let t^*_{mix}, t^*_{cov} be the worst-case mixing time and worst-case cover time, under Assumption 1 we have*

$$t^*_{cov} = \tilde{\mathcal{O}}\left(\sqrt{\tau}t^*_{mix}\,|\mathcal{S}|\right). \tag{22}$$

In the rest of this section we denote t_{mix}, t_{hit} and t_{cov} the worst-case mixing time, worst-case hitting time and worst-case cover time of the Markov chain considered in the context.

Lemma 14 *Matthews Let $\{X_t\}_{t\geq 1}$ be an irreducible finite Markov chain on Ω. t_{cov} and t_{hit} are the worst-case cover time and worst-case hitting time, respectively. Then*

$$t_{cov} \leq t_{hit}\left(1 + \frac{1}{2} + ... + \frac{1}{|\Omega| - 1}\right) \leq t_{hit}\left(\log(|\Omega|) + 1\right). \tag{23}$$

Lemma 15 *Suppose the transition matrix P on a finite space Ω is ergodic with stationary distribution ν. Define the ϵ-mixing time $t_{mix}(\epsilon) = \min\left\{t \geq 1 \,\middle|\, \|P^t(i,\cdot) - \nu\|_{TV} \leq \epsilon,\ \forall i \in \Omega\right\}$. Then*

$$t_{mix}(\epsilon) \leq \left\lceil \log_2 \epsilon^{-1} \right\rceil t_{mix},$$

where $t_{mix} := t_{mix}\left(\frac{1}{4}\right)$ is the worst-case mixing time.

Proof See Levin et al.. ■

Lemma 16 *Let $\{X_t\}_{t\geq 1}$ be an ergodic finite Markov chain on Ω with transition matrix P, worst-case hitting time t_{hit} and worst-case cover time t_{cov}. Suppose its stationary distribution ν satisfies $\nu \geq \frac{1}{\sqrt{\tau}|\Omega|}\mathbf{1}$ for some constant $\tau \geq 1$. Then*

$$t_{hit} \leq 9 \log\left(\sqrt{\tau}|\Omega|\right) \sqrt{\tau} t_{mix} \, |\Omega| \, . \tag{24}$$

Proof Let $\epsilon_0 = \frac{1}{4\sqrt{\tau}|\Omega|}$ and $t_0 = t_{mix}(\epsilon)$. Then by definition it holds that for any $i, j \in \Omega$

$$\left|P^{t_0}(i,j) - \nu_j\right| \leq 2 \left\|P^{t_0}(i,\cdot) - \nu\right\|_{TV} \leq 2\epsilon = \frac{1}{2\sqrt{\tau}|\Omega|}.$$

Hence

$$P^{t_0}(i,j) \geq \nu_j - \left|P^{t_0}(i,j) - \nu_j\right| \geq \frac{1}{\sqrt{\tau}|\Omega|} - \frac{1}{2\sqrt{\tau}|\Omega|} = \frac{1}{2\sqrt{\tau}|\Omega|},$$

which follows that

$$P\left(X_{t+t_0} = j \middle| \, X_t = i\right) \geq \frac{1}{2\sqrt{\tau}|\Omega|}, \ \forall \, t.$$

Fix any $j \in \Omega$, let τ_{hit} be the hitting time of j. Then for any $n \in \mathbb{N}$

$$\begin{aligned} P(\tau_{hit} > 1 + t_0 n) &\leq P(X_1 \neq j) \prod_{k=1}^{n} P\left(X_{1+t_0 k} \neq j \middle| \, X_{1+t_0(k-1)} \neq j\right) \\ &\leq 1 \prod_{k=1}^{n} \left(1 - \frac{1}{2\sqrt{\tau}|\Omega|}\right) = \left(1 - \frac{1}{2\sqrt{\tau}|\Omega|}\right)^n. \end{aligned}$$

Therefore we have

$$\begin{aligned} \mathbf{E}\left[\tau_{hit}\right] &= \sum_{t=0}^{\infty} P(\tau_{hit} > t) \\ &= 1 + \sum_{n=0}^{\infty} \sum_{t=1+t_0 n}^{t_0(n+1)} P(\tau_{hit} > t) \\ &\leq 1 + \sum_{n=0}^{\infty} t_0 P(\tau_{hit} > 1 + t_0 n) \\ &\leq 1 + \sum_{n=0}^{\infty} t_0 \left(1 - \frac{1}{2\sqrt{\tau}|\Omega|}\right)^n \\ &= 1 + 2 t_0 \sqrt{\tau}|\Omega|. \end{aligned}$$

Since the above holds for any $j \in \Omega$, we have $t_{hit} \leq 1 + 2t_0\sqrt{\tau}|\Omega|$.
Meanwhile, t_0 is bounded by Lemma 15

$$t_0 \leq \left\lceil \log_2 \epsilon_0^{-1} \right\rceil t_{mix} \leq 4 \log\left(\sqrt{\tau}|\Omega|\right) t_{mix},$$

which completes the proof. ■

Proof of Theorem 13

We know that for any stationary policy π, the Markov chain defined by P^π on the state space $\mathcal{S}$ satisfies the conditions of Lemma 14 and 16. Combining (23), (24) we have

$$t^*_{cov} \leq 9 \log\left(\sqrt{\tau}|\mathcal{S}|\right) \sqrt{\tau} t_{mix} \, |\mathcal{S}| \left(\log(|\mathcal{S}|) + 1\right),$$

which implies (22). ■

In summary, we have proved the following regret bound for Algorithm 1:

$$R(T) = \tilde{\mathcal{O}}\left((t^*_{mix})^2 \, \tau^{\frac{3}{2}} \sqrt{(\tau^3 + |\mathcal{A}|)|\mathcal{S}|T}\right). \tag{25}$$