Active Model Estimation in Markov Decision Processes

Jean Tarbouriech
Facebook AI Research & Inria Lille, SequeL team

Shubhanshu Shekhar
University of California, San Diego

Matteo Pirotta
Facebook AI Research

Mohammad Ghavamzadeh
Facebook AI Research

Alessandro Lazaric
Facebook AI Research

Abstract

We study the problem of efficient exploration in order to learn an accurate model of an environment, modeled as a Markov decision process (MDP). Efficient exploration in this problem requires the agent to identify the regions in which estimating the model is more difficult and then exploit this knowledge to collect more samples there. In this paper, we formalize this problem, introduce the first algorithm to learn an ϵ-accurate estimate of the dynamics, and provide its sample complexity analysis. While this algorithm enjoys strong guarantees in the large-sample regime, it tends to have a poor performance in early stages of exploration. To address this issue, we propose an algorithm that is based on maximum weighted entropy, a heuristic that stems from common sense and our theoretical analysis. The main idea here is to cover the entire state-action space with the weight proportional to the noise in the transitions. Using a number of simple domains with heterogeneous noise in their transitions, we show that our heuristic-based algorithm outperforms both our original algorithm and the maximum entropy algorithm in the small sample regime, while achieving similar asymptotic performance as that of the original algorithm.

1 INTRODUCTION

In most decision problems, the agent is provided with a goal that it tries to achieve by maximizing a reward signal. In such problems, the agent explores the environment in order to identify the high reward situations and reach the goal faster and more efficiently. Although solving a problem (achieving a goal) is usually the ultimate objective, it is sometimes equally important for an agent to understand its environment without pursuing any goals. In such scenarios, no reward function is defined and the agent explores the state-action space in order to discover what is possible and how the environment works. We refer to this scenario as *reward-free* or *unsupervised* exploration. Several objectives have been studied in reward-free exploration, including discovering incrementally reachable states (Lim and Auer, 2012), uniformly covering the state space (Hazan et al., 2019), estimating state-dependent random variables (Tarbouriech and Lazaric, 2019), a broad class of objectives that are defined as functions of the state visitation frequency induced by the agent's behavior (Cheung, 2019a,b), and learning a model of the environment that is suitable for computing near-optimal policies for a given collection of reward functions (Jin et al., 2020). A reward-free exploration algorithm is evaluated by the amount of exploration it uses to learn its objective.

None of the works above, except (Jin et al., 2020), focuses on learning the dynamics of an environment modeled as a Markov decision process (MDP). Even in (Jin et al., 2020), the setting is the simpler finite-horizon MDP, where many states are often irrelevant as they have no impact in defining the optimal policy for any reward functions. Although the problem of learning the dynamics has not been rigorously studied in the reward-free exploration setting, (Araya-López et al., 2011) proposed several heuristics for this problem. Studying active exploration for model estimation is also important in the theoretical understanding of the *simulation-to-real* problem, where the goal is to start with an inaccurate model (simulator) of the environment and learn a better one with minimum interaction with the world.

In this paper, we formalize the problem of reward-free exploration where the objective is to estimate both a uniformly accurate (minimizing the maximum error) and an average accurate (minimizing the average error) model of the environment, i.e., the transition probability function of the MDP. We identify the form of the model estimation error for a given policy and show that it depends on how often the policy visits noisy state-action pairs, i.e., those

Proceedings of the 36th Conference on Uncertainty in Artificial Intelligence (UAI), PMLR volume 124, 2020.

whose transition probability has high variance. Since optimizing the model estimation error over the policies is difficult, we upper bound it using Bernstein's inequality and obtain an objective function that relates the structure of the MDP with the accuracy of a model estimated by a certain state-action visitation (stationary distribution of a policy). We build on (Tarbouriech and Lazaric, 2019) and propose an algorithm that optimizes this objective over stationary distributions and prove sample complexity bounds for the accuracy (on average and in worst case) of the estimated model. In particular, our analysis highlights the intrinsic difficulty of the model estimation problem.

Our "exact" algorithm may be inefficient due to the specific form of the objective function we optimize and it tends to perform well only in large sample regimes (asymptotically). Furthermore, our theoretical guarantees only hold under restrictive assumptions on the MDP (i.e., ergodicity). To alleviate these limitations, we replace the objective function used by our algorithm with maximizing *weighted entropy*, where the goal is to visit state-action pairs weighted by the noise in their transitions. Although this is a heuristic, it stems from our derived objective function and is more tailored to the model estimation problem than the popular maximum entropy objective (Cheung, 2019a; Hazan et al., 2019) that uniformly covers the state space. We derive an algorithm based on the maximum weighted entropy heuristic, by modifying and extending the algorithm in (Cheung, 2019a), and we prove regret guarantees w.r.t. the stationary distribution maximizing the weighted entropy.

The main contributions of this paper are: **1)** We introduce an objective function and use it to derive and analyze an algorithm for accurate MDP model estimation in the reward-free exploration setting. The algorithm extends the one in (Tarbouriech and Lazaric, 2019) by removing the requirement of knowing the model. **2)** When we switch from our original objective function to the maximum weighted entropy heuristic, we show how the algorithm in (Cheung, 2019a) can be modified to handle an unknown objective (since the weights, i.e., noise in transitions are not known in advance).[1] **3)** We show that the algorithm based on the maximum weighted entropy heuristic outperforms both our theoretically driven algorithm and the maximum entropy algorithm in the small sample regime, while achieving similar asymptotic performance as that of our exact algorithm. Finally, **4)** we consider an undiscounted infinite-horizon setting, which is more general than the finite horizon MDP setting considered in the recent work (Jin et al., 2020), where a large portion of the states can be considered as irrelevant.

2 PROBLEM FORMULATION

We consider a reward-free finite *Markov decision process* (MDP) $M := \langle \mathcal{S}, \mathcal{A}, p \rangle$, where $\mathcal{S}$ is the set of S states, $\mathcal{A}$ is the set of A actions. Each state-action pair $(s, a) \in \mathcal{S} \times \mathcal{A}$ is characterized by an *unknown* transition probability distribution $p(\cdot \mid s, a)$ over next states. We denote by π a stochastic non-stationary policy that maps a history $(s_0, a_0, s_1, \ldots, s_t)$ of states and actions observed up to time t to a distribution over actions $\mathcal{A}$. We also consider stochastic *stationary* policies $\pi : \mathcal{S} \to \Delta(\mathcal{A})$, which map the current state s_t to a distribution over actions.

At a high-level, the objective of the agent is to estimate as accurately as possible the unknown transition dynamics. We refer to this objective as the *model estimation problem*, or ModEst for short. More formally, the agent interacts with the environment by following a possibly non-stationary policy π and, after n time steps, it returns an estimate $\widehat{p}_{\pi,n}$ of the transition dynamics. We evaluate the accuracy of $\widehat{p}_{\pi,n}$ in terms of its ℓ_1-norm distance (i.e., total variation) w.r.t. the model p either on average or in the worst-case over the state-action space:

$$\mathcal{E}_{\pi,n} := \frac{1}{SA} \sum_{(s,a)\in\mathcal{S}\times\mathcal{A}} \|\widehat{p}_{\pi,n}(\cdot|s,a) - p(\cdot|s,a)\|_1, \quad (1)$$

$$\mathcal{W}_{\pi,n} := \max_{(s,a)\in\mathcal{S}\times\mathcal{A}} \|\widehat{p}_{\pi,n}(\cdot|s,a) - p(\cdot|s,a)\|_1. \quad (2)$$

We say that an estimate $\widehat{p}_{\pi,n}$ is (ϵ, δ)-accurate in the sense of $\mathcal{E}$ (resp. $\mathcal{W}$) if $\mathbb{P}\big(\mathcal{E}_{\pi,n} \le \epsilon\big) \ge 1 - \delta$, i.e., it is ϵ-accurate with at least $1 - \delta$ probability (resp. $\mathcal{W}_{\pi,n}$). We then measure the sample complexity of a policy π as follows.

Definition 1. *Given an error $\epsilon > 0$ and a confidence level $\delta \in (0, 1)$, the* ModEst *sample complexity of an agent executing a policy π is defined as*

$$\mathcal{C}^{\mathcal{E}}_{\text{ModEst}}(\pi, \epsilon, \delta) := \min\Big\{ n \ge 1 : \mathbb{P}\big(\mathcal{E}_{\pi,n} \le \epsilon\big) \ge 1 - \delta \Big\},$$

$$\mathcal{C}^{\mathcal{W}}_{\text{ModEst}}(\pi, \epsilon, \delta) := \min\Big\{ n \ge 1 : \mathbb{P}\big(\mathcal{W}_{\pi,n} \le \epsilon\big) \ge 1 - \delta \Big\}.$$

According to the definition above, the objective is to find an algorithm that is able to return an (ϵ, δ)-accurate model with as little number of samples as possible, i.e., minimize the sample complexity. While the definition above follows a standard PAC-formulation, it is possible to consider the case when a fixed budget n is provided in advance and the agent should return an estimate which is as accurate as possible with confidence $1 - \delta$. In the following, we consider both settings.

Discussion. While both $\mathcal{E}_{\pi,n}$ and $\mathcal{W}_{\pi,n}$ effectively formalize the objective of accurate estimation of the dynamics, they have specific advantages and disadvantages.

If $\mathcal{W}_{\pi,n}$ is below ϵ, it is possible to compute near-optimal policies for any reward (e.g., Kearns and Singh, 2002).

[1] Interestingly, the algorithm in (Cheung, 2019a) *cannot* be applied to the original model estimation problem, as the objective function violates the assumptions in (Cheung, 2019a).

Proposition 1. *For any $\gamma \in (0,1)$ and reward function $r(s,a) \in [0,1]$, let $\widehat{\pi}$ be the infinite-horizon γ-discounted optimal policy computed using the dynamics $\widehat{p}$ returned after the* ModEst *phase. If $\widehat{p}$ is (ϵ, δ)-accurate in the sense of $\mathcal{W}$, then $\widehat{\pi}$ is guaranteed to be $O(\epsilon(1-\gamma)^{-2})$-optimal in the exact MDP with probability at least $1-\delta$.*

While this shows the advantage of targeting the worst-case accuracy, the maximum over state-action pairs in the formulation of $\mathcal{W}_{\pi,n}$ makes it more challenging to optimize than $\mathcal{E}_{\pi,n}$. Furthermore, consider the case where one single state-action pair has a very noisy dynamics but is overall uninteresting (e.g., the so-called noisy TV case), then targeting the worst-case error may require the agent to spend most of its time in estimating the dynamics of that single state-action pair, instead of collecting information about the rest of the MDP. A similar issue may rise in large MDPs, where lowering the estimation error uniformly over the whole state-action space may be prohibitive and lead to extremely large sample complexity.

Alternative metrics other than ℓ_1 could be used to measure the distance between $\widehat{p}_{\pi,n}$ and p. For instance, given the categorical nature of the transition distributions, a natural choice is to use the KL-divergence. Nonetheless, it has been shown that the KL-divergence is not a well-suited metrics for active learning as it promotes strategies close to uniform sampling (Shekhar et al., 2020). Furthermore, other metrics may not provide any performance guarantees such as Prop. 1.

One may wonder why we rely on Def. 1 to measure the accuracy, instead of the closely related objective defined in (Tarbouriech and Lazaric, 2019), where the *expected* estimation error is used in place of its high-probability version. A careful inspection of the analysis in (Tarbouriech and Lazaric, 2019) shows that they critically rely on the independence between the stochastic transitions in the MDP and the randomness in the observations collected in each state. This argument cannot be applied to our case, where the observations of interest are the next state s' generated during transitions (s,a) to s'. This makes transitions and observations intrinsically correlated and it prevents us from using their analysis.

While minimizing the sample complexity $\mathcal{C}_{\text{ModEst}}$ over policies π is a well-posed objective, the dependency on π is implicit and the space of non-stationary policies is combinatorial in states, actions and time, thus making it impossible to directly optimize for $\mathcal{C}_{\text{ModEst}}$. In the next section we refine the error definitions to derive an explicit convex surrogate for stationary policies.

2.1 A Surrogate Convex Objective Function

We focus on model estimates defined as the empirical frequency of transitions. After n steps, for any state-action pair (s,a) and any next state s', we define

$$\widehat{p}_{\pi,n}(s'|s,a) := \frac{T_{\pi,n}(s,a,s')}{T^{+}_{\pi,n}(s,a)}, \tag{3}$$

where $T_{\pi,n}(s,a) := \sum_{t=1}^{n} \mathbb{1}_{(s_t,a_t)=(s,a)}$ denotes the (random) number of times action a was taken in state s during the execution of policy π over n steps. Similarly, we define $T_{\pi,n}(s,a,s') := \sum_{t=1}^{n} \mathbb{1}_{(s_t,a_t,s_{t+1})=(s,a,s')}$ and we define $x^{+} := \max\{x,1\}$.

We introduce the notion of "noise" associated to each state-action pair.

Definition 2. *For each state-action pair (s,a), we define the transitional noise at (s,a) as*

$$V(s,a) := \sum_{s' \in \mathcal{S}} \sqrt{\sigma_p^2(s'|s,a)} / \sqrt{S},$$

where $\sigma_p^2(s'|s,a)$ is the variance of the transition from (s,a) to s', i.e., $\sigma_p^2(s'|s,a) := p(s'|s,a)(1-p(s'|s,a))$.

Denote by $\Gamma(s,a) := \|p(\cdot|s,a)\|_0$ the support of $p(\cdot|s,a)$, as well as $\Gamma := \max_{s,a} \Gamma(s,a)$. From the Cauchy-Schwarz inequality, we have

$$0 \le V(s,a) \le \sqrt{\Gamma(s,a)-1}/\sqrt{S} \le 1,$$

where the lower bound $V(s,a) = 0$ is achieved for all $(s,a) \in \mathcal{S} \times \mathcal{A}$ when M is a deterministic MDP.

We connect the error functions to a term depending on the transitional noise and the number of state-action visits.

Proposition 2. *For any fixed n and any policy π, with probability at least $1-\delta/3$,*

$$\mathcal{E}_{\pi,n} \le \frac{4}{SA} \log\left(\frac{6SAn}{\delta}\right) \sum_{s,a} \mathcal{F}(s,a;T_{\pi,n}),$$

$$\mathcal{W}_{\pi,n} \le 4 \log\left(\frac{6SAn}{\delta}\right) \max_{s,a} \mathcal{F}(s,a;T_{\pi,n}),$$

where

$$\mathcal{F}(s,a;T) := \frac{V(s,a)}{\sqrt{T(s,a)+1}} + \frac{S}{T(s,a)+1}.$$

This upper bound provides a much more explicit dependency between the behavior of π, the structure of the MDP, and the accuracy of the model estimation. In particular, it illustrates how rarely visited pairs with large variance lead to larger estimation errors.

Convex reparametrization. Prop. 2 suggests that $\mathcal{E}_{\pi,n}$ (resp. $\mathcal{W}_{\pi,n}$) could be optimized by directly minimizing the average (resp. the maximum) of its upper-bound $\mathcal{F}(s,a;T)$. While the function $\mathcal{F}$ is convex in T, the constraint that $T_{\pi,n}$ must be a valid state-action counter renders the optimization problem of minimizing $\mathcal{F}(T_{\pi,n})$ non-convex. To circumvent this issue, we introduce the state-action stationary distribution $\lambda_\pi \in \Delta(\mathcal{S} \times \mathcal{A})$ of a stationary policy π. Also known as an occupancy measure, $\lambda_\pi(s,a)$ is the expected frequency with which action a is executed in state s while following policy π. Let $\Lambda^{(p)}$ be the set of state-action stationary distributions, i.e.,

$$\Lambda^{(p)} := \Big\{ \lambda \in \Delta(\mathcal{S} \times \mathcal{A}) : \forall s \in \mathcal{S}, \\ \sum_{b \in \mathcal{A}} \lambda(s,b) = \sum_{(s',a) \in \mathcal{S} \times \mathcal{A}} p(s|s',a)\lambda(s',a) \Big\}.$$

We introduce a convenient function

$$\mathcal{G}_n(s,a;\lambda) := \frac{V(s,a)}{\sqrt{\lambda(s,a) + \frac{1}{n}}} + \frac{1}{\sqrt{n}} \frac{S}{\lambda(s,a) + \frac{1}{n}},$$

and its associated functions

$$\mathcal{L}_n^{\mathcal{E}}(\lambda) := \frac{1}{SA} \sum_{s,a} \mathcal{G}_n(s,a;\lambda),$$
$$\mathcal{L}_n^{\mathcal{W}}(\lambda) := \max_{s,a} \mathcal{G}_n(s,a;\lambda).$$

Let the empirical state-action frequency at any time t be $\widetilde{\lambda}_{\pi,t}(s,a) = T_{\pi,t}(s,a)/t$, then it is easy to see that

$$\mathcal{E}_{\pi,n} \leq \frac{4}{\sqrt{n}} \log\Big(\frac{6SAn}{\delta}\Big) \mathcal{L}_n^{\mathcal{E}}(\widetilde{\lambda}_{\pi,n}),$$
$$\mathcal{W}_{\pi,n} \leq \frac{4}{\sqrt{n}} \log\Big(\frac{6SAn}{\delta}\Big) \mathcal{L}_n^{\mathcal{W}}(\widetilde{\lambda}_{\pi,n}).$$

This suggests that we could directly minimize $\mathcal{E}_{\pi,n}$ (resp. $\mathcal{W}_{\pi,n}$) by optimizing the function $\mathcal{L}_n^{\mathcal{E}}$ (resp. $\mathcal{L}_n^{\mathcal{W}}$) over λ in the set of stationary distributions $\Lambda^{(p)}$ as follows

$$\min_{\lambda \in \Lambda^{(p)}} \mathcal{L}_n^{\mathcal{E}}(\lambda), \qquad \min_{\lambda \in \Lambda^{(p)}} \mathcal{L}_n^{\mathcal{W}}(\lambda). \tag{$\dagger$}$$

These functions are convex in both the objective function and the constraints and we denote by $\lambda_{\dagger,n}^{\mathcal{E}}$ and $\lambda_{\dagger,n}^{\mathcal{W}}$ their solutions. Roughly speaking, the optimal distributions should visit more often state-action pairs associated to a larger transitional noise.

Beside $\mathcal{L}_n$, we also use closely related functions obtained as asymptotic relaxation for $n \to \infty$ and define the optimization problems

$$\min_{\lambda \in \Lambda^{(p)}} \mathcal{L}_\infty^{\mathcal{E}}(\lambda) := \frac{1}{SA} \sum_{s,a} \frac{V(s,a)}{\sqrt{\lambda(s,a)}},$$
$$\min_{\lambda \in \Lambda^{(p)}} \mathcal{L}_\infty^{\mathcal{W}}(\lambda) := \max_{s,a} \frac{V(s,a)}{\sqrt{\lambda(s,a)}}. \tag{$\star$}$$

We refer to $\lambda_\star^{\mathcal{E}}$ and $\lambda_\star^{\mathcal{W}}$ the corresponding optimal state-action stationary distributions.

3 LEARNING TO OPTIMIZE ModEst

In this section, we build on (Berthet and Perchet, 2017; Tarbouriech and Lazaric, 2019), introduce a learning algorithm for the ModEst problem and prove its sample complexity for both $\mathcal{E}$ and $\mathcal{W}$ error functions. As the structure of the algorithm is the same for both cases, we describe it in its general form.

3.1 Setting the Stage

Leveraging (†) to design an algorithm minimizing the estimation error requires addressing two issues: **1)** the transitional noise $V(s,a)$ in the objective function and the transition dynamics p appearing in the constraint are unknown, **2)** despite $\mathcal{L}_n^{\mathcal{E}}$ and $\mathcal{L}_n^{\mathcal{W}}$ being convex, they are both poorly conditioned and difficult to optimize.

Parameters estimation. After t steps, the algorithm constructs estimate $\widehat{p}_t$ as in Eq. (3) using the samples collected so far. Furthermore, it defines the empirical variance of the transition from (s,a) to s' as $\widehat{\sigma}_t^2(s'|s,a) := \widehat{p}_t(s'|s,a)(1 - \widehat{p}_t(s'|s,a))$. Then we have the following high-probability guarantees on $\widehat{p}_t$ and an upper-confidence bound on the transitional noise.

Lemma 1. *Define* $\mathcal{B}_t(s,a,s') := \{\widetilde{p} \in \Delta(S) : |\widetilde{p}(s'|s,a) - \widehat{p}_t(s'|s,a)| \leq B_t(s,a,s')\}$ *and*

$$B_t(s,a,s') := 2\sqrt{\frac{\widehat{\sigma}_t^2(s'|s,a)}{T_t^+(s,a)}\ell_t} + \frac{6\ell_t}{T_t^+(s,a)},$$

where $\ell_t := \log\Big(\frac{6SAT_t^+(s,a)}{\delta}\Big)$. *Furthermore, let*

$$\widehat{V}_t^+(s,a) := \frac{1}{\sqrt{S}} \sum_{s' \in \mathcal{S}} \Big(\sqrt{\widehat{\sigma}_t^2(s'|s,a)} + \sqrt{2\frac{\ell'_t}{T_t^+(s,a)}}\Big),$$

where $\ell'_t = \log\Big(\frac{4S^2A(T_t^+(s,a))^2}{\delta}\Big)$. *Then with probability at least* $1-\delta$, *for any* $t > 0$ *and* $(s,a) \in \mathcal{S} \times \mathcal{A}$, *we have*

$$V(s,a) \leq \widehat{V}_t^+(s,a), \quad p(\cdot|s,a) \in \mathcal{B}_t(s,a).$$

This result allows us to build upper bounds on the error functions and correctly manage the constraint set $\Lambda^{(p)}$.

Smooth optimization. We first notice that both $\mathcal{L}_n^{\mathcal{E}}$ and $\mathcal{L}_n^{\mathcal{W}}$ are built on $\mathcal{G}_n$ which may grow unbounded when n grows to infinity and λ tends to zero in state-action pairs with non-zero transitional noise. Furthermore, the function is smooth with coefficient scaling as $n^{5/2}$, which diverges as $n \to +\infty$. This means that optimizing $\mathcal{L}_n$ functions may become increasingly more difficult for large n. In order to avoid this issue we restrict the space of λ. For any $\eta > 0$ we introduce the restricted simplex

$$\Lambda_\eta^{(p)} := \{\lambda \in \Lambda^{(p)} \mid \forall (s,a) \in \mathcal{S} \times \mathcal{A}, \lambda(s,a) \geq \eta\}.$$

Algorithm 1: FW-ModEst

```
Input: Constraint parameter $\eta > 0$, confidence $\delta \in (0,1)$, budget $n$, objective function $\mathcal{L}_n$ (either $\mathcal{L}_n^{\mathcal{E}}$ or $\overline{\mathcal{L}}_n^{\mathcal{W}}$).
Initialization: Set $t := 1$, state-action counter $T_0(s,a) := 0$ and empirical frequency $\widetilde{\lambda}_0(s,a) := \frac{1}{SA}$.
for episode $k = 1, 2, \dots$ do
    Set $t_k := t$.
    Compute the upper-confidence bound $\widehat{V}_k^+$ to the estimates of the transitional noise.
    Compute the objective function $\mathcal{L}_n$ using $\widehat{V}_k^+$.
    Compute the occupancy measure $\widehat{\phi}_{k+1}^+$ using (6).
    Derive the policy $\widetilde{\pi}_{k+1}^+$ using (7).
    Execute $\widetilde{\pi}_{k+1}^+$ for $\tau_k$ steps.
    Update the time index $t$ and empirical frequency $\widetilde{\lambda}_{k+1}$.
```

On this set we have that $\mathcal{G}_n$ is now bounded and smooth independently from n. Since $\mathcal{L}_n^{\mathcal{E}}$ is just an average of functions $\mathcal{G}_n$, it directly inherits these properties.

Proposition 3. *On the set $\Lambda_\eta^{(p)}$ the function $\mathcal{L}_n^{\mathcal{E}}$ has a smoothness constant scaling as $1/\eta^3$.*

Unfortunately, the same guarantees do not hold for $\mathcal{L}_n^{\mathcal{W}}$, as it is a non-smooth function of $\mathcal{G}_n$. We thus modify $\mathcal{L}_n^{\mathcal{W}}$ using a LogSumExp (LSE) transformation. We briefly recall its properties (see e.g., Beck, 2017).

Proposition 4. *For any $x \in (\mathbb{R}_+)^m$, let* $\mathrm{LSE}(x) := \log\left(\sum_{i=1}^m \exp(x_i)\right)$. *Then the following properties hold:*

1) LSE is convex and 1-smooth w.r.t. the ℓ_∞-norm.

2) $\max_{i\in[m]} x_i \leq \mathrm{LSE}(x) \leq \max_{i\in[m]} x_i + \log(m)$.

While the first property ensures convexity and smoothness, the second statement shows that the bias introduced by the transformation is logarithmic in the number of elements. We define the LSE version of $\mathcal{L}_n^{\mathcal{W}}$ as

$$\overline{\mathcal{L}}_n^{\mathcal{W}}(\lambda) := \log\Big(\sum_{s,a} \exp(\mathcal{G}_n(s,a;\lambda))\Big), \tag{4}$$

which satisfies the following property by the chain rule and Prop. 4.

Proposition 5. *On the set $\Lambda_\eta^{(p)}$ the function $\overline{\mathcal{L}}_n^{\mathcal{W}}$ has a smoothness constant scaling as $1/\eta^5$.*

Comparing Prop. 5 to Prop. 3, we notice that beside being biased, $\overline{\mathcal{L}}_n^{\mathcal{W}}$ is also less smooth than $\mathcal{L}_n^{\mathcal{E}}$ and it is thus possibly harder to optimize.

3.2 The Algorithm

We build on the estimation algorithm of (Tarbouriech and Lazaric, 2019) and propose FW-ModEst (Alg. 1), which proceeds through episodes $k = 1, 2, \dots$ and applies a Frank-Wolfe approach to minimize the model estimation error. The algorithm receives as input the state-action space, the parameter η used in defining the restricted space Λ, the budget n, and the objective function $\mathcal{L}_n$ to optimize (i.e., either $\mathcal{L}_n^{\mathcal{E}}$ or the smoothed $\overline{\mathcal{L}}_n^{\mathcal{W}}$).[2] We first review how $\mathcal{L}_n$ could be optimized by the Frank-Wolfe (FW) algorithm in the exact case, when the transition dynamics p and transitional noise $V(s,a)$ are known. Let λ_0 be the initial solution in $\Lambda_\eta^{(p)}$, then FW proceeds through iterations by computing

$$\phi_{k+1} = \arg\min_{\lambda\in\Lambda^{(p)}} \langle \nabla\mathcal{L}_n(\lambda_k), \lambda\rangle, \tag{5}$$

which is then used to update the candidate solution as $\lambda_{k+1} = \lambda_k + \beta_k(\phi_{k+1} - \lambda_k)$. Whenever the learning rate β_k is properly tuned, the overall process is guaranteed to converge to optimal value $\mathcal{L}_n(\lambda_{\dagger,n})$ at the rate $O(\kappa_\eta/k)$ (Jaggi, 2013), where κ_η is the smoothness of the function (see Prop. 3 and 5). FW-ModEst integrates the Frank-Wolfe scheme into a learning loop, in which the solution constructed by the agent at the beginning of episode k is the empirical frequency of visits $\widetilde{\lambda}_k$, and the optimization in (5) is replaced by an estimated version where the upper bound $V_k^+(s,a)$ from Lem. 1 is used in place of the transitional noise $V(s,a)$. Moreover, the confidence interval $\mathcal{B}_k$ over the transition dynamics is used to characterize the constraint $\Lambda^{(p)}$, thus leading to the optimization[3]

$$\widehat{\phi}_{k+1}^+ = \arg\min_{\widetilde{p}\in\mathcal{B}_k, \lambda\in\Lambda^{(\widetilde{p})}} \langle \nabla\widehat{\mathcal{L}}_{n,k}^+(\widetilde{\lambda}_k), \lambda\rangle, \tag{6}$$

where $\widehat{\mathcal{L}}_{n,k}^+$ is $\mathcal{L}_n$ with V replaced by $\widehat{V}_k^+$ in the formulation of $\mathcal{G}_n$. The optimization in (6) can be seen as the problem of solving an MDP with an optimistic choice of the dynamics within the confidence set $\mathcal{B}_k$ and where the reward is set to $-\nabla\widehat{\mathcal{L}}_{n,k}^+(\widetilde{\lambda}_k)$, i.e., state-action pairs with larger (negative) gradient have larger reward. This directly leads to solutions that try to increase the accuracy in state-action pairs where the current estimation of the model is likely to have a larger error. Notice that the output of the optimization is the distribution $\widehat{\phi}_{k+1}^+$, which cannot be directly used to update the solution λ_k as in FW, since the empirical frequency can only be updated by executing a policy collecting new samples. Thus, $\widehat{\phi}_{k+1}^+$ is used to define a policy π_{k+1}^+ as

$$\widetilde{\pi}_{k+1}^+(a|s) := \frac{\widehat{\phi}_{k+1}^+(s,a)}{\sum_{b\in\mathcal{A}} \widehat{\phi}_{k+1}^+(s,b)}, \tag{7}$$

which is then executed for τ_k steps until the end of the current episode. The samples collected throughout the episode are then used to compute the new frequency $\widetilde{\lambda}_{k+1}$.

[2] In the following we use $\mathcal{L}_n$ for the generic function the algorithm is required to optimize.

[3] The $\arg\min$ extracts only the solution λ.

The extended LP problem. While the overall structure of FW-ModEst is similar to the estimation algorithm in (Tarbouriech and Lazaric, 2019), the crucial difference is that the true dynamics p is unknown and (6) cannot be directly solved as an LP. In fact, while for any fixed $\widetilde{p}$ the set $\Lambda^{(\widetilde{p})}$ only poses linear constraints and optimizing over λ coincides with the standard dual LP to solve MDPs (Puterman, 2014), in our case we have to list all possible values of $\widetilde{p}$ in $\mathcal{B}_k$. We thus propose to rewrite (6) as an *extended* LP problem by considering the state - action - next-state occupancy measure $q(s,a,s')$, defined as $q(s,a,s') := p(s'|s,a)\lambda(s,a)$.[4] We recall from Lem. 1 that at any episode k, we have an estimate $\widehat{p}_k$ such that, w.h.p., for any $(s,a,s') \in \mathcal{S} \times \mathcal{A} \times \mathcal{S}$, $|\widehat{p}_k(s'|s,a) - p(s'|s,a)| \leq B_k(s,a,s')$. For notational convenience, we define $r_k(s,a) := -\nabla\widehat{\mathcal{L}}^{+}_{n,k}(\widetilde{\lambda}_k)(s,a)$. We then introduce the extended LP associated to (6) formulated over the variables q as

$$\max_{q\in\Delta(\mathcal{S}\times\mathcal{A}\times\mathcal{S})} \sum_{s,a,s'} r_k(s,a)q(s,a,s')$$

$$\text{s.t. } \forall j \in \mathcal{S}, \sum_{a,s} q(j,a,s) - \sum_{s,a} q(s,a,j) = 0,$$

$$q(s,a,j) - (\widehat{p}_k(j|s,a) + B_k(s,a,j))\sum_{s'} q(s,a,s') \leq 0,$$

$$-q(s,a,j) + (\widehat{p}_k(j|s,a) - B_k(s,a,j))\sum_{s'} q(s,a,s') \leq 0,$$

$$\sum_{s'} q(s,a,s') \geq \eta.$$

Crucially, the reparametrization in q enables to efficiently solve the problem as a "standard" LP. Let q_{k+1} be the solution of the problem above, then the state-action distribution can be easily recovered as $\widehat{\phi}^{+}_{k+1}(s,a) := \sum_{s'} q_{k+1}(s,a,s')$, and the corresponding policy is computed as in (7).

3.3 Sample Complexity

In order to derive the sample complexity of FW-ModEst, we build on the analysis in (Tarbouriech and Lazaric, 2019) and show that the algorithm controls the regret w.r.t. the optimal solution of the problem $\min \mathcal{L}_n(\lambda)$, i.e.,

$$\mathcal{R}_n := \mathcal{L}_n(\widetilde{\lambda}_n) - \mathcal{L}_n(\lambda_{\dagger,n}), \tag{8}$$

where, similar to the previous section, $\mathcal{L}_n$ can be either $\mathcal{L}^{\mathcal{E}}_n$ or $\overline{\mathcal{L}}^{\mathcal{W}}_n$, $\lambda_{\dagger,n}$ is the corresponding optimal solution and $\widetilde{\lambda}_n$ is the empirical frequency of visits after n steps. We require η and the MDP to satisfy the two following assumptions.[5]

Assumption 1. *The parameter η is such that*

$$\lambda_{\dagger,n} \in \arg\min_{\lambda\in\Lambda^{(p)}_{\eta}} \mathcal{L}_n(\lambda), \quad \lambda_{\star} \in \arg\min_{\lambda\in\Lambda^{(p)}_{\eta}} \mathcal{L}_{\infty}(\lambda),$$

i.e., the optimal solutions in the restricted set Λ coincide with the unrestricted solutions for both $\mathcal{L}_n$ and $\mathcal{L}_{\infty}$.

Assumption 2. *The MDP M is ergodic. We denote by $\gamma_{\min} := \min_{\pi} \gamma^{\pi}_{ps}$ the smallest pseudo-spectral gap (Paulin, 2015) over all stationary policies π and we assume $\gamma_{\min} > 0$.*

We are now ready to derive the model estimation error and sample complexity of FW-ModEst. By extending the analysis from (Tarbouriech and Lazaric, 2019) to unknown transition model and integrating the regret analysis into our estimation error problem, we obtain the following (see App. A for the proof and the explicit dependencies).

Theorem 1. *If* FW-ModEst *is run with a budget n and the length of the episodes is set to $\tau_k = 3k^2 - 3k + 1$, then under Asm. 1 and 2, w.p. $1-\delta$ and depending on the optimization function $\mathcal{L}_n$ given as input, we have that*

$$\mathcal{E}_{\pi,n} = \widetilde{O}\bigg(\frac{\mathcal{L}^{\mathcal{E}}_{\infty}(\lambda^{\mathcal{E}}_{\star})}{\sqrt{n}} + \frac{\Theta^{\mathcal{E}}}{n^{5/6}}\bigg),$$

$$\mathcal{W}_{\pi,n} = \widetilde{O}\bigg(\frac{\mathcal{L}^{\mathcal{W}}_{\infty}(\lambda^{\mathcal{W}}_{\star}) + \log(SA)}{\sqrt{n}} + \frac{\Theta^{\mathcal{W}}}{n^{5/6}}\bigg),$$

where $\Theta^{\mathcal{E}}$ (resp. $\Theta^{\mathcal{W}}$) scales polynomially with MDP constants S, A, and $\gamma^{-1}_{\min}$, and with algorithmic dependent constants $\log(1/\delta)$ *and η^{-1} when optimizing $\mathcal{E}$ (resp. $\mathcal{W}$). From this result we immediately obtain the sample complexity of* FW-ModEst

$$\mathcal{C}^{\mathcal{E}}(\epsilon,\delta) = \Omega\bigg(\frac{\mathcal{L}^{\mathcal{E}}_{\infty}(\lambda^{\mathcal{E}}_{\star})^2}{\epsilon^2} + \frac{\Theta^{\mathcal{E}}}{\epsilon^{6/5}}\bigg),$$

$$\mathcal{C}^{\mathcal{W}}(\epsilon,\delta) = \Omega\bigg(\frac{\big(\mathcal{L}^{\mathcal{W}}_{\infty}(\lambda^{\mathcal{W}}_{\star}) + \log(SA)\big)^2}{\epsilon^2} + \frac{\Theta^{\mathcal{W}}}{\epsilon^{6/5}}\bigg).$$

Model estimation performance. The previous result shows that FW-ModEst successfully targets the model estimation problem with provable guarantees. In particular, the model estimation errors above display a leading error term scaling as $\widetilde{O}(n^{-1/2})$ and a lower-order term scaling as $\widetilde{O}(n^{-5/6})$. Under Asm. 2, any stochastic policy π with non-zero probability to execute any action is guaranteed to reach all states with a non-zero probability. As a result, the associated model estimation error would decrease as $\widetilde{O}(n^{-1/2})$ as well, since $T_{\pi,n}(s,a)$ grows linearly with n in all state-action pairs. Nonetheless, the linear growth could be very small as it depends on the smallest probability of reaching any state by following π. As a result, despite the fact that π may achieve the same rate, FW-ModEst actually performs much better. In particular, a

[4] This construction resembles the extended LP used for loop-free stochastic shortest path problems in (Rosenberg and Mansour, 2019).

[5] Refer to (Tarbouriech and Lazaric, 2019) for more details about the assumptions.

trivial strategy like the uniform policy π_{unif} yields a main-order term decreasing in $\mathcal{L}_\infty(\lambda_{\pi_{\text{unif}}})/n^{1/2}$, whereas the term $\mathcal{L}_\infty(\lambda_\star)/n^{1/2}$ achieved in Thm. 1 is an upper bound to the smallest error that can be achieved for the specific MDP at hand by definition of $\lambda_\star$ (this gap between the uniform policy and FW-ModEst is experimentally displayed in Sect. 5). As such, Thm. 1 shows that FW-ModEst is able to adapt to the current problem and decrease the model estimation error at the best possible rate, up to an additive error (the lower-order term), which is decreasing to zero at the faster rate $\widetilde{O}(n^{-5/6})$.

Limitations. Despite its capability of tracking the performance of the best state-action distribution, FW-ModEst and its analysis suffer from several limitations. Asm. 2 poses significant constraints both on the choice of η and the ergodic nature of the MDP. Moreover, the lower-order terms in Thm. 1 depend inversely on the parameter η (via the optimization properties of $\mathcal{E}$ and $\mathcal{W}$, see App. A ④), which implicitly worsens the dependency on the state-action size, to the extent that the second term may effectively dominate the overall error even for moderately big values of n. This drawback is even stronger in the case of $\mathcal{W}$. In fact, as shown in Prop. 4, the function $\overline{\mathcal{L}}_n^{\mathcal{W}}$ is biased by $\log(SA)$, which is then reflected into the final accuracy of FW-ModEst. While it is possible to change the definition of $\overline{\mathcal{L}}_\infty^{\mathcal{W}}$ to reduce the bias, this would lead to a less smooth function, which would make the constants in the lower-order term even larger. We also notice that as λ appears in the denominator of all functions $\mathcal{L}$, optimizing $\mathcal{L}$ often becomes numerically unstable for large state-action space, where $\min_{s,a} \lambda(s,a) \leq 1/SA$. Finally, the episode length choice in Thm. 1 is often conservative in practice, where shorter episodes usually perform better.

While it remains an open question whether it is possible to achieve better results for the ModEst problem, in the next section we introduce a heuristic objective function that allows more efficient learning with looser assumptions.

4 WEIGHTED MaxEnt

We introduce a heuristic objective function for model estimation and we propose a variant of the algorithm in (Cheung, 2019a) for which we prove regret guarantees.

4.1 Maximum Weighted Entropy

A good exploration strategy to traverse the environment is to maximize the entropy of the empirical state frequency: this consists in the MaxEnt algorithm introduced in (Hazan et al., 2019). Yet this strategy aims at generating a uniform coverage of the state space. While it may be beneficial to perform the MaxEnt strategy in some settings, this is undesirable for the ModEst objective. First, the state-action space should be considered instead of only the state space. Second and crucially, whenever there is a discrepancy in the transitional noise, each state-action pair should not be visited uniformly as often. Indeed, the convex relaxation in Sect. 2 emphasizes that in order to minimize the model estimation error the agent should aim at visiting each state-action pair (s,a) proportionally to its transitional noise $V(s,a)$. A way to do this is to consider a *weighted* entropy objective function, with the weight of each state-action pair depending on its transitional noise.

Definition 3. *For any non-negative weight function $w : \mathcal{S}\times\mathcal{A} \to \mathbb{R}^+$, the weighted entropy H_w is defined on $\Delta(\mathcal{S}\times\mathcal{A})$ as follows:*

$$H_w(\lambda) := -\sum_{(s,a)\in\mathcal{S}\times\mathcal{A}} w(s,a)\lambda(s,a)\log(\lambda(s,a)).$$

To gain insight on H_w, we can argue that $w(s,a)$ represents the value or utility of each outcome (s,a) (see e.g., Guiaşu, 1971). Hence, the learner is encouraged to allocate importance on information regarding (s,a) proportionally to the weight $w(s,a)$. This can be translated in biasing exploration towards region of the state-action space with high weights $w(s,a)$. Similar to (Hazan et al., 2019), we also introduce a smoothed version of H_w and show its properties.

Lemma 2. *For any arbitrary set of non-negative weights w — for which we define $W := \max_{s,a} w(s,a)$ — and for any smoothing parameter $\mu > 0$, we introduce the following smoothed proxy*

$$H_{w,\mu}(\lambda) := \sum_{s,a} w(s,a)\lambda(s,a)\log\left(\frac{1}{\lambda(s,a)+\mu}\right).$$

The function $H_{w,\mu}$ satisfies the following properties:

1. $[\nabla H_{w,\mu}(\lambda)]_{s,a} = -w(s,a)\Big(\log(\lambda(s,a)+\mu) + \frac{\lambda(s,a)}{\lambda(s,a)+\mu}\Big)$.
2. *For any $\mu \leq \frac{1}{e} - \frac{1}{SA}$, we have $[\nabla H_{w,\mu}(\lambda)]_{s,a} \geq 0$.*
3. *$H_{w,\mu}$ is concave in λ, as well as $W\log\Big(\frac{1}{\mu}\Big)$-Lipschitz continuous and $\frac{2W}{\mu}$-smooth.*
4. *We have $|H_w(\lambda) - H_{w,\mu}(\lambda)| \leq \mu SAW$.*

In light of the intuition gained by inspecting the definition of $\mathcal{L}_n$, in the following we set the weights $w(s,a) := V(s,a)/\sqrt{SL}$, with $L := \log(SA/\delta)$, so as to encourage visiting state-action pairs with large transitional noise. We notice that, unlike the smooth versions of $\mathcal{E}$ and $\mathcal{W}$, $H_{w,\mu}$ has a much smaller smoothness constant and its bias can be easily controlled by the choice of μ. Moreover, the state-action distribution λ does not appear in the denominator as in $\mathcal{L}_n$. All these factors suggest that learning how to optimize $H_{w,\mu}$ may be much simpler than directly targeting the model estimation error.

Algorithm 2: WEIGHTED-MAXENT w/ TOC-UCRL2

```
Input: Smoothing parameter $\mu$, confidence $\delta \in (0,1)$.
Initialization: Set $t := 1$, state-action counter
    $T_0(s,a) := 0$ and empirical frequency $\widetilde{\lambda}_0(s,a) := \frac{1}{SA}$.
Compute gradient threshold $Q := 2\log\left(\frac{1}{\mu}\right)$.
for episode $k = 1, 2, \ldots$ do
    Set $t_k := t$.
    Compute the upper-confidence bound $\widehat{V}_k^+$ to the
      estimates of the transitional noise.
    Compute the weighted entropy $\widehat{\mathcal{H}}_k^+$ from (10).
    Compute a near-optimal policy
      $\widehat{\pi}_{k+1}^+ := \mathrm{EVI}(\nabla\widehat{\mathcal{H}}_k^+(\widetilde{\lambda}_k), \mathcal{B}_k, \frac{1}{\sqrt{t_k}})$.
    Initialize $\nu_k(s,a) := 0$, $\Phi := 0$ and $\theta_k^{\mathrm{ref}} := \widetilde{H}_k^+(\widetilde{\lambda}_{t_k})$.
    while $\Phi \le Q$ and $\nu_k(s,a) < T_k(s,a)$ do
        Execute action $a_t = \widehat{\pi}_{k+1}^+(s_t)$, observe the next
          state $s_{t+1}$.
        Compute gradient $\theta_{t+1} := \nabla\widehat{\mathcal{H}}_k^+(\widetilde{\lambda}_t)$.
        Update $\Phi \mathrel{+}= \|\theta_{t+1} - \theta_k^{\mathrm{ref}}\|_2$.
        Update $\nu_k(s_t,a_t) \mathrel{+}= 1$,
          $\widetilde{\lambda}_{t+1} := \frac{t}{t+1}\widetilde{\lambda}_t + \frac{1}{t+1}\mathbb{1}_{s_t,a_t}$, and $t \mathrel{+}= 1$.
    Set $T_{k+1}(s,a) := T_k(s,a) + \nu_k(s,a)$, update $\widetilde{\lambda}_{k+1}$.
```

4.2 Learning to Optimize the Weighted Entropy

We seek to design a learning algorithm maximizing H_w. Since $\mathcal{H}_{w,\mu}$ is concave, Lipschitz continuous and smooth in λ, the algorithm of (Cheung, 2019a) could have been readily applied to maximize it *if* the function was known. The key difference here is that the weights are *unknown*. We thus generalize the algorithm of (Cheung, 2019a) to handle this case. The resulting algorithm is outlined in Alg. 2. In it, $\mathrm{EVI}(r, \mathcal{B}, \varepsilon)$ denotes the standard extended value iteration scheme for reward function r, confidence region $\mathcal{B}$ around the transition probabilities, and accuracy ε (see (Cheung, 2019a) for more details).

Following the terminology of (Cheung, 2019a), in our setting the vectorial outcome at any state-action pair (s,a) is the standard basis vector for (s,a) in $\mathbb{R}^{S\times A}$ (i.e., $\mathbb{1}_{s,a}$). The objective is to minimize the regret

$$\mathcal{R}_n^{Ent} := \min_{\lambda\in\Lambda^{(p)}} \mathcal{H}_w(\lambda) - \mathcal{H}_w(\widetilde{\lambda}_t), \tag{9}$$

where $\Lambda^{(p)}$ is the space of state-action strationary distributions and $\widetilde{\lambda}_t$ is the empirical frequency of visits returned by the algorithm after t steps. Notice that unlike in Sect. 2 and 3, the optimum is computed over λ in the unrestricted set of stationary distribution (i.e., no η lower bound), thus making this regret more general and challenging than (8).

We extend the scope of TOC-UCRL2 (Cheung, 2019a) to handle function $\mathcal{H}_{w,\mu}$. For any time step t, let us define

$$\widehat{\mathcal{H}}_t^+(\lambda) := \sum_{s,a} \frac{\widehat{V}_t^+(s,a)}{\sqrt{SL}} \lambda(s,a) \log\left(\frac{1}{\lambda(s,a)+\mu}\right). \tag{10}$$

Then we can derive a bound similar to Lem. 1.

Lemma 3. *With probability at least $1-\delta$, at any time step t, we have component-wise*

$$\nabla\mathcal{H}(\widetilde{\lambda}_t) \le \nabla\widehat{\mathcal{H}}_t^+(\widetilde{\lambda}_t) \le \nabla\mathcal{H}(\widetilde{\lambda}_t) + u(t,\delta),$$

where the deviation $u(t,\delta)$ satisfies

$$[u(t,\delta)]_{s,a} = O\left(\sqrt{\frac{\log(\frac{SAt}{\delta})}{T_t(s,a)}}\right).$$

We now show how a simple extension of the algorithm of (Cheung, 2019a) can cope with the setting of unknown weights. At the beginning of each episode k, instead of feeding as reward the true, unknown current gradient, we simply feed an upper confidence bound of it, i.e., $\nabla\widehat{\mathcal{H}}_k^+(\widetilde{\lambda}_k)$, where the optimistic objective function $\widehat{\mathcal{H}}_k^+$ — which depends on the smoothing parameter μ — is defined as in (10). Initially, for small values of t, the weights are very similar, thus the algorithm targets a uniform exploration over the state-action space, that is, the original MAXENT objective. As more samples are collected, the estimation of the weights becomes more and more accurate, thus the exploration is gradually skewed towards regions of the state-action space with high transitional noise.

It is possible to extend the analysis of TOC-UCRL2 to this case and obtain a similar regret bound. As in (Cheung, 2019a), the only assumption we need is that the MDP M is communicating, a weaker requirement than Asm. 2.

Assumption 3. *The MDP M is communicating, with diameter $D := \max_{s\neq s'} \min_\pi \mathbb{E}[\tau_\pi(s\to s')] < +\infty$.*

Then we can prove the following result.

Theorem 2. *If* WEIGHTED-MAXENT *is run with a budget n and the smoothing parameter set to $\mu = \frac{1}{n^{1/3}S^{2/3}}$, then under Asm. 3, with overwhelming probability, we have*

$$\mathcal{R}_n^{Ent} = \widetilde{O}\left(\frac{DS^{1/3}}{n^{1/3}} + \frac{D\sqrt{\Gamma SA}}{\sqrt{n}}\right). \tag{11}$$

While they target different objectives, it is interesting to do a qualitative comparison between Thm. 2 and Thm. 1. We first extract from the error bound in Thm. 1 that FW-MODEST has a regret scaling as $\Theta/n^{1/3}$ w.r.t. $\mathcal{L}_\infty^{\mathcal{E}}(\lambda_\star^{\mathcal{E}})$. While the rate is the same as in (11), the difference is that the constants are much better for the weighted MAXENT case. In fact, it scales linearly with the diameter D

and sublinearly with the size of the state-action space, instead of high-order polynomial dependencies on constants such as the constraint parameter η and the MDP parameter $\gamma_{\min}$, which are likely to be very small in any practical application. As a result, we expect the weighted MaxEnt version of TOC-UCRL2 to approach the performance of the optimal weighted entropy solution $\arg\min_{\lambda \in \Lambda^{(p)}} \mathcal{H}_w(\lambda)$ faster than FW-ModEst approaches the performance of $\lambda_\star$.

While we do not have an explicit link between the model estimation error and the weighted entropy, we notice an important connection between $\mathcal{L}_\infty^{\mathcal{E}}$ and H_w in the unconstrained case (i.e., when Λ is reduced to the simplex over the state-action space with no constraint from the MDP). In this case, the solution to $\min_\lambda H_w(\lambda)$ is $\lambda_\star^{\text{Weighted-MaxEnt}}(s,a) \propto \exp(\alpha V(s,a))$, with α a suitable constant, which shows that the optimal distribution has a direct connection with the transitional noise $V(s,a)$. This exactly matches the intuition that a good distribution to minimize $\mathcal{L}_\infty^{\mathcal{E}}$ should spend more time on states with larger noise. This connection is further confirmed in the numerical simulations we report in Sect. 5.

One may wonder why Alg. 2 cannot be applied to target the ModEst problem directly. In fact, Alg. 2 can be applied to any Lipschitz function (i.e., function with bounded gradient) and in the restricted set $\Lambda_\eta^{(p)}$, the function $\mathcal{L}_n$ used in FW-ModEst has indeed a bounded Lipschitz constant. Nonetheless, the regret analysis in Thm. 2 adapted from (Cheung, 2019a) requires evaluating the objective function at *every step* t, which in our case means evaluating $\mathcal{L}_n$ at the empirical frequency $\widetilde{\lambda}_t$. Unfortunately, $\widetilde{\lambda}_t$ is a random quantity and it may not belong to $\Lambda_\eta^{(p)}$ at each step. This justifies the ergodicity assumption and the per-episode analysis of FW-ModEst, where episodes are long enough so that the ergodicity of the MDP guarantees that each state-action pair is visited *enough* and $\mathcal{L}_n$ is evaluated only in the restricted set, where it is *well-behaved*.

5 NUMERICAL SIMULATIONS

We illustrate how the proposed algorithms are able to effectively adapt to the characteristics of the environment. We consider two domains (NoisyRiverSwim and Wheel-of-Fortune) with high level of stochasticity, i.e., $\overline{V} := \sum_{s,a} V(s,a)/(SA)$ is large, and the transitional noise is *heterogeneously spread* across the state-action space, i.e., $\sigma(V) := \sqrt{\frac{1}{SA}\sum_{s,a}\big(V(s,a) - \overline{V}\big)^2}$ is large. Refer to App. C for details and additional experiments.

Optimal solution. We start evaluating the error $\mathcal{E}_{\pi^\star,n}$ associated to the optimal distribution $\lambda_\star$ computed for MaxEnt (Cheung, 2019a), Weighted-MaxEnt (Alg. 2) and FW-ModEst (Alg. 1). For each (s,a) and algorithm $\mathfrak{A}$, we estimate the transition kernel $\widehat{p}^{\mathfrak{A}}_{\pi^\star,n}(s'|s,a)$ by generating $k = n \cdot \lambda_\star^{\mathfrak{A}}(s,a) \vee 1$ samples from $p(\cdot|s,a)$ (see App. C for $\lambda_\star^{\mathfrak{A}}$). We select $n = 2 \cdot 10^6$ to simulate the asymptotic behavior of the algorithms. As shown in Tab. 1, Weighted-MaxEnt and FW-ModEst recover almost identical policies, leading to very similar estimation errors. As expected, MaxEnt is outperformed by the proposed algorithms at the task of model estimation.

	Wheel(5)	NoisyRiverSwim(12)
MaxEnt	$1.0045 \cdot 10^{-3}$	$0.4197 \cdot 10^{-2}$
Weighted-MaxEnt	$0.5091 \cdot 10^{-3}$	$0.2862 \cdot 10^{-2}$
FW-ModEst	$0.5091 \cdot 10^{-3}$	$0.2851 \cdot 10^{-2}$

Table 1: Error $\mathcal{E}_{\pi^\star,n}$ for $n = 2 \cdot 10^6$. We selected $\eta = 0.0001$ for FW-ModEst and $\mu = 0$ for the others.

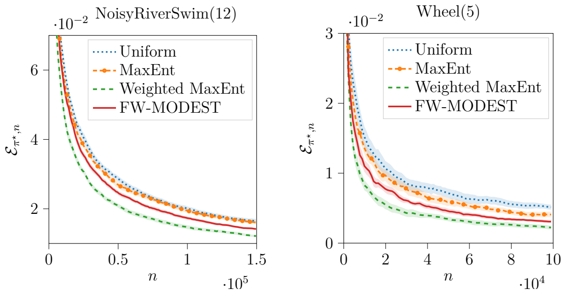

Figure 1: Model estimation error $\mathcal{E}_{\pi,n}$ for NoisyRiverSwim(12) and Wheel(5) (averaged over 20 runs).

Learning. Now that we have shown that Weighted-MaxEnt and FW-ModEst have a similar asymptotic performance, we can focus on the learning process. We first consider NoisyRiverSwim(12). Fig. 1 *(left)* shows that MaxEnt and uniform policy have a similar behavior in this domain. Both approaches are outperformed by FW-ModEst and Weighted-MaxEnt. Despite having the same optimal solution, Weighted-MaxEnt performs better than FW-ModEst. As suggested by the qualitative comparison between Thm. 1 and Thm. 2, the gap is due to the different learning process. We found Weighted-MaxEnt to be more numerically stable and more sample efficient than FW-ModEst. We observe a similar behavior in the Wheel-of-Fortune, see Fig. 1 *(right)*. In all our experiments, Weighted-MaxEnt has outperformed the other algorithms.

6 CONCLUSION

We studied the problem of reward-free exploration for model estimation and designed FW-ModEst, the first algorithm for this problem with sample complexity guarantees. We also introduced a heuristic algorithm (Weighted-MaxEnt) which requires much less restrictive assumptions and achieves better performance than FW-ModEst and MaxEnt in our numerical simulations.

References

Mauricio Araya-López, Olivier Buffet, Vincent Thomas, and François Charpillet. Active learning of MDP models. In *European Workshop on Reinforcement Learning*, pages 42–53. Springer, 2011.

Jean-Yves Audibert, Rémi Munos, and Csaba Szepesvári. Tuning bandit algorithms in stochastic environments. In *International conference on algorithmic learning theory*, pages 150–165. Springer, 2007.

Amir Beck. *First-order methods in optimization*, volume 25. SIAM, 2017.

Quentin Berthet and Vianney Perchet. Fast rates for bandit optimization with upper-confidence Frank-Wolfe. In *Advances in Neural Information Processing Systems*, pages 2225–2234, 2017.

Shalabh Bhatnagar, Richard S Sutton, Mohammad Ghavamzadeh, and Mark Lee. Natural actor–critic algorithms. *Automatica*, 45(11):2471–2482, 2009.

Wang Chi Cheung. Exploration-exploitation trade-off in reinforcement learning on online Markov decision processes with global concave rewards. *arXiv preprint arXiv:1905.06466*, 2019a.

Wang Chi Cheung. Regret minimization for reinforcement learning with vectorial feedback and complex objectives. In *Advances in Neural Information Processing Systems*, pages 724–734, 2019b.

Ronan Fruit, Matteo Pirotta, and Alessandro Lazaric. Improved analysis of UCRL2B, 2019. URL `https://rlgammazero.github.io/docs/ucrl2b_improved.pdf`.

Silviu Guiaşu. Weighted entropy. *Reports on Mathematical Physics*, 2(3):165–179, 1971.

Elad Hazan, Sham Kakade, Karan Singh, and Abby Van Soest. Provably efficient maximum entropy exploration. In *International Conference on Machine Learning*, pages 2681–2691, 2019.

Martin Jaggi. Revisiting Frank-Wolfe: Projection-free sparse convex optimization. In *International Conference on Machine Learning*, pages 427–435, 2013.

C. Jin, A. Krishnamurthy, M. Simchowitz, and T. Yu. Reward-free exploration for reinforcement learning. In *International Conference on Machine Learning*, 2020.

Michael Kearns and Satinder Singh. Near-optimal reinforcement learning in polynomial time. *Machine learning*, 49(2-3):209–232, 2002.

S. Lim and P. Auer. Autonomous exploration for navigating in MDPs. In *Proceedings of the 25th Annual Conference on Learning Theory*, 2012.

Andreas Maurer and Massimiliano Pontil. Empirical Bernstein bounds and sample variance penalization. *arXiv preprint arXiv:0907.3740*, 2009.

Daniel Paulin. Concentration inequalities for Markov chains by Marton couplings and spectral methods. *Electronic Journal of Probability*, 20, 2015.

Martin L Puterman. *Markov Decision Processes.: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, 2014.

Aviv Rosenberg and Yishay Mansour. Online convex optimization in adversarial Markov decision processes. In *International Conference on Machine Learning*, pages 5478–5486, 2019.

Shubhanshu Shekhar, Mohammad Ghavamzadeh, and Tara Javidi. Adaptive sampling for estimating probability distributions. In *International Conference on Machine Learning*, 2020.

Alexander L Strehl and Michael L Littman. An analysis of model-based interval estimation for Markov decision processes. *Journal of Computer and System Sciences*, 74(8):1309–1331, 2008.

Jean Tarbouriech and Alessandro Lazaric. Active exploration in Markov decision processes. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 974–982, 2019.

A Proofs in Sect. 3

Proof of Prop. 2. By direct application of Bernstein's inequality, we have that for any triplet $(s, a, s') \in \mathcal{S} \times \mathcal{A} \times \mathcal{S}$, with probability at least $1 - \delta/3$,

$$|p(s'|s,a) - \widehat{p}_{\pi,n}(s'|s,a)| \\ \leq \sqrt{\frac{2p(s'|s,a)(1-p(s'|s,a))\log(6/\delta)}{T^{+}_{\pi,n}(s,a)}} + \frac{\log(6/\delta)}{3T^{+}_{\pi,n}(s,a)}.$$

As a result, we can now derive a direct upper bound on both error functions. We start with $\mathcal{E}_{\pi,n}$. By a union bound argument, we get with probability at least $1 - \delta/3$, for each state-action pair $(s, a) \in \mathcal{S} \times \mathcal{A}$,

$$\begin{aligned}
\mathcal{E}_{\pi,n} &\overset{\text{(a)}}{\leq} \sqrt{2\log\left(\frac{6SAn}{\delta}\right)}\frac{1}{SA}\sum_{s,a}\frac{V(s,a)\sqrt{S}}{\sqrt{T^{+}_{\pi,n}(s,a)}} \\
&\quad + \log\left(\frac{6SAn}{\delta}\right)\frac{1}{3SA}\sum_{s,a}\frac{S}{T^{+}_{\pi,n}(s,a)} \\
&\overset{\text{(b)}}{\leq} \sqrt{2\log\left(\frac{6SAn}{\delta}\right)}\frac{\sqrt{2}}{SA}\sum_{s,a}\frac{V(s,a)\sqrt{S}}{\sqrt{T_{\pi,n}(s,a)+1}} \\
&\quad + \log\left(\frac{6SAn}{\delta}\right)\frac{2}{3SA}\sum_{s,a}\frac{S}{T_{\pi,n}(s,a)+1} \\
&\overset{\text{(c)}}{\leq} \frac{4}{SA}\log\left(\frac{6SAn}{\delta}\right)\sum_{s,a}\mathcal{F}(s,a;T_{\pi,n}),
\end{aligned}$$

where (a) uses that $T^{+}_{\pi,n}(s,a) \leq n$, (b) stems from the fact that $\max\{x, y\} = \frac{x+y+|x-y|}{2} \geq \frac{x+y}{2}$, and (c) introduces the following function

$$\mathcal{F}(s,a;T) := \frac{V(s,a)}{\sqrt{T(s,a)+1}} + \frac{S}{T(s,a)+1}.$$

In a similar way, we have that

$$\mathcal{W}_{\pi,n} \leq 4\log\left(\frac{6SAn}{\delta}\right)\max_{s,a}\mathcal{F}(s,a;T_{\pi,n}).$$

□

Proof of Lemma 1. From (Maurer and Pontil, 2009), with probability at least $1-\delta$, for any $(s, a, s') \in \mathcal{S} \times \mathcal{A} \times \mathcal{S}$ and any time step $t \geq 1$, we have the following concentration inequality for the estimation of the standard deviation

$$\left|\sqrt{\sigma^2_p(s'|s,a)} - \sqrt{\widehat{\sigma}^2_{\pi,t}(s'|s,a)}\right| \\ \leq \sqrt{2\frac{\log\left(\frac{4S^2A(T^{+}_{\pi,t}(s,a))^2}{\delta}\right)}{T^{+}_{\pi,t}(s,a)}}.$$

Hence summing over the next states $s' \in \mathcal{S}$ gives the first statement of Lem. 1. As for second statement, from Thm. 10 of (Fruit et al., 2019), the fact that the confidence intervals are constructed using the empirical Bernstein inequality (Audibert et al., 2007; Maurer and Pontil, 2009) implies that, with high probability, the true transition model belongs to the confidence set $\mathcal{B}_t$ at any time t. □

Proof of Thm. 1. ① **First**, we extend the regret bound of (Tarbouriech and Lazaric, 2019) to handle the setting of an unknown transition model: for clarity, we adopt the same notation and unravel the analysis. We write $\mathcal{L} := \mathcal{L}_n$. Denote by ρ_{k+1} the approximation error at the end of episode k, i.e., $\rho_{k+1} := \mathcal{L}(\widetilde{\lambda}_{k+1}) - \mathcal{L}(\lambda^{\star})$. Then from Eq. (24) of App. D.2 of (Tarbouriech and Lazaric, 2019),

$$\rho_{k+1} = (1-\beta_k)\rho_k + C_\eta\beta_k^2 + \beta_k\varepsilon_{k+1} + \beta_k\Delta_{k+1},$$

where $\beta_k := \tau_k/(t_{k+1} - 1)$ and

$$\begin{aligned}
\varepsilon_{k+1} &:= \langle\nabla\mathcal{L}_n(\widetilde{\lambda}_k), \widehat{\psi}^{+}_{k+1} - \psi^{\star}_{k+1}\rangle, \\
\Delta_{k+1} &:= \langle\nabla\mathcal{L}_n(\widetilde{\lambda}_k), \widetilde{\psi}_{k+1} - \widehat{\psi}^{+}_{k+1}\rangle.
\end{aligned}$$

The tracking error Δ_{k+1} can be bounded exactly as in (Tarbouriech and Lazaric, 2019). As for the optimization error ε_{k+1}, we first set

$$x_{k+1} := \sum_{s,a}\widehat{\psi}^{+}_{k+1}(s,a)\frac{\alpha(t_k - 1, s, a, \delta)}{\widetilde{\lambda}_k(s,a)^2},$$

where from Lem. 1 we have

$$\alpha(t_k - 1, s, a, \delta) := \sqrt{2S\frac{\log\left(2S^2A\,T^2_{\pi,t}(s,a)\,\delta^{-1}\right)}{T_{\pi,t}(s,a)}}.$$

Denote by $\mathcal{Q}$ the event under which Lem. 1 holds. We get under $\mathcal{Q}$ that

$$\begin{aligned}
\langle\nabla\mathcal{L}(\widetilde{\lambda}_k), \widehat{\psi}^{+}_{k+1}\rangle &\leq \langle\nabla\widehat{\mathcal{L}}^{+}_k(\widetilde{\lambda}_k), \widehat{\psi}^{+}_{k+1}\rangle + 2x_{k+1} \\
&= \min_{\widetilde{p}\in\mathcal{B}_k, \lambda\in\Lambda^{(\widetilde{p})}_\eta}\langle\nabla\widehat{\mathcal{L}}^{+}_k(\widetilde{\lambda}_k), \lambda\rangle + 2x_{k+1} \\
&\overset{\text{(a)}}{\leq} \langle\nabla\widehat{\mathcal{L}}^{+}_k(\widetilde{\lambda}_k), \psi^{\star}_{k+1}\rangle + 2x_{k+1} \\
&\leq \langle\nabla\mathcal{L}_k(\widetilde{\lambda}_k), \psi^{\star}_{k+1}\rangle + 2x_{k+1},
\end{aligned}$$

where (a) uses that under $\mathcal{Q}$ we have $p \in \mathcal{B}_k$ (Lem. 1) and that $\psi^{\star}_{k+1} \in \Lambda^{(p)}_\eta$. Moreover the error x_{k+1} can be bounded exactly as in (Tarbouriech and Lazaric, 2019). This enables to control the optimization error ε_{k+1} in a similar manner. Hence the fact that the transition dynamics are unknown does not affect the final regret bound of Thm. 1 of (Tarbouriech and Lazaric, 2019).

② **Second**, we proceed in deriving a sample complexity result for $\mathcal{E}_{\pi,n}$. We can bound the regret $\mathcal{R}^{\mathcal{E}}_n$, which is defined as follows

$$\mathcal{R}^{\mathcal{E}}_n := \mathcal{L}^{\mathcal{E}}_n(\widetilde{\lambda}_n) - \mathcal{L}^{\mathcal{E}}_n(\lambda^{\mathcal{E}}_{\dagger,n}),$$

where $\lambda^{\mathcal{E}}_{\dagger,n} \in \arg\min_{\lambda\in\Lambda_\eta} \mathcal{L}^{\mathcal{E}}_n$. From Thm. 1 of (Tarbouriech and Lazaric, 2019), if we select episode lengths τ_k scaling as $3k^2-3k+1$, we have that with overwhelming probability

$$\mathcal{R}^{\mathcal{E}}_n = \widetilde{O}\Big(\frac{\Theta^{\mathcal{E}}}{n^{1/3}}\Big), \tag{12}$$

with $\Theta^{\mathcal{E}}$ scaling polynomially with MDP constants S, A, and $\gamma_{\min}^{-1}$, and with algorithmic dependent constants such as $\log(1/\delta)$ and η^{-1} (see ④). Using that $\lambda^{\mathcal{E}}_\star \in \Lambda_\eta$, we have

$$\mathcal{L}^{\mathcal{E}}_n(\lambda^{\mathcal{E}}_{\dagger,n}) \le \mathcal{L}^{\mathcal{E}}_n(\lambda^{\mathcal{E}}_\star) \le \mathcal{L}^{\mathcal{E}}_\infty(\lambda^{\mathcal{E}}_\star) + \frac{S}{\eta\sqrt{n}}. \tag{13}$$

Combining Eq. (12) and (13) yields

$$\mathcal{E}_{\pi,n} \le \frac{4\log\big(\frac{6SAn}{\delta}\big)}{\sqrt{n}}\bigg(\mathcal{L}^{\mathcal{E}}_\infty(\lambda^{\mathcal{E}}_\star) + \frac{S}{\eta\sqrt{n}} + \widetilde{O}\Big(\frac{\Theta^{\mathcal{E}}}{n^{1/3}}\Big)\bigg).$$

Consequently, we get

$$\mathcal{E}_{\pi,n} = \widetilde{O}\bigg(\frac{\mathcal{L}^{\mathcal{E}}_\infty(\lambda^{\mathcal{E}}_\star)}{\sqrt{n}} + \frac{\Theta^{\mathcal{E}}}{n^{5/6}}\bigg).$$

③ **Third**, we derive a sample complexity result for $\mathcal{W}_{\pi,n}$. The function $\overline{\mathcal{L}}^{\mathcal{W}}_n$ that is fed to the learning algorithm as objective function is both convex and smooth in λ on the restricted simplex Λ_η (see Prop. 5). Moreover, at the beginning of each episode k, we can compute an optimistic estimate of the unknown gradient by replacing the unknown $V(s,a)$ with the optimistic quantities $\widehat{V}^+_{t_k}(s,a)$. Let us define $\lambda^{\mathcal{E}}_{\dagger,n} \in \arg\min_{\lambda\in\Lambda_\eta} \overline{\mathcal{L}}^{\mathcal{W}}_n$. The regret is defined as follows

$$\mathcal{R}^{\mathcal{W}}_n := \overline{\mathcal{L}}^{\mathcal{W}}_n(\widetilde{\lambda}_n) - \overline{\mathcal{L}}^{\mathcal{W}}_n(\lambda^{\mathcal{W}}_{\dagger,n}),$$

Then we have that

$$\mathcal{R}^{\mathcal{W}}_n = \widetilde{O}\Big(\frac{\Theta^{\mathcal{W}}}{n^{1/3}}\Big), \tag{14}$$

with $\Theta^{\mathcal{W}}$ scaling polynomially with MDP constants S, A, and $\gamma_{\min}^{-1}$, and with algorithmic dependent constants such as $\log(1/\delta)$ and η^{-1} (see ④). As a result, we have

$$\begin{aligned}
\mathcal{L}^{\mathcal{W}}_n(\widetilde{\lambda}_n) &\le \overline{\mathcal{L}}^{\mathcal{W}}_n(\widetilde{\lambda}_n) \\
&\le \overline{\mathcal{L}}^{\mathcal{W}}_n(\lambda^{\mathcal{W}}_{\dagger,n}) + \mathcal{R}^{\mathcal{W}}_n \\
&\le \overline{\mathcal{L}}^{\mathcal{W}}_n(\lambda^{\mathcal{W}}_\star) + \mathcal{R}^{\mathcal{W}}_n \\
&\le \mathcal{L}^{\mathcal{W}}_n(\lambda^{\mathcal{W}}_\star) + \log(SA) + \mathcal{R}^{\mathcal{W}}_n \\
&\le \mathcal{L}^{\mathcal{W}}_\infty(\lambda^{\mathcal{W}}_\star) + \frac{S}{\eta\sqrt{n}} + \log(SA) + \mathcal{R}^{\mathcal{W}}_n,
\end{aligned}$$

where the chain of inequalities follows from the definitions of $\lambda^{\mathcal{W}}_\star$ and $\lambda^{\mathcal{W}}_{\dagger,n}$ as well as point 2) of Prop. 4. Thus,

$$\mathcal{W}_{\pi,n} = \widetilde{O}\bigg(\frac{\mathcal{L}^{\mathcal{W}}_\infty(\lambda^{\mathcal{W}}_\star) + \log(SA)}{\sqrt{n}} + \frac{\Theta^{\mathcal{W}}}{n^{5/6}}\bigg).$$

④ **Finally**, we provide details on the dependencies of $\Theta^{\mathcal{E}}$ and $\Theta^{\mathcal{W}}$ in Eq. (12) and (14). Retracing the proof of Thm. 1 of (Tarbouriech and Lazaric, 2019) enables to extract that $\Theta^{\mathcal{E}}$ (resp. $\Theta^{\mathcal{W}}$) scales linearly with S, A, $\gamma_{\min}^{-1}$ and $\log(A^S/\delta)$, and polynomially with the algorithmic dependent constant η^{-1} on which the optimization properties of $\mathcal{E}$ (resp. $\mathcal{W}$) depend. Namely, we retrace that $\Theta = O\big(\alpha_\eta SA\gamma_{\min}^{-1}\log(A^S/\delta) + \kappa_\eta\big)$, where α_η denotes the Lipschitz constant (w.r.t. the ℓ_∞-norm) and κ_η denotes the smoothness constant (w.r.t. the ℓ_∞-norm) of the considered function $\mathcal{E}$ or $\mathcal{W}$. Detailing Prop. 3, on the set $\Lambda^{(p)}_\eta$ the function $\mathcal{L}^{\mathcal{E}}_n$ has a Lipschitz constant scaling as $\alpha^{\mathcal{E}}_\eta = (1/\eta^{3/2} + 1/(\eta^2\sqrt{n}))/(SA)$ and a smoothness constant scaling as $\kappa^{\mathcal{E}}_\eta = (1/\eta^{5/2} + 1/(\eta^3\sqrt{n}))/(SA)$. As for $\overline{\mathcal{L}}^{\mathcal{W}}_n$, while an exact computation of its optimization constants is more cumbersome, we can recover a Lipschitz constant scaling at most as $\alpha^{\mathcal{W}}_\eta = 1/\eta^4$ and a smoothness constant scaling at most as $\kappa^{\mathcal{W}}_\eta = 1/\eta^5$ (we excluded the $1/\sqrt{n}$ dependency and considered the worst-case $n=1$ for simplicity). While its smoothness constant remains polynomial in η^{-1}, the function $\overline{\mathcal{L}}^{\mathcal{W}}_n$ is less smooth than $\mathcal{L}^{\mathcal{E}}_n$, as explained in Sect. 3.1 on the difference between Prop. 3 and 5. □

B Proofs in Sect. 4

Proof of Lemma 3. The inequalities stem from a combination of Lem. 2 and Lem. 1. More specifically, the first inequality comes from point 2. of Lem. 2 and from Lem. 1 which states that $V(s,a) \le \widehat{V}^+_t(s,a)$. As for the second inequality, we notice that

$$\begin{aligned}
&\big|[\nabla\widehat{\mathcal{H}}^+_t(\widetilde{\lambda}_t) - \nabla\mathcal{H}(\widetilde{\lambda}_t)]_{s,a}\big| \\
&\quad\le \bigg|\frac{\widehat{V}^+_t(s,a)}{\sqrt{SL}} - \frac{V(s,a)}{\sqrt{SL}}\bigg|\log\Big(\frac{1}{\mu}\Big) \\
&\quad= O\Bigg(\sqrt{\frac{\log(\frac{SAt}{\delta})}{T_t(s,a)}}\Bigg).
\end{aligned}$$

□

Proof of Lemma 2. Properties 1., 3. and 4. stem from Lem. 4.3 of (Hazan et al., 2019). Moreover by choice of μ, we have for any (s,a) that $\mu \le \frac{1}{e} - \lambda(s,a)$, and hence

$$\frac{1}{\log(\lambda(s,a)+\mu)} \ge 1 \ge \frac{\lambda(s,a)}{\lambda(s,a)+\mu},$$

so 2. holds since the weights are non-negative. □

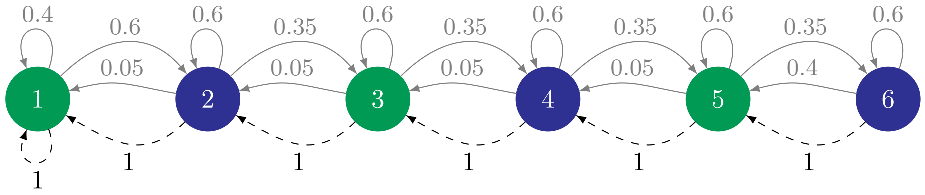

Figure 2: Example of the NoisyRiverSwim environment with $S = 6$ states. The figure reports only RIGHT (solid gray) and LEFT (dashed black) actions. For the states that are green (i.e., odd) we have that $p(s'|s, a_{odd}) = 1/S$ for all $s' \in \mathcal{S}$, and $p(s|s, a_{even}) = 1$. For the blue states (i.e., even) we have that $p(s'|s, a_{even}) = 1/S$ for all $s' \in \mathcal{S}$, and $p(s|s, a_{odd}) = 1$.

Proof of Thm. 2. We unravel the analysis of (Cheung, 2019a), whose Eq. (11) yields

$$\mathcal{R}_n^{Ent} \le \frac{4\log\left(\frac{1}{\mu}\right)\log(n)}{n} + \frac{1}{n}\sum_{t=1}^{n} \nabla\mathcal{H}(\widetilde{\lambda}_t)^\top(\lambda^\star - \mathbb{1}_{s_t,a_t}),$$

since in our setting the vectorial outcome at any state-action pair (s, a) considered in (Cheung, 2019a) corresponds to the standard basis vector for (s, a) in $\mathbb{R}^{S\times A}$ (i.e., $\mathbb{1}_{s,a}$). We can decompose the second term as follows

$$\begin{aligned}
&\sum_{t=1}^{n} \nabla\mathcal{H}(\widetilde{\lambda}_t)^\top(\lambda^\star - \mathbb{1}_{s_t,a_t}) \\
&= \sum_{t=1}^{n} \nabla\widehat{\mathcal{H}}_t^+(\widetilde{\lambda}_t)^\top(\lambda^\star - \mathbb{1}_{s_t,a_t}) \\
&+ \sum_{t=1}^{n} \underbrace{\left(\nabla\mathcal{H}(\widetilde{\lambda}_t) - \nabla\widehat{\mathcal{H}}_t^+(\widetilde{\lambda}_t)\right)^\top(\lambda^\star - \mathbb{1}_{s_t,a_t})}_{:=Z_t}.
\end{aligned}$$

On the one hand, Lem. 3 yields that with high probability, $\nabla\mathcal{H}(\widetilde{\lambda}_t) - \nabla\widehat{\mathcal{H}}_t^+(\widetilde{\lambda}_t) \le 0$ component-wise. On the other hand, $\lambda^\star(s, a) - \mathbb{1}_{s_t,a_t}$ belongs to $[-1, 0]$ only at $(s, a) = (s_t, a_t)$, and is positive otherwise. Hence, we have

$$Z_t \le \nabla\left[\widehat{\mathcal{H}}_t^+(\widetilde{\lambda}_t)\right]_{s_t,a_t} - \left[\nabla\mathcal{H}(\widetilde{\lambda}_t)\right]_{s_t,a_t} \le \left[u(t,\delta)\right]_{s_t,a_t},$$

where the deviation $u(t, \delta)$ is controlled by Lem. 3, which entails that

$$\frac{1}{n}\sum_{t=1}^{n} Z_t = O\left(\sqrt{\frac{\Gamma S A \log(\frac{n}{\delta})}{n}}\right).$$

The term $\frac{1}{n}\sum_{t=1}^{n} \nabla\widehat{\mathcal{H}}_t^+(\widetilde{\lambda}_t)^\top(\lambda^\star - \mathbb{1}_{s_t,a_t})$ can be bounded following the analysis of (Cheung, 2019a), since $\nabla\widehat{\mathcal{H}}_k^+(\widetilde{\lambda}_k)$ corresponds exactly to the rewards fed to the EVI procedure at the beginning of episode k. Moreover, from Lem. 1, we have

$$\max_{s,a,t} \frac{\widehat{V}_t^+(s,a)}{\sqrt{SL}} \le 2.$$

Hence from point 3. of Lem. 2, we get that the Lipschitz-constant of $\widehat{\mathcal{H}}_t^+$ at any time step t is upper bounded by $Q := 2\log\left(\frac{1}{\mu}\right)$. As a result, by setting the gradient threshold as equal to Q in Alg. 2, we obtain the same regret bound as Eq. (6.3) of (Cheung, 2019a). We finally thus get

$$\mathcal{R}_n^{Ent} = \widetilde{O}\left(\frac{DS^{1/3}}{n^{1/3}} + \frac{D\sqrt{\Gamma S A}}{\sqrt{n}}\right).$$

□

C Numerical Simulations

C.1 Description of the environments used

NoisyRiverSwim. NoisyRiverSwim is a noisy and reward-free variant of the standard environment RiverSwim introduced in (Strehl and Littman, 2008) (see Fig. 2). NoisyRiverSwim(S) is a stochastic chain with S states and 4 actions. At each of the S states forming the chain, the two usual actions of RiverSwim are available, plus two additional ones: action a_{even} (resp. action a_{odd}) brings any even (resp. odd) state to any other state of the environment. In odd states, action a_{even} self-loops deterministically, and vice versa. This variant of RiverSwim injects some additional stochasticity in the transition dynamics, over which the agent has some control insofar as the behavior of the "random teleportation" at a given state only takes place if the correct action is executed (that is, a_{even} if the state is even, a_{odd} otherwise).

Wheel-of-Fortune. This environment consists of S states (see Fig. 3), one of which is the center of the wheel (s_0) and the remaining $S - 1$ form the ring of the wheel ($s_1, s_2, \ldots, s_{S-1}$). $A = 5$ actions are available at each state:

- In the center state s_0, $p(s_0|s_0, a) = 1$ for all actions except for one action SPIN $a^\dagger$ for which $\forall n \in \{1, \ldots, S-1\}$, $p(s_n|s_0, a^\dagger) = 1/(S-1)$.

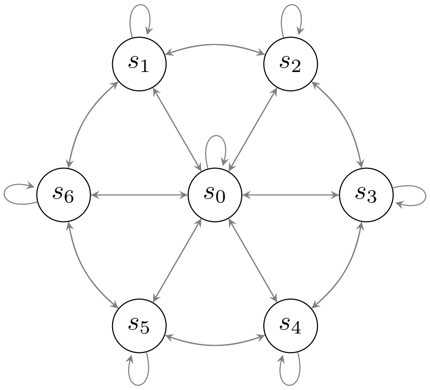

Figure 3: Example of the Wheel-Of-Fortune environment with $S = 7$ states.

- For all remaining states, i.e., $\{s_1, \ldots, s_{S-1}\}$, the actions LEFT, RIGHT, SELF-LOOP, CENTER bring deterministically the agent to the left neighboring state, the right neighboring state, the same state and the center state, respectively. The action NOISY equiprobably yields the outcome of either of the 4 previous actions.

Garnet. Finally we consider standard randomly generated environments (Bhatnagar et al., 2009). A Garnet instance $G(S, A, b)$ has S states and A actions, and for each state-action pair its branching factor $\Gamma(s, a)$ is randomly sampled in the interval $[0, b]$.

C.2 Optimal solution

In Sect. 5, we reported the estimation error $\mathcal{E}_{\pi^\star, n}$ computed using the optimal solution $\lambda_\star^{\mathfrak{A}}$ and $n = 2 \cdot 10^6$ samples. As shown in Tab. 1, WEIGHTED-MAXENT and FW-MODEST have almost identical estimation error. This can be explained by the fact that their optimal solutions are similar. Indeed, consider the Wheel(5) environment, with $\eta = 0.0001$ and $\mu = 0$. The optimal solutions are

$$\lambda_\star^{\text{WEIGHTED-MAXENT}} = \begin{pmatrix} 0. & 0. & 0. & 0. & 0.2 \\ 0. & 0. & 0. & 0. & 0.2 \\ 0. & 0. & 0. & 0. & 0.2 \\ 0. & 0. & 0. & 0. & 0.2 \\ 0. & 0. & 0. & 0. & 0.2 \end{pmatrix},$$

$$\lambda_\star^{\text{FW-MODEST}} = \begin{pmatrix} \eta & \eta & \eta & \eta & 0.1999 \\ \eta & \eta & \eta & \eta & 0.1995 \\ \eta & \eta & \eta & \eta & 0.1995 \\ \eta & \eta & \eta & \eta & 0.1995 \\ \eta & \eta & \eta & \eta & 0.1995 \end{pmatrix},$$

$$\lambda_\star^{\text{MAXENT}} = \begin{pmatrix} 0.043 & 0.043 & 0.043 & 0.043 & 0.1048 \\ 0.043 & 0.043 & 0.043 & 0.0176 & 0.0344 \\ 0.043 & 0.043 & 0.043 & 0.0176 & 0.0344 \\ 0.043 & 0.043 & 0.043 & 0.0176 & 0.0344 \\ 0.043 & 0.043 & 0.043 & 0.0176 & 0.0344 \end{pmatrix}.$$

It is easy to notice that WEIGHTED-MAXENT and FW-MODEST have almost identical solution, the only difference being due to the lower bound η. We observed similar results in the environment NoisyRiverSwim(12).

C.3 Additional experiments

As shown in Fig. 4, when the number of states varies in the Wheel-Of-Fortune environment (respectively 10 and 20 states), WEIGHTED-MAXENT outperforms MAXENT and the uniform policy. Finally, we compare the empirical performance of WEIGHTED-MAXENT, MAXENT and a uniform policy on a set of randomly generated Garnet instances. Fig. 5 reports the average and worst-case model estimation errors $(\mathcal{E}, \mathcal{W})$ for each randomly generated environment after n steps. The experimental protocol is as follows: for a fixed value of S, A and b, we generated 10 random instances of a Garnet $G(S, A, b)$ and for each instance we ran 20 times each algorithm in order to report the mean and standard deviation of the errors. In almost all the experiments, the best performance is achieved by WEIGHTED-MAXENT (see Fig. 5). We observe that, as expected, the gap in performance with MAXENT increases with the stochasticity of the environment, which is captured by the quantity $\sigma(V)$.

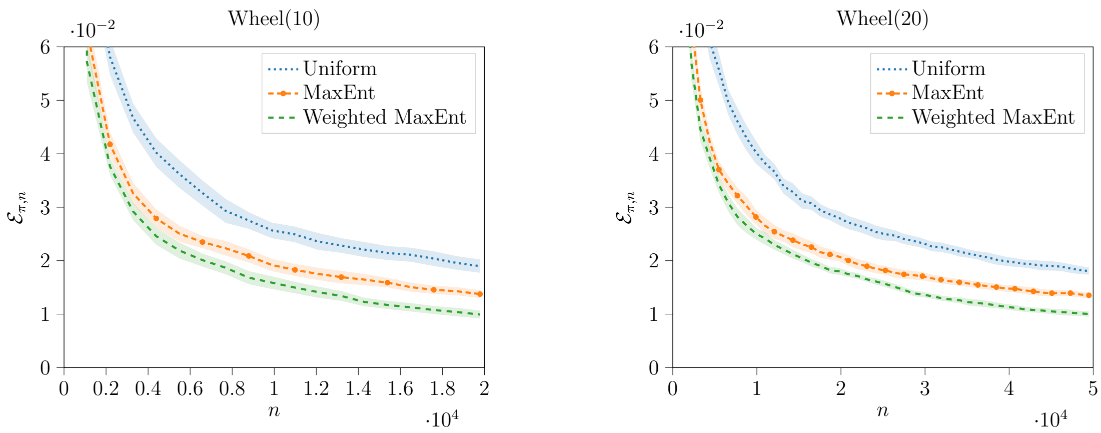

Figure 4: Estimation error $\mathcal{E}_{\pi,n}$ for the environment Wheel(10) and Wheel(20) (averaged over 20 runs).

First table: 10 randomly generated instances of $G(5,5,5)$, with $R = 20$ runs and a budget of $n = 10000$:

G(5,5,5)	$\sigma(V)$	Uniform		MaxEnt		Weighted-MaxEnt	
0	0.2569	0.0358 ± 0.0026	0.1289 ± 0.02	0.0361 ± 0.0022	0.1137 ± 0.0093	0.032 ± 0.0018	0.1023 ± 0.0103
1	0.2657	0.0367 ± 0.0026	0.1348 ± 0.0178	0.0348 ± 0.0015	0.1205 ± 0.0067	0.0287 ± 0.0023	0.1028 ± 0.0105
2	0.2242	0.04 ± 0.0021	0.1209 ± 0.0074	0.0386 ± 0.0021	0.1182 ± 0.0134	0.0388 ± 0.0029	0.1096 ± 0.0096
3	0.2221	0.0384 ± 0.0023	0.1266 ± 0.0175	0.0349 ± 0.0018	0.1087 ± 0.0062	0.0333 ± 0.0015	0.0994 ± 0.0103
4	0.2264	0.0358 ± 0.0022	0.1132 ± 0.0091	0.0347 ± 0.0015	0.1085 ± 0.0103	0.0323 ± 0.0018	0.0967 ± 0.0079
5	0.216	0.0397 ± 0.0023	0.1155 ± 0.0112	0.0389 ± 0.0025	0.1097 ± 0.0094	0.0389 ± 0.0019	0.1101 ± 0.0087
6	0.2555	0.0351 ± 0.0018	0.1272 ± 0.0109	0.0347 ± 0.0019	0.1166 ± 0.0076	0.0307 ± 0.0025	0.0974 ± 0.0096
7	0.2543	0.0299 ± 0.0019	0.1335 ± 0.0147	0.0295 ± 0.0023	0.1207 ± 0.0124	0.0251 ± 0.0018	0.0927 ± 0.0079
8	0.2697	0.0413 ± 0.0023	0.1588 ± 0.0215	0.0367 ± 0.0018	0.141 ± 0.02	0.0312 ± 0.002	0.0997 ± 0.0105
9	0.2534	0.0383 ± 0.0021	0.1142 ± 0.0101	0.0355 ± 0.0019	0.1145 ± 0.0114	0.0341 ± 0.0022	0.0936 ± 0.0076

Second table: 10 randomly generated instances of $G(10,10,5)$, with $R = 20$ runs and a budget of $n = 20000$:

G(10,10,5)	$\sigma(V)$	Uniform		MaxEnt		Weighted-MaxEnt	
0	0.1711	0.0389 ± 0.0018	0.1417 ± 0.0098	0.0367 ± 0.0019	0.1398 ± 0.0135	0.034 ± 0.001	0.1162 ± 0.0083
1	0.1806	0.0323 ± 0.0014	0.1266 ± 0.0082	0.0299 ± 0.0018	0.1227 ± 0.0111	0.0271 ± 0.0012	0.1057 ± 0.0066
2	0.1591	0.0434 ± 0.0013	0.1436 ± 0.0157	0.0438 ± 0.0013	0.1386 ± 0.0087	0.0413 ± 0.0013	0.124 ± 0.0093
3	0.1986	0.0333 ± 0.0015	0.1367 ± 0.0082	0.0331 ± 0.0013	0.1271 ± 0.013	0.0307 ± 0.0015	0.1169 ± 0.0107
4	0.1845	0.0378 ± 0.0017	0.1351 ± 0.0125	0.036 ± 0.0011	0.1209 ± 0.0075	0.0332 ± 0.0011	0.1122 ± 0.0072
5	0.1794	0.0358 ± 0.0016	0.1267 ± 0.0086	0.0352 ± 0.0012	0.1395 ± 0.0096	0.032 ± 0.0014	0.1107 ± 0.0075
6	0.1857	0.0338 ± 0.001	0.1372 ± 0.0142	0.0326 ± 0.0015	0.1211 ± 0.01	0.0309 ± 0.0012	0.1193 ± 0.0101
7	0.1705	0.039 ± 0.0013	0.1316 ± 0.0082	0.0386 ± 0.0014	0.1365 ± 0.0127	0.0383 ± 0.0016	0.1324 ± 0.0105
8	0.1833	0.0318 ± 0.0012	0.1324 ± 0.0111	0.0307 ± 0.001	0.123 ± 0.0116	0.027 ± 0.0014	0.1102 ± 0.0107
9	0.1992	0.0332 ± 0.0015	0.1282 ± 0.0069	0.0321 ± 0.0011	0.1266 ± 0.0098	0.029 ± 0.0014	0.1048 ± 0.0074

Third table: 10 randomly generated instances of $G(20,10,5)$, with $R = 20$ runs and a budget of $n = 40000$:

G(20,10,5)	$\sigma(V)$	Uniform		MaxEnt		Weighted-MaxEnt	
0	0.1168	0.0399 ± 0.0012	0.1903 ± 0.0193	0.0363 ± 0.0012	0.1558 ± 0.0141	0.0352 ± 0.0015	0.143 ± 0.0118
1	0.1403	0.0343 ± 0.0011	0.1878 ± 0.025	0.0311 ± 0.001	0.1388 ± 0.0095	0.0296 ± 0.0009	0.1346 ± 0.0102
2	0.1294	0.0391 ± 0.0009	0.1751 ± 0.0114	0.0351 ± 0.0012	0.1431 ± 0.0084	0.0342 ± 0.001	0.1344 ± 0.0086
3	0.1334	0.0377 ± 0.001	0.1644 ± 0.0118	0.0358 ± 0.0009	0.1293 ± 0.0067	0.0338 ± 0.0012	0.1302 ± 0.0068
4	0.1307	0.0321 ± 0.0009	0.132 ± 0.0118	0.0321 ± 0.0011	0.1445 ± 0.0107	0.0301 ± 0.0012	0.1204 ± 0.0048
5	0.1272	0.0387 ± 0.0011	0.1706 ± 0.0159	0.0373 ± 0.001	0.1599 ± 0.0169	0.0355 ± 0.0013	0.1339 ± 0.0065
6	0.1257	0.037 ± 0.0014	0.1615 ± 0.0188	0.0354 ± 0.0009	0.1388 ± 0.008	0.0341 ± 0.0008	0.1387 ± 0.0077
7	0.1261	0.0363 ± 0.001	0.1533 ± 0.011	0.0361 ± 0.0012	0.1448 ± 0.0111	0.0345 ± 0.001	0.141 ± 0.0114
8	0.1306	0.0348 ± 0.0009	0.1407 ± 0.0079	0.0352 ± 0.0012	0.1496 ± 0.0106	0.034 ± 0.0011	0.137 ± 0.0105
9	0.1275	0.0335 ± 0.0011	0.1514 ± 0.0101	0.0313 ± 0.0009	0.1397 ± 0.0086	0.0299 ± 0.0008	0.1313 ± 0.0098

Figure 5: Experiments on Garnet environments. The first column is the index of the randomly generated Garnet $G(S, A, b)$, the second column is the standard deviation of the transitional noise of the Garnet instance (see Sect. 5 for the definition of $\sigma(V)$). For each algorithm (uniform policy, MaxEnt and Weighted-MaxEnt), the first subcolumn denotes the average estimation error $\mathcal{E}$ after n steps, while the second subcolumn denotes the worst-case estimation error $\mathcal{W}$ after n steps. The value after the ± sign denotes the standard deviation of the errors over the $R = 20$ runs.