MATHEMATICS OF OPERATIONS RESEARCH
Vol. 34, No. 3, August 2009, pp. 737–757
ISSN 0364-765X | EISSN 1526-5471 | 09 | 3403 | 0737

DOI 10.1287/moor.1090.0397

Markov Decision Processes with Arbitrary Reward Processes

Jia Yuan Yu
Department of Electrical and Computer Engineering, McGill University, Montréal, Québec H3A 2A7, Canada,
jia.yu@mcgill.ca

Shie Mannor
Department of Electrical and Computer Engineering, McGill University, Montréal, Québec H3A 2A7, Canada, and
Technion, Technion City, 32000 Haifa, Israel, shie.mannor@mcgill.ca

Nahum Shimkin
Department of Electrical Engineering, Technion, Technion City, 32000 Haifa, Israel, shimkin@ee.technion.ac.il

We consider a learning problem where the decision maker interacts with a standard Markov decision process, with the exception that the reward functions vary arbitrarily over time. We show that, against every possible realization of the reward process, the agent can perform as well—in hindsight—as every stationary policy. This generalizes the classical no-regret result for repeated games. Specifically, we present an efficient online algorithm—in the spirit of reinforcement learning—that ensures that the agent's average performance loss vanishes over time, provided that the environment is oblivious to the agent's actions. Moreover, it is possible to modify the basic algorithm to cope with instances where reward observations are limited to the agent's trajectory. We present further modifications that reduce the computational cost by using function approximation and that track the optimal policy through infrequent changes.

Key words: Markov decision processes; online learning; no-regret algorithms
MSC2000 subject classification: Primary: 90C99; secondary: 93E99
OR/MS subject classification: Primary: Markov processes; secondary: dynamic programming, stochastic games
History: Received August 22, 2007; revised December 2, 2008. Published online in *Articles in Advance* August 6, 2009.

1. Introduction. No-regret algorithms for online decision problems have been a topic of much interest for over five decades, dating back to Hannan's seminal paper (Hannan [16]). A basic version of the online decision problem consists of a finite set of actions A and an infinite sequence of reward vectors r_t: $A \to \mathbb{R}$, $t = 0, 1, 2, \ldots$. A decision maker (or a corresponding online algorithm) chooses an action $a_t \in A$ at each decision instant t after observing the *previous* values of the reward vectors. The average regret after T steps is defined as

$$L_T = \max_{a \in A} \frac{1}{T} \sum_{t=0}^{T-1} r_t(a) - \frac{1}{T} \sum_{t=0}^{T-1} r_t(a_t).$$

Thus, L_T is the average difference between the reward that could be obtained by the best action in hindsight (i.e., given complete knowledge of the reward sequence) and the reward that was actually obtained. A *no-regret* algorithm satisfies $L_T \to 0$ as $T \to \infty$ with probability 1. Such algorithms have also been called *regret minimizing*, *Hannan consistent*, and *universally consistent* (Fudenberg and Levine [15]).

Certain distinctions should be made between different variants of the basic problem. The above-mentioned formulation, where the entire reward vector is observed, is closely connected to the problem of prediction with expert advice (Littlestone and Warmuth [19]). In the adversarial multiarmed *bandit* variant (Auer et al. [1]), only the component $r_t(a_t)$ of the reward vector r_t is observed at each time step. The equivalent *repeated game* formulation assumes a reward vector of the form $r_t(a) = R(a, b_t)$, where b_t is the action chosen by an opponent, R is a known payoff function, and observing the opponent's action b_t is equivalent to observing the reward vector r_t. Another important distinction exists between an oblivious opponent (or environment) and an adaptive one. In the former case, the reward vector sequence is assumed to be fixed in advance but unknown, whereas in the latter it is allowed to depend on previous choices of actions by the algorithm.

A variety of no-regret algorithms have been introduced over the years. These include Hannan's perturbed fictitious play (Hannan [16]), Blackwell's approachability-based scheme (Blackwell [5]), smooth fictitious play (Fudenberg and Levine [15]), calibrated forecasts (Filar and Vrieze [12]), multiplicative weights (Freund and Schapire [13]), and online gradient ascent (Zinkevich [28]). For an overview, see Filar and Vrieze [12], and Cesa-Bianchi and Lugosi [9]. A common theme in the work mentioned above is that the decision maker faces an *identical* decision problem at each stage. This falls short of addressing realistic decision problems that often take place in a dynamic and changing environment. Such an environment is commonly captured by a state variable, which evolves as a controlled Markov chain. The model thus obtained is that of a Markov decision process (MDP) augmented by arbitrarily varying rewards and (possibly) transitions. Furthermore, by modeling

the arbitrary elements as the actions of an opponent (actual or virtual), the model takes the form of a two-person *stochastic game* (Shapley [26]) played between the decision maker and an arbitrary opponent. In this work, we consider MDPs where only rewards change arbitrarily. Such a model arises as a simple extension to a standard online decision problem, as illustrated by the following example.

Example 1.1 (Multiarmed Bandit with Restrictions). Consider the standard adversarial multiarmed bandit problem (Auer et al. [1]), with the additional restriction that switching from one arm to another takes a certain number of time steps. This is easily captured within our MDP model by adding a state variable that recalls the next arm and the remaining time to reach it. This model may similarly account for other restrictions, such as bounds on the number of times a given arm can be chosen in a given interval, restrictions on the allowed transitions between arms, and so forth.

Regret minimization in such dynamic environments has been the topic of only a handful of papers so far. This may seem surprising, given the proliferation of interest in no-regret algorithms, on the one hand, and the extensive literature on MDPs and stochastic games, on the other hand. In Mannor and Shimkin [20], the problem has been considered within the general stochastic game model, where both the transition probabilities and the rewards are affected by the actions of both players, the opponent is adaptive, and the opponent's actions are observed at traversed states only. (Appropriate recurrence assumptions are naturally required, and are assumed in the rest of this discussion without further mention.) A central observation of that paper is that no-regret strategies do not exist for the general model (where regret is defined relative to the best *stationary* policy of the decision maker). An exception is the case where the transition probabilities are controlled by the opponent only, which can be treated by applying a no-regret algorithm at each state separately and independently of other states. For the general model, a relaxed goal was set and was shown to be attainable by using approachability arguments. We note that similar conclusions hold true for the (essentially simpler) model of repeated games with varying stage durations, as reported in Mannor and Shimkin [21]. Merhav et al. [22] have considered sequential decision problems where the loss functions have memory, which correspond to special MDPs, where every state is reachable from every other via a single action. They presented an algorithm using piecewise-constant policies and provided regret-minimizing guarantees similar to ours.

The paper by Even-Dar et al. [11], whose model is closest to the present one, focuses on MDPs with arbitrarily varying rewards. Specifically, it assumes that (1) The state dynamics are known, namely, the state transition probabilities are determined by the decision maker alone; (2) Oblivious opponent: The reward functions, although unknown to the decision maker, are fixed in advance; (3) Observed reward functions: The entire reward function r_t (for every state and action) is observed after each stage t. As mentioned in Even-Dar et al. [11], a simpleminded approach to the problem could start by associating each deterministic stationary policy with a separate expert, and applying existing experts algorithms in that setting. However, because the number of such policies is prohibitive for all but the smallest problems, this approach is computationally infeasible and slow to converge. Thus, more efficient algorithms must be devised. Under the above assumptions, Even-Dar et al. propose an elegant no-regret algorithm, and provide finite-time bounds on the expected regret. The suggested algorithm places an independent experts algorithm at each state; however, the feedback to each algorithm depends on the aggregate policy determined by the action choices of all the individual algorithms and by the value function that is computed for the aggregate policy.

Our work also relates to problems outside the regret-minimizing framework. Optimal control in MDPs with unknown but stationary reward processes can be solved using reinforcement learning, e.g., model-based and Q-learning algorithms (Watkins and Dayan [27]). In contrast to an ordinary stochastic game, the opponent in our model is not necessarily rational or self-optimizing. Our emphasis is providing the agent with policies that perform well against *every* possible opponent. A max-min solution to a zero-sum stochastic game, such as one produced by the R-max algorithm of Brafman and Tennenholtz [8], may well be too conservative when the opponent is not adversarial. It may be in the agent's interest to exploit the nonadversarial characteristic of the opponent. Our model corresponds to a stochastic game where an arbitrary opponent picks the reward functions, but does not affect state transitions.

The basic model that we consider here is similar to Even-Dar et al. [11]. We start by examining the abovementioned assumptions, and show that the oblivious opponent requirement is necessary for the existence of no-regret algorithms. This stands in sharp contrast to the standard (stateless) problem of prediction with expert advice, where no-regret is achievable even against an adaptive opponent. We then propose for this model a new no-regret algorithm in the style of Hannan [16], which we call the Lazy follow-the-perturbed-leader (FPL) algorithm. This algorithm periodically computes a single stationary policy, as the optimal policy against a properly perturbed version of the empirically observed reward functions, and applies the computed policy over a long-enough time interval. We provide a modification to this algorithm (the Q-FPL algorithm) that avoids the exact computation

of optimal policies by incorporating incremental improvement steps in the style of Q-learning (Bertsekas and Tsitsiklis [4]). Next, we extend our results to the model where only on-trajectory rewards are observed; namely, only the rewards along the actually traversed state-action pairs. Clearly, this is a more natural assumption in many cases, and may be viewed as a generalization of the bandits problem to the dynamic setting. Finally, we introduce a variant of our basic algorithm that minimizes regret with respect to nonstationary policies with infrequent changes, in the spirit of Herbster and Warmuth [17].

Our emphasis in this paper is on asymptotic analysis and almost-sure convergence; namely, we show that the long-term average regret vanishes with probability one. Explicit finite-time bounds on the expected regret are provided as intermediate results or as part of the proofs. To summarize, the main contributions of this paper are the following:

- Establishing the necessity of the oblivious opponent assumption in this model.
- A novel no-regret algorithm for MDPs with arbitrarily varying rewards that has diminishing computational effort per time step.
- The first reported no-regret algorithm for the MDP model when only on-trajectory rewards are observed.
- The incorporation of Q-learning style incremental updates that alleviate the computational load and spread out the load over time. Moreover, the Q-learning style updates eliminate the requirement of knowing the state transition probabilities.

The rest of this paper is organized as follows. We describe the model in §2, and motivate our obliviousness and ergodicity assumptions in §3. Section 4 describes and analyzes our main algorithm. The Q-FPL variant and related approximation results are described in §5. The extension to the case of on-trajectory reward observations is described in §6. In §7, we consider regret minimization with respect to a subset of nonstationary policies: the policies with a limited number of changes from one step to another. Section 8 contains concluding remarks.

2. Problem definition. We consider an agent facing a dynamic environment that evolves as a controlled Markov process with an arbitrarily varying reward process. The reward process can be thought of as driven by an abstract *opponent*, which may stand for the collective effect of other agents, or the moves of Nature. The controlled state component is a standard *Markov decision process* (MDP) that is defined by a triple (S, A, P), where S is the finite set of states, A is the finite set of actions available to the agent, and P is the transition probability—that is, $P(s' \mid s, a)$ is the probability that the next state is s' if the current state is s and the action a is taken.

The discrete steps are indexed by $t = 0, 1, \ldots$. We assume throughout the paper that the initial state at step 0 is fixed and denoted s_0. At the tth step, the following happen:

(i) The opponent chooses a reward function r_t: $S \times A \to [0, 1]$;
(ii) The state s_t is revealed;
(iii) The agent chooses an action a_t;
(iv) The entire reward function $r_t = \{r_t(s, a)\}_{(s,a)\in S\times A}$ is revealed; the agent receives reward $r_t(s_t, a_t)$;
(v) The next state $\mathbf{s}_{t+1}$ is determined stochastically according to the transition function P.

REMARK 2.1 (NOTATION). We associate random variables with a bold typeface (e.g., $\mathbf{s}_t$), and their realizations with a normal typeface (e.g., s_t).

In general, the opponent determines a sequence of reward functions $r_0, r_1, \ldots$, where r_t may be picked on the basis of the past state-action history $(s_0, a_0, \ldots, s_{t-1}, a_{t-1})$. In most of the following development, we consider oblivious opponents that pick the reward functions $r_1, r_2, \ldots$ independently of the past state-action history. This assumption is made exact in the following section.

We are interested in policies that respond to the observed sequence of rewards. When choosing action a_t at step t, we assume that the agent knows the current state s_t, as well as the past state-action history and the past reward functions. Hence, we define a *policy* as a mapping from the reward history $(r_0, \ldots, r_{t-1})$ and state-action history $(s_0, a_0, \ldots, s_{t-1}, a_{t-1}, s_t)$ to an action in the simplex $\Delta(A)$.[1] A *stationary* policy is a function μ: $S \to \Delta(A)$ that depends solely on the current state s_t—and not on the history of the rewards or states. We denote by Σ the set of stationary policies. A *deterministic stationary policy* is a mapping μ: $S \to A$ from the current state to an action. We first present in §4 a policy for the agent that assumes that the transition probability function P is known. However, this requirement is not crucial, and we shall dispense with it via simulation-based methods in §5.

Let us consider a sequence of state-action pairs $(\mathbf{s}_t, \mathbf{a}_t)_{t=0,1,\ldots}$ induced by following a stationary policy μ and starting from the initial state s_0. Let $d_t(\mu; s_0)$ denote the probability distribution of $(\mathbf{s}_t, \mathbf{a}_t)$. With respect to the

[1] $\Delta(A)$ denotes the set of all probability vectors over A.

stationary policy μ, if it admits a unique stationary state-action distribution, we denote the latter by $\pi(\mu)$. Given an arbitrary reward function r: $S \times A \to [0, 1]$, we introduce the following inner product notations to denote the expected reward at time step t starting from state s_0 and following policy μ, and the expected reward according to the stationary distribution associated with policy μ:

$$\begin{aligned} \langle r, d_t(\mu; s_0)\rangle &\triangleq \sum_{(s,a)\in S\times A} r(s,a) P_\mu((\mathbf{s}_t, \mathbf{a}_t) = (s,a) \mid s_0), \\ \langle r, \pi(\mu)\rangle &\triangleq \sum_{(s,a)\in S\times A} r(s,a)\pi(\mu)(s,a). \end{aligned} \tag{1}$$

2.1. Assumptions. Our main results require the following assumptions. Their necessity will become clear from the counterexamples of §3. We begin with the following ergodicity assumption.

Assumption 2.1 (Uniform Ergodicity). *The induced Markov chain is uniformly ergodic over the set of stationary policies. This guarantees that there exists a unique stationary distribution $\pi(\mu)$ for each policy μ. Moreover, there exists (cf. Bobkov and Tetali [6]) a uniform mixing time $\gamma \geq 0$; i.e., there exists a finite $\gamma \geq 0$ such that for every stationary policy $\mu \in \Sigma$, every initial state s_0, and $t \geq 0$, we have*

$$\|d_t(\mu; s_0) - \pi(\mu)\|_1 \leq 2e^{1-t/\gamma}.$$

Remark 2.2. The ergodic assumption is quite weak because it only requires that all recurrent states in the Markov chain communicate and that the chain is aperiodic. However, there may exist transient states, which may depend on the stationary policy employed.

The main results of this paper hold when the opponent is oblivious; in other words, the sequence of reward functions does not depend on the state-action history. There are two justifications for this approach. First, from a modeling perspective, the agent may interact with other agents that are truly oblivious, irrational, or have an unspecified or varying objective. This renders their behavior "unpredictable" and seemingly arbitrary. Second, in the presence of many agents, a single agent has little effect on the overall outcome (e.g., price of commodities, traffic in networks) due to the effect of large numbers (Aumann [2]). Moreover, as Example 3.1 shows, the regret cannot be made asymptotically small when the opponent is not oblivious. Formally, we state the obliviousness assumption as follows.

Assumption 2.2 (Oblivious Opponent). *The reward functions $r_0, r_1, \ldots$ are deterministic and fixed in advance.*

Remark 2.3. Alternatively, we may assume that the reward functions $r_0, r_1, \ldots$ are random variables on the null σ-algebra. Hence, every random variable $\mathbf{X}_t$ measurable by the σ-algebra generated by $(\mathbf{s}_0, \mathbf{a}_0, \ldots, \mathbf{s}_t, \mathbf{a}_t)$ satisfies the following:

$$\mathbb{E}[r_t(s,a)\mathbf{X}_t] = r_t(s,a)\mathbb{E}[\mathbf{X}_t] \quad \text{for all } (s,a) \in S \times A. \tag{2}$$

The following results can be shown to apply even when the reward functions are randomly chosen at each step, independently of the state-action history, so that Equation (2) holds. This case can be handled similarly to the deterministic one, at the expense of somewhat more cumbersome notation that we avoid here.

2.2. Regret. In general, the goal of the agent is to maximize its cumulative reward $\sum_{t=0}^{T-1} r_t(\mathbf{s}_t, \mathbf{a}_t)$ over a long time horizon of T steps, where T need not be specified a priori. We shall focus on policies that minimize the *regret*, which measures how worse off the agent is compared to the best stationary policy in retrospect. This regret arises from the lack of prior knowledge on the sequence of reward functions picked by the opponent. We present three related notions of regret that differ in how the sequence of reward functions is retained, and in the choice of initial state. All three definitions of regret for our model collapse to the classical notion of regret for repeated games (cf. Cesa-Bianchi and Lugosi [9]). Our basic definition for regret is the following.

Definition 2.1 (Worst-Case Regret). The *worst-case* average regret, with respect to the realization $r_0, \ldots, r_{T-1}$ of the reward process, is

$$L_T^W \triangleq \sup_{\mu\in\Sigma} \mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T-1} r_t(\tilde{\mathbf{s}}_t, \tilde{\mathbf{a}}_t)\right] - \frac{1}{T}\sum_{t=0}^{T-1} r_t(\mathbf{s}_t, \mathbf{a}_t), \tag{3}$$

where $\mathbb{E}$ denotes expectation over the sequence $(\tilde{\mathbf{s}}_t, \tilde{\mathbf{a}}_t)$ induced by the stationary policy μ. It is implicitly understood that both sequences $\tilde{\mathbf{s}}_t$ and $\mathbf{s}_t$ start at the initial state s_0 and follow the transition kernel P. This regret is a *random* quantity because the trajectory $(\mathbf{s}_t, \mathbf{a}_t)$ is random.

The above definition of regret is one possible extension of the concept of regret introduced in Hannan [16]. However, it is not the only natural definition of regret, and we shall provide two additional notions of regret. An alternative to defining the regret with respect to stationary policies is to take as basis for comparison an agent that possesses only prior knowledge of the *empirical frequency* of reward functions. In this case, it is natural to consider the MDP where the states, actions, and transition probabilities are as before, but where the reward function at every step t is

$$\hat{r}_T(s,a) \triangleq \frac{1}{T}\sum_{j=0}^{T-1} r_j(s,a) \quad \text{for all } (s,a) \in S \times A.$$

With this concept, we present the following definitions.

Definition 2.2 (Steady-State and Empirical-Frequency Regret). The *steady-state* average regret is

$$L_T^S \triangleq \sup_{\mu\in\Sigma} \langle \hat{r}_T, \pi(\mu)\rangle - \frac{1}{T}\sum_{t=0}^{T-1} r_t(\mathbf{s}_t, \mathbf{a}_t). \tag{4}$$

The *empirical-frequency* average regret is

$$L_T^E \triangleq \sup_{\mu\in\Sigma} \mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T-1} \hat{r}_T(\tilde{\mathbf{s}}_t, \tilde{\mathbf{a}}_t)\right] - \frac{1}{T}\sum_{t=0}^{T-1} r_t(\mathbf{s}_t, \mathbf{a}_t). \tag{5}$$

Under Assumptions 2.1 and 2.2, these three definitions are asymptotically equivalent, as established in the following lemma. This result is independent of the agent's learning algorithm. The proofs of this and other lemmata are provided in the appendix.

Lemma 2.1 (Asymptotic Equivalence). *If Assumptions 2.1 and 2.2 hold, then*

$$\left|L_T^E - L_T^S\right| \le 2e\gamma/T$$

and

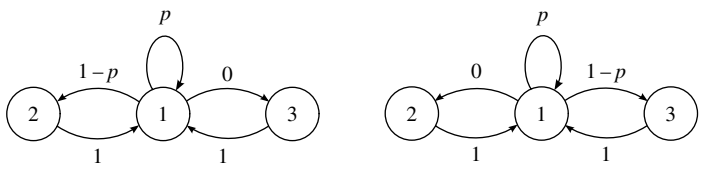
$$\left|L_T^W - L_T^S\right| \le 2e\gamma/T.$$

This equivalence allows us to employ throughout our analysis the simpler notion of steady-state regret (Equation (4)). We say that an agent's policy is a *no-regret policy*, with respect to one of the three definitions of regret, if the corresponding average regret tends to 0 with probability 1 as $T \to \infty$.

3. Counterexamples. In this section, we present examples where vanishing average regret cannot be guaranteed. The first example considers a nonoblivious opponent that modifies the reward function according to the agent's action history. The second example displays a periodic state trajectory.

Example 3.1 (Nonoblivious Opponent). Let the states $S = \{1, 2, 3\}$ be as in Figure 1. The agent has two actions to choose from: whether to go left or right. The corresponding transition probabilities are shown in Figure 1. The nonoblivious opponent assigns zero reward to state 1 at all stages. It gives a reward of 1 to state 2 if the agent took the action leading to state 3 at the previous time step; otherwise, it gives zero reward to state 2. Similarly, the opponent gives a reward of 1 to state 3 if the agent took the action leading to state 2, and a zero reward otherwise. Consequently, for every policy, the reward attained by the agent is $\sum_{t=0}^{T-1} r_t(\mathbf{s}_t, \mathbf{a}_t) = 0$, whereas we have either $\sum_{t=0}^{T-1} \mathbb{E}[r_t(\mathbf{s}_t, \text{left})] \ge 1/2 - p$ or $\sum_{t=0}^{T-1} \mathbb{E}[r_t(\mathbf{s}_t, \text{right})] \ge 1/2 - p$. As a result, the average worst-case regret is always positive and bounded away from 0. Because the MDP is ergodic, a similar argument shows that the same holds true for the two other definitions of regret.

We note that this example is stronger than the counterexample presented in Mannor and Shimkin [20], where the nonvanishing regret is attributed to lack of observation of the reward.



(a) Transition model if the agent chooses to go left.

(b) Transition model if the agent chooses to go right.

Figure 1. State transitions for Example 3.1.

Notes. Taking the left action in state 1 leads to state 2 with probability $1 - p$. There is a small probability p of staying in state 1, regardless of the action taken, thus making the MDP aperiodic. From state 2 or 3, the agent moves to state 1 deterministically.

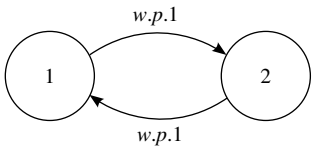


FIGURE 2. State transitions for Example 3.2.

EXAMPLE 3.2 (PERIODIC MDP). Consider an MDP with two states $S = \{1, 2\}$ as in Figure 2. The transition from state 1 to state 2, and vice versa, occurs with probability 1. The agent has a number of identical actions (same transitions and rewards). An oblivious opponent chooses the following rewards:

$$r_t(1) = 1, \qquad r_t(2) = 0, \quad \text{if } t \text{ is even},$$
$$r_t(1) = 0, \qquad r_t(2) = 1, \quad \text{if } t \text{ is odd}.$$

It follows that $\hat{r}_T(1) \to 1/2$ as $T \to \infty$, and similarly for $\hat{r}_T(2)$. If the initial state s_0 is 1, then the agent's cumulative reward is T; otherwise, if s_0 is 2, the cumulative reward is 0. This implies that the regret is either negative if $s_0 = 1$, or positive (and bounded away from zero) if $s_0 = 2$. Therefore, using the empirical-frequency or steady-state notion of regret, zero regret cannot be achieved for periodic MDPs, even if the opponent is oblivious. Nonetheless, in this example, the regret is zero if we adopt the notion of worst-case regret (Equation (3)). In this example, the value of the accumulated reward depends *solely* on the initial state s_0. Because we are interested in characterizing regret with respect to policies, such pathological cases shall be excluded.

In light of these counterexamples, we preclude via Assumptions 2.1 and 2.2 periodic MDPs and nonoblivious opponents.

4. Follow the perturbed leader.

In this section, we present the basic algorithm of this paper and show that it minimizes the regret under full observation of the reward functions.

4.1. Algorithm description.

The proposed algorithm is based on the concept, due to Hannan [16], of following the best action so far, subject to random perturbations that vanish with time. The algorithm works in phases. We partition the time steps $0, 1, \ldots$ into phases (i.e., intervals of consecutive steps[2]), denoted by $\tau_0, \tau_1, \ldots$. We denote by M the number of phases up to step T. The phases are constructed long enough so that the state-action distribution approaches stationarity. As a result, the number of phases M also becomes sublinear in T. The phases are nonetheless short enough so that the agent adapts fast enough to changes in the reward functions. This will be made precise in the results below. As a convention, we let the index t denote a step, whereas m denotes the index of phase τ_m. Moreover, we write $\tau_{0:m}$ to denote the union of phases $\tau_0 \cup \cdots \cup \tau_m$, and $|\tau_{0:m}|$ to denote its length. For ease of notation, we write the cumulative and average reward over one or more phases as

$$R_{\tau_m}(s, a) \triangleq \sum_{t \in \tau_m} r_t(s, a),$$
$$\hat{r}_{\tau_m}(s, a) \triangleq \frac{1}{|\tau_m|} R_{\tau_m}(s, a),$$
$$\hat{r}_{\tau_{0:m}}(s, a) \triangleq \frac{1}{|\tau_{0:m}|} \sum_{t \in \tau_{0:m}} r_t(s, a),$$

for all $(s, a) \in S \times A$. The algorithm takes as input the step index $t \in \tau_m$, the current state s_t, and the average reward function $\hat{r}_{\tau_{0:m-1}}$. It outputs a random action $\mathbf{a}_t$. For the purpose of randomization, the algorithm samples a sequence $\mathbf{n}_1, \mathbf{n}_2, \ldots$ of independent random variables in $\mathbb{R}^{|A|}$. The distribution of these random variables will be specified later.

Algorithm 1 (Lazy FPL)

(i) (*Initialize*). For $t \in \tau_0$, choose the action $\mathbf{a}_t$ according to an arbitrary stationary policy.

[2] The partition is constructed such that the order between steps within each phase is preserved.

(ii) (*Update.*) At the start of phase τ_m, $m = 1, 2, \ldots$, solve the following linear program for (λ_m, h_m):

$$\begin{aligned} \min_{\lambda\in\mathbb{R},\, h\in\mathbb{R}^{|S|}} \quad & \lambda \\ \text{subject to:} \quad & \lambda + h(s) \geq \hat{r}_{\tau_{0:m-1}}(s,a) + \sum_{s'\in S} P(s' \mid s,a)h(s'), \quad (s,a) \in S \times A, \\ & h(s^+) = 0 \quad \text{for some fixed } s^+ \in S. \end{aligned} \tag{6}$$

(iii) (*Follow the perturbed leader*). For $t \in \tau_m$, $m = 1, 2, \ldots$, choose the action

$$\mathbf{a}_t = \arg\max_{a\in A} \left\{ \hat{r}_{\tau_{0:m-1}}(s_t, a) + \mathbf{n}_t(a) + \sum_{s'\in S} P(s' \mid s_t, a) h_m(s') \right\}, \tag{7}$$

where the element of A with the lowest index is taken if the max is not unique.

Observe that the linear program (6) is a standard optimization problem for obtaining the optimal value function (and hence an optimal policy) in an average-reward MDP (Bertsekas [3]). The Lazy FPL algorithm perturbs the average reward function $\hat{r}_{\tau_{0:m-1}}$ with the random variable $\mathbf{n}_t$. Because the perturbing random variables $\mathbf{n}_t$ are identically distributed for all $t \in \tau_m$, whereas the other terms on the right-hand side of Equation (7) are fixed, it follows that the actions $\mathbf{a}_t$ follow the same mixed stationary policy over the phase τ_m. We denote this policy by σ_m. The lazy aspect of this algorithm comes from the fact that it updates its policy only once each phase, similar to other lazy learning schemes (e.g., Merhav et al. [22]).

Introducing randomness through perturbations guarantees that the stationary policies used in consecutive phases do not change too abruptly. This approach is similar to other regret minimization algorithms (e.g., Hannan [16], Kalai and Vempala [18]) and smooth fictitious play (Fudenberg and Kreps [14]). The motivation of increasing phase lengths is twofold. First, using a fixed policy over long phases is computationally efficient. Second, in addition to vanishing *expected* regret, we show that the regret vanishes *almost surely*, provided that the agent does not change its policy too often. One intuition is that, on the one hand, our bases for comparison are the steady-state rewards of stationary policies; on the other hand, taking long phases ensures that the agent's accumulated reward in each phase approaches the steady-state reward of the corresponding policy.

It is important to observe that prior knowledge of the time horizon T is not necessary to run the Lazy FPL algorithm. The only prerequisite is a preestablished scheme to partition every time interval into phases.

4.2. Results. In this section, we show that the Lazy FPL algorithm has the no-regret property. Our main result shows that increasing phase lengths in the Lazy FPL algorithm yields not only an efficient implementation, but also allows us to establish almost-sure convergence for the average regret. The proof relies on a probabilistic bound on the regret, which is derived using a modified version of Azuma's Inequality. The proof of this theorem will come after a number of intermediate results.

THEOREM 4.1 (NO-REGRET PROPERTY OF LAZY FPL). *Suppose that Assumptions 2.1 and 2.2 hold. Let the time horizon $0, 1, \ldots$ be partitioned into phases $\tau_0, \tau_1, \ldots$ such that there exists an $\epsilon \in (0, 1/3)$ for which $|\tau_m| = \lceil m^{1/3-\epsilon} \rceil$ for $m = 0, 1, \ldots$. Further, suppose that the random variables $\mathbf{n}_t(a)$ for $t = 1, 2, \ldots$ and $a \in A$ are independent and uniformly distributed[3] over the support $[-1/\zeta_m, 1/\zeta_m]$, where $\zeta_m \triangleq \sqrt{|\tau_{0:m}|}$ and $t \in \tau_m$. Then, the average regret of the Lazy FPL algorithm vanishes almost surely, i.e.,*

$$\limsup_{T\to\infty} L_T^W \leq 0, \quad \text{w.p. } 1.$$

REMARK 4.1. Theorem 4.1 makes no assumption about the sequence of reward functions $r_0, r_1, \ldots$ except for boundedness and obliviousness.

REMARK 4.2. Observe that the partition of Theorem 4.1 can be constructed incrementally over time without prior knowledge of the time horizon T. Moreover, having a slowly increasing phase length suffices for obtaining convergence.

[3] The random variable $\mathbf{n}_t(a)$ has probability density function

$$f_{\mathbf{n}_t(a)}(z) = \begin{cases} \zeta_m/2, & \text{if } z \in [-1/\zeta_m, 1/\zeta_m], \\ 0, & \text{otherwise.} \end{cases}$$

Theorem 4.1 builds upon the following proposition that establishes the rate of convergence of the expected average regret under the Lazy FPL algorithm.

PROPOSITION 4.1 (EXPECTED REGRET BOUND). *Suppose that the assumptions of Theorem 4.1 hold. In particular, suppose that there exists an* $\epsilon \in (0, 1/3)$ *such that* $|\tau_m| = \lceil m^{1/3-\epsilon} \rceil$ *for* $m = 0, 1, \ldots$ *. Then, the expected average regret of the Lazy FPL algorithm is bounded as follows*:

$$\mathbb{E}[L_T^W] \le \tfrac{4}{3}(2e\gamma + 2|A| + 4e + 1 + 2(|S|+3)|A|^2 \gamma \log(T))T^{-1/4+\epsilon}. \tag{8}$$

REMARK 4.3. The bound of Equation (8) is weaker than the $O(T^{-1/2})$ bound that was obtained for the algorithm of Even-Dar et al. [11]. This can be attributed to the fact that the Lazy FPL algorithm computes a single policy each phase and follows it throughout increasingly long phases. It is a common feature of lazy learning schemes (cf., e.g., Merhav et al. [22]).

The proof of Proposition 4.1 relies on the following lemmata. The proofs of the lemmata are provided in the appendix. The first lemma gives a convenient expression for expected regret.

LEMMA 4.1. *Let* s_0 *be an arbitrary state and* μ *be an arbitrary stationary policy. Let* $(\mathbf{s}_t, \mathbf{a}_t)$ *be the state-action pair at step* t *following policy* μ *and starting at initial state* s_0*. If the opponent is oblivious* (*Assumption 2.2*)*, then for every* $j = 0, \ldots, T-1$*, we have*

$$\mathbb{E}[r_j(\mathbf{s}_t, \mathbf{a}_t)] = \langle r_j, d_t(\mu; s_0)\rangle, \tag{9}$$

where the expectation is taken over both the MDP transitions and the randomization of policy μ*.*

Let $t \in \tau_m$. We define the following unperturbed counterpart to the action $\mathbf{a}_t$ of Equation (7):

$$\mathbf{a}_t^+ = \arg\max_{a \in A} \left\{ \hat{r}_{\tau_{0:m-1}}(s_t, a) + \sum_{s' \in S} P(s' \mid s_t, a) h_m(s') \right\},$$

where h_m is part of the solution to the linear program (6). Note that $\mathbf{a}_t$ is a random variable, whereas $\mathbf{a}_t^+$ is deterministic given the reward sequence. We also define the following stationary policies for all $(s, a) \in S \times A$:

$$\sigma_m(a; s) = \Pr(\mathbf{a}_t = a \mid \mathbf{s}_t = s),$$
$$\sigma_m^+(a; s) = \Pr(\mathbf{a}_t^+ = a \mid \mathbf{s}_t = s).$$

Note that σ_m is a mixed policy, whereas σ_m^+ is a deterministic one. Both are determined by the sequence of reward functions, and hence, independent of the state-trajectory. The following lemma—a consequence of Bertsekas [3, §4.3.3]—asserts the optimality of σ_m^+.

LEMMA 4.2 (OPTIMALITY). *Suppose that Assumption 2.1 holds. In phase* τ_m*, the policy* σ_m^+ *is optimal against the reward function* $\hat{r}_{\tau_{0:m-1}}$ *in the sense that*

$$\langle \hat{r}_{\tau_{0:m-1}}, \pi(\sigma_m^+)\rangle \ge \sup_{\mu \in \Sigma} \langle \hat{r}_{\tau_{0:m-1}}, \pi(\mu)\rangle,$$

where $\pi(\sigma_m^+)$ *is the stationary state-action distribution corresponding to policy* σ_m^+*.*

Next, we bound the rate of change of the empirical average reward function.

LEMMA 4.3 (DIFFERENCE IN PARTIAL AVERAGES). *Let* n *and* l *be nonnegative integers such that* $n \ge l$*. Then,*

$$\left\| \frac{1}{n}\sum_{j=0}^{n-1} r_j - \frac{1}{l}\sum_{j=0}^{l-1} r_j \right\|_\infty \le 2\frac{n-l}{n}.$$

The following lemma quantifies the change in policy of the Lazy FPL algorithm from phase to phase.

LEMMA 4.4 (POLICY CONTINUITY). *Suppose that the assumptions of Theorem 4.1 hold. Then, for* $m = 0, 1, \ldots$*, every* $s \in S$*, and for every positive integer* g*,*

$$\|\sigma_{m+1}(\cdot; s) - \sigma_m(\cdot; s)\|_1 = (|S|+3)|A|^2 \left(\zeta_{m+1} \frac{|\tau_{m+1}|}{|\tau_{0:m+1}|} + \frac{\zeta_{m+1} - \zeta_m}{\zeta_{m+1}} \right)$$

and

$$\|\pi(\sigma_{m+1}) - \pi(\sigma_m)\|_1 = (|S|+3)|A|^2 \left(\zeta_{m+1} \frac{|\tau_{m+1}|}{|\tau_{0:m+1}|} + \frac{\zeta_{m+1} - \zeta_m}{\zeta_{m+1}} \right) g + 4e^{1-g/\gamma}.$$

The following lemma characterizes the effect of randomization in the Lazy FPL algorithm on the expected cumulative reward.

LEMMA 4.5 (EFFECT OF RANDOMIZATION). *Suppose that the assumptions of Theorem 4.1 hold. For phases indexed $m = 1, 2, \ldots$, we have*

$$\langle R_{\tau_{0:m-1}}, \pi(\sigma_m)\rangle \geq \langle R_{\tau_{0:m-1}}, \pi(\sigma_m^+)\rangle - 2|A|\frac{|\tau_{0:m-1}|}{\zeta_m^2}.$$

We now prove Proposition 4.1 and Theorem 4.1.

PROOF OF PROPOSITION 4.1. The proof proceeds along the following lines. The oblivious opponent assumption makes stationary policies as good as any other within long phases. The ergodicity assumption allows us to concentrate on the stationary distributions of the baseline policies, as well as the policies of the sequence of phases. The perturbation noise enforces a certain continuity between policies of consecutive phases, yet it vanishes quickly enough as not to severely affect the optimality of the stationary policy computed at each phase. Letting M denote the number of phases up to time step T, we divide the proof into the following sequence of bounds:

$$\begin{aligned}
&\sum_{t=0}^{T-1}\mathbb{E}[r_t(\mathbf{s}_t,\mathbf{a}_t)]\\
&\quad\geq \sum_{m=0}^{M-1}(\langle R_{\tau_m},\pi(\sigma_m)\rangle - 2e\gamma) \quad \text{(Step 0)} &(10)\\
&\quad\geq \sum_{m=0}^{M-2}(\langle R_{\tau_m},\pi(\sigma_{m+1})\rangle - 2e\gamma - 4e - 2(|S|+3)|A|^2\gamma\log(T)) \quad \text{(Step 1)}\\
&\quad\geq T\cdot\sup_{\mu\in\Sigma}\langle\hat{r}_T,\pi(\mu)\rangle - (M-1)(2e\gamma+4e+2(|S|+3)|A|^2\gamma\log(T)) \quad \text{(Step 2)}\\
&\qquad - 2(M-1)|A| - M^{1/3}, &(11)
\end{aligned}$$

where the expectation $\mathbb{E}$ is over both the MDP transitions and the randomization through $\mathbf{n}_t$ in Algorithm 1. Equation (8) now follows from Equation (11) by Lemma 2.1 and the fact that because $|\tau_m| = \lceil m^{1/3-\epsilon}\rceil$ for $m = 0, \ldots, M-1$, we have $M \leq (4/3)T^{3/4+\epsilon}$.

Step 0. Let s^- denote the state at the beginning of phase τ_m. By Lemma 4.1 and Assumption 2.1, for every phase τ_m, we have

$$\begin{aligned}
\sum_{t\in\tau_m}\mathbb{E}[r_t(\mathbf{s}_t,\mathbf{a}_t)\mid s^-] &= \sum_{t\in\tau_m}\langle r_t, d_t(\sigma_m; s^-)\rangle\\
&\geq \sum_{t\in\tau_m}\langle r_t,\pi(\sigma_m)\rangle - \sum_{t=0}^{|\tau_m|-1}2e^{1-t/\gamma}\\
&\geq \langle R_{\tau_m},\pi(\sigma_m)\rangle - 2e\gamma,
\end{aligned}$$

as in Equation (10).

Step 1. By Lemma 4.4 with $\zeta_m = \sqrt{|\tau_{0:m}|}$ for $m = 0, 1, \ldots$, and by picking $g = \gamma\log(|\tau_{0:m+1}|)$, we obtain

$$\begin{aligned}
\|\pi(\sigma_m) - \pi(\sigma_{m+1})\|_1 &\leq g(|S|+3)|A|^2\left(\zeta_{m+1}\frac{|\tau_{m+1}|}{|\tau_{0:m+1}|} + \frac{\zeta_{m+1}-\zeta_m}{\zeta_{m+1}}\right) + 4e^{1-g/\gamma}\\
&\leq 2(|S|+3)|A|^2\gamma\frac{|\tau_{m+1}|\log(|\tau_{0:m+1}|)}{|\tau_{0:m+1}|^{1/2}} + \frac{4e}{|\tau_{0:m+1}|}.
\end{aligned}$$

It follows that

$$\begin{aligned}
\sum_{m=0}^{M-1}\langle R_{\tau_m},\pi(\sigma_m)\rangle &\geq \sum_{m=0}^{M-2}|\tau_m|\langle\hat{r}_{\tau_m},\pi(\sigma_{m+1})\rangle - |\tau_m|\left(2(|S|+3)|A|^2\gamma\frac{|\tau_{m+1}|\log(|\tau_{0:m+1}|)}{|\tau_{0:m+1}|^{1/2}} + \frac{4e}{|\tau_{0:m+1}|}\right)\\
&\geq \sum_{m=0}^{M-2}(\langle R_{\tau_m},\pi(\sigma_{m+1})\rangle - 2(|S|+3)|A|^2\gamma\log(T) - 4e),
\end{aligned}$$

where the second inequality follows from the construction of the partition. Indeed, choosing $|\tau_m| = \lceil m^{1/3-\epsilon} \rceil$ for $m = 0, \ldots, M-1$ implies that

$$|\tau_m|\,|\tau_{m+1}| \log(|\tau_{0:m+1}|) \le \log(T)\,|\tau_{0:m+1}|^{1/2}.$$

Step 2. In this step, we show that by taking into account rewards for phases $\tau_{m+1}, \ldots, \tau_{M-1}$, we cannot improve the expected reward for phases $\tau_1, \ldots, \tau_{m-1}$. To this end, we show by induction on $J = 0, \ldots, M-2$ that

$$\sum_{m=0}^{M-2} \langle R_{\tau_m}, \pi(\sigma_{m+1}) \rangle \ge \sum_{m=0}^{M-2} \langle R_{\tau_m}, \pi(\sigma_{M-1}) \rangle - 2(M-2)\,|A|. \tag{12}$$

For the base case of $J = 0$, we clearly have

$$\langle R_{\tau_0}, \pi(\sigma_1) \rangle \ge \langle R_{\tau_0}, \pi(\sigma_1) \rangle.$$

Assume that for some J, we have

$$\sum_{m=0}^{J} \langle R_{\tau_m}, \pi(\sigma_{m+1}) \rangle \ge \sum_{m=0}^{J} \langle R_{\tau_m}, \pi(\sigma_{J+1}) \rangle - 2J\,|A|.$$

Then,

$$\begin{aligned}
\sum_{m=0}^{J} \langle R_{\tau_m}, \pi(\sigma_{m+1}) \rangle &\ge \langle R_{\tau_{0:J}}, \pi(\sigma_{J+1}) \rangle - 2J\,|A| \\
&\ge \langle R_{\tau_{0:J}}, \pi(\sigma^+_{J+1}) \rangle - 2\,|A|\,\frac{|\tau_{0:J}|}{\zeta^2_{J+1}} - 2J\,|A| \\
&\ge \langle R_{\tau_{0:J}}, \pi(\sigma_{J+2}) \rangle - 2(J+1)\,|A|,
\end{aligned}$$

where the first inequality follows by definition, the second inequality follows from Lemma 4.5, and the third inequality uses the assumption that $\zeta_m = \sqrt{|\tau_{0:m}|}$ and the optimality of the policy σ^+_{J+1}. Finally, adding $\langle R_{\tau_{J+1}}, \pi(\sigma_{J+2}) \rangle$ to both sides of the above inequalities, we complete the induction step:

$$\sum_{m=0}^{J+1} \langle R_{\tau_m}, \pi(\sigma_{m+1}) \rangle \ge \sum_{m=0}^{J+1} \langle R_{\tau_m}, \pi(\sigma_{J+2}) \rangle - 2(J+1)\,|A|,$$

and Equation (12) follows.

Finally, observe that

$$\sum_{m=0}^{M-2} \langle R_{\tau_m}, \pi(\sigma_M) \rangle - 2(M-2)\,|A| \ge \sum_{m=0}^{M-1} \langle R_{\tau_m}, \pi(\sigma^+_M) \rangle - 2(M-2)\,|A| - 2\,|A| - |\tau_{M-1}| \tag{13}$$

by Lemma 4.5 and the fact that σ^+_M is an optimal policy in an MDP with reward function $\hat{r}_T \triangleq \hat{r}_{\tau_{0:M-1}}$. Equation (13) uses the fact that the reward attained in phase τ_{M-1} is bounded by $|\tau_{M-1}| \le M^{1/3}$. The required result of Equation (11) follows by observing that

$$\sum_{m=0}^{M-1} \langle R_{\tau_m}, \pi(\sigma^+_M) \rangle = T \langle \hat{r}_T, \pi(\sigma^+_M) \rangle = T \cdot \sup_{\mu \in \Sigma} \langle \hat{r}_T, \pi(\mu) \rangle,$$

where the first equality is due to the linearity of the inner product and the definition of $\hat{r}_T$, and the second equality is due to the optimality of σ^+_M against $\hat{r}_T$. □

Proof of Theorem 4.1. The proof relies on a modified version of Azuma's Inequality (Cesa-Bianchi and Lugosi [9, Appendix A.6]). We first define

$$\mathbf{V}_m = \sum_{t \in \tau_m} \mathbb{E}[r_t(\mathbf{s}_t, \mathbf{a}_t)] - r_t(\mathbf{s}_t, \mathbf{a}_t),$$

$$\mathbf{W}_{M-1} = \sum_{m=0}^{M-1} \mathbf{V}_m.$$

By Assumption 2.1, for all m, we have (with probability 1)

$$\mathbb{E}[\mathbf{V}_m \mid \mathbf{s}_t, \mathbf{a}_t \text{ for } t \in \tau_{0:m-1}] = 0.$$

Next, observe that for every real-valued x,

$$\begin{aligned}\mathbb{E}[e^{x\mathbf{W}_{M-1}}] &= \mathbb{E}[e^{x\mathbf{W}_{M-2}}\mathbb{E}[e^{x\mathbf{V}_{M-1}} \mid \mathbf{s}_t, \mathbf{a}_t \text{ for } t \in \tau_{0:M-2}]] \\ &\le \mathbb{E}[e^{x\mathbf{W}_{M-2}}]\exp\left(\frac{x^2}{8}4|\tau_{M-1}|^2\right),\end{aligned}$$

where the inequality follows from Cesa-Bianchi and Lugosi [9, Lemma A.1]. By recursion on M, we obtain

$$\mathbb{E}[e^{x\mathbf{W}_{M-1}}] \le \exp\left(\frac{x^2}{2}\sum_{m=0}^{M-1}|\tau_m|^2\right).$$

By Chebychev's Inequality, for every real x, we obtain

$$\begin{aligned}\Pr\left(\frac{1}{T}\mathbf{W}_{M-1} > \delta\right) &\le \frac{\mathbb{E}[e^{x\mathbf{W}_{M-1}}]}{e^{x\delta T}} \\ &\le \exp\left(-\frac{(\delta T)^2}{2\sum_{m=0}^{M-1}|\tau_m|^2}\right), \qquad (14)\end{aligned}$$

where the second inequality is obtained by choosing x to minimize the exponent. Next, observe that the phase partition $|\tau_m| = \lceil m^{1/3-\epsilon}\rceil$ defined in Proposition 4.1 implies that $M \le (4/3)T^{3/4+\epsilon}$ for every $\epsilon > 0$. Hence, we have $\sum_{m=0}^{M-1}|\tau_m|^2 \le (3/5)M^{5/3} \le (4/5)T^{5/4+5\epsilon/3}$. Following substitutions, we obtain

$$\begin{aligned}\Pr\left(\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}[r_t(\mathbf{s}_t,\mathbf{a}_t)] - \frac{1}{T}\sum_{t=0}^{T-1}r_t(\mathbf{s}_t,\mathbf{a}_t) > \delta\right) &\le \exp\left(-\frac{\delta^2T^2}{(8/5)T^{5/4+5\epsilon/3}}\right) \\ &= \exp(-(5/8)\delta^2T^{3/4-5\epsilon/3}).\end{aligned}$$

Therefore, by picking ϵ small enough, the right-hand side of Equation (14) is summable over nonnegative integers T for every $\delta > 0$. An application of Proposition 4.1 and the Borel-Cantelli Lemma completes the proof. □

5. Approximate algorithms. In many cases of interest, computing the exact policy σ_m at each phase τ_m of the Lazy FPL algorithm might be intractable due to the size of the state space. One solution is to compute an approximation ρ_m to σ_m. The policy ρ_m is still computed once every phase, but by using a computationally efficient method. We consider the approach of approximating the optimal state-action value function or *Q-function*. Recall that in average-reward MDPs, the Q-function $Q\colon S \times A \to \mathbb{R}$ represents the relative utility of choosing a particular action at a particular state. Let (λ_m, h_m) denote the optimal solution to the linear program (6) at the start of phase τ_m. The corresponding optimal Q-function is therefore defined as

$$Q_m^*(s,a) = \hat{r}_{\tau_{0:m-1}}(s,a) + \sum_{s'\in S}P(s' \mid s,a)h_m(s').$$

Definition 5.1. Let ϵ and δ be nonnegative constants. Consider an algorithm that computes an approximate Q-function Q_m for each phase τ_m and chooses an action

$$\mathbf{a}_t = \arg\max_{a\in A}\{Q_m(s_t,a) + \mathbf{n}_t(a)\}$$

at every step t in phase τ_m, with the random variable $\mathbf{n}_t$ distributed as in Theorem 4.1. Such an algorithm is an (ϵ,δ)-*approximation* algorithm if there exists an integer N such that, for $m \ge N$,

$$\Pr\left(\|Q_m(s,\cdot) - Q_m^*(s,\cdot)\|_1 \le \epsilon \text{ for every } s \in S\right) \ge 1-\delta, \qquad (15)$$

where Q_m^* is the optimal Q-function.

The following corollary (proved in the appendix) relaxes the need for an exact optimization procedure.

COROLLARY 5.1. *Let* P_σ *denote the matrix whose* (s', s)*-element is* $P(s' \mid s, \sigma(s))$*, i.e., the transition matrix induced by the stationary policy* $\sigma\colon S \to A$*. Let* Z_σ *denote the fundamental matrix (cf. Schweitzer* [25]*) associated with the same transition kernel* $P(s' \mid s, \sigma(s))$*. In other words,*

$$Z_\sigma \triangleq [I - \mathrm{P}_\sigma + \mathrm{P}_\sigma^\infty]^{-1}, \quad \textit{where } \mathrm{P}_\sigma^\infty \triangleq \lim_{K\to\infty} \frac{1}{K} \sum_{k=1}^{K} \mathrm{P}_\sigma^k.$$

Further, let the norm $\|M\|_\infty$ *of a matrix* M *denote its maximum absolute row-sum. Suppose that the assumptions of Theorem 4.1 hold. The average regret of an* (ϵ, δ)*-approximation algorithm is bounded as follows*:

$$\limsup_{T\to\infty} L_T^W \le \sup_{\sigma\in\Sigma} \|Z_\sigma\|_\infty (\epsilon + \delta), \quad \textit{w.p. } 1.$$

REMARK 5.1. If an algorithm is an (ϵ, δ)-approximation for every pair of positive numbers ϵ and δ, then the average regret tends to zero almost surely. It is also possible to obtain almost-sure convergence of the average regret to zero if the Q-functions Q_m computed by an approximation algorithm improve in accuracy from phase to phase, such that Equation (15) holds for sequences ϵ_m and δ_m that decrease quickly enough to zero.

In the following algorithm, we use Q-learning (Bertsekas and Tsitsiklis [4, Chapter 7]) to compute an approximation ρ_m to the policy σ_m of the Lazy FPL algorithm. In essence, Q-learning is employed as a method of solving the linear program of the Lazy FPL algorithm. It is well known that Q-learning is an iterative simulation-based method that does not need to keep track of the transition probabilities. Let Q_t denote the sequence of Q-functions, and $Q_{\tau_{0:m-1}}$ denote the Q-function obtained at the last step of phase τ_{m-1}. During every step t of phase τ_m, we choose our action to maximize the Q-function $Q_{\tau_{0:m-1}}$ obtained over the previous phases, perturbed by a random term $\mathbf{n}_t$; simultaneously, we update the sequence of Q-functions Q_t at every step.

Algorithm 2 (Q-FPL)

(i) (*Initialize.*) For $t \in \tau_0$, set $Q_t = 0$ and choose action $\mathbf{a}_t$ according to an arbitrary deterministic policy $\mu\colon S \to A$.

(ii) (*Update.*) At every step $t \in \tau_m$, for $m = 1, 2, \ldots$, set $\kappa_m = 1/\sqrt{m}$ and update Q_t iteratively as follows:

$$Q_t(s_{t-1}, a_{t-1}) = (1 - \kappa_m) Q_{t-1}(s_{t-1}, a_{t-1}) + \kappa_m (\hat{r}_{\tau_{0:m-1}}(s_{t-1}, a_{t-1}) + \max_{a\in A} Q_{t-1}(s_t, a) - Q_{t-1}(s', a')), \tag{16}$$

where s' and a' are fixed, and the term $Q_{t-1}(s', a')$ serves the purpose of normalization.

(iii) (*Perturb.*) At every step $t \in \tau_m$, for $m = 1, 2, \ldots$, choose action

$$\mathbf{a}_t = \arg\max_{a\in A} \{Q_{\tau_{0:m-1}}(s_t, a) + \mathbf{n}_t(a)\},$$

where the random variables $\mathbf{n}_t$ are distributed as in Theorem 4.1.

REMARK 5.2. As for the Lazy FPL algorithm, the reward function $\hat{r}_{\tau_{0:m-1}}$ is fixed throughout phase τ_m.

The sequence κ_m is selected such that it satisfies the conditions for stochastic approximation (cf. §4.3 of Borkar and Meyn [7]). Let $Q^*_{\tau_{0:m-1}}$ denote the optimal Q-function against the fixed reward function $\hat{r}_{\tau_{0:m-1}}$. By Borkar and Meyn [7, Theorem 2.4], within each phase where the reward function is fixed and the length is long enough, for every $\beta > 0$ and $\gamma > 0$, we have

$$\Pr(\|Q_{\tau_{0:m-1}} - Q^*_{\tau_{0:m-1}}\|_1 > \beta) < \gamma, \tag{17}$$

so that Equation (15) holds.[4] We observe that the Q-FPL algorithm is in fact an (ϵ, δ)-approximation algorithm for every positive ϵ and δ, which leads to the following corollary by an argument similar to Theorem 4.1.

COROLLARY 5.2. *Suppose that the assumptions of Theorem 4.1 hold. Then, the average regret of the Q-FPL algorithm tends to zero almost surely.*

Other algorithms, especially some actor-critic algorithms that are equivalent to Q-learning (Crites and Barto [10]), may be used as well, as long as they are (ϵ, δ)-approximations for every pair of positive ϵ and δ.

REMARK 5.3 (COMPUTATIONAL LOAD). The Q-FPL algorithm has a fixed computational load per step. This complexity is less demanding than that of Even-Dar et al. [11], although the latter is also fixed per step. In

[4] To be accurate, for the off-policy Q-function evaluation in Step 2 of the Q-FPL algorithm to converge at the end of each phase, we must ensure that the policy induced by Step 3 performs sufficient exploration. Hence, we sample an independent perturbation $\mathbf{n}_t$ at every time step.

comparison, the Lazy FPL algorithm requires solving an MDP at the beginning of every phase, but it has the advantage of diminishing the per-step complexity.

6. Observing rewards only on trajectory. In this section, we present a modification of the Lazy FPL algorithm in the spirit of Auer et al. [1] to deal with instances where the reward functions are partially observed. More precisely, we consider the case where the agent only observes the value of the reward function sequence on the traversed state-action trajectory. Consequently, we restrict the space of the agent's policies to those that map the observed reward-history $r_0(s_0, a_0), \ldots, r_{t-1}(s_{t-1}, a_{t-1})$ and the current state s_t to a mixed action.

Our approach is to construct an unbiased estimate of $\hat{r}_{\tau_{0:m-1}}$ at each phase τ_m. Following an initialization phase τ_0, we construct a *random* reward function at every step t. The length of the phase τ_0 and the policy adopted therein are such that, for $t \geq |\tau_0|$, $\Pr((\mathbf{s}_t, \mathbf{a}_t) = (s, a) \mid s_0) > 0$ for all $(s, a) \in S \times A$. For all $t \geq |\tau_0|$ and $(s, a) \in S \times A$, we let

$$\mathbf{z}_t(s, a) = \begin{cases} \dfrac{r_t(s, a)}{\Pr((\mathbf{s}_t, \mathbf{a}_t) = (s, a) \mid s_0)}, & \text{if } (\mathbf{s}_t, \mathbf{a}_t) = (s, a), \\ 0, & \text{otherwise.} \end{cases}$$

Observe that only the value of r_t at the traversed state-action pair $(\mathbf{s}_t, \mathbf{a}_t)$ is required to evaluate $\mathbf{z}_t$. The probability $\Pr((\mathbf{s}_t, \mathbf{a}_t) = (s, a) \mid s_0)$ is readily computed recursively using the transition probabilities associated with the policy followed at step $t - 1$. From the sequence $\mathbf{z}_j$, we construct $\hat{\mathbf{z}}_t \triangleq (1/t) \sum_{j=0}^{t-1} \mathbf{z}_j$ as an estimate of $\hat{r}_t = (1/t) \sum_{j=0}^{t-1} r_j$. In conformance with our notation, $\hat{\mathbf{z}}_{\tau_{0:m-1}}$ denotes $\hat{\mathbf{z}}_t$, where t is the first step of phase τ_m.

Algorithm 3 (Exploratory FPL)

(i) (*Initialize*). Let the length of phase τ_0 be long enough so that $\Pr((\mathbf{s}_t, \mathbf{a}_t) = (s, a) \mid s_0) > 0$ for $t \geq |\tau_0|$ and $(s, a) \in S \times A$. For $t \in \tau_0$, choose action $\mathbf{a}_t$ uniformly at random over A.

(ii) (*Estimate*). At every step $t = 1, 2, \ldots$, compute the estimate $\hat{\mathbf{z}}_t$ recursively.

(iii) (*Sample*). At the start of phase τ_m, for $m = 1, 2, \ldots$, sample an independent Bernoulli random variable $\mathbf{x}_m$ that takes value 1 with probability ϕ_m.

(iv) (*Lazy FPL*). If $\mathbf{x}_m = 0$, by substituting $\hat{\mathbf{z}}_{\tau_{0:m-1}}$ for $\hat{r}_{\tau_{0:m-1}}$, solve the linear program (6) and follow the policy of Equation (7) throughout phase τ_m.

(v) (*Explore*). If $\mathbf{x}_m = 1$, for $t \in \tau_m$ and $m = 1, 2, \ldots$, choose action $\mathbf{a}_t$ uniformly at random over A.

The following corollary (see the appendix for a proof outline) asserts a result analogous to Theorem 4.1 for the Exploratory FPL algorithm (Algorithm 3).

Corollary 6.1 (No-Regret Property of Exploratory FPL). *Suppose that the assumptions of Theorem 4.1 hold. Let M denote the number of phases up to time step T. Suppose that the agent follows the Exploratory FPL algorithm with a sequence $\phi_m > 0$, for $m = 0, \ldots, M-1$, ensuring infinitely many exploration phases, and such that*

$$\sum_{m=0}^{M-1} |\tau_m| \, \phi_m = O(M). \tag{18}$$

Then, the average regret of the Exploratory FPL algorithm vanishes almost surely.

Remark 6.1. If ϕ_m is set to a positive constant, then the Exploratory FPL algorithm reduces to an approximation algorithm governed by Corollary 5.1.

Remark 6.2. Corollary 6.1 guarantees that the Exploratory FPL algorithm minimizes regret in generalized multiarm bandit problems with a state variable.

7. Regret with respect to dynamic policies. In this section, we consider a more general notion of regret that encompasses some dynamic policies. Consider a sequence of policies $\vec{\mu} = (\mu_0, \ldots, \mu_{T-1})$, where every element μ_j of the sequence is a deterministic policy $\mu_j \colon S \to A$. Let the number of switches in this sequence of policies be

$$K(\vec{\mu}) = \sum_{j=1}^{T-1} 1_{[\mu_{j-1} \neq \mu_j]}.$$

Let K_0 be a fixed integer. A more challenging baseline of comparison for the cumulative reward is

$$\mathscr{B}_T(K_0) \triangleq \sup_{\substack{(\mu_0, \ldots, \mu_{T-1}): \\ K(\vec{\mu}) \leq K_0}} \mathbb{E}\left[\sum_{t=0}^{T-1} r_t(\tilde{\mathbf{s}}_t, \mu_t(\tilde{\mathbf{s}}_t))\right], \tag{19}$$

where $(\tilde{\mathbf{s}}_0, \mu_0(\tilde{\mathbf{s}}_0)), \ldots, (\tilde{\mathbf{s}}_{T-1}, \mu_{T-1}(\tilde{\mathbf{s}}_{T-1}))$ denote state-action pairs induced by the sequence of policies $\mu_0, \ldots, \mu_{T-1}$, and the maximum is taken over all possible sequences of policies with at most K_0 switches. If $K_0 = 0$, then Equation (19) reduces to the baseline considered so far (cf. Equation (3)). We present an algorithm that guarantees a reward consistent with the above baseline. This algorithm adapts the fixed-share algorithm of Herbster and Warmuth [17] to the MDP framework.

Algorithm 4 (Tracking FPL)

(i) (*Initialize*). Fix $\alpha \in [0, 1]$. For $t \in \tau_0$, choose action $\mathbf{a}_t$ according to an arbitrary deterministic policy $\mu\colon S \to A$.

(ii) (*Sample*). At the outset of phase τ_m, for $m = 1, 2, \ldots$, sample a Bernoulli random variable $\mathbf{x}_m$ with $\Pr(\mathbf{x}_m = 1) = \alpha$.

(iii) (*Fixed-share*). If $\mathbf{x}_m = 0$, sample a policy $\mathbf{y}_m$ uniformly at random from the set of deterministic policies $\{\mu\colon S \to A\}$, then follow the policy $\mathbf{y}_m$ throughout phase τ_m.

(iv) (*Lazy FPL*). If $\mathbf{x}_m = 1$, solve the linear program (6) and follow the policy of Equation (7) throughout phase τ_m.

Remark 7.1. Observe that, as before, the algorithm elects a single policy in each phase and follows it throughout. The fixed-share scheme occurs *once in each phase*—at the outset. Observe also that the uniformly random policy $\mathbf{y}_m$ can be constructed efficiently. As in the fixed-share algorithm of Herbster and Warmuth [17], the action at each step is equal to the previous action with probability $1 - \alpha + \alpha/|A|$, and equal to each different action with probability $\alpha/|A|$.

Remark 7.2. In the MDP setting, the most obvious extension of the fixed-share algorithm of Herbster and Warmuth [17] is to associate an expert to every deterministic policy $\mu\colon S \to A$. This creates an exponential number of such experts, which our approach avoids.

The following analog of Theorem 4.1 guarantees that the regret with respect to the reward achieved by the best sequence of policies with a finite number of switches vanishes asymptotically if the agent employs the Tracking FPL algorithm.

Theorem 7.1 (No-Regret Property of Tracking FPL). *Suppose that the assumptions of Theorem 4.1 hold. Let K_0 be a positive integer. Suppose further that the agent follows the Tracking FPL algorithm with the parameter $\alpha = K_0/(\lceil T/\lceil T^{1/3}\rceil\rceil - 1)$. Then, the average regret with respect to the baseline of Equation (19) vanishes almost surely, i.e.,*

$$\limsup_{T\to\infty}\left\{\frac{1}{T}\mathscr{B}_T(K_0) - \frac{1}{T}\sum_{t=0}^{T-1} r_t(\mathbf{s}_t, \mathbf{a}_t)\right\} \le 0, \quad \text{w.p.1.}$$

Remark 7.3. Although we only consider the case of a fixed number of switches K_0 and a fixed parameter α, it can be shown, by using the doubling trick of Cesa-Bianchi and Lugosi [9, §3.2], that the result of Theorem 7.1 holds as long as the number of switches K_0 increases slowly enough in T.

The proof of this theorem hinges on a bound on the rate of convergence of the expected regret similar to Proposition 4.1. To derive this bound, we first prove a bound for a different hypothetical—and less practical—algorithm. Consider Algorithm 5: a modified version of the exponentially weighted average forecaster (Cesa-Bianchi and Lugosi [9]), which also resembles the algorithm of Even-Dar et al. [11]. To every deterministic policy $\mu : S \to A$, we associate a weight $w_m(\mu)$ that is updated at every phase τ_m for $m = 0, 1, \ldots$. Once at the start of every phase, the algorithm picks a deterministic policy with probability proportional to its weight, and follows this policy throughout the phase. The weights are adjusted in the spirit of the Fixed-share algorithm (Herbster and Warmuth [17]) to track infrequent changes in optimal policy.

Algorithm 5 (Lazy Tracking Forecaster)

(i) (*Initialize.*) Fix $\alpha \in [0, 1]$ and $\eta \in (0, \infty)$. For every deterministic policy $\mu\colon S \to A$, set

$$w_0(\mu) = \frac{1}{|A|^{|S|}}.$$

(ii) (*Update weights and choose policy*). At the start of every phase τ_m, for $m = 1, 2, \ldots$, evaluate

$$w_m(\mu) = w_{m-1}(\mu)\exp(\eta\langle R_{\tau_{m-1}}, \pi(\mu)\rangle) \quad \text{for every } \mu\colon S \to A. \tag{20}$$

Sample a random variable $\mathbf{q}_m$ over the set of deterministic policies $\{\mu : S \to A\}$ and with the following probability measure:[5]

$$\Pr(\mathbf{q}_m = \mu) = (1-\alpha)\frac{w_m(\mu)}{\sum_{\mu': S\to A} w_m(\mu')} + \alpha\frac{1}{|A|^{|S|}} \quad \text{for all } \mu\colon S \to A. \tag{21}$$

(iii) (*Follow chosen policy*). For $t \in \tau_m$ and $m = 1, 2, \ldots,$ choose the action $\mathbf{a}_t = \mathbf{q}_m(s_t)$.

Remark 7.4. The main problem with the Lazy Tracking Forecaster algorithm is that the number of weight variables $|A|^{|S|}$ is exponential in the size of the state space.

Remark 7.5. The term $\langle R_{\tau_{m-1}}, \pi(\mu)\rangle$ in Equation (20) approximates the expected reward accumulated by following policy μ over the course of phase τ_{m-1}. The weights are updated recursively according to each policy's reward over the previous phase. The probability measure defined in Equation (21) tracks the optimal policy in the fashion of the fixed-share algorithm (Herbster and Warmuth [17]).

Remark 7.6. In contrast to the algorithms presented in the previous sections, the length of every phase is kept the same. By using the doubling trick of Cesa-Bianchi and Lugosi [9, §3.2], we can adapt the Lazy Tracking Forecaster algorithm to problems where the time horizon T is unknown. This technique partitions the time horizon into periods of exponentially increasing length and runs the Lazy Tracking Forecaster algorithm on each period independently.

As asserted in the following proposition, the Lazy Tracking Forecaster (Algorithm 5) minimizes the regret with respect to the new baseline of Equation (19). The proof (see the appendix) derives from existing results on the fixed-share algorithm of Herbster and Warmuth [17].

Proposition 7.1 (Expected Regret of Lazy Tracking Forecaster). *Let the length of all phases be* $|\tau| = \lceil T^{1/3}\rceil$. *Suppose that Assumptions 2.1 and 2.2 hold. If the agent follows the Lazy Tracking Forecaster algorithm with parameters* $\eta = T^{-2/3}$ *and* $\alpha = K_0/(\lceil T/\lceil T^{1/3}\rceil\rceil - 1)$, *then the following cumulative regret bound holds for large enough* T:

$$\mathcal{B}_T(K_0) - \sum_{t=0}^{T-1}\mathbb{E}[r_t(\mathbf{s}_t, \mathbf{a}_t)] \le |S|\log(|A|)(K_0+1)T^{2/3} + 2K_0\log(T^{2/3}/K_0)T^{2/3} + \tfrac{1}{2}T^{2/3} + (2e\gamma)T^{2/3}.$$

Remark 7.7. Observe that this bound is tighter than the bound of Proposition 4.1.

We now prove Theorem 7.1.

Proof of Theorem 7.1. Consider the Tracking FPL algorithm (Algorithm 4) and the Lazy Tracking Forecaster (Algorithm 5) with their parameters α set equal. Let all phases for the Lazy Tracking Forecaster algorithm have fixed length τ. Let M denote the number of phases for the Tracking FPL algorithm. By their definition, at every given step t and with probability α, the two algorithms follow a policy μ chosen uniformly at random. Hence, the difference in their expected cumulative reward is $1-\alpha$ times the same difference when the parameters α are set to 0. We will proceed to bound this latter quantity.

Observe that the Tracking FPL algorithm with $\alpha = 0$ is simply the Lazy FPL algorithm. The Lazy Tracking Forecaster with $\alpha = 0$ is just an exponentially weighted average forecaster (Cesa-Bianchi and Lugosi [9]) with one phase as the fundamental time step. Let $\mathbf{a}_t$ and $\mathbf{b}_t$ denote the actions generated by the Lazy Tracking Forecaster and the Tracking FPL algorithms, respectively. By setting the argument K_0 to the baseline $\mathcal{B}_T$ to 0, we shall derive the following bounds on their respective cumulative regrets:

$$\Omega(\sqrt{T\log(|A|)}) \le \mathcal{B}_T(0) - \sum_{t=0}^{T-1}\mathbb{E}[r_t(\mathbf{s}_t, \mathbf{a}_t)] \le \frac{|S|\log(|A|)}{\eta} + \frac{\eta\lceil T/|\tau|\rceil\,|\tau|^2}{2}, \tag{22}$$

$$\Omega(\sqrt{T\log(|A|)}) \le \mathcal{B}_T(0) - \sum_{t=0}^{T-1}\mathbb{E}[r_t(\mathbf{s}_t, \mathbf{b}_t)] \le \frac{4}{3}(2e\gamma + 2|A| + 4e + 1 + 2(|S|+3)|A|^2\gamma\log(T))T^{3/4+\epsilon}. \tag{23}$$

The upper bound of Equation (22) follows from an argument similar to Cesa-Bianchi and Lugosi [9, Theorem 2.1]; that of Equation (23) follows from Proposition 4.1. Both lower bounds are due to instances where the regret is no less than of the order of $\Omega(T^{1/2})$ (Cesa-Bianchi and Lugosi [9, Theorem 3.7]). The above bounds

[5] As in the fixed-share algorithm of Herbster and Warmuth [17], the action at each step is equal to the previous action with probability $1-\alpha+\alpha/|A|$, and equal to each different action with probability $\alpha/|A|$.

combine to give

$$\left|\sum_{t=0}^{T-1}\mathbb{E}[r_t(\mathbf{s}_t,\mathbf{a}_t)]-\sum_{t=0}^{T-1}\mathbb{E}[r_t(\mathbf{s}_t,\mathbf{b}_t)]\right| \le \frac{|S|\log(|A|)}{\eta}+\frac{\eta\lceil T/|\tau|\rceil\,|\tau|^2}{2}+\tfrac{4}{3}\big(2e\gamma+2|A|+4e+1+2(|S|+3)|A|^2\gamma\log(T)\big)T^{3/4+\epsilon}, \tag{24}$$

because the lower bounds are superseded by the upper bounds for all phase-partitions consistent with the assumptions of Proposition 4.1. By substituting the values $|\tau|=\lceil T^{1/3}\rceil$ and $\eta=T^{-2/3}$ and compounding the bound of Equation (24) to that of Proposition 7.1, we obtain the following bound:

$$\mathscr{B}_T(K_0)-\sum_{t=0}^{T-1}\mathbb{E}[r_t(\mathbf{s}_t,\mathbf{a}_t)] \le |S|\log(|A|)(K_0+2)T^{2/3}+2K_0\log(T^{2/3}/K_0)T^{2/3}+T^{2/3}+(2e\gamma)T^{2/3} + \tfrac{4}{3}\big(2e\gamma+2|A|+4e+1+2(|S|+3)|A|^2\gamma\log(T)\big)T^{3/4+\epsilon}. \tag{25}$$

At last, the claimed result follows by an argument similar to the proof of Theorem 4.1. □

Remark 7.8. The bound on expected cumulative regret of the Tracking FPL algorithm (cf. Equation (25)) is of the same order as that afforded by the Lazy FPL algorithm (cf. Proposition 4.1). This indicates that the critical factor in the convergence of the algorithm is its "laziness."

8. Conclusions. In this paper, we considered no-regret policies within the extended model of MDPs with arbitrarily varying rewards. We showed that a simple reinforcement learning algorithm achieves diminishing average regret against any oblivious opponent. In contrast to most of the online learning literature, the obliviousness of the opponent plays a key role in characterizing the performance that the agent can achieve. The algorithms presented in the different sections introduce techniques dealing with various possible challenges. The Lazy FPL algorithm deals with the Markovian dynamics and an unknown time horizon T. The Q-FPL algorithm circumvents the need to calculate the exact value functions. The Exploratory FPL algorithm overcomes partially observable reward functions. The Tracking FPL algorithm surmounts a more ambitious comparison baseline of regret composed of dynamic policies with infrequent changes. The salient features of all these algorithms can be combined to deal with combinations of the mentioned challenges.

An oblivious environment and a completely nonoblivious (i.e., omnipotent) environment are two opposite extremes. It would be interesting to model different levels of obliviousness and study their effect on the achievable regret. For example, one can consider opponents that select reward functions depending on delayed information or imperfect monitoring of the history (e.g., opponents that only observe visits by the agent to particular states). The main focus in this paper was computational efficiency from the reinforcement learning perspective, where low complexity per stage is desired. Optimizing the convergence rate of the regret remains an open topic for further research.

Appendix. Proofs.

Proof of Lemma 4.1. By introducing indicator functions, we obtain

$$\mathbb{E}[r_j(\mathbf{s}_t,\mathbf{a}_t)]=\mathbb{E}\left\{\sum_{(s,a)\in S\times A} r_j(s,a)\mathbf{1}_{[(\mathbf{s}_t,\mathbf{a}_t)=(s,a)]}\right\} \tag{26}$$

$$=\sum_{(s,a)\in S\times A} r_j(s,a)\mathbb{E}\mathbf{1}_{[(\mathbf{s}_t,\mathbf{a}_t)=(s,a)]} \tag{27}$$

$$=\sum_{(s,a)\in S\times A} r_j(s,a)\Pr((\mathbf{s}_t,\mathbf{a}_t)=(s,a)), \tag{28}$$

where Equation (26) follows by definition and the use of indicator functions, Equation (27) is justified by Assumption 2.2, and Equation (28) follows again by definition. □

Proof of Lemma 2.1. By Lemma 4.1, for a stationary policy $\mu\in\Sigma$, we have

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}[r_t(\mathbf{s}_t,\mathbf{a}_t)]=\frac{1}{T}\sum_{t=0}^{T-1}\langle r_t, d_t(\mu;s_0)\rangle.$$

By Assumption 2.1 and the summability of the sequence $e^{1-t/\gamma}$, we have

$$\left|\frac{1}{T}\sum_{t=0}^{T-1}\langle r_t, d_t(\mu; s_0)\rangle - \frac{1}{T}\sum_{t=0}^{T-1}\langle r_t, \pi(\mu)\rangle\right| \le \frac{1}{T}\sum_{t=0}^{T-1} 2e^{1-t/\gamma} = 2e\gamma/T.$$

By definition, we have

$$\frac{1}{T}\sum_{t=0}^{T-1}\langle r_t, \pi(\mu)\rangle = \langle \hat{r}_T, \pi(\mu)\rangle.$$

Putting these pieces together, we obtain

$$\left|\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}[r_t(\mathbf{s}_t, \mathbf{a}_t)] - \langle \hat{r}_T, \pi(\mu)\rangle\right| \le 2e\gamma/T.$$

By a similar argument, we have

$$\left|\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}[\hat{r}_T(\mathbf{s}_t, \mathbf{a}_t)] - \langle \hat{r}_T, \pi(\mu)\rangle\right| \le 2e\gamma/T.$$

The two claims follow from taking the supremum over the set of stationary policies. □

Proof of Lemma 4.3. For nonnegative integers n and m such that $n \ge m$, algebraic manipulation yields

$$\begin{aligned}\left\|\frac{1}{n}\sum_{j=0}^{n-1} r_j - \frac{1}{l}\sum_{j=0}^{l-1} r_j\right\|_\infty &= \left\|\frac{1}{n}\sum_{j=0}^{l-1} r_j + \frac{1}{n}\sum_{j=l}^{n-1} r_j - \frac{1}{l}\sum_{j=0}^{l-1} r_j\right\|_\infty \\ &\le \frac{1}{n}\left\|\sum_{j=l}^{n-1} r_j\right\|_\infty + \left|\frac{n-l}{n}\right|\left\|\frac{1}{l}\sum_{j=0}^{l-1} r_j\right\|_\infty \\ &\le 2\frac{n-l}{n},\end{aligned}$$

where the last inequality follows from the fact that $r_0, r_1, \ldots,$ are bounded by 1. □

Proof of Lemma 4.4. Let $t' \in \tau_{m+1}$ and $t \in \tau_m$. By the assumption of Theorem 4.1, the cumulative distribution functions of $\mathbf{n}_{t'}(a)$ and $\mathbf{n}_t(a)$ satisfy the following bounds for all $z, z' \in \mathbb{R}$:

$$\left|F_{\mathbf{n}_{t'}(a)}(z) - F_{\mathbf{n}_t(a)}(z)\right| \le \frac{\zeta_{m+1} - \zeta_m}{2\zeta_{m+1}},$$

$$\left|F_{\mathbf{n}_{t'}(a)}(z) - F_{\mathbf{n}_{t'}(a)}(z')\right| \le \frac{\zeta_{m+1}}{2}|z - z'|.$$

Likewise, for $a, a' \in A$, we have

$$\left|F_{\mathbf{n}_{t'}(a)-\mathbf{n}_{t'}(a')}(z) - F_{\mathbf{n}_t(a)-\mathbf{n}_t(a')}(z)\right| \le \frac{\zeta_{m+1} - \zeta_m}{2\zeta_{m+1}}, \tag{29}$$

$$\left|F_{\mathbf{n}_{t'}(a)-\mathbf{n}_{t'}(a')}(z) - F_{\mathbf{n}_{t'}(a)-\mathbf{n}_{t'}(a')}(z')\right| \le \frac{\zeta_{m+1}}{2}|z - z'|. \tag{30}$$

By Lemma 4.3, we have

$$\left\|\hat{r}_{\tau_{0:m+1}} - \hat{r}_{\tau_{0:m}}\right\|_\infty \le 2\left|\tau_{m+1}\right| / \left|\tau_{0:m+1}\right|. \tag{31}$$

Observe that the linear programs (cf. Equation (6)) at the mth and $m+1$th phases differ only in their right-hand constraint vectors, whose difference is bounded by Equation (31). It follows by Renegar [23, Theorem 1.1] that the optimal values λ_m and λ_{m+1} satisfy

$$|\lambda_{m+1} - \lambda_m| \le \left\|\hat{r}_{\tau_{0:m+1}} - \hat{r}_{\tau_{0:m}}\right\|_\infty.$$

Likewise, by Robinson [24, Corollary 3.1], the solutions h_{m+1} and h_m differ as follows:

$$\begin{aligned}\|h_{m+1} - h_m\|_\infty &\le (|S|+1)\left\|\hat{r}_{\tau_{0:m+1}} - \hat{r}_{\tau_{0:m}}\right\|_\infty &(32)\\ &\le 2(|S|+1)\left|\tau_{m+1}\right| / \left|\tau_{0:m+1}\right|. &(33)\end{aligned}$$

Starting from the definition of Algorithm 1, for every $s \in S$, $a \in A$, and $m = 0, 1, \ldots,$

$$\begin{aligned}\sigma_{m+1}(a; s) &\triangleq \Pr(\mathbf{a}_{t'} = a \mid \mathbf{s}_{t'} = s)\\ &= \Pr\Bigg(\hat{r}_{\tau_{0:m+1}}(s, a) + \sum_{s' \in S} P(s' \mid s_t, a) h_m(s') + \mathbf{n}_{t'}(a) > \hat{r}_{\tau_{0:m+1}}(s, a')\\ &\qquad + \sum_{s' \in S} P(s' \mid s_t, a') h_m(s')[h_{m+1}(\mathbf{s}_{t+1})] + \mathbf{n}_{t'}(a') \text{ for all } a' \neq a\Bigg) \qquad (34)\\ &= \prod_{a' \neq a} \Pr\Bigg(\mathbf{n}_{t'}(a) - \mathbf{n}_{t'}(a') > \hat{r}_{\tau_{0:m+1}}(s, a') - \hat{r}_{\tau_{0:m+1}}(s, a)\\ &\qquad + \sum_{s' \in S} P(s' \mid s_t, a') h_m(s') - \sum_{s' \in S} P(s' \mid s_t, a) h_m(s')\Bigg), \qquad (35)\end{aligned}$$

where the probability measure is over the randomization $\mathbf{n}_{t'}$, whereas the expectation is over the transition probabilities of the MDP. Equation (34) is due to the definition of Algorithm 1 (Equation (7)). Equation (35) is obtained by independence of the random variables $\mathbf{n}_{t'}(a)$ for $a \in A$. By comparing Equations (35) applied to σ_{m+1} and σ_m, and using Equations (31), (33), (29), and (30), we obtain

$$\|\sigma_{m+1}(\cdot; s) - \sigma_m(\cdot; s)\|_\infty \leq (|A| - 1)\left(\frac{\zeta_{m+1}}{2}(4 + 2(|S| + 1))\frac{|\tau_{m+1}|}{|\tau_{0:m+1}|} + \frac{\zeta_{m+1} - \zeta_m}{2\zeta_{m+1}}\right)$$

for all $s \in S$. For the 1-norm, we have

$$\|\sigma_{m+1}(\cdot; s) - \sigma_m(\cdot; s)\|_1 \leq (|S| + 3)\,|A|\,(|A| - 1)\left(\zeta_{m+1}\frac{|\tau_{m+1}|}{|\tau_{0:m+1}|} + \frac{\zeta_{m+1} - \zeta_m}{\zeta_{m+1}}\right) \qquad (36)$$

for all $s \in S$.

For the second part of the lemma, let P_μ be the transition matrix associated with a stationary policy $\mu\colon S \to A$. The element of P_μ in row (s', a') and column (s, a) is the probability that the next state-action pair is (s', a') if the current one is (s, a) and policy μ is followed. Let $d \in \Delta(S \times A)$ be a probability vector specifying the initial state-action pair $(s_0, \mu(s_0))$. We first show by induction that

$$\|\mathrm{P}^j_{\sigma_{m+1}} d - \mathrm{P}^j_{\sigma_m} d\|_1 \leq j(|S| + 3)\,|A|^2\left(\zeta_{m+1}\frac{|\tau_{m+1}|}{|\tau_{0:m+1}|} + \frac{\zeta_{m+1} - \zeta_m}{\zeta_{m+1}}\right) \qquad (37)$$

for $j = 1, 2, \ldots$. Let $e_1, \ldots, e_{|S \times A|}$ denote the elementary vectors in $\mathbb{R}^{|S \times A|}$. For the base case $j = 1$, we have

$$\begin{aligned}\left\|\mathrm{P}_{\sigma_{m+1}} d - \mathrm{P}_{\sigma_m} d\right\|_1 &\leq \max_{n = 1, \ldots, |S \times A|} \left\|\mathrm{P}_{\sigma_{m+1}} e_n - \mathrm{P}_{\sigma_m} e_n\right\|_1\\ &= \max_{(s, a) \in S \times A}\left|\sum_{(s', a') \in S \times A} P(s' \mid s, a)\sigma_{m+1}(a'; s') - P(s' \mid s, a)\sigma_m(a'; s')\right|\\ &= \max_{(s, a) \in S \times A}\left|\sum_{s' \in S} P(s' \mid s, a)\sum_{a'} \sigma_{m+1}(a'; s') - \sigma_m(a'; s')\right|\\ &\leq \max_{s' \in S}\left|\sum_{a' \in A} \sigma_{m+1}(a'; s') - \sigma_m(a'; s')\right|\\ &= \max_{s' \in S} \|\sigma_{m+1}(\cdot; s') - \sigma_m(\cdot; s')\|_1\\ &\leq (|S| + 3)\,|A|^2\left(\zeta_{m+1}\frac{|\tau_{m+1}|}{|\tau_{0:m+1}|} + \frac{\zeta_{m+1} - \zeta_m}{\zeta_{m+1}}\right),\end{aligned}$$

where the last inequality follows from Equation (36). Next, suppose that for some j, we have

$$\left\|\mathrm{P}^j_{\sigma_{m+1}} d - \mathrm{P}^j_{\sigma_m} d\right\|_1 = j(|S| + 3)\,|A|^2\left(\zeta_{m+1}\frac{|\tau_{m+1}|}{|\tau_{0:m+1}|} + \frac{\zeta_{m+1} - \zeta_m}{\zeta_{m+1}}\right).$$

By the triangle inequality and the same argument as the base case, we obtain

$$\begin{aligned}\left\|\mathrm{P}_{\sigma_{m+1}}^{j+1}d-\mathrm{P}_{\sigma_m}^{j+1}d\right\|_1 &\le \left\|\mathrm{P}_{\sigma_{m+1}}\mathrm{P}_{\sigma_{m+1}}^{j}d-\mathrm{P}_{\sigma_m}\mathrm{P}_{\sigma_{m+1}}^{j}d\right\|_1+\left\|\mathrm{P}_{\sigma_m}\mathrm{P}_{\sigma_{m+1}}^{j}d-\mathrm{P}_{\sigma_m}\mathrm{P}_{\sigma_m}^{j}d\right\|_1\\ &= (|S|+3)\,|A|^2\left(\zeta_{m+1}\frac{|\tau_{m+1}|}{|\tau_{0:\,m+1}|}+\frac{\zeta_{m+1}-\zeta_m}{\zeta_{m+1}}\right)\\ &\quad + j(|S|+3)\,|A|^2\left(\zeta_{m+1}\frac{|\tau_{m+1}|}{|\tau_{0:\,m+1}|}+\frac{\zeta_{m+1}-\zeta_m}{\zeta_{m+1}}\right),\end{aligned}$$

which establishes Equation (37). At last, by the triangle inequality, Equation (37), and Assumption 2.1, it follows that for every positive integer g, and every initial state s_0 and corresponding distribution d,

$$\begin{aligned}\|\pi(\sigma_{m+1})-\pi(\sigma_m)\|_1 &= \left\|\mathrm{P}_{\sigma_{m+1}}^{g}d-\mathrm{P}_{\sigma_m}^{g}d\right\|_1+\left\|\pi(\sigma_{m+1})-\mathrm{P}_{\sigma_{m+1}}^{g}d\right\|_1+\left\|\pi(\sigma_m)-\mathrm{P}_{\sigma_m}^{g}d\right\|_1\\ &\le g(|S|+3)\,|A|^2\left(\zeta_{m+1}\frac{|\tau_{m+1}|}{|\tau_{0:\,m+1}|}+\frac{\zeta_{m+1}-\zeta_m}{\zeta_{m+1}}\right)+4e^{1-g/\gamma}. \quad \square\end{aligned}$$

Proof of Lemma 4.5. Let $t\in\tau_m$; let action $\mathbf{a}_t^+$ follow policy σ_m^+, and action $\mathbf{a}_t$ follow σ_m. Recall that the action $\mathbf{a}_t^+$ is an optimal action against an MDP with fixed reward function $\hat{r}_{\tau_{0:\,m-1}}$. Let us consider the following random variables for $(s,a)\in S\times A$:

$$\hat{r}_{\tau_{0:\,m-1}}(s,a)+\sum_{s'\in S}P(s'\mid s_t,a)h_m(s')+\mathbf{n}_t(a). \tag{38}$$

For ease of notation, we define, for $(s,a)\in S\times A$,

$$\xi_m(s,a)=\hat{r}_{\tau_{0:\,m-1}}(s,a)+\sum_{s'\in S}P(s'\mid s_t,a)h_m(s').$$

Observe that $\xi_m(s,\mathbf{a}_t^+)\ge\xi_m(s,a)$ for every $a\ne\mathbf{a}_t^+$ by definition. Let Ψ denote the interval over which the supports of the random variables $\mathbf{n}_t(\mathbf{a}_t^+)+\xi_m(s,\mathbf{a}_t^+)$ and $\mathbf{n}_t(a)+\xi_m(s,a)$ overlap. This interval Ψ has length $2/\zeta_m-(\xi_m(s,\mathbf{a}_t^+)-\xi_m(s,a))$. Combining this fact with the fact that $\mathbf{n}_t(\mathbf{a}_t^+)$ and $\mathbf{n}_t(a)$ are independent and have uniform distributions specified by the assumption of Theorem 4.1, we have, for every $s\in S$,

$$\begin{aligned}\Pr(\mathbf{a}_t=a\mid\mathbf{s}_t=s) &= \Pr(\mathbf{n}_t(a)+\xi_m(s,a)>\mathbf{n}_t(\mathbf{a}_t^+)+\xi_m(s,\mathbf{a}_t^+))\\ &= \tfrac{1}{2}\Pr(\mathbf{n}_t(\mathbf{a}_t^+)+\xi_m(s,\mathbf{a}_t^+)\in\Psi,\mathbf{n}_t(a)+\xi_m(s,a)\in\Psi)\\ &\le \begin{cases}\dfrac{\zeta_m}{4}(2/\zeta_m-(\xi_m(s,\mathbf{a}_t^+)-\xi_m(s,a)))^2, & \text{if } \xi_m(s,\mathbf{a}_t^+)-\xi_m(s,a)\le 2/\zeta_m,\\ 0, & \text{otherwise.}\end{cases}\end{aligned} \tag{39}$$

Observe next that

$$\begin{aligned}\left|\langle R_{\tau_{0:\,m-1}},\pi(\sigma_m)-\pi(\sigma_m^+)\rangle\right| &= |\tau_{0:\,m-1}|\left|\langle\hat{r}_{\tau_{0:\,m-1}},\pi(\sigma_m)\rangle-\langle\hat{r}_{\tau_{0:\,m-1}},\pi(\sigma_m^+)\rangle\right|\\ &\le |\tau_{0:\,m-1}|\max_{s\in S}\sum_{a\ne\mathbf{a}_t^+}(\xi_m(s,\mathbf{a}_t^+)-\xi_m(s,a))\Pr(\mathbf{a}_t=a\mid\mathbf{s}_t=s)\\ &\le |\tau_{0:\,m-1}|\,(|A|-1)(2/\zeta_m)\frac{\zeta_m}{4}(2/\zeta_m)^2\\ &\le 2\,|A|\,\frac{|\tau_{0:m-1}|}{\zeta_m^2},\end{aligned}$$

where the second-to-last inequality follows by Equation (39).

Proof of Corollary 5.1 (Outline). The desired result follows an approach similar to Proposition 4.1 and Theorem 4.1. First, let ρ_m denote the policy induced by the (ϵ,δ)-approximation algorithm for the mth phase. Let P_{ρ_m} and $\pi(\rho_m)$ denote the transition probability matrix and the stationary distribution associated with ρ_m; and likewise for σ_m. Observe that, by Definition 5.1,

$$\left\|\mathrm{P}_{\rho_m}-\mathrm{P}_{\sigma_m}\right\|_\infty\le\max_{s\in S}\|\rho_m(\cdot;s)-\sigma_m(\cdot;s)\|_1\le\epsilon+\delta.$$

By Schweitzer [25, §6], the stationary distributions $\pi(\rho_m)$ and $\pi(\sigma_m)$ satisfy

$$\|\pi(\rho_m) - \pi(\sigma_m)\|_1 \le \left\|Z_{\sigma_m}\right\|_\infty \left\|P_{\rho_m} - P_{\sigma_m}\right\|_\infty \le \sup_{\sigma\in\Sigma} \|Z_\sigma\|_\infty (\epsilon + \delta).$$

Hence, we have

$$\begin{aligned}\sum_{t=0}^{T-1} \mathbb{E}[r_t(\mathbf{s}_t, \mathbf{a}_t)] &\ge \sum_{m=0}^{M-1} (\langle R_{\tau_m}, \pi(\rho_m)\rangle - 2e\gamma) \\ &\ge \sum_{m=0}^{M-1} (\langle R_{\tau_m}, \pi(\sigma_m)\rangle - 2e\gamma) - \sup_{\sigma\in\Sigma} \|Z_\sigma\|_\infty (\epsilon + \delta)T,\end{aligned}$$

where the first inequality is justified by the same argument as Step 0 of the proof of Proposition 4.1. This bound is similar to Equation (10) of the proof of Proposition 4.1 with one additional term. The claimed result follows by arguments similar to the proofs of Proposition 4.1 and Theorem 4.1. □

Proof of Corollary 6.1 (Outline). By introducing exploration phases as described above, we ensure that $\mathbf{z}_t$ is an unbiased estimator for r_t. Indeed, observe that for every $(s, a) \in S \times A$ and t large enough,

$$\begin{aligned}\Pr((\mathbf{s}_t, \mathbf{a}_t) = (s, a) \mid s_0) &= \Pr(\mathbf{s}_t = s \mid s_0) \Pr(\mathbf{a}_t = a \mid \mathbf{s}_t = s) \\ &\ge \Pr(\mathbf{s}_t = s \mid s_0)\phi_m / |A|.\end{aligned}$$

Next, observe that the ergodicity assumption (Assumption 2.1) guarantees that there exists an $\epsilon > 0$ such that for every $s \in S$ and large enough t,

$$\Pr(\mathbf{s}_t = s \mid s_0) > \epsilon.$$

Moreover, we have $\phi_m > 0$ by assumption. Hence, if the opponent is oblivious and for large enough t, we obtain

$$\mathbb{E}[\mathbf{z}_t(s, a)] = r_t(s, a) \quad \text{for all } (s, a) \in S \times A,$$

and in turn,

$$\mathbb{E}\left[\frac{1}{t}\sum_{j=0}^{t-1} \mathbf{z}_j(s, a)\right] = \hat{r}_t(s, a) \quad \text{for all } (s, a) \in S \times A.$$

Therefore, we conclude by Lemma 4.2 that the policy induced by the Exploratory FPL algorithm is still optimal against $\hat{r}_{\tau_{0:m-1}} + \mathbf{n}_t$. All the remaining steps of the proof of Proposition 4.1 hold unchanged if we exclude the exploration phases. Because these phases incur an overhead of the order of $O(M)$ by Equation (18), we obtain a bound analogous to Equation (8). Finally, the claim follows by the same argument as the proof of Theorem 4.1. □

Proof of Proposition 7.1. For ease of notation, we write $M = \lceil T/|\tau| \rceil$ to denote the number of phases of the Lazy Tracking Forecaster algorithm. Observe that Lazy Tracking Forecaster is the same as the tracking forecaster of Herbster and Warmuth [17], with the exception that the fundamental time step is an entire phase in our new setting. Our claim follows from Cesa-Bianchi and Lugosi [9, Theorem 5.2 and Corollary 5.1] by adjusting the time scale.

The crucial observation is that at Step 2 of Algorithm 5, the weights are not updated according to the aggregate reward obtained by following policy μ over each phase τ_m, but according to the expected reward in the stationary state-action distribution of each policy μ in each phase τ_m. Consequently, Cesa-Bianchi and Lugosi [9, Theorem 5.2] gives the bound

$$\mathcal{B}_T(K_0) - \sum_{m=0}^{M-1} \langle R_{\tau_m}, \pi(\mathbf{q}_m)\rangle \le \frac{|S|\log(|A|)}{\eta}(K_0 + 1) + \frac{1}{\eta}(M-1)H\left(\frac{K_0}{M-1}\right) + \frac{\eta M |\tau|^2}{2}.$$

The required result follows by observing that we can approximate

$$\sum_{j\in\tau_m} \mathbb{E}[r_j(\mathbf{s}_j, \mathbf{a}_j) \mid s^-],$$

where the actions $\mathbf{a}_j$ follow policy $\mathbf{q}_m$ and s^- is the state of the MDP at the beginning of phase τ_m, by

$$\langle R_{\tau_m}, \pi(\mathbf{q}_m)\rangle \triangleq \sum_{j\in\tau_m} \langle r_j, \pi(\mathbf{q}_m)\rangle.$$

As shown in Step 0 of the proof of Proposition 4.1, we have

$$\left| \sum_{j \in \tau_m} \mathbb{E}[r_j(\mathbf{s}_j, \mathbf{a}_j) \mid s^-] - \langle R_{\tau_m}, \pi(\mathbf{q}_m) \rangle \right| \leq 2e\gamma$$

for $m = 0, \ldots, M-1$, which accounts for the term $2e\gamma M$. Finally, the claim follows by substituting $|\tau| = \lceil T^{1/3} \rceil$ and $\eta = T^{-2/3}$, and observing that for $0 \leq p < 1/2$, we have

$$H(p) < 2p \log(1/p),$$

so that for large enough T,

$$H\left(\frac{K_0}{\lceil T/|\tau| \rceil - 1} \right) < 2 \frac{K_0}{\lceil T/|\tau| \rceil - 1} \log\left(\frac{\lceil T/|\tau| \rceil - 1}{K_0} \right). \quad \square$$

Acknowledgments. This research was partially funded by the NSERC Postgraduate Graduate Scholarship, the McGill Engineering Doctoral Award, ISF Undergrant 890015, and the Horev Fellowship.

References

[1] Auer, P., N. Cesa-Bianchi, Y. Freund, R. E. Schapire. 2002. The nonstochastic multiarmed bandit problem. *SIAM J. Comput.* **32**(1) 48–77.
[2] Aumann, R. J. 1964. Markets with a continuum of traders. *Econometrica* **32** 39–50.
[3] Bertsekas, D. P. 2001. *Dynamic Programming and Optimal Control*, 2nd ed, Vol. 2. Athena Scientific, Nashua, NH.
[4] Bertsekas, D. P., J. N. Tsitsiklis. 1996. *Neuro-Dynamic Programming*. Athena Scientific, Nashua, NH.
[5] Blackwell, D. 1956. An analog of the minimax theorem for vector payoffs. *Pacific J. Math.* **6**(1) 1–8.
[6] Bobkov, S. G., P. Tetali. 2006. Modified logarithmic Sobolev inequalities in discrete settings. *J. Theoret. Probab.* **19**(2) 289–336.
[7] Borkar, V. S., S. P. Meyn. 2000. The O.D.E. method for convergence of stochastic approximation and reinforcement learning. *SIAM J. Control Optim.* **38**(2) 447–469.
[8] Brafman, R. I., M. Tennenholtz. 2003. R-max—A general polynomial time algorithm for near-optimal reinforcement learning. *J. Machine Learning Res.* **3** 213–231.
[9] Cesa-Bianchi, N., G. Lugosi. 2006. *Prediction, Learning, and Games*. Cambridge University Press, New York.
[10] Crites, R. H., A. G. Barto. 1995. An actor/critic algorithm that is equivalent to *Q*-learning. *Advances in Neural Information Processing Systems 7*. MIT Press, Cambridge, 401–408.
[11] Even-Dar, E., S. Kakade, Y. Mansour. 2004. Experts in a Markov decision process. *Advances in Neural Information Processing Systems 17*. MIT Press, Cambridge, 401–408.
[12] Filar, J., K. Vrieze. 1997. *Competitive Markov Decision Processes*. Springer-Verlag, New York.
[13] Freund, Y., R. E. Schapire. 1999. Adaptive game playing using multiplicative weights. *Games Econom. Behav.* **29**(12) 79–103.
[14] Fudenberg, D., D. M. Kreps. 1993. Learning mixed equilibria. *Games Econom. Behav.* **5**(3) 320–367.
[15] Fudenberg, D., D. K. Levine. 1998. *The Theory of Learning in Games*. MIT Press, Cambridge.
[16] Hannan, J. 1957. Approximation to Bayes risk in repeated play. *Contributions to the Theory of Games*, Vol. 3. Princeton University Press, Princeton, NJ, 97–139.
[17] Herbster, M., M. K. Warmuth. 1998. Tracking the best expert. *Machine Learning* **32**(2) 151–178.
[18] Kalai, A., S. Vempala. 2005. Efficient algorithms for online decision problems. *J. Comput. System Sci.* **71**(3) 291–307.
[19] Littlestone, N., M. K. Warmuth. 1994. The weighted majority algorithm. *Inform. Comput.* **108**(2) 212–261.
[20] Mannor, S., N. Shimkin. 2003. The empirical Bayes envelope and regret minimization in competitive Markov decision processes. *Math. Oper. Res.* **28**(2) 327–345.
[21] Mannor, S., N. Shimkin. 2008. Regret minimization in repeated matrix games with variable stage duration. *Games Econom. Behav.* **63**(1) 227–258.
[22] Merhav, N., E. Ordentlich, G. Seroussi, M. J. Weinberger. 2002. On sequential strategies for loss functions with memory. *IEEE Trans. Inform. Theory* **48**(7) 1947–1958.
[23] Renegar, J. 1994. Some perturbation theory for linear programming. *Math. Programming* **65**(1) 73–91.
[24] Robinson, S. M. 1973. Bounds for error in the solution set of a perturbed linear program. *Linear Algebra Its Appl.* **6** 69–81.
[25] Schweitzer, P. J. 1968. Perturbation theory and finite Markov chains. *J. Appl. Probab.* **5** 410–413.
[26] Shapley, L. 1953. Stochastic games. *Proc. National Acad. Sci.* **39**(10) 1095–1100.
[27] Watkins, C., P. Dayan. 1992. *Q*-learning. *Machine Learning* **8** 279–292.
[28] Zinkevich, M. 2003. Online convex programming and generalized infinitesimal gradient ascent. *Proc. Twentieth Internat. Conf. Machine Learning*. AAAI Press, Cambridge, MA, http://www.hpl.hp.com/conferences/icml2003/titlesAndAuthors.html.