POLITEX: Regret Bounds for Policy Iteration Using Expert Prediction

Yasin Abbasi-Yadkori [1] **Peter L. Bartlett** [2] **Kush Bhatia** [2] **Nevena Lazić** [3] **Csaba Szepesvári** [4] **Gellért Weisz** [4]

Abstract

We present POLITEX (POLicy ITeration with EXpert advice), a variant of policy iteration where each policy is a Boltzmann distribution over the sum of action-value function estimates of the previous policies, and analyze its regret in continuing RL problems. We assume that the value function error after running a policy for τ time steps scales as $\varepsilon(\tau) = \varepsilon_0 + \tilde{\mathcal{O}}(\sqrt{d/\tau})$, where ε_0 is the worst-case approximation error and d is the number of features in a compressed representation of the state-action space. We establish that this condition is satisfied by the LSPE algorithm under certain assumptions on the MDP and policies. Under the error assumption, we show that the regret of POLITEX in uniformly mixing MDPs scales as $\tilde{\mathcal{O}}(d^{1/2}T^{3/4} + \varepsilon_0 T)$, where $\tilde{\mathcal{O}}(\cdot)$ hides logarithmic terms and problem-dependent constants. Thus, we provide the first regret bound for a fully practical model-free method which only scales in the number of features, and not in the size of the underlying MDP. Experiments on a queuing problem confirm that POLITEX is competitive with some of its alternatives, while preliminary results on Ms Pacman (one of the standard Atari benchmark problems) confirm the viability of POLITEX beyond linear function approximation.

1. Introduction

We study online no-regret model-free algorithms for infinite horizon reinforcement learning (RL) problems, which capture long-horizon tasks such as routing problems and game playing. Model-based reinforcement learning (RL) algorithms estimate a model of the transition dynamics and plan according to the model, while model-free algorithms (e.g., (Mnih et al., 2015)) directly optimize the objective of interest. Model-free algorithms can often be competitive with model-based algorithms, and theoretical evidence suggests that this is not completely accidental (Strehl et al., 2006; Azar et al., 2017; Jin et al., 2018; Abbasi-Yadkori et al., 2019). However, existing theory either applies to settings with no function approximation (Strehl et al., 2006; Azar et al., 2017; Jin et al., 2018), or to systems with particular structure where action-value functions are known to belong to a known linear function space (Abbasi-Yadkori et al., 2019). In this paper we ask the following question: *Can we design computationally-efficient, provable model-free algorithms for infinite horizon RL problems with value function generalization?* To this end, we propose POLITEX, a variant of policy iteration (PI), and analyze its performance for infinite-horizon average-cost problems in terms of high-probability regret with respect to a fixed reference policy.

PI algorithms alternate between estimating the value of a policy and generating a new policy, typically based on the most recent value estimate (Bertsekas, 2011). In POLITEX, the policy in each phase is a Boltzmann distribution over the sum of value function estimates of *all* previous policies. POLITEX is simple and efficient to implement whenever an effective value function estimation method is available. Importantly, no confidence sets or posterior distributions for transition dynamics or value functions are needed. We discuss and empirically evaluate versions of POLITEX that rely on (1) linear value functions estimated using the least-squares policy evaluation (LSPE) method of Bertsekas & Ioffe (1996) and (2) deep neural networks and the nonlinear TD(0) algorithm. Preliminary experimental results demonstrate the benefits of POLITEX over several baselines, both with linear and neural function approximation.

The algorithm and analysis are based on a reduction of the control of MDPs to expert prediction problems (Even-Dar et al., 2009), where we have an expert algorithm in each state, and the losses fed to the algorithm are value functions. The Boltzmann policy arises as a result of using the exponential weights algorithm in each state. The regret of POLITEX depends on the value function error. In the case of linear function approximation, we show that the LSPE error is controlled for uniformly mixing MDPs whenever a certain "feature excitation" condition is met. Interestingly, our results do not depend on the size of the MDP.

[1]Adobe Research [2]UC Berkeley [3]Google Brain [4]DeepMind. Correspondence to: Nevena Lazic <nevena@google.com>.

Proceedings of the 36th International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019.

1.1. Related work

Model-based RL: Model-based online RL has been a topic of intense research in recent years. Algorithms typically either (1) construct confidence sets for transition dynamics and the reward function, and find policies using the *optimism principle* (Auer et al., 2010; Bartlett & Tewari, 2009; Abbasi-Yadkori & Szepesvári, 2011; Abbasi-Yadkori, 2012), or (2) maintain a posterior distribution over the unknown quantities and use Thompson sampling (Osband & Van Roy, 2014; Osband et al., 2016; 2017; Russo et al., 2018). While such algorithms are well-understood and have strong theoretical guarantees in the case of tabular MDPs (Auer et al., 2010; Bartlett & Tewari, 2009; Strehl & Littman, 2008) and linear continuous systems (Abbasi-Yadkori & Szepesvári, 2011; Abbasi-Yadkori, 2012), they are difficult to apply to general RL problems with large state spaces, in part because the construction of appropriate confidence sets or posteriors can be challenging in practice.

Model-free RL: Model-free RL algorithms avoid estimating transition dynamics, and instead find optimal policies by directly optimizing estimated *value functions* (Sutton, 1988). In the case of tabular MDPs, regret bounds are typically obtained by constructing confidence sets for value functions and using the optimism principle. Constructing confidence sets for value functions can be challenging due to recursive nature of Bellman equations, and becomes significantly more complicated in large problems with non-realizable value function approximation. As a result, theoretical analysis of existing algorithms are limited to finite-horizon tabular MDPs, where confidence intervals can be propagated backwards and there are only a finite number of stages (Osband et al., 2016; 2017; Wen & Van Roy, 2017; Jin et al., 2018). For MDPs with rich observations and function approximation, Jiang et al. (2017) introduce a complexity measure called Bellman rank, and propose an optimistic algorithm for episodic RL with PAC guarantees for MDPs with low Bellman rank. However, the algorithm involves an elimination step which has no known efficient implementation.

Reduction to expert prediction: Our approach is based on a reduction of MDP control to an expert prediction problem. The reduction was first proposed by Even-Dar et al. (2009) for the online control of finite-state MDPs with changing cost functions. A similar approach is also presented by Yu et al. (2009). It has since been extended to structured prediction (Daumé III et al., 2009; Ross et al., 2011), finite MDPs with known dynamics and bandit feedback (Neu et al., 2014), LQ tracking with known dynamics (Abbasi-Yadkori et al., 2014), linearly solvable MDPs (Neu & Gómez, 2017), and adaptive control of LQ systems (Abbasi-Yadkori et al., 2019). The work of Abbasi-Yadkori et al. (2019) served as the starting point for the current paper. The main difference to this work is that in LQR problems the value functions are quadratic and there is no approximation error (which simplifies the analysis), while the state-action space is continuous and unbounded (which poses different analysis challenges). Here, we consider MDPs with finite but possibly large state spaces, and linear value function approximation. Our algorithm is a version of policy iteration, where the policy in each phase is a Boltzmann distribution over the sum of all previous value function estimates, as opposed to the most recent one. This is a direct consequence of using an expert algorithm, in this case the exponentially-weighted average forecaster.

Value function approximation: Many existing works attempt to extend count-based exploration methods to settings with rich observations and function approximation using heuristics. Such approaches include those of Ostrovski et al. (2017); O'Donoghue et al. (2018); Fortunato et al. (2018); Machado et al. (2017; 2018); Bellemare et al. (2016); Taïga et al. (2018). Our regret analysis relies on finite-sample bounds for linear value function estimation; in particular, we analyze the LSPE algorithm of Bertsekas & Ioffe (1996), and adapt the asymptotic convergence analysis of Yu & Bertsekas (2009) to the finite-sample case. Convergence analysis of various temporal difference learning methods has a rich history (Tsitsiklis & Van Roy, 1997; 1999; Antos et al., 2008; Sutton et al., 2009; Maei et al., 2010; Lazaric et al., 2012; Geist & Scherrer, 2014; Farahmand et al., 2016; Liu et al., 2012; 2015). Yu & Bertsekas (2009) have shown almost-sure convergence of on-policy average-cost LSPE. A finite-time analysis of the LSTD algorithm for discounted problems has been shown by Lazaric et al. (2012), and sharpened for the LQ problem by Tu & Recht (2017).

Similar algorithms: POLITEX can be seen as a softened and averaged version of policy iteration. Most existing performance bounds for policy iteration, such as that of Lazaric et al. (2012), apply to the discounted setting and involve a *concentrability coefficient* term, which is the result of using an argument based on the contraction-mapping theorem (Szepesvári, 2010). Our results do not depend on this term or on the size of the MDP. Two recent algorithms with a similar form to POLITEX are MPO (Abdolmaleki et al., 2018) and Quinoa (Degrave et al., 2018), which optimize a relative entropy-regularized objective.

2. Problem definition and background

2.1. Definitions and notation

For an integer d, we let $[d] = \{1, 2, \ldots, d\}$ denote the first d positive integers. We use $\Delta_{\mathcal{S}}$ to denote the space of probability distributions μ defined on the set $\mathcal{S}$. For finite $\mathcal{S}$, we also identify $\mu \in \Delta_{\mathcal{S}}$ with the corresponding probability mass function and will thus write $\mu(s)$ for the probability of set $\{s\} \subset \mathcal{S}$. We will also treat μ as a vector by fixing a canonical ordering of the elements in $\mathcal{S}$. This convention will be used extensively and we will also identify matrices with transition kernels, as usual in the Markov chain literature. For a vector $v \in \mathbb{R}^d$, we use $\|v\|_\infty$, $\|v\|$, and $\|v\|_1$ to denote its ℓ_∞, ℓ_2, and ℓ_1 norms, respectively. For a distribution $\mu \in \Delta_{[d]}$, we define the distribution-weighted norm $\|v\|_\mu^2 = \sum_i^d \mu(i) v[i]^2$. The corresponding 1-norm for functions f is denoted by $\|f\|_{L^1(\mu)} = \int |f(x)|\mu(dx)$. We let $\mathbf{1}$ denote the all-ones vector of an appropriate dimension, and for $f, g : X \to \mathbb{R}$, $\langle f, g\rangle = \sum_x f(x)g(x)$. We use $\log$ to denote the natural logarithm function. For a function $f : X \to \mathbb{R}$ and a function $g : \mathbb{R} \to \mathbb{R}$, $g(f)$ stands for the $X \to \mathbb{R}$ function that maps x to $g(f(x))$ (i.e., g is applied pointwise). We use $a, b \le c$ to denote $a \le c$ and $b \le c$.

2.2. Problem definition

We model the interaction between the agent (i.e. algorithm) and the environment as Markov decision process (MDP). An MDP is a tuple $\langle \mathcal{X}, \mathcal{A}, c, P\rangle$, where $\mathcal{X}$ is a finite state space of cardinality S, $\mathcal{A}$ is a finite action space of cardinality A, $c : \mathcal{X} \times \mathcal{A} \to [0, 1]$ is a cost function, and $P : \mathcal{X} \times \mathcal{A} \to \Delta_\mathcal{X}$ is the transition probability distribution that maps each state-action pair to a distribution over the states.[1] At each time step $t = 1, 2, \dots$, the agent receives the state of the environment $x_t \in \mathcal{X}$, chooses an action $a_t \in \mathcal{A}$, and suffers a cost $c(x_t, a_t)$. The environment then transitions to the next state according to $x_{t+1} \sim P(\cdot|x_t, a_t)$. Initially, the agent does not know P and c. We also assume that x_1, the initial state, is chosen at random from some unknown distribution.

A (stationary Markov) policy is a mapping $\pi : \mathcal{X} \to \Delta_\mathcal{A}$ from a state to a distribution over actions. Following a policy means that in any time step, upon receiving state x, an action $a \in \mathcal{A}$ is chosen with probability $\pi(a|x)$. Let π^* be an unknown baseline policy and let $\{(x_t^*, a_t^*)\}_{t=1,2,\dots}$ denote the state-action sequence that results from following policy π^*. The regret of the algorithm with respect to π^* is defined as

$$\mathfrak{R}_T = \sum_{t=1}^T c(x_t, a_t) - \sum_{t=1}^T c(x_t^*, a_t^*) \,. \tag{1}$$

Our goal is to design a learning algorithm that guarantees a small regret with high probability.

2.3. Value functions

We will assume that the following holds throughout:

Assumption A1 ***(Single recurrent class)*** The states of the MDP under any policy form a single recurrent class.

[1] To simplify the presentation, we consider the finite state MDPs, but our arguments can be extended to infinite and continuous MDPs under appropriate conditions.

MDPs satisfying this condition are also known as *unichain* MDPs (Section 8.3.1 Puterman, 1994).

Under Assumption A1, the states under π form a Markov chain that has a unique stationary distribution over the states, denoted by μ_π. The same holds for the state-action pairs under π, whose stationary distribution we denote by ν_π. Note that ν_π satisfies $\nu_\pi(x, a) = \mu_\pi(x)\pi(a|x)$ for any (x, a). Let $\{(x_t^\pi, a_t^\pi)\}_{t=1,2,\dots}$ be the sequence of state-action pairs that result from following policy π. The average cost of a policy π is defined as

$$\lambda_\pi := \lim_{T\to\infty} \mathbf{E}\left[\frac{1}{T}\sum_{t=1}^T c(x_t^\pi, a_t^\pi)\right] .$$

The corresponding bias (or value) function associated with a stationary policy π is given by

$$V_\pi(x) := \lim_{T\to\infty} \mathbf{E}\left[\sum_{t=1}^T (c(x_t^\pi, a_t^\pi) - \lambda_\pi) \mid x_1^\pi = x\right]$$

when the state-chain is aperiodic, and is the Cesaro-limit, $\lim_{T\to\infty} \mathbf{E}\left[\frac{1}{T}\sum_{m=1}^T \sum_{t=1}^m (c(x_t^\pi, a_t^\pi) - \lambda_\pi) \mid x_1^\pi = x\right]$, otherwise. The state-action value function of a policy corresponds to the value of taking an action a in state x and then following the policy, and is given by

$$Q_\pi(x, a) = c(x, a) - \lambda_\pi + \mathbf{E}[V_\pi(x')|x, a] \,, \tag{2}$$

where $x' \sim P(\cdot|x, a)$. Define the $SA \times SA$ transition matrix H_π by $(H_\pi)_{(x,a),(x',a')} = P(x'|x, a)\pi(a'|x')$. For convenience, we will interchangeably write $H_\pi(x', a'|x, a)$ to denote this value. We will also do this for other transition matrices/kernels. Under our assumption, up to addition of a scalar multiple of $\mathbf{1}$, Q_π (viewed as a vector) is the unique solution to the Bellman equation

$$Q_\pi = c - \lambda_\pi \mathbf{1} + H_\pi Q_\pi \,. \tag{3}$$

Any *Boltzmann policy* $\pi(a|x) \propto \exp(-\eta Q(x, a))$ is invariant to shifting Q by a constant.

3. The POLITEX algorithm

Our proposed algorithm, POLITEX (POLicy ITeration with EXpert advice) is shown in Algorithm listing 1. In each phase i, POLITEX executes policy π_i and at the end of the phase computes an estimate $\widehat{Q}_i$ of Q_{π_i}, the state-action value function of π_i (a policy evaluation step). The next policy is a Boltzmann distribution over the sum of all past state-action value estimates:

$$\pi_{i+1}(a|x) \propto \exp\left(-\eta \sum_{j=1}^i \widehat{Q}_j(x, a)\right) , \tag{4}$$

where $\eta > 0$ is a learning rate. Thus, POLITEX can also be viewed as a "softened" and "averaged" version of policy

Algorithm 1 POLITEX: POLicy ITeration using EXperts

Input: phase length $\tau > 0$, initial state x_0

Set $\widehat{Q}_0(x,a) = 0 \ \forall x, a$
for $i := 1, 2, \dots,$ **do**
Set $\pi_i(a|x) \propto \exp\left(-\eta \sum_{j=0}^{i-1} \widehat{Q}_j(x,a)\right)$
Execute π_i for τ time steps and collect data

$$\mathcal{Z}_i = \{(x_t, a_t, c_t, x_{t+1})\}_{t=\tau(i-1)+1}^{\tau i}$$

Compute $\widehat{Q}_i$ from $\mathcal{Z}_1, \dots, \mathcal{Z}_i, \pi_1, \dots, \pi_i$
end for

iteration. Intuitively, averaging reduces noise, allowing for noisier estimates $(\widehat{Q}_j)_j$ and thus switching to a new policy faster, while the exponential weighting increases robustness. The choice of the Boltzmann policy is not arbitrary: The motivation will become clear in the regret analysis, where, following Even-Dar et al. (2009), we connect learning in MDPs to online learning. Based on its form, POLITEX can be thought of as a generalization of the MDP-E algorithm of Even-Dar et al. (2009).

We leave the choice of how $\widehat{Q}_i$ is estimated to the user (thus, POLITEX is better viewed as a learning schema). We expect longer phase lengths τ to lead to better estimates, and we make this more precise in the next section. While Algorithm 1 suggests that all data should be collected and stored, this is only for the sake of being clear which data can be used to estimate $\widehat{Q}_i$. In practice, one may use any incremental algorithms (e.g., TD-learning of Sutton (1988)). Similarly, one may use either on-policy methods (thus, restricting the data used to estimate $\widehat{Q}_i$ to $\mathcal{Z}_i$), or off-policy methods (using all past data).

Eq. (4) suggests that the computation cost grows over time with the number of past phases. With linear function approximators of the form $\widehat{Q}_j(x,a) = \langle \psi(x,a), \widehat{w}_j \rangle$, this can be avoided by noting that $\sum_{j=0}^{i-1} \widehat{Q}_j(x,a) = \langle \psi(x,a), \sum_{j=0}^{i} \widehat{w}_j \rangle$. With other forms of function approximators, one can either truncate the sum (removing all but the last, say, p) terms, or use approximator-specific solutions. One possibility is to directly estimate π_i based on π_{i-1}, exploiting that $\pi_i(x,a) \propto \pi_{i-1}(x,a) \exp(-\eta \widehat{Q}_{i-1}(x,a))$, which makes MPO (Abdolmaleki et al., 2018) and Quinoa (Degrave et al., 2018) a special case of POLITEX.

4. The regret of POLITEX: General results

Consider any learning agent that produces the state-action sequence $\{(x_t, a_t)\}_{t=1,2,\dots}$ while interacting with an MDP. For a fixed time step t, let $\pi_{(t)}$ denote the policy that is used to generate a_t: Thus, $\pi_{(t)}$ depends on the past observations of the learner, and a_t has distribution $\pi_{(t)}(\cdot|x_t)$ given the past observations up to time step $t-1$ and x_t. Then,

$$\mathfrak{R}_T = \overline{\mathfrak{R}}_T + V_T + W_T \,, \quad \text{where} \tag{5a}$$

$$\overline{\mathfrak{R}}_T = \sum_{t=1}^{T} (\lambda_{\pi_{(t)}} - \lambda_{\pi^*}) \,, \tag{5b}$$

$$V_T = \sum_{t=1}^{T} (c(x_t, a_t) - \lambda_{\pi_{(t)}}) \,, \tag{5c}$$

$$W_T = \sum_{t=1}^{T} (\lambda_{\pi^*} - c(x_t^*, a_t^*)) \,. \tag{5d}$$

In line with the literature, we call $\overline{\mathfrak{R}}_T$ the *pseudo-regret*. This term depends on the difference between the average cost of the followed policies and that of the reference policy, and it can be viewed as a noise-reduced measure of how good the policies are compared to the reference. The terms V_T and W_T capture the deviations of the individual costs from their long-term averages. If the policies are changing slowly, or they are kept fixed for extended periods of time, we expect V_T and W_T to capture the noise in the costs.

Our first result is a bound on $\overline{\mathfrak{R}}_T$. For a probability distribution μ on $\mathcal{X}$ and a stochastic policy π, define $\mu \otimes \pi$ to be the distribution on $\mathcal{X} \times \mathcal{A}$ that puts the probability mass $\mu(x)\pi(a|x)$ on pair $(x,a) \in \mathcal{X} \times \mathcal{A}$. Recall also that μ^* is the stationary distribution of π^* over the states, while $\nu^* = \mu^* \otimes \pi^*$ is the same over the state-action pairs.

Theorem 4.1. *Let $E = \lfloor T/\tau \rfloor$ and fix $0 < \delta < 1$. Let $\varepsilon(\delta, \tau) > 0$ and $Q_{\max} > 0$ and $b \in \mathbb{R}$ be such that for any $i \in [E]$, with probability $1 - \delta$,*

$$\|Q_{\pi_i} - \widehat{Q}_i\|_{L^1(\nu^*)}, \, \|Q_{\pi_i} - \widehat{Q}_i\|_{L^1(\mu^* \otimes \pi_i)} \le \varepsilon(\delta, \tau) \tag{6}$$

and $\widehat{Q}_i(x,a) \in [b, b + Q_{\max}]$ for any $(x,a) \in \mathcal{X} \times \mathcal{A}$. Letting $\eta = \sqrt{8 \log(A)/E}/Q_{\max}$, with probability $1 - \delta$, the regret of POLITEX relative to the reference policy π^* satisfies

$$\overline{\mathfrak{R}}_T \le 2T\, \varepsilon(\delta/(2E), \tau) + E^{1/2} \tau Q_{\max} S_\delta(A, \mu^*) \,,$$

where

$$S_\delta(A, \mu^*) = \sqrt{\frac{\log(A)}{2}} + \left\langle \mu^*, \sqrt{\frac{\log(2/\delta) + \log(1/\mu^*)}{2}} \right\rangle .$$

Note that in the definition of S_δ, $0 \log(1/0)$ is interpreted as zero. In general, we expect $\varepsilon(\delta, \tau)$ to increase as δ decreases to 0 and decrease as τ increases. Thus, increasing τ (the length of the phases) decreases the first term, but because $E \approx T/\tau$, it increases the other terms.

The proof requires a result on the so-called prediction with expert advice problem (hence, explaining the letters 'EX' in POLITEX). In this problem setting, a learner and an environment interact sequentially for T rounds as follows. At the beginning of round $t = 1, 2, \ldots, T$, the environment picks a loss vector $\ell_t \in [0, 1]^A$, while simultaneously the learner picks an expert index $I_t \in [A]$. Both can use their respective past information in their choices. After both made their choices, the learner observes the loss vector ℓ_t, while the environment observes I_t. The goal of the learner is to minimize its regret $\mathfrak{R}_{T,j}$ relative to some choice $j \in [A]$, where $\mathfrak{R}_{T,j} = \sum_{t=1}^T \ell_{t,I_t} - \sum_{t=1}^T \ell_{t,j}$. Since both are allowed to randomize, the regret is random.

The so-called exponentially weighted average (EWA) forecaster in round t chooses $I_t = i$ with probability $\pi_t(i) \propto \exp(-\eta \sum_{s=1}^{t-1} \ell_{s,i})$. Note that this is just the Boltzmann policy in RL parlance. The parameter η is called the learning rate of EWA.

Theorem 4.2 (Corollary 4.2, Cesa-Bianchi & Lugosi (2006))**.** *Set $\eta = \sqrt{8 \log(A)/T}$. Then, regardless of how the environment plays, for any $0 < \delta < 1$ and $i \in [A]$, with probability $1 - \delta$, the regret of EWA with the above choice of η satisfies $\mathfrak{R}_{T,i} \leq \sqrt{T \log(A)/2} + \sqrt{T \log(1/\delta)/2}$.*

We will need a slightly modified version of this result when performance is measured using the "pseudo-regret": $\overline{\mathfrak{R}}_{T,\pi^*} = \sum_{t=1}^T \langle \pi_t, \ell_t \rangle - \sum_{t=1}^T \langle \pi^*, \ell_t \rangle$, where π_t is the distribution over $[A]$ chosen by EWA as described above and π^* is an arbitrary distribution over $[A]$. By checking the proof, it is easy to see that the theorem holds when $\mathfrak{R}_{T,i}$ is replaced by $\overline{\mathfrak{R}}_{T,\pi^*}$. With this, we are ready to present the proof.

Proof of Theorem 4.1. For $Q : \mathcal{X} \times \mathcal{A} \to \mathbb{R}$ and policy π, let $Q(x, \pi) = \sum_a \pi(a|x) Q(x, a)$. By the performance difference formula (Cao, 1999, Eq. (10)),

$$\lambda_{\pi_{(t)}} - \lambda_{\pi^*} = \langle \mu_{\pi^*}, Q_{\pi_{(t)}}(\,\cdot\,, \pi_{(t)}) - Q_{\pi_{(t)}}(\,\cdot\,, \pi^*) \rangle \,.$$

By adding and subtracting empirical estimates, we write $\overline{\mathfrak{R}}_T = \overline{\mathfrak{R}}_{T,1} + \overline{\mathfrak{R}}_{T,2}$, where

$$\begin{aligned}
\overline{\mathfrak{R}}_{T,1} &= \sum_{t=1}^T \langle \mu_{\pi^*}, \widehat{Q}_{\pi_{(t)}}(\,\cdot\,, \pi_{(t)}) - \widehat{Q}_{\pi_{(t)}}(\,\cdot\,, \pi^*) \rangle \\
\overline{\mathfrak{R}}_{T,2} &= \sum_{t=1}^T \langle \mu_{\pi^*}, Q_{\pi_{(t)}}(\,\cdot\,, \pi_{(t)}) - \widehat{Q}_{\pi_{(t)}}(\,\cdot\,, \pi_{(t)}) \rangle \\
&\quad + \sum_{t=1}^T \langle \mu_{\pi^*}, \widehat{Q}_{\pi_{(t)}}(\,\cdot\,, \pi^*) - Q_{\pi_{(t)}}(\,\cdot\,, \pi^*) \rangle \,.
\end{aligned}$$

Note that for any $t \in [T]$ and $i \in [E]$ such that $t \in \{\tau(i-1)+1, \ldots, \tau i\}$, $\pi_{(t)} = \pi_i$. Thus, by Eq. (6), with probability $1 - E\delta$, for all $t \in [T]$,

$$\|Q_{\pi_{(t)}} - \widehat{Q}_{\pi_{(t)}}\|_{L^1(\nu^*)}, \|Q_{\pi_{(t)}} - \widehat{Q}_{\pi_{(t)}}\|_{L^1(\mu^* \otimes \pi_{(t)})} \leq \varepsilon(\delta, \tau)$$

and thus $\overline{\mathfrak{R}}_{T,2} \leq 2T\varepsilon(\delta, \tau)$ on the same event when the previous inequality holds.

Fix a state $x \in \mathcal{X}$ and assume that $\mathcal{A} = \{1, \ldots, A\}$. If we set $\eta = \sqrt{8 \log(A)/T}/Q_{\max}$ in POLITEX, then $\pi_{(i)}(\cdot|x)$ becomes the distribution that would be chosen by the EWA forecaster with environment losses $\ell_{i,a} = (\widehat{Q}_i(x, a) - b)/Q_{\max}$, $a \in \mathcal{A}$, and EWA stepsize η as in Theorem 4.2. Thus, by the note after Theorem 4.2,

$$\begin{aligned}
\widetilde{\mathfrak{R}}_T(x) &:= \sum_{i=1}^E \langle \pi_i(\cdot|x), \ell_i \rangle - \langle \pi^*(\cdot|x), \ell_i \rangle \\
&\leq \sqrt{E \log(A)/2} + \sqrt{E \log(1/\delta)/2}.
\end{aligned}$$

By a union bound, simultaneously for all $x \in \mathcal{X}$ such that $\mu^*(x) > 0$,

$$\begin{aligned}
\widetilde{\mathfrak{R}}_T(x) &\leq \sqrt{E \log(A)/2} + \sqrt{E \log(1/\mu^*(x)\delta)/2}\,, \\
\langle \mu^*, \widetilde{\mathfrak{R}}_T \rangle &\leq \sqrt{E \log(A)/2} + \langle \mu^*, \sqrt{E \log(1/\delta\mu^*)/2} \rangle \,.
\end{aligned}$$

Noting that $\overline{\mathfrak{R}}_{T,1} \leq \tau Q_{\max} \langle \mu^*, \widetilde{\mathfrak{R}}_T \rangle$ and using another union bound gives the final result. □

Theorem 4.1 immediately implies the following corollary which discusses the performance of POLITEX in cases where the value functions can be computed exactly. The proof is in Appendix C.

Corollary 4.3. *When we can compute value functions exactly (known MDP), after n policy updates with POLITEX with parameters $\tau = 1$ and $E = n$, a policy selected uniformly at random from policies computed so far is $O(Q_{\max}\sqrt{\log(A)/n})$ close to the optimal average cost.*

To get a regret bound for POLITEX, we also need a bound on V_T and W_T. We bound these terms under the assumption that all policies mix at the same speed. Obviously, this assumption implies Assumption A1.

Assumption A2 ***(Uniformly fast mixing)*** There exists a constant $\kappa > 0$ such that for any distribution ν',

$$\sup_\pi \|(\nu_\pi - \nu')^\top H_\pi\|_1 \leq \exp(-1/\kappa) \|\nu_\pi - \nu'\|_1 \,.$$

Under this assumption, the following bound holds:

Lemma 4.4. *Let Assumption A2 hold. With probability at least $1 - \delta$, we have that*

$$\begin{aligned}
|W_T| &\leq \kappa + 4\kappa\sqrt{2T \log{(2/\delta)}}\,, \\
|V_T| &\leq E\kappa + 4E\kappa\sqrt{2\tau \log{(2T/\delta)}}.
\end{aligned}$$

The proof can be found in Appendix A. While the size of V_T is independent of the algorithm, decreasing E (longer, fewer phases) will decrease W_T. Putting everything together, we get the main result of the paper:

Theorem 4.5. *Let the assumptions of Theorem 4.1 hold and in addition let Assumption A2 hold. Then, with probability* $1-2\delta$*,*

$$\mathfrak{R}_T \le (E+1)\kappa + 2T\,\varepsilon\left(\tfrac{\delta}{2E},\tau\right) + E^{\frac{1}{2}}\tau Q_{\max} S_\delta(A,\mu^*) + 4T^{1/2}\kappa\sqrt{2\log(2/\delta)} + 4E\tau^{1/2}\kappa\sqrt{2\log(2T/\delta)}\,,$$

where $S_\delta(A,\mu^*)$ *is defined in Theorem 4.1.*

In the next section, we will see that in a special case we can choose

$$\varepsilon(\delta,\tau) = \varepsilon_0 + C\sqrt{\frac{\log(1/\delta)}{\tau}} \tag{7}$$

for some $\varepsilon_0 > 0$ that captures the irreducible error of the value estimation method ("approximation error") and $C > 0$ is some constant. In this case, balancing the various terms gives the following result:

Corollary 4.6. *Let the conditions postulated in Theorem 4.5 hold. In addition, assume that Eq. (7) holds. Then, for some universal constant* $C' > 0$*, for* T *larger than another universal constant, for any* $\delta \in (0,1)$*, with probability at least* $1-\delta$*,*

$$\mathfrak{R}_T \le C' T^{3/4}\cdot\left(Q_{\max} S_\delta(A,\mu^*) + (\kappa + C)\sqrt{\log(T/\delta)}\right) + 2\varepsilon_0 T\,.$$

The proof can be found in Appendix B, where a more precise expression is also given which is valid for any $T > 1$.

5. POLITEX with linear value function approximation

In this section we consider POLITEX with linear value function approximation, i.e., when the action-value function approximation is of the form $(x,a) \mapsto \langle \psi(x,a), w\rangle$, where $\psi : \mathcal{X}\times\mathcal{A} \to \mathbb{R}^d$ is a fixed map chosen by the user, and $w \in \mathbb{R}^d$ is a weight vector to be learned from data. We consider this case to uncover conditions (e.g., on ψ) under which POLITEX can perform well while using a nontrivial function approximation technique. The estimation method we use is the so-called "least-squares policy evaluation" (LSPE) method by Bertsekas & Ioffe (1996). LSPE can be seen as solving a sample-based approximation to a projected version of the Bellman equation underlying the policy to be evaluated, as detailed in the next subsection. The version of LSPE we analyze is given in Section 5.2. Unlike the version studied by Yu & Bertsekas (2009), we consider a non-incremental, "batch" version of LSPE, which is easier to analyze and may be more sample-efficient than the incremental LSPE method that processes all examples (transitions) once.

5.1. Projected Bellman equation

Given the map ψ as above, we can form the $SA\times d$ matrix Ψ whose rows correspond to $\psi(x,a)^\top$ for a canonical ordering of the state-action pairs.

Our analysis will consider weighted value function errors $\|\Psi w - Q_\pi\|_{\nu_\pi}$, where ν_π is a distribution over state-action pairs. Let $\mathrm{span}(\Psi) = \{\Psi w : w \in \mathbb{R}^d\}$ denote the span of Ψ. We make the following assumption on Ψ:

Assumption A3 ***(Linearly independent features)*** The columns of the matrix $[\Psi\ \mathbf{1}]$ are linearly independent.

This assumption is necessary as we are solving the average-cost Bellman equations. Let $\Pi_\pi : \mathbb{R}^{SA} \to R^{SA}$ be the projection defined by $\Pi_\pi x = \mathrm{argmin}_{y\in\mathrm{span}(\Psi)} \|y-x\|^2_{\nu_\pi}$. As is well-known, Π_π is a linear operator which, in matrix form, can be written as $\Pi_\pi = \Psi(\Psi^\top D_\pi \Psi)^{-1}\Psi^\top D_\pi$ where D_π is the $(SA)\times(SA)$ diagonal matrix with ν_π on its diagonal. As mentioned, LSPE aims at finding a solution to the so-called projected Bellman equation (PBE),

$$\widetilde{Q}_\pi = \Pi_\pi(c - \lambda_\pi \mathbf{1} + H_\pi \widetilde{Q}_\pi)\,. \tag{8}$$

Clearly, any solution to this equation lies in $\mathrm{span}(\Psi)$. Under Assumptions A1 and A3, Prop. 4 of Yu & Bertsekas (2009) and a simple argument show that this equation has a unique solution. Under Assumption A3, this gives rise to a unique weight vector $\widetilde{w}_\pi$ so that $\widetilde{Q}_\pi = \Psi\widetilde{w}_\pi$. It also follows that $I - \Pi_\pi H_\pi$ is nonsingular. Let $\rho = \|(I-\Pi_\pi H_\pi)^{-1}\Pi_\pi H_\pi(I-\Pi_\pi)\|^2_{\nu_\pi}$. Theorem 2.2 of Yu & Bertsekas (2010) shows that if ρ is not too large, solving Eq. (8) is reasonable in the sense that one loses at most by a multiplicative factor of $\sqrt{1+\rho}$:

$$\|Q_\pi - \widetilde{Q}_\pi\|_{\nu_\pi} \le \sqrt{1+\rho}\min_{Q\in\mathrm{span}(\Psi)}\|Q_\pi - Q\|_{\nu_\pi}\,. \tag{9}$$

5.2. The LSPE algorithm

The LSPE algorithm is derived from the damped version of the projected Bellman equation. For a damping factor γ, this takes the form

$$\widetilde{Q}_\pi = (1-\gamma)\widetilde{Q}_\pi + \gamma\Pi_\pi(c - \lambda_\pi + H_\pi\widetilde{Q}_\pi)\,, \tag{10}$$

which can also be read as an update rule, where the left-hand side is assigned to the right-hand side in an iterative fashion. Let $\psi_i = \psi(x_i,a_i)$, $a_i \sim \pi(\cdot|x_i)$, $x_{i+1} \sim P(\cdot|x_i,a_i)$, $c_i = c(x_i,a_i)$, $i = 1,\dots,\tau$ be the data available from executing policy π. Let $\widehat{\lambda}_\pi = \frac{1}{\tau}\sum_{i=1}^\tau c_i$ be the empirical average cost. The (batch) LSPE update rule underlying Eq. (10) in weight-space is as follows:

$$w^{(k+1)} = (1-\gamma)w^{(k)} + \gamma\Big(\sum_{i=1}^\tau \psi_i\psi_i^\top\Big)^{-1}\sum_{j=1}^\tau \psi_j\big({\psi_{j+1}}^\top w^{(k)} + c_j - \widehat{\lambda}_\pi\big)\,, \tag{11}$$

where $k = 0, 1, \ldots$ and (say) $w^{(0)} = 0$.

In Appendix D, we prove the following bound on the estimation error $\|\Psi w^{(k)} - \widetilde{Q}_\pi\|_{\nu_\pi} = \|\Psi w^{(k)} - \Psi \widetilde{w}\|_{\nu_\pi}$:

Theorem 5.1. *Let $\sigma = \lambda_{\min}(\Psi^\top D_\pi \Psi)$. With probability at least $1-\delta$, for a problem-specific constant $\alpha \in (0,1)$ and universal constant $C > 0$, for τ big enough, for any $k > 0$, we have*

$$\|\Psi(w^{(k)} - \widetilde{w})\|_{\nu_\pi} \le \alpha^{k-1}\|\Psi(w^{(0)} - \widetilde{w})\|_{\nu_\pi} + \frac{C\|\widetilde{w}\|\sqrt{d}\,\kappa\,\mathrm{polylog}(d, k/\delta)}{\sqrt{\tau}(1-\alpha)\sigma^3}\,.$$

The proof relies in part on the contractiveness of the operator $F_{\pi,\gamma} := (1-\gamma)I + \gamma\Pi_\pi H_\pi$. In particular, by Prop. 4 of Yu & Bertsekas (2009), under assumptions A1 and A3, for any $\gamma \in (0,1)$ there exists $\alpha_\pi \in [0,1)$ such that $\|F_\gamma\|_{\nu_\pi} \le \alpha_\pi$. To get a policy-independent but MDP specific constant, we let $\alpha := \sup_\pi \alpha_\pi$.

Since we have no control of what policies POLITEX will come up with, we make the following assumption:

Assumption A4 ***(Uniformly excited features)*** There exists a positive real σ such that for any policy π, $\lambda_{\min}(\Psi^\top D_\pi \Psi) \ge \sigma$.

The assumption states that any policy generates a stationary distribution under which the features span all directions in the feature space uniformly well. This condition is a close relative of the condition of persistent excitation from the control literature (Narendra & Annaswamy, 1987). Assumption A4 can be restrictive. For example, it is violated by deterministic policies when learning with finite MDPs with a tabular representation (i.e., when a separate feature is assigned to each state-action pair). Appendix F outlines a way to relax this assumption, so that in the tabular case we only require policies to visit every state rather than every state-action pair.

5.3. POLITEX with LSPE

Let us now return to the problem of bounding the regret of POLITEX when used with LSPE and linear function approximation. By Eq. (6), in order to state a result for POLITEX, we need to control $\|Q_{\pi_i} - \widehat{Q}_i\|_{L^1(\nu^*)}$ and $\|Q_{\pi_i} - \widehat{Q}_i\|_{L^1(\mu^*\otimes\pi_i)}$. We show the bound for the second term; the bound for the first one can be obtained in an entirely analogous fashion. First, note that for any policy π,

$$\|Q_{\pi_i} - \widehat{Q}_i\|_{L^1(\mu^*\otimes\pi)} \le \|Q_{\pi_i} - \widetilde{Q}_i\|_{L^1(\mu^*\otimes\pi)} + \|\widetilde{Q}_i - \widehat{Q}_i\|_{L^1(\mu^*\otimes\pi)}\,.$$

Here, the first term is the irreducible approximation error due to solving the projected Bellman equation induced by π_i using the feature-map ψ. To control the second term, note that for any w, $\|\Psi(w - \widetilde{w})\|^2_{\nu_\pi} \ge \sigma^{-1}\|w - \widetilde{w}\|^2$. Thus,

$$\|\Psi(w - \widetilde{w})\|^2_{L^1(\mu^*\otimes\pi)} \le C_\Psi^2\|w - \widetilde{w}\|^2 \le \sigma C_\Psi^2 \|\Psi(w - \widetilde{w})\|^2_{\nu_\pi},$$

where $C_\Psi = \max_{x,a}\|\psi(x,a)\|$. Combining this with Theorem 5.1 gives a bound as required by Eq. (6).

To bound $Q_{\max}$, we show in Appendix D.3 that for any $C' \ge 2 + \frac{2C_\Psi \alpha}{\sigma}$, and for some constants C_3 and C_4 and for $\tau \ge ((C_3/C_2 + C'C_4)/(1-\alpha))^2$, with high probability, for any i the LSPE estimate satisfies $\|w^{(i)}\| \le C'C_2$. Here, C_2 is a constant such that $\|\widetilde{w}\| \le C_2$, $\|w^{(0)}\| \le C_2$ for all policies. Along with feature boundedness, this gives a bound on $Q_{\max}$. Thus we have the following theorem.

Theorem 5.2. *Under Assumptions A1, A3, A2, and A4, for some constant C, for any fixed T large enough, the regret of POLITEX with LSPE with respect to a reference policy π^* is bounded with probability at least $1-\delta$ as*

$$\mathfrak{R}_T \le 2\varepsilon_0 T + CT^{3/4}\cdot\Big(\sqrt{2\log A} + \kappa\sqrt{\log(4T/\delta)} + \langle\mu^*, \sqrt{\log(1/\delta\mu^*)/2}\rangle + \frac{\kappa}{\sigma^3}\sqrt{d\log(4d/\delta)}\Big)\,.$$

6. Experiments

6.1. Queueing networks

We first study the performance of POLITEX with linear function approximation on the 4-dimensional and 8-dimensional queueing network problems described in de Farias & Van Roy (2003) (Figures 6 and 7). A queueing network contains K servers, each server s has N_s queues, and can process at most one queue at each time step. If a queue q is selected for processing, one of its jobs departs to the next queue (or out of the system) with probability d_q. A subset of the queues receive new jobs with probability α at each time step. The system state is a vector x_t with the number of jobs in each queue, and the cost is $\|x_t\|_1$. A policy selects a non-empty queue for each server to process.

We use the similar features as de Farias & Van Roy (2003), including bias, state x, binary indicator features for each pair and triplet of queues, indicating that all are non-empty. We learn separate weights on these features for each action using LSPE, for which we regularize the projection matrix as $\Pi_{\pi,\beta} = \Psi^\top(\Psi^\top D_\pi\Psi + \beta I)^{-1}\Psi^\top D_\pi$. We compare POLITEX to the following: (1) Standard least-squares policy iteration (LSPI), where the policy π_i in phase i a Boltzmann distribution using the most recent value function estimate $\widehat{Q}_{i-1}$. We use the same estimation procedure as for POLITEX. (2) A version of RLSVI (Osband et al., 2017) where we randomize the value function parameters. In particular, we update the posterior mean and covariance of a regression-based estimate of parameters w after each step. At the beginning of each phase i, we sample parameters $\overline{w}_i$ from this posterior, and act greedily with respect to $Q_i = \Psi\overline{w}_i$.

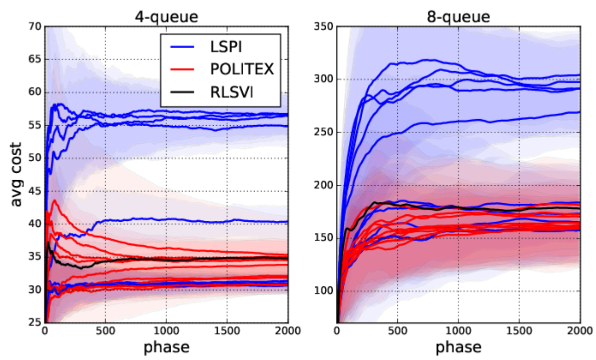

Figure 1. Average cost at the end of each phase for the 4-queue and 8-queue environments (mean and standard deviation of 50 runs), for different values of η.

We initialize to empty queues and run policies for $E = 2000$ phases of length $\tau = E$. For all policies, we bias the covariance of the value functions with $\beta = 0.1$. For LSPI and POLITEX, experiment with $\eta = k/\sqrt{T}$, for $k \in \{1, 5, 10, 20, 100, 500, 1000, 2000, 4000\}$; the value $k = 1$ was best. Fig. 1 shows the running average cost for each policy. For the best choice of η, POLITEX and LSPI achieve similar performance, and slightly outperform RLSVI. LSPI performance deteriorates for higher values of η (corresponding to greedier policies with less exploration), suggesting that it is more sensitive to value estimation error.

6.2. Atari with deep neural networks

In the next experiment, we examine whether the promising theoretical results presented in this paper lead to a practical algorithm when applied in the context of neural networks. We compare a version of POLITEX to DQN (Mnih et al., 2013) on a standard Atari environment running Ms Pacman. We approximate state-action value functions using neural networks and use SOLO FTRL by Orabona & Pál (2015) for tuning the learning rate η, which makes POLITEX adapt to the range of the action-value function estimates. We execute policies that are based only on the most recent n neural networks, where n is a parameter to be chosen. Further implementation details are given in Appendix G. Rather than evaluating learned policies after training (as in (Mnih et al., 2013)), in line with the setting of this paper, we plot the rewards obtained by the agents against the total number of frames of gameplay. Fig. 2 shows that POLITEX achieves higher game scores with seemingly more stable learning than DQN, indicating the viability of the method. Note that we expect POLITEX to be more stable (due to the averaging that it uses), and we speculate that the higher scores are a consequence of this stability.

7. Discussion and future work

Policy iteration: We highlight an advantage of our approach compared to existing results for approximate PI methods. Let π_k be the policy generated by a PI method after k iterations. Lazaric et al. (2012) bound the performance error $\|V_{\pi_*} - V_{\pi_k}\|_\mu$ for the LSPI algorithm, where μ is a distribution over the states. The bound involves a term known as the *concentrability coefficient*, which can be very large if μ is far from the distribution used to generate the samples at each iteration, and is present even in the realizable case. This term is an artifact of contraction-based arguments for analyzing policy iteration methods. Since our regret-based analysis avoids contraction arguments, such terms do not appear in our performance bounds. We emphasize that a regret objective is fundamentally very different than the above weighted difference between V_{π_*} and V_{π_k}.

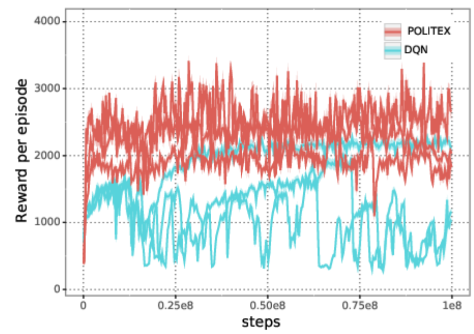

Figure 2. Ms Pacman game scores obtained by the agents at the end of each game. The plots are based on three runs of each algorithm with different random seeds.

Nonlinear value functions. POLITEX produces a policy in each phase based on all past value functions. With linear function approximation, once can keep simply average the weight vectors obtained in each phase. However, with nonlinear functions such as deep neural networks, we may need to store all past value functions, which might be memory intensive. In practice, we can only maintain a fixed number networks, as in our Atari experiment, or learn features in the first phase and only update the weights of the last layer in the subsequent phases, similarly to Levine et al. (2017). An alternative is to keep an explicit function approximator to represent policies, similarly to the concurrent work of Abdolmaleki et al. (2018) or Degrave et al. (2018).

Contributions and future work. We have presented POLITEX, a practical model-free algorithm with function approximation for continuing RL problems. Under a uniform mixing assumption and with linear function approximation, the regret of POLITEX scales as $\tilde{O}(d^{1/2}T^{3/4} + \varepsilon_0 T)$. We have also provided a new finite-sample analysis of the LSPE algorithm. Preliminary experimental results demonstrate the effectiveness of POLITEX both with linear and neural function approximation. Future work may include relaxing assumptions and finding computationally efficient ways to use POLITEX with non-linear function approximators.

8. Acknowledgments

We thank Jonas Degrave and Abbas Abdolmaleki for helpful discussions.

References

Abbasi-Yadkori, Y. *Online Learning for Linearly Parametrized Control Problems*. PhD thesis, University of Alberta, 2012.

Abbasi-Yadkori, Y. and Szepesvári, C. Regret bounds for the adaptive control of linear quadratic systems. In *COLT*, 2011.

Abbasi-Yadkori, Y., Bartlett, P., and Kanade, V. Tracking adversarial targets. In *ICML*, 2014.

Abbasi-Yadkori, Y., Lazic, N., and Szepesvári, C. Model-free linear quadratic control via reduction to expert prediction. In *AISTATS*, 2019.

Abdolmaleki, A., Springenberg, J. T., Tassa, Y., Munos, R., Heess, N., and Riedmiller, M. Maximum a posteriori policy optimisation. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=S1ANxQW0b.

Antos, A., Szepesvári, C., and Munos, R. Learning near-optimal policies with Bellman-residual minimization based fitted policy iteration and a single sample path. *Machine Learning*, 71(1):89–129, 2008.

Asis, K. D., Hernandez-Garcia, J. F., Holland, G. Z., and Sutton, R. S. Multi-step reinforcement learning: A unifying algorithm. *CoRR*, abs/1703.01327, 2017. URL http://arxiv.org/abs/1703.01327.

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

Azar, M. G., Osband, I., and Munos, R. Minimax regret bounds for reinforcement learning. In *ICML*, pp. 263–272, 2017.

Bartlett, P. L. and Tewari, A. Regal: A regularization based algorithm for reinforcement learning in weakly communicating MDPs. In *UAI*, 2009.

Bellemare, M., Srinivasan, S., Ostrovski, G., Schaul, T., Saxton, D., and Munos, R. Unifying count-based exploration and intrinsic motivation. In *NeurIPS*, 2016.

Bertsekas, D. P. Approximate policy iteration: A survey and some new methods. *Journal of Control Theory and Applications*, 9(3):310–335, 2011.

Bertsekas, D. P. and Ioffe, S. Temporal differences-based policy iteration and applications in neuro-dynamic programming. *Lab. for Info. and Decision Systems Report LIDS-P-2349, MIT, Cambridge, MA*, 14, 1996.

Cao, X. R. Single sample path-based optimization of Markov chains. *Journal of optimization theory and applications*, 100(3):527–548, 1999.

Cesa-Bianchi, N. and Lugosi, G. *Prediction, learning, and games*. Cambridge University Press, 2006.

Cho, G. and Meyer, C. Comparison of perturbation bounds for the stationary distribution of a Markov chain. *Linear Algebra and its Applications*, 335:137–150, 2001.

Daumé III, H., Langford, J., and Marcu, D. Search-based structured prediction. *Machine Learning*, 75:297–325, 2009.

de Farias, D. P. and Van Roy, B. The linear programming approach to approximate dynamic programming. *Operations research*, 51(6):850–865, 2003.

Degrave, J., Abdolmaleki, A., Springenberg, J. T., Heess, N., and Riedmiller, M. Quinoa: a Q-function you infer normalized over actions. In *Deep RL Workshop/NeurIPS*, 2018. URL http://tinyurl.com/y39ppt8e.

Even-Dar, E., Kakade, S. M., and Mansour, Y. Online Markov decision processes. *Mathematics of Operations Research*, 34(3):726–736, 2009.

Farahmand, A.-m., Ghavamzadeh, M., Szepesvári, C., and Mannor, S. Regularized policy iteration with nonparametric function spaces. *Journal of Machine Learning Research*, 17(1):4809–4874, 2016.

Fortunato, M., Azar, M. G., Piot, B., Menick, J., Hessel, M., Osband, I., Graves, A., Mnih, V., Munos, R., Hassabis, D., Pietquin, O., Blundell, C., and Legg, S. Noise networks for exploration. In *ICLR*, 2018.

Geist, M. and Scherrer, B. Off-policy learning with eligibility traces: A survey. *Journal of Machine Learning Research*, 15:289–333, 2014.

Horgan, D., Quan, J., Budden, D., Barth-Maron, G., Hessel, M., Van Hasselt, H., and Silver, D. Distributed prioritized experience replay. *arXiv preprint arXiv:1803.00933*, 2018.

Jiang, N., Krishnamurthy, A., Agarwal, A., Langford, J., and Schapire, R. E. Contextual decision processes with low Bellman rank are PAC-learnable. In *ICML*, 2017.

Jin, C., Allen-Zhu, Z., Bubeck, S., and Jordan, M. I. Is Q-learning provably efficient? In *NeurIPS*, pp. 4868–4878, 2018.

Lazaric, A., Ghavamzadeh, M., and Munos, R. Finite-sample analysis of least-squares policy iteration. *Journal of Machine Learning Research*, 13(Oct):3041–3074, 2012.

Levine, N., Zahavy, T., Mankowitz, D. J., Tamar, A., and Mannor, S. Shallow updates for deep reinforcement learning. In *NeurIPS*, pp. 3135–3145, 2017.

Liu, B., Mahadevan, S., and Liu, J. Regularized off-policy TD-learning. In *NeurIPS*, 2012.

Liu, B., Liu, J., Ghavamzadeh, M., Mahadevan, S., and Petrik, M. Finite-sample analysis of proximal gradient TD algorithms. In *UAI*, 2015.

Machado, M. C., Bellemare, M. G., and Bowling, M. A Laplacian framework for option discovery in reinforcement learning. In *ICML*, 2017.

Machado, M. C., Bellemare, M. G., and Bowling, M. Count-based exploration with the successor representation. *arXiv preprint arXiv:1807.11622*, 2018.

Maei, H. R., Szepesvári, C., Bhatnagar, S., and Sutton, R. S. Toward off-policy learning control with function approximation. In *ICML*, 2010.

Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., and Riedmiller, M. Playing Atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540): 529, 2015.

Narendra, K. S. and Annaswamy, A. M. Persistent excitation in adaptive systems. *International Journal of Control*, 45 (1):127–160, 1987.

Neu, G. and Gómez, V. Fast rates for online learning in linearly solvable Markov decision processes. In *COLT*, 2017.

Neu, G., György, A., Szepesvári, C., and Antos, A. Online Markov decision processes under bandit feedback. *IEEE Transactions on Automatic Control*, 59:676–691, 2014.

O'Donoghue, B., Osband, I., Munos, R., and Mnih, V. The uncertainty Bellman equation and exploration. In *ICML*, 2018.

Orabona, F. and Pál, D. Scale-free algorithms for online linear optimization. In *ALT*, pp. 287–301, 2015.

Osband, I. and Van Roy, B. Model-based reinforcement learning and the Eluder dimension. In *NIPS*, 2014.

Osband, I., Wen, Z., and Roy, B. V. Generalization and exploration via randomized value functions. In *ICML*, 2016.

Osband, I., Roy, B. V., Russo, D., and Wen, Z. Deep exploration via randomized value functions. *Journal of Machine Learning Research*, 2017.

Ostrovski, G., Bellemare, M. G., van den Oord, A., and Munos, R. Count-based exploration with neural density models. In *ICML*, 2017.

Puterman, M. *Markov decision processes : Discrete stochastic dynamic programming*. John Wiley & Sons, New York, 1994.

Ross, S., Gordon, G. J., and Bagnell, J. A. A reduction of imitation learning and structured prediction to no-regret online learning. In *AISTATS*, 2011.

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

Schaul, T., Quan, J., Antonoglou, I., and Silver, D. Prioritized experience replay. *arXiv preprint arXiv:1511.05952*, 2015.

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

Strehl, A. L., Li, L., Wiewiora, E., Langford, J., and Littman, M. L. PAC model-free reinforcement learning. In *ICML*, pp. 881–888, 2006.

Sutton, R. S. Learning to predict by the methods of temporal differences. *Machine learning*, 3:9–44, 1988.

Sutton, R. S., Szepesvári, C., and Maei, H. R. A convergent $o(n)$ algorithm for off-policy temporal-difference learning with linear function approximation. In *NeurIPS*, 2009.

Szepesvári, C. *Algorithms for Reinforcement Learning*. Morgan and Claypool, 2010.

Taïga, A. A., Courville, A., and Bellemare, M. G. Approximate exploration through state abstraction. *arXiv preprint arXiv:1808.09819*, 2018.

Tropp, J. A. User-friendly tail bounds for sums of random matrices. *Foundations of computational mathematics*, 12 (4):389–434, 2012.

Tsitsiklis, J. N. and Van Roy, B. An analysis of temporal-difference learning with function approximation. *IEEE Transactions on Automatic Control*, 42:674–690, 1997.

Tsitsiklis, J. N. and Van Roy, B. Average cost temporal-difference learning. *Automatica*, 35:1799–1808, 1999.

Tu, S. and Recht, B. Least-squares temporal difference learning for the linear quadratic regulator. *arXiv preprint arXiv:1712.08642*, 2017.

van Hasselt, H., Guez, A., and Silver, D. Deep reinforcement learning with double q-learning. *CoRR*, abs/1509.06461, 2015. URL http://arxiv.org/abs/1509.06461.

Wang, Z., de Freitas, N., and Lanctot, M. Dueling network architectures for deep reinforcement learning. *CoRR*, abs/1511.06581, 2015. URL http://arxiv.org/abs/1511.06581.

Wen, Z. and Van Roy, B. Efficient reinforcement learning in deterministic systems with value function generalization. *Mathematics of Operations Research*, 2017.

Yu, H. and Bertsekas, D. P. Convergence results for some temporal difference methods based on least squares. *IEEE Transactions on Automatic Control*, 54(7):1515–1531, 2009.

Yu, H. and Bertsekas, D. P. Error bounds for approximations from projected linear equations. *Mathematics of Operations Research*, 35(2):306–329, 2010.

Yu, J. Y., Mannor, S., and Shimkin, N. Markov decision processes with arbitrary reward processes. *Mathematics of Operations Research*, 34(3):737–757, 2009.

A. Proof of Lemma 4.4

In this section we prove Lemma 4.4.

We start with a lemma that will be useful in a later proof as well.

Lemma A.1. *Let Assumption A2 hold and let $\{(x_t, a_t)\}_{t=1}^m$ be the state-action sequence obtained when following a policy π from initial distribution ν_0 (i.e., $(x_0, a_0) \sim \nu_0$) and for $t \in [m]$, let X_t be a binary indicator vector with a non-zero at the linear index of the state-action pair (x_t, a_t). Define for $i = 1, \ldots, m$,*

$$B_i = \mathbf{E}\left[\sum_{t=1}^m X_t | X_1, ..., X_i\right], \quad \text{and} \;\; B_0 = \mathbf{E}\left[\sum_{t=1}^m X_t\right].$$

Then, $(B_i)_{i=0}^m$ is a vector-valued martingale ($\mathbf{E}[B_i - B_{i-1}|B_0, \ldots, B_{i-1}] = 0$ for $i = 1, \ldots, m$) and $\|B_i - B_{i-1}\|_1 \le 4\kappa$ holds for $i \in [m]$.

The martingale constructed in the lemma is known as the Doob martingale underlying the sum $\sum_{t=1}^m X_t$.

Proof. That $(B_i)_i$ is a martingale follows from the definitions. It remains to show that the bound on its increments. Let H be the transition matrix for the state-action pairs under π. Define $X_0 = \nu_0$. Then, for $t = 0, \ldots, m-1$, $\mathbf{E}[X_{t+1}|X_t] = H^\top X_t$ and by the Markov property, for any $i \in [m]$,

$$\begin{aligned}
B_i &= \sum_{t=1}^i X_t + \sum_{t=i+1}^m \mathbf{E}[X_t|X_i] = \sum_{t=1}^i X_t + \sum_{t=1}^{m-i} (H^t)^\top X_i \,, \text{ and} \\
B_0 &= \sum_{t=1}^m (H^t)^\top X_0 \,.
\end{aligned}$$

Fix some $i \in [m]$. We have,

$$\begin{aligned}
B_i - B_{i-1} &= \sum_{t=1}^i X_t - \sum_{t=1}^{i-1} X_t + \sum_{t=1}^{m-i} (H^t)^\top X_i - \sum_{t=1}^{m-i+1} (H^t)^\top X_{i-1} \\
&= \sum_{t=0}^{m-i} (H^t)^\top X_i - \sum_{t=0}^{m-i} (H^t)^\top H X_{i-1} \\
&= \sum_{t=0}^{m-i} (H^t)^\top (X_i - H X_{i-1}) \,.
\end{aligned}$$

Letting ν denote the stationary distribution of π, by the triangle inequality and the uniform mixing assumption, for any $t \ge 0$,

$$\begin{aligned}
\|(H^t)^\top (X_i - H X_{i-1})\|_1 &\le \|(H^t)^\top (X_i - \nu)\|_1 + \|(H^t)^\top (\nu - H X_{i-1}\|_1 \\
&\le \exp(-t/\kappa)(\|X_i - \nu\|_1 + \|\nu - H X_{i-1}\|_1) \\
&\le 4 \exp(-t/\kappa) \,.
\end{aligned}$$

Thus, using another triangle inequality, $\|B_i - B_{i-1}\|_1 \le 2 \sum_{t=0}^\infty \exp(-t/\kappa) \le 4\kappa$. ∎

Proof of Lemma 4.4. We begin with a bound on $W_T = \sum_{t=1}^T (\lambda_{\pi^*} - c_t^*)$. We decompose the term W_T as follows:

$$W_T = \sum_{t=1}^T \lambda_{\pi^*} - c_t^* = \sum_{t=1}^T \lambda_{\pi^*} - \mathbf{E}[c_t^*] + \sum_{t=1}^T \mathbf{E}[c_t^*] - c_t^* = \underbrace{\sum_{t=1}^T (\nu^* - \nu_t^*)^\top c}_{W_{T,1}} + \underbrace{\sum_{t=1}^T \mathbf{E}[c_t^*] - c_t^*}_{W_{T,2}}, \tag{12}$$

where ν^* denotes the stationary distribution of the policy π^* and ν_t^* is the state-action distribution after t time steps and we used that $\mathbf{E}[c_t^*] = \nu_t^\top c$. We bound $W_{T,1}$ in the equation above using the uniform mixing assumption:

$$\left|\sum_{t=1}^{T}(\nu^* - \nu_t^*)^\top c\right| \leq \sum_{t=1}^{T} \|\nu^* - \nu_t^*\|_1 \|c\|_\infty \leq \sum_{t=1}^{T} \exp(-t/\kappa) \leq \kappa. \tag{13}$$

Let $(B_t)_{t=0,\dots,T}$ be the Doob martingale from Lemma A.1 for $(x_t, a_t) = (x_t^*, a_t^*)$ and $\pi = \pi^*$. Then $B_0 = \sum_{t=1}^T \nu_t^*$ and $B_T = \sum_{t=1}^T X_t$ where $X_t(x,a) = \mathbb{I}\{(x_t^*, a_t^*) = (x,a)\}$. It follows from the definitions that $W_{T,2} = \langle B_0 - B_T, c\rangle$. By Lemma A.1, $|\langle B_i - B_{i-1}, c\rangle| \leq \|B_i - B_{i-1}\|_1 \|c\|_\infty \leq 4\kappa$. Hence, by Azuma's inequality, with probability at least $1-\delta$, $W_{T,2} \leq 4\kappa\sqrt{2T\log\left(\frac{2}{\delta}\right)}$. Summing the bounds on $W_{T,1}$ and $W_{T,2}$ gives the desired result.

The bound for V_T follows similarly by noticing that POLITEX makes $E-1$ policy switches and each one of them is executed for τ rounds. Using a union bound over all $E-1$ events, we get with probability at least $1-\delta$,

$$|V_T| \leq E\kappa + 4E\kappa\sqrt{2\tau\log\left(\frac{2T}{\delta}\right)}. \tag{14}$$

□

B. Proof of Corollary 4.6

In this section we prove Corollary 4.6.

By Theorem 4.5, we have

$$\begin{aligned}\mathfrak{R}_T \leq\ & (E+1)\kappa + 2T\,\varepsilon(\delta/(2E), \tau) \\ & + E^{1/2}\tau Q_{\max} S_\delta(A, \mu^*) \\ & + 4\kappa\sqrt{2T\log(2/\delta)} + 4E\kappa\sqrt{2\tau\log(2T/\delta)}\,.\end{aligned}$$

On the other hand, by (7),

$$\varepsilon(\delta, \tau) = \varepsilon_0 + C\sqrt{\frac{\log(1/\delta)}{\tau}}\,.$$

Combining this equation with the previous bound and $T \leq E\tau$ gives

$$\begin{aligned}\mathfrak{R}_T \leq\ & 2T\varepsilon_0 + \kappa + 4\kappa\sqrt{2T\log(2/\delta)} \\ & + E^{1/2}\tau\, Q_{\max} S_\delta(A, \mu^*) \\ & + E\tau^{1/2}\left(\sqrt{(4C)^2\log(2E/\delta)} + \kappa\sqrt{32\log(2T/\delta)}\right) + E\kappa\,.\end{aligned}$$

Choosing $E = \tau = T^{1/2}$ gives

$$\begin{aligned}\mathfrak{R}_T \leq\ & 2T\varepsilon_0 \\ & + T^{3/4}\left(Q_{\max} S_\delta(A, \mu^*) + \left(4C + 4\sqrt{2}\kappa\right)\sqrt{\log(2T/\delta)}\right) \\ & + T^{1/2}\kappa\left(1 + 4\sqrt{2\log(2/\delta)}\right) \\ & + \kappa\,.\end{aligned}$$

Thus, for some constant C' universal constant, for T larger than a universal constant,

$$\mathfrak{R}_T \leq C' T^{3/4}\cdot\left(Q_{\max} S_\delta(A, \mu^*) + (\kappa + C)\sqrt{\log(T/\delta)}\right) + 2\varepsilon_0 T\,.$$

C. Proof of Corollary 4.3

By Theorem 4.1,

$$\overline{\mathfrak{R}}_T = \sum_{t=1}^{n} (\lambda_{\pi_{(t)}} - \lambda_{\pi^*}) = \widetilde{O}(\sqrt{n}) \, .$$

Thus,

$$\frac{1}{n} \sum_{t=1}^{n} \lambda_{\pi_{(t)}} \leq \lambda_{\pi^*} + \widetilde{O}(1/\sqrt{n}) \, .$$

D. Finite-time analysis of LSPE

In this section we prove the bound on the LSPE estimation error given in Theorem 5.1. For simplicity we will drop the π subscripts in this section; for example instead of Q_π, Π_π, D_π, we will write Q, Π, D.

Proof. For the purpose of analysis, it will be convenient to express the above update in terms of empirical estimates of the matrices $H, D \in [0,1]^{SA \times SA}$ computed from counts of state-action pairs and transitions, which we denote by $\widehat{H}_\tau$ and $\widehat{D}_\tau$. Letting $\widehat{\Pi}_\tau = \Psi(\Psi^\top \widehat{D}_\tau \Psi)^{-1} \Psi^\top \widehat{D}_\tau$, the update rule satisfies

$$\Psi w^{(k+1)} = (1-\gamma)\Psi w^{(k)} + \gamma \widehat{\Pi}_\tau \big(\widehat{H}_\tau \Psi w^{(k)} + c - \widehat{\lambda}_\tau \mathbf{1} \big) \, .$$

Next, we rewrite this update as a deterministic update plus stochastic noise:

$$\begin{aligned} \Psi w^{(k+1)} &= \big((1-\gamma) I + \gamma \widehat{\Pi}_\tau \widehat{H}_\tau \big) \Psi w^{(k)} + \gamma \widehat{\Pi}_\tau (c - \hat{\lambda}_\tau \mathbf{1}) \\ &= ((1-\gamma) I + \gamma \Pi H) \Psi w^{(k)} + \gamma \Pi (c - \lambda \mathbf{1}) + \gamma \Psi (Z_\tau \Psi w^{(k)} + \zeta_\tau) \\ &= F_\gamma \Psi w^{(k)} + \gamma \Pi (c - \lambda \mathbf{1}) + \gamma (Z_\tau \Psi w^{(k)} + \zeta_\tau) \, , \end{aligned} \tag{15}$$

where $F_\gamma := \Pi((1-\gamma)I + \gamma H)$, and the noise terms Z_τ and ζ_τ are defined as

$$Z_\tau = \widehat{\Pi}_\tau \widehat{H}_\tau - \Pi H \, , \tag{16}$$

$$\zeta_\tau = \widehat{\Pi}_\tau (\mathbf{c} - \lambda_t \mathbf{1}) - \Pi(\mathbf{c} - \lambda \mathbf{1}) \tag{17}$$

Note that the true parameters $\widetilde{w}$ satisfy

$$\Psi \widetilde{w} = \Pi H (\Psi \widetilde{w}) + \Pi(c - \lambda \mathbf{1}) = F_\gamma \Psi \widetilde{w} + \gamma \Pi (c - \lambda \mathbf{1}) \, . \tag{18}$$

As discussed in Section 5.1, $\widetilde{w}$ exists and is unique. Subtracting (15) from (18) and recursing we get

$$\begin{aligned} \Psi(\widetilde{w} - w^{(k+1)}) &= F_\gamma \Psi(\widetilde{w} - w^{(k)}) + \gamma (Z_\tau \Psi w^{(k)} + \zeta_\tau) \\ &= F_\gamma^k \Psi(\widetilde{w} - w^{(0)}) + \gamma \sum_{i=0}^{k} F_\gamma^{k-i} (Z_\tau \Psi w^{(i)} + \zeta_\tau) \, . \end{aligned}$$

By Prop. 4 of Yu & Bertsekas (2009), under assumptions A1 and A3, for any $\gamma \in (0,1)$ there exists $\alpha \in [0,1)$ such that $\|F_\gamma\|_\nu \leq \alpha$. Taking the ν-weighted norm of both sides gives us the following error bound:

$$\|\Psi(\widetilde{w} - w^{(k+1)})\|_\nu \leq \alpha^k \|\Psi(\widetilde{w} - w^{(0)})\|_\nu + \frac{C}{1-\alpha} \left(\max_i \|Z_\tau \Psi \widehat{w}^{(i)}\|_\nu + \|\zeta_\tau\|_\nu \right) .$$

In Appendix D.1 we show that with probability at least $1-\delta$, the noise terms scale as

$$\begin{aligned} \|Z_\tau \Psi \widehat{w}\|_\nu &= O\left(\sqrt{1/\tau} \sigma^{-2} \|\widehat{w}\| C_\Psi^5 \kappa \mathrm{polylog}(d, 1/\delta) \right) \\ \|\zeta_\tau\|_\nu &= O\left(\sqrt{d/\tau} \kappa \sigma^{-2} C_\Psi^3 \|\Psi\|_{\max} \mathrm{polylog}(d, 1/\delta) \right) \end{aligned}$$

where $C_\Psi = \max_{x,a} \|\psi(x,a)\|$, and $\|\Psi\|_{\max}$ is the maximum absolute entry of Ψ. Appendix D.3 provides an upper bound on $\|\widehat{w}^{(k)}\|$. Combining these with a union bound over $i \in \{1, \dots, k\}$ finishes the proof. $\square$

D.1. Bounding $\|Z_\tau \Psi w\|_\nu$ and $\|\zeta_\tau\|_\nu$

In this section, we will construct upper bounds on the norm of $d \times d$ matrices $\Psi^\top B \Psi$ for several different choices of B. Let ψ_i be the feature vector corresponding to the i row of Ψ, and let $C_\Psi = \max_{x,a} \|\psi(x,a)\|$. For any matrix B, we have

$$\|\Psi^\top B \Psi\| = \|\sum_{i=1}^{SA}\sum_{j=1}^{SA} B_{ij}\psi_i\psi_j^\top\| \le \sum_{i=1}^{SA}\sum_{j=1}^{SA} |B_{ij}|\|\psi_i\psi_j^\top\| \le C_\Psi^2 \sum_{i=1}^{SA}\sum_{j=1}^{SA}|B_{ij}| = C_\Psi^2\|B\|_{1,1}\,. \tag{19}$$

Thus $\|\Psi^\top D\Psi\| \le C_\Psi^2$ and $\|\Psi^\top DH\Psi\| \le C_\Psi^2$. We will also consider the case where B is the average of τ zero-mean random matrices. We show in Appendix D.2 that $\|\Psi^\top(\widehat{D}_\tau - D)\Psi\|$, $\|\Psi^\top D(\widehat{H}_\tau - H)\Psi\|$, and $\|\Psi^\top(\widehat{D}_\tau - D)H\Psi\|$ scale as $O(\tau^{-1/2}\kappa {C_\Psi}^2 \text{polylog}(d, 1/\delta))$.

D.1.1. BOUNDING $\|Z_\tau \Psi w\|_\nu$

We first bound $\|Z_\tau\Psi w\|_\nu$, which can be decomposed as

$$\|Z_\tau\Psi w\|_\nu = \|D^{1/2}(\widehat{\Pi}_\tau\widehat{H}_\tau - \Pi H)\Psi w\| \le \|D^{1/2}\Pi(\widehat{H}_\tau - H)\Psi w\| + \|D^{1/2}(\widehat{\Pi}_\tau - \Pi)\widehat{H}_\tau\Psi w\|\,. \tag{20}$$

For the first term in (20),

$$\begin{aligned}\|D^{1/2}\Pi(\widehat{H}_\tau - H)\Psi w\| &\le \|D^{1/2}\Psi\|\|(\Psi^\top D\Psi)^{-1}\|\|\Psi^\top D(\widehat{H}_\tau - H)\Psi w\| \\ &\le C_\Psi\sigma^{-1}\|\Psi^\top D(\widehat{H}_\tau - H)\Psi\|\|w\| \\ &= O\left(\tau^{-1/2}\kappa {C_\Psi}^3\sigma^{-1}\|w\|\text{polylog}(d,1/\delta)\right)\,.\end{aligned}$$

For the second term in (20), letting $M = \Psi(\Psi^\top D\Psi)^{-1}\Psi^\top$ and $\widehat{M} = \Psi(\Psi^\top\widehat{D}_\tau\Psi)^{-1}\Psi^\top$, we have

$$\begin{aligned}\|D^{1/2}(\widehat{\Pi}_\tau - \Pi)\widehat{H}_\tau\Psi w\| &= \|D^{1/2}(\widehat{M}\widehat{D}_\tau - MD)\widehat{H}_\tau\Psi w)\| \\ &\le \|D^{1/2}M(\widehat{D}_\tau - D)\widehat{H}_\tau\Psi w\| + \|D^{1/2}(\widehat{M} - M)\widehat{D}_\tau\widehat{H}_\tau\Psi w\|\,.\end{aligned} \tag{21}$$

For the first term in (21), we have

$$\begin{aligned}\|D^{1/2}\Psi(\Psi^\top D\Psi)^{-1}\Psi^\top(D - \widehat{D}_\tau)\widehat{H}_\tau\Psi w\| &\le C_\Psi\sigma^{-1}\|\Psi^\top(D - \widehat{D}_\tau)\widehat{H}_\tau\Psi\|\|w\| \\ &= O\left(\tau^{-1/2}\kappa {C_\Psi}^3\sigma^{-1}\|w\|\text{polylog}(d,1/\delta)\right)\,.\end{aligned} \tag{22}$$

For the remaining term, using the matrix inversion lemma,

$$\begin{aligned}(\Psi^\top\widehat{D}_\tau\Psi)^{-1} &= (\Psi^\top D\Psi - \Psi^\top(D - \widehat{D}_\tau)\Psi)^{-1} \\ &= (\Psi^\top D\Psi)^{-1} + (\Psi^\top D\Psi)^{-1}(\Psi^\top(D - \widehat{D}_\tau)\Psi)(\Psi^\top\widehat{D}_\tau\Psi)^{-1} \\ \widehat{M} - M &= \Psi(\Psi^\top D\Psi)^{-1}(\Psi^\top(D - \widehat{D}_\tau)\Psi)(\Psi^\top\widehat{D}_\tau\Psi)^{-1}\Psi^\top\,.\end{aligned}$$

Thus,

$$\begin{aligned}\|D^{1/2}(\widehat{M} - M)\widehat{D}_\tau\widehat{H}_\tau\Psi w\| &= \|D^{1/2}\Psi(\Psi^\top D\Psi)^{-1}\Psi^\top(D - \widehat{D}_\tau)\Psi(\Psi^\top\widehat{D}_\tau\Psi)^{-1}\Psi^\top\widehat{D}_\tau\widehat{H}_\tau\Psi w\| \\ &\le C_\Psi^3\sigma^{-1}\|w\|\|(\Psi^\top\widehat{D}_\tau\Psi)^{-1}\|\|\Psi^\top(D - \widehat{D}_\tau)\Psi\| \\ &= O\left(\tau^{-1/2}\kappa {C_\Psi}^5\sigma^{-2}\|w\|\text{polylog}(d,1/\delta)\right)\,.\end{aligned} \tag{23}$$

In the last step, we have used that given exponentially fast mixing, for large enough τ, we will have $\|\Psi^\top\widehat{D}_\tau\Psi\| \ge \sigma/2$.

D.1.2. BOUNDING $\|\zeta_\tau\|_\nu$

The noise term ζ_τ can be decomposed as

$$\|\zeta_\tau\|_\nu \le |\widehat{\lambda}_\tau - \lambda| \|\widehat{\Pi}_\tau \mathbf{1}\|_\nu + \|(\widehat{\Pi}_\tau - \Pi)(c - \lambda \mathbf{1})\|_\nu \tag{24}$$

For the first term, by Lemma 4.4 (the bound on W_T with $\pi^* = \pi$, $T = \tau$), $|\widehat{\lambda}_\tau - \lambda|$ scales as $O(\tau^{-1/2})$. Furthermore,

$$\|\widehat{\Pi}_\tau \mathbf{1}\|_\nu = \|D^{1/2}\Psi(\Psi^\top \widehat{D}_\tau \Psi)^{-1}\Psi^\top \widehat{D}_\tau 1\| \le C_\Psi^2 \|(\Psi^\top \widehat{D}_\tau \Psi)^{-1}\| \|\widehat{D}_\tau^{1/2}\mathbf{1}\| \,.$$

We bound the second term similarly to the second term in (20), with $c - \lambda\mathbf{1}$ in place of $\widehat{H}_\tau \Psi w$. Thus, instead of having a factor $\|\Psi^\top \widehat{D}_\tau \widehat{H}_\tau \Psi w\| \le C_\Psi^2 \|w\|$, we now have a factor $\|\Psi^\top \widehat{D}_\tau (c - \lambda \mathbf{1})\|$. Each entry of $\Psi^\top \widehat{D}_\tau (c - \lambda\mathbf{1})$ is bounded by $2\|\Psi\|_{\max}$, and so $\|\Psi^\top \widehat{D}_\tau (c - \lambda\mathbf{1})\| \le 2\sqrt{d}\|\Psi\|_{\max}$. Thus with probability at least $1 - \delta$,

$$\|\zeta_\tau\|_\nu = O\left(\tau^{-1/2}\kappa C_\Psi^3 \sigma^{-2}\sqrt{d}\|\Psi\|_{\max}\mathrm{polylog}(d, 1/\delta)\right).$$

D.2. Bounding $\|\Psi^\top(\widehat{D}_\tau - D)\Psi\|$, $\|\Psi^\top(\widehat{D}_\tau - D)\widehat{H}_\tau\Psi\|$, and $\|\Psi^\top D(\widehat{H}_\tau - H)\Psi\|$

In this subsection, we will rely on the following version of the matrix Azuma inequality. Let $(\mathcal{F}_k)_k$ be a filtration and define $\mathbf{E}_k[\cdot] := \mathbf{E}[\cdot|\mathcal{F}_k]$.

Theorem D.1 (Matrix Azuma (Tropp, 2012))**.** *Consider a finite $(\mathcal{F})_k$-adapted sequence $\{X_k\}$ of Hermitian matrices of dimension d, and a fixed sequence $\{A_k\}$ of self-adjoint matrices that satisfy $\mathbf{E}_{k-1}X_k = 0$ and $X_k^2 \preceq A_k^2$ almost surely. Let $v = \|\sum_k A_k^2\|$. Then for all $t \ge 0$, with probability at least δ,*

$$\|\sum_k X_k\| \le 2\sqrt{2v\ln(d/\delta)}\,.$$

A version of the inequality for non-Hermitian matrices of dimension $d_1 \times d_2$ can be obtained by applying the theorem to a Hermitian dilation of X, $\mathcal{D}(B) = \left[\begin{smallmatrix} 0 & X \\ X^* & 0 \end{smallmatrix}\right]$, which satisfies $\lambda_{\max}(\mathcal{D}(X)) = \|X\|$ and $\mathcal{D}(X)^2 = \left[\begin{smallmatrix} XX^* & 0 \\ 0 & X^*X \end{smallmatrix}\right]$. In this case, we have that $v = \max\left(\|\sum_k X_k X_k^*\|, \|\sum_k X_k^* X_k\|\right)$.

D.2.1. BOUNDING $\|\Psi^\top(\widehat{D}_\tau - D)\Psi\|$ AND $\|\Psi^\top(\widehat{D}_\tau - D)\widehat{H}_\tau\Psi\|$

We start by bounding $\|\Psi^\top(\widehat{D}_\tau - D)\Psi\|$. Let $(B_i)_i$ be the Doob martingale defined in Lemma A.1. By this lemma, $\|B_i - B_{i-1}\|_1 \le 4\kappa$.

Note that $\widehat{D}_\tau = \tau^{-1}B_\tau$. We decompose $\|\Psi^\top(\widehat{D}_\tau - D)\Psi\|$ as follows:

$$\|\Psi^\top(\widehat{D}_\tau - D)\Psi\| \le \tau^{-1}\|\Psi^\top \mathrm{diag}(B_\tau - B_0)\Psi\| + \|\Psi^\top(\tau^{-1}B_0 - D)\Psi\|\,.$$

Since $\Psi^\top B_i \Psi$ is a matrix-valued martingale, we bound the first term by applying the matrix Azuma inequality to its difference sequence, which satisfies $\|(\Psi^\top \mathrm{diag}(B_i - B_{i-1})\Psi)^2\| \le C_\Psi^4 \|B_i - B_{i-1}\|_1^2 \le 16C_\Psi^4\kappa^2$. Thus we get

$$\|\Psi^\top \mathrm{diag}(B_\tau - B_0)\Psi\| \le 8C_\Psi^2\kappa\sqrt{2\tau\ln(2d/\delta)}.$$

We bound the second term using the fast mixing assumption:

$$\|\Psi^\top(\tau^{-1}B_0 - D)\Psi\| \le \tau^{-1}\sum_{t=1}^{\tau}\|\Psi^\top \mathrm{diag}(\nu_t - \nu)\Psi\| \le \tau^{-1}C_\Psi^2\,\|\nu_0 - \nu\|_1\kappa\,.$$

The same bounds apply to $\|\Psi^\top(\widehat{D}_\tau - D)\widehat{H}_\tau\Psi\|$ by observing that since rows of $\widehat{H}_\tau$ sum to 1,

$$\|\Psi^\top \mathrm{diag}(B_i - B_{i-1})\widehat{H}_\tau\Psi\|^2 \le C_\Psi^4\|\mathrm{diag}(B_i - B_{i-1})\widehat{H}_\tau\|_{1,1}^2 \le C_\Psi^4\|B_i - B_{i-1}\|_1^2\,.$$

D.2.2. BOUNDING $\|\Psi^\top D(\widehat{H}_\tau - H)\Psi\|$

Next we bound $\|\Psi^\top D(\widehat{H}_\tau - H)\Psi\|$. We decompose this quantity as

$$\|\Psi^\top D(\widehat{H}_\tau - H)\Psi\| \le \|\Psi^\top (\widehat{D}_\tau - D)\widehat{H}_\tau \Psi\| + \|\Psi^\top (\widehat{D}_\tau \widehat{H}_\tau - DH)\Psi\| \,.$$

We already have a bound for the first term, and bound the second term next. Notice that we can write $\tau \widehat{D}_\tau \widehat{H}_\tau = \sum_{t=2}^{\tau} X_{t-1} X_t^\top$. We define the martingale sequence

$$Y_i = \mathbf{E}\left[\tau \widehat{D}_\tau \widehat{H}_\tau | X_1, .., X_i\right] = \sum_{t=2}^{i} X_{t-1} X_t^\top + \sum_{t=i+1}^{\tau} \mathbf{E}[X_{t-1} X_t^\top | X_i] = \sum_{t=2}^{i} X_{t-1} X_t^\top + \sum_{t=1}^{\tau-i} \mathrm{diag}(X_i^\top H^{t-1}) H \quad (25)$$

$$Y_0 = \sum_{t=1}^{\tau} \mathrm{diag}(\nu_0^\top H^{t-1}) H \,. \quad (26)$$

The difference sequence can again be bounded using the fast mixing assumption:

$$\begin{aligned}
\|Y_i - Y_{i-1}\|_{1,1} &= \|\sum_{t=0}^{\tau-i-1} \mathrm{diag}(X_i^\top H^t - X_i^\top H^{t+1}) H\|_{1,1} \\
&\le \sum_{t=0}^{\tau-i-1} \|\,\mathrm{diag}(X_i^\top H^t - \nu) H\|_{1,1} + \|\,\mathrm{diag}(X_{i-1}^\top H^{t+1} - \nu) H\|_{1,1} \\
&\le (\|X_{i-1} - \nu\|_1 + \|X_i - \nu\|_1)(1 - \exp(-1/\kappa))^{-1} \le 4\kappa \,.
\end{aligned}$$

We decompose $\|\Psi^\top (\widehat{D}_\tau \widehat{H}_\tau - DH)\Psi\|$ as follows:

$$\|\Psi^\top (\widehat{D}_\tau \widehat{H}_\tau - DH)\Psi\| = \tau^{-1} \|\Psi^\top (Y_\tau - Y_0)\Psi\| + \|\Psi^\top (\tau^{-1} Y_0 - DH)\Psi\| \,.$$

We use the matrix Azuma inequality for the first part, where $v \le \tau \|\Psi^\top (Y_i - Y_{i-1})\Psi\|^2 \le \tau 16 C_\Psi^2 \kappa^2$. Thus,

$$\|\Psi^\top (Y_\tau - Y_0)\Psi\| \le 8 C_\Psi^2 \kappa \sqrt{2\tau \ln(2d/\delta)} \,.$$

For the second part, we have

$$\|\Psi^\top (\tau^{-1} Y_0 - DH)\Psi\| \le \tau^{-1} C_\Psi^2 \|\sum_{t=1}^{\tau} \mathrm{diag}(\nu_{t-1} - \nu) H\|_{1,1} \le \tau^{-1} C_\Psi^2 \kappa \|\nu_0 - \nu\|_1$$

Putting all terms together, we have the following bounds:

$$\|\Psi^\top (\widehat{D}_\tau - D)\Psi\| \le \tau^{-1/2} C_\Psi^2 \kappa \left(8\sqrt{2\ln(2d/\delta)} + \tau^{-1/2} \|\nu_0 - \nu\|_1\right) \,, \quad (27)$$

$$\|\Psi^\top (\widehat{D}_\tau - D)\widehat{H}_\tau \Psi\| \le \tau^{-1/2} C_\Psi^2 \kappa \left(8\sqrt{2\ln(2d/\delta)} + \tau^{-1/2} \|\nu_0 - \nu\|_1\right) \,, \quad (28)$$

$$\|\Psi^\top D(\widehat{H}_\tau - H)\Psi\| \le 2\tau^{-1/2} C_\Psi^2 \kappa \left(8\sqrt{2\ln(2d/\delta)} + \tau^{-1/2} \|\nu_0 - \nu\|_1\right) \,. \quad (29)$$

D.3. Bounding $\widehat{Q}(x,a)$ and $\|\widehat{w}\|$

We have that

$$\widetilde{w} = -(\Psi^\top D_\pi (H_\pi - I)\Psi)^{-1} \Psi^\top D_\pi (c - \lambda_\pi \mathbf{1}) \,.$$

Let the constant C_2 be such that

$$\sup_\pi \|(\Psi^\top D_\pi (H_\pi - I)\Psi)^{-1} \Psi^\top D_\pi (c - \lambda_\pi \mathbf{1})\| \le C_2, \qquad \|w^{(0)}\| \le C_2 \,.$$

Let constant $C_\Psi > 1$ be such that $\max_{x,a} \|\psi(x,a)\| \le C_\Psi$. We show that for some small constant $C' > 1$, for any i, the estimate of LSPE algorithm satisfies $\|w^{(i)}\| \le C' C_2$. We prove by induction. The induction base holds because

$\|w^{(0)}\| \leq C_2$. Next, assume the statement holds for all $i \in \{0, \dots, k\}$. By the argument in Appendix D.1, for some positive constants C_3, C_4, with high probability,

$$\|\zeta_\tau\|_\nu \leq C_3 \,, \qquad \|Z_t \Psi w^{(i)}\|_\nu \leq \frac{C' C_2 C_4}{\sqrt{\tau}} \,.$$

Given LSPE iteration

$$\Psi(\widetilde{w} - w^{(k+1)}) = F_\gamma^k \Psi(\widetilde{w} - w^{(0)}) + \gamma \sum_{i=0}^{k} F_\gamma^{k-i}(Z_\tau \Psi w^{(i)} + \zeta_\tau) \,,$$

we get

$$\sigma\|\widetilde{w} - w^{(k+1)}\| \leq \|\Psi(\widetilde{w} - w^{(k+1)})\|_\nu \leq \alpha^k (2C_\Psi C_2) + \frac{C_3 + C' C_2 C_4}{(1-\alpha)\sqrt{\tau}} \,.$$

Thus,

$$\|w^{(k+1)}\| \leq C_2 + \alpha^k \frac{2C_\Psi C_2}{\sigma} + \frac{C_3 + C' C_\Psi C_4}{(1-\alpha)\sigma\sqrt{\tau}} \,.$$

For any $C' \geq 2 + \frac{2C_\Psi \alpha}{\sigma}$, for $\tau \geq ((C_3/C_2 + C'C_4)/(1-\alpha))^2$, it follows that $\|w^{(k+1)}\| \leq C' C_2$.

E. Guaranteeing $\sup_\pi \alpha_\pi < 1$

We will need a perturbation bound for the stationary distribution of Markov chains. Let $\mathcal{P}$ be the set of $n \times n$ irreducible transition matrices. For $P \in \mathcal{P}$, let $\mu(P)$ be the unique stationary distribution underlying P.

Lemma E.1. *There exist a function $\kappa : \mathcal{P} \to [0, \infty)$ such that $\|\mu(P) - \mu(P')\| \leq \kappa(P)\|P - P'\|$ holds for any $P, P' \in \mathcal{P}$.*

The norm in the lemma can be chosen freely (this only changes the definition of κ) Cho & Meyer (2001) list and compare several functions κ (by fixing various norms), which are expressed as some algebraic function of P. It follows from this lemma that the map $P \mapsto \mu(P)$ is continuous over $\mathcal{P}$.

Fix $0 < \gamma < 1$. Recall that (by slightly changing the notation), $\alpha_\pi = \|F_\pi\|_{\nu_\pi}$ where

$$F_\pi := \Pi_\pi((1-\gamma)I + \gamma H_\pi \,),$$

where $H_\pi(x, a; y, b) = \pi(b|y)P(y|x, a)$,

$$\Pi_\pi = \Psi G_\pi^{-1} \Psi^T D_\pi \,,$$
$$G_\pi = \Psi^\top D_\pi \Psi \,,$$

and $D_\pi = \mathrm{diag}(\nu_\pi)$. Let Π_{S} be the set of stationary (randomizing) policies, Π_{S}^+ be the subset of these which select all actions with positive probability at every state, $\Theta \subset \mathbb{R}^d$, $\pi : \Theta \to \Pi_{\mathrm{S}}^+$ be a map.

Proposition E.2. *Let $\Pi = \pi(\Theta)$, Θ compact, the map $\pi : \Theta \to \Pi_{\mathrm{S}}^+$ continuous, assumptions A1, A3 hold and assume that is G_π nonsingular for any policy $\pi \in \Pi$. Then,* $\sup_{\pi \in \Pi} \alpha_\pi < 1$.

Note that the condition that G_π is nonsingular for any policy π follows from assumption A4.

Proof. Note that A1 and the fact that $\Pi \subset \Pi_{\mathrm{S}}^+$ imply that for any $\pi \in \Pi$, ν_π is fully supported on $\mathcal{X} \times \mathcal{A}$. Then, thanks to A3, Prop. 4 of Yu & Bertsekas (2009) applies to the Markov chain of state-action pairs that results from using π in the MDP implies that $\alpha_\pi < 1$ holds for any policy $\pi \in \Pi$.

We claim that *(a)* $\mathcal{N} := \{\nu_\pi \,:\, \pi \in \Pi\} \subset [0, 1]^{\mathcal{X} \times \mathcal{A}}$ is compact and that *(b)* $f : \mathcal{N} \to \mathbb{R}$, $\nu_\pi \mapsto \alpha_{\nu_\pi} := \alpha_\pi$ is well-defined and is continuous. The result then follows by Weierstrass' theorem that shows that continuous functions on compacta take on their extreme values.

Compactness of $\mathcal{N}$ follows from the continuity of the $\pi : \Theta \to \Pi_{\mathrm{S}}$, that $\Pi = \pi(\Theta) \subset \Pi_{\mathrm{S}}^+$ and Lemma E.1.

That f is well-defined can be seen because if ν is a stationary distribution over $\mathcal{X} \times \mathcal{A}$ with full support over the states, an underlying policy can be extracted from ν as the collection of conditional $\{\nu(x, a)/\sum_b \nu(x, b)\}_x$. As a result, thanks to its structure, F_π can be expressed as a function of ν_π alone. The continuity of f follows because the composition of continuous maps is continuous. $\square$

Input: Trajectory length T, initial state x_0

Let $E = \tau = \sqrt{T}$ and set $Q_0(x, a) = 0 \ \forall x, a$
for $i := 1, 2, \ldots, E$ **do**
  Set $\pi_i(a|x) \propto \exp\left(-\eta \sum_{j=0}^{i-1} \widehat{Q}_j(x, a)\right)$
  Execute π_i for τ rounds and collect data $\mathcal{Z}_i = \{(x_t, a_t, c_t, x_{t+1})\}_{t=\tau(i-1)}^{\tau i}$
  Compute $\widehat{V}_i$ from $\mathcal{Z}_i$
  Collect data $\mathcal{Z}'_i = \{(x_t, a_t, c_t, x_{t+1})\}_{t=\tau(i-1)}^{\tau i}$ where x_t is sampled from μ_i , a_t is sampled randomly and (c_t, x_{t+1}) are sampled from dynamics
  Compute $\widehat{Q}_i$ from $\widehat{V}_i$ and $\mathcal{Z}'_i$
end for

Figure 3. A version of the POLITEX algorithm with two-stage value estimation.

F. A two stage algorithm

POLITEX estimates $Q_\pi(x, a)$ directly without estimating $V_\pi(x)$. This simplifying design choice comes at a cost; for example, Assumption A4 is violated by deterministic policies under tabular representation. One way to relax A4 is via a two-stage algorithm which first estimates state value functions $V_\pi(x)$, and then estimates $Q_\pi(x, a)$ using $\widehat{V}_\pi(x)$ and exploratory data. Such an algorithm is presented in 3. The analysis of this variant will require milder conditions as follows. Assume we use linear function approximation of the form $\widehat{V}_\pi = \Phi\widehat{\theta}_\pi$ to estimate state value of policy π, where Φ is a $S \times d$ feature matrix and $\widehat{V}_\pi$ is the value estimate. In this case, Assumption A4 can be replaced by the requirement that for any policy π, $\lambda_{\min}(\Phi^\top D_{\mu_\pi}\Phi) \geq \sigma$. In the tabular setting, this condition requires that any policy has a non-zero probability of visiting any state.

G. Atari experiment setup

Our implementation of the Atari experiment is based on Horgan et al. (2018), which is a distributed implementation of DQN Mnih et al. (2015), featuring Dueling networks Wang et al. (2015), N-step returns Asis et al. (2017), Prioritized replay Schaul et al. (2015), and Double Q-learning van Hasselt et al. (2015). We used Q-learning and epsilon-greedy exploration for DQN with a portfolio of epsilon values ranging from 0.04 to 0.0006. For POLITEX, we used TD learning and Boltzmann exploration with the learning rate η set according to SOLO FTRL by (Orabona & Pál, 2015): For a given state x,

$$\eta = \alpha\sqrt{2.75}\sqrt{\frac{\log d}{P_t}},$$

where $\alpha = 10$ is a tuneable constant multiplier (chosen based on preliminary experiments); d is the number of actions in the game and

$$P_t = \min_{c\in\mathbb{R}} \sum_{i=1}^{t} \|Q_i(x, \cdot) - c\mathbb{1}\|_\infty^2 ,$$

where $Q_i(x, \cdot)$ are the state-action values for all past Q-networks indexed from 1 to the current timestep t, $\mathbb{1}$ is a vector of all ones and the minimisation over c achieves robustness against the changing ranges of state-action values. The minimisation is one-dimensional convex optimisation problem which we solve numerically.

Both methods used the same neural network architecture of 3 convolutional layers followed by 1 fully connected layer. Each learner step samples a batch of 128 experiences from the experience replay memory, prioritised by TD-error. The actors take 16 game steps in total for each learning step. For DQN, the target network is updated with the online network every 2500 steps; for POLITEX, we enter a new phase and terminate the current phase when the number of learning steps taken in the current phase reaches 100 times the square root of the total learning steps taken. When a new phase is started, the freshly learned neural network is copied in a circular buffer of size 10, which is used by the actors to calculate the averaged Q-values, weighted by the length of each phase. Since the phase length are also unequal, we use a weighted sum where the weight corresponding to a network is the number of samples used to train that networks.