Machine Learning manuscript No.
(will be inserted by the editor)

Model Selection in Reinforcement Learning

Amir massoud Farahmand[1], **Csaba Szepesvári**[1]

Department of Computing Science
University of Alberta
Edmonton, Canada, T6G 2E8
e-mail: {amirf,szepesva}@ualberta.ca

The date of receipt and acceptance will be inserted by the editor

Abstract We consider the challenge of automating parameter tuning in reinforcement learning. More specifically, we consider the batch (off-line, non-interactive) reinforcement learning setting and the problem of learning an action-value function with a small Bellman error. We propose a complexity regularization-based model selection algorithm and prove its *adaptivity*: the procedure is shown to perform almost as well as if the best parameter setting was known ahead of time. We also discuss other approaches to derive adaptive procedures in reinforcement learning.

Key words Reinforcement learning, model selection, complexity regularization, adaptivity, offline learning, off-policy learning, finite-sample bounds

1 Introduction

A major goal of benchmarking is to find out which algorithms can be expected to work better on a new problem instance. This is a special case of the problem of recommending parameter settings for algorithms, which is a central issue in all learning algorithms. In reinforcement learning, one of the most important open questions is how to choose the function approximation technique or the parameters thereof. Examples of the latter include the number, location, and shape of basis functions of a radial basis function network, the number of layers and neurons in a neural network, or the number of tilings and their resolutions in the case of tile coding (cf. Chapter 8 of the book of Sutton and Barto 1998). Other examples are the regularization coefficient or kernel parameters of a regularized kernel-based reinforcement learning algorithm (e.g., Jung and Polani 2006a; Farahmand et al. 2009b; Taylor and Parr 2009; Kolter and Ng 2009). It is widely recognized that the best choice of the parameters is problem dependent. Hence, it makes sense

to choose the parameters in a data-dependent manner, the ultimate goal being to pick them so that the resulting performance is as good (or almost as good) as if the algorithm's best, unknown, problem-dependent parameter setting were used.

Data-dependent parameter tuning can be done either in an *ad hoc* or in a *systematic* manner. The advantage of the systematic approach is that it eliminates the "human in the loop". Therefore, it leads to reproducible results and helps to avoid mistakes. Further, as computing power becomes cheaper, automated parameter tuning becomes economically more beneficial than an approach based on human supervision.

In this paper we study the problem of automatic parameters tuning in the batch, i.e., *offline and non-interactive learning*, scenario. We assume that the environment is a Markovian Decision Process (Bertsekas and Shreve, 1978). In this scenario the data collection process finishes before the learning process starts. Further, data collection is typically uncontrolled or loosely controlled. In particular, the learning algorithm has no influence on how the data is collected. This is the standard situation when a new controller is to be designed for a mission critical system (e.g., Druet et al. 2000; Abbeel et al. 2007), but it also happens in other cases. That the learning system cannot interact with the environment makes this problem hard because there is no simple way to evaluate the performance of a chosen policy.

The goal of this paper is to investigate parameter tuning when we want to find a good fixed point to the Bellman optimality operator. Searching for the fixed point of the Bellman optimality operator is what value-function based reinforcement learning methods do to guide their search for a good policy. Examples of algorithms tailored to the offline, non-interactive setting include instances of generalized value iteration (e.g., Ernst et al. 2005; Riedmiller 2005; Antos et al. 2008a) or that of generalized policy iteration (e.g., Lagoudakis and Parr 2003; Antos et al. 2007, 2008b; Farahmand et al. 2009b). All the cited work use a procedure that relies on a function approximation technique with tuneable parameters whose choice is known to be critical for the success of the approach. In this paper we take the goal of minimizing the Bellman error as granted. We have a short discussion on the validity of this goal in Section 3.

To make the goal of systematic parameter tuning clear, consider the following setting: Assume that we are given a learning algorithm $\mathcal{A}$ that takes the data $\mathcal{D}_n$ and a class of functions $\mathcal{F}$ and then proposes an action-value function $Q_n = \mathcal{A}(\mathcal{D}_n, \mathcal{F})$ which is an element of $\mathcal{F}$. The goal of $\mathcal{A}$ is to come up with a function $Q_n \in \mathcal{F}$ whose Bellman error is close to that of the best choice from $\mathcal{F}$.

Now, let us turn to discussing the role of $\mathcal{F}$. When choosing the function class $\mathcal{F}$ (i.e., the function approximation technique and its parameters) the following must be kept in mind: A small class is problematic because even the best element of it will have a large Bellman error, leading to *underfitting*. On the other hand, a large $\mathcal{F}$ is also problematic because a procedure searching for a function with a small *empirical* Bellman error in $\mathcal{F}$ will have

a good chance of fitting to the noise in the data, which results *overfitting*. For a quantification of the tradeoff involved in choosing $\mathcal{F}$ in the case of generalized policy iteration see Antos et al. (2008b); Farahmand et al. (2009b). For similar results in the case of generalized value iteration in the finite (infinite) action setting see Farahmand et al. (2009a) (respectively, Antos et al. 2008a).

In this framework we formulate the goal of an automated parameter selection process as follows. Let p represent the parameter(s) that leads to different function spaces, and let $\mathcal{F}(p)$ be the corresponding function space. For simplicity, assume that p is ordered and $\mathcal{F}(p) \subset \mathcal{F}(p')$ when $p \leq p'$ (e.g., in the case of linear function approximation, $\mathcal{F}(p)$ could be the space of functions spanned by the first p functions in an infinite series of basis functions). The goal now can be described as follows: Let p^* be the best parameter setting for an algorithm on a given problem. The goal is to design a parameter-selecting algorithm that, given the data, chooses the parameters such that the resulting performance is as good as the performance of the algorithm $\mathcal{A}$ running on the data with $\mathcal{F}(p^*)$, i.e., resulting in a performance that is competitive with that which could have been obtained if $\mathcal{A}$ was tuned to the problem in hindsight. If a parameter tuning algorithm achieves this goal, we call it *adaptive*. It is our goal here to design such an adaptive procedure.

1.1 Results

In supervised learning, a classical method to achieve adaptivity is *complexity regularization* (Barron, 1991; Bartlett et al., 2002; Wegkamp, 2003; Lugosi and Wegkamp, 2004). A straightforward adoption of complexity regularization to our problem suggests the following procedure: Assume that the possible parameter settings are enumerated in a list $p_1, p_2, \ldots$. For $k = 1, 2, \ldots$, run algorithm $\mathcal{A}$ using the function space $\mathcal{F}(p_k)$ to obtain an action-value function candidate Q_k. Next, estimate the Bellman error of Q_k, e.g. using a hold-out data with n observations. Let the resulting estimate be $\mathrm{BE}_n(Q_k)$. Then choose

$$\hat{k} = \mathrm{argmin}_{k \geq 1} \left[C_1 \, \mathrm{BE}_n(Q_k) + C_2 \, \frac{\log k}{n} \right],$$

where $C_1 \geq 1$ and $C_2 > 0$ is a well-chosen constant. General model selection results can then be used to show that this procedure is indeed adaptive, provided that $\mathrm{BE}_n(Q_k)$ is an unbiased estimate of the Bellman error of Q_k (cf. Theorem 1 below).

Unfortunately, we know of no way of deriving an unbiased estimate of the Bellman error of Q_k based on a finite amount of data. Therefore the above procedure is only of theoretical interest. The main contribution of the paper is a procedure similar to the above one (cf. Section 4), but one which can in fact be implemented and still achieves adaptivity, thus overcoming

the difficulty of not being able to measure the error directly. The main result of the paper is a proof that this procedure is adaptive (cf. Theorem 2 and Corollary 1 in Section 5). In addition to this, we discuss some alternative methods to achieve adaptivity in the offline, non-interactive reinforcement learning setting (Section 6).

In the next two sections we briefly review the the necessary background (Section 2), followed by a formal definition of the learning problem (Section 3).

2 Background

In this section, we provide a very brief summary of some of the concepts and definitions from the theory of Markov Decision Processes (MDP) and reinforcement learning (RL). We assume that the reader is familiar with these concepts: The purpose of the section is to introduce the notation used. For further information about MDPs and reinforcement learning the reader is referred to Bertsekas and Shreve (1978); Puterman (1994); Bertsekas and Tsitsiklis (1996); Sutton and Barto (1998); Szepesvári (2009). In addition to this background on MDPs, in the second part of the section we introduce the assumption on the learning scenario considered, as well as some less standard notations. Thus, readers are advised to pay some extra attention to this second half.

2.1 Background on Markov Decision Processes

Definition 1 *A* finite-action discounted MDP *is a 4-tuple* $(\mathcal{X}, \mathcal{A}, P, \gamma)$*, where* $\mathcal{X}$ *is a measurable state space,* $\mathcal{A}$ *is a finite set of actions,* P *is a mapping with domain* $\mathcal{X} \times \mathcal{A}$ *and* $0 \leq \gamma < 1$ *is a discount factor. Mapping* P *evaluated at* $(x, a) \in \mathcal{X} \times \mathcal{A}$ *gives a distribution over* $\mathbb{R} \times \mathcal{X}$*, which we shall denote by* $P(\cdot, \cdot|x, a)$*.*

An MDP encodes a temporal evolution of a controlled discrete-time stochastic process. The dynamical system starts at time $t = 0$ with random initial state $X_0 \sim P_{\text{init}}$.[1] At stage t, action $A_t \in \mathcal{A}$ is selected by the agent controlling the process. As a result the pair (R_t, X_{t+1}) is drawn from $P(\cdot, \cdot|X_t, A_t)$: $(R_t, X_{t+1}) \sim P(\cdot, \cdot|X_t, A_t)$. Here, R_t is the reward that the agent receives and X_{t+1} is the next state. The procedure is then repeated.

For our purposes it suffices to deal with action selection procedures, or policies, which select an action deterministically, do not change in time and which base the selection of the action on the current state.

Definition 2 (Deterministic Markov stationary policy) *A mapping* $\pi : \mathcal{X} \to \mathcal{A}$ *is called a deterministic Markov stationary policy, or just* policy

[1] Here, P_{init} is a distribution over the states, which, could be part of the definition of MDPs.

in short. Following a policy π in an MDP means that at each time step $A_t = \pi(X_t)$.

To study MDPs, two auxiliary functions are of central importance: the *value* and the *action-value functions* of a policy π.

Definition 3 (Value functions) *The value function V^π and the action-value function Q^π for a policy π are defined as follows: Let $(R_t; t \geq 0)$ be the sequence of rewards when the process is started from a state X_0 drawn from a positive probability distribution over $\mathcal{X}$. Then*

$$V^\pi(x) \stackrel{\text{def}}{=} \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t R_t \,\middle|\, X_0 = x\right].$$

Further, let $(R_t; t \geq 0)$ be the sequence of rewards when the process is started such that (X_0, A_0) is drawn from a positive probability distribution over $\mathcal{X} \times \mathcal{A}$. Then

$$Q^\pi(x,a) \stackrel{\text{def}}{=} \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t R_t \,\middle|\, X_0 = x, A_0 = a\right].$$

In words, the value function V^π evaluated at state x gives the total expected discounted return of using policy π from that state and defines the basis of comparison of policies.

It is easy to see that for any policy π, if the absolute value of the immediate expected reward

$$r(x,a) = \int r \, \mathcal{R}(dr|x,a)$$

is uniformly bounded by $R_{\max}$, then the functions V^π and Q^π are bounded by $V_{\max} = Q_{\max} = R_{\max}/(1-\gamma)$.

For a discounted MDP, we define the *optimal value* and *action-value* functions by

$$V^*(x) = \sup_\pi V^\pi(x) \qquad (\forall x \in \mathcal{X}),$$
$$Q^*(x,a) = \sup_\pi Q^\pi(x,a) \qquad (\forall x \in \mathcal{X}, \forall a \in \mathcal{A}).$$

We say that a policy π is *optimal* if it achieves the best values in every state, i.e., if $V^\pi = V^*$.

We say that a policy π is *greedy* w.r.t. an action-value function Q and write

$$\pi = \hat{\pi}(\cdot; Q),$$

if $\pi(x) \in \arg\max_{a \in \mathcal{A}} Q(x,a)$ holds for all $x \in \mathcal{X}$ (If there exist multiple maximizers, some maximizer is chosen in an arbitrary deterministic manner). Greedy policies are important because a greedy policy w.r.t. Q^* is an

optimal policy. Hence, knowing Q^* is sufficient for behaving optimally (cf. Proposition 4.3 of Bertsekas and Shreve 1978).

As it turns out, Q^π and Q^* are fixed points of certain operators, which we define next.

Definition 4 (Bellman Operators) *Fix a policy π. The Bellman operators $T^\pi : B(\mathcal{X}) \to B(\mathcal{X})$ (for the value function V) and $T^\pi : B(\mathcal{X} \times \mathcal{A}) \to B(\mathcal{X} \times \mathcal{A})$ (for the action-value function Q) are defined as*

$$(T^\pi V)(x) \stackrel{\text{def}}{=} r(x, \pi(x)) + \gamma \int V^\pi(y) P(dy|x, a),$$
$$(T^\pi Q)(x, a) \stackrel{\text{def}}{=} r(x, a) + \gamma \int Q(y, \pi(y)) P(dy|x, a).$$

The fixed point of this operator is the (action-)value function of the policy π, i.e. $T^\pi Q = Q$ and $T^\pi V = V$ (cf. Proposition 4.2(b) of Bertsekas and Shreve 1978).

Definition 5 (Bellman Optimality Operators) *The Bellman optimality operators $T^* : B(\mathcal{X}) \to B(\mathcal{X})$ and $T^* : B(\mathcal{X} \times \mathcal{A}) \to B(\mathcal{X} \times \mathcal{A})$ are defined as*

$$(T^* V)(x) \stackrel{\text{def}}{=} \max_a \left\{ r(x, a) + \gamma \int V(y) P(dy|x, a) \right\},$$
$$(T^* Q)(x, a) \stackrel{\text{def}}{=} r(x, a) + \gamma \int \max_{a'} Q(y, a') P(dy|x, a).$$

Again, these operators enjoy a fixed-point property similar to that of the Bellman operators: $T^* Q^* = Q^*$ and $T^* V^* = V^*$ (cf. Proposition 4.2(a) of Bertsekas and Shreve 1978). The Bellman optimality operator thus provides a vehicle to compute the optimal action value function and to compute an optimal policy.

2.2 Offline Learning Problem and Empirical Bellman Operators

In a learning scenario, the Bellman (optimality) operators are not accessible. In the *offline* learning scenario, all that is known about the problem is in the form of a batch of data

$$\mathcal{D}_n = \{(X_1, A_1, R_1, Y_1), \dots, (X_n, A_n, R_n, Y_n)\},$$

where $(R_i, Y_i) \sim \mathcal{P}(\cdot, \cdot|X_i, A_i)$, $A_i \sim \pi_b(\cdot|X_i)$, and $X_i \sim \nu_{\mathcal{X}}$, with $\nu_{\mathcal{X}}$ being a fixed distribution over the states ($i = 1, \dots, n$). We shall also denote by ν the common distribution underlying (X_i, A_i). Samples X_i and X_{i+1} may be sampled independently, or may be coupled through $X_{i+1} = Y_i$. In the latter case the data comes from a single trajectory. Under either of these assumptions we say that the data $\mathcal{D}_n$ meets the *standard offline sampling assumption*.

The assumption that the states $\{X_i\}$ are identically distributed and that a fixed stationary policy is used to generate the data can be relaxed, but would complicate the analysis. Hence, we stick to the above assumptions, for simplicity.

Given the data $\mathcal{D}_n$, we may define the so-called empirical Bellman operators:

Definition 6 (Empirical Bellman Operators) *Let $\mathcal{D}_n$ be a dataset as above. Define the multiset $S_n = [[(X_1, A_1), \dots, (X_n, A_n)]]$ (i.e., if a state-action pair is repeated in the list it will be listed as many times as it is repeated). The empirical Bellman operator $\hat{T}^\pi : S_n \to \mathbb{R}$ is defined as*

$$(\hat{T}^\pi Q)(X_i, A_i) \stackrel{\text{def}}{=} R_i + \gamma Q(Y_i, \pi(Y_i)), \quad i = 1, \dots, n,$$

while the empirical Bellman optimal operator $\hat{T}^* : S_n \to \mathbb{R}$ is defined as

$$(\hat{T}^* Q)(X_i, A_i) \stackrel{\text{def}}{=} R_i + \gamma \max_{a'} Q(Y_i, a'), \quad i = 1, \dots, n.$$

The following proposition, which follows immediately from the definitions, shows that the empirical Bellman operators provide an unbiased estimate to the respective Bellman operators (Note that $\hat{T}^\pi$ and $\hat{T}^*$ depend on the data, and hence they are random. The dependence is suppressed to simplify the notation).

Proposition 1 *For $1 \le i \le n$, it holds true that*

$$\mathbb{E}\left[\hat{T}^\pi Q(X_i, A_i) | X_i, A_i\right] = T^\pi Q(X_i, A_i),$$
$$\mathbb{E}\left[\hat{T}^* Q(X_i, A_i) | X_i, A_i\right] = T^* Q(X_i, A_i).$$

In what follows we shall use $\|Q\|_\nu$ to denote the $L^2(\nu)$-norm of a measurable function $Q : \mathcal{X} \times \mathcal{A} \to \mathbb{R}$:

$$\|Q\|_\nu^2 \stackrel{\text{def}}{=} \int_{\mathcal{X}\times\mathcal{A}} |Q(x,a)|^2 d\nu(x,a),$$

and its empirical counterpart as

$$\|Q\|_n^2 \stackrel{\text{def}}{=} \frac{1}{n} \sum_{i=1}^n |Q(X_i, A_i)|^2.$$

Remember that $(X_i, A_i) \sim \nu$ by assumption. Hence, for any fixed function Q, we have $\mathbb{E}\left[\|Q\|_n^2\right] = \|Q\|_\nu^2$.

3 Problem Definition

Assume that we are given a list of action-value functions $Q_1, Q_2, \dots, Q_P$ (with the possibility of $P = \infty$) and a dataset $\mathcal{D}_n$, the latter of which satisfies the standard offline sampling assumption. The goal of this paper is to devise a procedure that selects the action-value function with the smallest Bellman error. Thus, the ideal procedure that implements this criterion would return $Q_{\hat{k}}$, where

$$\hat{k} = \operatorname{argmin} \left\{ \|Q_k - T^* Q_k\|_\nu^2 \, : \, 1 \le k \le P \right\}.$$

(Here ν is the distribution of states in $\mathcal{D}_n$, introduced at the end of the previous section.)

The motivation for this problem is that we expect that an action-value function with a small Bellman error leads to a greedy policy with a good performance. The performance of a policy is assumed to be measured with respect to some fixed, known, initial state-distribution ρ over the states and is defined as

$$V_\rho^\pi = \int V^\pi(x) \, d\rho(x) \tag{1}$$

which is the expected total discounted return of policy π provided that the initial state distribution is ρ. One can relate V_ρ^π to the Bellman error $\|Q_k - T^* Q_k\|_\nu$ in a way similar to Theorem 5.3 of Munos (2007) (see also Antos et al. 2008b).

The idea of using the Bellman error as a criterion of optimization is not new. The algorithms implementing generalized policy iteration can be viewed as working towards minimizing it, e.g. Lagoudakis and Parr (2003); Antos et al. (2008b). There are also some basis generation/adaptation methods that use Bellman error to guide their search, e.g. Menache et al. (2005); Keller et al. (2006); Parr et al. (2007).

Nevertheless, the Bellman error is not easy to work with. This is because neither T^* nor T^π is available in the learning setting. Moreover, even though $\hat{T}^*$ (T^π) provides an unbiased estimate to T^* (resp., T^π) in the sense of Proposition 1, they cannot be used in a simple manner to estimate the Bellman error. One might think that the mean-squared empirical Bellman residual, $\|Q - \hat{T}^* Q\|_n^2$ is a reasonable estimate to the Bellman error. However, it is easy to see that $\mathbb{E}\left[\|Q - \hat{T}^* Q\|_n^2\right] = \|Q - T^* Q\|_\nu^2 + \mathbb{E}\left[\|\hat{T}^* Q - T^* Q\|_n^2\right] \neq \|Q - T^* Q\|_\nu^2$, i.e., it is a biased estimate. In fact, from the above decomposition, we see that selecting the policies based on the mean-squared empirical Bellman residual leads to favoring policies for which the variance-like term $\mathbb{E}\left[\|T^* Q - \hat{T}^* Q\|_\nu^2\right]$ is small (see Menache et al. 2005; Antos et al. 2008b).

Our contribution in this paper is in showing how, despite this, the selection of the Bellman error minimizing candidate can be done almost as well as if one would have access to $T^* Q_k$.

Remark 1 In the analysis below, for the sake of simplicity, we assume that Q_ks are fixed deterministic functions. In practice, Q_ks would be estimated

based on some data. In this case, Q_k would become random (data-dependent) functions. However, our results still continue to hold provided that the data $\mathcal{D}_n$ is a fresh data in the sense that the transitions in $\mathcal{D}_n$ are independent of the data used to generate $(Q_k)_{k\geq 1}$ given the first state in $\mathcal{D}_n$. In particular, in this case the results can be stated and proven on the conditional field defined by the sigma algebra generated by $(Q_k)_{k\geq 1}$ and would take essentially the same form as presented here.

4 Algorithm

As mentioned before, thanks to the definition of the empirical Bellman operator $\hat{T}^*$, the regression function underlying

$$\mathcal{D}_{n,k} = \left\{ \left((X_1, A_1), (\hat{T}^* Q_k)(X_1, A_1) \right), \ldots, \left((X_n, A_n), (\hat{T}^* Q_k)(X_n, A_n) \right) \right\} \tag{2}$$

is just T^*Q_k (cf. Proposition 1). Thus, we can feed $\mathcal{D}_{n,k}$ to a regression procedure which in turn, ideally, returns a "good" approximation to T^*Q_k. Let us denote the action-value function returned by such a regression algorithm by $\tilde{Q}_k$. If $\tilde{Q}_k$ is indeed close to T^*Q_k, then by calculating $\|Q_k - \tilde{Q}_k\|_n^2 \approx \|Q_k - \tilde{Q}_k\|_\nu^2 \approx \|Q_k - T^*Q_k\|_\nu^2$ we can base the selection of the action-value function with the smallest Bellman error on computing

$$\mathrm{argmin}_{1\leq k\leq P} \|Q_k - \tilde{Q}_k\|_n^2.$$

The problem with this procedure is that it might be overly optimistic and thus results in an uncontrolled error. To see why this is the case, imagine that for some k_0 for which $\|Q_{k_0} - T^*Q_{k_0}\|_\nu^2$ is "large", the regression procedure returns an estimate such that $\|Q_{k_0} - \tilde{Q}_{k_0}\|_\nu^2 \ll \|Q_{k_0} - T^*Q_{k_0}\|_\nu^2$ (say, the regression procedure might be biased towards action-values close to zero, Q_{k_0} might be close to zero, while $T^*Q_{k_0}$ might be far from zero). As a result, the above procedure will be likely to select k_0, and thus might miss some other index with a lower Bellman error. Basically, the procedure must be guarded against underestimating the Bellman error.

The basis of the new procedure is the following simple inequality:

$$\|Q_k - T^*Q_k\|_\nu^2 \leq 2 \left[\|Q_k - \tilde{Q}_k\|_\nu^2 + \|T^*Q_k - \tilde{Q}_k\|_\nu^2 \right]. \tag{3}$$

The first term of the right-hand side can be estimated by $\|Q_i - \tilde{Q}_i\|_n^2$. Assume further that we are provided with $\bar{b}_k$, a (tight) high-probability upper bound on the second term. Then, the action-value function corresponding to the minimizer of $\|Q_k - \tilde{Q}_k\|_n^2 + \bar{b}_k$ can also be expected to have a small Bellman error. Further, if $\bar{b}_k$ is a tight upper bound, adding $\bar{b}_k$ to the empirical error estimate will not introduce any further bias in addition that was (potentially) already present in the procedure.

There is one more detail that we need to take care of. We would like our procedure to handle situations where the number of candidate action-value functions, P, can be very large, potentially in the range of the sample

size or even larger. This is advantageous since it frees the user from being constrained to a "small" number of action-value function candidates. In fact, we design our procedure so that even $P = \infty$ is allowed. As a consequence of this, we add another penalty term that plays the role of a complexity penalty. The resulting procedure is given in Algorithm 1.

Algorithm 1 BERMIN($\{Q_k\}_{k=1,2,\dots}, \mathcal{D}_n, \text{REGRESS}(\cdot), \delta, a, b$)

1: Split $\mathcal{D}_n$ into two disjoint parts of the same size: $\mathcal{D}_n = \mathcal{D}'_n \cup \mathcal{D}''_n$.
2: **for** $k = 1, 2, \dots$ **do**
3: $\quad (\tilde{Q}_k, \bar{b}_k) \leftarrow \text{REGRESS}(\mathcal{D}'_{n,k}, \delta/3)$
4: $\quad e_k \leftarrow \frac{1}{|\mathcal{D}''_n|} \sum_{(X,A)\in\mathcal{D}''_n} (Q_k(X,A) - \tilde{Q}_k(X,A))^2$
5: $\quad \mathcal{R}_k^{\text{RL}} \leftarrow \frac{1}{(1-a)^2}\, e_k + \bar{b}_k$
6: **end for**
7: $\hat{k} \leftarrow \text{argmin}_{k=1,2,\dots} \left[\mathcal{R}_k^{\text{RL}} + b\, \frac{\log(k)}{(1-a)a\, n}\right]$
8: **return** $\hat{k}$

The algorithm's inputs are the candidate action-value functions, the data-set $\mathcal{D}_n$, a regression procedure REGRESS, a desired error probability δ, and two constants: $0 < a < 1$, and $b > 0$. Here a is a tuning parameter, while $b = \frac{20K\tau}{3}$ where $K = 4B^2(1 + \frac{1}{(1-a)^2})$ and τ is defined in Assumption 1.

In the first line the dataset is split into two disjoint parts. The important point here is that both parts should have $\Theta(n)$ data points. In line 3 the regression procedure is called with the dataset $\mathcal{D}'_{n,k}$ derived from $\mathcal{D}'_n$ using (2) (i.e., $D_{n,k}$ depends on Q_k) and the third of the target error probability. By assumption, the procedure returns both an estimate of T^*Q_k and a bound on the error of the estimate that holds with probability $\delta/3$, simultaneously for all $k \geq 1$. In line 4, the dataset $\mathcal{D}''_n$ is used to estimate $\|Q_k - \tilde{Q}_k\|_\nu^2$. In the next line the two error estimates are combined to yield $\mathcal{R}_k^{\text{RL}}$. In line 7 this estimate is further biased upwards based on which the index of the selected candidate action-value function is calculated. This index becomes the output of the procedure.

Of course, in practice the procedure would be used with $P < +\infty$ action-value function candidates, or the algorithm could be run in an infinite loop or until timeout. With a fixed P, the procedure's computational complexity then depends on the choice of P, the computational complexity of the regression procedure REGRESS and the number of datapoints n.

5 Theoretical Analysis

The next theorem is a general complexity regularization-based model selection result that will be used to derive our RL/Planning model selection theorem. This theorem and its proof technique, is similar to Theorem 3 of Bartlett et al. (2002) with the difference that our result is stated for a more

abstract definition of loss and concentration of empirical risk around loss, thus crystallizing the essence of their result. The proof is given for the sake of completeness. For further similar results on complexity regularization see Barron (1991); Lugosi and Wegkamp (2004).

Theorem 1 (Model Selection Theorem) *Consider a deterministic sequence of losses* L_k $(k = 1, 2, \dots)$ *and the sequence of random variables* $\mathcal{R}_k$ $(k = 1, 2, \dots)$ *such that*

$$\mathbb{P}\left(L_k - (1-a)\mathcal{R}_k > \varepsilon\right) \le c_1 \exp(-c_2\varepsilon), \tag{4}$$

$$\mathbb{P}\left(\frac{1}{1+a}\mathcal{R}_k - \mathbb{E}\left[\mathcal{R}_k\right] > \varepsilon\right) \le c_3 \exp(-c_4\varepsilon), \tag{5}$$

are satisfied for all $\varepsilon > 0$ *with some* $c_1, c_2, c_3, c_4 > 0$ *and* $0 < a < 1$.

Define $\hat{k}$*, the index of the selected model, by*

$$\hat{k} \leftarrow \operatorname{argmin}_{k \ge 1}[\mathcal{R}_k + C_k],$$

for a (deterministic) sequence C_k $(k = 1, 2, \dots)$ *that satisfies*

$$c_5 \stackrel{\text{def}}{=} \sum_{k \ge 1} \exp\left(-c_2(1-a)C_k\right) < \infty, \tag{6}$$

$$c_6 \stackrel{\text{def}}{=} \sum_{k \ge 1} \exp\left(-c_4 \frac{1+2a}{1+a} C_k\right) < \infty. \tag{7}$$

(A) Then

$$L_{\hat{k}} < (1-a^2) \min_{k=1,2,\dots} \left\{\mathbb{E}\left[\mathcal{R}_k\right] + 2C_k\right\} + \frac{2\ln(\frac{2c_1c_5}{\delta})}{c_2} + \frac{2(1-a^2)\ln(\frac{2c_3c_6}{\delta})}{c_4},$$

with probability at least $1 - \delta$.

(B) Define the set of "α*-bad" model indices as*

$$\mathcal{A}_\alpha \stackrel{\text{def}}{=} \left\{k' : L_{k'} > (1-a^2) \min_{k=1,2,\dots} \left\{\mathbb{E}\left[\mathcal{R}_k\right] + 2C_k\right\} + \alpha\right\}.$$

Then it holds that

$$\mathbb{P}\left(\hat{k} \in \mathcal{A}_\alpha\right) \le c_1 c_5 \exp\left(-\frac{c_2\alpha}{2}\right) + c_3 c_6 \exp\left(-\frac{c_4\alpha}{2(1-a^2)}\right).$$

In words, if $\mathcal{R}_k$ is an overestimation of L_k and concentrates fast around its mean then the model selection procedure selects the correct model with overwhelming probability.

Proof Fix some $0 < \alpha_1, \alpha_2$. An elementary reasoning shows that

$$\begin{aligned}
&\mathbb{P}\left(L_{\hat{k}} > (1-a)(1+a)\min_k\{\mathbb{E}\left[\mathcal{R}_k\right] + 2C_k\} + \alpha_1 + \alpha_2\right) \leq \\
&\mathbb{P}\left(L_{\hat{k}} > (1-a)\min_k\{\mathcal{R}_k + C_k\} + \alpha_1\right) \\
&+ \mathbb{P}\left((1-a)\min_k\{\mathcal{R}_k + C_k\} > (1-a)(1+a)\min_k\{\mathbb{E}\left[\mathcal{R}_k\right] + 2C_k\} + \alpha_2\right)
\end{aligned}$$

We upper bound both terms of the right-hand side. For the first term we have:

$$\begin{aligned}
&\mathbb{P}\left(L_{\hat{k}} > (1-a)\min_k\{\mathcal{R}_k + C_k\} + \alpha_1\right) \\
&\leq \mathbb{P}\left(\max_k\{L_k\} > (1-a)\min_k\{\mathcal{R}_k + C_k\} + \alpha_1\right) \\
&\leq \mathbb{P}\left(\max_k\{L_k - (1-a)\mathcal{R}_k - (1-a)C_k\} > \alpha_1\right) \\
&\leq \sum_{k\geq 1}\mathbb{P}\left(L_k - (1-a)\mathcal{R}_k > \alpha_1 + (1-a)C_k\right) \\
&\leq \sum_{k\geq 1} c_1 \exp\left(-c_2\left\{\alpha_1 + (1-a)C_k\right\}\right) \\
&= \left(c_1 \sum_{k\geq 1}\exp\left(-c_2(1-a)C_k\right)\right)\exp\left(-c_2\alpha_1\right) \\
&= c_1 c_5\, \exp\left(-c_2\alpha_1\right).
\end{aligned}$$

where we used the fact that $\max_\theta f(\theta) + \max_\theta(-g(\theta)) \leq \max_\theta(f(\theta) - g(\theta))$ in the second inequality, and the union bound in the third inequality.

For the second term, we can write:

$$
\begin{aligned}
&\mathbb{P}\left((1-a)\min_k\{\mathcal{R}_k + C_k\} > (1-a)(1+a)\min_k\{\mathbb{E}\left[\mathcal{R}_k\right] + 2C_k\} + \alpha_2\right) \\
&\leq \mathbb{P}\left((1-a)\max_k\{\mathcal{R}_k + C_k\} > (1-a)(1+a)\min_k\{\mathbb{E}\left[\mathcal{R}_k\right] + 2C_k\} + \alpha_2\right) \\
&\leq \mathbb{P}\left(\max_k\left\{\mathcal{R}_k + C_k - (1+a)\mathbb{E}\left[\mathcal{R}_k\right] - 2(1+a)C_k\right\} > \frac{\alpha_2}{1-a}\right) \\
&\leq \mathbb{P}\left(\max_k\left\{\frac{1}{1+a}\mathcal{R}_k - \mathbb{E}\left[\mathcal{R}_k\right] - \frac{1+2a}{1+a}C_k\right\} > \frac{\alpha_2}{(1-a)(1+a)}\right) \\
&\leq \sum_{k\geq 1}\mathbb{P}\left(\frac{1}{1+a}\mathcal{R}_k - \mathbb{E}\left[\mathcal{R}_k\right] > \frac{\alpha_2}{(1-a)(1+a)} + \frac{1+2a}{1+a}C_k\right) \\
&\leq \sum_{k\geq 1} c_3 \exp\left(-c_4\left(\frac{\alpha_2}{(1-a)(1+a)} + \frac{1+2a}{1+a}C_k\right)\right) \\
&= c_3 c_6 \ \exp\left(-\frac{c_4\alpha_2}{(1-a)(1+a)}\right)
\end{aligned}
$$

Part (B) follows from combining the inequalities obtained so far and if $\alpha_1 = \alpha_2 = \alpha/2$. To prove part (A) choose α_1, α_2 such that

$$c_1 c_5 \ \exp\left(-c_2\alpha_1\right) = c_3 c_6 \ \exp\left(-\tfrac{c_4\alpha_2}{(1-a)(1+a)}\right) = \frac{\delta}{2}$$

to get

$$L_{\hat{k}} < (1-a^2)\min_{k=1,2,\ldots}\{\mathbb{E}\left[\mathcal{R}_k\right] + 2C_k\} + \frac{2\ln(\frac{2c_1c_5}{\delta})}{c_2} + \frac{2(1-a^2)\ln(\frac{2c_3c_6}{\delta})}{c_4}$$

with probability at least $1-\delta$. □

5.1 Proof of the Main Result

In this section we state and prove the main result of the paper concerning BERMIN defined in Section 4.

We prove the result under the following assumptions:

Assumption 1 *Assume that the following hold:*

1. *The standard offline sampling assumptions are satisfied by the dataset $\mathcal{D}_n$ and the time-homogeneous Markov chain $X_1, X_2, \ldots, X_n$ uniformly quickly forgets is past with a forgetting time of τ (cf. Definition 7 in Appendix B);*
2. *The functions Q_k, $\tilde{Q}_k$, T^*Q_k $(k \geq 1)$ are bounded by a deterministic quantity $B > 0$;*
3. *The functions Q_k $(k \geq 1)$ are deterministic;*

4. *Simultaneously, for all* $k \geq 1$, $\|\tilde{Q}_k - T^* Q_k\|_\nu^2 \leq \bar{b}_k$ *holds with probability at least* $1 - \delta/3$, *where* $\bar{b}_k$ *is* $\sigma(\mathcal{D}'_n)$*-measurable;*
5. $\bar{b}_k \leq 4B^2$ *holds a.s.;*
6. $\mathcal{D}''_n$ *holds* n *datapoints.*

A couple of remarks are in order on these assumptions:

Remark 2 The he standard offline sampling assumptions were discussed in Section 2.2. The additional assumption here demands that the Markov chain should be "fast mixing". The actual definition, which we think is often satisfied, is somewhat technical and is given in the appendix. Here we note that this condition is satisfied if the Markov chain is uniformly ergodic (or, in other words, if the so-called Doeblin condition holds for it (Meyn and Tweedie, 1993)). Note that if the chain mixes but the "mixing rate" is slow, the result presented below would still hold, but with a possibly worse rate.

Remark 3 If the immediate rewards are bounded with probability one, most algorithms would return value functions which are bounded by some deterministic quantity. If this is not known, but a bound $r_{\max}$ on the immediate reward function is known then boundedness can be achieved by truncating the value functions Q_k, $\tilde{Q}_k$ so that they take values in the interval $[-B, B] = [-r_{\max}/(1-\gamma), r_{\max}/(1-\gamma)]$ (i.e., instead of $Q_k(x,a)$, using $\min(\max(Q_k(x,a), -B), B)$). As far as the approximation of Q^* is considered this introduces no loss of quality since all value functions are known to take values in this interval.

Remark 4 In Remark 1, we have already commented on how to lift the assumption that the functions Q_k are deterministic.

Remark 5 The high probability estimate $\bar{b}_k$ can be obtained as follows: Fix k. It is enough to construct a high probability estimate for $\|\tilde{Q}_k - T^* Q_k\|_\nu^2$. since the estimates can be combined using a union bound in the standard way. Let $\|f\|_{\mathcal{D}''_n}^2$ denote the empirical 2-norm of f measured on data $\mathcal{D}''_n$. We have

$$\begin{aligned}\|\tilde{Q}_k - T^* Q_k\|_\nu^2 &= \mathbb{E}\left[\|\tilde{Q}_k - T^* Q_k\|_{\mathcal{D}''_n}^2 \,|\, \mathcal{D}'_n\right] \\ &= \mathbb{E}\left[\|\tilde{Q}_k - \hat{T}^* Q_k\|_{\mathcal{D}''_n}^2 \,|\, \mathcal{D}'_n\right] - \mathbb{E}\left[\|T^* Q_k - \hat{T}^* Q_k\|_{\mathcal{D}''_n}^2\right].\end{aligned}$$

Here, $\mathbb{E}\left[\|\tilde{Q}_k - \hat{T}^* Q_k\|_{\mathcal{D}''_n}^2 \,|\, \mathcal{D}'_n\right]$ can be estimated based using $\bar{b}_{k,1} = \|\tilde{Q}_k - \hat{T}^* Q_k\|_{\mathcal{D}''_n}^2$. The rate of convergence of $\bar{b}_{k,1}$ is parametric (i.e., fast). To estimate $L_k^* = \mathbb{E}\left[\|T^* Q_k - \hat{T}^* Q_k\|_{\mathcal{D}''_n}^2\right]$, following Devroye et al. (2003), we may use $\mathcal{D}'_n$ to construct a consistent estimator $\tilde{Q}'_k$ of $T^* Q_k$ that converges with an adaptive, fast rate. Then use the empirical error $\bar{b}_{k,2}$ of this estimator measured on $\mathcal{D}''_n$ as the estimate of L_k^*. Following the steps of Theorem 3.1of Devroye et al. (2003), together with a noncentral concentration inequality (cf. Appendix B), one can see that $\bar{b}_{k,2}$ converges at the same rate as the

consistent estimator of T^*Q_k. Hence, $\bar{b}_k = \bar{b}_{k,1} - \bar{b}_{k,2}$ also assumes a fast rate. For some other ideas for data-dependent bounds see e.g. Lugosi and Wegkamp (2004).

Remark 6 Of course, the success of BERMIN will depend critically on the quality of the regression procedure used. If algorithm $\mathcal{A}$ is used to compute Q_k with a function set $\mathcal{F}(p_k)$ then a minimum requirement for the regression procedure used to compute $\tilde{Q}_k$ is that its guaranteed error should not be larger than that of using $\mathcal{A}$ with $\gamma = 0$, $\mathcal{F}(p_k)$ and the dataset

$$\mathcal{D}'_{n,k} = \{((X_1, A_1, R'_1, Y_1), \ldots, ((X_n, A_n, R'_i, Y_n)\}$$

and $R'_i = (\hat{T}^*Q_k)(X_i, A_i)$, $i = 1, \ldots, n$ (that is, for this problem the immediate reward function is T^*Q_k). This requirement will thus always be met if one in fact uses algorithm $\mathcal{A}$ to compute $\tilde{Q}_k$ with the above settings. However, one should not be limited to this choice. In fact, knowing that $\mathcal{A}$ uses $\mathcal{F}(p_k)$, it makes sense to use a regression procedure that is made adaptive for the sequence $(\mathcal{F}(p_k))_{k\geq 1}$. This can be done based on Theorem 1 and many other ways (for some recent work on this, see Wegkamp 2003; van der Vaart et al. 2006; Arlot and Celisse 2009.

Our main result is as follows:

Theorem 2 (Model Selection for RL/Planning) *Let Assumption 1 hold, $0 < a < 1$ and consider the algorithm* BERMIN *defined in Section 4. Assume that the constant b in the procedure is selected to be*

$$b = \frac{20K\tau}{3},$$

where $K = 4B^2(1 + \frac{1}{(1-a)^2})$. Let $\hat{k}$ be the index selected by BERMIN. *Then, with probability at least $1 - \delta$,*

$$\|Q_{\hat{k}} - T^*Q_{\hat{k}}\|_\nu^2 \leq 4(1-a^2)\min_{k\geq 1}\left\{\frac{2}{(1-a)^2}\|Q_k - T^*Q_k\|_\nu^2 + \frac{3}{(1-a)^2}\bar{b}_k + 2C_k\right\} + 34K\tau\frac{\ln(6.3/\delta)}{an},$$

where $C_k = b\log(k)/(a(1-a)n)$.

The result will be discussed after the proof.

Proof Let

$$\mathcal{R}_k = \frac{1}{(1-a)^2}\|\tilde{Q}_k - Q_k\|_n^2 + \bar{b}_k, \quad L_k = \|\tilde{Q}_k - Q_k\|_\nu^2 + (1-a)\bar{b}_k,$$

and $\bar{\mathcal{R}}_k = \mathbb{E}\left[\mathcal{R}_k \,|\, \mathcal{D}'_n\right]$. Hence,

$$\bar{\mathcal{R}}_k = \frac{1}{(1-a)^2}\|\tilde{Q}_k - Q_k\|_\nu^2 + \bar{b}_k$$

and the index returned by the algorithm is defined by

$$\hat{k} = \operatorname{argmin}_{k \geq 1} \left[\mathcal{R}_k + C_k \right],$$

where

$$C_k = \frac{20 K \tau \, \ln k}{3\, a(1-a)\, n}.$$

By our assumptions $\mathcal{R}_k \geq 0$ concentrates around $\bar{\mathcal{R}}_k$ and is bounded by $K = 4B^2(1 + \frac{1}{(1-a)^2})$. Hence, $\operatorname{Var}\left[\mathcal{R}_k | \mathcal{D}'_n\right] \leq \mathbb{E}\left[\mathcal{R}_k^2 | \mathcal{D}'_n\right] \leq K \mathbb{E}\left[\mathcal{R}_k | \mathcal{D}'_n\right]$. Thus, thanks to the assumptions on $\mathcal{D}''_n$, $\bar{\mathcal{R}}_k$ and the definitions of $\mathcal{R}_k, \bar{\mathcal{R}}_k$, Lemma 3 of Appendix C applied to the conditional probability field defined by $\mathcal{D}'_n$ gives that

$$\mathbb{P}\left(\frac{1}{1+a} \mathcal{R}_k - \bar{\mathcal{R}}_k > \varepsilon \;\middle|\; \mathcal{D}'_n \right) \leq \exp(-c' a\, n \varepsilon) \tag{8}$$

holds for $c' = (1+a)/(K'(1+2/3a)) \geq 3/(5K')$ with $K' = K\tau$. Also,

$$\begin{aligned}
&\mathbb{P}\left(L_k - (1-a)\mathcal{R}_k > \varepsilon \;\middle|\; \mathcal{D}'_n \right) \\
&\quad = \mathbb{P}\left(\|\tilde{Q}_k - Q_k\|_\nu^2 + (1-a)\bar{b}_k - \varepsilon > \frac{1}{1-a} \|\tilde{Q}_k - Q_k\|_n^2 + (1-a)\bar{b}_k \;\middle|\; \mathcal{D}'_n \right) \\
&\quad = \mathbb{P}\left(\|\tilde{Q}_k - Q_k\|_\nu^2 - \varepsilon > \frac{1}{1-a} \|\tilde{Q}_k - Q_k\|_n^2 \;\middle|\; \mathcal{D}'_n \right)
\end{aligned}$$

Thus, again by Lemma 2,

$$\mathbb{P}\left(L_k - (1-a)\mathcal{R}_k > \varepsilon \;\middle|\; \mathcal{D}'_n \right) \leq \exp(-c a\, n \varepsilon), \tag{9}$$

where $c = (1-a)/(K'(1+2/3a)) \geq 3(1-a)/(5K')$. Inequalities (8),(9) show that the conditions of Theorem 1 are satisfied by L_k, $\mathcal{R}_k$ with $c_1 = c_3 = 1$, $c_2 = 3an/(5K')$, $c_4 = 3(1-a)an/(5K')$ and if all expectations and probabilities conditioned on $\mathcal{D}'_n$, provided that $C_k \geq 2\ln k/(c_2(1-a))$ and $C_k \geq 2\frac{1+a}{c_4(1+2a)} \ln k$ hold true. However, since $2\ln k/(c_2(1-a)) \leq 10K'/(3an(1-a)) \ln k$ and $2\frac{1+a}{c_4(1+2a)} \ln k \leq 20K'/(3an(1-a)) \ln k$, these conditions are satisfied thanks to the choice of C_k. In particular, we have $c_5 = c_6 = \pi^2/6 \leq 1.05$.

Now, by Part (A) of Theorem 1,

$$L_{\hat{k}} \leq (1-a^2) \min_{k \geq 1} \left[\frac{1}{(1-a)^2} \|\tilde{Q}_k - Q_k\|_\nu^2 + \bar{b}_k + 2C_k \right] + \Delta_1, \tag{10}$$

holds with probability $1 - \delta/3$. Here

$$\Delta_1 = \frac{2\ln(\frac{2c_5}{\delta/3})}{c_2} + \frac{2(1-a^2)\ln(\frac{2c_6}{\delta/3})}{c_4} \leq 10K' \frac{\ln(6.3/\delta)}{an}.$$

From now on consider the event $\mathcal{E}$ when the inequality (10), the inequalities

$$\|\tilde{Q}_k - T^* Q_k\|_\nu^2 \le \bar{b}_k, \quad k \ge 1, \tag{11}$$

and

$$\hat{k} \notin \mathcal{A}_\alpha$$

all hold simultaneously. Here $\alpha > 0$ will be chosen later and

$$\mathcal{A}_\alpha = \left\{ k' : L_{k'} > (1-a^2) \min_{k \ge 1} \left[\frac{1}{(1-a)^2} \|\tilde{Q}_k - Q_k\|_\nu^2 + \bar{b}_k + 2C_k \right] + \alpha \right\}.$$

Since by definition, $L_{\hat{k}} \ge \|\tilde{Q}_{\hat{k}} - Q_{\hat{k}}\|_\nu^2$, we have

$$\|\tilde{Q}_{\hat{k}} - Q_{\hat{k}}\|_\nu^2 \le (1-a^2) \min_{k \ge 1} \left[\frac{1}{(1-a)^2} \|\tilde{Q}_k - Q_k\|_\nu^2 + \bar{b}_k + 2C_k \right] + \Delta_1.$$

Chaining this with the elementary inequality

$$\|Q_{\hat{k}} - T^* Q_{\hat{k}}\|_\nu^2 \le 2 \left(\|Q_{\hat{k}} - \tilde{Q}_{\hat{k}}\|_\nu^2 + \|\tilde{Q}_{\hat{k}} - T^* Q_{\hat{k}}\|_\nu^2 \right),$$

we get

$$\begin{aligned} &\|Q_{\hat{k}} - T^* Q_{\hat{k}}\|_\nu^2 \le \\ &\quad \le 2(1-a^2) \min_{k \ge 1} \left[\frac{1}{(1-a)^2} \|\tilde{Q}_k - Q_k\|_\nu^2 + \bar{b}_k + 2C_k \right] \\ &\qquad + 2\Delta_1 + 2\|\tilde{Q}_{\hat{k}} - T^* Q_{\hat{k}}\|_\nu^2. \end{aligned}$$

Since $\|\tilde{Q}_{\hat{k}} - T^* Q_{\hat{k}}\|_\nu^2 \le L_{\hat{k}}$ and thanks to $\hat{k} \notin \mathcal{A}_\alpha$,

$$L_{\hat{k}} \le (1-a^2) \min_{k \ge 1} \left[\frac{1}{(1-a)^2} \|\tilde{Q}_k - Q_k\|_\nu^2 + \bar{b}_k + 2C_k \right] + \alpha,$$

we get

$$\|Q_{\hat{k}} - T^* Q_{\hat{k}}\|_\nu^2 \le 4(1-a^2) \min_{k \ge 1} \left[\frac{1}{(1-a)^2} \|\tilde{Q}_k - Q_k\|_\nu^2 + \bar{b}_k + 2C_k \right] + 2(\Delta_1 + \alpha).$$

Now, by (11),

$$\|\tilde{Q}_k - Q_k\|_\nu^2 \le 2 \left(\|Q_k - T^* Q_k\|_\nu^2 + \bar{b}_k \right).$$

Hence,

$$\begin{aligned} &\|Q_{\hat{k}} - T^* Q_{\hat{k}}\|_\nu^2 \le \\ &\quad \le 4(1-a^2) \min_{k \ge 1} \left[\frac{2}{(1-a)^2} \|Q_k - T^* Q_k\|_\nu^2 + \frac{3}{(1-a)^2} \bar{b}_k + 2C_k \right] \\ &\qquad + 2(\Delta_1 + \alpha). \end{aligned} \tag{12}$$

It remains to choose α. By Part (B) of Theorem 1, $\mathbb{P}\left(\hat{k} \in \mathcal{A}_\alpha \,|\, \mathcal{D}'_n\right)$ is "small":

$$\begin{aligned}\mathbb{P}\left(\hat{k} \in \mathcal{A}_\alpha \,|\, \mathcal{D}'_n\right) &\le c_5 \exp\left(-\frac{c_2\alpha}{2}\right) + c_6 \exp\left(-\frac{c_4\alpha}{2(1-a^2)}\right) \\ &\le 1.05 \exp\left(-\min(\tfrac{c_2}{2}, \tfrac{c_4}{2(1-a^2)})\,\alpha\right) \\ &\le 1.05 \exp\left(-\alpha 3an/(20K')\right).\end{aligned}$$

Now, choose α such that $1.05 \exp\left(-3\alpha an/(20K')\right) = \delta/3$. This gives $\alpha = 20K' \ln(3.15/\delta)/(3an)$.

Thus, with this choice of α, and by the union bound, $\mathbb{P}(\mathcal{E}) \ge 1-\delta$. Plugging into (12) the value of α obtained, as well as the upper bound on Δ_1 that was obtained earlier, after some simplification we get the desired result:

$$\begin{aligned}&\|Q_{\hat{k}} - T^* Q_{\hat{k}}\|_\nu^2 \le \\ &\quad 4(1-a^2) \min_{k\ge 1}\left[\frac{2}{(1-a)^2}\|Q_k - T^*Q_k\|_\nu^2 + \frac{3}{(1-a)^2}\bar{b}_k + 2C_k\right] \\ &\qquad + 34K' \frac{\ln(6.3/\delta)}{an}.\end{aligned}$$

□

Remark 7 The result also holds true for policy evaluation when the goal is to select a Q_k that minimizes the Bellman error $\|T^\pi Q_k - Q_k\|_\nu$ for any given policy π assuming that in the definition of the dataset $\hat{T}^\pi$ is used in place of $\hat{T}^*$. In fact, the only property of $\hat{T}^*$ that we used in the proof was the property in Proposition 1.

5.2 Adaptivity

Assume that $\mathcal{A}$ satisfies w.p. $1-\delta$

$$\|Q_k - T^*Q_k\|_\nu^2 \le a(p_k, T^*) + c_{T^*} b(p_k, n, \ln(1/\delta)) \tag{13}$$

for any fixed value of $k \ge 1$. Here $a(p_k, T^*)$ does not depend on n and δ and captures how well $\mathcal{F}(p_k)$ approximates the fixed-point of T^*. Further, b does not depend on T^* and bounds the so-called estimation error. We also assume that $c_{T^*} \le C^* V_{\max}$ with some $C^* > 0$. One expects that $a(p_k, T^*)$ should satisfy

$$a(p, T^*) \le a \inf_{Q \in \mathcal{F}(p)} \|Q - T^*Q\|_\nu^2. \tag{14}$$

for some $a > 0$. This, however, might be difficult to achieve ($a(p, T^*)$ could be larger), hence we will only assume this when $\gamma = 0$. This is reasonable, as in this case the problem reduces to regression for which bounds satisfying

this property are available. We further assume that $a^* = \inf_{k\geq 1} a(p_k, T^*) = 0$. This is reasonable when $\cup_k \mathcal{F}(p_k)$ is so large that it covers essentially all the functions.

We further assume that $b(p_k, n, \ln(1/\delta)) \to 0$ when $n \to \infty$, it is monotonically increasing in p_k, and there exists $c > 0$ s.t. for any $p \leq p'$, $0 < \delta < 1$,

$$\limsup_{n\to\infty} \frac{b(p, n, \ln(1/\delta))}{b(p', n, \ln(1/\delta))} \leq c,$$

i.e., a larger value of p' (leading to a more complex function space) leads to a slower rate. Bounds satisfying these properties are available for e.g. LSPI (see Antos et al. (2008b)).

Assume that the bound (13) is tight– at least up to a constant factor and asymptotically. Because of the tightness of the bound, the best we can hope to achieve is

$$\beta_n = \inf_{k\geq 1} \Big[a(p_k, T^*) + c_{T^*}\, b(p_k, n, \ln(1/\delta)) \Big].$$

Let $\bar{b}_{k,n}(\delta)$ be the estimate returned by $\mathcal{A}$ on its loss (we assume that $\mathcal{A}$ can return this). Hence, for any fixed $k \geq 1$, $0 < \delta < 1$, we have that

$$\|Q_k - T^* Q_k\|_\nu^2 \leq \bar{b}_{k,n}(\delta) \tag{15}$$

holds with probability at least $1 - \delta$. We assume that $\bar{b}_{k,n}(\delta)$ is tight in the sense that for any $0 < \delta < 1$, $k \geq 1$,

$$\bar{b}_{k,n}(\delta) \leq C \Big[a(p_k, T^*) + c_{T^*}\, b(p_k, n, \ln(1/\delta)) \Big] \tag{16}$$

for some $C > 0$.

Our goal is to bound the performance of BERMIN when $\mathcal{A}$ is used in place of REGRESS as suggested in Remark 6. Let T_k^* be the operator defined by $T_k^* Q = Q_k$ (hence, the operator returns the constant function Q_k). Assume that upon the k^{th} call, we pass $\frac{\delta}{4.15k^2}$ as the confidence value to $\mathcal{A}$ (which now must accept such a parameter).

From Theorem 2, w.p. $1 - \delta$,

$$\left\|Q_{\hat{k}} - T^* Q_{\hat{k}}\right\|_\nu^2 \leq \min_{k\geq 1} \Big[c_1 \left\|Q_k - T^* Q_k\right\|_\nu^2 + c_2 \bar{b}_{k,n}(\delta/(c_3 k^2)) + 2C_k(n) \Big] \tag{17}$$

holds for some constants $c_1, c_2, c_3 > 0$. Note that one has

$$C_k(n) = O(b(p_k, n, \ln(ck^2/\delta))),$$

since $b(p_k, n, \ln(ck^2/\delta))$ is at best a parametric rate.

By relations (14) and (16) and since $c_{T_k^*} \le C^* V_{\max}$,

$$\begin{aligned}\bar{b}_{k,n}(\delta/(ck^2)) &\le C\left[a \inf_{Q\in\mathcal{F}(p_k)} \|Q - Q_k\|_\nu^2 + c_{T_k^*} b(p_k, n, \ln(c_3 k^2/\delta))\right] \\ &\le 4C\left[a\,\|Q_k - T^* Q_k\|_\nu^2 + C^* V_{\max} b(p_k, n, \ln(c_3 k^2/\delta))\right]\end{aligned}$$

where we used $\|Q - Q_k\|_\nu^2 \le 2(\|Q - T^*Q_k\|_\nu^2 + \|Q_k - T^*Q_k\|_\nu^2)$. Plugging into (17), we get with some constant $C', C'' > 0$, that the following inequalities hold simultaneously for $k \ge 1$ w.p. $1-\delta$:

$$\begin{aligned}\left\|Q_{\hat{k}} - T^* Q_{\hat{k}}\right\|_\nu^2 &\le C' \inf_{k\ge 1}\left[\,\|Q_k - T^* Q_k\|_\nu^2 + \, b(p_k, n, \ln(c_3 k^2/\delta)) + C_k(n)\right] \\ &\le \underbrace{C'' \inf_{k\ge 1}\left[a(p_k, T^*) + b(p_k, n, \ln(c_3 k^2/\delta)) + C_k(n)\right]}_{\beta'_n}.\end{aligned}$$

Here in the last step we used (13).

Thus, we have the following corollary:

Corollary 1 *Let the same assumptions of this section, as well as those of Theorem 2 hold. If* Regress $= \mathcal{A}$ *with* $\gamma = 0$ *as suggested above, then with this choice* BErMin *is adaptive in the sense that the rate* β'_n *is essentially the same as that of* β_n*. Precisely, for any* $S_n \to \infty$, $\limsup_{n\to\infty} \beta'_n/(S_n\beta_n) \le 1$.

Note that instead of $\mathcal{A}$, we can use any other sufficiently powerful regression method. What the corollary here tells us is that it is not worthwhile to use a very expensive procedure. The intuitive explanation is that it is not worthwhile to spend resources on a high quality estimation of the error of a low quality action-value function.

6 Discussion

In reinforcement learning, the goal is to find a well-performing policy given some data. Of course, the problem of automatic parameter tuning comes up in this setting too. As discussed beforehand, the results of this paper are indirectly applicable to find well-performing policies if one wishes to search for a policy by first finding an action-value function with a small Bellman error. Are there any alternatives to this approach?

One approach is as follows. As before, assume that the possible parameter settings are enumerated. Assume that the policy obtained when using parameters p_k is π_k. Then the problem eventually boils down to selecting the best performing policy given the list $\pi_1, \pi_2, \ldots$. We can also assume that we are given some "fresh" data (on which the policies do not depend). Let the performance be measured with respect to some fixed, known initial state distribution, ρ. Hence, the problem is to find

$$k^* = \mathrm{argmax}_{k\ge 1} V_\rho^{\pi_k},$$

where

$$V_\rho^{\pi_k} = \int V^{\pi_k}(x)\, d\rho(x).$$

There are at least two quite distinct approaches that look useful for this purpose. The first approach is to estimate a model of the MDP based on the data and then use it to approximately evaluate the policies (disregarding computational issues, policy evaluation with respect to an initial distribution can be done exactly given a generative model using straightforward Monte-Carlo policy evaluation, assuming that one can sample from the initial state distribution). The second approach is to estimate the value function underlying the policies without resorting to a model. Let us consider this approach first.

Let the value function estimated for policy π_k be $\hat{V}^{\pi_k}$. Then

$$\Delta_k = |\hat{V}_\rho^{\pi_k} - V_\rho^{\pi_k}| \le \|\hat{V}^{\pi_k} - V^{\pi_k}\|_{L^1(\rho)}. \tag{18}$$

Hence, one way of controlling the error of estimating $V_\rho^{\pi_k}$ (which is necessary for the success of the selection procedure), is to control the error of estimating V^{π_k}. The value function of a policy can be estimated directly by using e.g. LSTD (Bradtke and Barto, 1996; Boyan, 2002). As usual, LSTD requires a well-chosen function approximation technique. The problem of choosing the appropriate parameters (function approximation) is a perfect fit for our procedure (cf. Remark 7). Then, one can imagine using the techniques of this paper to derive performance bounds. However, the details of this remain to be done as a future work.

Let us now investigate the other approach when a model is estimated first. Let $\hat{M}$ be the model that is estimated based on the data (again, details of how this should be done are not of our concern here and are thus left unspecified). The inequality (18) still applies, just now $\hat{V}_\rho^{\pi_k}$ is the value of π_k in the model $\hat{M}$ and $\hat{V}^{\pi_k}$ is its value function in $\hat{M}$ (as said before, we assume for simplicity that these are computed exactly). The basic question then is how the model's inaccuracies propagates to $\|\hat{V}^{\pi_k} - V^{\pi_k}\|_{L^1(\rho)}$.

The simplest (standard) approach to investigate this is to exploit that the Bellman operators are contractions in the supremum norm. However, this leads to conservative bounds (in particular, the supremum tends to propagate into how the accuracy of $\hat{M}$ should be measured). Here, we show an alternative that avoids supremum norms.

We start with the following lemma, which is essentially Lemma 5.16 by Szepesvári (2001):

Lemma 1 *Let B be a Banach-space with norm $\|\cdot\|$. Pick some $V \in B$. Let $T_1, T_2 : B \to B$, $U_s = T_2^s V$. Assume that T_1, T_2 satisfy the following:* (i) $\lim_{s\to\infty} \|U_s - V_2^*\| \to 0$ *and V_2^* is a fixed point of T_2;* (ii) *T_1 is a γ-contraction in $\|\cdot\|$ with fixed point V_1^*. Then, with $\alpha = \sup_{s\ge 1} \|T_1 U_s - T_2 U_s\|$,*

$$\|V_1^* - V_2^*\| \le \frac{\alpha}{1-\gamma}.$$

Proof See the proof of Lemma 5.16 by Szepesvári (2001). □

Let us now return to bounding $\|\hat{V}^{\pi_k} - V^{\pi_k}\|_{L^1(\rho)}$. Let $\hat{\rho}_k$ be the stationary distribution of policy π_k in the model $\hat{M}$.

We make some (simplifying) assumptions, the discussion of which goes beyond the scope of the present paper given that our goal here is to merely illustrate the nature of the bounds for a model-based approach in an ideal situation. The assumptions are: First, we assume that there is a common upper bound on the densities $d\hat{\rho}_k/d\mu$, where μ is an appropriate common reference measure. We also assume that ρ admits a density with respect to $\hat{\rho}_k$ which is bounded in $\|\cdot\|_{L^2(\rho)}$ independently of k. Let $C > 0$ be an upper bound on these quantities. In addition, we assume that the transition models assume a density (both for the estimated and the true models) with respect to some reference measure, which for simplicity we take to be μ. Further, we assume that $\mu(\mathcal{X})$ is finite.[2]

Using the Cauchy-Schwartz inequality, we get

$$\|\hat{V}^{\pi_k} - V^{\pi_k}\|_{L^1(\rho)} \le C\|\hat{V}^{\pi_k} - V^{\pi_k}\|_{L^2(\hat{\rho}_k)}.$$

Now, use Lemma 1 with the Banach-space $B = L^2(\mathcal{X}, \hat{\rho}_k)$, choosing T_1 to be the policy evaluation operator of π_k in the model $\hat{M}$, T_2 to be the policy evaluation operator of π_k in the true MDP M and $V = 0$. Then $U_s = T_2^s V$ converges to V^{π_k} (i.e., $V_2^* = V^{\pi_k}$), T_1 is a γ-contraction in the norm of B (by a simple extension of Lemma 6.4 of Bertsekas and Tsitsiklis (1996)) and its unique fixed point is $V_1^* = \hat{V}^{\pi_k}$. Let $\alpha = \sup_{s\ge 1} \|T_1 U_s - T_2 U_s\|$. Then, by the above lemma,

$$\|\hat{V}^{\pi_k} - V^{\pi_k}\|_{L^2(\hat{\rho}_k)} \le \frac{\alpha}{1-\gamma}.$$

Thus, it remains to bound α.

By the triangle inequality,

$$\|T_1 V - T_2 V\|_{L^2(\hat{\rho}_k)} \le \|\hat{r}^{\pi_k} - r^{\pi_k}\|_{L^2(\hat{\rho}_k)} + \|(\hat{P}^{\pi_k} - P^{\pi_k})V\|_{L^2(\hat{\rho}_k)},$$

where $\hat{r}$ ($\hat{P}$) is the immediate reward function (resp., transition model) in $\hat{M}$, r (P) is the immediate reward function (resp., transition model) in M, and upper indexing with the policy means that the action selection is dictated by policy π_k. Now, by Jensen's inequality,

$$\begin{aligned}
\|(\hat{P}^{\pi_k} - P^{\pi_k})V\|^2_{L^2(\hat{\rho}_k)} &\le \int\int \left(\hat{p}(y|x,\pi_k(x)) - p(y|x,\pi_k(x))\right)^2 V^2(y) d\mu(y) d\hat{\rho}_k(x) \\
&\le K \int\int \delta_p(y|x)^2 d\mu(y) d\hat{\rho}_k(x) \\
&\le CK\|\delta_p\|^2_{L^2(\mu\times\mu)},
\end{aligned}$$

[2] The reasoning below allows for other conditions under which the conclusions essentially hold. Again, since our goal here is to illustrate the possibilities, we do not discuss these here.

where $\delta_p(y|x) = \max_{a\in\mathcal{A}} |\hat{p}(y|x,a) - p(y|x,a)|$ and K is a bound on the magnitude of the values of V.

Putting together the inequalities we get that for any $k \geq 1$,

$$\|\hat{V}^{\pi_k} - V^{\pi_k}\|_{L^1(\rho)} \leq \varepsilon(\hat{M}) \stackrel{\text{def}}{=} \frac{C'}{1-\gamma} \left\{ \|\hat{r}^{\pi_k} - r^{\pi_k}\|_{L^2(\mu)} + \|\delta_p\|_{L^2(\mu\times\mu)} \right\} \tag{19}$$

for an appropriate constant $C' > 0$. (In fact, this bound holds for any policy (not just $\pi_1, \pi_2, \ldots$).) The nice feature of this bound is that supremum-norm of the errors are avoided.

Let the estimated model be $\hat{M}_n$ (assuming n transitions are used to estimate it). Choose the policy based on

$$\hat{k} = \text{argmax}_{k\geq 1} \hat{V}_\rho^{\pi_k}, \tag{20}$$

where the value functions are computed in $\hat{M}_n$. Then

$$V_\rho^{\pi_{\hat{k}}} \geq \hat{V}_\rho^{\pi_{\hat{k}}} - \varepsilon(\hat{M}_n) \geq \hat{V}_\rho^{\pi_{k^*}} - \varepsilon(\hat{M}_n) \geq V_\rho^{\pi_{k^*}} - 2\varepsilon(\hat{M}_n).$$

Here, the first and the last inequalities follow from (19), while the second follows from (20). Thus, we get the following theorem:

Theorem 3 *Under the stated conditions, the loss due to using the approximate models in the model selection procedure is bounded by* $2\varepsilon(\hat{M}_n)$*, where* $\varepsilon(\hat{M})$ *is defined by* (19).

Thus, in order to keep the error small (in addition to satisfying the conditions) one should make every effort to keep the model identification error $\varepsilon(\hat{M}_n)$ small. The best approach is to use an adaptive procedure. Note that model identification is a supervised learning problem, so when identifying a model one may resort to standard approaches. Although the result makes the error of the policy selection procedure explicit, it does not lead to an easy to evaluate recommendation on whether a model-based approach should be followed. If the message of the paper is taken seriously, one should perhaps develop an adaptive procedure to decide this.

Remark 8 Model-based approaches to reinforcement learning and planning have been in use for a while. The works of Brafman and Tennenholtz (2003); Strehl and Littman (2008); Sutton et al. (2008); Farahmand et al. (2009c) are examples of this approach. However simple it is, none of these methods consider using the model to simulate sample trajectories for the goal of policy selection.

Remark 9 Policy search has been analyzed in the literature in some previous works. Ng and Jordan (2000) and Bartlett and Tewari (2007) give finite time bounds for policy search, but only in the case when interaction with the controlled system is allowed, or, equivalently, when an accurate model of the environment is available. They do not consider how the search should be organized. In the book by Chang et al. (2008) some specific policy

search methods are considered, with guaranteed convergence. However, no rates are derived here (i.e., the tradeoff introduced by the choice of Π is not considered). Local methods based on policy gradient are discussed in Section 3.4 of the survey by Szepesvári (2009). However, such local methods may get stuck in local minima and hence they fail to be consistent.

7 Conclusion

In this paper we focused on principled ways for tuning the parameters of reinforcement learning algorithms. More specifically, we focussed on the off-line, non-interactive case and when the goal is to find an action-value function whose Bellman error is small and when the parameters influence the choice of the function space. Our main result shows that the performance of the proposed method is almost as good as that of an oracle which picks the best parameters based on the knowledge of the MDP. We then discussed the related problem of finding a good policy and gave a result that shows how the model identification propagates into a bound on the performance of a method that uses a learned model to evaluate candidate policies and then picks the best. Earlier, but recent works where the flexibility of the function approximation technique is considered include those of Engel et al. (2005); Menache et al. (2005); Ernst et al. (2005); Jung and Polani (2006b); Whiteson and Stone (2006); Loth et al. (2007); Parr et al. (2007); Xu et al. (2007); Kolter and Ng (2009), in addition to the ones which were cited earlier. The main difference between these work and the present work is that, as far as we know, adaptivity has never been considered earlier in reinforcement learning.

One issue not considered in the paper is the procedures' computational complexity. Our method is very flexible in that the computational complexity is mainly governed by the number of candidates considered and as long as this number grows to infinity our results still hold. However, one must be warned that if only a few parameter settings are considered then the transient performance will get worse.

The main message of our results is that asymptotically it is possible to learn as fast as if one knew the best parameters to be used. This is an inherently asymptotic result. In fact, it is easy to construct an example that shows that a result like this cannot be made non-asymptotic. Although an asymptotic result might seem weak, we would like to emphasize that achieving this asymptotics is non-trivial. One difficulty with asymptotic results is that on a particular problem instance it may always happen that the performance of some ad hoc method (which may even be inconsistent) is better than that of a procedure which comes with an asymptotic guarantee. Nevertheless, we think that it still makes sense to prefer methods which have good theoretical properties, because of the lack of any sound alternative choice. Who would like to use a method, whose performance deteriorates in the limit of an infinite amount of data? Usually, this means that the

procedure will be at minimum sensitive to the problem. (For an example of a recent discovery of an algorithm that is widely used, but shares this unfortunate property, see Nadler et al. 2009.)

Finally, let us comment on the implications of our results on benchmarking. First, it might seem that our result implies that benchmarking is not needed if one uses an adaptive procedure like ours. Although this holds in an asymptotic sense, benchmarking can still provide useful information. For example, it is often the case that the limitations of algorithms are not well understood. In those cases benchmarking can point to the class of problems which are difficult (easy) for the algorithm. This is very useful for improving our understanding of the algorithms and eventually of the algorithms themselves. Another possible (incorrect) conclusion of our study might be that it is difficult to benchmark algorithms that work on batch data. Indeed, estimating the performance (together with confidence intervals and in the lack of prior knowledge, like smoothness) given only batch data looks like a difficult problem. However, luckily, as it is well known, one can benchmark such algorithms by creating simulators and testing the algorithms on the simulators.

As for future work, the present work makes only the first steps towards creating reinforcement learning algorithms that require minimum human supervision. We expect more algorithms and algorithms that work on other contexts, too. Adaptivity in planning and online, interactive learning would also be interesting to consider. The online problem looks hard. In planning, racing algorithms can potentially be used (Mnih et al., 2008; Farahmand et al., 2009a). However, it remains for future work to understand how to do this the best way.

Appendix

Here we provide some auxiliary technical results that are needed in our proofs. We start with a noncentral tail inequality, followed by a Bernstein-like concentration inequality for a certain dependent sequence. Finally, the put together the two results to obtain a noncentral tail inequality for the considered class of dependent sequences.

A A Noncentral Tail Inequality

Lemma 2 (Noncentral Tail Inequality) *Let X be a random variables taking values in $[0, B]$. Assume the following Bernstein-like tail inequality holds for X:*

$$\mathbb{P}\left(\mathbb{E}\left[X\right] - X \geq \varepsilon\right) \leq \exp\left(-\frac{V\varepsilon^2}{\mathbb{E}\left[X\right] + \varepsilon}\right) \tag{21}$$

for some $V > 0$. Then, for any $0 < a < 1$,

$$\mathbb{P}\left(\mathbb{E}[X] - \frac{1}{1-a}X \geq \varepsilon\right) \leq \exp\left(-\frac{V(1-a)a\varepsilon}{(1+a)}\right).$$

Similarly, if

$$\mathbb{P}\left(X - \mathbb{E}[X] \geq \varepsilon\right) \leq \exp\left(-\frac{V\varepsilon^2}{\mathbb{E}[X] + \varepsilon}\right) \tag{22}$$

holds for some $V > 0$, then for any $0 < a < 1$,

$$\mathbb{P}\left(\frac{1}{1+a}\mathbb{E}_n[X] - \mathbb{E}[X] \geq \varepsilon\right) \leq \exp\left(-Va\varepsilon\right).$$

Proof First notice that

$$\begin{aligned}
\mathbb{P}\left(\mathbb{E}[X] - (1-a)^{-1}\mathbb{E}_n[X] \geq \varepsilon\right) &= \mathbb{P}\left(\mathbb{E}[X] - \mathbb{E}_n[X] \geq \varepsilon(1-a) + a\mathbb{E}[X]\right) \\
&\leq \exp\left(-\frac{V\left\{(1-a)\varepsilon + a\mathbb{E}[X]\right\}^2}{(1+a)\mathbb{E}[X] + (1-a)\varepsilon}\right) \\
&\leq \exp\left(-\frac{V\left\{(1-a)\varepsilon + a\mathbb{E}[X]\right\}^2}{\left\{(1-a)\varepsilon + a\mathbb{E}[X]\right\}\left(\frac{1+a}{a}\right)}\right) \\
&\leq \exp\left(-\frac{Va\left\{(1-a)\varepsilon + a\mathbb{E}[X]\right\}}{1+a}\right) \\
&\leq \exp\left(-\frac{V(1-a)a\varepsilon}{1+a}\right),
\end{aligned}$$

where we used (21) to get the first inequality, added a positive value to upper bound the denominator in the second inequality, and used the fact that $\mathbb{E}[X] \geq 0$ to derive the second and the last inequality.

Similarly, thanks to (22),

$$\begin{aligned}
\mathbb{P}\left((1+a)^{-1}\mathbb{E}_n[X] - \mathbb{E}[X] > \varepsilon\right) &= \mathbb{P}\left(\mathbb{E}_n[X] - \mathbb{E}[X] > \varepsilon(1+a) + a\mathbb{E}[X]\right) \\
&\leq \exp\left(-\frac{V\left\{(1+a)\varepsilon + a\mathbb{E}[X]\right\}^2}{(1+a)\mathbb{E}[X] + (1+a)\varepsilon}\right) \\
&\leq \exp\left(-\frac{V\left\{(1+a)\varepsilon + a\mathbb{E}[X]\right\}^2}{\left\{(1+a)\varepsilon + a\mathbb{E}[X]\right\}\left(\frac{1+a}{a}\right)}\right) \\
&\leq \exp\left(-\frac{Va\left\{(1+a)\varepsilon + a\mathbb{E}[X]\right\}}{1+a}\right) \\
&\leq \exp\left(-Va\varepsilon\right).
\end{aligned}$$

B Concentration Inequality for Hidden Markov Processes (HMPs)

The classical Bernstein inequality for independent and identically distributed sequences (e.g. Györfi et al. (2002, Appendix A)) can be shown to hold for sequences of dependent random variables under various conditions. Such extensions are very useful when studying reinforcement learning algorithms when the standard assumption is that the data comes from some Markov chain. In this section we give such an extension based on Samson (2000).

Let $X_1, \ldots, X_n$ be a time-homogeneous Markov chain with transition kernel $\mathcal{P}(\cdot|\cdot)$ taking values in some measurable space $\mathcal{X}$. We shall consider the concentration of the average of the Hidden-Markov Process

$$(X_1, f(X_1)), \ldots, (X_n, f(X_n)),$$

where $f : \mathcal{X} \to [0, B]$ is a fixed measurable function. To arrive at such an inequality we need a characterization of how fast (X_i) forgets its past.

For $i > 0$ let $\mathcal{P}^i(\cdot|x)$ be the i-step transition probability kernel: $\mathcal{P}^i(A|x) = \mathbb{P}(X_{i+1} \in A \,|\, X_1 = x)$ (for all $A \subset \mathcal{X}$ measurable). Define the upper-triangular matrix $\Gamma_n = (\gamma_{ij}) \in \mathbb{R}^{n \times n}$ as follows:

$$\gamma_{ij}^2 = \sup_{(x,y) \in \mathcal{X}^2} \left\| \mathcal{P}^{j-i}(\cdot|x) - \mathcal{P}^{j-i}(\cdot|y) \right\|_{\mathrm{TV}}. \tag{23}$$

for $1 \le i < j \le n$ and let $\gamma_{ii} = 1$ $(1 \le i \le n)$.

Matrix Γ_n, and its operator norm $\|\Gamma_n\|$ with respect to the Euclidean distance, are a measure of dependence for the random sequence $X_1, X_2, \ldots, X_n$. For example if the X_is are independent, $\Gamma_n = \mathbf{I}$ and $\|\Gamma_n\| = 1$. In general, $\|\Gamma_n\|$, which appears in the forthcoming concentration inequalities for dependent sequences, can grow with n. Since the concentration bounds are homogeneous in $n/\|\Gamma_n\|^2$, a larger value $\|\Gamma_n\|^2$ means a smaller "effective" sample size. This motivates the following definition:

Definition 7 *We say that a time-homogeneous Markov chain* uniformly quickly forgets its past *if* $\tau = \sup_{n \ge 1} \|\Gamma_n\|^2 < +\infty$. *Further,* τ *is called the* forgetting time *of the chain.*

Conditions under which a Markov chain uniformly quickly forgets its past are therefore of major interest.

The following proposition, extracted from the discussion on pages 421–422 of the paper by Samson (2000), gives such a condition:

Proposition 2 *Let* μ *be some nonnegative measure on* $\mathcal{X}$ *with nonzero mass* μ_0. *Let* $\mathcal{P}^k$ *be the* k*-step transition kernel as defined above. Assume that there exists some integer* r *such that for all* $x \in \mathcal{X}$ *and all measurable sets* A,

$$\mathcal{P}^r(A|x) \le \mu(A). \tag{24}$$

Then,

$$\|\Gamma_n\| \le \frac{\sqrt{2}}{1 - \rho^{\frac{1}{2r}}},$$

where $\rho = 1 - \mu_0$.

Meyn and Tweedie (1993) calls homogeneous Markov chains which satisfy the majorization condition (24) *uniformly ergodic*. We note in passing that there are other cases when $\|\Gamma_n\|$ is known to be independent of n. Most notable, this holds when the Markov chain is contracting. The matrix Γ_n can also be defined for more general dependent processes and such that the theorem below remains valid. With such a definition, $\|\Gamma_n\|$ can be shown to be bounded for general Φ-dependent processes.

The following result is a trivial corollary of Theorem 2 of Samson (2000) (Theorem 2 is stated for empirical processes and can be considered as a generalization of Talagrand's inequality to dependent random variables):

Theorem 4 *Let f be a measurable function on $\mathcal{X}$ whose values lie in $[0, B]$, $(X_i)_{1 \le i \le n}$ be a homogeneous Markov chain taking values in $\mathcal{X}$ and let Γ_n be the matrix with elements defined by (23). Let*

$$Z = \frac{1}{n} \sum_{i=1}^{n} f(X_i).$$

Then, for every $\varepsilon \ge 0$,

$$\mathbb{P}\left(Z - \mathbb{E}\left[Z\right] \ge \varepsilon\right) \le \exp\left(-\frac{\varepsilon^2 n}{2B \left\|\Gamma_n\right\|^2 \left(\mathbb{E}\left[Z\right] + \varepsilon\right)}\right),$$
$$\mathbb{P}\left(\mathbb{E}\left[Z\right] - Z \ge \varepsilon\right) \le \exp\left(-\frac{\varepsilon^2 n}{2B \left\|\Gamma_n\right\|^2 \mathbb{E}\left[Z\right]}\right).$$

C Noncentral Concentration Inequality for HMPs

By putting together the results of the last two sections we obtain the following result:

Lemma 3 *Let $X_1, X_2, \ldots, X_n$ be a time-homogenous Markov chain taking values in some measurable space $\mathcal{X}$, and f be a measurable function with $0 \le f \le B$. Let $Z = \frac{1}{n} \sum_{i=1}^{n} f(X_i)$. Let Γ_n be the matrix with elements defined by (23). Then, for any $0 < a < 1$,*

$$\mathbb{P}\left(\mathbb{E}\left[f(X)\right] - \frac{1}{1-a} Z \ge \varepsilon\right) \le \exp\left(-\frac{(1-a)an\varepsilon}{2B \left\|\Gamma_n\right\|^2 (1+a)}\right),$$
$$\mathbb{P}\left(\frac{1}{1+a} Z - \mathbb{E}\left[f(X)\right] \ge \varepsilon\right) \le \exp\left(-\frac{an\varepsilon}{2B \left\|\Gamma_n\right\|^2}\right).$$

Proof According to Theorem 4,

$$\mathbb{P}\left(Z - \mathbb{E}\left[f(X)\right] \geq t\right) \leq \exp\left(-\frac{\varepsilon^2 n}{2B\left\|\Gamma_n\right\|^2\left(\mathbb{E}\left[f(X)\right] + \varepsilon\right)}\right),$$

$$\mathbb{P}\left(\mathbb{E}\left[f(X)\right] - Z \geq t\right) \leq \exp\left(-\frac{\varepsilon^2 n}{2B\left\|\Gamma_n\right\|^2\mathbb{E}\left[f(X)\right]}\right)$$

$$\leq \exp\left(-\frac{\varepsilon^2 n}{2B\left\|\Gamma_n\right\|^2\left(\mathbb{E}\left[f(X)\right] + \varepsilon\right)}\right).$$

These inequalities have the same form as the Bernstein-like inequality in Lemma 2 with the choice of $V = \frac{1}{B\|\Gamma_n\|^2}$, and therefore imply the result.

References

Abbeel, P., A. Coates, M. Quigley, and A. Y. Ng: 2007, 'An Application of Reinforcement Learning to Aerobatic Helicopter Flight'. In Schölkopf et al. (2007), pp. 1–8, MIT Press. 2

Antos, A., R. Munos, and C. Szepesvári: 2008a, 'Fitted Q-iteration in continuous action-space MDPs'. In: J. C. Platt, D. Koller, Y. Singer, and S. T. Roweis (eds.): *Advances in Neural Information Processing Systems 20*. Cambridge, MA, USA, pp. 9–16, MIT Press. 2, 3

Antos, A., C. Szepesvári, and R. Munos: 2007, 'Value-iteration based fitted policy iteration: learning with a single trajectory'. In: *IEEE ADPRL*. pp. 330–337. 2

Antos, A., C. Szepesvári, and R. Munos: 2008b, 'Learning near-optimal policies with Bellman-residual minimization based fitted policy iteration and a single sample path'. *Machine Learning* **71**(1), 89–129. 2, 3, 8, 19

Arlot, S. and A. Celisse: 2009, 'A survey of cross-validation procedures for model selection'. Accepted by Statistics Surveys. 15

Barron, A. R.: 1991, 'Complexity Regularization with Application to Artificial Neural Networks'. In: G. Roussas (ed.): *Nonparametric Function Estimation and Related Topics*. Kluwer Academic Publishers, pp. 561–576. 3, 11

Bartlett, P. and A. Tewari: 2007, 'Sample complexity of policy search with known dynamics'. In Schölkopf et al. (2007), pp. 97–104, MIT Press. 23

Bartlett, P. L., S. Boucheron, and G. Lugosi: 2002, 'Model Selection and Error Estimation'. *Machine Learning* **48**(1–3), 85–113. 3, 10

Bertsekas, D. P. and S. E. Shreve: 1978, *Stochastic Optimal Control: The Discrete-Time Case*. Academic Press. 2, 4, 6

Bertsekas, D. P. and J. N. Tsitsiklis: 1996, *Neuro-Dynamic Programming*. Athena Scientific, Belmont, MA. 4, 22

Boyan, J. A.: 2002, 'Technical update: Least-squares temporal difference learning'. *Machine Learning* **49**, 233–246. 21

Bradtke, S. J. and A. G. Barto: 1996, 'Linear least-squares algorithms for temporal difference learning'. *Machine Learning* **22**, 33–57. 21

Brafman, R. I. and M. Tennenholtz: 2003, 'R-max - a general polynomial time algorithm for near-optimal reinforcement learning'. *Journal of Machine Learning Research* **3**, 213–231. 23

Chang, H. S., M. C. Fu, J. Hu, and S. I. Marcus: 2008, *Simulation-based Algorithms for Markov Decision Processes*. Springer Verlag. 23

Devroye, L., D. Schäfer, L. Györfi, and H. Walk: 2003, 'The estimation problem of minimum mean squared error'. *Statistics & Decisions* **21**, 15—28. 14

Druet, C., D. Ernst, and L. Wehenkel: 2000, 'Application of Reinforcement Learning to Electrical Power System Closed-Loop Emergency Control'. In: D. A. Zighed, H. J. Komorowski, and J. M. Zytkow (eds.): *4th European Conference on the Principles of Data Mining and Knowledge Discovery*, Vol. 1910 of *Lecture Notes in Computer Science*. pp. 86–95, Springer. 2

Engel, Y., S. Mannor, and R. Meir: 2005, 'Reinforcement learning with Gaussian processes'. In: L. De Raedt and S. Wrobel (eds.): *Proceedings of the 22nd International Conference on Machine Learning (ICML-05)*, Vol. 119 of *ACM International Conference Proceeding Series*. New York, NY, USA, pp. 201–208, ACM. 24

Ernst, D., P. Geurts, and L. Wehenkel: 2005, 'Tree-based batch mode reinforcement learning'. *Journal of Machine Learning Research* **6**, 503–556. 2, 24

Farahmand, A.-m., M. Ghavamzadeh, C. Szepesvári, and S. Mannor: 2009a, 'Regularized Fitted Q-Iteration for Planning in Continuous-Space Markovian Decision Problems'. In: *Proceedings of American Control Conference (ACC)*. pp. 725–730. 3, 25

Farahmand, A.-m., M. Ghavamzadeh, C. Szepesvári, and S. Mannor: 2009b, 'Regularized Policy Iteration'. In: D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou (eds.): *Advances in Neural Information Processing Systems 21*. MIT Press, pp. 441–448. 1, 2, 3

Farahmand, A.-m., A. Shademan, M. Jägersand, and C. Szepesvári: 2009c, 'Model-based and Model-free Reinforcement Learning for Visual Servoing'. In: *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*. pp. 2917–2924. 23

Györfi, L., M. Kohler, A. Krzyżak, and H. Walk: 2002, *A Distribution-Free Theory of Nonparametric Regression*. Springer Verlag, New York. 27

Jung, T. and D. Polani: 2006a, 'Least Squares SVM for Least Squares TD Learning'. In: *In Proc. 17th European Conference on Artificial Intelligence*. pp. 499–503. 1

Jung, T. and D. Polani: 2006b, 'Least Squares SVM for Least Squares TD Learning'. In: *ECAI*. pp. 499–503. 24

Keller, P. W., S. Mannor, and D. Precup: 2006, 'Automatic basis function construction for approximate dynamic programming and reinforcement learning'. In: *ICML '06: Proceedings of the 23rd international conference*

on Machine learning. New York, NY, USA, pp. 449–456, ACM. 8

Kolter, J. Z. and A. Y. Ng: 2009, 'Regularization and feature selection in least-squares temporal difference learning'. In: *ICML '09: Proceedings of the 26th Annual International Conference on Machine Learning*. New York, NY, USA, pp. 521–528, ACM. 1, 24

Lagoudakis, M. and R. Parr: 2003, 'Least-squares policy iteration'. *Journal of Machine Learning Research* **4**, 1107–1149. 2, 8

Loth, M., M. Davy, and P. Preux: 2007, 'Sparse Temporal Difference Learning using LASSO'. In: *IEEE International Symposium on Approximate Dynamic Programming and Reinforcement Learning*. 24

Lugosi, G. and M. Wegkamp: 2004, 'Complexity regularization via localized random penalties'. *Annals of Statistics* **32**, 1679–1697. 3, 11, 15

Menache, I., S. Mannor, and N. Shimkin: 2005, 'Basis Function Adaptation in Temporal Difference Reinforcement Learning'. *Annals of Operations Research* **134**(1), 215–238. 8, 24

Meyn, S. and R. Tweedie: 1993, *Markov Chains and Stochastic Stability*. New York: Springer-Verlag. 14, 28

Mnih, V., C. Szepesvári, and J.-Y. Audibert: 2008, 'Empirical Bernstein stopping'. In: W. W. Cohen, A. McCallum, and S. T. Roweis (eds.): *Proceedings of the 25th International Conference Machine Learning (ICML-08)*, Vol. 307 of *ACM International Conference Proceeding Series*. New York, NY, USA, pp. 672–679, ACM. 25

Munos, R.: 2007, 'Performance Bounds in L_p norm for Approximate Value Iteration'. *SIAM Journal on Control and Optimization*. 8

Nadler, B., N. Srebro, and X. Zhou: 2009, 'Semi-Supervised Learning with the Graph Laplacian: The Limit of Infinite Unlabelled Data'. In: *NIPS-09*. 25

Ng, A. Y. and M. Jordan: 2000, 'PEGASUS: A policy search method for large MDPs and POMDPs'. In: C. Boutilier and M. Goldszmidt (eds.): *Proceedings of the 16th Conference in Uncertainty in Artificial Intelligence (UAI'00)*. San Francisco CA, pp. 406–415, Morgan Kaufmann. 23

Parr, R., C. Painter-Wakefield, L. Li, and M. L. Littman: 2007, 'Analyzing feature generation for value-function approximation'. In: Z. Ghahramani (ed.): *Proceedings of the 24th International Conference on Machine Learning (ICML-07)*, Vol. 227 of *ACM International Conference Proceeding Series*. New York, NY, USA, pp. 737–744, ACM. 8, 24

Puterman, M.: 1994, *Markov Decision Processes — Discrete Stochastic Dynamic Programming*. New York, NY: John Wiley & Sons, Inc. 4

Riedmiller, M.: 2005, 'Neural Fitted Q Iteration – First Experiences with a Data Efficient Neural Reinforcement Learning Method'. In: *16th European Conference on Machine Learning*. pp. 317–328. 2

Samson, P.-M.: 2000, 'Concentration of Measure Inequalities for Markov Chains and Φ-Mixing Processes'. *The Annals of Probability* **28**(1), 416–461. 27, 28

Schölkopf, B., J. C. Platt, and T. Hoffman (eds.): 2007, 'Advances in Neural Information Processing Systems 19'. Cambridge, MA, USA: MIT Press.

29
Strehl, A. and M. Littman: 2008, 'Online Linear Regression and Its Application to Model-Based Reinforcement Learning'. In: J. Platt, D. Koller, Y. Singer, and S. Roweis (eds.): *Advances in Neural Information Processing Systems 20*. Cambridge, MA: MIT Press, pp. 1417–1424. 23
Sutton, R. S. and A. G. Barto: 1998, *Reinforcement Learning: An Introduction (Adaptive Computation and Machine Learning)*. The MIT Press. 1, 4
Sutton, R. S., C. Szepesvári, A. Geramifard, and M. Bowling: 2008, 'Dyna-style Planning with Linear Function Approximation and Prioritized Sweeping'. In: *Proceedings of the 24th Conference on Uncertainty in Artificial Intelligence*. 23
Szepesvári, C.: 2001, 'Efficient Approximate Planning in Continuous Space Markovian Decision Problems'. *AI Communications* **13**, 163–176. 21, 22
Szepesvári, C.: 2009, 'Reinforcement Learning Algorithms for MDPs'. Technical report, University of Alberta. 4, 24
Taylor, G. and R. Parr: 2009, 'Kernelized value function approximation for reinforcement learning'. In: *ICML '09: Proceedings of the 26th Annual International Conference on Machine Learning*. New York, NY, USA, pp. 1017–1024, ACM. 1
van der Vaart, A. W., S. Dudoit, and M. J. van der Laan: 2006, 'Oracle Inequalities for Multi-fold Cross Validation'. *Statistics and Decisions* **24**, 351–372. 15
Wegkamp, M.: 2003, 'Model Selection in Nonparametric Regression'. *The Annals of Statistics* **31**(1), 252–273. 3, 15
Whiteson, S. and P. Stone: 2006, 'Evolutionary Function Approximation for Reinforcement Learning'. *Journal of Machine Learning Research* **7(May)**, 877–917. 24
Xu, X., D. Hu, and X. Lu: 2007, 'Kernel-Based Least Squares Policy Iteration for Reinforcement Learning'. *IEEE Transactions on Neural Networks* **18**, 973–992. 24