Journal of Machine Learning Research 3 (2002) 397-422 Submitted 11/01; Published 11/02

Using Confidence Bounds for Exploitation-Exploration Trade-offs

Peter Auer PAUER@IGI.TU-GRAZ.AC.AT
Graz University of Technology
Institute for Theoretical Computer Science
Inffeldgasse 16b
A-8010 Graz, Austria

Editor: Philip M. Long

Abstract

We show how a standard tool from statistics — namely confidence bounds — can be used to elegantly deal with situations which exhibit an exploitation-exploration trade-off. Our technique for designing and analyzing algorithms for such situations is general and can be applied when an algorithm has to make exploitation-versus-exploration decisions based on uncertain information provided by a random process.

We apply our technique to two models with such an exploitation-exploration trade-off. For the adversarial bandit problem with shifting our new algorithm suffers only $\tilde{O}\left((ST)^{1/2}\right)$ regret *with high probability* over T trials with S shifts. Such a regret bound was previously known only *in expectation*. The second model we consider is associative reinforcement learning with linear value functions. For this model our technique improves the regret from $\tilde{O}\left(T^{3/4}\right)$ to $\tilde{O}\left(T^{1/2}\right)$.

Keywords: Online Learning, Exploitation-Exploration, Bandit Problem, Reinforcement Learning, Linear Value Function

1. Introduction

In this paper we consider situations which exhibit an exploitation-exploration trade-off. In such a scenario an algorithm repeatedly makes decisions to maximize its rewards — the exploitation — but the algorithm has only limited knowledge about the process generating the rewards. Thus occasionally the algorithm might decide to do exploration which improves the knowledge about the reward generating process, but which is not necessarily maximizing the current reward.

If the knowledge about the reward generating process can be captured by a set of random variables, then confidence bounds provide a very useful tool to deal with the exploitation-exploration trade-off. The estimated means (or a similar quantity) of the random variables reflect the current knowledge of the algorithm in a condensed form and guide further exploitation. The widths of the confidence bounds reflect the uncertainty of the algorithm's knowledge and will guide further exploration. By relating means and widths we can obtain criteria on when to explore and when to exploit. How such a criterion is constructed depends on the actual model under consideration. In the remainder of this paper we consider two such models in detail, the adversarial bandit problem with shifting and associative reinforcement learning with linear value functions. The bandit problem is maybe the most generic way to model an exploitation-exploration trade-off (Robbins, 1952, Lai and Robbins, 1985, Berry and Fristedt, 1985, Agrawal, 1995, Auer et al., 1995, Sutton and Barto, 1998). In this paper we will consider a worst-case variant of the bandit problem with shifting. Furthermore, we will consider associative reinforcement learning with linear value functions (Kaelbling, 1994a,b, Sutton and Barto, 1998, Abe and Long, 1999). In this model exploration is more involved since knowledge about a functional dependency has to be collected.

Using confidence bounds to deal with an exploitation-exploration trade-off is not a new idea (e.g. Kaelbling, 1994a,b, Agrawal, 1995). What is new in this paper is that we use confidence bounds in rather complicated situations and that we are still able to prove rigorous performance bounds. Thus we believe that confidence bounds can be successfully applied in many such situations with an exploitation-exploration trade-off. Furthermore, since algorithms which use confidence bounds can be tuned quite easily, we expect that such algorithms prove useful in practical applications.

In Section 2 we start off with the *random* bandit problem. The random bandit problem is a typical model for the trade-off between exploitation and exploration. Using upper confidence bounds, very simple and almost optimal algorithms for the random bandit problem have been derived. We shortly review this previous work since it illuminates the main ideas of using upper confidence bounds. In Section 3 we introduce the *adversarial* bandit problem with shifting and compare our new results with the previously known results. In Section 4 we define the model for associative reinforcement learning with linear value functions and discuss our results for this model.

2. Upper Confidence Bounds for the Random Bandit Problem

The random bandit problem was originally proposed by Robbins (1952). It formalizes an exploitation-exploration trade-off where in each trial $t = 1, \ldots, T$ one out of K possible alternatives has to be chosen. We denote the choice for trial t by $i(t) \in \{1, \ldots, K\}$. For the chosen alternative a reward $x_{i(t)}(t) \in \mathbf{R}$ is collected and the rewards for the other alternatives $x_i(t) \in \mathbf{R}$, $i \in \{1, \ldots, K\} \setminus \{i(t)\}$, are *not* revealed. The goal of an algorithm for the bandit problem is to maximize its total reward $\sum_{t=1}^{T} x_{i(t)}(t)$. For the random bandit problem[1] it is assumed that in each trial the rewards $x_i(t)$ are drawn independently from some fixed but unknown distributions $\mathcal{D}_1, \ldots, \mathcal{D}_K$. The expected total reward of a learning algorithm should be close to the expected total reward given by the best distribution $\mathcal{D}_i$. Thus the regret of a learning algorithm for the random bandit problem is defined as

$$\bar{R}(T) = \max_{i \in \{1,\ldots,K\}} \mathbf{E}\left[\sum_{t=1}^{T} x_i(t)\right] - \mathbf{E}\left[\sum_{t=1}^{T} x_{i(t)}(t)\right].$$

In this model the exploitation-exploration trade-off is reflected on one hand by the necessity for trying all alternatives, and on the other hand by the regret suffered when trying an

1. The term "bandit problem" (or more precisely "K-armed bandit problem") reflects the problem of a gambler in a room with various slot machines. In each trial the gambler has to decide which slot machine he wants to play. To maximize his total gain or reward his (rational) choice will be based on the previously collected rewards.

alternative which is not optimal: too little exploration might make a sub-optimal alternative look better than the optimal one because of random fluctuations, too much exploration prevents the algorithm from playing the optimal alternative often enough which also results in a larger regret.

Lai and Robbins (1985) have shown that an optimal algorithm achieves $\bar{R}(T) = \Theta(\ln T)$ as $T \to \infty$ when the variances of the distributions $\mathcal{D}_i$ are finite.[2] Agrawal (1995) has shown that a quite simple learning algorithm suffices to obtain such performance. This simple algorithm is based on upper confidence bounds of the form $\hat{\mu}_i(t) + \sigma_i(t)$ for the expected rewards μ_i of the distributions $\mathcal{D}_i$. Here $\hat{\mu}_i(t)$ is an estimate for the true expected reward μ_i and $\sigma_i(t)$ is chosen such that $\hat{\mu}_i(t) - \sigma_i(t) \leq \mu_i \leq \hat{\mu}_i(t) + \sigma_i(t)$ with high probability. In each trial t the algorithm selects the alternative with maximal upper confidence bound $\hat{\mu}_i(t) + \sigma_i(t)$. Thus an alternative i is selected if $\hat{\mu}_i(t)$ is large or if $\sigma_i(t)$ is large. Informally we may say that a trial is an exploration trial if an alternative with large $\sigma_i(t)$ is chosen since in this case the estimate $\hat{\mu}_i(t)$ is rather unreliable. When an alternative with large $\hat{\mu}_i(t)$ is chosen we may call such a trial an exploitation trial. Since $\sigma_i(t)$ decreases rapidly with each choice of alternative i, the number of exploration trials is limited. If $\sigma_i(t)$ is small then $\hat{\mu}_i(t)$ is close to μ_i and an alternative is selected in an exploitation trial only if it is indeed the optimal alternative with maximal μ_i. Thus the use of upper confidence bounds automatically trades off between exploitation and exploration.

3. The Adversarial Bandit Problem with Shifts

The *adversarial* bandit problem was first analyzed by Auer et al. (1995). In contrast to the random bandit problem the adversarial bandit problem makes no statistical assumptions on how the rewards $x_i(t)$ are generated. Thus, the rewards might be generated in an adversarial way to make life hard for an algorithm in this model. Since the rewards are not random any more, the regret of an algorithm for the adversarial bandit problem is defined as

$$R(T) = \max_{i \in 1,\ldots,K} \sum_{t=1}^{T} x_i(t) - \sum_{t=1}^{T} x_{i(t)}(t) \tag{1}$$

where $R(T)$ might be a random variable depending on a possible randomization of the algorithm. In our paper (Auer et al., 1995) we have derived a randomized algorithm which achieves $\mathbf{E}[R(T)] = O\left(T^{2/3}\right)$ for bounded rewards. In a subsequent paper (Auer et al., 1998) this algorithm has been improved to yield $\mathbf{E}[R(T)] = O\left(T^{1/2}\right)$. In the same paper we have also shown that a variant of the algorithm satisfies $R(T) = O\left(T^{2/3}(\ln T)^{1/3}\right)$ with high probability. This bound was improved again (Auer et al., 2000) as we can show that $R(T) = O\left(T^{1/2}(\ln T)^{1/2}\right)$ with high probability. This is almost optimal since a lower bound $\mathbf{E}[R(T)] = \Omega\left(T^{1/2}\right)$ has already been shown (Auer et al., 1995). This lower bound holds even if the rewards are generated at random as for the random bandit problem. The reason is that for suitable distributions $\mathcal{D}_i$ we have

$$\mathbf{E}\left[\max_{i \in \{1,\ldots,K\}} \sum_{t=1}^{T} x_i(t)\right] = \max_{i \in \{1,\ldots,K\}} \mathbf{E}\left[\sum_{t=1}^{T} x_i(t)\right] + \Omega\left(\sqrt{T}\right)$$

2. Their result is even more general.

and thus $\mathbf{E}[R(T)] = \bar{R}(T) + \Omega\left(\sqrt{T}\right)$.

In the current paper we consider an extension of the adversarial bandit problem where the bandits are allowed to "shift": the algorithm keeps track of the alternative which gives highest reward even if this best alternative changes over time. Formally we compare the total reward collected by the algorithm with the total reward of the best *schedule* with S segments $\{1, \ldots, t_1\}$, $\{t_1 + 1, \ldots, t_2\}$, $\ldots$, $\{t_{S-1} + 1, \ldots, T\}$. The regret of the algorithm with respect to the best schedule with S segments is defined as

$$R_S(T) = \max_{0=t_0<t_1<\cdots<t_S=T} \left(\sum_{s=1}^{S} \left[\max_{i\in\{1,\ldots,K\}} \sum_{t=t_{s-1}+1}^{t_s} x_i(t) \right] \right) - \sum_{t=1}^{T} x_{i(t)}(t).$$

We will show that $R_S(T) = O\left(\sqrt{TS\ln(T)}\right)$ with high probability. This is essentially optimal since any algorithm which has to solve S independent adversarial bandit problems of length T/S will suffer $\Omega\left(\sqrt{TS}\right)$ regret. For a different algorithm Auer et al. (2000) show the bound $\mathbf{E}[R_S(T)] = O\left(\sqrt{TS\ln(T)}\right)$, but for that algorithm the variance of the regret is so large that no interesting bound on the regret can be given that holds with high probability.

3.1 The Algorithm for the Adversarial Bandit Problem with Shifting

Our algorithm ShiftBand (Figure 1) for the adversarial bandit problem with shifting combines several approaches which have been found useful in the context of the bandit problem or in the context of shifting targets. One of the main ingredients is that the algorithm calculates estimates for the rewards of all the alternatives. For a single trial these estimates are given by $\hat{x}_i(t)$ since the expectation of such an estimate equals the true reward $x_i(t)$.

Another ingredient is the exponential weighting scheme which for each alternative calculates a weight $w_i(t)$ from an estimate of the cumulative rewards so far. Such exponential weighting schemes have been used for the analysis of the adversarial bandit problem by Auer et al. (1995, 1998, 2000). In contrast to previous algorithms (Auer et al., 1995, 1998) we use in this paper an estimate of the cumulative rewards which does not give the correct expectation but which — as an upper confidence bound — overestimates the true cumulative rewards. This over-estimation emphasizes exploration over exploitation, which in turn gives more reliable estimates for the true cumulative rewards. This is the reason why we are able to give bounds on the regret which hold with high probability. In the algorithm ShiftBand the upper bound on the cumulative regret is present only implicitly. Intuitively the sum

$$\sum_{\tau=1}^{t} \left(\hat{x}_i(\tau) + \frac{\alpha}{p_i(\tau)\sqrt{TK/S}} \right)$$

can be seen as this upper confidence bound on the cumulative regret $\sum_{\tau=1}^{t} x_i(\tau)$. In the analysis of the algorithm the relationship between this confidence bound and the cumulative regret will be made precise. In contrast to the random bandit problem this confidence bound is not a confidence bound for an external random process[3] but a confidence bound for the

3. This external random process generates the rewards of the random bandit problem.

Algorithm ShiftBand
Parameters: Reals $\alpha, \beta, \eta > 0$, $\gamma \in (0,1]$, the number of trials T, the number of segments S.
Initialization: $w_i(1) = 1$ for $i = 1, \dots, K$.
Repeat for $t = 1, \dots, T$

1. Choose $i(t)$ at random accordingly to the probabilities $p_i(t)$ where

$$p_i(t) = (1-\gamma)\frac{w_i(t)}{W(t)} + \frac{\gamma}{K} \quad \text{and} \quad W(t) = \sum_{i=1}^{K} w_i(t).$$

2. Collect reward $x_{i(t)}(t) \in [0,1]$.

3. For $i = 1, \dots, K$ set $\hat{x}_i(t) = \begin{cases} x_i(t)/p_i(t) & \text{if } i = i(t) \\ 0 & \text{otherwise,} \end{cases}$

   and calculate the update of the weights as

$$w_i(t+1) = w_i(t) \cdot \exp\left\{\eta\left(\hat{x}_i(t) + \frac{\alpha}{p_i(t)\sqrt{TK/S}}\right)\right\} + \frac{\beta}{K}W(t)\ .$$

Figure 1: Algorithm ShiftBand

effect of the internal randomization of the algorithm. The application of confidence bounds to capture the internal randomization of the algorithm shows that confidence bounds are a quite versatile tool.

The weights reflect the algorithm's belief in which alternative is best. An alternative is either chosen at random proportional to its weight, or with probability γ an arbitrary alternative is chosen for additional exploration.

The final ingredient is a mechanism for dealing with shifting. The basic approach is to lower bound the weights $w_i(t)$ appropriately. This is achieved by the term $\frac{\beta}{K}W(t)$ in the calculation of the weights $w_i(t)$. Lower bounding the weights means that it does not take too long for a small weight to become large again if it corresponds to the new best alternative after a shift. Similar ideas for dealing with changing targets have been used bt Littlestone and Warmuth (1994), Auer and Warmuth (1998), and Herbster and Warmuth (1998).

In the remainder of this section we assume that for all rewards $x_i(t) \in [0,1]$. If the rewards $x_i(t)$ are not in $[0,1]$ but bounded, then an appropriate scaling gives results similar to the theorems below.

Theorem 1 *We use the notation of (1) and Figure 1. If T, S, K, and δ, are such that $T \geq 144\, KS \ln(TK/\delta)$, and algorithm* ShiftBand *is run with parameters $\alpha = 2\sqrt{\ln(T^3K/\delta)}$, $\beta = 1/T$, $\gamma = 2K\eta$, and $\eta = \sqrt{\ln(TK)\, S/(TK)}$, then the regret of the algorithm satisfies*

$$R_S(T) \leq 11\sqrt{TKS\ln(T^3K/\delta)} \tag{2}$$

with probability at least $1-\delta$.

We note that for the bound in (2) the number of trials T and the number of shifts S have to be known in advance. Using the doubling trick it can be shown that for a slight modification of algorithm ShiftBand (2) holds with a slightly worse constant even if the number of trials T is not known in advance. If S is not known and the algorithm is run with parameter S_0 then we can prove the following generalization of Theorem 1.

Theorem 2 *We use the notation of (1) and Figure 1. If* T, S_0, K, *and* δ, *are such that* $T \geq 144\, K S_0 \ln(TK/\delta)$, *and algorithm* ShiftBand *is run with parameters* $\alpha = 2\sqrt{\ln(T^3K/\delta)}$, $\beta = 1/T$, $\gamma = 2K\eta$, *and* $\eta = \sqrt{\ln(TK)\, S_0/(TK)}$, *then the regret of the algorithm satisfies*

$$R_S(T) \leq \left(8\sqrt{S_0} + 3\frac{S}{\sqrt{S_0}}\right)\sqrt{TK\ln(T^3K/\delta)}. \tag{3}$$

with probability at least $1-\delta$.

It is easy to see that Theorem 1 follows from Theorem 2 if S is known in advance and S_0 is set equal to S.

Remark 3 *Finally, we remark that Theorems 1 and 2 hold unchanged even if the rewards depend on the algorithm's past choices* $i(1),\ldots,i(t-1)$ *(but not on future choices* $i(t), i(t+1), \ldots$*). This can be seen from the proof below. This allows the application of our result in game theory along the line of our matrix game application (see Auer et al., 1995).*

3.2 Proof of Theorem 2

We start with an outline of the proof. We need to show that the regret of our algorithm is not much worse than the regret of the best schedule of shifts[4] $\{1,\ldots,t_1\}$, $\{t_1+1,\ldots,t_2\}$, $\ldots,\{t_{S-1}+1,\ldots,T\}$. To achieve this we will show that for each segment $\{T_1+1,\ldots,T_2\}$ the regret of our algorithm on this segment is close to the regret of the best alternative for this segment. Considering a segment $(T_1, T_2]$ we show an upper bound on the true cumulative reward (Lemma 4) which holds with high probability. Then we show that the cumulative reward obtained by our algorithm is close to this bound (Lemma 5).

The upper bound which is shown in Lemma 4 relies on the fact that

$$\frac{1}{\eta}\ln\frac{w_i(T_2+1)}{w_i(T_1+1)} \geq \sum_{t=T_1+1}^{T_2}\left(\hat{x}_i(t) + \frac{\alpha}{p_i(t)\sqrt{TK/S_0}}\right)$$

is essentially an upper bound on the true cumulative reward of alternative i for this segment,

$$\sum_{t=T_1+1}^{T_2} x_i(t).$$

The lemma draws from martingale theory and is similar in spirit to Hoeffding's inequality (Hoeffding, 1963). A similar analysis was used by Auer et al. (2000).

4. Since the number of shifts is bounded by S even the best schedule will suffer some regret.

In Lemma 5 we show that the cumulative reward of algorithm ShiftBand for segment $(T_1, T_2]$ is close to $\max_i \frac{1}{\eta} \ln \frac{w_i(T_2+1)}{w_i(T_1+1)}$, which is the upper bound on the true cumulative reward of the best alternative. The proof of the lemma combines the proof for algorithm Hedge (Auer et al., 1995) with techniques to deal with shifting targets (Auer and Warmuth, 1998, Herbster and Warmuth, 1998).

Lemma 4 *Choose $\delta > 0$ and $2\sqrt{\ln(T^3K/\delta)} \leq \alpha \leq 4\sqrt{TK/S_0}$. Then the probability that for all $0 \leq T_1 < T_2 \leq T$ and all $i \in \{1, \ldots, K\}$ the inequality*

$$\sum_{t=T_1+1}^{T_2} x_i(t) \leq \frac{1}{\eta} \ln \frac{w_i(T_2+1)}{w_i(T_1+1)} + \alpha\sqrt{TK/S_0}$$

holds is at least $1 - \delta$.

Proof. We fix some $i \in \{1, \ldots, K\}$ and some segment $(T_1, T_2]$ and define a random variable f_t as

$$f_t = \min\left\{ \frac{\alpha}{4\sqrt{TK/S_0}}, \frac{\alpha\sqrt{TK/S_0}}{2\sum_{\tau=T_1+1}^{t} \frac{1}{p_i(\tau)}} \right\}.$$

Thus $f_{t+1} \leq f_t$ and $f_t \leq 1$. Since

$$w_i(t+1) \geq w_i(t) \cdot \exp\left\{ \eta \left(\hat{x}_i(t) + \frac{\alpha}{p_i(t)\sqrt{TK/S_0}} \right) \right\}$$

we get

$$\begin{aligned}
&\mathbf{P}\left\{ \sum_{t=T_1+1}^{T_2} x_i(t) > \frac{1}{\eta} \ln \frac{w_i(T_2+1)}{w_i(T_1+1)} + \alpha\sqrt{TK/S_0} \right\} \\
&\leq \mathbf{P}\left\{ \sum_{t=T_1+1}^{T_2} x_i(t) > \sum_{t=T_1+1}^{T_2} \left[\hat{x}_i(t) + \frac{\alpha}{p_i(t)\sqrt{TK/S_0}} \right] + \alpha\sqrt{TK/S_0} \right\} \\
&= \mathbf{P}\left\{ \sum_{t=T_1+1}^{T_2} \left[x_i(t) - \hat{x}_i(t) - \frac{\alpha}{2p_i(t)\sqrt{TK/S_0}} \right] \right. \\
&\qquad\qquad \left. > \alpha\sqrt{TK/S_0} + \sum_{t=T_1+1}^{T_2} \frac{\alpha}{2p_i(t)\sqrt{TK/S_0}} \right\} \\
&\leq \mathbf{P}\left\{ \sum_{t=T_1+1}^{T_2} \left[x_i(t) - \hat{x}_i(t) - \frac{\alpha}{2p_i(t)\sqrt{TK/S_0}} \right] \right. \\
&\qquad\qquad \left. > \max\{\alpha\sqrt{TK/S_0}, \sum_{t=T_1+1}^{T_2} \frac{\alpha}{2p_i(t)\sqrt{TK/S_0}}\} \right\} \\
&= \mathbf{P}\left\{ \sum_{t=T_1+1}^{T_2} \left[x_i(t) - \hat{x}_i(t) - \frac{\alpha}{2p_i(t)\sqrt{TK/S_0}} \right] > \frac{\alpha^2}{4f_{T_2}} \right\}
\end{aligned}$$

$$
\begin{aligned}
&= \mathbf{P}\left\{ f_{T_2} \sum_{t=T_1+1}^{T_2} \left[x_i(t) - \hat{x}_i(t) - \frac{\alpha}{2p_i(t)\sqrt{TK/S_0}} \right] > \alpha^2/4 \right\} \\
&= \mathbf{P}\left\{ \exp\left\{ f_{T_2} \sum_{t=T_1+1}^{T_2} \left[x_i(t) - \hat{x}_i(t) - \frac{\alpha}{2p_i(t)\sqrt{TK/S_0}} \right] \right\} > \exp\left\{\alpha^2/4\right\} \right\} \\
&\leq \mathbf{E}\left[\exp\left\{ f_{T_2} \sum_{t=T_1+1}^{T_2} \left[x_i(t) - \hat{x}_i(t) - \frac{\alpha}{2p_i(t)\sqrt{TK/S_0}} \right] \right\} \right] \cdot \exp\left\{-\alpha^2/4\right\} \qquad (4)
\end{aligned}
$$

by Markov's inequality. We set

$$
V_t = \exp\left\{ f_t \sum_{\tau=T_1+1}^{t} \left[x_i(\tau) - \hat{x}_i(\tau) - \frac{\alpha}{2p_i(\tau)\sqrt{TK/S_0}} \right] \right\}
$$

and find that

$$
V_t = \exp\left\{ f_t \left[x_i(t) - \hat{x}_i(t) - \frac{\alpha}{2p_i(t)\sqrt{TK/S_0}} \right] \right\} \cdot (V_{t-1})^{f_t/f_{t-1}}.
$$

We denote by $\mathbf{E}_t$ the expectation conditioned on the previous trials $1, \ldots, t-1$. Since $f_t[x_i(t) - \hat{x}_i(t)] \leq 1$, $e^x \leq 1 + x + x^2$ for $x \leq 1$, $\mathbf{E}_t\left[\hat{x}_i(t)\right] = x_i(t)$, and

$$
\begin{aligned}
\mathbf{E}_t\left[(x_i(t) - \hat{x}_i(t))^2\right] &= p_i(t)\left(x_i(t) - x_i(t)/p_i(t)\right)^2 + (1 - p_i(t))\left(x_i(t) - 0\right)^2 \\
&= x_i(t)^2\left(1/p_i(t) - 1\right) \\
&\leq 1/p_i(t)
\end{aligned}
$$

we have that

$$
\mathbf{E}_t\left[\exp\left\{f_t[x_i(t) - \hat{x}_i(t)]\right\}\right] \leq 1 + f_t^2/p_i(t).
$$

Since $f_t \leq \frac{\alpha}{2\sqrt{TK/S_0}}$, $v^u \leq 1 + v$ for $u \in [0,1]$, and $(1+f)e^{-f} \leq 1$, we get

$$
\begin{aligned}
\mathbf{E}_t\left[V_t\right] &= \mathbf{E}_t\left[\exp\left\{f_t\left[x_i(t) - \hat{x}_i(t)\right]\right\}\right] \cdot \exp\left\{ -\frac{\alpha f_t}{2p_i(t)\sqrt{TK/S_0}} \right\} \cdot (V_{t-1})^{f_t/f_{t-1}} \\
&\leq \left(1 + \frac{f_t^2}{p_i(t)}\right) \cdot \exp\left\{ -\frac{f_t^2}{p_i(t)} \right\} \cdot (1 + V_{t-1}) \\
&\leq 1 + V_{t-1}.
\end{aligned}
$$

Thus it follows by induction over t that $\mathbf{E}\left[V_{T_2}\right] \leq T_2 - T_1$. In combination with (4), using the definition of α, and summing over all i, T_1, and T_2, this proves the lemma. $\square$

Lemma 5 *Choose $\alpha \leq \sqrt{TK/S_0}$, $\beta \leq 1/8$, $\eta \leq \frac{1}{12K}$, and $2K\eta \leq \gamma \leq 1/6$. Then for all segments $(T_1, T_2]$,*

$$\begin{aligned}\sum_{t=T_1+1}^{T_2} x_{i(t)}(t) \quad \geq \quad & \frac{1-\gamma-2K\eta}{\eta} \max_i \ln \frac{w_i(T_2+1)}{w_i(T_1+1)} \\ & - (T_2 - T_1)\left(\alpha\sqrt{\frac{KS_0}{T}} + \frac{\alpha^2 S_0}{T} + \frac{\beta}{\eta}\right) \\ & - \frac{1}{\eta}\ln\frac{K(e+\beta)}{\beta}.\end{aligned}$$

Proof.

$$\begin{aligned}
&\frac{W(t+1)}{W(t)} \\
&= \sum_{i=1}^{K} \frac{w_i(t+1)}{W(t)} \\
&= \sum_{i=1}^{K} \frac{w_i(t)}{W(t)} \cdot \exp\left\{\eta\left(\hat{x}_i(t) + \frac{\alpha}{p_i(t)\sqrt{TK/S_0}}\right)\right\} + \beta \\
&\leq \sum_{i=1}^{K} \frac{p_i(t)-\gamma/K}{1-\gamma} \cdot \left(1 + \eta\hat{x}_i(t) + \frac{\alpha\eta}{p_i(t)}\sqrt{\frac{S_0}{TK}} + 2\eta^2(\hat{x}_i(t))^2 + \frac{2\alpha^2\eta^2 S_0}{(p_i(t))^2 TK}\right) + \beta \\
&\qquad \text{(since } \exp\{a+b\} \leq 1 + a + b + 2a^2 + 2b^2 \text{ for } 0 \leq a, b \leq 1/2, \\
&\qquad \eta\hat{x}_i(t) \leq \eta/p_i(t) \leq \eta(K/\gamma) \leq 1/2, \text{ and } (\eta/p_i(t))(\alpha/\sqrt{TK/S_0}) \leq 1/2) \\
&\leq 1 + \sum_{i=1}^{K} \frac{p_i(t)}{1-\gamma} \cdot \left(\eta\hat{x}_i(t) + \frac{\alpha\eta}{p_i(t)}\sqrt{\frac{S_0}{TK}} + 2\eta^2(\hat{x}_i(t))^2 + \frac{2\alpha^2\eta^2 S_0}{(p_i(t))^2 TK}\right) + \beta \\
&= 1 + \frac{\eta}{1-\gamma}\sum_{i=1}^{K} p_i(t)\hat{x}_i(t) + \frac{\alpha\eta}{1-\gamma}\sqrt{\frac{KS_0}{T}} + \frac{2\eta^2}{1-\gamma}\sum_{i=1}^{K} p_i(t)(\hat{x}_i(t))^2 \\
&\quad + \frac{2\alpha^2\eta^2}{1-\gamma}\frac{S_0}{TK}\sum_{i=1}^{K}\frac{1}{p_i(t)} + \beta \\
&\leq 1 + \frac{\eta}{1-\gamma}x_{i(t)}(t) + \frac{\alpha\eta}{1-\gamma}\sqrt{\frac{KS_0}{T}} + \frac{2\eta^2}{1-\gamma}\sum_{i=1}^{K}\hat{x}_i(t) + \frac{\alpha^2\eta}{1-\gamma}\frac{S_0}{T} + \beta\ .
\end{aligned}$$

For the last inequality we used the definition of $\hat{x}_i(t)$ and $p_i(t) \geq \gamma/K \geq 2\eta$. Since $\ln(1+x) \leq x$ for $x \leq 1$, taking logarithms and summing over $t = T_1+1, \ldots, T_2$ yields

$$\begin{aligned}\ln\frac{W(T_2+1)}{W(T_1+1)} \quad \leq \quad & \frac{\eta}{1-\gamma}\sum_{t=T_1+1}^{T_2} x_{i(t)}(t) + \frac{\alpha\eta(T_2-T_1)}{1-\gamma}\sqrt{\frac{KS_0}{T}} \\ & + \frac{2\eta^2}{1-\gamma}\sum_{i=1}^{K}\sum_{t=T_1+1}^{T_2}\hat{x}_i(t) + \frac{\alpha^2\eta}{1-\gamma}S_0\frac{T_2-T_1}{T} + \beta(T_2-T_1)\end{aligned}$$

and

$$\begin{aligned}\sum_{t=T_1+1}^{T_2} x_{i(t)}(t) &\geq \frac{1-\gamma}{\eta}\ln\frac{W(T_2+1)}{W(T_1+1)} - 2\eta\sum_{i=1}^{K}\sum_{t=T_1+1}^{T_2}\hat{x}_i(t)\\ &\quad -(T_2-T_1)\left(\alpha\sqrt{\frac{KS_0}{T}}+\frac{\alpha^2 S_0}{T}+\frac{\beta}{\eta}\right).\end{aligned}$$

From the definition of $w_i(t)$ we find that $w_i(t+1)/w_i(t) \geq \exp\{\eta\hat{x}_i(t)\}$. Thus

$$\eta\sum_{i=1}^{K}\sum_{t=T_1+1}^{T_2}\hat{x}_i(t) \leq K\max_i \eta\sum_{t=T_1+1}^{T_2}\hat{x}_i(t) \leq K\max_i \ln\frac{w_i(T_2+1)}{w_i(T_1+1)}\ .$$

Again from the definition of $w_i(t)$ we find that

$$\begin{aligned}w_i(t+1) &= w_i(t)\cdot\exp\left\{\eta\hat{x}_i(t)+(\eta/p_i(t))(\alpha/\sqrt{TK/S_0})\right\}+\frac{\beta}{K}W(t)\\ &\leq w_i(t)\cdot\exp\{1\}+\beta W(t)\end{aligned}$$

since $\eta\hat{x}_i(t)\leq 1/2$ and $(\eta/p_i(t))(\alpha/\sqrt{TK/S_0})\leq 1/2$. Thus $W(t+1)=\sum_{i=1}^{k} w_i(t+1)\leq \sum_{i=1}^{k} w_i(t)\cdot\exp\{1\}+\beta W(t)=(e+\beta)W(t)$ and $w_i(t+1)\geq\frac{\beta}{K}W(t)$ and we get

$$\ln\frac{W(T_2+1)}{W(T_1+1)} \geq \ln\frac{w_i(T_2+1)}{W(T_1+1)} \geq \ln\frac{w_i(T_2+1)}{(e+\beta)W(T_1)} \geq \ln\frac{w_i(T_2+1)}{w_i(T_1+1)}+\ln\frac{\beta}{K(e+\beta)}\ .$$

Collecting terms gives the statement of the lemma. □

We are now ready to prove Theorem 2 by combining Lemmas 4 and 5 and substituting the parameters. We recall that $\gamma = 2K\eta$. With probability at least $1-\delta$ we have for each segment $(T_1, T_2]$ that

$$\begin{aligned}\sum_{t=T_1+1}^{T_2} x_{i(t)}(t) &\geq (1-\gamma-2K\eta)\left(\sum_{t=T_1+1}^{T_2} x_i(t)-\alpha\sqrt{TK/S_0}\right)\\ &\quad -(T_2-T_1)\left(\alpha\sqrt{\frac{KS_0}{T}}+\frac{\alpha^2 S_0}{T}+\frac{\beta}{\eta}\right)\\ &\quad -\frac{1}{\eta}\ln\frac{K(e+\beta)}{\beta}\\ &\geq \sum_{t=T_1+1}^{T_2} x_i(t)-\alpha\sqrt{TK/S_0}-\frac{1}{\eta}\ln\frac{K(e+\beta)}{\beta}\\ &\quad -(T_2-T_1)\left(4K\eta+\alpha\sqrt{\frac{KS_0}{T}}+\frac{\alpha^2 S_0}{T}+\frac{\beta}{\eta}\right)\\ &\geq \sum_{t=T_1+1}^{T_2} x_i(t)-\alpha\sqrt{\frac{TK}{S_0}}-\frac{\alpha}{2}\sqrt{\frac{TK}{S_0}}\end{aligned}$$

$$
\begin{aligned}
&- (T_2 - T_1)\left(2\alpha\sqrt{\frac{KS_0}{T}} + \alpha\sqrt{\frac{KS_0}{T}} + \frac{\alpha}{6}\sqrt{\frac{S_0}{T}} + \frac{\alpha}{2}\sqrt{\frac{K}{TS_0}} \right) \\
\geq\ & \sum_{t=T_1+1}^{T_2} x_i(t) - \frac{3}{2}\alpha\sqrt{\frac{TK}{S_0}} - 4\alpha(T_2 - T_1)\sqrt{\frac{KS_0}{T}}.
\end{aligned}
$$

Now we sum over all S segments $(0, t_1], (t_1, t_2], \ldots, (t_{S-1}, T]$ and get the theorem.

4. Associative Reinforcement Learning with Linear Value Functions

The second model for an exploitation-exploration trade-off that we consider in this paper is an extension of a model proposed by Abe and Long (1999). It is also a special case of the reinforcement learning model (Kaelbling, 1994a,b, Sutton and Barto, 1998). In this model again a learning algorithm has to choose an alternative $i(t) \in \{1, \ldots, K\}$ in each trial $t = 1, \ldots, T$, observes the reward $x_{i(t)}(t)$ of the chosen alternative $i(t)$, and still tries to maximize its cumulative reward $\sum_{t=1}^{T} x_{i(t)}(t)$. The significant difference in comparison with the bandit problem is that the algorithm is provided with additional information. For each alternative i a *feature vector* $\mathbf{z}_i(t) \in \mathbf{R}^d$ is given to the learning algorithm and the algorithm chooses an alternative based on these feature vectors. The meaning of this feature vector is that it describes the expected reward for alternative i in trial t: it is assumed that there is an unknown weight vector $\mathbf{f} \in \mathbf{R}^d$ which is fixed for all trials and alternatives, such that $\mathbf{f} \cdot \mathbf{z}_i(t)$ gives the expected reward $\mathbf{E}\left[x_i(t)\right]$ for all $i \in \{1, \ldots, K\}$ and all $t = 1, \ldots, T$. This means that all $x_i(t)$ are assumed to be independent random variables with expectation $\mathbf{E}\left[x_i(t)\right] = \mathbf{f} \cdot \mathbf{z}_i(t)$.

In this model we compare the performance of a learning algorithm with the performance of an optimal strategy which knows the weight vector $\mathbf{f}$. Such an optimal strategy will choose alternative $i^*(t)$ which maximizes $\mathbf{f} \cdot \mathbf{z}_i(t)$. Thus the loss of a learning algorithm against this optimal strategy is given by

$$B(T) = \sum_{t=1}^{T} x_{i^*(t)}(t) - \sum_{t=1}^{T} x_{i(t)}(t). \tag{5}$$

Using the terms of reinforcement learning the current feature vectors $\mathbf{z}_1(t), \ldots, \mathbf{z}_K(t)$ represent the state of the environment and the choice of an alternative represents the action of the learning algorithm. Compared with reinforcement learning the main restriction of our model is that it does not capture how actions might influence future states of the environment. We also assume that the value function (which gives the expected reward) is a linear function of the feature vectors. This is often a quite reasonable assumption, provided that the feature vectors were designed appropriately (Sutton and Barto, 1998).

As an example that motivates our model we mention the problem of choosing internet banner ads (Abe and Long, 1999, Abe and Nakamura, 1999). An internet ad server has the goal to display ads which the user is likely to click on. It is reasonable to suppose that the probability of a click can be approximated by a linear function of a combination of the user's and the ad's features. If the ad server is able to learn this linear function then it can select ads which are most likely to be clicked on by the user.

Compared with the bandit problem, associative reinforcement learning with linear value functions might seem easier since additional information is available for the learning algorithm. But this advantage is balanced by the much harder evaluation criterion: for the bandit problem (without shifts) the learning algorithm has to compete only with the single best alternative, whereas in the reinforcement learning model the algorithm has to compete with a strategy which might choose different alternatives in each trial depending on the feature vectors. Thus the reinforcement learning algorithm needs to learn and approximate the unknown weight vector $\mathbf{f}$.

The exploitation-exploration trade-off in the associative reinforcement learning model is more subtle than for the bandit problem. Observing the feature vectors the learning algorithm might either go with the alternative which looks best given the past observations, or it might choose an alternative which does not look best but which provides more information about the unknown weight vector $\mathbf{f}$. Again we use confidence bounds to deal with this trade-off, but the application of confidence bounds is more involved than for the bandit problem. Nevertheless, we can improve the bounds given by Abe and Long (1999), which stated that $\mathbf{E}\left[B(T)\right] = O\left(T^{3/4}\right)$: for our algorithm we prove that $B(T) = O\left((Td)^{1/2} \cdot \ln T\right)$ with high probability. The appearance of d (the dimension of the weight vector $\mathbf{f}$ and the feature vectors $\mathbf{z}_i(t)$) is necessary since Abe and Long (1999) showed a lower bound of $\mathbf{E}\left[B(T)\right] = \Omega\left(T^{3/4}\right)$ when $d = \Omega\left(T^{1/2}\right)$.

4.1 An Algorithm for Associative Reinforcement Learning with Linear Value Functions

We denote by $||\cdot||$ the Euclidean norm and assume that the unknown weight vector satisfies $||\mathbf{f}|| \leq 1$ and that all feature vectors also satisfy $||\mathbf{z}_i(t)|| \leq 1$. Furthermore we assume that the rewards $x_i(t)$ are bounded in $[0, 1]$. If these conditions are not satisfied than an appropriate scaling gives similar results. We will make no further assumptions about the feature vectors $\mathbf{z}_i(t)$.

In the following we will need to do some linear algebra. We assume that the feature vectors are column vectors with $\mathbf{z}_i(t) \in \mathbf{R}^{d\times 1}$. We denote by Z' the transposed matrix of Z and we denote by $\Delta\left(\lambda_1, \ldots, \lambda_d\right)$ the diagonal matrix with the elements $\lambda_1, \ldots, \lambda_d$.

Our algorithm LinRel (Figure 2) calculates upper confidence bounds (9) for the means $\mathbf{E}\left[x_i(t)\right] = \mathbf{f} \cdot \mathbf{z}_i(t)$ and chooses the alternative with the largest upper confidence bound, again trading off exploitation controlled by the estimation of the mean, and exploration controlled by the width of the confidence interval. The main idea of the algorithm is to estimate the mean $\mathbf{E}\left[x_i(t)\right]$ from a weighted sum of previous rewards. For this we write a feature vector $\mathbf{z}_i(t)$ as a linear combination of some previously chosen feature vectors $\mathbf{z}_{i(\tau)}(\tau)$ where $\tau \in \Psi(t) \subseteq \{1, \ldots, t-1\}$ (except for d trials this is always possible),

$$\mathbf{z}_i(t) = \sum_{\tau\in\Psi(t)} a_i(\tau)\; \mathbf{z}_{i(\tau)}(\tau) = Z(m) \cdot \mathbf{a}_i(m)'$$

for some $\mathbf{a}_i(t) \in \mathbf{R}^{1\times|\Psi(t)|}$, where $Z(t)$ is a matrix of previously chosen feature vectors as defined in Figure 2. Then

$$\mathbf{f} \cdot \mathbf{z}_i(t) = \sum_{\tau\in\Psi(t)} a_i(\tau)\;\left(\mathbf{f} \cdot \mathbf{z}_{i(\tau)}(\tau)\right) = \sum_{\tau\in\Psi(t)} a_i(\tau)\; \mathbf{E}\left[x_{i(\tau)}(\tau)\right] = \mathbf{E}\left[\mathbf{x}(t)\right] \cdot \mathbf{a}_i(t)'$$

where $\mathbf{x}(t)$ is the vector of previous rewards as defined in Figure 2. Thus $\mathbf{x}(t) \cdot \mathbf{a}_i(t)'$ is a good estimate for $\mathbf{f} \cdot \mathbf{z}_i(t)$. To get a narrow confidence interval we need to keep the variance of this estimate small. For calculating this variance we would like to view $\mathbf{x}(t) \cdot \mathbf{a}_i(t)'$ as a sum of independent random variables $x_{i(\tau)}(\tau)$ with coefficients $a_i(\tau)$. Unfortunately this is not true for the vanilla version of our algorithm where $\Psi(t) = \{1, \ldots, t-1\}$ since previous rewards influence following choices. To achieve independence we will have to choose $\Psi(t)$ more carefully, the details will be given later. For now we assume independence and since $x_{i(\tau)}(\tau) \in [0,1]$, the variance of this estimate is bounded by $||\mathbf{a}_i(t)||^2 /4$. Thus we are interested in a linear combination for $\mathbf{z}_i(t)$ which minimizes $||\mathbf{a}_i(t)||^2$. Minimizing $||\mathbf{a}_i(t)||^2$ under the constraint $\mathbf{z}_i(t) = Z(t) \cdot \mathbf{a}_i(t)'$ gives

$$\mathbf{a}_i(t) = \mathbf{z}_i(t)' \cdot \left(Z(t) \cdot Z(t)'\right)^{-1} \cdot Z(t) \ . \tag{6}$$

(Here we assume that $Z(t) \cdot Z(t)'$ is invertible which is not necessarily true. In the next paragraph we deal with this issue.) Then we get $\mathbf{x}(t) \cdot \mathbf{a}_i(t)' = \mathbf{x}(t) \cdot Z(t)' \cdot (Z(t) \cdot Z(t)')^{-1} \cdot \mathbf{z}_i(t)$ as estimate for $\mathbf{E}[x_i(t)]$. It is interesting to notice that this can be written as $\mathbf{x}(t) \cdot \mathbf{a}_i(t)' = \hat{\mathbf{f}} \cdot \mathbf{z}_i(t)$ where $\hat{\mathbf{f}} = \mathbf{x}(t) \cdot Z(t)' \cdot (Z(t) \cdot Z(t)')^{-1}$ is the least square estimate of the linear model $\mathbf{E}[\mathbf{x}(t)] = \mathbf{f} \cdot Z(t)$.

To get confidence bounds we now need to upper bound $||\mathbf{a}_i(t)||^2$. It turns out that we get useful bounds only if $Z(t) \cdot Z(t)'$ is sufficiently regular in the sense that all eigenvalues are sufficiently large. If some of the eigenvalues are small we have to deal with them separately[5]. This is the reason why we do not use (6) to calculate $\mathbf{a}_i(t)$ but use the more complicated (7), see Figure 2. Essentially (7) projects the feature vectors into the linear subspace of eigenvectors with large eigenvalues and ignores directions which correspond to eigenvectors with small eigenvalues.

Our goal is to bound the performance of algorithm LinRel by $\tilde{O}\left(\sqrt{Td}\right)$. Unfortunately we are not able to show such a bound for the vanilla version of our algorithm with $\Psi(t) = \{1, \ldots, t-1\}$. Instead, we will use a master algorithm SupLinRel which uses LinRel as a subroutine with appropriate choices for $\Psi(t)$ and for which we can prove appropriate performance bounds. The algorithm SupLinRel is presented and analysed in Section 4.3. Theorem 6 below gives the performance bound for SupLinRel. We believe that for most practical applications the simpler algorithm LinRel with $\Psi(t) = \{1, \ldots, t-1\}$ achieves the same — or even better — performance, but there are artificial scenarios where this might be not true.

Theorem 6 *We use the notation of (5) and Figure 3. When algorithm* SupLinRel *is run with parameter $\delta/(1+\ln T)$ then with probability $1-\delta$ the regret of the algorithm is bounded by*

$$B(T) \le 44 \cdot (1 + \ln(2KT \ln T))^{3/2} \cdot \sqrt{Td} + 2\sqrt{T} \ .$$

Remark 7 *A similar result as in Theorem 6 can be obtained when the mean of each alternative i is governed by a separate weight vector $\mathbf{f}_i$ such that $\mathbf{E}\left[x_{i(t)}(t)\right] = \mathbf{f}_i \cdot \mathbf{z}_{i(t)}(t)$.*

5. It seems that making $Z(t) \cdot Z(t)'$ regular by adding a multiple of the identity matrix as in ridge regression is not sufficient. This would result in too small confidence intervals by overestimating the confidence. But this regards only the theoretical analysis, and we believe that in most practical applications its is sufficient to add in the identity matrix.

Algorithm LINREL
Parameters: $\delta \in [0,1]$, the number of trials T.
Inputs:
The indexes of selected feature vectors, $\Psi(t) \subseteq \{1, \ldots, t-1\}$.
The new feature vectors $\mathbf{z}_1(t), \ldots, \mathbf{z}_K(t)$.

1. Let $Z(m) = \left(\mathbf{z}_{i(\tau)}(\tau)\right)_{\tau \in \Psi(t)}$ be the matrix of selected feature vectors and $\mathbf{x}(m) = \left(x_{i(\tau)}(\tau)\right)_{\tau \in \Psi(t)}$ the vector of corresponding rewards.
2. Calculate the eigenvalue decomposition
$$Z(t) \cdot Z(t)' = U(t)' \cdot \Delta\left(\lambda_1(t), \ldots, \lambda_d(t)\right) \cdot U(t)$$
where $\lambda_1(t), \ldots, \lambda_k(t) \geq 1$, $\lambda_{k+1}(t), \ldots, \lambda_d(t) < 1$, and $U(t)' \cdot U(t) = \Delta(1, \ldots, 1)$.
3. For each feature vector $\mathbf{z}_i(t)$ set $\tilde{\mathbf{z}}_i(t) = (\tilde{z}_{i,1}(t), \ldots, \tilde{z}_{i,d}(t))' = U(t) \cdot \mathbf{z}_i(t)$ and $\tilde{\mathbf{u}}_i(t) = (\tilde{z}_{i,1}(t), \ldots, \tilde{z}_{i,k}(t), 0, \ldots)'$, $\tilde{\mathbf{v}}_i(t) = (0, \ldots, 0, \tilde{z}_{i,k+1}(t), \ldots, \tilde{z}_{i,d}(t))'$.
4. Calculate
$$\mathbf{a}_i(t) = \tilde{\mathbf{u}}_i(t)' \cdot \Delta\left(\frac{1}{\lambda_1(t)}, \ldots, \frac{1}{\lambda_k(t)}, 0, \ldots, 0\right) \cdot U(t) \cdot Z(t). \tag{7}$$
5. Calculate the upper confidence bounds and its widths, $i = 1, \ldots, K$,
$$\begin{aligned} \mathrm{width}_i(t) &= ||\mathbf{a}_i(t)|| \left(\sqrt{\ln(2TK/\delta)}\right) + ||\tilde{\mathbf{v}}_i(t)||, & (8) \\ \mathrm{ucb}_i(t) &= \mathbf{x}(t) \cdot \mathbf{a}_i(t)' + \mathrm{width}_i(t). & (9) \end{aligned}$$
6. Choose that alternative $i(t)$ which maximizes the upper confidence bound $\mathrm{ucb}_i(t)$.

Figure 2: Algorithm LINREL

4.2 Analysis of Algorithm LinRel

Before we turn to the master algorithm SUPLINREL in the next section, we first analyse its main ingredient, the algorithm LINREL.

At first we show that (9) is indeed a confidence bound on the mean $\mathbf{E}\left[x_{i(t)}(t)\right]$. For this we use the Azuma-Hoeffding bound.

Lemma 8 (Azuma, 1967, Alon and Spencer, 1992) *Let $X_1, \ldots, X_m$ be random variables with $|X_\tau| \leq a_\tau$ for some $a_1, \ldots, a_m > 0$. Then*
$$\mathbf{P}\left\{\left|\sum_{\tau=1}^{m} X_\tau - \sum_{\tau=1}^{m} \mathbf{E}\left[X_\tau | X_1, \ldots, X_{\tau-1}\right]\right| \geq B\right\} \leq 2\exp\left\{-\frac{B^2}{2\sum_{\tau=1}^{m} a_\tau^2}\right\}.$$

Lemma 9 *We use the notation of Figure 2. Let $\Psi(t)$ be constructed in such a way that for fixed $\mathbf{z}_{i(\tau)}(\tau)$, $\tau \in \Psi(t)$, the rewards $x_{i(\tau)}(\tau)$, $\tau \in \Psi(t)$, are independent random variables with means $\mathbf{E}\left[x_{i(\tau)}(\tau)\right] = \mathbf{f} \cdot \mathbf{z}_{i(\tau)}(\tau)$. Then with probability $1 - \delta/T$ we have that for all $i \in \{1, \ldots, K\}$,*

$$|\mathbf{x}(t) \cdot \mathbf{a}_i(t)' - \mathbf{f} \cdot \mathbf{z}_i(t)| \le ||\mathbf{a}_i(t)|| \left(\sqrt{2\ln(2TK/\delta)}\right) + ||\tilde{\mathbf{v}}_i(t)|| \,.$$

Remark 10 *With such a construction of $\Psi(t)$ we will deal in Section 4.3. Observe that the independence of $\mathbf{a}_i(t)$ and $\mathbf{x}(t)$ is crucial in the following proof.*

Proof of Lemma 9. For each $i = 1, \ldots, K$ we use Lemma 8 with $X_\tau = x_{i(\tau)}(\tau) \cdot a_i(\tau)$. Note that $a_i(\tau)$ only depends on $Z(t)$ and $\mathbf{z}_i(t)$, but not on $\mathbf{x}(t)$ ([6]). Then $|X_\tau| \le a_i(\tau)|$,

$$\sum_{\tau \in \Psi(t)} X_\tau = \mathbf{x}(t) \cdot \mathbf{a}_i(t)',$$

$$\sum_{\tau \in \Psi(t)} \mathbf{E}\left[X_\tau | (X_\sigma)_{\sigma \in \Psi(t), \sigma < \tau}\right] = \sum_{\tau \in \Psi(t)} \mathbf{E}\left[X_\tau\right] = \sum_{\tau \in \Psi(t)} \mathbf{f} \cdot \mathbf{z}_{i(\tau)}(\tau) \cdot a_i(\tau) = \mathbf{f} \cdot Z(t) \cdot \mathbf{a}_i(t)',$$

and

$$\mathbf{P}\left\{\left|\mathbf{x}(t) \cdot \mathbf{a}_i(t)' - \mathbf{f} \cdot Z(t) \cdot \mathbf{a}_i(t)'\right| \ge ||\mathbf{a}_i(t)|| \left(\sqrt{2\ln(2TK/\delta)}\right)\right\} \le \frac{\delta}{TK} \,.$$

Since

$$\begin{aligned}
\mathbf{z}_i(t) &= U(t)' \cdot \tilde{\mathbf{z}}_i(t) \\
&= U(t)' \cdot \tilde{\mathbf{u}}_i(t) + U(t)' \cdot \tilde{\mathbf{v}}_i(t) \\
&= U(t)' \cdot \Delta\left(\lambda_1(t), \ldots, \lambda_d(t)\right) \\
&\quad \cdot U(t) \cdot \left(Z(t) \cdot Z(t)'\right)^{-1} \cdot Z(t) \cdot Z(t)' \cdot U(t)' \\
&\quad \cdot \Delta\left(\frac{1}{\lambda_1(t)}, \ldots, \frac{1}{\lambda_k(t)}, 0, \ldots, 0\right) \cdot \tilde{\mathbf{u}}_i(t) \\
&\quad + U(t)' \cdot \tilde{\mathbf{v}}_i(t) \\
&= \left(Z(t) \cdot Z(t)'\right) \cdot \left(Z(t) \cdot Z(t)'\right)^{-1} \cdot Z(t) \cdot \mathbf{a}_i(t)' + U(t)' \cdot \tilde{\mathbf{v}}_i(t) \\
&= Z(t) \cdot \mathbf{a}_i(t)' + U(t)' \cdot \tilde{\mathbf{v}}_i(t)
\end{aligned}$$

and $||\mathbf{f}|| \le 1$ we have

$$|\mathbf{f} \cdot \mathbf{z}_i(t) - \mathbf{f} \cdot Z(t) \cdot \mathbf{a}_i(t)'| \le ||\mathbf{f}|| \cdot \left|\left|U(t)' \cdot \tilde{\mathbf{v}}_i(t)\right|\right| \le ||\tilde{\mathbf{v}}_i(t)|| \,.$$

Summing over i gives the lemma. □

By Lemma 9 we can bound the expected loss of the algorithm's choice against the optimal choice in terms of the $\mathbf{a}_i(t)$ and $\tilde{\mathbf{v}}_i(t)$. To proceed with the analysis of LINREL we need to bound $\left|\left|\mathbf{a}_{i(t)}(t)\right|\right|$ and $\left|\left|\tilde{\mathbf{v}}_{i(t)}(t)\right|\right|$. For this we show that $\left|\left|\mathbf{a}_{i(t)}(t)\right|\right|$ and $\left|\left|\tilde{\mathbf{v}}_{i(t)}(t)\right|\right|$ are related to the amount of change between the eigenvalues of $Z(t) \cdot Z(t)'$ and $Z(t+1) \cdot Z(t+1)'$ if $\Psi(t+1) = \Psi(t) \cup \{t\}$. In the following lemma $\left|\left|\mathbf{a}_{i(t)}(t)\right|\right|$ is bounded by the relative change of the eigenvalues larger than 1 and $\left|\left|\tilde{\mathbf{v}}_{i(t)}(t)\right|\right|$ is bounded by the absolute change of the eigenvalues smaller than 5.

6. If a reward $x_{i(\tau)}(\tau)$ would influence following choices of alternatives then it would also influence $Z(t)$ and thus the coefficients $a_i(\tau)$. In such a case the Azuma-Hoeffding were not applicable.

Lemma 11 *Let $\Psi(t+1) = \Psi(t) \cup \{t\}$. The eigenvalues $\lambda_1(t), \ldots, \lambda_d(t)$ of $Z(t) \cdot Z(t)'$ and the eigenvalues of $\lambda_1(t+1), \ldots, \lambda_d(t+1)$ of $Z(t+1) \cdot Z(t+1)'$ can be arranged in such a way that*

$$\lambda_j(t) \leq \lambda_j(t+1) \tag{10}$$

and

$$\left|\left|\mathbf{a}_{i(t)}(t)\right|\right|^2 \ \leq\ 10 \sum_{j:\lambda_j(t)\geq 1} \frac{\lambda_j(t+1)-\lambda_j(t)}{\lambda_j(t)}, \tag{11}$$

$$\left|\left|\tilde{\mathbf{v}}_{i(t)}(t)\right|\right|^2 \ \leq\ 4 \sum_{j:\lambda_j(t+1)<5} \left[\lambda_j(t+1)-\lambda_j(t)\right]. \tag{12}$$

The proof of this lemma is somewhat technical and is therefore given in Appendix A. Some intuition about the lemma and its proof can be gained from the following observations. First, we get from (7) that

$$\left|\left|\mathbf{a}_{i(t)}(t)\right|\right|^2 = \mathbf{a}_{i(t)}(t) \cdot \mathbf{a}_{i(t)}(t)' = \tilde{\mathbf{u}}_i(t)' \cdot \Delta\left(\frac{1}{\lambda_1(t)}, \ldots, \frac{1}{\lambda_k(t)}, 0, \ldots, 0\right) \cdot \tilde{\mathbf{u}}_i(t)\ . \tag{13}$$

Next, the sum of the eigenvalues of $Z(t+1) \cdot Z(t+1)'$ equals the sum of the eigenvalues of $Z(t) \cdot Z(t)'$ plus $\left|\left|\mathbf{z}_{i(t)}(t)\right|\right|^2 = \left|\left|\tilde{\mathbf{u}}_{i(t)}(t)\right|\right|^2 + \left|\left|\tilde{\mathbf{v}}_{i(t)}(t)\right|\right|^2$, as stated in the following lemma:

Lemma 12 *If $\Psi(t+1) = \Psi(t) \cup \{t\}$ then for the eigenvalues of $Z(t) \cdot Z(t)'$ and $Z(t+1) \cdot Z(t+1)'$ the following holds:*

$$\begin{aligned} \sum_{j=1}^{d} \lambda_j(t+1) \ &=\ \sum_{j=1}^{d} \lambda_j(t) + \left|\left|\mathbf{z}_{i(t)}(t)\right|\right|^2 \qquad (14) \\ &=\ \sum_{j=1}^{d} \lambda_j(t) + \left|\left|\tilde{\mathbf{u}}_{i(t)}(t)\right|\right|^2 + \left|\left|\tilde{\mathbf{v}}_{i(t)}(t)\right|\right|^2\ . \end{aligned}$$

In the proof of Lemmas 11 and 12 (given in Appendix A) we will show that an even stronger statement holds, namely $\lambda_j(t+1) \approx \lambda_j(t) + \tilde{z}_{i(t),j}(t)^2$. From this and (13) Lemma 11 can be derived. In a more abstract view the eigenvalues of $Z(t) \cdot Z(t)'$ serve as a potential function for $\sum_{\tau\in\Psi(t)} \left|\left|\mathbf{a}_{i(\tau)}(\tau)\right|\right| + \sum_{\tau\in\Psi(t)} \left|\left|\tilde{\mathbf{v}}_{i(t)}(\tau)\right|\right|$: the eigenvalues of $Z(t) \cdot Z(t)'$ grow with this sum. From (14) we get that the sum of eigenvalues $\sum_{j=1}^{d} \lambda_j(T+1)$ is bounded by $\Psi(T+1)$,

$$\sum_{j=1}^{d} \lambda_j(T+1) \leq |\Psi(T+1)|\ , \tag{15}$$

since $||\mathbf{z}_i(t)|| \leq 1$. With this observation we can bound $\sum_{t\in\Psi(T+1)} \left|\left|\mathbf{a}_{i(t)}(t)\right|\right|$ and $\sum_{t\in\Psi(T+1)} \left|\left|\tilde{\mathbf{v}}_{i(t)}(t)\right|\right|$.

Lemma 13 *With the notation of Figure 2 we have*

$$\sum_{t\in\Psi(T+1)} \left|\left|\mathbf{a}_{i(t)}(t)\right|\right| \ \leq\ 2\sqrt{5d|\Psi(T+1)|\ln|\Psi(T+1)|}\ ,$$

$$\sum_{t\in\Psi(T+1)} \left|\left|\tilde{\mathbf{v}}_{i(t)}(t)\right|\right| \ \leq\ 5\sqrt{d|\Psi(T+1)|}\ .$$

Proof. Let the eigenvalues $\lambda_j(t)$, $j = 1, \ldots, d$, $t \in \Psi(T+1)$, be arranged such that (10), (11) and (12) hold. Then

$$\begin{aligned}
\sum_{t\in\Psi(T+1)} \left|\left|\mathbf{a}_{i(t)}(t)\right|\right| &\leq \sum_{t\in\Psi(T+1)} \sqrt{10 \sum_{j:\lambda_j(t)\geq 1} \left(\frac{\lambda_j(t+1)}{\lambda_j(t)} - 1\right)}\,, \\
\sum_{t\in\Psi(T+1)} \left|\left|\tilde{\mathbf{v}}_{i(t)}(t)\right|\right| &\leq \sum_{t\in\Psi(T+1)} \sqrt{4 \sum_{j:\lambda_j(t+1)<5} \left[\lambda_j(t+1) - \lambda_j(t)\right]}\,.
\end{aligned}$$

It is not hard to see that $\sum_{t\in\Psi} \sqrt{\sum_{j=1}^{d}(h_{j,t} - 1)}$ is maximal under the constraints $h_{j,t} \geq 1$ and $\sum_{j=1}^{d} \prod_{t\in\Psi} h_{j,t} \leq |\Psi|$, when $h_{j,t} = (|\Psi|/d)^{1/|\Psi|}$ for all t and j. Since

$$\sum_{j=1}^{d} \prod_{t:\lambda_j(t)\geq 1} \frac{\lambda_j(t+1)}{\lambda_j(t)} \leq \sum_{j=1}^{d} \lambda_j(T+1) \leq |\Psi(T+1)|$$

and $(\psi/d)^{1/\psi} - 1 \leq \frac{2}{\psi} \ln \psi$ for $\psi, d \geq 1$, it follows that

$$\begin{aligned}
\sum_{t\in\Psi(T+1)} \left|\left|\mathbf{a}_{i(t)}(t)\right|\right| &\leq |\Psi(T+1)| \sqrt{10d \left(\frac{2}{|\Psi(T+1)|} \ln |\Psi(T+1)|\right)} \\
&= 2\sqrt{5d|\Psi(T+1)| \ln |\Psi(T+1)|}.
\end{aligned}$$

Similarly $\sum_{t\in\Psi} \sqrt{\sum_{j=1}^{d} \Delta_{j,t}}$ is maximal under the constraint $\sum_{t\in\Psi} \Delta_{j,t} \leq 5$ if $\Delta_{j,t} = 5/|\Psi|$. Thus

$$\sum_{t\in\Psi(T+1)} \left|\left|\tilde{\mathbf{v}}_{i(t)}(t)\right|\right| \leq |\Psi(T+1)| \sqrt{4 \frac{5d}{|\Psi(T+1)|}} \leq 5\sqrt{d|\Psi(T+1)|}$$

since $\sum_{t:\lambda_j(t+1)<5} [\lambda_j(t+1) - \lambda_j(t)] \leq 5$. $\square$

To get a bound on the performance of our algorithm we want to combine Lemmas 9 and 13: for each trial t Lemma 9 bounds the loss in terms of $\left|\left|\mathbf{a}_{i(t)}(t)\right|\right|$ and $\left|\left|\tilde{\mathbf{v}}_{i(t)}(t)\right|\right|$, and Lemma 13 bounds the sums of $\left|\left|\mathbf{a}_{i(t)}(t)\right|\right|$ and $\left|\left|\tilde{\mathbf{v}}_{i(t)}(t)\right|\right|$. But these sums include only the $t \in \Psi(T+1)$ while we need to bound the loss over all t. Since the $x_{i(t)}(t)$ must be independent for $t \in \Psi(T+1)$ we cannot simply include all t in $\Psi(T+1)$. Thus the master algorithm presented in the next section uses a more complicated scheme which puts the trials $t = 1, \ldots, T$ in one of S sets $\Psi^{(1)}, \ldots, \Psi^{(S)}$.

4.3 The Master Algorithm

The master algorithm SupLinRel maintains S sets of trials, $\Psi^{(1)}(t), \ldots, \Psi^{(S)}(t)$ where each set $\Psi^{(s)}(t)$ contains the trails for which a choice was made at *stage s*. For Lemma 9 to be applicable these sets need to be selected in such a way that for each $\Psi^{(s)}(t)$ the selected rewards $x_{i(t)}(t)$ are independent for all $t \in \Psi^{(s)}(T+1)$. This is achieved by the algorithm described in Figure 3.

Algorithm SupLinRel
Parameters: $\delta \in [0,1]$, the number of trials T.
Initialization: Let $S = \ln T$ and set $\Psi^{(1)}(1) = \cdots = \Psi^{(S)}(1) = \emptyset$.
Repeat for $t = 1, \ldots, T$

1. Initialize the set of feasible alternatives $A_1 := \{1, \ldots, K\}$, set $s := 1$.
2. Repeat until an alternative $i(t)$ is chosen:
   (a) Use LinRel with $\Psi^{(s)}(t)$ to calculate the upper confidence bounds $\mathrm{ucb}_i^{(s)}(t)$ and its widths $\mathrm{width}_i^{(s)}(t)$ for all $i \in A_s$.
   (b) If $\mathrm{width}_i^{(s)}(t) > 2^{-s}$ for some $i \in A_s$ then choose this alternative and store the corresponding trial in $\Psi^{(s)}$,
   $$\Psi^{(s)}(t+1) = \Psi^{(s)}(t) \cup \{t\},\ \Psi^{(\sigma)}(t+1) = \Psi^{(\sigma)}(t) \text{ for } \sigma \neq s.$$
   (c) Else if $\mathrm{width}_i^{(s)}(t) \leq 1/\sqrt{T}$ for all $i \in A_s$ then choose that alternative $i \in A_s$ which maximizes the maximum upper confidence bound $\mathrm{ucb}_i^{(s)}(t)$. Do not store this trial,
   $$\Psi^{(\sigma)}(t+1) = \Psi^{(\sigma)}(t) \text{ for all } \sigma = 1, \ldots, S.$$
   (d) Else if $\mathrm{width}_i^{(s)}(t) \leq 2^{-s}$ for all $i \in A_s$ then set
   $$A_{s+1} = \left\{ i \in A_s \,\middle|\, \mathrm{ucb}_i^{(s)}(t) \geq \max_{j \in A_s} \mathrm{ucb}_j^{(s)}(t) - 2 \cdot 2^{-s} \right\}$$
   and set $s := s + 1$.

Figure 3: Algorithm SupLinRel

The algorithm chooses an alternative either if it is sure that the expected reward of this alternative is close to the optimal expected reward (step 2c: the width of the upper confidence bounds is only $1/\sqrt{T}$), or if the width of the upper confidence bound is so big that more exploration is needed (step 2b). The sets $\Psi^{(s)}$ are arranged such that the allowed width of the confidence intervals in respect to $\Psi^{(s)}$ is 2^{-s}. Thus feature vectors are filtered through the stages $s = 1, \ldots, S$ until some exploration is necessary or until the confidence is sufficiently small. In step 2d only those alternatives which are sufficiently close to the optimal alternative are passed to the next stage $s+1$: if the widths of all alternatives $i \in A$ are at most 2^{-s} and $\mathrm{ucb}_i^{(s)}(t) < \mathrm{ucb}_j^{(s)}(t) - 2 \cdot 2^{-s}$ for some $j \in A$ then i cannot be the optimal alternative. Eliminating alternatives which are obviously bad reduces the possible loss in the next stage.

A main property of SupLinRel is the independence of the rewards $x_{i(\tau)}(\tau)$, $\tau \in \Psi^{(s)}(t)$, for each stage s.

Lemma 14 *For each $s = 1, \ldots, S$, for each $t = 1, \ldots, T$, and for any fixed sequence of feature vectors $\mathbf{z}_{i(\tau)}(\tau)$, $\tau \in \Psi^{(s)}(t)$, the rewards $x_{i(\tau)}(\tau)$, $\tau \in \Psi^{(s)}(t)$, are independent random variables with mean $\mathbf{E}\left[x_{i(\tau)}(\tau)\right] = \mathbf{f} \cdot \mathbf{z}_{i(\tau)}(\tau)$.*

Proof. Only in step 2b a trial t can be added to $\Psi^{(s)}(t)$. If trial t is added to $\Psi^{(s)}(t)$, only depends on the results of trials $\tau \in \bigcup_{\sigma<s} \Psi^{(\sigma)}(t)$ and on $\text{width}_i^{(s)}(t)$. From (8) we find that $\text{width}_i^{(s)}(t)$ only depends on the feature vectors $\mathbf{z}_{i(\tau)}(\tau)$, $\tau \in \Psi^{(s)}(t)$, and on $\mathbf{z}_i(t)$. This implies the lemma. □

The analysis of algorithm SupLinRel is done by a series of lemmas which make the intuition about SupLinRel precise. First we bound the maximal loss of the feasible alternatives at stage s.

Lemma 15 *With probability $1 - \delta S$, for any t and any stage s,*

$$\mathit{ucb}_i^{(s)}(t) - 2 \cdot \mathit{width}_i^{(s)}(t) \leq \mathbf{E}\left[x_i(t)\right] \leq \mathit{ucb}_i^{(s)}(t) \quad \text{for any } i, \tag{16}$$

$$i^*(t) \in A_s \ ,$$

and

$$\mathbf{E}\left[x_{i^*(t)}(t)\right] - \mathbf{E}\left[x_i(t)\right] \leq 8 \cdot 2^{-s} \quad \text{for any } i \in A_s. \tag{17}$$

Proof. Using Lemma 9 and summing over s and t gives (16).

Obviously the lemma holds for $s = 1$. If $s > 1$ then $A_s \subseteq A_{s-1}$ and step 2b implies that $\text{width}_i^{(s-1)}(t) \leq 2^{-(s-1)}$ and $\text{width}_{i^*(t)}^{(s-1)}(t) \leq 2^{-(s-1)}$. Step 2d implies $\text{ucb}_i^{(s-1)}(t) \geq \text{ucb}_{i^*(t)}^{(s-1)}(t) - 2 \cdot 2^{-(s-1)}$. Thus

$$\mathbf{E}\left[x_i(t)\right] \geq \text{ucb}_i^{(s-1)}(t) - 2 \cdot 2^{-(s-1)} \geq \text{ucb}_{i^*(t)}^{(s-1)}(t) - 4 \cdot 2^{-(s-1)} \geq \mathbf{E}\left[x_{i^*(t)}(t)\right] - 4 \cdot 2^{-(s-1)}$$

and

$$\text{ucb}_{i^*(t)}^{(s-1)}(t) \geq \mathbf{E}\left[x_{i^*(t)}(t)\right] \geq \mathbf{E}\left[x_j(t)\right] \geq \text{ucb}_j^{(s-1)}(t) - 2 \cdot 2^{-(s-1)}$$

for any $j \in A_{s-1}$. □

The next lemma bounds the number of trials for which an alternative is chosen at stage s.

Lemma 16 *For all s,*

$$|\Psi^{(s)}(T+1)| \leq 5 \cdot 2^s \left(1 + \ln(2TK/\delta)\right) \sqrt{d|\Psi^{(s)}(T+1)|} \ .$$

Proof. By Lemma 13

$$\begin{aligned}
\sum_{\tau \in \Psi^{(s)}(T+1)} \text{width}_{i(\tau)}^{(s)}(\tau) &\leq 2\sqrt{5d|\Psi^{(s)}(T+1)| \ln |\Psi^{(s)}(T+1)|} \left(\sqrt{\ln(2TK/\delta)}\right) \\
&\quad +5\sqrt{d|\Psi^{(s)}(T+1)|} \\
&\leq 5\left(1 + \ln(2TK/\delta)\right) \sqrt{d|\Psi^{(s)}(T+1)|} \ .
\end{aligned}$$

By step 2b of SupLinRel

$$\sum_{\tau \in \Psi^{(s)}(T+1)} \mathrm{width}^{(s)}_{i(\tau)}(\tau) \geq 2^{-s} |\Psi^{(s)}(T+1)| \ .$$

Combining these two gives the lemma. □

We are now ready for the proof of Theorem 6.

Proof of Theorem 6. Let Ψ_0 be the set of trials for which an alternative is chosen in step 2c. Since $2^{-S} \leq 1/\sqrt{T}$ we have $\{1, \ldots, T\} = \Psi_0 \cup \bigcup_s \Psi^{(s)}(T+1)$. Thus by Lemmas 15 and 16,

$$\begin{aligned}
\mathbf{E}\left[B(T)\right] &= \sum_{t=1}^{T} \left[\mathbf{E}\left[x_{i^*(t)}(t)\right] - \mathbf{E}\left[x_{i(t)}(t)\right]\right] \\
&= \sum_{t \in \Psi_0} \left[\mathbf{E}\left[x_{i^*(t)}(t)\right] - \mathbf{E}\left[x_{i(t)}(t)\right]\right] \\
&\quad + \sum_{s=1}^{S} \sum_{t \in \Psi^{(s)}(T+1)} \left[\mathbf{E}\left[x_{i^*(t)}(t)\right] - \mathbf{E}\left[x_{i(t)}(t)\right]\right] \\
&\leq \frac{2}{\sqrt{T}} |\Psi_0| + \sum_{s=1}^{S} 8 \cdot 2^{-s} \cdot |\Psi^{(s)}(T+1)| \\
&\leq \frac{2}{\sqrt{T}} |\Psi_0| + \sum_{s=1}^{S} 40 \cdot (1 + \ln(2TK/\delta)) \cdot \sqrt{d |\Psi^{(s)}(T+1)|} \\
&\leq 2\sqrt{T} + 40 \cdot (1 + \ln(2TK/\delta)) \cdot \sqrt{STd}
\end{aligned}$$

with probability $1 - \delta S$. Applying the Azuma-Hoeffding bound of Lemma 8 with $a_\tau = 2$ and $B = 4\sqrt{T/\delta}$ we get

$$B(T) \leq 2\sqrt{T} + 44 \cdot (1 + \ln(2TK/\delta)) \cdot \sqrt{STd}$$

with probability $1 - \delta(S+1)$. Replacing δ by $\delta/(S+1)S$, substituting $S = \ln T$, and simplifying yields

$$B(T) \leq 2\sqrt{T} + 44 \cdot (1 + \ln(2KT \ln T))^{3/2} \cdot \sqrt{Td}$$

with probability $1 - \delta$. □

5. Conclusion

By the example of two models we have shown how confidence bounds for suitably chosen random variables provide an elegant tool for dealing with exploitation-exploration trade-offs. For the adversarial bandit problem we used confidence bounds to reduce the variance of the algorithm's performance resulting from the algorithm's internal randomization. For associative reinforcement learning with linear value functions we used a deterministic algorithm and the confidence bounds assessed the uncertainty about the random behavior of the environment. In both models we got a significant improvement in performance over previously known algorithms.

Currently algorithm LinRel is empirically evaluated in realistic scenarios. For a practical application of the algorithm the theoretical confidence bounds need to be fine tuned so that they optimally trade off between exploration and exploitation. While the theorems in this paper show the correct magnitude of the considered bounds, it is left to further research to optimize the constants in the bounds. In practical applications these constants make a significant difference.

Acknowledgments

I want to thank my colleagues Nicoló Cesa-Bianchi, Yoav Freund, Rob Schapire, and Thomas Korimort for encouraging discussions, and Ursi Swaton for her help with the preparation of the manuscript. Nicoló also pointed out a major flaw in an earlier version of the proofs. A preliminary version of this paper was presented at FOCS'2000 (Auer, 2000). This work was supported by the ESPRIT Working Group in Neural and Computational Learning II, NeuroCOLT 2 27150. Last but not least I want to thank the anonymous referees for their helpful comments on an earlier version of this paper.

Appendix A. Proof of Lemmas 11 and 12

We start with a simple lemma about the rotation of two coordinates.

Lemma 17 *For any λ_1, λ_2, z, we have $\begin{pmatrix} \lambda_1 & z \\ z & \lambda_2 \end{pmatrix} = U' \cdot \Delta(\lambda_1 + y, \lambda_2 - y) \cdot U$ for some $y \geq 0$ and some matrix U with $U' \cdot U = \Delta(1,1)$. Furthermore, if $\lambda_1 \geq \lambda_2$, then $y \leq \frac{z^2}{\lambda_1 - \lambda_2}$.*

Proof. We calculate the eigenvalues ν_1 and ν_2 of the matrix $\begin{pmatrix} \lambda_1 & z \\ z & \lambda_2 \end{pmatrix}$ as the solutions of equation $(\lambda_1 - \nu)(\lambda_2 - \nu) - z^2 = 0$ and find $\nu_1 = (\lambda_1 + \lambda_2)/2 + \sqrt{(\lambda_1 - \lambda_2)^2/4 + z^2}$, $\nu_2 = (\lambda_1 + \lambda_2)/2 - \sqrt{(\lambda_1 - \lambda_2)^2/4 + z^2}$. Thus $\nu_1 = \lambda_1 + y$ and $\nu_2 = \lambda_2 - y$ with

$$y = (\lambda_2 - \lambda_1)/2 + \sqrt{(\lambda_1 - \lambda_2)^2/4 + z^2}.$$

For $\lambda_1 \geq \lambda_2$ we find

$$\begin{aligned} \sqrt{(\lambda_1 - \lambda_2)^2/4 + z^2} &\leq \sqrt{(\lambda_1 - \lambda_2)^2/4} + z^2 \frac{1}{2\sqrt{(\lambda_1 - \lambda_2)^2/4}} \\ &= (\lambda_1 - \lambda_2)/2 + \frac{z^2}{\lambda_1 - \lambda_2}. \end{aligned}$$

□

The next two lemmas deal with the eigenvalues of matrices $\Delta(\lambda_1, \ldots, \lambda_d) + \mathbf{z} \cdot \mathbf{z}'$.

Lemma 18 *If $\lambda_1 \geq \cdots \geq \lambda_d \geq 0$ then the smallest eigenvalue of $\Delta(\lambda_1, \ldots, \lambda_d) + \mathbf{z} \cdot \mathbf{z}'$ is at least λ_d for any $\mathbf{z} \in \mathbf{R}^{d \times 1}$.*

Proof. Assume that $\mathbf{u}$ is an eigenvector of $\Delta(\lambda_1,\ldots,\lambda_d)+\mathbf{z}\cdot\mathbf{z}'$ with $(\Delta(\lambda_1,\ldots,\lambda_d)+\mathbf{z}\cdot\mathbf{z}')\cdot\mathbf{u}=\nu\mathbf{u}$. Then $\lambda_j u_j + z_j(\mathbf{z}'\cdot\mathbf{u}) = \nu u_j$ for all j. Assume that $\mathbf{z}'\cdot\mathbf{u}>0$. Then there is a j with $u_j z_j > 0$ so that u_j and z_j have the same sign. Thus $\nu>\lambda_j$. If $\mathbf{z}'\cdot\mathbf{u}<0$ then there is a j with $u_j z_j<0$ so that u_j and z_j have different sign. Thus again $\nu>\lambda_j$. If $\mathbf{z}'\cdot\mathbf{u}=0$ then $\nu=\lambda_j$. □

Lemma 19 *Let $\lambda_1\geq\cdots\geq\lambda_d\geq 0$. The eigenvalues $\nu_1,\ldots,\nu_d$ of a matrix $\Delta(\lambda_1,\ldots,\lambda_d)+\mathbf{z}\cdot\mathbf{z}'$ with $||\mathbf{z}||\leq 1$ can be arranged such that there are $y_{h,j}\geq 0$, $1\leq h<j\leq d$, and the following holds:*

$$\nu_j \geq \lambda_j \ ,$$

$$\nu_j = \lambda_j + z_j^2 - \sum_{h=1}^{j-1} y_{h,j} + \sum_{h=j+1}^{d} y_{j,h} \ , \tag{18}$$

$$\sum_{h=1}^{j-1} y_{h,j} \leq z_j^2 \ , \tag{19}$$

$$\sum_{h=j+1}^{d} y_{j,h} \leq \nu_j - \lambda_j \ , \tag{20}$$

$$\sum_{j=1}^{d} \nu_j = \sum_{j=1}^{d} \lambda_j + ||\mathbf{z}||^2 \ . \tag{21}$$

If $\lambda_h > \lambda_j + 1$ then

$$y_{h,j} \leq \frac{z_j^2 z_h^2}{\lambda_h - \lambda_j - 1} \ . \tag{22}$$

The intuition for this lemma is that essentially the new eigenvalue ν_j equals $\lambda_j + z_j^2$, but that a fraction of z_j^2 might be contributed to larger eigenvalue instead of being contributed to ν_j. This is the meaning of the quantities $y_{j,h}$: the eigenvalue ν_j receives the amount $y_{j,h}$ from z_h^2 and gives the amount $y_{h,j}$ to a larger eigenvalue.

Proof. Clearly (18) implies (21) and (18), (19) imply (20). We prove Lemma 19 by induction on d. We apply Lemma 17 repeatedly to obtain the following transformation:

$$\begin{aligned}
\Delta(\lambda_1,\ldots,\lambda_d)+\mathbf{z}\cdot\mathbf{z}' &= \begin{pmatrix} \lambda_1+z_1^2 & \cdots & z_1 z_{d-1} & z_1 z_d \\ \vdots & \ddots & \vdots & \vdots \\ z_1 z_{d-1} & \cdots & \lambda_{d-1}+z_{d-1}^2 & z_{d-1}z_d \\ z_1 z_d & \cdots & z_{d-1}z_d & \lambda_d + z_d^2 \end{pmatrix} \\
&= U'\cdot\begin{pmatrix} \tilde\lambda_1+z_1^2 & \cdots & z_1 z_{d-1} & 0 \\ \vdots & \ddots & \vdots & \vdots \\ z_1 z_{d-1} & \cdots & \tilde\lambda_{d-1}+z_{d-1}^2 & 0 \\ 0 & \cdots & 0 & \tilde\lambda_d \end{pmatrix}\cdot U
\end{aligned}$$

where $\tilde{\lambda}_h = \lambda_h + y_{h,d}$, $y_{h,d} \geq 0$, for $h = 1, \ldots, d-1$, and $\tilde{\lambda}_d = \lambda_d + z_d^2 - \sum_{h=1}^{d-1} y_{h,d}$. Lemma 18 implies that $\sum_{h=1}^{d-1} y_h \leq z_d^2$. Thus $\tilde{\lambda}_j \geq \lambda_j$ for $j = 1, \ldots, d$. Hence we may proceed by induction with the matrix

$$\begin{pmatrix} \tilde{\lambda}_1 + z_1^2 & \cdots & z_1 z_{d-1} \\ \vdots & \ddots & \vdots \\ z_1 z_{d-1} & \cdots & \tilde{\lambda}_{d-1} + z_{d-1}^2 \end{pmatrix}.$$

Then (22) follows from Lemma 17 since all elements in the diagonal grow (compared to the original values λ_h) and no element grows by more than 1. $\square$

Lemma 12 follows immediately from Lemma 19. For the proof of Lemma 11 we find that the eigenvalues of

$$\begin{aligned} & Z(t+1) \cdot Z(t+1)' \\ = \;& Z(t) \cdot Z(t)' + \mathbf{z}_{i(t)}(t) \cdot \mathbf{z}_{i(t)}(t)' \\ = \;& U(t)' \cdot \Delta\left(\lambda_1(t), \ldots, \lambda_d(t)\right) \cdot U(t) + U(t) \cdot U(t)' \cdot \mathbf{z}_{i(t)}(t) \cdot \mathbf{z}_{i(t)}(t)' \cdot U(t) \end{aligned}$$

are the eigenvalues of

$$\Delta\left(\lambda_1(t), \ldots, \lambda_d(t)\right) + \tilde{\mathbf{z}}_{i(t)}(t) \cdot \tilde{\mathbf{z}}_{i(t)}(t)' .$$

Using the notation of Lemma 19 let $\lambda_1 \geq \cdots \geq \lambda_d \geq 0$ be the eigenvalues of $Z(t) \cdot Z(t)'$, $\nu_1, \ldots, \nu_d$ the eigenvalues of $Z(t+1) \cdot Z(t+1)'$, and $\mathbf{z} = \tilde{\mathbf{z}}_{i(t)}(t)$. From (13) we have that

$$\left|\left|\mathbf{a}_{i(t)}(t)\right|\right|^2 = \sum_{j:\lambda_j \geq 1} \frac{z_j^2}{\lambda_j} .$$

For bounding z_j^2 we use (18),

$$z_j^2 \leq \nu_j - \lambda_j + \sum_{h=1}^{j-1} y_{h,j} .$$

For $\lambda_h > \lambda_j + 3$ we have by (22) that

$$y_{h,j} \leq \frac{z_j^2 z_h^2}{\lambda_h - \lambda_j - 1} \leq \frac{z_j^2 z_h^2}{2}$$

and

$$\sum_{h:\lambda_h > \lambda_j + 3} y_{h,j} \leq \frac{z_j^2}{2} \sum_{h:\lambda_h > \lambda_j + 3} z_h^2 \leq \frac{z_j^2}{2}$$

since $||\mathbf{z}|| \leq 1$. Hence

$$z_j^2 \leq \nu_j - \lambda_j + \sum_{h=1}^{j-1} y_{h,j} \leq \nu_j - \lambda_j + z_j^2/2 + \sum_{h<j:\lambda_h \leq \lambda_j + 3} y_{h,j}$$

and thus

$$z_j^2 \le 2 \left[\nu_j - \lambda_j + \sum_{h<j:\lambda_h \le \lambda_j+3} y_{h,j} \right] . \tag{23}$$

If $\lambda_j \ge 1$ and $\lambda_h \le \lambda_j + 3$ then $\lambda_j \ge \lambda_h/4$ and we get

$$\sum_{j:\lambda_j \ge 1} \sum_{h<j:\lambda_h \le \lambda_j+3} \frac{y_{h,j}}{\lambda_j} \le 4 \sum_{j:\lambda_j \ge 1} \sum_{h<j:\lambda_h \le \lambda_j+3} \frac{y_{h,j}}{\lambda_h} \le 4 \sum_{h:\lambda_h \ge 1} \sum_{j=h+1}^{d} \frac{y_{h,j}}{\lambda_h} \le 4 \sum_{h:\lambda_h \ge 1} \frac{\nu_h - \lambda_h}{\lambda_h}$$

by (20). Thus

$$\begin{aligned} \left|\left|\mathbf{a}_{i(t)}(t)\right|\right|^2 &= \sum_{j:\lambda_j \ge 1} \frac{z_j^2}{\lambda_j} \\ &\le 2 \sum_{j:\lambda_j \ge 1} \frac{\nu_j - \lambda_j}{\lambda_j} + 2 \sum_{j:\lambda_j \ge 1} \sum_{h<j:\lambda_h \le \lambda_j+3} \frac{y_{h,j}}{\lambda_j} \\ &\le 2 \sum_{j:\lambda_j \ge 1} \frac{\nu_j - \lambda_j}{\lambda_j} + 8 \sum_{h:\lambda_h \ge 1} \frac{\nu_h - \lambda_h}{\lambda_h} \\ &\le 10 \sum_{j:\lambda_j \ge 1} \frac{\nu_j - \lambda_j}{\lambda_j} . \end{aligned}$$

From (23) we also get

$$\begin{aligned} \left|\left|\tilde{\mathbf{v}}_{i(t)}(t)\right|\right|^2 &= \sum_{j:\lambda_j < 1} z_j^2 \\ &\le 2 \sum_{j:\lambda_j < 1} (\nu_j - \lambda_j) + 2 \sum_{j:\lambda_j < 1} \sum_{h<j:\lambda_h \le \lambda_j+3} y_{h,j} \\ &\le 2 \sum_{j:\lambda_j < 1} (\nu_j - \lambda_j) + 2 \sum_{h:\lambda_h < 4} \sum_{j=h+1}^{d} y_{h,j} \\ &\le 2 \sum_{j:\lambda_j < 1} (\nu_j - \lambda_j) + 2 \sum_{h:\lambda_h < 4} (\nu_h - \lambda_h) \\ &\le 4 \sum_{j:\nu_j < 5} (\nu_j - \lambda_j) \end{aligned}$$

by (20).

References

N. Abe and P. M. Long. Associative reinforcement learning using linear probabilistic concepts. In *Proc. 16th International Conf. on Machine Learning*, pages 3–11. Morgan Kaufmann, San Francisco, CA, 1999.

N. Abe and A. Nakamura. Learning to optimally schedule internet banner advertisements. In *Proc. 16th International Conf. on Machine Learning*, pages 12–21. Morgan Kaufmann, San Francisco, CA, 1999.

R. Agrawal. Sample mean based index policies with $O(\log n)$ regret for the multi-armed bandit problem. *Advances in Applied Probability*, 27:1054–1078, 1995.

N. Alon and J. H. Spencer. *The Probabilistic Method*. Wiley, New York, 1992.

P. Auer. Using upper confidence bounds for online learning. In *Proceedings of the 41th Annual Symposium on Foundations of Computer Science*, pages 270–293. IEEE Computer Society, 2000.

P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire. Gambling in a rigged casino: The adversarial multi-armed bandit problem. In *Proceedings of the 36th Annual Symposium on Foundations of Computer Science*, pages 322–331. IEEE Computer Society Press, Los Alamitos, CA, 1995.

P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire. Gambling in a rigged casino: The adversarial multi-armed bandit problem. NeuroCOLT2 Technical Report NC2-TR-1998-025, Royal Holloway, University of London, 1998. Accessible via http at www.neurocolt.org.

P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire. Gambling in a rigged casino: The adversarial multi-armed bandit problem. Journal version, 2000.

P. Auer and M. K. Warmuth. Tracking the best disjunction. *Machine Learning*, 32:127–150, 1998. A preliminary version has appeared in *Proceedings of the 36th Annual Symposium on Foundations of Computer Science*.

K. Azuma. Weighted sums of certain dependent random variables. *Tohoku Math. J.*, 3: 357–367, 1967.

D. A. Berry and B. Fristedt. *Bandit Problems*. Chapman and Hall, 1985.

M. Herbster and M. K. Warmuth. Tracking the best expert. *Machine Learning*, 32:151–178, 1998.

W. Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58:13–30, 1963.

L. P. Kaelbling. Associative reinforcement learning: A generate and test algorithm. *Machine Learning*, 15:299–319, 1994a.

L. P. Kaelbling. Associative reinforcement learning: Functions in k-DNF. *Machine Learning*, 15:279–298, 1994b.

T. L. Lai and H. Robbins. Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics*, 6:4–22, 1985.

N. Littlestone and M. K. Warmuth. The weighted majority algorithm. *Information and Computation*, 108(2):212–261, 1994.

H. Robbins. Some aspects of the sequential design of experiments. *Bulletin American Mathematical Society*, 55:527–535, 1952.

R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction.* MIT Press, Cambridge, MA, 1998.