Ann Math Artif Intell (2011) 61:203–230
DOI 10.1007/s10472-011-9258-6

Multi-armed bandits with episode context

Christopher D. Rosin

Published online: 26 August 2011

Abstract A multi-armed bandit episode consists of n trials, each allowing selection of one of K arms, resulting in payoff from a distribution over [0, 1] associated with that arm. We assume contextual side information is available at the start of the episode. This context enables an arm predictor to identify possible favorable arms, but predictions may be imperfect so that they need to be combined with further exploration during the episode. Our setting is an alternative to classical multi-armed bandits which provide no contextual side information, and is also an alternative to contextual bandits which provide new context each individual trial. Multi-armed bandits with episode context can arise naturally, for example in computer Go where context is used to bias move decisions made by a multi-armed bandit algorithm. The UCB1 algorithm for multi-armed bandits achieves worst-case regret bounded by $O\left(\sqrt{Kn\log(n)}\right)$. We seek to improve this using episode context, particularly in the case where K is large. Using a predictor that places weight $M_i > 0$ on arm i with weights summing to 1, we present the PUCB algorithm which achieves regret $O\left(\frac{1}{M_*}\sqrt{n\log(n)}\right)$ where M_* is the weight on the optimal arm. We illustrate the behavior of PUCB with small simulation experiments, present extensions that provide additional capabilities for PUCB, and describe methods for obtaining suitable predictors for use with PUCB.

Keywords Computational learning theory · Multi-armed bandits · Contextual bandits · UCB · PUCB · Computer Go

Mathematics Subject Classifications (2010) 68Q32 · 68T05

C. D. Rosin (✉)
Parity Computing, Inc., 6160 Lusk Blvd, Suite C205, San Diego, CA 92121, USA
e-mail: c.rosin@paritycomputing.com

1 Introduction

In the stochastic multi-armed bandit problem, fixed but unknown payoff distributions over $[0, 1]$ are associated with each of K arms. The "multi-armed bandit" name comes from envisioning a casino with a choice of K "one-armed bandit" slot machines. In each trial, an agent can pull one of the arms and receive its associated payoff, but does not learn what payoffs it might have received from other arms. Over a sequence of trials, the agent's goal is to mix exploration to learn which arms provide favorable payoffs, and exploitation of the best arms. The agent's goal over n trials is to achieve total payoff close to the total payoff of the best single arm. The difference between the agent's payoff and the best arm's payoff is called the regret [3].

A foundation for the work here is the UCB1 algorithm for stochastic multi-armed bandits [3]. UCB1 maintains empirical average payoff $x(t, i)$ for each arm i, and on trial t pulls the arm maximizing upper confidence bound $x(t, i) + \sqrt{\frac{2\log(t)}{s_i}}$ where s_i is the number of previous pulls of i. The acronym "UCB" comes from "upper confidence bound." This simple algorithm successfully achieves worst-case expected regret upper-bounded by $O\left(\sqrt{Kn\log(n)}\right)$.

UCB1-based algorithms have played a key role in recent progress in software for playing the game of Go, and this provides a motivating example for the theoretical work in this paper. Computer Go has been very challenging [5, 9], but major advances have been obtained using Monte Carlo techniques that evaluate positions using random playouts [7, 19, 20]. An important development has been the efficient combination of Monte Carlo evaluation with tree search. In particular, the UCT algorithm [23] applies UCB1 to choose moves at each node (board position) of the search tree. The bandit arms correspond to legal moves from the position, and payoffs are obtained from Monte Carlo playout results. UCT has been effective for Go [19], and has generally replaced earlier heuristics for Monte Carlo tree search (e.g., [7, 13]) which lacked the theoretical regret bounds behind UCB1. Following success in Go, UCT-based methods have been applied successfully to other domains [16, 18].

There has been ongoing theoretical development of multi-armed bandit algorithms, including issues that may be relevant to Go and Monte Carlo tree search. This includes the "pure exploration" problem relevant at the root of a search tree [8], studies of bandits with large numbers of arms when there is no initial preference among them [31], and fundamental improvements in UCB-style bandit algorithms and regret bounds [1, 2]. But computer Go has progressed further with trial-and-error development of more complex heuristics (e.g. [11, 12]) for which there is little theoretical understanding of the type available for UCB1.

In this paper, we examine one particular issue of importance in computer Go, which is the integration of contextual information to approximately predict good arms at the start of a sequence of multi-armed bandit trials. In Go, while a node's board position in principle would allow a predictor to make a perfect move recommendation, in practice this has been extremely difficult [5]. But it is possible for relatively simple approximate predictors in this domain to make useful initial recommendations, with further bandit-based exploration improving upon this. A predictor's recommendations can be especially useful in multi-armed bandit applications like Go where the number of arms is large. Several heuristic approaches for combining multi-armed bandit algorithms (especially UCB1) with recommendations of a predictor have been developed for Go. These include using the predictor to rank the moves and start exploration with only the best ones while significantly delaying entry of those further down the list [14, 15], and an additive bias to UCB1's payoff estimates [10]. The algorithm presented in this paper modifies UCB1 with a novel form of additive bias, and we show this enables an advantageous regret bound in the theoretically tractable stochastic multi-armed bandit setting.

Our theoretical learning model considers sequences of multi-armed bandit trials called *episodes*, with contextual side information obtained before the first trial and fixed throughout an episode. We call this a *multi-armed bandit problem with episode context.* A predictor uses the context to make an approximate recommendation of which arms are likely to be best. The multiple trials of the episode then provide an opportunity to improve upon the predictor's recommendation. In the computer Go example, the context corresponds to the board position at a node of the search tree, and a predictor performs static analysis of the position to make initial recommendations before the start of bandit trials at the node. But the learning model may describe aspects of other applications as well—for example, in web advertising the content of a webpage may cause a predictor to recommend some ads over others, and then a bandit algorithm can test these recommendations and improve upon them during repeated user visits to the webpage. While applications like Go and web advertising have complex features not captured by our simple learning model, our focus on a theoretically tractable setting allows us to study a core problem in depth and make comparisons to other theoretical work on multi-armed bandits.

The definition of episode context used here is intended as an alternative to the contextual multi-armed bandits which have been studied by others under various names [24, 29]. In that prior work, context was allowed to change at the start of every trial. Trial-specific context could give side information about the payoffs available for that particular trial. While providing a more general setting, effective usage of trial-specific context presents a problem that is more difficult than necessary for modelling applications like bandit-based decisions in computer Go. Regret bounds from that previous theoretical work on contextual multi-armed bandits do not satisfy our technical goals described below. This paper's "episode context" setting contrasts with previous work on contextual multi-armed bandits. Episode context provides side information about the episode's overall payoff distributions, rather than providing more specific information about any one trial's payoffs drawn from the distributions. Episode context informs initial decisions, but then we also have an opportunity to improve upon these decisions over a sequence of trials having the same payoff distributions. By taking advantage of this, we are able to obtain better regret bounds from an efficient UCB-style algorithm.

Goals and outline of paper In the stochastic multi-armed bandit problem, each arm is associated with an unknown payoff distribution that is fixed throughout the episode. Without use of context, worst-case regret is lower-bounded by $\Omega(\sqrt{Kn})$ [4], and the UCB1 algorithm achieves worst-case regret at most $O\left(\sqrt{Kn\log(n)}\right)$ [1, 3, 21, 28]. In our setting, a predictor uses context to assign a vector of arm weights $\boldsymbol{M}$ at the start of the episode, and we seek a regret bound that depends on the weights in a way that improves worst-case regret's dependence on K. In Section 2 we seek a worst-case regret bound of the form $O(f(\boldsymbol{M})g(n))$, with $f(\boldsymbol{M})$ measuring how suitable the predictor is for improving worst-case regret. An advantage of a bound of this form is that the right choice of $\boldsymbol{M}$ does not depend on n. In seeking a bound of this form, we do not want $g(n)$ worse than $\sqrt{n\log(n)}$. That is, we do not want to worsen UCB1's worst-case dependence on episode length, no matter how poor the predictor.

In the style of UCB1, we want to make fast decisions each trial on the basis of accumulated statistics, without requiring a complete record of history for frequent reanalysis. We seek an efficient algorithm that is effective both for short episodes (including $n < K$) and long episodes ($n \gg K$).

The algorithm PUCB in Section 2 meets these goals, yielding a regret bound of the form $O\left(\frac{1}{M_*}\sqrt{n\log(n)}\right)$.

In Section 2.3, we then show simple extensions that give PUCB additional favorable properties possessed by UCB1. The extension in Section 2.3.1 enables PUCB to revert smoothly towards UCB1's worst-case regret bound of the form $O\left(\sqrt{Kn\log(n)}\right)$ as the M_i become uniform (such uniform M_i provide no information about which arm is best). A separate second extension in Section 2.3.2 enables PUCB to achieve (after an initial period) regret that scales logarithmically in n, in the case where the optimal arm is better than the other arms by a sufficiently large margin. This section also gives insight into the the behavior of PUCB for large n.

Then, Section 3 describes methods for obtaining suitable predictors for use with PUCB. Section 3.1 shows that optimizing the regret bound on training episodes can yield a predictor which successfully generalizes to enable a good PUCB regret bound on unseen test episodes. In Section 3.2 we compute the expected regret that results when predictor weights are set based on approximate knowledge of the probability that each arm is optimal. These sections also give insight into the average behavior of PUCB across episodes drawn from a distribution.

Section 4 illustrates the behavior of PUCB with small simulation experiments.

An early version of this work was presented at ISAIM 2010 [26].

2 Multi-armed bandit episodes

2.1 Definitions

The multi-armed bandit problem is an interaction between an agent and an environment. A multi-armed bandit *episode* consists of a sequence of trials. Each episode is associated with a context chosen by the environment from a fixed set Z of possible contexts. A *predictor* maps context $z \in Z$ to a vector of real-valued weights $\boldsymbol{M}$ with $0 < M_i \leq 1$ for each i, and with the restriction $\sum_i M_i = 1$. The predictor is assumed to be available to the agent at the start of the episode. Formally, an episode is a tuple $(K, z, n, \mathbf{D})$ with $\mathbf{D}$ consisting of a payoff distribution D_i over $[0, 1]$ for each arm i with $1 \leq i \leq K$. D_i has mean μ_i, not revealed to the agent. The choice of episodes is controlled by the environment.

An episode $(K, z, n, \mathbf{D})$ proceeds as follows:

1. K and z are revealed to the agent.
2. Each trial t with $1 \leq t \leq n$, the agent pulls arm i_t and receives a payoff chosen independently according to D_{i_t}.
3. Following trial $t = n$ the environment notifies the agent that the episode is over.

Let $*$ be any fixed arm. Given the agent's policy for selecting i_t, the *expected regret* to arm $*$ for the episode is the expected value of the difference between the payoff achieved by pulling $*$ every trial, and the payoff achieved by the agent:

$$R_* = n\mu_* - E\left[\sum_{t=1}^{n} \mu_{i_t}\right]$$

where expectation E is taken over sequences of payoffs and any randomization in the agent's policy.

Throughout, we use log to denote natural logarithm.

2.2 Multi-armed bandit policy PUCB

In an episode in which the payoff distributions with means μ_i are selected by the environment, let $*$ be the optimal arm and set $\Delta_i = \mu_* - \mu_i$ comparing the mean of arm i to that of $*$. The original UCB1 policy [3] achieves expected regret at most:

$$R_* \le \left(8 \sum_{i:\mu_i<\mu_*} \frac{\log(n)}{\Delta_i}\right) + \left(1 + \frac{\pi^2}{3}\right)\left(\sum_{i=1}^{K} \Delta_i\right) \tag{1}$$

In addition to the bound expressed above, regret is $\Delta_i \le 1$ per trial in which arm i is pulled, and this may be used to obtain an additional upper bound of at most n for the episode. Discussions of UCB1 often focus on the case of constant Δ_i and the logarithmic dependence of regret on n as n increases. However, with an adversarial environment selecting worst-case payoff distributions with knowledge of horizon n, Δ_i that is $\Theta\left(\sqrt{K\log(n)/n}\right)$ yields an expected regret bound of $O\left(\sqrt{Kn\log(n)}\right)$ [1, 21, 28]. We focus here on this worst-case bound.

Note that if all Δ_i are a larger $\Theta\left(K\sqrt{\log(n)/n}\right)$, then the problem becomes easier and UCB1 obtains an expected regret bound of $O(\sqrt{n\log(n)})$.

We now present our new algorithm PUCB ("Predictor + UCB"—see Fig. 1), which is a modification of UCB1. PUCB uses additive penalties proportional to $\frac{1}{M_i}$; it seeks to overcome worst-case Δ_i by placing substantial additive penalties on arms that have low weight. For example, if two arms each have weight $\frac{1}{4}$ but all remaining arms have weight $\frac{1}{2K}$ then PUCB will place penalties proportional to $K\sqrt{\log(n)/n}$ on these latter arms. Indeed, since the sum of the weights is 1 and average weight is $\frac{1}{K}$,

On trial t pull arm $i_t = \text{argmax}_i\ (x(t,i) + c(t,s_i) - m(t,i))$ where:

- s_i is the number of previous pulls on i.
- $x(t,i)$ is i's empirical average payoff at the start of trial t if $s_i > 0$, otherwise 1.
- $c(t,s) = \sqrt{\frac{3\log(t)}{2s}}$ if $s > 0$, otherwise 0. (confidence bound term)
- $m(t,i) = \frac{2}{M_i}\sqrt{\frac{\log(t)}{t}}$ if $t > 1$, otherwise $\frac{2}{M_i}$. (predictor-based penalty term)

Fig. 1 Multi-armed bandit policy PUCB

it must be the case that most arms receive a large penalty—so if one of the few arms favored by the weights is optimal, we will show that regret is small. But even arms with low weight will be explored eventually, and the penalty on such arms will be overcome more quickly if they are found to obtain high average payoff.

This ability, to give simultaneous consideration to the values of both predictor weight and average payoff, helps motivate the choice of an additive penalty term in PUCB's design. This contrasts with possible alternative approaches like the "progressive widening" heuristic [14, 15] which uses a predictor to delay the entry of low-weight arms, but then gives those arms full consideration immediately upon entry. In that approach, using only short delays for low-weight arms may not obtain sufficient advantage when the predictor's preferences are correct. But a long delay for low-weight arms can suffer very large regret if the low-weight arms turn out to have much higher payoff than the other available arms. The additive penalty in PUCB allows a more gradual exploration of low-weight arms, moving to exploit them more quickly if they do indeed have much higher payoff. This feature is important in the proof of the regret bound below.

The additive predictor-based term decays as $\sqrt{\log(t)/t}$. A substantially slower decay would lack sufficient sensitivity to large differences in average payoffs, and could suffer large regret for that reason. A substantially faster decay (e.g., [10]) could allow the confidence bound term and stochastic variations in average payoff to swamp out important differences in the predictor term, and thereby lose some of the potential advantage of the predictor. The choice of $\sqrt{\log(t)/t}$ decay is also important in the proof of the regret bound below.

PUCB carefully handles the case where arm i has not yet been pulled, to achieve a result that holds for small $n < K$ as well as larger values of n; this is different from UCB1 which starts with one pull on each of the K arms.

PUCB also differs slightly from the original UCB1 in using a $\frac{3}{2}$ constant in $c(t, s)$ whereas UCB1 used 2; other authors have discussed bounds for UCB-like algorithms using a range of values for this constant [2].

Note PUCB is deterministic, so the expected value of regret depends only on the random sequence of payoffs and not on any randomization by the agent.

Theorem 1 *Given weights $M_i > 0$ with $\sum_i M_i = 1$, compared to any fixed arm $*$ PUCB achieves expected regret for the episode of $R_* \leq 17\frac{1}{M_*}\sqrt{n\log(n)}$ for $n > 1$.*

Usually $*$ would be chosen to be the optimal arm maximizing μ_*, but Section 3.1 uses near-optimal $*$ as well.

Outline of proof Regret is incurred when the agent selects an arm i that is worse than $*$. The proof analyzes the causes of selecting such an arm i:

- **Arm *i*'s first pull:** The number of distinct arms pulled at least once is bounded in Corollary 1.
- **Empirical average payoffs inaccurate by more than confidence bound:** this is rare (via Hoeffding's inequality), and the associated regret is bounded in R^{tail}.
- The remaining cases have empirical average payoffs which are close to accurate.
    - **M_i not substantially lower than M_*:** this is limited by $\sum_i M_i$ and is eventually overcome by decreasing penalty and $*$'s higher average payoff. The

associated regret is bounded in $R^{\text{close}}_{(a)}$. Note that, at least for $n \gg K$, this is the main source of regret in the proof.

- **M_i substantially lower than M_* but i selected anyway:** the $m(t,i)$ penalty and decreases in i's confidence bound term limit this. The associated regret is bounded in $R^{\text{close}}_{(b)}$.

We will present some notation, prove an initial lemma and corollary, and then proceed to the main proof. The proof is structured so that parts of it can be reused in analyzing the PUCB extensions in Section 2.3.

Notation Let $s_{t,i}$ denote the value of s_i at the start of trial t, and extend this so that if $t > n$ then $s_{t,i}$ is the final total number of pulls on i for the episode. Let $X(i,s)$ denote the value of empirical average payoff of arm i after s previous pulls on i. For s greater than the total number of pulls on i during the episode, extend with additional independently drawn samples from the distribution associated with arm i and include these in $X(i,s)$. That is, $X(i,s)$ continues to be the empirical average of s independently drawn samples from the distribution associated with arm i even for large s; this simplifies the analysis below. Note that $X(i,s_{t,i})$ is the value of $x(t,i)$ at the start of trial t (and is 1 if $s_{t,i}=0$). Let $V_{t,i} = X(i,s_{t,i}) + c(t,s_{t,i}) - m(t,i)$; an arm with maximal $V_{t,i}$ is pulled on trial t.

For any condition Q, let notation $\{Q\}$ indicate 1 if condition Q is true and 0 otherwise.

Arms are numbered $1 \le i \le K$. Note that $\sum_i$ is used to indicate a sum over all arms i.

The following lemma and corollary will help handle relatively small n. We want to show that low M_i arms are not pulled, during a sufficiently short episode.

Lemma 1 *Arms i satisfying*

$$M_i \le \frac{M_*}{1.61\sqrt{n}}$$

cannot be pulled during the episode.

Proof of Lemma 1 Before i's first pull, assuming $t > 1$:

$$V_{t,i} = 1 - 2\frac{1}{M_i}\sqrt{\frac{\log(t)}{t}}$$

By the assumption on M_i:

$$\begin{aligned} V_{t,i} &\le 1 - (2)(1.61)\frac{1}{M_*}\sqrt{\frac{n\log(t)}{t}} \\ &\le 1 - (3.22)\frac{1}{M_*}\sqrt{\log(t)} = \left(1 - 1.22\frac{1}{M_*}\sqrt{\log(t)}\right) - 2\frac{1}{M_*}\sqrt{\log(t)} \end{aligned}$$

For $t > 1$, and because $\frac{1}{M_*} \ge 1$, $1.22\frac{1}{M_*}\sqrt{\log(t)} \ge 1$, so:

$$V_{t,i} \le -2\frac{1}{M_*}\sqrt{\log(t)} \le -m(t,*) < V_{t,*}$$

whether or not $*$ has been pulled yet at trial t. For $t = 1$, $V_{t,i} < V_{t,*}$ as well because $\frac{1}{M_i} > \frac{1}{M_*}$ by the assumption on M_i. Therefore arm i cannot be pulled for any t during the episode. □

Corollary 1 *Fewer than $\frac{1.61\sqrt{n}}{M_*}$ distinct arms are pulled during the episode.*

Proof of Corollary 1 By Lemma 1, only arms satisfying:

$$M_i > \frac{M_*}{1.61\sqrt{n}}$$

can be pulled during the episode. Since $\sum_i M_i = 1$, the number of distinct arms satisfying this condition is less than $\frac{1.61\sqrt{n}}{M_*}$. □

Proof of Theorem 1 We will assume below that $n \geq 4$; for $1 < n < 4$ the bound in Theorem 1 is clearly true because $\frac{1}{M_*} \geq 1$ and regret is at most 1 per trial.

Consider only arms i with $\mu_i < \mu_*$, since pulls of other arms with $\mu_i \geq \mu_*$ do not incur any positive regret. For each arm i with $\mu_i < \mu_*$ and with i pulled at least once during the episode, we will bound its total number of pulls. Parts of this follow previous UCB1 analyses [2, 3].

Say arm i has first pull $t = F_i$. Counting subsequent pulls

$$s_{n+1,i} = 1 + \sum_{t=F_i+1}^{n} \{i_t = i\}$$

In the sum, a pull $i_t = i$ requires that:

$$X(*, s_{t,*}) + c(t, s_{t,*}) - m(t, *) \leq X(i, s_{t,i}) + c(t, s_{t,i}) - m(t, i)$$

For this to be true, at least one of the following must hold:

$$X(*, s_{t,*}) + c(t, s_{t,*}) < \mu_* \tag{2}$$

$$X(i, s_{t,i}) - c(t, s_{t,i}) > \mu_i \tag{3}$$

$$\mu_* - m(t, *) \leq \mu_i + 2c(t, s_{t,i}) - m(t, i) \tag{4}$$

Note (2) is false for $s_{t,*} = 0$ since $X(*, 0) = 1$ by definition. And in (3) we only consider pulls beyond the first ($s_{t,i} \geq 1$) since $t > F_i$. So we only need consider (2) for pulls of $*$ beyond its first, and (3) for pulls of i beyond its first.

Let $s^{\text{tail}}_{n+1,i}$ denote the number of pulls of i with $F_i + 1 \leq t \leq n$ and for which (2) or (3) is satisfied, and let $s^{\text{close}}_{n+1,i}$ denote the number of pulls of i with t in this interval and (4) satisfied. So total pulls:

$$s_{n+1,i} \leq 1 + s^{\text{tail}}_{n+1,i} + s^{\text{close}}_{n+1,i}$$

Define $A = \{i : \mu_i < \mu_* \text{ and } s_{n+1,i} > 0\}$ as the set of arms that are worse than $*$ and are pulled at least once. Define $\Delta_i = \mu_* - \mu_i$. From the definition of R_*:

$$R_* \leq \sum_{i \in A} \Delta_i E[s_{n+1,i}]$$

with expectation E taken over random sequences of payoffs.

$$R_* \leq \sum_{i \in A} \Delta_i \left(1 + E\left[s^{\text{tail}}_{n+1,i}\right] + E\left[s^{\text{close}}_{n+1,i}\right]\right)$$

Define:

$$R^{\text{tail}} = \sum_{i \in A} \Delta_i E\left[s^{\text{tail}}_{n+1,i}\right]$$

$$R^{\text{close}} = \sum_{i \in A} \Delta_i E\left[s^{\text{close}}_{n+1,i}\right]$$

Now, using the fact that $\Delta_i \leq 1$:

$$R_* \leq |A| + R^{\text{tail}} + R^{\text{close}} \tag{5}$$

Since Corollary 1 gives $|A| \leq \frac{1.61\sqrt{n}}{M_*}$:

$$R_* \leq \frac{1.61\sqrt{n}}{M_*} + R^{\text{tail}} + R^{\text{close}} \tag{6}$$

We will analyze R^{tail} and R^{close} separately.

Regret R^{tail}

$$\begin{aligned} s^{\text{tail}}_{n+1,i} \leq & \sum_{t=F_i+1}^{n} \{X(*, s_{t,*}) + c(t, s_{t,*}) < \mu_*\} \\ & + \sum_{t=F_i+1}^{n} \{X(i, s_{t,i}) - c(t, s_{t,i}) > \mu_i\} \\ \leq & \sum_{t=F_i+1}^{n} \{\exists s_* \leq t \text{ s.t. } X(*, s_*) + c(t, s_*) < \mu_*\} \\ & + \sum_{t=F_i+1}^{n} \{\exists s_i \leq t \text{ s.t. } X(i, s_i) - c(t, s_i) > \mu_i\} \\ \leq & \sum_{t=F_i+1}^{n} \sum_{s_*=1}^{t} \{X(*, s_*) + c(t, s_*) < \mu_*\} \\ & + \sum_{t=F_i+1}^{n} \sum_{s_i=1}^{t} \{X(i, s_i) - c(t, s_i) > \mu_i\} \end{aligned}$$

Using E to denote expected value over sequences of payoffs:

$$\begin{aligned} E\left[s^{\text{tail}}_{n+1,i}\right] \leq & \sum_{t=F_i+1}^{n} \sum_{s_*=1}^{t} Pr\{X(*, s_*) + c(t, s_*) < \mu_*\} \\ & + \sum_{t=F_i+1}^{n} \sum_{s_i=1}^{t} Pr\{X(i, s_i) - c(t, s_i) > \mu_i\} \end{aligned}$$

Bound the probabilities using Hoeffding's inequality: $sX(j, s)$ is the sum of s random variables in [0, 1], and has expected value $s\mu_j$. Applying Hoeffding's inequality:

$$Pr\{X(*, s_*) < \mu_* - c(t, s_*)\} \leq e^{-3\log(t)} = t^{-3}$$

$$Pr\{X(i, s_i) > \mu_i + c(t, s_i)\} \leq e^{-3\log(t)} = t^{-3}$$

$$E\left[s^{\text{tail}}_{n+1,i}\right] \leq \left(\sum_{t=F_i+1}^{n}\sum_{s_*=1}^{t} t^{-3}\right) + \left(\sum_{t=F_i+1}^{n}\sum_{s_i=1}^{t} t^{-3}\right) \leq 2\sum_{t=F_i+1}^{n} t^{-2} < 2\sum_{t=1}^{\infty} t^{-2} = \frac{\pi^2}{3}$$

In the worst case, regret for these pulls is at most 1 per trial. Recall that A is the set of arms worse than $*$ and with at least one pull; summing $E[s^{\text{tail}}_{n+1,i}]$ for $i \in A$ gives:

$$R^{\text{tail}} \leq \frac{\pi^2}{3}|A| \tag{7}$$

By Corollary 1, $|A| < \frac{1.61\sqrt{n}}{M_*}$, so:

$$R^{\text{tail}} \leq \frac{\pi^2}{3}\frac{1.61\sqrt{n}}{M_*} < \frac{5.3\sqrt{n}}{M_*} \tag{8}$$

Regret R^{close} Assign each arm one of two types (a) and (b) as defined below, giving $R^{\text{close}} = R^{\text{close}}_{(a)} + R^{\text{close}}_{(b)}$. We will bound $R^{\text{close}}_{(a)}$ and $R^{\text{close}}_{(b)}$ separately.

Type (a) arms Type (a) arms are defined to be those satisfying:

$$m(n, i) - m(n, *) < \frac{m(n, i)}{2}$$

Let $A_{(a)}$ denote the indices i of type (a) arms. At worst, in every trial one of the type (a) arms is pulled with (4) being satisfied:

$$\sum_{\{i\in A_{(a)}\}} s^{\text{close}}_{n+1,i} \leq n$$

Because these pulls satisfy (4), for each pull:

$$\mu_* - m(t, *) \leq \mu_i + 2c(t, s_i) - m(t, i)$$

$$\mu_* - \mu_i \leq 2c(t, s_i) + m(t, *)$$

So, the regret associated with $s^{\text{close}}_{n+1,i}$ summed over all type (a) arms is at most $\sum_{t=1}^{n}(2c(t, s_{i_t}) + m(t, *))$ for some choice of sequence of i_t with all $i_t \in A_{(a)}$.

$$R^{\text{close}}_{(a)} \leq \sum_{t=1}^{n}(2c(t, s_{i_t}) + m(t, *)) \leq \left(\sum_{\{i\in A_{(a)}\}}\sum_{s_i=1}^{s^{\text{close}}_{n+1,i}} 2c(n, s_i)\right) + \left(\sum_{t=1}^{n} m(t, *)\right)$$

$$\leq \left(\sum_{\{i\in A_{(a)}\}}\left(2\sqrt{\frac{3}{2}\log(n)}\right)\sum_{s_i=1}^{s^{\text{close}}_{n+1,i}}\frac{1}{\sqrt{s_i}}\right) + \left(\left(2\frac{1}{M_*}\sqrt{\log(n)}\right)\sum_{t=1}^{n}\frac{1}{\sqrt{t}}\right)$$

Note that the last line requires $m(t,*) \le \frac{2}{M_*}\sqrt{\log(n)/t}$, which is valid given our constraint that $n \ge 4$.

Applying inequality $\sum_{j=1}^{k}(1/\sqrt{j}) < 2\sqrt{k}$ to both terms:

$$R_{(a)}^{\text{close}} \le \left(\sum_{\{i\in A_{(a)}\}} 4\sqrt{\frac{3}{2}\log(n)s_{n+1,i}^{\text{close}}}\right) + 4\frac{1}{M_*}\sqrt{n\log(n)}$$

with:

$$\sum_{\{i\in A_{(a)}\}} s_{n+1,i}^{\text{close}} \le n$$

Now, since Jensen's inequality gives $\frac{1}{|S|}\sum_{i\in S}\sqrt{s_i} \le \sqrt{\frac{1}{|S|}\sum_{i\in S}s_i}$:

$$\sum_{\{i\in A_{(a)}\}} 4\sqrt{\frac{3}{2}\log(n)s_{n+1,i}^{\text{close}}} \le 4\sqrt{\frac{3}{2}\log(n)}\sqrt{|A_{(a)}|}\sqrt{\sum_{\{i\in A_{(a)}\}} s_{n+1,i}^{\text{close}}}$$

$$\le \sqrt{24|A_{(a)}|n\log(n)}$$

So:

$$R_{(a)}^{\text{close}} \le \sqrt{n\log(n)}\left(\sqrt{24|A_{(a)}|} + \frac{4}{M_*}\right) \tag{9}$$

Because $m(n,i) - m(n,*) < \frac{m(n,i)}{2}$ for type (a) arms, $m(n,i) < 2m(n,*)$ and so:

$$M_i > \frac{1}{2}M_* \tag{10}$$

Since $\sum_i M_i = 1$ the maximum number of such arms is at most $2\frac{1}{M_*}$. This is an upper bound on $|A_{(a)}|$, giving:

$$R_{(a)}^{\text{close}} \le \sqrt{n\log(n)}\left(\sqrt{\frac{48}{M_*}} + \frac{4}{M_*}\right) < \frac{11\sqrt{n\log(n)}}{M_*} \tag{11}$$

since $\sqrt{\frac{1}{M_*}} \le \frac{1}{M_*}$ for $M_* \le 1$.

Type (b) arms Type (b) arms have $m(n,i) - m(n,*) \ge \frac{m(n,i)}{2}$. Given our constraint that $n \ge 4$, we have $m(t,i) - m(t,*) \ge m(n,i) - m(n,*)$ and so $m(t,i) - m(t,*) \ge \frac{m(n,i)}{2}$. Assume there is a pull on type (b) arm i at time t with (4) satisfied and

$$s_{t,i} > \frac{6\log(n)}{(\mu_* - \mu_i + (m(n,i)/2))^2} \tag{12}$$

giving:

$$\begin{aligned}2c(t, s_{t,i}) &< 2\sqrt{\frac{\frac{3}{2}\log(t)(\mu_* - \mu_i + (m(n,i)/2))^2}{6\log(n)}}\\ 2c(t, s_{t,i}) &< \mu_* - \mu_i + \frac{m(n,i)}{2}\\ 2c(t, s_{t,i}) &< \mu_* - \mu_i + m(t,i) - m(t,*)\\ \mu_* - m(t,*) &> \mu_i + 2c(t, s_{t,i}) - m(t,i)\end{aligned}$$

and so (4) is false. So, there cannot be any pulls on i with $s_{t,i}$ this large and (4) true. If t' is the time of the final pull on i with (4) true, then we have shown:

$$s_{t'+1,i} \leq 1 + \frac{6\log(n)}{(\mu_* - \mu_i + (m(n,i)/2))^2}$$

Of these pulls, $s^{\text{close}}_{n+1,i}$ by definition excludes the first pull, so

$$s^{\text{close}}_{n+1,i} \leq s_{t'+1,i} - 1 \leq \frac{6\log(n)}{\left(\mu_* - \mu_i + \frac{m(n,i)}{2}\right)^2}$$

Let $\Delta_i = \mu_* - \mu_i$. Regret associated with $s^{\text{close}}_{n+1,i}$ for type (b) arm i is at most:

$$\begin{aligned}\Delta_i\left(\frac{6\log(n)}{\left(\Delta_i + \frac{m(n,i)}{2}\right)^2}\right) &= \frac{6\log(n)}{\left(\Delta_i + \frac{m(n,i)}{2}\right)\left(1 + \frac{m(n,i)}{2\Delta_i}\right)}\\ &= \frac{6\log(n)}{\Delta_i + 2\frac{m(n,i)}{2} + \frac{m(n,i)^2}{4\Delta_i}}\\ &\leq \frac{6\log(n)}{m(n,i)} = 3M_i\sqrt{n\log(n)}\end{aligned}$$

Sum this over all arms i to upper-bound regret associated with $s^{\text{close}}_{n+1,i}$ across all type (b) arms:

$$R^{\text{close}}_{(b)} \leq \sum_i 3M_i\sqrt{n\log(n)} \tag{13}$$

and using $\sum_i M_i = 1$:

$$R^{\text{close}}_{(b)} \leq 3\sqrt{n\log(n)} \tag{14}$$

Total regret Using $R^{\text{close}} = R^{\text{close}}_{(a)} + R^{\text{close}}_{(b)}$ and substituting (8), (11) and (14) into (6):

$$\begin{aligned}R_* &\leq \frac{11}{M_*}\sqrt{n\log(n)} + 3\sqrt{n\log(n)} + \frac{5.3\sqrt{n}}{M_*} + \frac{1.61\sqrt{n}}{M_*}\\ &\leq 14\sqrt{n\log(n)}\frac{1}{M_*} + \frac{7\sqrt{n}}{M_*}\end{aligned}$$

For $n > 500$, $\sqrt{\log(n)} > \frac{7}{3}$ and so taking $n > 500$:

$$R_* \leq 14\sqrt{n\log(n)}\frac{1}{M_*} + 3\frac{\sqrt{n\log(n)}}{M_*} \leq 17\sqrt{n\log(n)}\frac{1}{M_*}$$

Regret is at most 1 per trial, and so for $n \leq 500$ this bound holds trivially since in that case $17\sqrt{n\log(n)} \geq n$. So for all $n > 1$ we have proven the theorem. □

The constant isn't tight; we used loose bounds to simplify. For example, the bound in Corollary 1 is loose for $n \gg K^2$. If we have good lower bounds on $\frac{1}{M_*}$ or n, the constant in $m(t, i)$ as well as the proof itself can be used to improve the constant.

2.3 Additional capabilities for PUCB

We discuss two modifications that separately give PUCB additional favorable properties possessed by UCB1. The modifications affect only the choice of weights M_i and the analysis; PUCB is itself unchanged.

2.3.1 Recovering UCB1's Worst-case bound

Given a set of predictor weights, Theorem 1 cannot simultaneously match UCB1's worst-case regret $O\left(\sqrt{Kn\log(n)}\right)$ for all possible choices of the identity of $*$. If the identity of $*$ is aligned with the predictor's recommendations, then Theorem 1 gives a good bound, but if $*$ is chosen to be an arm given low weight by the predictor then the bound is worse than UCB1's worst-case regret. Uniform M_i yields $O\left(K\sqrt{n\log(n)}\right)$. Here, we summarize a variant of PUCB that can recover UCB1's worst-case regret (up to a constant).

Run PUCB with a predictor with weights that needn't satisfy $\sum_i M_i = 1$: let $Z = \sum_{i=1}^{K} M_i$ for the now-variable total. We will refer to this variation as "Unrestricted PUCB."

Theorem 2 *With weights $0 < M_i \leq 1$, compared to any fixed arm $*$, for $n > 1$ expected regret for the episode satisfies:*

$$R_* \leq 4\sqrt{n\log(n)}\left(\sqrt{Z} + \sqrt{\frac{1}{M_*}}\right)^2 + \min\left(5K, 5n, \frac{7Z\sqrt{n}}{M_*}\right)$$

Proof The proof closely follows that of Theorem 1, and definitions from there are used here as well. As in the proof of Theorem 1, we assume that $n \geq 4$. For $1 < n < 4$ the bound in Theorem 2 is clearly true because regret is at most 1 per trial, and the second term in the bound is at least 5.

Lemma 1 did not depend on $\sum_i M_i$ and so still holds here; only arms satisfying $M_i > \frac{M_*}{1.61\sqrt{n}}$ can be pulled during the episode. Here, $\sum_i M_i = Z$, and so the number of distinct arms satisfying this condition is less than $\frac{1.61Z\sqrt{n}}{M_*}$. In addition, the number of distinct arms pulled is at most K, and is at most n as well. Therefore, we upper bound the number of distinct arms pulled by:

$$\min\left(K, n, \frac{1.61Z\sqrt{n}}{M_*}\right) \tag{15}$$

The proof of (5) did not depend on $\sum_i M_i$ and so still holds here, and now $|A|$ is upper-bounded by (15) so that:

$$R_* \leq \min\left(K, n, \frac{1.61Z\sqrt{n}}{M_*}\right) + R^{\text{tail}} + R^{\text{close}} \tag{16}$$

For R^{tail}, the proof of (7) did not depend on $\sum_i M_i$ and so still holds here. Again using (15) to upper-bound $|A|$:

$$R^{\text{tail}} \leq \frac{\pi^2}{3}\min\left(K, n, \frac{1.61Z\sqrt{n}}{M_*}\right) \tag{17}$$

For $R^{\text{close}}_{(a)}$, the proof of (9) and (10) did not depend on $\sum_i M_i$ and so still hold. Here $\sum_i M_i = Z$ so (10) gives an upper bound on $|A_{(a)}|$ of $2\frac{Z}{M_*}$. Applying this to (9) gives:

$$R^{\text{close}}_{(a)} \leq \sqrt{n\log(n)}\left(4\sqrt{\frac{3Z}{M_*}} + \frac{4}{M_*}\right) \tag{18}$$

For $R^{\text{close}}_{(b)}$, the proof of (13) did not depend on $\sum_i M_i$ and so still holds here. Using $\sum_i M_i = Z$ gives:

$$R^{\text{close}}_{(b)} \leq 3Z\sqrt{n\log(n)} \tag{19}$$

Finally, using $R^{\text{close}} = R^{\text{close}}_{(a)} + R^{\text{close}}_{(b)}$ and substituting (17), (18) and (19) into (16):

$$\begin{aligned}
R_* &\leq \left(1 + \frac{\pi^2}{3}\right)\min\left(K, n, \frac{1.61Z\sqrt{n}}{M_*}\right) + \sqrt{n\log(n)}\left(4\sqrt{\frac{3Z}{M_*}} + \frac{4}{M_*} + 3Z\right) \\
&\leq \min\left(5K, 5n, \frac{7Z\sqrt{n}}{M_*}\right) + 4\sqrt{n\log(n)}\left(2\sqrt{\frac{Z}{M_*}} + \frac{1}{M_*} + Z\right) \\
&\leq \min\left(5K, 5n, \frac{7Z\sqrt{n}}{M_*}\right) + 4\sqrt{n\log(n)}\left(\sqrt{Z} + \sqrt{\frac{1}{M_*}}\right)^2
\end{aligned}$$

□

As an example using unrestricted PUCB, with uniform $M_i = \frac{1}{\sqrt{K}}$ for all i, $\frac{1}{M_*} = Z = \sqrt{K}$. Noting that $\min(5K, 5n) \leq 5\sqrt{Kn}$, Theorem 2 bounds regret by $O\left(\sqrt{Kn\log(n)}\right)$ independent of the choice of $*$. This is comparable to the worst-case bound for UCB1.

Even for nonuniform predictor weights, the bound on Unrestricted PUCB can provide a meaningful alternative to Theorem 1. The following corollary applies to large n for which the theorem can be simplified. The particular constant is chosen to enable the use of the corollary in a later comparison (in Theorem 7).

Corollary 2 *With $n > 400K^2$ and weights $0 < M_i \leq 1$, compared to any fixed arm $*$, expected episode regret satisfies $R_* \leq 4.25\sqrt{n\log(n)}\left(\sqrt{Z} + \sqrt{1/M_*}\right)^2$.*

Proof Under conditions here we have:

$$5K \leq 5\sqrt{n/400} \leq 0.25\sqrt{n\log(n)} \leq 0.25\sqrt{n\log(n)}\left(\sqrt{Z} + \sqrt{1/M_*}\right)^2$$

which lets us simplify the bound from Theorem 2 to yield the corollary. □

2.3.2 Large Δ_i and long episodes

Returning to the original PUCB with $\sum_i M_i = 1$, we consider a different issue. Define $\Delta_i = \mu_* - \mu_i$, choosing $*$ to be the optimal arm. For suitably large episode length n, and with all Δ_i sufficiently large for suboptimal arms i, the original UCB1 regret bound (1) can be much better (logarithmic in n) than PUCB's bound in Theorem 1 which assumes worst-case Δ_i. We show that PUCB can also obey an improved regret bound for sufficiently large n in the case with all Δ_i sufficiently large.

First, given $M_i > 0$ with $\sum_i M_i = 1$, set $M'_i = \frac{2}{3}M_i + \frac{1}{3K}$ and use these M'_i instead with PUCB. Note $\sum_i M'_i = 1$. Now, $M'_i > \frac{1}{3K}$ for all i. For any arm $*$ we have $\frac{1}{M'_*} < \frac{3}{2}\frac{1}{M_*}$, so the original PUCB bound in Theorem 1 still applies (with regret bound worsened by constant factor $\frac{3}{2}$). Here we apply a relatively simple analysis to show an additional bound focused on improved scaling of regret for sufficiently large n. Note that the resulting bound does not depend on $\boldsymbol{M}$; this theorem also helps illustrate the behavior of PUCB for large n when the predictor can become irrelevant and UCB1-like behavior takes over.

Theorem 3 *Let $*$ denote the optimal arm. Use weights $M'_i > \frac{1}{3K}$. Now, on episodes which satisfy, for some $n_0 > 2$, $\Delta_i \geq 48K\sqrt{\log(n_0)/n_0}$ for all i with $\mu_i < \mu_*$, PUCB achieves expected regret for the episode:*

$$R_* \leq n_0 + \left(8\sum_{i:\mu_i<\mu_*}\frac{\log(n)}{\Delta_i}\right) + 5K$$

The main idea is that the large Δ_i overwhelm the penalty $m(t, *)$ once $t > n_0$, even if the predictor is completely wrong (that is, even if M'_* is as small as allowed here).

Proof We are reanalyzing PUCB, and the proof follows that of Theorem 1. Definitions from there are used here as well. But here we will not split the analysis of R^{close} into type (a) and (b) arms. As before we assume $n \geq 4$ below, noting that the bound of Theorem 3 holds trivially for $1 < n < 4$.

We combine (5) with the observation that $|A| \leq K$ to obtain:

$$R_* \leq K + R^{\text{tail}} + R^{\text{close}} \tag{20}$$

Combining (7) with $|A| \leq K$ gives:

$$R^{\text{tail}} \leq \frac{\pi^2}{3} K < 4K \tag{21}$$

Now, we analyze R^{close} for all suboptimal arms by adapting Theorem 1's analysis for type (b) arms. Assume there is a pull on suboptimal arm i (beyond its first pull) at time $t > n_0$ with (4) satisfied and

$$s_{t,i} > \frac{6 \log(n)}{\left(\mu_* - \mu_i - 6K\sqrt{\log(n_0)/n_0}\right)^2} \tag{22}$$

giving:

$$2c(t, s_{t,i}) < 2\sqrt{\frac{\frac{3}{2}\log(t)\left(\mu_* - \mu_i - 6K\sqrt{\log(n_0)/n_0}\right)^2}{6\log(n)}}$$

$$2c(t, s_{t,i}) < \mu_* - \mu_i - 6K\sqrt{\log(n_0)/n_0}$$

Note that, by the constraint we placed on M'_*:

$$\begin{aligned} m(t,i) - m(t,*) &\geq (1-3K)\left(2\sqrt{\log(t)/t}\right) \\ &\geq -6K\sqrt{\log(t)/t} \end{aligned}$$

And for $t > n_0$, given that $n_0 > 2$, we have:

$$m(t,i) - m(t,*) \geq -6K\sqrt{\log(n_0)/n_0}$$

So we have:

$$2c(t, s_{t,i}) < \mu_* - \mu_i + m(t,i) - m(t,*)$$

$$\mu_* - m(t,*) > \mu_i + 2c(t, s_{t,i}) - m(t,i)$$

and so (4) is false. So, there cannot be any pull on suboptimal arm i (beyond its first pull) at time $t > n_0$ with (4) and (22) satisfied.

Now analyze total $s^{\text{close}}_{n+1,i}$ in two cases depending on the time t' of the final pull on i with (4) true. For the first case: if $t' > n_0$, then we have shown:

$$s_{t'+1,i} \leq 1 + \frac{6\log(n)}{\left(\mu_* - \mu_i - 6K\sqrt{\log(n_0)/n_0}\right)^2}$$

Of these pulls, $s^{\text{close}}_{n+1,i}$ by definition excludes the first pull, so

$$s^{\text{close}}_{n+1,i} \leq s_{t'+1,i} - 1 \leq \frac{6\log(n)}{\left(\mu_* - \mu_i - 6K\sqrt{\log(n_0)/n_0}\right)^2}$$

For the second case: if $t' \leq n_0$ then we simply use:

$$s_{n+1,i}^{\text{close}} \leq s_{n_0,i}$$

No matter which of these two cases applies, the following holds:

$$s_{n+1,i}^{\text{close}} \leq s_{n_0,i} + \frac{6\log(n)}{\left(\mu_* - \mu_i - 6K\sqrt{\log(n_0)/n_0}\right)^2}$$

Using notation $\Delta_i = \mu_* - \mu_i$, regret associated with $s_{n+1,i}^{\text{close}}$ for arm i is at most:

$$\begin{aligned} \Delta_i s_{n_0,i} + \Delta_i \left(\frac{6\log(n)}{\left(\Delta_i - 6K\sqrt{\log(n_0)/n_0}\right)^2} \right) &\leq s_{n_0,i} + \Delta_i \left(\frac{6\log(n)}{\left(\Delta_i - \frac{1}{8}\Delta_i\right)^2} \right) \\ &\leq s_{n_0,i} + \frac{8\log(n)}{\Delta_i} \end{aligned}$$

where we have used $\Delta_i \leq 1$ in the first term and the original constraint that $\Delta_i \geq 48\ K\ \sqrt{\log(n_0)/n_0}$ in the second term. Sum this over all suboptimal arms i to upper-bound regret associated with $s_{n+1,i}^{\text{close}}$ across all suboptimal arms:

$$R^{\text{close}} \leq \sum_{i:\mu_i<\mu_*} \left(s_{n_0,i} + \frac{8\log(n)}{\Delta_i} \right) \leq n_0 + \sum_{i:\mu_i<\mu_*} \frac{8\log(n)}{\Delta_i} \tag{23}$$

Finally, substituting (21) and (23) into (20):

$$R_* \leq K + 4K + n_0 + \sum_{i:\mu_i<\mu_*} \frac{8\log(n)}{\Delta_i}$$

which yields the bound given in the theorem. □

3 Obtaining predictors for use with PUCB

We describe two approaches to obtaining suitable predictors for use with PUCB. This section also gives insight into the average behavior of PUCB across episodes drawn from a distribution.

3.1 Offline training that minimizes $1/M_*$

In computer Go it is common to prepare a predictor in advance via an offline process, then freezing it for later use in bandit-based search [14, 19]. One specific approach that has been successful in heuristic use [14] is convex optimization to minimize a logarithmic loss function over training data, using a predictor of the form described in Section 3.1.1. Other authors have also given theoretical consideration to the offline preparation of predictors in a different kind of contextual multi-armed bandits setting [30]. Here, we consider offline training that minimizes $1/M_*$ on test episodes; this produces a predictor that minimizes the regret bound from Theorem 1. This predictor is then frozen for subsequent use with PUCB.

The approach below samples arms sufficiently during training episodes j to find approximately optimal target arms b_j. It then chooses a predictor which minimizes $1/M_{b_j}$ on these training episodes, which effectively optimizes the regret bound of Theorem 1. Uniform convergence then shows that the resulting predictor generalizes to test episodes drawn from the same distribution. With this resulting predictor, PUCB obeys a regret bound on these test episodes. The overall approach is fairly straightforward, but showing that it works requires a significant amount of detail which we present in the remainder of this section.

The agent experiences each of T *training episodes* drawn independently from a fixed but unknown distribution Q. After training on these, the agent then outputs its chosen predictor M for subsequent use with PUCB on *test episodes* drawn independently from the same distribution Q. The distribution Q may range over a vast set of possible episodes with associated contexts, so the predictor learning task requires generalization from a limited set of training episodes to a much larger set of unseen test episodes.

We assume a class of predictors parameterized by $\boldsymbol{x} \in \mathbb{R}^d$ with $\|\boldsymbol{x}\| \leq B$. Given episode context z, the predictor puts weight $M(\boldsymbol{x}; z)_i$ on arm i. To enable successful generalization in the procedure below, we assume that for all z and i the function $1/M(\boldsymbol{x}; z)_i$ is L-Lipschitz with respect to $\boldsymbol{x}$. To enable efficient training via convex optimization, we may also assume here that for all z and i, $1/M(\boldsymbol{x}; z)_i$ is a convex function of $\boldsymbol{x}$. However, convexity is not necessary for Theorem 4 below.

Denote episode q's context by z_q and number of arms by K_q. Let E_q denote expectation over episodes q drawn from Q.

We assume that we are targeting test episodes of maximum length n trials. This maximum length needs to be known at training time so that training episodes are sampled enough to identify sufficiently accurate target arms; larger n require more accuracy in identifying target arms.

Below we describe the agent's offline training procedure (in boldface) as well as its performance. The procedure requires a constant δ in the range $0 < \delta < 0.25$ (this controls failure probability and is used twice).

1. **Draw T training episodes from Q.**
2. **Set $\epsilon = \sqrt{\log(n)/n}$. For each training episode q_j, sample each arm $\lceil \frac{4}{\epsilon^2} \log(2K_{q_j}T/\delta) \rceil$ times and choose the arm obtaining highest observed average payoff as the episode's target arm b_j.** For any one training episode j, expected payoff for b_j is within ϵ of that of the episode's optimal arm, with probability at least $1 - \delta/T$ [17, Theorem 6]. So, with probability at least $1 - \delta$, all target arms b_j are within ϵ of optimal.
3. **Choose x_{trained} to be a value of x which approximately minimizes the function $W_{\text{bound}}(x) = (1/T) \sum_j \left(1/M(x; z_{q_j})_{b_j}\right)$.**
   - As an example, in the case where $1/M(\boldsymbol{x}; z)_i$ is a convex function of $\boldsymbol{x}$ for all z and i, we may use an efficient convex optimization method [27] to approximately minimize $W_{\text{bound}}(\boldsymbol{x})$.
4. **Output predictor $M(x_{\text{trained}}; z)$.**

Theorem 4 *Using predictor $M(\boldsymbol{x}_{\text{trained}}; z)$ output by the offline training procedure, on test episodes drawn from Q, PUCB obtains expected episode regret to the optimal arm bounded by:*

$$O\left(\left(W_{\text{bound}}(\boldsymbol{x}_{\text{trained}}) + \sqrt{\frac{B^2L^2d\log(T)\log(d/\delta)}{T}}\right)\sqrt{n\log(n)}\right)$$

with overall success probability at least $1-2\delta$.

Proof For episode q let G_q be the set of arms with expected payoff within ϵ of optimal. Define $W(\boldsymbol{x}; q) = \min_{i \in G_q}(1/M(\boldsymbol{x}; z_q)_i)$. For each q, the functions $1/M(\boldsymbol{x}; z_q)_i$ are L-Lipschitz with respect to $\boldsymbol{x}$ for all i. The minimum of several such functions is also L-Lipschitz, since it can be seen by examining cases that for real-valued functions $f_1(\boldsymbol{x})$ and $f_2(\boldsymbol{x})$ we have for any $\boldsymbol{x}$ and $\boldsymbol{y}$ in the domain:

$$\left|\left(\min_{i\in\{1,2\}} f_i(\boldsymbol{x})\right) - \left(\min_{i\in\{1,2\}} f_i(\boldsymbol{y})\right)\right| \leq \max_{i\in\{1,2\}} \left|f_i(\boldsymbol{x}) - f_i(\boldsymbol{y})\right|$$

Thus, $W(\boldsymbol{x}; q)$ is L-Lipschitz for all q. Define $\hat{W}(\boldsymbol{x}) = (1/T)\sum_j W(\boldsymbol{x}; q_j)$ as an average over the training set, and $W(\boldsymbol{x}) = E_q[W(\boldsymbol{x}; q)]$ as an expected value over test episodes q drawn from Q. Our conditions suffice for uniform convergence of $\hat{W}(\boldsymbol{x})$ to $W(\boldsymbol{x})$ so that the following [27, Theorem 5] holds for all $\boldsymbol{x}$ with probability at least $1-\delta$:

$$W(\boldsymbol{x}) \leq \hat{W}(\boldsymbol{x}) + O\left(\sqrt{\frac{B^2L^2d\log(T)\log(d/\delta)}{T}}\right) \tag{24}$$

Also with probability at least $1-\delta$, all training episodes j have $b_j \in G_{q_j}$ and thus $W(\boldsymbol{x}; q) \leq 1/M(\boldsymbol{x}; z_{q_j})_{b_j}$, giving that for all $\boldsymbol{x}$:

$$\hat{W}(\boldsymbol{x}) \leq W_{\text{bound}}(\boldsymbol{x}) \tag{25}$$

For episode q, let arm $g_q = \text{argmin}_{i\in G_q}(1/M(\boldsymbol{x}; z_q)_i)$. Using PUCB with predictor $M(\boldsymbol{x}; z_q)$, Theorem 1 bounds expected regret to arm g_q as:

$$17(1/M(\boldsymbol{x}; z_q)_{g_q})\sqrt{n\log(n)} = 17W(\boldsymbol{x}; q)\sqrt{n\log(n)}$$

The regret to the *optimal* arm is at most $n\epsilon$ more than the regret to arm g_q, so regret to the optimal arm for episode q is at most:

$$n\epsilon + 17W(\boldsymbol{x}; q)\sqrt{n\log(n)}$$

Across episodes q drawn from Q, expected regret is at most:

$$n\epsilon + 17W(\boldsymbol{x})\sqrt{n\log(n)}$$

Combining this with (24) and (25) above, and substituting step 2's choice of $\epsilon = \sqrt{\log(n)/n}$ and applying the resulting bound to $\boldsymbol{x}_{\text{trained}}$, we may simplify to obtain the theorem. □

Note that regret can vary substantially from one episode to another, and the theorem bounds an expected value over q drawn from Q as opposed to giving an episode-specific bound. The procedure optimizes W_{bound} which implies optimization of a regret bound if PUCB were used over the training set. Then across test episodes the theorem shows an expected regret bound which, for appropriate choice of parameters, is not too much worse than that implied by W_{bound}.

For Theorem 4 to give a useful bound, we require an appropriate class of predictors for which $\boldsymbol{x}_{\text{trained}}$ can be found giving small $W_{\text{bound}}(\boldsymbol{x}_{\text{trained}})$. We also require d, B, and L to be not too large so that reasonably-sized training set size T will give a good result. We next give a concrete example of a predictor class.

3.1.1 Generalized Bradley–Terry predictor

Generalized Bradley–Terry statistical models have been successfully applied heuristically in conjunction with multi-armed bandits in computer Go [14, 15]. Establish a fixed mapping from context z and arm i to a team of I feature indices $\text{feat}(z, i, 1) \ldots \text{feat}(z, i, I)$. Features associated with any one arm are distinct, but the same feature can be associated with multiple arms. For simplicity of this presentation we assume that all teams have I features, and that all episodes have K arms.

Let $\boldsymbol{x}$ be a vector of d feature weights, each in $[-K\log(cKI)/2, K\log(cKI)/2]$. So for purposes of Theorem 4 we have $B = K\log(cKI)\sqrt{d}/2$. We may have $d \gg I$ with only a subset of feature weights being used in any one episode.

Use a form of Bradley–Terry predictor that gives for each i:

$$M(\boldsymbol{x}; z)_i = \frac{e^{\sum_{j=1}^{I} \frac{1}{KI} x_{\text{feat}(z,i,j)}}}{\sum_{k=1}^{K} e^{\sum_{j=1}^{I} \frac{1}{KI} x_{\text{feat}(z,k,j)}}}$$

Note that $\sum_i M(\boldsymbol{x}; z)_i = 1$ as required for our purposes. As an example, with $I = 1$ and $c = 1$ the value of $M(\boldsymbol{x}; z)_i$ can range approximately between $\frac{1}{K^3}$ and $1 - \frac{1}{K}$. We have:

$$\frac{1}{M(\boldsymbol{x}; z)_i} = \sum_{k=1}^{K} e^{\left(\sum_{j=1}^{I} \frac{1}{KI} x_{\text{feat}(z,k,j)}\right) - \left(\sum_{j=1}^{I} \frac{1}{KI} x_{\text{feat}(z,i,j)}\right)}$$

This is a convex function of vector $\boldsymbol{x}$, so note that efficient convex optimization can be successfully applied in step 3 of the procedure of Section 3.1. Bounding the maximum magnitude of the gradient: the partial derivative with respect to a single component of vector $\boldsymbol{x}$ has at most K nonzero terms, each term at most $\frac{1}{KI} e^{\frac{1}{KI} 2IK \log(cIK)/2} = c$ for a total of at most cK across all terms for the component. There are d such components, and so we have that $1/M(\boldsymbol{x}; z)_i$ is L-Lipschitz with $L \leq cK\sqrt{d}$. Theorem 4 thus yields a regret bound that is polynomial in K, d, and I, and polynomially-sized training set sizes will suffice.

Note that the convexity that enables efficient optimization here comes from the form of the Bradley–Terry predictor, combined with the simplicity of the $1/M_*$ function being optimized. Convexity would not easily be obtained, for example, if we had based our approach on optimizing the more complex bound from Theorem 2. So, the simplicity of the regret bound from Theorem 1 yields an additional advantage here in enabling predictor training based on efficient convex optimization.

3.2 Probability distribution over arms

In some applications, we may be able to map context onto a probability distribution which approximately reflects the probability that an arm will be optimal. For example, in computer Go, one can use samples of human expert games to create predictors which output an approximate distribution over correct move choices in any board position [6, 14]. This in turn approximately reflects the probability that a move will emerge as the eventual optimal choice of bandit-based search. In the context of PUCB, such a probability distribution can be turned into weights M_i as follows.

Theorem 1's regret bound is proportional to $1/M_*$ for optimal arm $*$. If we have probability distribution P_i that exactly reflects the probability that arm i will be optimal, then expected regret is bounded by $17\sqrt{n\log(n)}\sum_i \frac{P_i}{M_i}$. This bound is minimized by M_i proportional to $\sqrt{P_i}$:

Theorem 5 *If arm i is optimal with probability P_i, then using PUCB with weights $M_i = \frac{\sqrt{P_i}}{\sum_j \sqrt{P_j}}$ yields expected episode regret at most $17\sqrt{n\log(n)}\left(\sum_j \sqrt{P_j}\right)^2$.*

Proof If arm i is optimal with probability P_i, then the expected value of the bound from Theorem 1 is:

$$\sum_i \left(P_i \left(17\sqrt{n\log(n)}\frac{\sum_j \sqrt{P_j}}{\sqrt{P_i}}\right)\right)$$

which simplifies to yield the theorem. □

If most P_i satisfy $P_i \ll \frac{1}{K}$ (e.g. most $P_i \sim \frac{1}{K^2}$), this bound can be a substantial improvement over UCB1.

If the agent has only approximate probability estimates P_i, and the (unknown) true probabilities are R_i but our P_i are sufficiently accurate, then the regret bound degrades smoothly:

Theorem 6 *If arm i is optimal with probability R_i, and for all i we have $R_i \le \alpha P_i$ for some $\alpha \ge 1$, then using PUCB with $M_i = \sqrt{P_i}/\left(\sum_j \sqrt{P_j}\right)$ yields expected episode regret at most $\alpha 17\sqrt{n\log(n)}\left(\sum_j \sqrt{P_j}\right)^2$.*

Proof Expected value of the bound from Theorem 1 is:

$$\sum_i \left(R_i \left(17\sqrt{n\log(n)}\frac{\sum_j \sqrt{P_j}}{\sqrt{P_i}}\right)\right) \le \sum_i \left(\alpha P_i \left(17\sqrt{n\log(n)}\frac{\sum_j \sqrt{P_j}}{\sqrt{P_i}}\right)\right)$$

which simplifies as in Theorem 5. □

Rather than using PUCB with the bound from Theorem 1, if we instead use Unrestricted PUCB from Section 2.3.1 then we may take $M_i = \sqrt{P_i}$ without normalizing M_i to sum to 1. We apply this to the simplified regret bound of Corollary 2 (which allows convenient comparison with Theorem 5 above) to obtain:

Theorem 7 *If $n > 400K^2$ and arm i is optimal with probability P_i, then using Unrestricted PUCB with $M_i = \sqrt{P_i}$ yields expected episode regret at most $17\sqrt{n\log(n)}\left(\sum_j \sqrt{P_j}\right)$.*

Proof The expected value of the bound from Corollary 2 is:

$$\sum_i \left(P_i \left(4.25\sqrt{n\log(n)} \left(\sqrt{1/\sqrt{P_i}} + \sqrt{\sum_j \sqrt{P_j}} \right)^2 \right)\right)$$

$$= 4.25\sqrt{n\log(n)} \sum_i \left(\sqrt{P_i} + P_i \left(\sum_j \sqrt{P_j}\right) + 2P_i\sqrt{1/\sqrt{P_i}}\sqrt{\sum_j\sqrt{P_j}}\right)$$

$$= 4.25\sqrt{n\log(n)} \left(\left(\sum_i \sqrt{P_i}\right) + \left(\sum_j\sqrt{P_j}\right) + 2\left(\sum_i P_i\sqrt{1/\sqrt{P_i}}\right)\sqrt{\sum_j\sqrt{P_j}}\right)$$

$$\le 4.25\sqrt{n\log(n)}\left(2\left(\sum_j\sqrt{P_j}\right) + 2\sqrt{\sum_i\sqrt{P_i}}\sqrt{\sum_j\sqrt{P_j}}\right)$$

which simplifies to the bound stated in the Theorem. Above, the last term was bounded using $\sum_i P_i\sqrt{1/\sqrt{P_i}} \le \sqrt{\sum_i P_i/\sqrt{P_i}}$ by Jensen's inequality. □

Theorem 7 is an improvement over Theorem 5.

Finally, corresponding to Theorem 6 above we have:

Theorem 8 *If $n > 400K^2$ and arm i is optimal with probability R_i, and for all i we have $R_i \le \alpha P_i$ for some $\alpha \ge 1$, then using Unrestricted PUCB with $M_i = \sqrt{P_i}$ yields expected episode regret at most $\alpha 17\sqrt{n\log(n)}\left(\sum_j \sqrt{P_j}\right)$.*

Proof The expected value of the bound from Corollary 2 is:

$$\sum_i \left(R_i \left(4.25\sqrt{n\log(n)} \left(\sqrt{1/\sqrt{P_i}} + \sqrt{\sum_j \sqrt{P_j}} \right)^2 \right)\right)$$

$$\le \sum_i \left(\alpha P_i \left(4.25\sqrt{n\log(n)} \left(\sqrt{1/\sqrt{P_i}} + \sqrt{\sum_j \sqrt{P_j}} \right)^2 \right)\right)$$

which then simplifies as in Theorem 7. □

4 Simulation examples

The theorems in Section 2 describe the worst-case scaling of PUCB regret, and are especially informative with long episodes and large numbers of arms. In this section, we use simple simulation experiments to illustrate actual behavior of PUCB, in episodes of only modest length. We compare to a variant of UCB1, here denoted “UCB,” which is identical to UCB1 except that the constant 2 in the confidence bound term is replaced by $\frac{3}{2}$. This variant performed better in preliminary simulation experiments, and provides a more direct comparison to PUCB which also uses a constant of $\frac{3}{2}$ in the confidence bound term. As noted previously, others have discussed UCB-like algorithms with a range of values for this constant (e.g., [2]).

Simulation experiments here use episode length $n = 500$, and various numbers of arms K. Payoffs are sampled from a Bernoulli distribution with payoff rate 0.65 for a single “optimal arm” and payoff rate 0.35 for all remaining “suboptimal arms.”

We illustrate using two different distributions $\boldsymbol{P}$ over the choice of optimal arm, using Unrestricted PUCB from Section 2.3.1 with predictor weights $M_i = \sqrt{P_i}$ as described in Section 3.2. At the start of each episode the environment samples independently from distribution $\boldsymbol{P}$ to select the identity of the optimal arm. For illustration purposes in these simple examples, probabilities $\boldsymbol{P}$ and corresponding predictor weights $\boldsymbol{M}$ are constant across all episodes, and do not depend on any episode context.

We use the term “preferred arm” to refer to the arm with highest probability of being optimal according to $\boldsymbol{P}$. Note the preferred arm is also given highest weight by the predictor. The term “non-preferred arms” refers to the remaining arms. In both of the distributions $\boldsymbol{P}$ used here, the single preferred arm $i = 1$ has the highest probability of being optimal. In addition to PUCB and UCB, we compare to the simple policies of always pulling the single preferred arm $i = 1$, and pulling random arms. Results for these simple policies are computed analytically, and each result presented for PUCB and UCB is an average over one million simulations.

4.1 Uniform distribution over arms $i > 1$

Here, for $i = 1$ we set $P_1 = 0.5$, and for $i > 1$ set a uniform $P_i = \frac{0.5}{K-1}$. Figure 2 shows average trajectories with $K = 3$. With the small number of arms $K = 3$, there is ample time to find the best arm even with the modest episode length $n = 500$, but this provides a useful illustration of behavior. Trajectories are shown for UCB, and for PUCB with results split episodes across episodes where the preferred arm $i = 1$ is optimal and episodes where some other arm $i > 1$ is optimal. With K this small, the trajectories are fairly close but the best result comes from PUCB when $i = 1$ is optimal, the worst from PUCB when $i = 1$ is suboptimal (that is, the initial prediction is essentially wrong), and UCB is in between.

Figure 3 shows, for the same set of simulations, the frequency with which the optimal arm is pulled. After initially looping through the arms, UCB gradually converges towards the optimal arm. PUCB initially focuses exclusively on the preferred arm $i = 1$, and only begins to explore significantly after an initial period. Note that PUCB exploration is delayed somewhat longer when the preferred arm $i = 1$ is optimal, due to the impact of the higher absolute payoff obtained by PUCB initially. PUCB exploration yields improvement when the preferred arm $i = 1$ is suboptimal but necessarily yields a temporary dip in performance in the case where the preferred arm $i = 1$ is optimal.

When K becomes larger, the weight M_i goes down for $i > 1$ and exploration of arms $i > 1$ is further delayed. Figure 4 shows total regret as K varies (since this

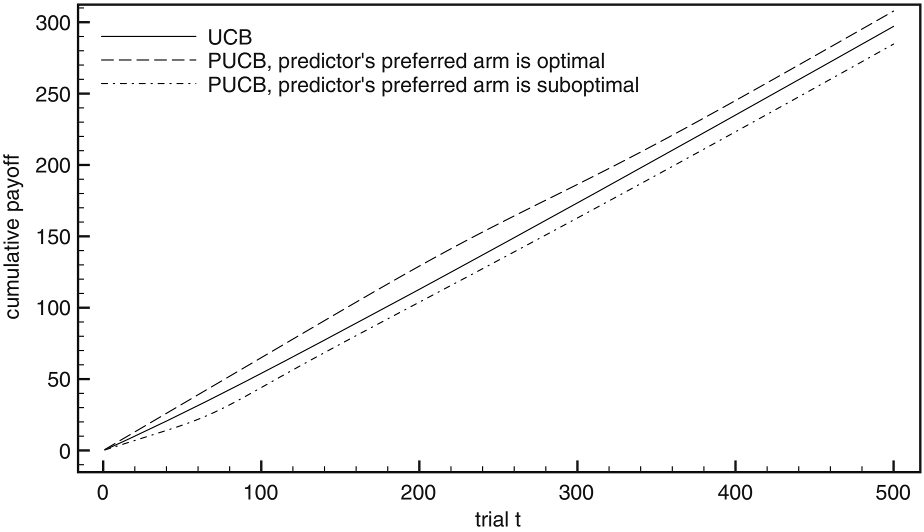

Fig. 2 Cumulative payoff, with optimal arm selected using a uniform distribution over non-preferred arms $i > 1$

graph shows regret, higher numbers on the *y* axis are worse, unlike the previous two graphs). At small *K* there is sufficient time to approximately find the best arm and PUCB performs comparably to UCB. For larger *K*, UCB results necessarily approach those of random guessing. Also for larger *K*, PUCB outperforms UCB but

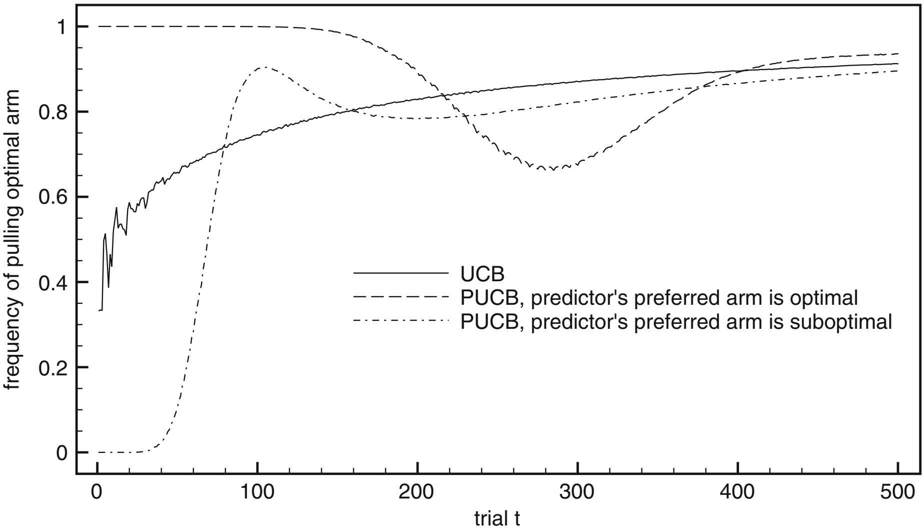

Fig. 3 Frequency of pulling optimal arm in each trial, with uniform distribution over non-preferred arms $i > 1$

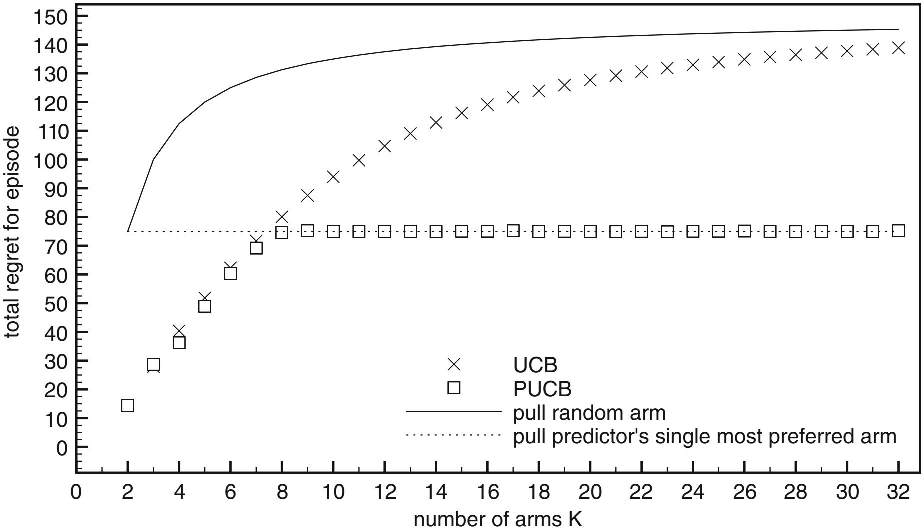

Fig. 4 Total regret for varying numbers of arms, with uniform distribution over non-preferred arms $i > 1$

on this example quickly reaches the point where it cannot obtain much better results than those that could be obtained by always trusting the predictor's single most preferred arm $i = 1$. PUCB automatically makes an appropriate transition between the smaller-K and larger-K regimes.

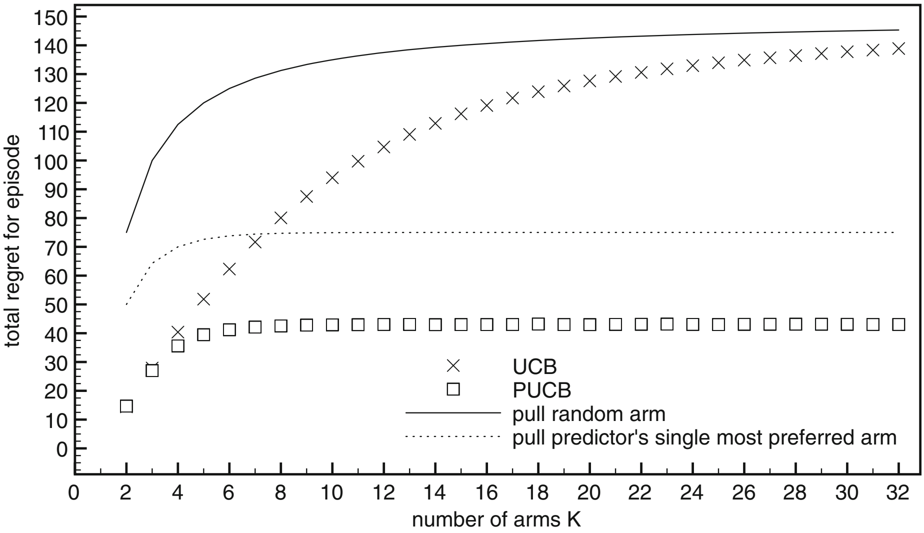

Fig. 5 Total regret for varying numbers of arms, with exponentially decreasing distribution over non-preferred arms $i > 1$

4.2 Exponentially decreasing distribution over arms $i > 1$

Here we use $P_{i+1} = P_i/2$, with the constraint $\sum_i P_i = 1$. Note that P_1 will quickly approach 0.5 as K increases, and this makes results somewhat comparable to those of the previous section. Figure 5 shows total regret as K varies. In this case, even when the preferred arm $i = 1$ is not optimal, the predictor provides good information about which other arms $i > 1$ are most likely to be optimal, and PUCB effectively uses this information to consistently outperform the policy that always pulls the preferred arm $i = 1$.

5 Discussion

In the theoretically tractable stochastic multi-armed bandit setting, we have studied the problem of combining initial predictor recommendations with subsequent exploration. We have presented an efficient algorithm PUCB for multi-armed bandits with episode context. PUCB can benefit greatly from an accurate predictor, but it can also use further exploration to overcome initially inaccurate predictor recommendations. Simple simulation experiments illustrate this behavior. We have quantified PUCB's regret in terms of the quality of the predictor. PUCB can obtain substantially lower regret than UCB1 when an accurate predictor is available. We have also shown how to give PUCB additional favorable properties possessed by UCB1.

We have also described methods for obtaining predictors suitable for use with PUCB. The first method chooses predictors that optimize PUCB's regret bound, and the second method sets predictor weights based on approximate knowledge of the probability distribution of optimal arms. The performance of these methods is characterized in terms of expected regret across episodes drawn from a distribution. This gives insight into PUCB's average performance, when a predictor's recommendations are beneficial for some episodes but may be misleading for others.

An open question: is there an alternative approach that unifies predictor learning (across episodes) and PUCB (within-episode) into a single online learning method without the training/test distinction of Section 3.1? Note that Section 3.1 has longer training episodes (in step 2) than test episodes. If predictor learning needs reliable identification of near-optimal target arms, a known lower bound [25] suggests an unavoidable need for training episodes longer than test episodes. Perhaps in some alternative approach that avoids the training/test distinction, there could be a way to make do with more limited target information; this has been studied in other settings [22].

Acknowledgements Thanks to Mark Land for comments on an earlier draft. Thanks to the anonymous reviewers for their helpful comments.

References

1. Audibert, J.Y., Bubeck, S.: Minimax policies for adversarial and stochastic bandits. In: 22nd Annual Conference on Learning Theory (COLT 2009) (2009)
2. Audibert, J.Y., Munos, R., Szepesvári, C.: Exploration-exploitation tradeoff using variance estimates in multi-armed bandits. Theor. Comput. Sci. **410**(19), 1876–1902 (2009)

3. Auer, P., Cesa-Bianchi, N., Fischer, P.: Finite-time analysis of the multiarmed bandit problem. Mach. Learn. **47**(2–3), 235–256 (2002)
4. Auer, P., Cesa-Bianchi, N., Freund, Y., Schapire, R.E.: The nonstochastic multiarmed bandit problem. SIAM J. Comput. **32**(1), 48–77 (2002)
5. Bouzy, B., Cazenave, T.: Computer Go: An AI oriented survey. Artif. Intell. **132**(1), 39–103 (2001)
6. Bouzy, B., Chaslot, G.: Bayesian generation and integration of K-nearest-neighbor patterns for 19x19 Go. In: IEEE Symposium on Computational Intelligence in Games (CIG05), pp. 176–181 (2005)
7. Bouzy, B., Helmstetter, B.: Monte-Carlo Go developments. In: van den Herik, H.J., Iida, H., Heinz, E.A. (eds.) Advances in Computer Games (ACG 2003), IFIP, vol. 263, pp. 159–174. Springer, New York (2003)
8. Bubeck, S., Munos, R., Stoltz, G.: Pure exploration in multi-armed bandits problems. In: Gavaldà, R., Lugosi, G., Zeugmann, T., Zilles, S. (eds.) Algorithmic Learning Theory (ALT 2009), Lecture Notes in Computer Science, vol. 5809, pp. 23–37. Springer, New York (2009)
9. Cai, X., Wunsch, D.C.: Computer Go: A grand challenge to AI. In: Duch, W., Mandziuk, J. (eds.) Challenges for Computational Intelligence, Studies in Computational Intelligence, vol. 63, pp. 443–465. Springer, New York (2007)
10. Chaslot, G., Winands, M., Uiterwijk, J., van den Herik, H., Bouzy, B.: Progressive strategies for Monte-Carlo tree search. In: Proceedings of the 10th Joint Conference on Information Sciences (JCIS 2007), pp. 655–661 (2007)
11. Chaslot, G., Chatriot, L., Fiter, C., Gelly, S., Hoock, J., Perez, J., Rimmel, A., Teytaud, O.: Combining expert, offline, transient and online knowledge in Monte-Carlo exploration. http://www.lri.fr/~teytaud/eg.pdf (2008)
12. Chaslot, G., Fiter, C., Hoock, J.B., Rimmel, A., Teytaud, O.: Adding expert knowledge and exploration in Monte-Carlo tree search. In: Advances in Computer Games (ACG12). Springer, New York (2009)
13. Coulom, R.: Efficient selectivity and backup operators in Monte-Carlo tree search. In: van den Herik, H.J., Ciancarini, P., Donkers, H.H.L.M. (eds.) Computers and Games (CG 2006), Lecture Notes in Computer Science, vol. 4630, pp. 72–83. Springer, New York (2006)
14. Coulom, R.: Computing Elo ratings of move patterns in the game of Go. In: Computer Games Workshop 2007 (2007)
15. Coulom, R.: Monte-Carlo tree search in crazy stone. In: 12th Game Programming Workshop (GPW-07) (2007)
16. de Mesmay, F., Rimmel, A., Voronenko, Y., Püschel, M.: Bandit-based optimization on graphs with application to library performance tuning. In: Danyluk, A.P., Bottou, L., Littman, M.L. (eds.) International Conference on Machine Learning (ICML 2009), pp. 729–736. ACM, New York (2009)
17. Even-Dar, E., Mannor, S., Mansour, Y.: Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. J. Mach. Learn. Res. **7**, 1079–1105 (2006)
18. Finnsson, H., Björnsson, Y.: Simulation-based approach to general game playing. In: Fox, D., Gomes, C.P. (eds.) Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence, AAAI 2008, Chicago, IL, USA, 13–17 July 2008, pp. 259–264. AAAI Press, Menlo Park (2008)
19. Gelly, S., Silver, D.: Combining online and offline knowledge in UCT. In: Ghahramani, Z. (ed.) International Conference on Machine Learning (ICML 2007), pp. 273–280. ACM, New York (2007)
20. Gelly, S., Silver, D.: Achieving master level play in 9 x 9 computer Go. In: Fox, D., Gomes, C.P. (eds.) Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence, AAAI 2008, Chicago, IL, USA, 13–17 July 2008, pp. 1537–1540. AAAI Press, Menlo Park (2008)
21. Juditsky, A., Nazin, A., Tsybakov, A., Vayatis, N.: Gap-free bounds for multi-armed stochastic bandit. In: World Congress of the International Federation of Automatic Control (IFAC) 2008 (2008)
22. Kakade, S.M., Shalev-Shwartz, S., Tewari, A.: Efficient bandit algorithms for online multiclass prediction. In: Cohen, W.W., McCallum, A., Roweis, S.T. (eds.) International Conference on Machine Learning (ICML 2008), pp. 440–447. ACM, New York (2008)
23. Kocsis, L., Szepesvari, C.: Bandit based Monte-Carlo planning. In: European Conference on Machine Learning (ECML 2006), pp. 282–293 (2006)

24. Langford, J., Zhang, T.: The epoch-greedy algorithm for multi-armed bandits with side information. In: Platt, J.C., Koller, D., Singer, Y., Roweis, S.T. (eds.) Neural Information Processing Systems (NIPS). MIT Press, Cambridge (2007)
25. Mannor, S., Tsitsiklis, J.N.: The sample complexity of exploration in the multi-armed bandit problem. J. Mach. Learn. Res. **5**, 623–648 (2004)
26. Rosin, C.D.: Multi-armed bandits with episode context. In: The Eleventh International Symposium on Artificial Intelligence and Mathematics (ISAIM 2010) (2010)
27. Shalev-Shwartz, S., Shamir, O., Srebro, N., Sridharan, K.: Stochastic convex optimization. In: 22nd Annual Conference on Learning Theory (COLT 2009) (2009)
28. Streeter, M.J., Smith, S.F.: A simple distribution-free approach to the max k-armed bandit problem. In: Benhamou, F. (ed.) Principles and Practice of Constraint Programming (CP 2006), Lecture Notes in Computer Science, vol. 4204, pp. 560–574. Springer, New York (2006)
29. Strehl, A.L., Mesterharm, C., Littman, M.L., Hirsh, H.: Experience-efficient learning in associative bandit problems. In: Cohen, W.W., Moore, A. (eds.) International Conference on Machine Learning (ICML 2006), pp. 889–896. ACM, New York (2006)
30. Strehl, A.L., Langford, J., Li, L., Kakade, S.M.: Learning from Logged Implicit Exploration Data. In: Lafferty, J., Williams, C.K.I, Shawe-Taylor, J., Zemel, R.S., Culotta, A. (eds.) Neural Information Processing Systems (NIPS) (2010)
31. Teytaud, O., Gelly, S., Sebag, M.: Anytime many-armed bandits. In: Zucker, J., Cornuéjols, A. (eds.) Conférence d'Apprentissage (CAP07), pp. 387–402 (2007)