Multi-armed Bandits with Episode Context

Christopher D. Rosin
Parity Computing, Inc. 6160 Lusk Blvd, Suite C205, San Diego, CA 92121
c.rosin@paritycomputing.com

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 $O(\sqrt{Kn\log(n)})$ regret. 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 $O(\frac{1}{M_*}\sqrt{n\log(n)})$ where M_* is the weight on the optimal arm. We also discuss methods for obtaining suitable predictors for use with PUCB.

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 (Auer, Cesa-Bianchi, & Fischer 2002).

A foundation for the work here is the UCB1 algorithm for stochastic multi-armed bandits (Auer, Cesa-Bianchi, & Fischer 2002). UCB1 maintains an empirical average payoff x_i on each arm i, and each trial pulls the arm maximizing an upper confidence bound $x_i + \sqrt{\frac{2\log(n)}{s_i}}$ where s_i is the number of previous pulls of i, and n is the total number of trials so far. The acronym "UCB" comes from "upper confidence bound." This simple algorithm successfully achieves worst-case expected regret upper-bounded by $O(\sqrt{Kn\log(n)})$.

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 (Bouzy & Cazenave 2001; Cai & Wunsch 2007), but major advances have been obtained using Monte Carlo techniques that evaluate positions using random playouts (Bouzy & Helmstetter 2003; Gelly & Silver 2007; 2008). An important development has been the efficient combination of Monte Carlo evaluation with tree search. In particular, the UCT algorithm (Kocsis & Szepesvari 2006) 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 (Gelly & Silver 2007), and has generally replaced earlier heuristics for Monte Carlo tree search (e.g. (Bouzy & Helmstetter 2003; Coulom 2006)) which lacked the theoretical regret bounds behind UCB1. Following success in Go, UCT-based methods have been applied successfully to other domains (Finnsson & Björnsson 2008; de Mesmay *et al.* 2009).

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" bandit problem relevant at the root of a search tree (Bubeck, Munos, & Stoltz 2009), studies of bandits with large numbers of arms (Teytaud, Gelly, & Sebag 2007), and fundamental improvements in UCB-style bandit algorithms and regret bounds (Audibert, Munos, & Szepesvári 2009; Audibert & Bubeck 2009). But computer Go has progressed further with trial-and-error development of more complex heuristics (e.g. (Chaslot *et al.* 2008)) for which there is little theoretical understanding of the type available for UCB1.

In this paper, we examine theoretically 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 ban-

Copyright © 2009, authors listed above. All rights reserved.

dit 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 (Bouzy & Cazenave 2001). 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 (Coulom 2007a; 2007b), and an additive bias to UCB1's payoff estimates (Chaslot *et al.* 2007). The algorithm presented in this paper modifies UCB1 with a novel form of additive bias, and we show this enables an advantageous regret bound.

This paper's 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.

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 (Langford & Zhang 2007; Wang, Kulkarni, & Poor 2005; Strehl *et al.* 2006). In this prior work, the context has been allowed to change every trial. This is more general, but is also more difficult than necessary for modelling applications like bandit-based decisions in computer Go. Also, regret bounds from previous theoretical work on contextual multi-armed bandits do not satisfy our technical goals described below.

Goals: 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 $\Omega(\sqrt{Kn})$ (Auer *et al.* 2002), and the UCB1 algorithm achieves worst-case regret at most $O(\sqrt{Kn\log(n)})$ (Auer, Cesa-Bianchi, & Fischer 2002; Streeter & Smith 2006). In our setting, a predictor uses context to assign a vector of arm weights $\mathbf{M}$ 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 $O(f(\mathbf{M})g(n))$, with $f(\mathbf{M})$ 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 $\mathbf{M}$ 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.

We seek an efficient algorithm that is effective both for short episodes (including $n < K$) and long episodes ($n \gg K$).

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

In Section 2.3, we then show how the predictor can be modified to give PUCB additional favorable properties possessed by UCB1. Specifically, one modification enables PUCB to revert smoothly towards UCB1's worst-case regret bound of the form $O(\sqrt{Kn\log(n)})$ as the M_i become uniform (such uniform M_i provide no information about which arm is best). A separate second modification 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.

Then, Section 3 describes methods for obtaining suitable predictors for use with PUCB.

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 $\mathbf{M}$ with $M_i > 0$ for each i, and $\sum_i M_i = 1$. 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 $1 \le i \le K$. 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 $1 \le t \le n$, the agent pulls arm i_t and receives payoff x_{i_t} chosen independently according to D_i.
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:

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

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 (Auer, Cesa-Bianchi, & Fischer 2002) achieves expected regret at most:

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

In addition to the bound expressed above, regret is $\Delta_i \leq 1$ per trial in which arm i is pulled, and so total regret is also upper-bounded by 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 $\Theta(\sqrt{K\log(n)/n})$ yields an expected regret bound of $O(\sqrt{Kn\log(n)})$ (Streeter & Smith 2006; Juditsky *et al.* 2008). We focus here on this worst-case bound.

Note that if all Δ_i are a larger $\Theta(K\sqrt{\log(n)/n})$, 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 Figure 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 $\frac{1}{K}$, 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.

The policy 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 UCB1 using a range of values for this constant (Audibert, Munos, & Szepesvári 2009).

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.

First we will present some notation and prove a lemma.

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 x_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_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.

The following lemma will help handle relatively small n.

Lemma 2 *At most $\frac{1.61\sqrt{n}}{M_*}$ distinct arms are pulled during the episode.*

Proof of Lemma 2: We want to show that low M_i arms are not pulled, during a sufficiently short episode. Assume

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

Now show that i cannot be pulled during the episode. 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:

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

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

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

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. Since $\sum_i M_i = 1$, the number of arms not satisfying the original assumption is at most $\frac{1.61\sqrt{n}}{M_*}$. This is an upper bound on the number of distinct arms that may be

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

- s_i is the number of previous pulls on i
- x_i is i's average payoff so far if $s_i > 0$, otherwise 1
- $c(t,s) = \sqrt{\frac{3\log(t)}{2s}}$ if $s > 0$, otherwise 0
- $m(t,i) = \frac{2}{M_i}\sqrt{\frac{\log(t)}{t}}$ if $t > 1$, otherwise $\frac{2}{M_i}$

Figure 1: Multi-armed bandit policy PUCB

pulled during the episode. □

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 (Auer, Cesa-Bianchi, & Fischer 2002; Audibert, Munos, & Szepesvári 2009).

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

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

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:

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

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

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

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

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

with expectation E taken over random sequences of payoffs.

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

Since $\Delta_i \leq 1$ and Lemma 2 gives $|\{i \in A\}| \leq \frac{1.61\sqrt{n}}{M_*}$:

$$R_* \leq \frac{1.61\sqrt{n}}{M_*} + \sum_{i \in A} \Delta_i (E[s^{\text{tail}}_{n+1,i}] + E[s^{\text{close}}_{n+1,i}])$$

Define:

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

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

So that:

$$R_* \leq \frac{1.61\sqrt{n}}{M_*} + R^{\text{tail}} + R^{\text{close}} \tag{5}$$

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

Regret R^{tail}:

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

Using E to denote expected value over sequences of payoffs:

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

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:

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

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

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

In the worst case, regret for these pulls is at most 1 per trial. By Lemma 2, the number of arms with at least one pull is at most $\frac{1.61\sqrt{n}}{M_*}$, so the regret associated with $s^{\text{tail}}_{n+1,i}$, summed across all arms i with at least one pull, is bounded:

$$R^{\text{tail}} \leq (\frac{\pi^2}{3}) \frac{1.61\sqrt{n}}{M_*} < \frac{5.3\sqrt{n}}{M_*} \tag{6}$$

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 that have $m(n,i) - m(n,*) < \frac{m(n,i)}{2}$. 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:

$$\begin{aligned} \mu_* - m(t,*) &\leq \mu_i + 2c(t,s_i) - m(t,i) \\ \mu_* - \mu_i &\leq 2c(t,s_i) + m(t,*) \end{aligned}$$

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 $i_t \in A_{(a)}$.

$$\begin{aligned} R^{\text{close}}_{(a)} &\leq \sum_{t=1}^{n}(2c(t,s_{i_t}) + m(t,*)) \\ &\leq (\sum_{\{i \in A_{(a)}\}} \sum_{s_i=1}^{s^{\text{close}}_{n+1,i}} 2c(n,s_i)) + (\sum_{t=1}^{n} m(t,*)) \\ &\leq (\sum_{\{i \in A_{(a)}\}} (2\sqrt{\frac{3}{2}\log(n)}) \sum_{s_i=1}^{s^{\text{close}}_{n+1,i}} \frac{1}{\sqrt{s_i}}) + \\ &\quad ((2\frac{1}{M_*}\sqrt{\log(n)}) \sum_{t=1}^{n} \frac{1}{\sqrt{t}}) \end{aligned}$$

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

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

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

with:

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

Because $m(n,i) - m(n,*) < \frac{m(n,i)}{2}$ for type (a) arms, $m(n,i) < 2m(n,*)$ and so $M_i > \frac{1}{2}M_*$. 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 $|\{i \in A_{(a)}\}|$. Now, since Jensen's inequality gives $\frac{1}{|S|}\sum_{i \in S}\sqrt{s_i} \leq \sqrt{\frac{1}{|S|}\sum_{i \in S} s_i}$:

$$\begin{aligned} &\sum_{\{i \in A_{(a)}\}} 4\sqrt{\frac{3}{2}\log(n) s^{\text{close}}_{n+1,i}} \\ &\leq 4\sqrt{\frac{3}{2}\log(n)}\sqrt{|\{i \in A_{(a)}\}|}\sqrt{\sum_{\{i \in A_{(a)}\}} s^{\text{close}}_{n+1,i}} \\ &\leq 4\sqrt{\frac{3}{2}\log(n)}\sqrt{2\frac{1}{M_*}}\sqrt{n} = 4\sqrt{3n\log(n)\frac{1}{M_*}} \end{aligned}$$

So:

$$R^{\text{close}}_{(a)} \leq \sqrt{n\log(n)}(4\sqrt{\frac{3}{M_*}} + \frac{4}{M_*}) < \frac{11\sqrt{n\log(n)}}{M_*} \quad (7)$$

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

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

$$s_{t,i} > \frac{6\log(n)}{(\mu_* - \mu_i + (m(n,i)/2))^2} \quad (8)$$

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

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

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

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

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

$$R^{\text{close}}_{(b)} \leq 3\sqrt{n\log(n)} \quad (9)$$

Total regret: Using $R^{\text{close}} = R^{\text{close}}_{(a)} + R^{\text{close}}_{(b)}$ and substituting (6), (7) and (9) into (5):

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

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 Lemma 2 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.

Recovering UCB1's Worst-Case Bound: Theorem 1 can't match UCB1's worst-case regret $O(\sqrt{Kn\log(n)})$, independent of the identity of $*$. Uniform M_i yields the worse $O(K\sqrt{n\log(n)})$. 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.

Theorem 3 *With weights $0 < M_i \leq 1$, compared to any fixed arm $*$, for $n > 1$ expected regret for the episode satisfies $R_* \leq 4\sqrt{n\log(n)}(\sqrt{Z} + \sqrt{\frac{1}{M_*}})^2 + \min(5K, \frac{7Z\sqrt{n}}{M_*})$*

The proof closely follows that of Theorem 1; we summarize the differences here.

The upper bound on number of distinct arms pulled in Lemma 2 is replaced by $\min(K, \frac{1.61Z\sqrt{n}}{M_*})$. The proof of this modified Lemma 2 is essentially the same until the final step where $\sum_i M_i = Z$ is now used instead of $\sum_i M_i = 1$. The bound is also modified by capping it at the total number of arms K. This modified bound also serves as a bound on the regret from the first pull of each arm, replacing the first term in (5).

The analysis of R^{tail} is the same until its final step where the modified Lemma 2 from above is applied to yield the modified bound:

$$R^{\text{tail}} \leq (\frac{\pi^2}{3})\min(K, \frac{1.61Z\sqrt{n}}{M_*}) \tag{10}$$

The maximum number of type (a) arms becomes $2\frac{Z}{M_*}$ instead of $2\frac{1}{M_*}$. This leads to

$$R^{\text{close}}_{(a)} \leq \sqrt{n\log(n)}(4\sqrt{\frac{3Z}{M_*}} + \frac{4}{M_*}) \tag{11}$$

For type (b) arms, the analysis is the same until its final step, where the sum of $3M_i\sqrt{n\log(n)}$ over i is equal to $3Z\sqrt{n\log(n)}$; this is the bound on $R^{\text{close}}_{(b)}$.

The bound on R_* is the sum of these modified terms above, which simplifies to the form stated in Theorem 3.

As an example of Theorem 3: with $M_i = \frac{1}{\sqrt{K}}$ for all i, $\frac{1}{M_*} = Z = \sqrt{K}$ and the bound is $O(\sqrt{Kn\log(n)})$, independent of the choice of $*$. This is comparable to the worst-case bound for UCB1.

Large Δ_i: 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 $\sum_i M'_i = 1$. 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 $\frac{3}{2}$). Now, $M'_i > \frac{1}{3K}$ for all i.

Theorem 4 *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:*

$$R_* \leq 17\frac{1}{M'_*}\sqrt{n_0\log(n_0)} + (8\sum_{i:\mu_i<\mu_*}\frac{\log(n)}{\Delta_i}) + 5K$$

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).

For the first n_0 trials, use the original Theorem 1 with the weights M'_i. This bounds regret for these n_0 trials by at most $17\frac{1}{M'_*}\sqrt{n_0\log(n_0)}$. This gives the first term of the overall regret bound in Theorem 4.

For trials after the first n_0, the proof that additional regret is bounded by $(8\sum_{i:\mu_i<\mu_*}\frac{\log(n)}{\Delta_i}) + 5K$ is an adaptation of the proof of Theorem 1. We summarize the differences here. Note that some regret from the first n_0 trials will be counted again here; this is done to enable a simpler analysis.

To account for arms that receive their first pull after the first n_0 trials, replace Lemma 2's upper bound, on the number of arms pulled, by K. Due to this, the first term in (5) is replaced by K, and the upper bound in (6) is replaced by

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

Eliminate the analysis for type (a) arms; R^{close} is analyzed for all suboptimal arms by adapting the analysis for type (b) arms as follows. Assume there is a pull on suboptimal arm i (beyond its first pull) at time $t > n_0$ with (4) satisfied and

$$s_{t,i} > \frac{6\log(n)}{(\mu_* - \mu_i - 6K\sqrt{\log(n_0)/n_0})^2} \tag{12}$$

giving:

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

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

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

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

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. This bounds the time of the final pull on arm i with $t > n_0$ and with (4) true. The remaining analysis of R^{close} proceeds as in the original proof for type (b) arms, using the form of (12) instead of (8). Using the constraint on Δ_i, the regret bound associated with $s^{\text{close}}_{n+1,i}$ can then be simplified to $8\log(n)/\Delta_i$. Summing the regret terms yields the result.

3 Obtaining Predictors for Use with PUCB

We summarize two approaches to obtaining suitable predictors for use with PUCB.

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 (Gelly & Silver 2007; Coulom 2007a). One specific approach that has been successful in heuristic use (Coulom 2007a) is convex optimization to minimize a logarithmic loss function over training data, using a model of the form described in Section 3.1.1. Here, we consider offline training that minimizes a loss function based on $1/M_*$; this produces a predictor that minimizes the regret bound from Theorem 1. This predictor is then frozen for subsequent use with PUCB.

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 $x \in \mathbb{R}^d$ with $\|x\| \leq B$. Given episode context z, the predictor puts weight $M(x;z)_i$ on arm i. To enable successful generalization in the procedure below, we assume that $1/M(x;z)_i$ is L-Lipschitz with respect to x. To enable efficient learning, we may also assume that for all z and i, $1/M(x;z)_i$ is a convex function of x.

Below we describe the agent's procedure (in boldface) as well as its performance. The procedure uses input parameters $0 < \epsilon < 1$ (during training, target arms are within ϵ of optimal) and $0 < \delta < 0.25$ (this controls failure probability and is used several times).

- **Draw T training episodes from Q.**
- **For episode j with context z_j and K_j arms, sample each arm $\frac{4}{\epsilon^2}\log(2K_jT/\delta)$ times and choose target arm b_j obtaining highest observed average payoff.** Arm b_j has expected payoff within ϵ of the episode's optimal arm, with probability at least $1 - \delta/T$ [(Even-Dar, Mannor, & Mansour 2006) Theorem 6].
- **Choose x that minimizes (or approximately minimizes) $F(x) = \frac{1}{T}\sum_j 1/M(x;z_j)_{b_j}$.** This can be viewed as minimizing average $1/M_*$ over the training set (with $*$ here being within ϵ of the optimal arm).
  - As a specific example of how this minimization can be done in the convex case: stochastic convex optimization (Shalev-Shwartz *et al.* 2009) efficiently finds x such that $F(x)$ is within $O(\sqrt{B^2L^2\log(1/\delta)/T})$ of optimal, with probability at least $1-\delta$ [(Shalev-Shwartz *et al.* 2009) equation (7)].
- Now, with probability at least $1 - \delta$, the chosen x has expected value of $1/M(x;z)_b$ for episodes drawn from Q that is within $O(\sqrt{L^2B^2d\log(T)\log(d/\delta)/T})$ of the $F(x)$ observed on the training set [(Shalev-Shwartz *et al.* 2009) Theorem 5]. That is, the chosen x generalizes successfully with high probability. Here, b is some arm with expected payoff for the episode within ϵ of the episode's optimal arm.
- **Run PUCB with predictor weights $M(x;z)_i$ on test episodes drawn from Q.** The predictor enables PUCB to obtain bounded regret with respect to some arm b that is within ϵ of optimal. To express the overall regret bound comparing to the *optimal* arm, combine the bounds above with Theorem 1: expected episode regret to the optimal arm $*$ is bounded by:

$$O(\epsilon + \sqrt{n\log(n)}(F(x) + \sqrt{B^2L^2d\log(T)\log(d/\delta)/T})) \quad (13)$$

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

Regret can be reduced by increasing training set size, and by decreasing ϵ via longer training episodes. As an example of the latter, if targeting test episodes of length n then choosing $\epsilon = \sqrt{\log(n)/n}$ allows (13) to be simplified by absorbing the ϵ term into the second ($\sqrt{n\log(n)}$) term. Note, though, that this results in training episodes that are substantially longer than test episodes.

Success also depends on choosing an appropriate class of predictors for which x can be found that has small $F(x)$. We next give a concrete example of a predictor class.

3.1.1 Generalized Bradley-Terry Model Generalized Bradley-Terry statistical models have been successfully applied heuristically in conjunction with multi-armed bandits

in computer Go (Coulom 2007a; 2007b). Establish a fixed mapping from context z and arm i to a team of W feature indices $\text{feat}(z,i,1)\ldots\text{feat}(z,i,W)$. The same feature can be associated with multiple arms. In an episode with K arms, use a form of the Bradley-Terry model that gives for each i:

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

Now, $\frac{1}{M_i}$ is a convex function of vector x, and stochastic convex optimization can be successfully applied within the procedure of Section 3.1.

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 models which output an approximate distribution over correct move choices in any board position (Coulom 2007a; Bouzy & Chaslot 2005). 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 $1/M_*$. 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 $\sqrt{P_i}$ (scaling M_i to sum to 1), giving expected regret

$$O(\sqrt{n\log(n)}(\sum_i \sqrt{P_i})^2) \tag{14}$$

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

If our P_i are only approximate, and the (unknown) true probabilities are R_i but our P_i are sufficiently accurate so that $R_i \leq \alpha P_i$ for some $\alpha \geq 1$ and for all i, then the regret bound is only degraded by a factor of α.

If the case where $n > K^2$, then with the approach of Theorem 3 we may take $M_i = \sqrt{P_i}$ without normalizing M_i to sum to 1. The regret bound of Theorem 3 then simplifies to

$$O(\sqrt{n\log(n)}(\sum_i \sqrt{P_i})) \tag{15}$$

which is an improvement over (14). As above, if P_i are approximate but the true probabilities satisfy $R_i \leq \alpha P_i$ then the bound is degraded only by factor α.

4 Discussion

We have shown an efficient algorithm PUCB for multi-armed bandits with episode context. PUCB combines the recommendations of a predictor with further exploration during an episode, and its regret is quantified in terms of the quality of the predictor. We have also described methods for obtaining predictors suitable for use with PUCB.

It remains to be seen whether PUCB can be adapted for practical use in a full application like Go.

An open question: can we unify predictor learning (across episodes) and PUCB (within-episode) into one online learning algorithm without the training/test distinction of Section 3.1? If predictor learning needs reliable identification of near-optimal target arms, a known lower bound (Mannor & Tsitsiklis 2004) suggests training episodes need to be longer than test episodes. But there may exist alternatives that make do with limited target information; this has been studied in other settings (Kakade, Shalev-Shwartz, & Tewari 2008).

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

References

Audibert, J.-Y., and Bubeck, S. 2009. Minimax policies for adversarial and stochastic bandits. In *COLT*.

Audibert, J.-Y.; Munos, R.; and Szepesvári, C. 2009. Exploration-exploitation tradeoff using variance estimates in multi-armed bandits. *Theor. Comput. Sci.* 410(19):1876–1902.

Auer, P.; Cesa-Bianchi, N.; Freund, Y.; and Schapire, R. E. 2002. The nonstochastic multiarmed bandit problem. *SIAM J. Comput.* 32(1):48–77.

Auer, P.; Cesa-Bianchi, N.; and Fischer, P. 2002. Finite-time analysis of the multiarmed bandit problem. *Machine Learning* 47(2-3):235–256.

Bouzy, B., and Cazenave, T. 2001. Computer Go: An AI oriented survey. *Artif. Intell.* 132(1):39–103.

Bouzy, B., and Chaslot, G. 2005. Bayesian generation and integration of K-nearest-neighbor patterns for 19x19 Go. In *IEEE Symposium on Computational Intelligence in Games*, 176–181.

Bouzy, B., and Helmstetter, B. 2003. Monte-Carlo Go developments. In van den Herik, H. J.; Iida, H.; and Heinz, E. A., eds., *ACG*, volume 263 of *IFIP*, 159–174. Kluwer.

Bubeck, S.; Munos, R.; and Stoltz, G. 2009. Pure exploration in multi-armed bandits problems. In Gavaldà, R.; Lugosi, G.; Zeugmann, T.; and Zilles, S., eds., *ALT*, volume 5809 of *Lecture Notes in Computer Science*, 23–37. Springer.

Cai, X., and Wunsch, D. C. 2007. Computer Go: A grand challenge to AI. In Duch, W., and Mandziuk, J., eds., *Challenges for Computational Intelligence*, volume 63 of *Studies in Computational Intelligence*. Springer. 443–465.

Chaslot, G.; Winands, M.; Uiterwijk, J.; van den Herik, H.; and Bouzy, B. 2007. Progressive strategies for Monte-Carlo tree search. In *Proceedings of the 10th Joint Conference on Information Sciences (JCIS 2007)*, 655–661.

Chaslot, G.; Chatriot, L.; Fiter, C.; Gelly, S.; Hoock, J.; Perez, J.; Rimmel, A.; and Teytaud, O. 2008. Combining expert, offline, transient and online knowledge in Monte-Carlo exploration. http://www.lri.fr/~teytaud/eg.pdf.

Coulom, R. 2006. Efficient selectivity and backup operators in Monte-Carlo tree search. In van den Herik, H. J.; Ciancarini, P.; and Donkers, H. H. L. M., eds., *Computers and Games*, volume 4630 of *Lecture Notes in Computer Science*, 72–83. Springer.

Coulom, R. 2007a. Computing Elo ratings of move patterns in the game of Go. In *Computer Games Workshop*.

Coulom, R. 2007b. Monte-Carlo tree search in Crazy Stone. In *12th Game Programming Workshop*.

de Mesmay, F.; Rimmel, A.; Voronenko, Y.; and Püschel, M. 2009. Bandit-based optimization on graphs with application to library performance tuning. In Danyluk, A. P.; Bottou, L.; and Littman, M. L., eds., *ICML*, volume 382 of *ACM International Conference Proceeding Series*, 92. ACM.

Even-Dar, E.; Mannor, S.; and Mansour, Y. 2006. Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. *Journal of Machine Learning Research* 7:1079–1105.

Finnsson, H., and Björnsson, Y. 2008. Simulation-based approach to general game playing. In Fox and Gomes (2008), 259–264.

Fox, D., and Gomes, C. P., eds. 2008. *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence, AAAI 2008, Chicago, Illinois, USA, July 13-17, 2008*. AAAI Press.

Gelly, S., and Silver, D. 2007. Combining online and offline knowledge in UCT. In Ghahramani, Z., ed., *ICML*, volume 227 of *ACM International Conference Proceeding Series*, 273–280. ACM.

Gelly, S., and Silver, D. 2008. Achieving master level play in 9 x 9 computer Go. In Fox and Gomes (2008), 1537–1540.

Juditsky, A.; Nazin, A.; Tsybakov, A.; and Vayatis, N. 2008. Gap-free bounds for multi-armed stochastic bandit. In *World Congr. of IFAC*.

Kakade, S. M.; Shalev-Shwartz, S.; and Tewari, A. 2008. Efficient bandit algorithms for online multiclass prediction. In Cohen, W. W.; McCallum, A.; and Roweis, S. T., eds., *ICML*, volume 307 of *ACM International Conference Proceeding Series*, 440–447. ACM.

Kocsis, L., and Szepesvari, C. 2006. Bandit based Monte-Carlo planning. In *ECML*.

Langford, J., and Zhang, T. 2007. The epoch-greedy algorithm for multi-armed bandits with side information. In Platt, J. C.; Koller, D.; Singer, Y.; and Roweis, S. T., eds., *NIPS*. MIT Press.

Mannor, S., and Tsitsiklis, J. N. 2004. The sample complexity of exploration in the multi-armed bandit problem. *Journal of Machine Learning Research* 5:623–648.

Shalev-Shwartz, S.; Shamir, O.; Srebro, N.; and Sridharan, K. 2009. Stochastic convex optimization. In *COLT*.

Streeter, M. J., and Smith, S. F. 2006. A simple distribution-free approach to the max k-armed bandit problem. In Benhamou, F., ed., *CP*, volume 4204 of *Lecture Notes in Computer Science*, 560–574. Springer.

Strehl, A. L.; Mesterharm, C.; Littman, M. L.; and Hirsh, H. 2006. Experience-efficient learning in associative bandit problems. In Cohen, W. W., and Moore, A., eds., *ICML*, volume 148 of *ACM International Conference Proceeding Series*, 889–896. ACM.

Teytaud, O.; Gelly, S.; and Sebag, M. 2007. Anytime many-armed bandits. In *CAP07*.

Wang, C.-C.; Kulkarni, S.; and Poor, H. 2005. Bandit problems with side observations. *IEEE Tr. Aut. Cont.* 50:338–355.