Online combinatorial optimization with stochastic decision sets and adversarial losses

Gergely Neu  Michal Valko
SequeL team, INRIA Lille – Nord Europe, France
{gergely.neu,michal.valko}@inria.fr

Abstract

Most work on sequential learning assumes a fixed set of actions that are available all the time to choose from. However, in practice, actions can consist of picking subsets of readings from sensors that may break from time to time, road segments that can be blocked or goods that are out of stock. In this paper we study learning algorithms that are able to deal with stochastic availability of such unreliable composite actions. We propose and analyze algorithms based on the Follow-The-Perturbed-Leader prediction method for several learning settings differing in the feedback provided to the learner. Our algorithms rely on a novel loss estimation technique that we call Counting Asleep Times. We deliver regret bounds for our algorithms for the previously studied full information and (semi-)bandit settings, as well as a natural middle point between the two that we call the restricted information setting. A special consequence of our results is a significant improvement of the best known performance guarantees achieved by an efficient algorithm for the sleeping bandit problem with stochastic availability. Finally, we evaluate our algorithms empirically and show their improvement over the known approaches.

1 Introduction

In online learning problems [4] we aim to sequentially select actions from a given set in order to minimize a given loss. However, in many sequential learning problems we have to deal with situations when some of the actions are not available to be taken. A simple and well-studied problem where such situations arise is that of sequential routing [8], where we have to select every day an itinerary for commuting from home to work so as to minimize the total time spent driving (or even worse, stuck in a traffic jam). In this scenario, some road segments may be blocked for maintenance, forcing us to work with the rest of the road network. This problem is isomorphic to packet routing in ad-hoc computer networks where some links might not be always available because of a faulty transmitter or a depleted battery. Another important class of sequential decision-making problems when unavailability occurs is recommender systems [11]. Here, some items may be out of stock or some service may not be applicable at some time (e.g., a movie not shown that day, bandwidth issues in video streaming services). In these cases, the advertiser may refrain from recommending unavailable items. Other reasons include a distributor being overloaded with commands or facing shipment problems.

Learning problems with such partial availability restrictions have been previously studied in the framework of prediction with expert advice. Freund et al. [7] considered the problem of online prediction with specialist experts, where some experts’ predictions might not be available from time to time. Kleinberg et al. [15] gave efficient algorithms for this problem that work under stochastic assumptions on the losses, referring to this setting as prediction with sleeping experts. They have also introduced the setting of sleeping bandit problems where the learner only gets partial feedback about its decisions. They gave an inefficient algorithm for the non-stochastic case, with some suggestions that it might be difficult to learn efficiently in this general setting. This was later reaffirmed
by Kanade and Steinke [14], who show reduce the problem of PAC learning of DNF formulas to a
non-stochastic sleeping experts problem, proving the hardness of learning in this setup. The stochastic
assumption on the losses was introduced by Kanade et al. [13], who gave efficient algorithms for
the sleeping experts and bandits problems under a stochastic a assumption on the decision set.

In this paper, we extend the work of Kanade et al. [13] to combinatorial settings where the action
set of the learner is possibly huge, but has a compact representation. We also assume stochastic
action availability, where in each decision period some actions might be missing at random, with
some fixed but unknown probability. Specifically, we assume that at any fixed decision round $t$, the
decision set of the learner is drawn from a fixed probability distribution independently of the history
of interaction between the learner and the environment. The goal of the learner is to minimize the
sum of losses associated with its decisions. As usual in online settings, we measure the performance
of the learning algorithm by its regret defined as the gap between the total loss of the best fixed
decision-making policy from a pool of policies and the total loss of the learner. When all actions
are available without disruption, it is sensible to measure regret against the set of stationary policies
playing a fixed action in all decision rounds. These policies, however, are not realizable in our
setting, since not all actions are available in all time steps. Following Kanade et al. [13] (see also
Kleinberg et al. [15]), we set our goal as minimizing regret against the best fixed policy.

We study the above online combinatorial optimization setting under three feedback assumptions.
Besides the full-information setting considered by Kanade et al. [13], we also consider a restricted
feedback scheme as a natural middle ground between assuming full information and bandit feedback
by assuming that the learner gets to know the losses associated only with available actions. This
extension (also studied by [15]) is crucially important in practice, since in most cases it is unrealistic
to expect that an unavailable expert would report its loss. Finally, we also consider the case of
semi-bandit feedback that arises naturally in a number of practical problems that can be described
as instances of online combinatorial optimization.

Our main contributions in this paper are two algorithms called SLEEPING CAT and SLEEPING CAT-
BANDIT that work in the restricted and semi-bandit information schemes, respectively. The best
know competitor of our algorithms is the BSFPL algorithm of Kanade et al. [13] that works in two
phases. The initial phase is dedicated to the estimation of the distribution of the available actions.
In the main phase, BSFPL randomly alternates between exploration and exploitation. Our algo-

2 Background

We now give the formal definition of the learning problem. We consider a sequential interaction
scheme between a learner and an environment where in each round $t \in [T] = \{1, 2, \ldots, T\}$, the
learner has to choose an action $V_t$ from a subset $S_t$ of a known decision set $S \subseteq \{0,1\}^d$ with
$\|v\|_1 \leq m$ for all $v \in S$. We assume that the environment selects $S_t$ according to some fixed (but
unknown) distribution $\mathcal{P}$, independently of the interaction history. Unaware of the learner’s decision,
the environment also decides on a loss vector $\ell_t \in [0,1]^d$ that will determine the loss suffered by
the learner, which is of the form $V_t^\top \ell_t$. We make no assumptions on how the environment generates
the sequence of loss vectors, that is, we are interested in algorithms that work in non-oblivious (or
adaptive) environments. At the end of each round, the learner receives some feedback based on the
loss vector and the action of the learner. The goal of the learner is pick its actions so as to minimize
the losses it accumulates by the end of the $T$th round. This setup generalizes the setting of online
combinatorial optimization considered by Cesa-Bianchi and Lugosi [5], Audibert et al. [1], where
the decision set is assumed to be fixed throughout the learning procedure. The interaction protocol
is summarized on Figure 1 for reference.

While not explicitly proven by Kanade et al. [13], their technique can be extended to work in the restricted
setting, where it can be shown to guarantee a regret of $O(T^{3/4})$.
The performance of the learner is measured with respect to the best fixed ranking \( \pi \) choice function as a formal treatment of the problem, let us fix any component \( i \) of the loss vector \( \ell_t \) and get some feedback:

(a) in the full information setting, the learner observes \( \ell_t \),
(b) in the restricted setting, the learner observes \( \ell_{t,i} \) for all \( i \in D_t \),
(c) in the semi-bandit setting, the learner observes \( \ell_{t,i} \) for all \( i \) such that \( V_{t,i} = 1 \).

We distinguish between three different feedback schemes, the simplest one being the full information scheme where the loss vectors are completely revealed to the learner at the end of each round. In the restricted-information scheme, we make a much milder assumption that the learner is informed about the losses of the available actions. Precisely, we define the set of available components as

\[ D_t = \{ i \in [d] : \exists v \in S_t : v_i = 1 \} \]

and assume that the learner can observe the \( i \)-th component of the loss vector \( \ell_t \) if and only if \( i \in D_t \). This is a sensible assumption in a number of practical applications, e.g., in sequential routing problems where components are associated with links in a network, or in online advertising settings where components of the decision vectors represent customer-ad allocations. Finally, in the semi-bandit scheme, we assume that the learner only observes losses associated with the components of its own decision, that is, the feedback is \( \ell_{t,i} \) for all \( i \) such that \( V_{t,i} = 1 \). The observation history \( F_t \) is defined as the sigma-algebra generated by the actions chosen by the learner and the observations handed out by the environment by the end of round \( t \).

The performance of the learner is measured with respect to the best fixed policy (otherwise known as a choice function in discrete choice theory [16]) of the form \( \pi : 2^S \to \hat{S} \). In words, a policy \( \pi \) will pick action \( \pi(\hat{S}) \in \hat{S} \) whenever the environment selects action set \( \hat{S} \). The (total expected) regret of the learner is defined as

\[ R_T = \max_\pi \sum_{t=1}^T \mathbb{E} \left[ (V_t - \pi(S_t))^\top \ell_t \right]. \tag{1} \]

Note that the above expectation integrates over both the randomness injected by the learner and the stochastic process generating the decision sets. The attentive reader might notice that this regret criterion is very similar to that of Kanade et al. [13], who study the setting of prediction with expert advice (where \( S = \Delta_{[d]} \)) and measure regret against the best fixed ranking of experts.

### 3 Loss estimation by Counting Asleep Times

In this section, we describe our method used for estimating unobserved losses that works without having to explicitly learn the availability distribution \( P \). To explain the concept on a high level, let us now consider our simpler partial-observability setting, the restricted-information setting. For the formal treatment of the problem, let us fix any component \( i \in [d] \) and define \( A_{t,i} = \mathbb{1}_{\{i \in D_t\}} \) and \( a_i = \mathbb{E} [A_{t,i} | F_{t-1}] \). Had we known the observation probability \( a_i \), we would be able to estimate the \( i \)-th component of the loss vector \( \ell_t \) by \( \hat{\ell}_{t,i} = (\ell_{t,i} A_{t,i}) / a_i \), as the quantity \( \ell_{t,i} A_{t,i} \) is observable.

It is easy to see that the estimate \( \hat{\ell}_{t,i} \) is unbiased by definition — but, unfortunately, we do not know \( a_i \), so we have no hope to compute it. A simple idea used by Kanade et al. [13] is to devote the first \( T_0 \) rounds of interaction solely to the purpose of estimating \( a_i \) by the sample mean \( \hat{a}_i = \)
\[
(\sum_{t=1}^{T_0} A_{t,i})/T_0. \text{ While this trick gets the job done, it is obviously wasteful as we have to throw away all loss observations from before the estimates are sufficiently concentrated.}^2
\]

We take a much simpler approach based on the simple observation that the “asleep-time” of component \(i\) is a geometrically distributed random variable with parameter \(a_i\). The asleep-time of component \(i\) starting from time \(t\) is formally defined as

\[
N_{t,i} = \min\{n > 0 : i \in D_{t+n}\},
\]

which is the number of rounds until the next observation of the loss associated with component \(i\).

Using the above definition, we construct our loss estimates as the vector \(\hat{\ell}_t, i\) whose \(i\)-th component is

\[
\hat{\ell}_{t,i} = \ell_{t,i} A_{t,i} N_{t,i}. \tag{2}
\]

It is easy to see that the above loss estimates are unbiased as

\[
E[\ell_{t,i} A_{t,i} N_{t,i} | \mathcal{F}_{t-1}] = E[A_{t,i} | \mathcal{F}_{t-1}] E[N_{t,i} | \mathcal{F}_{t-1}] = \ell_{t,i} a_i \cdot \frac{1}{a_i} = \ell_{t,i}
\]

for any \(i\). We will refer to this loss-estimation method as Counting Asleep Times (CAT).

Looking at the Definition 2, the attentive reader might worry that the vector \(\hat{\ell}_t\) depends on future realizations of the random decision sets and thus can be useless for practical use. However, observe that there is no reason that the learner should use the estimate \(\hat{\ell}_{t,i}\) before component \(i\) wakes up in round \(t + N_{t,i}\) — which is precisely the time when the estimate becomes well-defined. This suggests a very simple implementation of CAT: whenever a component falls asleep, estimate its loss by the last observation from that component! More formally, set

\[
\hat{\ell}_{t,i} = \begin{cases} 
\ell_{t,i}, & \text{if } i \in D_t \\
\hat{\ell}_{t-1,i}, & \text{otherwise}.
\end{cases}
\]

It is easy to see that at the beginning of any round \(t\), the two alternative definitions match for all components \(i \in D_t\). In the next section, we confirm that this requirement is sufficient for running our algorithm.

4 Algorithms & their analyses

For all information settings, we base our learning algorithms on the Follow-the-Perturbed-Leader (FPL) prediction method of Hannan [9], as popularized by Kalai and Vempala [12]. This algorithm works by additively perturbing the total estimated loss of each component, and then running an optimization oracle over the perturbed losses to choose the next action. More precisely, our algorithms maintain the cumulative sum of their loss estimates \(\hat{L}_t = \sum_{s=1}^{t} \hat{\ell}_s\) and pick the action

\[
V_t = \arg \min_{v \in S_t} v^\top \left( \eta \hat{L}_{t-1} - Z_t \right),
\]

where \(Z_t\) is a perturbation vector with independent exponentially distributed components with unit expectation, generated independently of the history, and \(\eta > 0\) is a parameter of the algorithm. Our algorithms for the different information settings will be instances of FPL that employ different loss estimates suitable for the respective settings. In the first part of this section, we present the main tools of analysis that will be used for each resulting method.

As usual for analyzing FPL-based methods [12, 10, 18], we start by defining a hypothetical forecaster that uses a time-independent perturbation vector \(\tilde{Z}\) with standard exponential components and peaks one step into the future. However, we need an extra trick to deal with the randomness of the decision set: we introduce the time-independent decision set \(\tilde{S} \sim \mathcal{P}\) (drawn independently of the filtration \((\mathcal{F}_t)_t\)) and define

\[
\tilde{V}_t = \arg \min_{v \in \tilde{S}} v^\top \left( \eta \hat{L}_t - \tilde{Z} \right).
\]

\(^2\)Notice that we require “sufficient concentration” from \(1/\hat{a}_i\) and not only from \(\hat{a}_i\)! The deviation of such quantities is rather difficult to control, as demonstrated by the complicated analysis of Kanade et al. [13].
Clearly, this forecaster is infeasible as it uses observations from the future. Also observe that $	ilde{V}_{t-1} \sim V_t$ given $\mathcal{F}_{t-1}$. The following two lemmas show how analyzing this forecaster can help in establishing the performance of our actual algorithm.

**Lemma 1.** For any sequence of loss estimates, the expected regret of the hypothetical forecaster against any fixed policy $\pi : 2^S \to S$ satisfies

$$
E \left[ \sum_{t=1}^{T} (\tilde{V}_t - \pi(\tilde{S}))^\top \hat{\ell}_t \right] \leq \frac{m (\log d + 1)}{\eta}.
$$

The statement is easily proven by applying the follow-the-leader/be-the-leader lemma (see, e.g., [4, Lemma 3.1]) to the loss sequence $(\hat{\ell}_1 - \tilde{Z}, \hat{\ell}_2, \ldots, \hat{\ell}_T)$, reorganizing, and using the upper bound $E[\|Z\|_\infty] \leq \log d + 1$.

The following result can be extracted from the proof of Theorem 1 of Neu and Bartók [18].

**Lemma 2.** For any sequence of nonnegative loss estimates,

$$
E \left[ (\tilde{V}_t - \ell_t) \hat{\ell}_t \right] \leq \eta E \left[ (\tilde{V}_t \hat{\ell}_t)^2 \right].
$$

In the next subsections, we apply these results to obtain bounds for the three information settings.

### 4.1 Algorithm for full information

In the simplest setting, we can use $\hat{\ell}_t = \ell_t$, which yields the following theorem:

**Theorem 1.** Define

$$
L^*_T = \max \left\{ \min \mathbb{E} \left[ \sum_{t=1}^{T} \pi(S_t)^\top \ell_t \right] , 4(\log d + 1) \right\}.
$$

Setting $\eta = \sqrt{(\log d + 1)/L^*_T}$, the regret of FPL in the full information scheme satisfies

$$
R_T \leq 2m \sqrt{2L^*_T (\log d + 1)}.
$$

As this result is comparable to the best available bounds for FPL [10, 18] in the full information setting and a fixed decision set, it reinforces the observation of Kanade et al. [13], who show that the sleeping experts problem with full information and stochastic availability is no more difficult than the standard experts problem. The proof of Theorem 1 follows directly from combining Lemmas 1 and 2 and some standard tricks. For completeness, details are provided in Appendix A.

### 4.2 Algorithm for restricted feedback

In this section, we use the CAT loss estimate defined in Equation 2 as $\hat{\ell}_t$ in FPL, and call the resulting method SLEEPINGCAT. The following theorem gives the performance guarantee for this algorithm.

**Theorem 2.** Define $Q_t = \sum_{i=1}^{d} \mathbb{E} [V_{t,i} | i \in D_t]$. The total expected regret of SLEEPINGCAT against the best fixed policy is upper bounded as

$$
R_T \leq \frac{m(\log d + 1)}{\eta} + 2\eta m \sum_{t=1}^{T} Q_t.
$$

**Proof sketch of Theorem 2.** We start by observing $\mathbb{E} \left[ \pi(\tilde{S})^\top \hat{\ell}_t \right] = \mathbb{E} \left[ \pi(S_t)^\top \ell_t \right]$, where we used that $\hat{\ell}_t$ is independent of $\tilde{S}$ and is an unbiased estimate of $\ell_t$, and also that $S_t \sim \tilde{S}$. The proof is completed by combining this with Lemmas 1 and 2, and the bound

$$
\mathbb{E} \left[ (\tilde{V}_{t-1} \hat{\ell}_t)^2 \right] \leq 2m Q_t.
$$

This lemma can be proven in the current case by virtue of the fixed decision set $\tilde{S}$, allowing the necessary recursion steps to go through.
The proof of this last statement follows from a tedious calculation that we defer to Appendix B.

Below, we provide two ways of further bounding the regret under various assumptions. The first one provides a universal upper bound that holds without any further assumptions.

**Corollary 1.** Setting \( \eta = \sqrt{(\log d + 1)/(2dT)} \), the regret of SLEEPINGCat against the best fixed policy is bounded as

\[
R_T \leq 2m \sqrt{d \log d + 1}.
\]

The proof follows from the fact that \( Q_t \leq d \) no matter what \( \mathcal{P} \) is. A somewhat surprising feature of our bound is its scaling with \( \sqrt{d \log d} \), which is much worse than the logarithmic dependence exhibited in the full information case. It is easy to see, however, that this bound is not improvable in general – see Appendix D for a simple example. The next bound, however, shows that it is possible to improve this bound by assuming that most components are reliable in some sense, which should be the case in practice.

**Corollary 2.** Assuming \( \alpha_i \geq \beta \) for all \( i \), we have \( Q_t \leq 1/\beta \), and setting \( \eta = \sqrt{\beta (\log d + 1)/(2T)} \) guarantees that the regret of SLEEPINGCat against the best fixed policy is bounded as

\[
R_T \leq 2m \sqrt{\frac{2T(\log d + 1)}{\beta}}.
\]

### 4.3 Algorithm for semi-bandit feedback

In this section, we turn our attention to the problem of learning with semi-bandit feedback where the learner only gets to observe the losses associated with its own decision. Specifically, we assume that the learner observes all components \( i \) of the loss vector such that \( V_{t,i} = 1 \). The extra difficulty in this setting is that our actions influence the feedback that we receive, so we have to be more careful when defining our loss estimates. Ideally, we would like to work with unbiased estimates of the form

\[
\hat{\ell}_{t,i} = \frac{\ell_{t,i}}{q^*_{t,i}} V_{t,i}, \quad \text{where} \quad q^*_{t,i} = \mathbb{E} [V_{t,i} | \mathcal{F}_{t-1}] = \sum_{\mathcal{S} \in 2^d} \mathcal{P}(\mathcal{S}) \mathbb{E} [V_{t,i} | \mathcal{F}_{t-1}, \mathcal{S}_t = \mathcal{S}]. \tag{3}
\]

for all \( i \in [d] \). Unfortunately though, we are in no position to compute these estimates, as this would require perfect knowledge of the availability distribution \( \mathcal{P} \)! Thus we have to look for another way to compute reliable loss estimates. A possible idea is to use

\[ q_{t,i} \cdot a_i = \mathbb{E} [V_{t,i} | \mathcal{F}_{t-1}, \mathcal{S}_t] \cdot \mathbb{P} \{ i \in D_t \}. \]

instead of \( q^*_{t,i} \) in Equation 3 to normalize the observed losses. This choice yields another unbiased loss estimate as

\[
\mathbb{E} \left[ \frac{\hat{\ell}_{t,i} V_{t,i}}{a_i} \bigg| \mathcal{F}_{t-1} \right] = \frac{\ell_{t,i}}{a_i} \mathbb{E} \left[ \frac{V_{t,i}}{q_{t,i}} \bigg| \mathcal{F}_{t-1}, \mathcal{S}_t \right] \mathcal{F}_{t-1} = \frac{\ell_{t,i}}{a_i} \mathbb{E} [A_{t,i} | \mathcal{F}_{t-1}] = \ell_{t,i}, \tag{4}
\]

which leaves us with the problem of computing \( q_{t,i} \) and \( a_i \). While this also seems to be a tough challenge, we now show to compute something very close to the above estimates.

Besides our trick used for estimating the \( 1/a_i \)'s by the random variables \( N_{t,i} \), we now also have to face the problem of not being able to find a closed-form expression for the \( q_{t,i} \)'s. We follow the geometric resampling approach of Neu and Bartók [18] and draw an additional sequence of \( M \) perturbation vectors \( Z_l'(1), \ldots, Z_l'(M) \) and use them to define

\[
V'_l(k) = \arg\min_{v \in \mathcal{S}_t} \left\{ \hat{\ell}_{t-1} - Z'_l(k) \right\}
\]

for all \( k \in [M] \). Using these simulated predictions, we define

\[
K_{t,i} = \min \left\{ \{k \in [M] : V'_{t,i}(k) = V_{t,i}\} \cup \{M\} \right\}.
\]

and

\[
\hat{\ell}_{t,i} = \ell_{t,i} K_{t,i} N_{t,i} V_{t,i} \tag{5}
\]
for all $i$. Setting $M = \infty$ makes this expression equivalent to $\frac{\ell_{t,i}V_{t,i}}{q_{t,i}a_i}$ in expectation, yielding yet another unbiased estimator for the losses. Our analysis, however, crucially relies on setting $M$ to a finite value so as to control the variance of the loss estimates. We are not aware of any other work that achieves a similar variance-reduction effect without explicitly exploring the action space \cite{17, 6, 5, 3}, making this alternative bias-variance tradeoff a unique feature of our analysis. We call the algorithm resulting from using the loss estimates above SLEEPINGCatBandit. The following theorem gives the performance guarantee for this algorithm.

**Theorem 3.** Define $Q_t = \sum_{i=1}^d \mathbb{E}[V_{t,i} | i \in D_t]$. The total expected regret of SLEEPINGCatBandit against the best fixed policy is bounded as

$$R_T \leq \frac{m(\log d + 1)}{\eta} + 2\eta M \sum_{t=1}^T Q_t + \frac{dT}{eM}.$$ 

Proof of Theorem 3. First, observe that $\mathbb{E}[\hat{\ell}_{t,i} | \mathcal{F}_{t-1}] \leq \ell_{t,i}$ as $\mathbb{E}[K_{t,i}N_{t,i} | \mathcal{F}_{t-1}] \leq 1/(q_{t,i}a_i)$ by definition. Thus, we can get $\mathbb{E}[\pi(S)^T\hat{\ell}_t] \leq \mathbb{E}[\pi(S_t)^T\ell_t]$ by a similar argument that we used in the proof of Theorem 2. After yet another long and tedious calculation (see Appendix C), we can prove

$$\mathbb{E}\left[\left(V_{t-1}^T\hat{\ell}_t\right)^2 | \mathcal{F}_{t-1}\right] \leq 2MmQ_t.$$ 

The proof is concluded by combining this bound with Lemmas 1 and 2 and the upper bound

$$\mathbb{E}[V_t^T\ell_t | \mathcal{F}_{t-1}] \leq \mathbb{E}\left[V_{t-1}^T\hat{\ell}_t | \mathcal{F}_{t-1}\right] + \frac{d}{eM},$$

which can be proven by directly following the techniques of Neu and Bartók \cite{18}.

**Corollary 3.** Setting $\eta = \left(\sqrt{m(\log d + 1)}/2dT\right)^{2/3}$ and $M = \frac{1}{\sqrt{m}} \cdot \left(\frac{dT}{\sqrt{2m(\log d + 1)}}\right)^{1/3}$ guarantees that the regret of SLEEPINGCatBandit against the best fixed policy is bounded as

$$R_T \leq \left(2mdT\right)^{2/3} \cdot (\log d + 1)^{1/3}.$$ 

The proof of the corollary follows from bounding $Q_t \leq d$ and plugging the parameters into the bound of Theorem 3.

This corollary implies that SLEEPINGCatBandit achieves a regret of $(2KT)^{2/3} \cdot (\log K + 1)^{1/3}$ in the case when $S = [K]$, that is, in the $K$-armed sleeping bandit problem considered by Kanade et al. \cite{13}. This improves their bound of $O((KT)^{4/5} \log T)$ by a large margin, thanks to the fact that we did not have to explicitly learn the distribution $\mathcal{P}$.

Still, one might ask if it is possible to achieve a regret of order $\sqrt{T}$ in the semi-bandit setting. While the EXP4 algorithm of Auer et al. \cite{2} can be used to obtain such regret guarantee, running this algorithm is out of question as its time and space complexity can be double-exponential in $d$ (see also the comments in \cite{15}). Had we had access to the loss estimates \eqref{eq:loss_estimates}, we would be able to control the regret of FPL as the term on the right hand side of Lemma 1 could be bounded by $md$, which is sufficient for obtaining a regret bound of $O(m\sqrt{dT \log d})$. Unfortunately, we cannot achieve similar results even with the information-theoretically feasible estimate of Equation 4. This obstacle is present in the simple multi-armed bandit case, too. We conclude that learning in the bandit setting requires significantly more knowledge about $\mathcal{P}$ than the knowledge of the $a_i$’s. The question if we can extend the CAT technique to estimate all the relevant quantities of $\mathcal{P}$ is an interesting problem left for future investigation.

Finally, we note that similarly to the improvement of Corollary 2, it is possible to replace the factor $d^{2/3}$ by $(d/\beta)^{1/3}$ if we assume that $a_i \geq \beta$ for all $i$ and some $\beta > 0$.

5 **Experiments**

In this section we present the empirical evaluation of our algorithms for bandit and semi-bandit settings, and compare them to its counterparts \cite{13}. We demonstrate that the wasteful exploration of BSFPL does not only result in worse regret bounds but also degrades its empirical performance.
For the bandit case, we evaluate SLEEPINGCATBANDIT using the same setting as Kanade et al. [13]. We consider an experiment with $T = 10,000$ and 5 arms, each of which are available independently of each other with probability $p$. Losses for each arm are constructed as random walks with Gaussian increments of standard deviation 0.002, initialized uniformly on $[0, 1]$. Losses outside $[0, 1]$ are truncated. In our first experiment (Figure 2, left), we study the effect of changing $p$ on the performance of BSFPL and SLEEPINGCATBANDIT. Notice that when $p$ is very low, there are few or no arms to choose from. In this case the problems are easy by design and all algorithms suffer low regret. As $p$ increases, the policy space starts to blow up and the problem becomes more difficult. When $p$ approaches one, it collapses into the set of single arms and the problem gets easier again. Observe that the behavior of SLEEPINGCATBANDIT follows this trend. On the other hand, the performance of BSFPL steadily decreases with increasing availability. This is due to the explicit exploration rounds in the main phase of BSFPL, that suffers the loss of the uniform policy scaled by the exploration probability. The performance of the uniform policy is plotted for reference.

To evaluate SLEEPINGCATBANDIT in the semi-bandit setting, we consider the shortest path problem on grids of $3 \times 3$ and $10 \times 10$ nodes, which amounts to 12 and 180 edges respectively. For each edge, we generate a random-walk loss sequence in the same way as in our first experiment. In each round $t$, the learner has to choose a path from the lower left corner to the upper right one composed from available edges. The individual availability of each edge is sampled with probability 0.9, independent of the others. Whenever an edge gets disconnected from the source, it becomes unavailable itself, resulting in a quite complicated action-availability distribution. Once a learner chooses a path, the losses of chosen road segments are revealed and the learner suffers their sum.

Since [13] does not provide a combinatorial version of their approach, we compare against COMBSFPL, a straightforward extension of BSFPL. As in BSFPL, we dedicate an initial phase to estimate the availabilities of each component, requiring $d$ oracle calls per step. In the main phase, we follow BSFPL and alternate between exploration and exploitation. In exploration rounds, we test for the reachability of a randomly sampled edge and update the reward estimates as in BSFPL.

Figure 2 (middle and right) shows the performance of COMBSFPL and SLEEPINGCATBANDIT for a fixed loss sequence, averaged over 20 samples of the component availabilities. We also plot the performance of a random policy that follows the perturbed leader with all-zero loss estimates. First observe that the initial exploration phase sets back the performance of COMBSFPL significantly. The second drawback of COMBSFPL is the explicit separation of exploration and the exploitation rounds. This drawback is far more apparent when the number of components increases, as it is the case for the $10 \times 10$ grid graph with 180 components. As COMBSFPL only estimates the loss of one edge per exploration step, sampling each edge as few as 50 times eats up 9,000 rounds from the available 10,000. SLEEPINGCATBANDIT doesn’t suffer from this problem as it uses all its observations in constructing the loss estimates.

Our experiments give evidence that the implicit exploration of SLEEPINGCATBANDIT results in significantly better performance than the approach of [13], which suffers both from the initial exploration phase and from the explicit exploration rounds in the main phase.
References


A Proof of Theorem 1

We begin by applying Lemma 1, exploiting that \( \hat{\ell}_t = \ell_t \) and \( \bar{S} \) is an independent copy of \( S_t \):

\[
\sum_{t=1}^{T} \mathbb{E} \left[ \tilde{V}_t^\top \ell_t \right] - \sum_{t=1}^{T} \mathbb{E} \left[ \pi(S_t)^\top \ell_t \right] = \mathbb{E} \left[ \sum_{t=1}^{T} \left( \tilde{V}_t - \pi(\bar{S}) \right)^\top \hat{\ell}_t \right] \leq \frac{m (\log d + 1)}{\eta}
\]

Next, we apply Lemma 2 to obtain

\[
\mathbb{E} \left[ (V_t - \tilde{V}_t)^\top \hat{\ell}_t \right] \leq \eta \mathbb{E} \left[ (\tilde{V}_t^\top \hat{\ell}_t)^2 \right] \leq \eta m \mathbb{E} [V_t^\top \ell_t],
\]

where we used that \( V_{t-1} \sim V_t \) and \( \tilde{V}_{t-1}^\top \ell_t \leq m \). Introducing the notation

\[
C_T = \sum_{t=1}^{T} \mathbb{E} [V_t^\top \ell_t],
\]

we get by combining the above bounds that

\[
C_T - L_T^* \leq \frac{m (\log d + 1)}{\eta} + \eta m C_T.
\]

After reordering, we get

\[
C_T - L_T^* \leq \frac{1}{1 - m \eta} \left( \frac{m (\log d + 1)}{\eta} + \eta m L_T^* \right).
\]

The bound follows from plugging in the choice of \( \eta \) and observing that \( 1 - m \eta \geq 1/2 \) holds by the assumption of the theorem.

B Proof details for Theorem 2

In the restricted information case, the term on the right hand side of the bound of Lemma 2 can be upperbounded as follows:

\[
\mathbb{E} \left[ \left( \tilde{V}_{t-1}^\top \hat{\ell}_t \right)^2 \right] = \mathbb{E} \left[ \sum_{j=1}^{d} \sum_{k=1}^{d} \left( \tilde{V}_{t-1,j} \hat{\ell}_{t,j} \right) \left( \tilde{V}_{t-1,k} \hat{\ell}_{t,k} \right) \bigg| \mathcal{F}_{t-1} \right]
\]

\[
\leq \mathbb{E} \left[ \sum_{j=1}^{d} \sum_{k=1}^{d} \frac{N_{t,j}^2 + N_{t,k}^2}{2} \left( \tilde{V}_{t-1,j} A_{t,j} \hat{\ell}_{t,j} \right) \left( \tilde{V}_{t-1,k} A_{t,k} \hat{\ell}_{t,k} \right) \bigg| \mathcal{F}_{t-1} \right]
\]

(using the definition of \( \hat{\ell}_t \) and \( 2AB \leq A^2 + B^2 \))

\[
= \mathbb{E} \left[ \sum_{j=1}^{d} \sum_{k=1}^{d} N_{t,j}^2 \left( \tilde{V}_{t-1,j} A_{t,j} \hat{\ell}_{t,j} \right) \left( \tilde{V}_{t-1,k} A_{t,k} \hat{\ell}_{t,k} \right) \bigg| \mathcal{F}_{t-1} \right]
\]

(by symmetry)

\[
\leq 2 \mathbb{E} \left[ \sum_{j=1}^{d} \frac{1}{a_j} \left( \tilde{V}_{t-1,j} A_{t,j} \hat{\ell}_{t,j} \right) \sum_{k=1}^{d} \tilde{V}_{t-1,k} \hat{\ell}_{t,k} \bigg| \mathcal{F}_{t-1} \right]
\]

(using \( \mathbb{E} \left[ N_{t,j}^2 \big| \mathcal{F}_{t-1} \right] = (2 - a_j)/a_j^2 \leq 2/a_j^2 \))

\[
= 2m \mathbb{E} \left[ \sum_{j=1}^{d} \frac{1}{a_j} \left( \tilde{V}_{t-1,j} \hat{\ell}_{t,j} \right) \bigg| \mathcal{F}_{t-1} \right]
\]

(using \( \| \tilde{V}_t \|_1 \leq m \) and \( \mathbb{E} \left[ A_{t,j} \big| \mathcal{F}_{t-1} \right] = a_j \))

\[
\leq 2m \sum_{j=1}^{d} \mathbb{E} \left[ \tilde{V}_{t,j} \big| j \in D_t, \mathcal{F}_{t-1} \right],
\]

where in the last line, we used that \( \tilde{V}_{t-1} \) is identically distributed as \( V_t \).
C Proof details for Theorem 3

In the semi-bandit case, the term on the right hand side of the bound of Lemma 2 can be upper-bounded as follows:

\[ E \left[ \left( \tilde{V}_{t-1}^{\prime} \tilde{\ell}_t \right)^2 \right] \leq E \left[ \sum_{j=1}^{d} \sum_{k=1}^{d} \frac{K_{t,j}^2 + K_{t,k}^2}{2} \cdot N_{t,j}^2 + N_{t,k}^2 \left( \tilde{V}_{t-1,j} V_{t,j} \tilde{\ell}_{t,j} \right) \left( \tilde{V}_{t-1,k} V_{t,k} \tilde{\ell}_{t,k} \right) \right] \]

(\text{using the definition of } \tilde{\ell}_t \text{ and } 2AB \leq A^2 + B^2)

\[ = E \left[ \sum_{j=1}^{d} \sum_{k=1}^{d} \frac{K_{t,j}^2 N_{t,j}^2 + K_{t,k}^2 N_{t,k}^2 + K_{t,j}^2 N_{t,k}^2 + K_{t,k}^2 N_{t,j}^2}{4} \left( \tilde{V}_{t-1,j} V_{t,j} \tilde{\ell}_{t,j} \right) \left( \tilde{V}_{t-1,k} V_{t,k} \tilde{\ell}_{t,k} \right) \right] \]

\[ \leq \frac{m}{2} \cdot E \left[ \sum_{j=1}^{d} M K_{t,j} N_{t,j}^2 \left( \tilde{V}_{t-1,j} V_{t,j} \tilde{\ell}_{t,j} \right) \right] \]

(\text{using } K_{t,j} \leq M \text{ and } V_{t,k} \leq A_{t,k} \text{ and } \| \tilde{V}_t \|_1 \leq m)

\[ \leq \frac{m}{2} \cdot E \left[ \sum_{j=1}^{d} M N_{t,j}^2 \left( \tilde{V}_{t-1,j} A_{t,j} \tilde{\ell}_{t,j} \right) \right] \]

(\text{using } E[ K_{t,j} V_{t,j} | F_{t-1}, S_t ] \leq A_{t,j} \text{ by definition of } K_{t,j} \text{ and independence of } K_{t,j} \text{ and } V_{t,j})

\[ \leq 2MmE \left[ \sum_{j=1}^{d} \frac{1}{a_j} \left( \tilde{V}_{t-1,j} A_{t,j} \tilde{\ell}_{t,j} \right) \right] \]

(\text{using } \| \tilde{V}_t \|_1 \leq m \text{ and } E[ N_{t,j}^2 | F_{t-1} ] = (2 - a_j)/a_j^2 \leq 2/a_j^2)

\[ = 2MmE \left[ \sum_{j=1}^{d} \frac{1}{a_j} \left( \tilde{V}_{t-1,j} \ell_{t,j} \right) \right] \]

\[ \leq 2Mm \sum_{j=1}^{d} E[ V_{t,j} | j \in D_t, F_{t-1} ], \]

where in the last line, we used that \( \tilde{V}_{t-1} \) is identically distributed as \( V_t \).
Consider a sleeping experts problem with \(d\) experts with loss sequence \((\ell_t)_t\). In each round \(t = 1, 2, \ldots, T\) the learner picks \(I_t\). For simplicity, assume that \(d\) is even and let \(N = d/2\). Let \(\mathcal{P}\) be such that it assigns a probability of \(1/N\) to each pair \((2i-1, 2i)\) of experts, that is, only two experts are awake at each time. The regret of any learning algorithm in this problem can be written as

\[
R_T = \max_\pi \mathbb{E}\left[ \sum_{t=1}^T (\ell_t, I_t - \ell_t, \pi(S_t)) \right] = \max_\pi \mathbb{E}\left[ \sum_{j=1}^N \sum_{t=1}^T \mathbb{1}_{\{2i \in S_t\}} (\ell_t, I_t - \ell_t, \pi(S_t)) \right].
\]

We now define \(N\) full-information games \(G_1, \ldots, G_N\) with two experts each as follows: In game \(G_i\), the number of rounds is \(T_i = \sum_{t=1}^T \mathbb{1}_{\{2i \in S_t\}}\), the decision of the learner in round \(t\) is \(J_t\) and the sequence of loss functions is \((\ell_t(i))_t\) so that the regret in game \(G_i\) is defined as

\[
R_T(i) = \max_j \sum_{t=1}^{T_i} (\ell_t, J_t(i) - \ell_t, j(i)).
\]

It is well-known (e.g., [4, Section 3.7]) that there exists a distribution of losses that guarantees that \(\mathbb{E}[R_T(i)] \geq c\sqrt{T_i}\) for some constant \(c > 0\), no matter what algorithm the learner uses. The result follows from observing that there exists a mapping between the full-information games \(G_1, \ldots, G_N\) and our original problem such that

\[
\mathbb{E}\left[ \sum_{j=1}^N \sum_{t=1}^T \mathbb{1}_{\{2i \in S_t\}} (\ell_t, I_t - \ell_t, \pi(S_t)) \right] = \sum_{i=1}^N \mathbb{E}[R_T(i)] \geq c \sum_{i=1}^N \sqrt{T_i}.
\]

That is, we get that

\[
R_T \geq c \sum_{i=1}^N \mathbb{E}\left[ \sqrt{T_i} \right].
\]

As \(\lim_{T \to \infty} \frac{T}{T_i} = N\) holds almost surely, we get that

\[
\lim_{T \to \infty} \frac{R_T}{\sqrt{TN}} \geq c,
\]

showing that there exist sleeping experts problems for any even \(d\) where the guarantees of Corollary 1 cannot be improved asymptotically.