

---

# Combinatorial Bandits

---

Nicolò Cesa-Bianchi \*

Università degli Studi di Milano, Italy  
cesa-bianchi@dsi.unimi.it

Gábor Lugosi†

ICREA and Pompeu Fabra University  
lugosi@upf.es

## Abstract

We study sequential prediction problems in which, at each time instance, the forecaster chooses a binary vector from a certain fixed set  $\mathcal{S} \subseteq \{0, 1\}^d$  and suffers a loss that is the sum of the losses of those vector components that equal to one. The goal of the forecaster is to achieve that, in the long run, the accumulated loss is not much larger than that of the best possible vector in the class. We consider the “bandit” setting in which the forecaster has only access to the losses of the chosen vectors. We introduce a new general forecaster achieving a regret bound that, for a variety of concrete choices of  $\mathcal{S}$ , is of order  $\sqrt{nd \ln |\mathcal{S}|}$  where  $n$  is the time horizon. This is not improvable in general and is better than previously known bounds. We also point out that computationally efficient implementations for various interesting choices of  $\mathcal{S}$  exist.

## 1 Introduction

Consider a sequential prediction problem in which a forecaster is to choose, at every time instance  $t = 1, \dots, n$ , an element from a set  $\mathcal{S}$  of  $N$  actions. After making a choice, the forecaster suffers a loss corresponding to the chosen action. The goal of the forecaster is to achieve that the accumulated loss is not much larger than that of the best possible fixed action, chosen in hindsight. The difference between the achieved and optimal cumulative losses is called the *regret*. It is well known (see [6] for a survey) that randomized prediction strategies exist that guarantee that the expected regret of the forecaster is bounded by a constant times  $\sqrt{n \ln N}$ , regardless of the sequence of losses, as long as they are bounded. The logarithmic dependence on the number of actions allows one to compete with very large classes of actions (also called experts). However, large classes of experts raise nontrivial computational issues. The construction of computationally efficient forecasters for various cases of

---

\*The authors gratefully acknowledge partial support by the PASCAL2 Network of Excellence under EC grant no. 216886. This publication only reflects the authors’ views.

†Supported by the Spanish Ministry of Science and Technology grant MTM2006-05650.

structured classes of experts is a thoroughly studied problem. Once again, we refer to [6] for a survey.

An interesting variant of the sequential prediction problem is the *adversarial multi-armed bandit* problem in which the forecaster only observes the loss of the chosen action and uses the randomized choices to gather information. It was shown by Auer et al. [2] that an expected regret of the order of  $\sqrt{nN \ln N}$  is achievable in this case. There has been a flurry of activity to address versions of the adversarial bandit problem for large and structured classes of experts, see Awerbuch and Kleinberg [3], McMahan and Blum [16], Dani and Hayes [8], György, Linder, Lugosi, and Ottucsák [10], Dani, Hayes, and Kakade [7], Abernethy, Hazan, and Rakhlin [1], Bartlett, Dani, Hayes, Kakade, and Tewari [4]. The efforts have been focused on two main issues: (1) obtaining regret bounds as small as possible; (2) constructing computationally feasible forecasters.

In this paper we propose a new general methodology for the cases when the finite class of experts has a certain combinatorial structure. The obtained regret bounds improve on those derived from general on-line optimization results. We also show that in some interesting cases nontrivial efficient algorithms exist.

The paper is organized as follows. In Section 2 we formulate the problem. In Section 3 we discuss the relationship of our results to earlier work. The general prediction strategy is defined and the main performance bound is established in Section 4. Various applications are described in Section 6, including a multitask bandit problem, learning permutations, learning spanning trees of a complete graph, and learning balanced cut sets.

## 2 Statement of the problem

In the *combinatorial bandit problem* a set  $\mathcal{S} \subseteq \{0, 1\}^d$  of elements  $\mathbf{v}(k)$  for  $k = 1, \dots, N$  is given (this is the set of “experts” or “actions”), and a forecaster plays with the opponent the following repeated game:

For each step  $t = 1, 2, \dots$

1. The opponent secretly chooses a loss vector  $\ell_t \in [0, 1]^d$
2. The forecaster chooses  $K_t \in \{1, \dots, N\}$
3. The cost  $\ell_t(K_t) = \ell_t^\top \mathbf{v}(K_t)$  is announced to the forecaster.

The forecaster’s goal is to control the *regret*

$$\widehat{L}_n - \min_{k=1,\dots,N} L_n(k) = \sum_{t=1}^n \ell_t(K_t) - \min_{k=1,\dots,N} \sum_{t=1}^n \ell_t^\top v(k).$$

The forecaster is allowed to use randomization. More precisely, at every time instance  $t$ , the forecaster chooses a distribution  $p_{t-1}(1), \dots, p_{t-1}(N)$  over the set  $\{1, \dots, N\}$  (i.e.,  $p_{t-1}(k) \geq 0$  for all  $k = 1, \dots, N$  and  $\sum_{k=1}^N p_{t-1}(k) = 1$ ) and draws an index  $K_t = k$  with probability  $p_{t-1}(k)$ . Thus, the regret is a random variable. In this paper we investigate the behavior of the *expected regret*

$$\max_{k=1,\dots,N} \mathbb{E} \left[ \widehat{L}_n - L_n(k) \right]$$

where the expectation is with respect to the forecaster’s internal randomization.

The most important parameters of the problem are the time horizon  $n$ , the dimension  $d$ , the cardinality  $N$  and the maximum “weight” of any expert  $B = \max_{v \in \mathcal{S}} \sum_{i=1}^d v_i$ . Note that  $B$  is an upper bound for the loss  $\ell_t(k) = \ell_t^\top v(k)$  of any expert.

We note here that the combinatorial bandit problem is a special case of the online linear optimization problem considered, for example, in [1, 4, 7]. In the online linear optimization problem the expert class is an arbitrary (finite) subset  $\mathcal{S}$  of  $\mathbb{R}^d$  and the loss of an expert  $v \in \mathcal{S}$ , at time  $t$ , is the inner product  $\ell_t^\top v$ . The fact that we restrict  $\mathcal{S}$  to be a subset of the binary hypercube  $\{0, 1\}^d$  allows us to exploit the combinatorial structure of the class of experts in a transparent way. Arguably, the most interesting examples of online linear optimization fit in the present framework.

### 3 Relation to previous work

When  $d = N$  and  $v(1), \dots, v(N)$  are the standard basis vectors, then the model is identical to the adversarial bandit problem introduced by Auer et al. [2], who proved a regret bound of the order of  $\sqrt{nN \ln N}$  that holds not only in expectation but also with high probability. A well-studied instance of our general framework is the *path planning* problem, in which  $d$  is the number of edges of a fixed graph and  $v(1), \dots, v(N)$  represent all paths between two fixed vertices of the graph. At each time instance the forecaster chooses a path and suffers a loss that is the sum of the losses over the individual edges of the chosen path. Takimoto and Warmuth [19] and Kalai and Vempala [14] exhibit computationally efficient forecasters in the “full-information” case, that is, when the forecaster has access to the losses over every edge of the graph.

The partial information setting considered in this paper was first studied by Awerbuch and Kleinberg [3] who proved a regret bound of the order  $n^{2/3}$  for the restricted model of oblivious opponent. McMahan and Blum [16], achieved a regret bound of the order of  $n^{3/4}$  for the general model.

Both [3] and [16] study the somewhat more general framework of *online linear optimization*, introduced by Kalai and Vempala [14]. György et al. [10] considered the problem of path planning in a less demanding partial information framework, when the loss of every edge on the chosen path is

revealed to the forecaster. They exhibit a computationally efficient forecaster achieving a regret of order  $B\sqrt{nd \ln N}$  with high probability. Even though [10] only considers the path planning problem, it is not difficult to extend their results to the more general setup of this paper. However, the model considered here, that is, when the forecaster only receives information about the total loss of the chosen object, is more challenging. Dani, Hayes, and Kakade [7] were the first to prove an expected regret bound with the optimal  $\sqrt{n}$  dependence on the time horizon. Their bound is of the form  $Bd\sqrt{n \ln N}$ . Bartlett, Dani, Hayes, Kakade, Rakhlin, and Tewari [4] show that this bound also holds with high probability. The forecaster of [7] is based on exponential weights and can be computed efficiently whenever efficient implementations of the exponentially weighted average forecaster are available. This is certainly possible for the path planning problem, but there are various other interesting examples—see the discussion of the examples in Section 6 below. Abernethy, Hazan, and Rakhlin [1] consider a very different approach which allows one to construct computationally efficient forecasters for a large variety of problems and has an expected regret of the order of  $Bd\sqrt{nd \ln n}$ . This requires the construction of a *self-concordant* function tailored to the problem at hand. Even though the existence of such a function is guaranteed, its construction may be a nontrivial task in some applications.

In this paper we take an approach similar to that of Dani, Hayes, and Kakade [7]. Like [7], we construct unbiased estimates of each loss component  $\ell_{t,i}$ ,  $i = 1, \dots, d$  and define an exponentially weighted average forecaster based on these estimates. The main difference is in the *exploration* part of the algorithm. Following Awerbuch and Kleinberg [3], Dani, Hayes and Kakade construct a *barycentric spanner* of the set  $\mathcal{S}$  and ensure exploration by mixing the exponential weights with the uniform distribution on spanners. Instead, we use a mixing term derived from the uniform distribution over  $\mathcal{S}$ . This allows us to achieve an expected regret bound of the order of  $B\sqrt{nd \ln N}$  whenever the smallest eigenvalue of a certain matrix associated with  $\mathcal{S}$  is not too small. The largest part of our efforts is dedicated to show that this smallest eigenvalue can indeed be handled by exploiting the combinatorial structure of the class of experts in a number of interesting cases. Note that, as shown in [7], the bound  $B\sqrt{nd \ln |\mathcal{S}|}$  is not improvable in general when  $\mathcal{S} \subseteq \{0, 1\}^d$ .

### 4 The forecasting strategy

Our algorithm COMBAND (see figure) maintains a weight vector defined, at each time  $t$ , by  $w_{t,i} = \exp(-\eta \widetilde{L}_{t,i})$  for  $i = 1, \dots, d$ , where  $\widetilde{L}_{t,i} = \widetilde{\ell}_{1,i} + \dots + \widetilde{\ell}_{t,i}$  is a cumulative pseudo-loss (defined below) and  $\eta > 0$  is a fixed parameter. These weights define corresponding weights  $\overline{w}_t \in \mathbb{R}^N$  over the elements of  $\mathcal{S}$  in the obvious way

$$\overline{w}_t(k) = \prod_{i: v_i(k)=1} w_{t,i}.$$

Let  $\overline{W}_t = \sum_{k=1}^N \overline{w}_t(k)$  and let  $q_t(k) = \overline{w}_t(k)/\overline{W}_t$ . At each time  $t$ , COMBAND plays  $v(K_t) \in \mathcal{S}$ , where  $K_t$  is drawn from the distribution  $p_{t-1} = (1 - \gamma)q_{t-1} + \gamma\mu$  on  $1, \dots, N$ .

**Algorithm** COMBAND

**Parameters** Action set  $\mathcal{S} \subseteq \{0, 1\}^d$ , prior distribution  $\mu$  over  $\mathcal{S}$ , mixing coefficient  $\gamma > 0$

**Initialization**  $q_0 =$  uniform distribution on  $\mathcal{S}$

**For**  $t = 1, 2, \dots$

1. Let  $p_{t-1} = (1 - \gamma)q_{t-1} + \gamma\mu$
2. Draw action  $K_t$  from  $p_{t-1}$
3. Incur and observe cost  $\ell_t(K_t) = \ell_t^\top v(K_t)$
4. Let  $P_{t-1} = \mathbb{E}[\mathbf{V}\mathbf{V}^\top]$  where  $\mathbf{V}$  has law  $p_{t-1}$
5. Let  $\tilde{\ell}_t = \ell_t(K_t)P_{t-1}^+ v(K_t)$
6. Update probs.  $q_t(k) \propto q_{t-1}(k) \exp(-\eta \tilde{\ell}_t(k))$  for all  $k = 1, \dots, N$ .

Here  $\mu$  is any prior distribution on  $\{1, \dots, N\}$  and  $\gamma > 0$  is a parameter. In all of our applications we choose  $\mu$  to be the uniform distribution, although in principle different distributions are also allowed. The vector of pseudo-losses  $\tilde{\ell}_t = (\tilde{\ell}_{t,1}, \dots, \tilde{\ell}_{t,d})$  is defined by

$$\tilde{\ell}_t = \ell_t(K_t)P_{t-1}^+ v(K_t) \quad (1)$$

where  $P^+$  is the pseudo-inverse of the  $d \times d$  correlation matrix  $\mathbb{E}[\mathbf{V}\mathbf{V}^\top]$  for  $\mathbf{V} \in \mathcal{S}$  distributed according to  $p_{t-1}$ . (Throughout the paper, we use an index  $k = 1, \dots, N$  and its corresponding element  $v(k) \in \mathcal{S}$  interchangeably.)

As we said in the previous section, COMBAND can be viewed as a variant of the GEOMETRICHEDGE algorithm of Dani, Hayes and Kakade. The only substantial difference is that we perform exploration by drawing actions from a distribution  $\mu$  over the entire set  $\mathcal{S}$  (Step 1 in figure) instead of drawing from a spanner. This fact gives us a finer control on the loss estimates  $\tilde{\ell}_{t,i}$  in which the factor  $\|P_{t-1}^+\|$  occurs—see (1) above. Indeed, while [7] only achieves  $\|P_{t-1}^+\| \leq d/\gamma$  due to the mix of the barycentric spanners in  $P_t$ , we can afford the more detailed bound  $\|P_{t-1}^+\| \leq 1/(\gamma\lambda_{\min})$ , where  $\lambda_{\min}$  is the smallest nonzero eigenvalue of the correlation matrix of the initial sampling distribution  $\mu$ . In concrete cases, the computation of tight lower bounds on  $\lambda_{\min}$  allows us to obtain better regret bounds. An additional feature, exploited to derive an even tighter estimate on  $|\tilde{\ell}_{t,i}|$ —see (4), is that we assume a reference coordinate system for  $\mathcal{S}$  which the definition of  $B = \max_k \|v(k)\|_1$  relies on. In [7], on the contrary, actions were defined in terms of the nonorthogonal basis provided by the spanner, and this is why the corresponding bound on  $|\tilde{\ell}_{t,i}|$  is worse.

For COMBAND we prove the following regret bound.

**Theorem 1** *Let  $\mathcal{S} \subseteq \{0, 1\}^d$  and let  $M = \mathbb{E}[\mathbf{V}\mathbf{V}^\top]$  where  $\mathbf{V} \in \mathcal{S}$  is a random vector distributed according to an arbitrary distribution  $\mu$  with support  $\mathcal{S}$ . If COMBAND is run with parameters  $\mathcal{S}$ ,  $\mu$ , and  $\gamma = B\sqrt{\frac{\ln N}{\lambda_{\min}(B+d\lambda_{\min})}}$  where*

*$N = |\mathcal{S}|$ ,  $\lambda_{\min}$  is the smallest nonzero eigenvalue of  $M$ , and  $B \geq \|v\|_1$  for all  $v \in \mathcal{S}$ , then its expected regret after  $n$  steps satisfies*

$$\max_{k=1, \dots, N} \mathbb{E} \left[ \widehat{L}_n - L_n(k) \right] \leq 2B \sqrt{\left( \frac{B}{d\lambda_{\min}} + 1 \right) nd \ln N}$$

Thus, the success of the forecaster crucially depends on the value of the smallest nonzero eigenvalue  $\lambda_{\min}$  of  $\mu$ 's correlation matrix  $M$ . In Section 6 we work out various examples in which  $B/d\lambda_{\min} = \mathcal{O}(1)$ . So in all these cases we obtain

$$\mathbb{E} \left[ \widehat{L}_n - L_n(k) \right] = \mathcal{O} \left( B\sqrt{nd \ln N} \right). \quad (2)$$

Rewriting the above condition as  $\lambda_{\min} = \Omega(B/d)$ , and observing that  $M$  has trace bounded by  $B$ , reveals that we achieve (2) whenever the eigenvalues of  $M$  tend to be equal.

Bound (2) improves on the bound of Dani, Hayes, and Kakade [7] by a factor of  $\sqrt{d}$  and on the bound of Abernethy, Hazan, and Rakhlin [1] by a factor of  $d\sqrt{\ln(n/N)}$ .<sup>1</sup> Computationally, both COMBAND and GEOMETRICHEDGE face the problem of sampling from distributions defined over  $\mathcal{S}$ . In many cases this can be done efficiently, as we discuss in Section 6. The algorithm of [1], instead, works in a completely different way. It performs a randomized gradient descent in the convex hull of  $\mathcal{S}$ , translating each point  $x_t$  in the convex hull into a distribution over  $\mathcal{S}$ . This is done in such a way that sampling  $K_t$  from this distribution ensures  $\mathbb{E}[\ell_t^\top v(K_t)] = \ell_t^\top x_t$ . The efficiency of this procedure depends on the specific choice of  $\mathcal{S}$  (for the path planning problem efficient procedures exist). Moreover, in order to guarantee a good regret, gradient descent is implemented using a *self-concordant* function tailored to the problem. Even if the existence of such a function is guaranteed, its construction may be a non-trivial issue in some applications.

## 5 Proof of Theorem 1

Before proving Theorem 1, we state and prove some auxiliary results.

**Lemma 2** *Let  $\mathbf{V}$  be a random vector with range  $\{0, 1\}^d$  and let  $M = \mathbb{E}[\mathbf{V}\mathbf{V}^\top]$ . Then  $M M^+ v = v$  for all  $v \in \{0, 1\}^d$  such that  $\mathbb{P}\{\mathbf{V} = v\} > 0$ .*

**Proof:** To prove the statement we show that for all  $x \in \mathbb{R}^d$  such that  $Mx = 0$  and for all  $v \in \{0, 1\}^d$  such that  $\mathbb{P}\{\mathbf{V} = v\} > 0$ , it must be the case that  $x^\top v = 0$ . Pick any  $x \in \mathbb{R}^d$  such that  $Mx = 0$ . This implies  $x^\top Mx = 0$ . Using the definition of  $M$  we obtain

$$0 = x^\top Mx = \sum_{v \in \{0, 1\}^d} (x^\top v)^2 \mathbb{P}\{\mathbf{V} = v\}.$$

But then it must be the case that  $x^\top v = 0$  for all  $v$  such that  $\mathbb{P}\{\mathbf{V} = v\} > 0$ . ■

Let  $Q_t = \mathbb{E}[\mathbf{V}\mathbf{V}^\top]$  where  $\mathbf{V}$  has law  $q_t$ . Note that  $Q_t$  is always positive semidefinite since it is a convex combination of positive semidefinite matrices  $v(k)v(k)^\top$ .

**Corollary 3**  *$P_t P_t^+ v = v$  for all  $t$  and all  $v \in \mathcal{S}$ .*

**Proof:** Weyl's inequality implies that  $\lambda_i(P_t) \geq \gamma \lambda_i(M)$ , where  $\lambda_i(P_t)$  is the  $i$ -th largest eigenvalue of  $P_t = (1 - \gamma)Q_t + \gamma M$  and  $\lambda_i(M)$  is the  $i$ -th largest eigenvalue of  $M$ .

<sup>1</sup>In all applications of Section 6,  $\ln N = \mathcal{O}(\sqrt{d} \ln d)$ . Hence the improvement on [1] is at least by a factor of  $d^{3/4} \sqrt{\ln(n/d)}$ .

Therefore the null space of  $P_t$  is included in the null space of  $M$ . This, together with Lemma 2 and the positive semidefiniteness of  $Q_t$ , implies the claimed statement.  $\blacksquare$

**Proof of Theorem 1:** Let  $\mathbb{E}_t$  be the expectation operator conditioned on the first  $t - 1$  random draws  $K_1, \dots, K_{t-1}$  (i.e., expectation w.r.t. the distribution  $p_{t-1}$ ). For all  $k = 1, \dots, N$  let  $\ell_t(k) = \ell_t^\top \mathbf{v}(k)$  so that  $\mathbb{E}_t \ell_t(K_t) \mathbf{v}(K_t) = P_{t-1} \ell_t$ . Let also  $\tilde{\ell}_t(k) = \tilde{\ell}_t^\top \mathbf{v}(k)$ . Hence, by Corollary 3,  $\mathbb{E}_t \tilde{\ell}_t = \ell_t$ . An adaptation of the proof of [7, Theorem 3.3] then gives

$$\mathbb{E} \left[ \sum_{t=1}^n \ell_t(K_t) - L_n(k) \right] \leq \frac{\ln N}{\eta} + \gamma B n + B^2 d \eta n \quad (3)$$

under the condition  $\eta |\tilde{\ell}_t(k)| \leq 1$  for all  $t$  and  $k$ . In order to enforce this condition we write

$$\begin{aligned} |\tilde{\ell}_t(k)| &= |\mathbf{v}(k)^\top \tilde{\ell}_t| = \ell_t(K_t) |\mathbf{v}(k)^\top P_{t-1}^+ \mathbf{v}(K_t)| \\ &\leq B \|P_{t-1}^+\| \max_{\mathbf{v} \in \mathcal{S}} \|\mathbf{v}\|^2 \leq \frac{B^2}{\lambda_{\min}(P_{t-1})} \end{aligned} \quad (4)$$

where  $\lambda_{\min}(P_{t-1})$  is the smallest nonzero eigenvalue of  $P_{t-1}$ , and we used  $\ell_t(K_t) \leq B$  and  $\|\mathbf{v}\|^2 = \|\mathbf{v}\|_1 \leq B$ . Let  $\lambda_{\min} = \lambda_{\min}(M)$ . By Weyl's inequality,  $\lambda_{\min}(P_{t-1}) \geq \gamma \lambda_{\min}$ , which in turn implies that  $|\tilde{\ell}_t(k)| \leq B/(\gamma \lambda_{\min})$ . Hence we choose  $\eta = \gamma \lambda_{\min}/B^2$  and (3) becomes

$$\mathbb{E} \left[ \sum_{t=1}^n \ell_t(K_t) - L_n(k) \right] \leq \frac{B^2 \ln N}{\gamma \lambda_{\min}} + \gamma(B + d \lambda_{\min})n.$$

Letting  $\gamma = B \sqrt{\frac{\ln N}{\lambda_{\min}(B + d \lambda_{\min})}}$  finally yields

$$\mathbb{E} \left[ \sum_{t=1}^n \ell_t(K_t) - L_n(k) \right] \leq 2B \sqrt{\left( \frac{B}{d \lambda_{\min}} + 1 \right) nd \ln N}$$

which ends the proof of Theorem 1.

## 6 Applications

In order to apply Theorem 1 to concrete classes  $\mathcal{S}$  we need to find lower bounds on the smallest eigenvalue  $\lambda_{\min} = \lambda_{\min}(M)$  of the linear transformation

$$M = \sum_{k=1}^N \mathbf{v}(k) \mathbf{v}(k)^\top \mu(k)$$

restricted to the vector space  $U$  spanned by the elements  $\mathbf{v}(1), \dots, \mathbf{v}(N)$  of  $\mathcal{S}$ . Since  $\mu$  has support  $\mathcal{S}$ , Lemma 2 implies that this smallest eigenvalue is strictly positive. Thus we want to bound

$$\lambda_{\min} = \min_{\mathbf{x} \in U: \|\mathbf{x}\|=1} \mathbf{x}^\top M \mathbf{x}.$$

In all of our examples we assume  $\mu$  is uniform over the index set  $\{1, \dots, N\}$ . It is convenient to consider a random vector  $\mathbf{V}$ , uniformly distributed over  $\mathcal{S}$ . Then we have

$$\lambda_{\min} = \min_{\mathbf{x} \in U: \|\mathbf{x}\|=1} \mathbb{E} \mathbf{x}^\top \mathbf{V} \mathbf{V}^\top \mathbf{x}.$$

Since  $\mathbf{x}^\top \mathbf{V} \mathbf{V}^\top \mathbf{x} = (\mathbf{V}^\top \mathbf{x})^2$  we have the following lemma.

### Lemma 4

$$\lambda_{\min} = \min_{\mathbf{x} \in U: \|\mathbf{x}\|=1} \mathbb{E} \left[ (\mathbf{V}^\top \mathbf{x})^2 \right].$$

In what follows we write any  $\mathbf{x} \in U$  as  $\mathbf{x} = \sum_{k=1}^N a(k) \mathbf{v}(k)$  where we let  $\sum_k a(k) = \alpha$ .

### 6.1 A multitask bandit problem

In this first example we consider the case when the decision maker acts in  $m$  games in parallel. For simplicity, assume that in each one of the  $m$  games, the decision maker selects one of  $R$  possible actions (a possibly different action in each game). After selecting the  $m$  actions, only the sum of the losses suffered in the  $m$  games is observed.

**Proposition 5** For the multitask bandit,  $\lambda_{\min} = 1/R$ .

In this case  $B = m$ ,  $d = mR$ ,  $B/d \lambda_{\min} = 1$ , and  $N = R^m$ . Therefore the optimal regret bound (2) holds and becomes

$$\mathbb{E} \left[ \hat{L}_n - L_n(k) \right] \leq 2m^2 \sqrt{2nR \ln R}.$$

Thus, when playing  $m$  games in parallel, the price of getting information about the sum of the losses in spite of the losses suffered separately in each game is just a factor of  $m$  in the regret bound. In this special case COMBAND can be implemented efficiently since it suffices to sample actions independently in each one of the  $R$  games.

**Proof:** We can write the elements of  $\mathcal{S} \subseteq \{0, 1\}^d$  as vectors  $\mathbf{v}(k) \in \{0, 1\}^d$ ,  $k = 1, \dots, R^m$ , with components  $v_{j,i}(k)$ ,  $j = 1, \dots, m$ ,  $i = 1, \dots, R$ . These vectors satisfy

$$\sum_{i=1}^R v_{j,i}(k) = 1 \quad (5)$$

for each  $j = 1, \dots, m$  and  $k = 1, \dots, N = R^m$ . According to Lemma 4, we want to lower bound  $\mathbb{E}[(\mathbf{V}^\top \mathbf{x})^2]$  uniformly over  $\mathbf{x}$  in the span of  $\mathcal{S}$ , where  $\mathbf{V}$  is uniformly distributed over  $\mathcal{S}$ . We denote the components of  $\mathbf{V}$  by  $V_{j,i}$ ,  $j = 1, \dots, m$ ,  $i = 1, \dots, R$  and the corresponding components of  $\mathbf{x}$  by  $x_{j,i}$ . We calculate

$$\mathbb{E} \left[ (\mathbf{V}^\top \mathbf{x})^2 \right] = \text{VAR} [\mathbf{V}^\top \mathbf{x}] + \mathbb{E}^2 [\mathbf{V}^\top \mathbf{x}]$$

where  $\mathbf{x} = \sum_{k=1}^N a(k) \mathbf{v}(k)$  is such that  $\|\mathbf{x}\| = 1$ . By (5), for each  $j = 1, \dots, m$ ,

$$\sum_{i=1}^R x_{j,i} = \sum_{k=1}^N a(k) \sum_{i=1}^R v_{j,i}(k) = \sum_{k=1}^N a(k) = \alpha.$$

Thus,

$$\mathbb{E} \mathbf{V}^\top \mathbf{x} = \sum_{j=1}^m \sum_{i=1}^R x_{j,i} \mathbb{E} V_{j,i} = \sum_{j=1}^m \frac{1}{R} \sum_{i=1}^R x_{j,i} = \frac{m}{R} \alpha.$$

On the other hand, since the  $R$ -vectors  $(V_{j,1}, \dots, V_{j,R})$  are independent for  $j = 1, \dots, m$ ,

$$\begin{aligned} \text{VAR}[\mathbf{V}^\top \mathbf{x}] &= \sum_{j=1}^m \text{VAR} \left[ \sum_{i=1}^R x_{j,i} V_{j,i} \right] \\ &= \sum_{j=1}^m \left( \mathbb{E} \left[ \left( \sum_{i=1}^R x_{j,i} V_{j,i} \right)^2 \right] - \mathbb{E}^2 \left[ \sum_{i=1}^R x_{j,i} V_{j,i} \right] \right) \\ &= \sum_{j=1}^m \left( \frac{1}{R} \sum_{i=1}^R x_{j,i}^2 - \left( \frac{1}{R} \sum_{i=1}^R x_{j,i} \right)^2 \right) \\ &= \frac{1}{R} - \frac{m}{R^2} \alpha^2. \end{aligned}$$

Thus,

$$\mathbb{E}[(\mathbf{V}^\top \mathbf{x})^2] = \frac{1}{R} + \frac{m(m-1)}{R^2} \alpha^2 \geq \frac{1}{R}$$

with equality whenever  $\alpha = 0$ .  $\blacksquare$

## 6.2 Perfect matchings: learning permutations

Consider the complete bipartite graph  $K_{m,m}$  and let  $\mathcal{S}$  contain all perfect matchings. Thus,  $d = m^2$  (the number of edges of  $K_{m,m}$ ),  $\mathcal{S}$  has  $N = m!$  members, and  $B = m$ . Each  $\mathbf{v}(k) \in \mathcal{S}$  may be represented by an  $m \times m$  permutation matrix  $[v_{i,j}(k)]_{m \times m}$ ; that is, a doubly stochastic zero-one matrix such that  $\sum_{j=1}^m v_{i,j}(k) = 1$  for all  $i = 1, \dots, m$  and  $\sum_{i=1}^m v_{i,j}(k) = 1$  for all  $j = 1, \dots, m$ . Online learning of perfect matchings (or, equivalently, permutations) was considered by Helmbold and Warmuth [11] who introduced a computationally efficient forecaster with good regret bounds in the full-information setting. Here we show that COMBAND performs well for this problem and point out that it has a computationally efficient implementation. The next proposition shows that the term  $\lambda_{\min}$  in Theorem 1 is sufficiently large.

Let  $[V_{i,j}]_{m \times m}$  be chosen uniformly at random from the collection  $[v_{i,j}(k)]_{m \times m}$ ,  $k = 1, \dots, N$ , representing a random permutation (i.e., perfect matching).

**Proposition 6** *For the perfect matchings on  $K_{m,m}$ ,*

$$\lambda_{\min} = \frac{1}{m-1}.$$

It follows from the proposition that  $B/d\lambda_{\min} \leq 1$ , and therefore the optimal bound (2) holds and it takes the form

$$\mathbb{E} \left[ \widehat{L}_n - L_n(k) \right] \leq 2m^2 \sqrt{2n \ln(m!)}.$$

The fact that COMBAND can be implemented efficiently follows from a beautiful and deep result of Jerrum, Sinclair, and Vigoda [13] who were the first to describe a polynomial-time randomized algorithm for approximating the permanent of a matrix with non-negative entries. To see the connection, observe that the sum of the weights  $\overline{W}_t = \sum_{k=1}^{m!} \overline{w}_t(k)$  is just the permanent of a matrix with entries  $\exp(-\eta \widetilde{L}_{t,(i,j)})$ ,  $i, j \in \{1, \dots, m\}$  where  $\widetilde{L}_{t,(i,j)}$  is the estimated cumulative

loss of edge  $(i, j)$ . The algorithm of Jerrum, Sinclair, and Vigoda is based on random sampling perfect matchings from the (approximate) distribution given by the  $\overline{w}_t(k)$  which is exactly what we need to draw a random perfect matching according to the exponentially weighted average distribution.

**Proof:** By Lemma 4, we need a lower bound for

$$\mathbb{E}[(\mathbf{V}^\top \mathbf{x})^2] = \mathbb{E} \left[ \left( \sum_{i=1}^m \sum_{j=1}^m V_{i,j} x_{i,j} \right)^2 \right]$$

where  $\mathbf{x} = \sum_{k=1}^N a(k) \mathbf{v}(k)$  is such that  $\sum_{i,j=1}^m x_{i,j}^2 = 1$ . Observe that for any fixed  $i$ ,

$$\sum_{j=1}^m x_{i,j} = \sum_{k=1}^N a(k) \sum_{j=1}^m v_{i,j}(k) = \sum_{k=1}^N a_k = \alpha$$

and similarly, for any fixed  $j$ ,  $\sum_{i=1}^m x_{i,j} = \sum_{k=1}^N a_k = \alpha$ . Since

$$\mathbb{P}\{V_{i,j} = 1, V_{i',j'} = 1\} = \begin{cases} \frac{1}{m} & \text{if } i = i' \text{ and } j = j', \\ \frac{1}{m(m-1)} & \text{if } i \neq i' \text{ and } j \neq j', \\ 0 & \text{otherwise} \end{cases}$$

we have

$$\begin{aligned} \mathbb{E}[(\mathbf{V}^\top \mathbf{x})^2] &= \mathbb{E} \left[ \left( \sum_{i,j=1}^m V_{i,j} x_{i,j} \right)^2 \right] \\ &= \sum_{i,j=1}^m \sum_{i',j'=1}^m x_{i,j} x_{i',j'} \mathbb{P}\{V_{i,j} = 1, V_{i',j'} = 1\} \\ &= \frac{1}{m} \sum_{i,j=1}^m x_{i,j}^2 + \frac{1}{m(m-1)} \sum_{i,j=1}^m \sum_{i' \neq i} \sum_{j' \neq j} x_{i,j} x_{i',j'} \\ &= \frac{1}{m} + \frac{1}{m(m-1)} \sum_{i,j=1}^m \sum_{i' \neq i} \sum_{j' \neq j} x_{i,j} x_{i',j'} \end{aligned}$$

The second term on the right-hand side may be written as

$$\begin{aligned} \sum_{i,j=1}^m \sum_{i' \neq i} \sum_{j' \neq j} x_{i,j} x_{i',j'} &= \sum_{i,j=1}^m \sum_{i',j'=1}^m x_{i,j} x_{i',j'} \\ &- \sum_{i,j=1}^m \sum_{j'=1}^m x_{i,j} x_{i',j'} - \sum_{i,j=1}^m \sum_{i'=1}^m x_{i,j} x_{i',j'} + \frac{1}{m} \sum_{i,j=1}^m x_{i,j}^2 \\ &= \left( \sum_{i,j=1}^m x_{i,j} \right)^2 - \sum_{i=1}^m \left( \sum_{j=1}^m x_{i,j} \right)^2 \\ &\quad - \sum_{j=1}^m \left( \sum_{i=1}^m x_{i,j} \right)^2 + 1 \\ &= \left( m \sum_{k=1}^N a(k) \right)^2 - 2m \left( \sum_{k=1}^N a(k) \right)^2 + 1. \end{aligned}$$

Summarizing, we have that for all  $\mathbf{x} = \sum_{k=1}^N a(k)\mathbf{v}(k)$  such that  $\|\mathbf{x}\| = 1$ ,

$$\begin{aligned}\mathbb{E}\left[(\mathbf{V}^\top \mathbf{x})^2\right] &= \frac{1}{m} + \frac{1}{m(m-1)} \left( (m\alpha)^2 - 2m\alpha^2 + 1 \right) \\ &= \frac{1}{m-1} + \frac{m-2}{m-1} \alpha^2\end{aligned}$$

which is at least  $1/(m-1)$  with equality whenever  $\alpha = 0$ .  $\blacksquare$

### 6.3 Spanning trees

Next we consider an online decision problem in which, at each time instance, the decision maker chooses a spanning tree in a graph of  $m$  nodes. The loss of a spanning tree is the sum of the losses over the edges of the tree. Such a problem is meaningful in certain mobile communication networks, in which a minimum-cost subnetwork is to be selected at each time frame to assure connectedness of the whole network. This problem fits in our general framework if we let  $\mathcal{S}$  be the family of all spanning trees of the complete graph  $K_m$ . Thus,  $d = \binom{m}{2}$ ,  $B = m - 1$ , and by Cayley's formula there are  $N = m^{m-2}$  spanning trees.

In order to estimate  $\lambda_{\min}$  for this case, we start with a general lemma that applies for all sufficiently "symmetric" classes  $\mathcal{S}$ . More precisely, we consider the case when the elements of  $\mathcal{S} \subseteq \{0, 1\}^d$  are the incidence vectors of certain subsets of the edges of a complete graph  $K_m$  (i.e.,  $d = \binom{m}{2}$  in these cases). If  $i$  and  $j$  are distinct edges of  $K_m$ , we write  $i \sim j$  when  $i$  and  $j$  are adjacent (i.e., they have a common endpoint) and  $i \not\sim j$  when  $i$  and  $j$  are disjoint.

We require that  $\mathcal{S}$  is sufficiently symmetric, so that if  $\mathbf{V}$  is drawn uniformly at random from  $\mathcal{S}$ , then the probability  $\mathbb{P}\{V_i = 1, V_j = 1\}$  can take at most three different values depending on whether  $i = j$ ,  $i \sim j$ , or  $i \not\sim j$ .

In such cases, if  $\mathbf{x} = (x_1, \dots, x_d)$  is any vector in  $\mathbb{R}^d$ , then

$$\begin{aligned}\mathbb{E}\left[(\mathbf{V}^\top \mathbf{x})^2\right] &= \sum_{i=1}^d \sum_{j=1}^d x_i x_j \mathbb{P}\{V_i = 1, V_j = 1\} \\ &= C_1 \sum_{i=1}^d x_i^2 + C_2 \sum_{i,j:i \sim j} x_i x_j + C_3 \sum_{i,j:i \not\sim j} x_i x_j\end{aligned}\quad (6)$$

where

$$\begin{aligned}C_1 &\stackrel{\text{def}}{=} \mathbb{P}\{V_i = 1\} && \forall i = 1, \dots, d \\ C_2 &\stackrel{\text{def}}{=} \mathbb{P}\{V_i = 1, V_j = 1\} && \forall i, j = 1, \dots, d \text{ s.t. } i \sim j \\ C_3 &\stackrel{\text{def}}{=} \mathbb{P}\{V_i = 1, V_j = 1\} && \forall i, j = 1, \dots, d \text{ s.t. } i \not\sim j\end{aligned}$$

are quantities independent of  $i, j$ .

This property is true for collections  $\mathcal{S}$  of "symmetric" subsets of  $K_m$ , such as spanning trees, balanced cuts, planar graphs, Hamiltonian cycles, cliques of a certain size, etc. The following result provides a general lower bound for the smallest eigenvalue of the associated matrix  $M$ .

**Lemma 7** *If (6) holds and  $\mathbf{x} \in \mathbb{R}^d$  has unit norm, then*

$$\mathbb{E}\left[(\mathbf{V}^\top \mathbf{x})^2\right] \geq C_1 - C_3 - |C_2 - C_3| m - \frac{(C_2 - C_3)^2}{C_3}.$$

**Proof:** Since  $\|\mathbf{x}\| = 1$ , we have

$$\begin{aligned}\mathbb{E}\left[(\mathbf{V}^\top \mathbf{x})^2\right] &= C_1 + C_2 \sum_{i,j:i \sim j} x_i x_j + C_3 \sum_{i,j:i \not\sim j} x_i x_j \\ &= C_1 - C_3 + (C_2 - C_3) \sum_{i,j:i \sim j} x_i x_j + C_3 \sum_{i,j=1}^d x_i x_j.\end{aligned}$$

Denote the summation over all pairs of adjacent edges by

$$A_m = \sum_{i,j:i \sim j} x_i x_j \quad \text{and let} \quad B_m = \left( \sum_{i=1}^d x_i \right)^2.$$

With this notation, we have

$$\mathbb{E}\left[(\mathbf{V}^\top \mathbf{x})^2\right] = C_1 - C_3 + (C_2 - C_3)A_m + C_3 B_m. \quad (7)$$

Next we need an appropriate estimate for  $A_m$ . By the Cauchy-Schwarz inequality, and using the fact that  $\|\mathbf{x}\| = 1$ ,

$$\begin{aligned}|A_m| &= \left| \sum_{i=1}^d x_i \sum_{j:i \sim j} x_j \right| \\ &\leq \sqrt{\sum_{i=1}^d \left( \sum_{j:i \sim j} x_j \right)^2} \\ &= \sqrt{\sum_{i=1}^d \left( \sum_{j,l:j \sim i, l \sim i} x_j x_l \right)} \\ &= \sqrt{(m-2) \sum_{i,j:i \sim j} x_i x_j + 4 \sum_{i,j:i \not\sim j} x_i x_j}.\end{aligned}\quad (8)$$

The last equality holds because a pair of edges is counted  $m-2$  times if they are adjacent ( $m-2$  is the number of edges adjacent to both) and 4 times if they are not adjacent. We may write the argument of the square root in (8) as

$$\begin{aligned}(m-2) \sum_{i,j:i \sim j} x_i x_j + 4 \sum_{i,j:i \not\sim j} x_i x_j \\ = (m-6) \sum_{i,j:i \sim j} x_i x_j + 4 \sum_{i,j} x_i x_j - 4 \\ \leq m |A_m| + 4 B_m.\end{aligned}\quad (9)$$

Thus, substituting (9) in (8), and using  $B_m \geq 0$ , we get

$$|A_m| \leq \sqrt{m |A_m| + 4 B_m}.$$

Solving the above for  $|A_m|$  and overapproximating gives

$$|A_m| \leq m + 2\sqrt{B_m}$$

which, substituted into (7) yields

$$\mathbb{E}\left[(\mathbf{V}^\top \mathbf{x})^2\right] \geq C_1 - C_3 - |C_2 - C_3| \left( m + 2\sqrt{B_m} \right) + C_3 B_m.$$

Observing that

$$\begin{aligned}C_3 B_m - 2|C_2 - C_3| \sqrt{B_m} \\ = \left( \sqrt{C_3 B_m} - \frac{|C_2 - C_3|}{\sqrt{C_3}} \right)^2 - \frac{(C_2 - C_3)^2}{C_3} \\ \geq -\frac{(C_2 - C_3)^2}{C_3}\end{aligned}$$

concludes the proof.  $\blacksquare$

Interestingly, the proof above does not use that fact that  $x$  in the space spanned by the incidence vectors of  $\mathcal{S}$ . Thus, the matrix  $\mathbb{E}[\mathbf{V}\mathbf{V}^\top]$  is positive definite whenever the lower bound of Lemma 7 is positive. This also implies that the matrix  $P_t$ , which is used to define the pseudo-losses (1), is positive definite, and thus  $P_t^+$  can be replaced by  $P_t^{-1}$ .

Now we may use Lemma 7 to bound  $\lambda_{\min}$  in the case of spanning trees of the complete graph  $K_m$ . All we need is to calculate the values of  $C_1$ ,  $C_2$ , and  $C_3$ . We do it by applying the theory of electric networks.

**Lemma 8** *If  $\mathbf{V}$  is the incidence vector of a uniform random spanning tree of  $K_m$ , then*

$$\begin{aligned}\mathbb{P}\{V_i = 1\} &= \frac{2}{m} \\ \mathbb{P}\{V_i = 1, V_j = 1\} &= \frac{3}{m^2} \quad \text{if } i \sim j \\ \mathbb{P}\{V_i = 1, V_j = 1\} &= \frac{4}{m^2} \quad \text{if } i \not\sim j.\end{aligned}$$

**Proof:** Since every spanning tree has  $m - 1$  edges,

$$\mathbb{P}\{V_1 = 1\} + \dots + \mathbb{P}\{V_d = 1\} = m - 1$$

where  $d = \binom{m}{2}$ . By symmetry,  $\mathbb{P}\{V_i = 1\} = 2/m$  for all  $i = 1, \dots, d$ . The other two cases can be handled by the ‘‘Transfer Current’’ theorem of Burton and Pemantle [5], see also Lyons and Peres [15], which implies that for any  $i \neq j$ ,

$$\mathbb{P}\{V_i = 1, V_j = 1\} = \frac{4}{m^2} - Y(i, j)^2$$

where  $Y(i, j)$  is the voltage difference across the edge  $j$  when a unit current is imposed between the endpoints of edge  $i$ . (For the basic notions of electric networks we refer, e.g., to the books of Doyle and Snell [9] and Lyons and Peres [15].)

First note that if  $i$  and  $j$  are not adjacent then  $Y(i, j) = 0$ . This is simply because, by symmetry, every vertex not belonging to edge  $i$  has the same voltage, so there is no current flowing through edge  $j$ . Thus,  $\mathbb{P}\{V_i = 1, V_j = 1\} = 4/m^2$  in this case.

In order to address the case when edges  $i$  and  $j$  are adjacent,  $i \sim j$ , note that, by a result of Kirchoff (1847), the voltage difference between the endpoints of  $i$  equals the probability  $2/m$  that  $i$  belongs to a random spanning tree (see, e.g., the remark to Corollary 4.4 in [15]). By the above considerations, there is current flow only along paths of length two between the endpoints of  $i$ , that is paths that go through edges  $j \sim i$ . Hence the voltage difference between the endpoints of  $j$  is half the voltage difference between the endpoints of  $i$ , that is  $|Y(i, j)| = 1/m$ .  $\blacksquare$

**Corollary 9** *For the spanning trees of  $K_m$ ,*

$$\lambda_{\min} \geq \frac{1}{m} - \frac{17}{4m^2}.$$

Since  $d = \binom{m}{2}$  and  $B = m - 1$ , the inequality above implies that  $B/(d\lambda_{\min}) < 7$  whenever  $m \geq 6$ , and therefore the optimal bound (2) holds. Since  $N = m^{m-2}$ , the performance bound of COMBAND in this case implies

$$\mathbb{E} \left[ \widehat{L}_n - L_n(k) \right] \leq 4m^{5/2} \sqrt{n \ln m} \quad \text{for } m \geq 6.$$

Finding computationally efficient algorithms for generating random spanning trees has been an intensive area of research. Although some of these algorithms may be successfully used in practical implementations, we are not aware of any algorithm that guarantees an efficient implementation of COMBAND under all circumstances. Instead of surveying the vast literature, we mention the celebrated method of Propp and Wilson [17], who present an algorithm that, given a graph with non-negative weights  $w_{(i,j)}$  over the edges, samples a random spanning tree from a distribution such that the probability of any spanning tree  $k$  is proportional to  $\overline{w}_t(k) = \prod_{(i,j) \in k} w_{(i,j)}$ . The expected running time of the algorithm is bounded by the cover time of an associated Markov chain that is defined as a random walk over the graph in which the transition probabilities are proportional to the edge weights. If we apply Propp and Wilson’s algorithm with weights  $w_{(i,j)} = \exp(-\eta \widetilde{L}_{t,(i,j)})$  over the complete graph  $K_m$ , then we obtain an implementation of the exponentially weighted average forecaster. Unfortunately, there is no guarantee that the cover time is bounded by a polynomial of  $m$ , though in practice we expect a fast running time in most cases. It is an interesting open problem to find an efficient sampling algorithm for all possible assignments of weights.

#### 6.4 Cut sets

In this section we consider balanced cuts of the complete graph  $K_{2m}$ . A balanced cut is the collection of all edges between a set of  $m$  vertices and its complement. Thus, each balanced cut has  $B = m^2$  edges and there are  $N = \binom{2m}{m}$  balanced cuts.

Our starting point in estimating  $\lambda_{\min}$  is (7). First, we compute  $C_1$ ,  $C_2$ , and  $C_3$ .

**Lemma 10** *If  $\mathbf{V}$  is the incidence vector of a uniform random  $m$ -cut in  $K_{2m}$ , then*

$$\begin{aligned}\mathbb{P}\{V_i = 1\} &= \frac{m}{2m-1} \\ \mathbb{P}\{V_i = 1, V_j = 1\} &= \frac{m(m-1)}{(2m-1)(2m-2)} \quad \text{if } i \sim j \\ \mathbb{P}\{V_i = 1, V_j = 1\} &= \frac{2m(m-1)^2}{(2m-1)(2m-2)(2m-3)} \quad \text{if } i \not\sim j.\end{aligned}$$

**Proof:** The sample space is all choices of  $m$ -subsets of  $2m$  vertices (note that each  $m$ -cut is counted twice). Fix an edge  $i = (i_-, i_+)$ . Then the number of  $m$ -subsets that contain  $i_-$  and do not contain  $i_+$  is clearly  $\binom{2m-2}{m-1}$ . By symmetry, this is also the number of  $m$ -subsets that contain  $i_+$  and do not contain  $i_-$ . Therefore

$$\mathbb{P}\{V_i = 1\} = 2 \times \frac{\binom{2m-2}{m-1}}{\binom{2m}{m}} = \frac{m}{2m-1}.$$

Now fix two edges  $i$  and  $j$  that share a vertex, say  $i_- = j_-$ . The number of  $m$ -subsets that contain  $i_- = j_-$  and do not contain neither  $i_+$  nor  $j_+$  is  $\binom{2m-3}{m-1}$ . This is the same as the number of  $m$ -subsets that do not contain  $i_- = j_-$  and contain both  $i_+$  and  $j_+$ . Hence, if  $i \sim j$ ,

$$\mathbb{P}\{V_i = 1, V_j = 1\} = 2 \times \frac{\binom{2m-3}{m-1}}{\binom{2m}{m}} = \frac{m(m-1)}{(2m-1)(2m-2)}.$$

Finally, fix two disjoint edges  $i$  and  $j$ . The number of  $m$ -subsets that contain  $i_+, j_+$  and do not contain neither  $i_-$  nor  $j_-$  is  $\binom{2m-4}{m-2}$ . By symmetry, this is also the number of  $m$ -subsets that contain  $i_-, j_-$  and do not contain neither  $i_+$  nor  $j_+$ , which is the same as the number of those that contain  $i_-, j_+$  and not  $i_+$  or  $j_-$ , etc. Hence, for  $i \not\sim j$ ,

$$\begin{aligned} \mathbb{P}\{V_i = 1, V_j = 1\} &= 4 \times \frac{\binom{2m-4}{m-2}}{\binom{2m}{m}} \\ &= \frac{2m(m-1)^2}{(2m-1)(2m-2)(2m-3)} \end{aligned}$$

concluding the proof.  $\blacksquare$

Now we may make use of the fact that each balanced cut has the same number of edges. Thus, if  $\mathbf{x} = \sum_{k=1}^{\binom{2m}{m}} a(k) \mathbf{v}(k)$  is a linear combination of the incidence vectors of all balanced cuts with  $\|\mathbf{x}\| = 1$ , we have  $\sum_i x_i = m^2 \alpha$  where  $\alpha = \sum_{k=1}^{\binom{2m}{m}} a(k)$ , which implies that  $B_m = m^4 \alpha^2$ .

To compute  $A_m$ , observe that for any fixed  $i$ , the number of edges in any balanced cut adjacent to  $i$  is  $2m$  if the cut doesn't contain  $i$  and  $2(m-1)$  otherwise, that is,

$$\sum_{j: j \sim i} v_i(k) = \begin{cases} 2(m-1) & \text{if } v_i(k) = 1 \\ 2m & \text{if } v_i(k) = 0 \end{cases}$$

so

$$\begin{aligned} \sum_{j: j \sim i} x_j &= \sum_{k=1}^N a(k) \sum_{j: j \sim i} v_i(k) = \sum_{k=1}^N a(k) (2m - 2v_i(k)) \\ &= 2m\alpha - 2 \sum_{k=1}^N a(k) v_i(k) = 2m\alpha - 2x_i. \end{aligned}$$

Therefore, we have

$$A_m = \sum_{i, j: i \sim j} x_i x_j = \sum_i x_i \sum_{j: j \sim i} x_j = m^3 \alpha^2 - 2.$$

Substituting these values in (7), we have, for  $m \geq 2$ ,

$$\begin{aligned} \mathbb{E} \left[ (\mathbf{V}^\top \mathbf{x})^2 \right] &= \frac{1}{4} + \frac{8m-7}{4(2m-1)(2m-3)} \\ &\quad + \alpha^2 \frac{m^4(m-1)(2m^2-2m-1)}{(2m-1)(2m-2)(2m-3)}. \end{aligned}$$

The minimum is achieved for  $\alpha = 0$ , which proves the following.

**Proposition 11** *For the balanced cuts in  $K_{2m}$ , if  $m \geq 2$  then*

$$\lambda_{\min} = \frac{1}{4} + \frac{8m-7}{4(2m-1)(2m-3)}.$$

In this case we have  $d = \binom{2m}{2}$ ,  $B = m^2$ , and  $N = \binom{2m}{m} \leq 4^m$ . By Proposition 11 we clearly have  $B/(d\lambda_{\min}) \leq 2$  for all  $m \geq 2$ , and therefore the optimal bound (2) applies and it takes the form

$$\mathbb{E} \left[ \widehat{L}_n - L_n(k) \right] \leq 2m^{7/2} \sqrt{6n \ln 4}.$$

In this case computationally efficient implementations also exist. Such an implementation may be based on an algorithm of Randall and Wilson [18] who, building on Jerrum and Sinclair [12], show how to sample efficiently spin configurations of a ferromagnetic Ising model. The straightforward details are omitted.

## 6.5 Hamiltonian cycles

In our next examples we consider the set  $\mathcal{S}$  of all Hamiltonian cycles in  $K_m$ , that is all  $N = (m-1)!/2$  cycles that visit each vertex exactly once and returns to the starting vertex. The corresponding randomized prediction problem may be thought of as an online version of the traveling salesman problem. This problem is computationally notoriously difficult and one cannot expect polynomial-time implementations. Nevertheless, we show that small regret bounds are achievable by COMBAND. To this end, we calculate  $\lambda_{\min}$ .

**Proposition 12** *If  $m \geq 4$ , then for the class of all Hamiltonian cycles in  $K_m$   $\lambda_{\min} = 2/(m-1)$ .*

Since  $d = \binom{m}{2}$ ,  $N = (m-1)!/2$ , and  $B = m$ , we have  $B/(d\lambda_{\min}) = 1$ . Thus the optimal bound (2) applies achieving

$$\mathbb{E} \left[ \widehat{L}_n - L_n(k) \right] \leq 2m^2 \sqrt{n \ln(m!)}.$$

**Proof:** Once again, our analysis is based on (7). First we calculate the values of the constants  $C_1, C_2, C_3$ . Since each Hamiltonian cycle has  $m$  edges, if  $\mathbf{V}$  is a random Hamiltonian cycle, then  $C_1 = \mathbb{P}\{V_i = 1\} = 2/(m-1)$ . Also, since the degree of every vertex in a Hamiltonian cycle is 2, for any two adjacent edges  $i \sim j$ ,  $C_2 = \mathbb{P}\{V_i = 1, V_j = 1\} = 1/\binom{m-1}{2}$ . On the other hand, if  $i \not\sim j$ , then

$$\begin{aligned} \mathbb{P}\{V_i = 1, V_j = 1\} &= \mathbb{P}\{V_i = 1\} \mathbb{P}\{V_j = 1 | V_i = 1\} \\ &= \frac{2}{m-1} \times \frac{m-3}{\binom{m}{2} - 2(m-2) - 1} \end{aligned}$$

because there are  $\binom{m}{2} - 2(m-2) - 1$  edges in  $K_m$  that are not adjacent to  $i$  and all of them are equally likely to be any of the remaining  $m-3$  edges of the cycle  $\mathbf{V}$ . Thus,  $C_3 = 4/(m-1)(m-2)$ .

Now let  $\mathbf{x} = \sum_{k=1}^N a(k) \mathbf{v}(k)$  be a linear combination of the incidence vectors of all Hamiltonian cycles such that  $\|\mathbf{x}\| = 1$ . The crucial observation is the following: since every  $\mathbf{v}(k)$  has  $m$  edges, and the degree of every vertex equals 2, we have

$$\sum_i x_i = \sum_{k=1}^N a(k) \sum_i v_i(k) = m\alpha.$$

This implies that

$$B_m = \left( \sum_{i=1}^d x_i \right)^2 = m^2 \alpha^2.$$

Observe that for any fixed  $i$ , the number of edges in any Hamiltonian cycle adjacent to  $i$  is 4 if the cycle doesn't contain  $i$  and 2 otherwise, that is,

$$\sum_{j: j \sim i} v_i(k) = \begin{cases} 2 & \text{if } v_i(k) = 1 \\ 4 & \text{if } v_i(k) = 0 \end{cases}$$

Thus,

$$\begin{aligned} \sum_{j: j \sim i} x_j &= \sum_{k=1}^N a(k) \sum_{j: j \sim i} v_i(k) = \sum_{k=1}^N a(k) (4 - 2v_i(k)) \\ &= 4\alpha - 2 \sum_{k=1}^N a(k) v_i(k) = 4\alpha - 2x_i. \end{aligned}$$



Using this, we have

$$\begin{aligned} A_m &= \sum_i x_i \sum_{j:j \sim i} x_j = \sum_i x_i (4\alpha - 2x_i) \\ &= 4m\alpha^2 - 2 \sum_i x_i^2 = 4m\alpha^2 - 2. \end{aligned}$$

Substituting these values in (7), we have

$$\begin{aligned} \mathbb{E} \left[ (\mathbf{V}^\top \mathbf{x})^2 \right] &= \frac{2(m-4)}{(m-1)(m-2)} + \frac{2(2m^2\alpha^2 - 4m\alpha^2 + 2)}{(m-1)(m-2)} \\ &= \frac{2}{m-1} + \frac{4m\alpha^2}{m-1} \geq \frac{2}{m-1}. \end{aligned}$$

with equality achieved for  $\sum_k a(k) = 0$ .  $\blacksquare$

## 6.6 Stars

A star is a subgraph of  $K_m$  which contains all  $m-1$  edges incident on a fixed vertex. Thus, there are  $m$  different stars in  $K_m$ . Consider the set  $\mathcal{S}$  of all stars and let  $\mathbf{V}$  be the incidence vector of a random star, chosen uniformly.

**Proposition 13** *For the stars in  $K_m$ ,*

$$\lambda_{\min} = \frac{m-3}{2(m-2)} + \frac{1}{m}.$$

Here  $d = \binom{m}{2}$ ,  $N = m$ , and  $B = m-1$ . Thus we have  $B/(d\lambda_{\min}) \leq \frac{1}{2}$  and the optimal bound (2) applies with

$$\mathbb{E} \left[ \widehat{L}_n - L_n(k) \right] \leq m^2 \sqrt{3n \ln m}.$$

The implementation of COMBAND is trivially efficient in this case.

**Proof:** Clearly,  $\mathbb{P}\{V_i = 1\} = 2/m$ ,  $\mathbb{P}\{V_i = 1, V_j = 1\} = 1/m$  if  $i \sim j$  and  $\mathbb{P}\{V_i = 1, V_j = 1\} = 0$  if  $i \not\sim j$ . Therefore,

$$\mathbb{E} \left[ (\mathbf{V}^\top \mathbf{x})^2 \right] = \frac{2}{m} + \frac{1}{m} A_m$$

where  $A_m = \sum_{i,j:i \sim j} x_i x_j$ . Let  $\mathbf{x} = \sum_{k=1}^m a_k \mathbf{v}_k$  be such that  $\|\mathbf{x}\| = 1$ . This means that

$$1 = \sum_{i=1}^d \left( \sum_{k=1}^m a_k v_i^{(k)} \right)^2 = \sum_{k=1}^m \sum_{k'=1}^m a_k a_{k'} \sum_{i=1}^d v_i^{(k)} v_i^{(k')}.$$

Since

$$\sum_{i=1}^d v_i^{(k)} v_i^{(k')} = \begin{cases} 1 & \text{if } k \neq k' \\ m-1 & \text{if } k = k', \end{cases}$$

we have

$$(m-2) \sum_{k=1}^m a_k^2 + \left( \sum_{k=1}^m a_k \right)^2 = 1. \quad (10)$$

Now

$$\begin{aligned} A_m &= \sum_{i,j:i \sim j} \left( \sum_{k=1}^m a_k v_i^{(k)} \right) \left( \sum_{k=1}^m a_k v_j^{(k)} \right) \\ &= \sum_{k,k'=1}^m a_k a_{k'} \left( \sum_{i,j:i \sim j} v_i^{(k)} v_j^{(k')} \right). \end{aligned}$$

Observe that

$$\sum_{i,j:i \sim j} v_i^{(k)} v_j^{(k')} = \begin{cases} 2(m-1) & \text{if } k \neq k' \\ \binom{m-1}{2} & \text{if } k = k', \end{cases}$$

so

$$A_m = \left( \binom{m-1}{2} - 1 \right) \sum_{k=1}^m a_k^2 + 2(m-1) \left( \sum_{k=1}^m a_k \right)^2$$

Expressing  $\sum_{k=1}^m a_k^2$  from (10), and substituting in the expression above, we obtain

$$\begin{aligned} A_m &= \frac{m(m-3)}{2(m-2)} + \left( \sum_{k=1}^m a_k \right)^2 \left( 2(m-1) - \frac{m(m-3)}{2(m-2)} \right) \\ &\geq \frac{m(m-3)}{2(m-2)}. \end{aligned}$$

In conclusion,

$$\lambda_{\min} \geq \frac{2}{m} + \frac{m-3}{2(m-2)}$$

with equality for  $\sum_k a_k = 0$ .  $\blacksquare$

## 6.7 $m$ -sized subsets

Consider  $\mathcal{S}$  to be the set of all  $\mathbf{v} \in \{0,1\}^d$  such that  $\sum_{i=1}^d v_i = m$  for some fixed  $m$  with  $1 \leq m < d$ .

**Proposition 14** *For the  $m$ -sized subsets,*

$$\lambda_{\min} = \frac{m(d-m)}{d(d-1)}.$$

We have  $B = m$ ,  $N = \binom{d}{m}$ . Then

$$\frac{B}{d\lambda_{\min}} = \frac{d-1}{d-m}.$$

Thus the optimal bound (2) applies whenever  $m = o(d)$ . In this case the regret bound has the form

$$\mathbb{E} \left[ \widehat{L}_n - L_n(k) \right] = \mathcal{O} \left( m^{3/2} \sqrt{nd \ln d} \right).$$

Note that also in this case COMBAND can be implemented efficiently using dynamic programming (see, e.g., Takimoto and Warmuth [19]).

**Proof:** Pick  $\mathbf{x} \in U$  such that  $\|\mathbf{x}\| = 1$ . Note that

$$\sum_{i=1}^d x_i = \sum_{k=1}^N a(k) \sum_{i=1}^d v_i(k) = m \sum_{k=1}^N a(k) = m\alpha.$$

Since for any  $i$ ,

$$\mathbb{P}\{V_i = 1\} = \frac{\binom{d-1}{m-1}}{\binom{d}{m}} = \frac{m}{d}$$

and for any  $i \neq j$

$$\mathbb{P}\{V_i = 1, V_j = 1\} = \frac{\binom{d-2}{m-2}}{\binom{d}{m}} = \frac{m(m-1)}{d(d-1)}$$

we can write

$$\begin{aligned}
\mathbb{E} \left[ (\mathbf{V}^\top \mathbf{x})^2 \right] &= \sum_{i=1}^d \sum_{j=1}^d x_i x_j \mathbb{P}\{V_i = 1, V_j = 1\} \\
&= \frac{m}{d} \sum_{i=1}^d x_i^2 + \frac{m(m-1)}{d(d-1)} \sum_{i,j:i \neq j} x_i x_j \\
&= \left( \frac{m}{d} - \frac{m(m-1)}{d(d-1)} \right) \sum_{i=1}^d x_i^2 + \frac{m(m-1)}{d(d-1)} \sum_{i,j} x_i x_j \\
&= \left( \frac{m}{d} - \frac{m(m-1)}{d(d-1)} \right) + \frac{m(m-1)}{d(d-1)} m^2 \alpha^2 \\
&= \frac{m(d-m)}{d(d-1)} + \frac{m^3(m-1)}{d(d-1)} \alpha^2 \geq \frac{m(d-m)}{d(d-1)}
\end{aligned}$$

with equality whenever  $\alpha = 0$ . ■

## 7 Conclusions

In this work we have investigated the problem of bandit online linear optimization when the action set  $\mathcal{S}$  is a finite subset of  $\{0, 1\}^d$ , the action vectors  $\mathbf{v} \in \mathcal{S}$  satisfy  $\|\mathbf{v}\|_1 \leq B$ , and the loss vectors  $\ell_t$  satisfy  $\|\ell_t\|_\infty \leq 1$ . Our setting is a special case of both [1] and [7], as in these papers  $\mathbf{v}$  and  $\ell_t$  need only satisfy the condition  $|\ell_t^\top \mathbf{v}| \leq 1$ , which does not refer to any system of coordinates. We take advantage of the additional assumptions to prove better expected regret bounds for several concrete choices of  $\mathcal{S}$ . These improvements are obtained through a new randomized forecasting strategy, COMBAND, closely related to the GEOMETRICHEDGE algorithm of [7].

Although the regret of COMBAND can not be improved in general, in some interesting cases (like the path planning problem) COMBAND has a suboptimal performance because a uniform initial sampling distribution  $\mu$  causes the smallest nonzero eigenvalue  $\lambda_{\min}$  to get too small. In general,  $\mu$  can be chosen in order to maximize  $\lambda_{\min}$  by solving a semidefinite program. We conjecture that for the path planning problem this choice of  $\mu$  is polytime computable, and COMBAND, run with this  $\mu$ , has optimal regret  $B\sqrt{nd \ln |\mathcal{S}|}$ .

### Acknowledgements

Thanks to the reviewers for their insightful comments.

## References

- [1] J. Abernethy, E. Hazan, and A. Rakhlin. Competing in the dark: An efficient algorithm for bandit linear optimization. In *Proceedings of the 21st Annual Conference on Learning Theory (COLT 2008)*, pages 263–274. Omnipress, 2008.
- [2] P. Auer, N. Cesa-Bianchi, Y. Freund, and R.E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM Journal on Computing*, 32:48–77, 2002.
- [3] B. Awerbuch and R.D. Kleinberg. Adaptive routing with end-to-end feedback: distributed learning and geometric approaches. In *Proceedings of the 36th Annual ACM Symposium on Theory of Computing (STOC 2004)*, pages 45–53. ACM Press, 2004.
- [4] P. Bartlett, V. Dani, T. Hayes, S.M. Kakade, A. Rakhlin, and A. Tewari. High-probability regret bounds for bandit online linear optimization. In *Proceedings of the 21st Annual Conference on Learning Theory (COLT 2008)*, pages 335–342. Omnipress, 2008.
- [5] R. Burton and R. Pemantle. Local characteristics, entropy and limit theorems for spanning trees and domino tilings via transfer-impedances. *The Annals of Probability*, 21:1329–1371, 1993.
- [6] N. Cesa-Bianchi and G. Lugosi. *Prediction, Learning, and Games*. Cambridge Univ. Press. New York, 2006.
- [7] V. Dani, T. Hayes, and S.M. Kakade. The price of bandit information for online optimization. In *NIPS 20*, pages 345–352. MIT Press, 2008.
- [8] V. Dani and T.P. Hayes. Robbing the bandit: less regret in online geometric optimization against an adaptive adversary. In *Proceedings of the 17th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2006)*, pages 937–943. ACM/SIAM, 2006.
- [9] P.G. Doyle and J.L. Snell. *Random walks and electric networks*, volume 22 of *Carus Mathematical Monographs*. Mathematical Association of America. Washington, DC, 1984.
- [10] A. György, T. Linder, G. Lugosi, and Gy. Ottucsák. The on-line shortest path problem under partial monitoring. *Journal of Machine Learning Research*, 8:2369–2403, 2007.
- [11] D.P. Helmbold and M.K. Warmuth. Learning permutations with exponential weights. In *Proceedings of the 20th Annual Conference on Learning Theory (COLT 2007)*, pages 469–483. Springer, 2007.
- [12] M. Jerrum and A. Sinclair. Polynomial-time approximation algorithms for the Ising model. *SIAM Journal on Computing*, 22:1087–1116, 1993.
- [13] M. Jerrum, A. Sinclair, and E. Vigoda. A polynomial-time approximation algorithm for the permanent of a matrix with nonnegative entries. *Journal of the ACM*, 51:671–697, 2004.
- [14] A. Kalai and S. Vempala. Efficient algorithms for the online decision problem. In *Proceedings of the 16th Annual Conference on Learning Theory (COLT 2003)*, pages 26–40. Springer, 2003.
- [15] R. Lyons and Y. Peres. *Probability on Trees and Networks*. To be published, 2008.
- [16] H.B. McMahan and A. Blum. Online geometric optimization in the bandit setting against an adaptive adversary. In *Proceedings of the 17th Annual Conference on Learning Theory (COLT 2004)*, pages 109–123. Springer, 2004.
- [17] J.G. Propp and D.B. Wilson. How to get a perfectly random sample from a generic Markov chain and generate a random spanning tree of a directed graph. *Journal of Algorithms*, 27:170–217, 1998.
- [18] D. Randall and D.B. Wilson. Sampling spin configurations of an Ising system. In *Proceedings of the 10th ACM-SIAM Symposium on Discrete Algorithms (SODA 1999)*, pages 959–960. ACM/SIAM, 1999.
- [19] E. Takimoto and M. Warmuth. Path kernels and multiplicative updates. *Journal of Machine Learning Research*, 4(5):773–818, 2004.