No Internal Regret via Neighborhood Watch

Dean P. Foster
Department of Statistics
University of Pennsylvania

Alexander Rakhlin
Department of Statistics
University of Pennsylvania

Abstract

We present an algorithm which attains $O(\sqrt{T})$ internal (and thus external) regret for finite games with partial monitoring under the *local observability condition*. Recently, this condition has been shown by Bartók, Pál, and Szepesvári [4] to imply the $O(\sqrt{T})$ rate for partial monitoring games against an i.i.d. opponent, and the authors conjectured that the same holds for non-stochastic adversaries. Our result is in the affirmative, and it completes the characterization of possible rates for finite partial-monitoring games, an open question stated by Cesa-Bianchi, Lugosi, and Stoltz [6]. Our regret guarantees also hold for the more general model of partial monitoring with random signals.

1 Introduction

Imagine playing a repeated T-round zero-sum game while receiving only some partial information about the moves of the opponent. More precisely, the game under consideration is defined by the pair (L, H), where $L \in \mathbb{R}^{N\times M}$ is the loss matrix, and $H \in \Sigma^{N\times M}$ is a signal matrix defined over some alphabet Σ. Both of these are known to the players. At time $t \in \{1, \ldots, T\}$ the row player (or, *learner*) chooses $i_t \in \{1, \ldots, N\}$ and the column player (or, *opponent*) chooses $j_t \in \{1, \ldots, M\}$. The learner then observes the value H_{i_t,j_t}, the (i_t, j_t) element of H. Neither the move of the opponent nor the incurred loss L_{i_t,j_t} is observed by the row player. The column player, on the other hand, is aware of all the past moves of the learner. In this paper, we are concerned with rates for external and internal regret achievable in this scenario by the learner.

Appearing in Proceedings of the 15^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2012, La Palma, Canary Islands. Volume XX of JMLR: W&CP XX.

The setting described above captures many problems of interest. For instance, if within each row of H the entries are distinct, the row player effectively learns the move j_t of the opponent. If, on the other hand, $L = H$, the loss is reported to the player, yet there might be ambiguity about the actual move of the column player. The latter setting corresponds to the so-called *multiarmed bandit feedback*. Other interesting scenarios include the *apple tasting* problem, the *dynamic pricing* problem, the *label efficient prediction* problem. We refer to [6] for the definitions and discussions.

The question of characterizing the rates of regret growth in terms of the matrices L and H has been raised by Cesa-Bianchi, Lugosi, and Stoltz [6]. Under a linear dependence between the matrices L and H, the authors proved $O(T^{2/3})$ rates for external regret, yet noted that there exist games with the $\Theta(\sqrt{T})$ behavior (e.g. the multiarmed bandit games). Similar distinction in available rates also appears to hold for internal regret: an $O(T^{2/3})$ upper bound was shown in [6], while the rate of $O(\sqrt{T})$ is achievable for bandit feedback by the result of Blum and Mansour [5].

Recently, Bartók, Pál, and Szepesvári in [3, 4] made key insights into the problem of partial monitoring. In particular, [4] characterized the rates for external regret against an *i.i.d.* (stochastic) opponent. The authors showed that rates can only be one of $\Theta(1), \Theta(\sqrt{T}), \Theta(T^{2/3})$ and $\Theta(T)$, and that a so-called *local observability condition* plays a key role in determining this growth behavior. In the non-stochastic (adversarial) case, however, no general characterization is available to date, with the notable exception of games with two adversarial actions [3]. As suggested by [4], to provide a complete characterization for external regret against non-stochastic opponents, it would be enough to show an upper bound of $O(\sqrt{T})$ under the *local observability condition*. The characterization would follow because [4] proves a $\Omega(T^{2/3})$ lower bound when local observability does not hold (yet the game is not hopeless with $\Omega(T)$ regret) and the upper bound of $O(T^{2/3})$ is achieved by the algorithm of Piccolboni and Schindelhauer [10] through the analysis of [6].

This paper presents an algorithm, Neighborhood Watch, with an upper bound of $O(\sqrt{T})$ for both internal and external regret against a non-stochastic opponent under the local observability condition. Together with the results mentioned above, this completes the characterization of possible rates for both internal and external regret. It is remarkable that the condition of local observability that characterizes games against a stochastic environment also characterizes games against non-stochastic opponents.

We now summarize our approach. First, we define a notion of *local* internal regret which postulates that the player does not benefit by switching any of its actions to a neighboring action. The neighbor relation is defined by the neighborhood graph of best responses to mixed strategies of the opponent. Second, we show that small *local* internal regret implies small (global) internal regret. We then present an algorithm which randomly chooses a neighborhood and then chooses an action in the neighborhood. A key property satisfied by the two-level procedure is a certain flow condition. Under this condition, external regret of sub-algorithms on local neighborhoods can be turned into a statement about local internal regret (and, hence, global internal regret). External regret of the sub-algorithms, in turn, can be upper bounded because local observability condition allows us to estimate relative losses of neighboring actions.

2 Notation and definitions

We follow the notation of [4]. Let ℓ_i denote the ith row of L. Without loss of generality, assume that each row of H contains unique sets of symbols. Let $\sigma_1, \ldots, \sigma_{s_i}$ be the list of symbols in the ith row of H. The signal matrix $S_i \in \{0,1\}^{s_i \times M}$ is defined by $S_i(k,j) = \mathbf{I}\{H_{i,j} = \sigma_k\}$ where $\mathbf{I}\{\}$ is the indicator function. For a pair i,k of actions define $S_{(i,k)} \in \{0,1\}^{(s_i+s_k)\times M}$ by stacking S_i on top of S_k. Note that, upon playing action i, the signal $H_{i,j}$ arising from the unobserved action j is equivalent to the feedback $S_i e_j$.

Let $\mathcal{C} = \{C_1, \ldots, C_N\}$ be a partition of the simplex Δ_M according to the best response (action) of the player to the mixed strategy of the adversary:

$$C_i = \{q \in \Delta_M : i \text{ is best response for } q\}.$$

Without loss of generality, we assume that no action is completely dominated by others; that is, each C_i is non-empty. Further, for simplicity we assume that $\mathcal{C}$ is indeed a partition and there are no degeneracies (we can modify the argument by defining neighborhood action sets as in [4]). *Neighboring actions* are naturally defined as those that share a boundary in the partition. Let $\mathcal{G}$ be the graph obtained by connecting the neighboring cells of the partition $\mathcal{C}$. The vertex set of $\mathcal{G}$ is precisely the set $\{1, \ldots, N\}$ of player's actions. For each action i, let the set of its neighbors N_i be called the *neighbor set*. By convention, any vertex is its own neighbor: $i \in N_i$. We will often use the terms *action* and *vertex* interchangeably, thanks to the one-to-one correspondence.

Definition 2.1 (Bartók, Pál, Szepesvári [4])**.** *The game is called locally observable if $\ell_i - \ell_j \in Im\ S^{\intercal}_{(i,j)}$ for all neighboring actions i, j.*

Under the local observability condition, for each pair of local actions i, j there exists a vector $v_{(i,j)}$ such that $\ell_j - \ell_i = S^{\intercal}_{(i,j)} v_{(i,j)}$. Since L and H are known, we can compute vectors $v_{(i,j)}$ and use them to construct unbiased estimates of true loss differences.

Notation Let $[N]$ denote the set $\{1, \ldots, N\}$. For a subset $S \subset [N]$ we use $1_S \in \{0,1\}^N$ to denote the vector with ones on the coordinates in S and zeros outside. A vector $a \in \mathbb{R}^N$ indexed by j is sometimes denoted by $[a_j]_{j\in[N]}$. The scalar product between two vectors a and b will be variously written as $a^{\intercal}b$ or $a \cdot b$. Standard basis vectors are denoted by $\{e_i\}$.

3 Internal Regret in the Neighborhood

Let ϕ be any mapping $\{1, \ldots, N\} \mapsto \{1, \ldots, N\}$ (called *departure function* [6]), and let i_t and j_t denote the moves at time t of the player and the opponent, respectively. At the end of the game, regret with respect to ϕ is calculated as the difference of the incurred cumulative cost and the cost that would have been incurred had we played action $\phi(i_t)$ instead of i_t, for all t. Let Φ be a set of departure functions. Φ-regret is defined as

$$\frac{1}{T}\sum_{t=1}^{T} c(i_t, j_t) - \inf_{\phi\in\Phi} \frac{1}{T}\sum_{t=1}^{T} c(\phi(i_t), j_t),$$

and the cost function considered in this paper is simply $c(i,j) = e_i^{\intercal} L e_j$. If $\Phi = \{\phi_k : k \in [N]\}$ consists of constant mappings $\phi_k(i) = k$, the regret is called *external*. For (global) internal regret, the set Φ consists of all departure functions $\phi_{i\to j}$ such that $\phi_{i\to j}(i) = j$ and $\phi_{i\to j}(h) = h$ for $h \neq i$.

Definition 3.1. *A departure function $\phi_{i\to j}$ is called a* local departure function *if j is a neighbor of i in the neighborhood graph $\mathcal{G}$. Regret defined with respect to the set of all local departure functions is called* local internal regret.

Under the local observability condition, we can estimate the differences in performance between the action and its neighbors in a way similar to non-stochastic multiarmed bandit methods. We can, therefore, ensure that any time we chose an action, its loss was not much more than that of any of its neighbors. That is, local observability condition leads to an algorithm with no external regret and, under the flow condition detailed later, no local internal regret. A key observation is that no local internal regret implies no global internal regret. Intuitively, this stems from the fact that the second-best-response action must be a neighbor of the best-response action. Hence, ensuring small internal regret against the neighbors is enough to guarantee small internal regret.

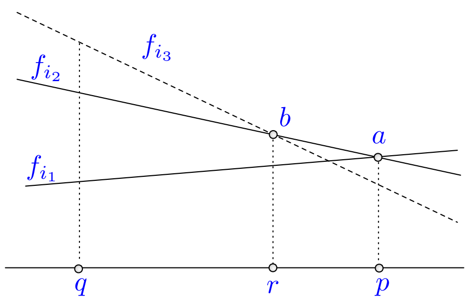

Figure 1: Illustration of the argument in Lemma 3.1: A second-best action must either be a neighbor, or it must be dominated everywhere by other actions.

Lemma 3.1. *Local internal regret is equal to internal regret up to a constant factor.*

Proof. It is enough to show that, for any distribution $q \in \Delta_M$, any best response i_1 and any second-best response i_2 are neighbors in the graph $\mathcal{G}$. By the way of contradiction, we assume that actions i_1 and i_2 are not neighbors (that is, C_{i_1} and C_{i_2} do not share a face). We will then arrive at the conclusion that i_2 must be dominated by other actions, which is a contradiction because of our assumption that no action is completely dominated (that is minorized) by others.

Let $g(s) = \min_{i \in [N]} e_i^\intercal Ls$ be the minimum loss against the mixed strategy s. Since g is a minimum of N linear functions $\{f_k(s) \triangleq (e_k^\intercal L) \cdot s\}_{k=1}^N$, it is concave and piece-wise linear. The linear parts of g correspond to the elements of the partition $\mathcal{C}$. By our assumption, $f_{i_1}(q) < f_{i_2}(q)$ and there is no hyperplane f_{i_3} achieving at q a value in the interval $(f_{i_1}(q), f_{i_2}(q))$. Let

$$S = \{(s,t) : t = f_{i_1}(s) = f_{i_2}(s) \text{ for some } s \in \Delta_M\},$$

the intersection of two hyperplanes over the simplex. Note that projection of S onto the simplex would be precisely the boundary separating C_{i_1} and C_{i_2} if these were the only two actions. This set cannot be empty, for otherwise action i_2 is dominated by i_1. Now, pick any $p \in \Delta_M$ such that $f_{i_1}(p) = f_{i_2}(p)$, and let $a = (p, f_{i_1}(p))$. We will now work with the one-dimensional problem along the line in the simplex defined by (q,p). The fact that i_1 and i_2 are not neighbors along the direction (q,p) means that there is another action i_3 such that $f_{i_3}(p) < f_{i_1}(p) = f_{i_2}(p)$. Since $f_{i_3}(q) \geq f_{i_2}(q) > f_{i_1}(q)$, there must be a point $b = (r, f_{i_3}(r)) = (r, f_{i_2}(r))$ of intersection of f_{i_3} and f_{i_2} for some $r \in [q,p]$. It is easy to see that i_2 is completely minorized along the direction (q,p): on one side of r it is dominated by i_1, while on the other — by i_3.

The argument above works for any direction from q towards the boundary between C_{i_1} and C_{i_2} if i_1 and i_2 were the only actions. Hence, i_2 is globally dominated by other actions, a contradiction. $\square$

Algorithm 1 Neighborhood Watch Algorithm

For all $i = \{1, \dots, N\}$, initialize algorithm $\mathcal{A}_i$ with $q_i^1 = x_i^1 = \mathbf{1}_{N_i}/|N_i|$
for t=1,. . . , T **do**
  Let $Q^t = [q_1^t, \dots, q_N^t]$, where q_i^t is given by $\mathcal{A}_i$
  Find p^t satisfying $p^t = Q^t p^t$
  Draw k_t from p^t
  Play I_t drawn from $q_{k_t}^t$ and obtain signal $S_{I_t} e_{j_t}$
  Run local algorithm $\mathcal{A}_{k_t}$ with the received signal
  For any $i \neq k_t$, $q_i^{t+1} \leftarrow q_i^t$
end for

4 Method and Analysis

The method is a two-level procedure motivated by Foster and Vohra [7] and Blum and Mansour [5]. The intuition stems from the following observation. Suppose for each vertex i we have a distribution $q_i \in \Delta_N$ supported on the neighbor set N_i. Let $p \in \Delta_N$ be defined by $p = Qp$ where Q is the matrix $[q_1, \dots, q_N]$. Then there are two equivalent ways of sampling an action from p. First way is to directly sample the vertex according to p. Second is to sample a vertex i according to p and then choose a vertex j within the neighbor set N_i according to q_i. Because of the stationarity (or *flow*) condition $p = Qp$, the two ways are equivalent. This idea of finding a fixed point is implicit in [7], and Blum and Mansour [5] show how stationarity can be used to convert external regret guarantees into an internal regret statement. We show here that, in fact, this conversion can be done "locally" and only with "comparison" information between neighboring actions.

Our procedure is as follows. We run N different algorithms $\mathcal{A}_1, \dots, \mathcal{A}_N$, each corresponding to a vertex and its neighbor set. Within this neighbor set we ob-

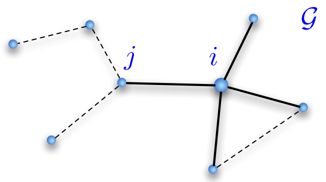

Figure 2: To each vertex i in the graph $\mathcal{G}$ we associate an algorithm $\mathcal{A}_i$. The algorithm plays an action from the distribution q_i^t over its neighborhood set N_i and receives partial information about relative loss between the node i and its neighbor. The other piece of the partial information comes from the times when a neighboring algorithm $\mathcal{A}_j$ is run and the action i is picked.

tain small regret because we can construct estimates of loss differences among the actions, thanks to the local observability condition. Each algorithm $\mathcal{A}_i$ produces a distribution $q_i^t \in \Delta_N$ at round t, reflecting the relative performance of the vertex i and its neighbors. Since $\mathcal{A}_i$ is only concerned with its local neighborhood, we require that q_i^t has support on N_i and is zero everywhere else. The meta algorithm Neighborhood Watch combines the distributions $Q^t = [q_1^t, \ldots, q_N^t]$ and computes p^t as a fixed point

$$p^t = Q^t p^t \ . \tag{1}$$

How do we choose our actions? At each round, we draw $k_t \sim p_t$ and then $I_t \sim q_{k_t}^t$ according to our two-level scheme. The action I_t is the action we play in the partial monitoring game against the adversary. Let the action played by the adversary at time t be denoted by j_t. Then the feedback we obtain is $S_{I_t} e_{j_t}$. This information is passed to $\mathcal{A}_{k_t}$ which updates the distributions $q_{k_t}^t$. The algorithmic details can be found in Section 4.2, where we construct estimates of loss differences and prove unbiasedness. In Section 4.3, we analyze regret of our method and prove Theorem 4.1. First, however, let us state the main results.

4.1 Main Result

The main result of the paper is the following internal regret guarantee.

Theorem 4.1. *Under the local observability condition, the local internal regret of Algorithm 1 is bounded as*

$$\sup_{\phi} \mathbb{E}\left\{ \sum_{t=1}^{T} (e_{I_t} - e_{\phi(I_t)})^{\intercal} L e_{j_t} \right\} \leq 4N\bar{v}\sqrt{6(\log N)T}$$

where $\bar{v} = \max_{(i,j)} \|v_{(i,j)}\|_\infty$ *and supremum is taken over all local departure functions.*

Algorithm 2 Local Algorithm $\mathcal{A}_i$

1: If $t=1$, initialize $s=1$

2: For $r \in \{\tau_i(s-1)+1, \ldots, \tau_i(s)\}$ (i.e. for all r since the last time $\mathcal{A}_i$ was run) construct

$$b_{(i,j)}^r = v_{i,j}^{\intercal} \begin{bmatrix} \mathbf{I}\{I_r = i\} S_i \\ \mathbf{I}\{k_r = i\} \mathbf{I}\{I_r = j\} S_j / q_i^r(j) \end{bmatrix} e_{j_r}$$

for all $j \in N_i$

3: Define for all $j \in N_i$,

$$h_{(i,j)}^s = \sum_{r=\tau_i(s-1)+1}^{\tau_i(s)} b_{(i,j)}^r$$

and let

$$\tilde{f}_i^s = \left[h_{(i,j)}^s \cdot \mathbf{I}\{j \in N_i\} \right]_{j \in [N]}$$

4: Pass the cost $\tilde{f}_i^s$ to a full-information online convex optimization algorithm over the simplex (e.g. Exponential Weights Algorithm) and receive the next distribution x^{s+1} supported on N_i

5: Define

$$q_i^{t+1} \leftarrow (1-\gamma)x^{s+1} + (\gamma/|N_i|) 1_{N_i}$$

6: Increase the count $s \leftarrow s+1$

The next Corollary is immediate given Lemma 3.1:

Corollary 4.1. *Internal regret of Algorithm 1 is also bounded at the same rate as in Theorem 4.1.*

We remark that high probability bounds can also be obtained in a rather straightforward manner, using, for instance, the approach of [1]. Another extension, the case of random signals, is discussed in Section 5.

4.2 Estimating loss differences

The random variable k_t drawn from p^t at time t determines which algorithm is active on the given round. Let

$$\tau_i(s) = \min\{t \ : \ s = \sum_{r=1}^{t} \mathbf{I}\{k_t = i\}\}$$

denote the (random) time when the algorithm $\mathcal{A}_i$ is invoked for the s-th time. By convention, $\tau_i(0) = 0$. Further, define

$$\pi_i(t) = \min\{t' \geq t \ : \ k_{t'} = i\}$$

to denote the next time the algorithm is run on or after time t. When invoked for the s-th time, the algorithm $\mathcal{A}_i$ constructs estimates

$$b_{(i,j)}^r \triangleq v_{i,j}^{\intercal} \begin{bmatrix} \mathbf{I}\{I_r = i\} S_i \\ \mathbf{I}\{k_r = i\} \mathbf{I}\{I_r = j\} S_j / q_i^r(j) \end{bmatrix} e_{j_r}$$

$\forall r \in \{\tau_i(s-1)+1, \ldots, \tau_i(s)\}$, $\forall j \in N_i$, for all the rounds after it has been run the last time, until (and including) the current time $r = \tau_i(s)$. We can assume $b^t_{(i,j)} = 0$ for any $j \notin N_i$. The estimates $b^t_{(i,j)}$ can be constructed by the algorithm because $S_{I_r} e_{j_r}$ is precisely the feedback given to the algorithm.

Let $\mathcal{F}_t$ be the σ-algebra generated by the random variables $\{k_1, I_1, \ldots, k_t, I_t\}$. For any t, the (conditional) expectation,

$$\begin{aligned}\mathbb{E}\left[b^t_{(i,j)}|\mathcal{F}_{t-1}\right] &= \sum_{k=1}^{N} p^t_k q^t_k(i) \cdot v^{\intercal}_{i,j} \begin{bmatrix} S_i \\ 0 \end{bmatrix} e_{j_t} \\ &\quad + p^t_i q^t_i(j) \cdot v^{\intercal}_{i,j} \begin{bmatrix} 0 \\ S_j / q^t_i(j)) \end{bmatrix} e_{j_t} \\ &= p^t_i v^{\intercal}_{i,j} S_{(i,j)} e_{j_t} \\ &= p^t_i (\ell_j - \ell_i)^{\intercal} e_{j_t} \\ &= p^t_i (e_j - e_i)^{\intercal} L e_{j_t} \qquad (2)\end{aligned}$$

where in the second equality we used the fact that $\sum_{k=1}^{N} p^t_k q^t_k(i) = p^t_i$ by stationarity (1). Thus each algorithm $\mathcal{A}_i$, on average, has access to *unbiased* estimates of the loss differences within its neighborhood set.

Recall that algorithm $\mathcal{A}_i$ is only aware of its neighborhood, and therefore we peg coordinates of q^t_i to zero outside of N_i. However, for convenience, our notation below still employs full N-dimensional vectors, and we keep in mind that only coordinates indexed by N_i are considered and modified by $\mathcal{A}_i$.

When invoked for the s-th time (that is, $t = \tau_i(s)$), $\mathcal{A}_i$ constructs linear functions (cost estimates) $\tilde{f}^s_i \in \mathbb{R}^N$ defined by

$$\tilde{f}^s_i = \left[h^s_{(i,j)} \cdot \mathbf{I}\{j \in N_i\}\right]_{j \in [N]},$$

where

$$h^s_{(i,j)} = \sum_{r=\tau_i(s-1)+1}^{\tau_i(s)} b^r_{(i,j)} .$$

Next lemma shows that $\tilde{f}^s_i \cdot q^{\tau(s)}_i$ has the same conditional expectation as the actual loss of the meta algorithm Neighborhood Watch at time $t = \tau_i(s)$. That is, by bounding expected regret of the black-box algorithm operating on $\{\tilde{f}^s_i\}$, we bound the actual regret suffered by the meta algorithm on the rounds when $\mathcal{A}_i$ was invoked.

Lemma 4.1. *Consider algorithm $\mathcal{A}_i$. It holds that*

$$\begin{aligned}&\mathbb{E}\left\{ (q^{\tau_i(s+1)}_i - e_u)^{\intercal} L e_{j_{\tau_i(s+1)}} \;\middle|\; \mathcal{F}_{\tau_i(s)} \right\} \\ &= \mathbb{E}\left\{ \tilde{f}^{s+1}_i \cdot (q^{\tau_i(s+1)}_i - e_u) \;\middle|\; \mathcal{F}_{\tau_i(s)} \right\}\end{aligned}$$

for any $u \in N_i$.

The proof is is deferred to the appendix.

4.3 Regret Analysis

For each algorithm $\mathcal{A}_i$, the estimates $\tilde{f}^s_i$ are passed to a full-information black box algorithm which works only on the coordinates N_i. From the point of view of the full-information black box, the game has length $T_i = \max\{s : \tau_i(s) \le T\}$, the (random) number of times action i has been played within T rounds.

We proceed similarly to [1]: we use a full-information online convex optimization procedure with an entropy regularizer (also known as the Exponential Weights Algorithm) which receives the vector $\tilde{f}^s_i$ and returns the next mixed strategy $x^{s+1} \in \Delta_N$ (in fact, effectively in $\Delta_{|N_i|}$). We then define

$$q^{t+1}_i = (1-\gamma)x^{s+1} + (\gamma/|N_i|)1_{N_i}$$

where γ is to be specified later. Since $\mathcal{A}_i$ is run at time t, we have $\tau_i(s) = t$ by definition. The next time $\mathcal{A}_i$ is active (that is, at time $\tau_i(s+1)$), the action $I_{\tau_i(s+1)}$ will be played as a random draw from $q^{t+1}_i = q^{\tau_i(s+1)}_i$; that is, the distribution is not modified on the interval $\{\tau_i(s)+1, \ldots, \tau_i(s+1)\}$.

We prove Theorem 4.1 by a series of lemmas. The first one is a direct consequence of an external regret bound for a Follow the Regularized Leader (FTRL) algorithm in terms of local norms [1]. For a strictly convex "regularizer" F, the local norm $\|\cdot\|_x$ is defined by $\|z\|_x = \sqrt{z^{\intercal}\nabla^2 F(x) z}$ and its dual is $\|z\|^*_x = \sqrt{z^{\intercal}\nabla^2 F(x)^{-1} z}$.

Lemma 4.2. *The full-information algorithm utilized by $\mathcal{A}_i$ has an upper bound*

$$\begin{aligned}\mathbb{E}\left\{\sum_{s=1}^{T_i} \tilde{f}^s_i \cdot (q^{\tau_i(s)}_i - e_{\phi(i)})\right\} &\le \eta \mathbb{E}\left\{\sum_{s=1}^{T_i} (\|\tilde{f}^s_i\|^*_{x^s})^2\right\} \\ &\quad + \eta^{-1}\log N + T\gamma\bar{\ell}\end{aligned}$$

on its external regret, where $\phi(i) \in N_i$ is any neighbor of i, $\bar{\ell} = \max_{i,j} L_{i,j}$, and η is a learning rate parameter to be tuned later.

Proof. Since our decision space is a simplex, it is natural to use the (negative) entropy regularizer, in which case FTRL is the same as the Exponential Weights Algorithm. According to [1, Thm 2.1], for any comparator u with zero support outside $|N_i|$, the following regret guarantee holds:

$$\sum_{s=1}^{T_i} \tilde{f}^s_i \cdot (x^s - u) \le \eta \sum_{s=1}^{T_i} (\|\tilde{f}^s_i\|^*_{x^s})^2 + \eta^{-1}\log(|N_i|) .$$

An easy calculation shows that in the case of entropy regularizer F, the Hessian $\nabla^2 F(x) = \mathrm{diag}(x_1^{-1}, x_2^{-1}, \ldots, x_N^{-1})$ and $\nabla^2 F(x)^{-1} = \mathrm{diag}(x_1, x_2, \ldots, x_N)$.

Let $\phi : \{1, \dots, N\} \mapsto \{1, \dots, N\}$ be a local departure function (see Definition 3.1). We can then write a regret guarantee

$$\sum_{s=1}^{T_i} \tilde{f}_i^s \cdot (x^s - e_{\phi(i)}) \le \eta \sum_{s=1}^{T_i} (\|\tilde{f}_i^s\|_{x^s}^*)^2 + \eta^{-1} \log(|N_i|) \ .$$

Since, in fact, we play according to a slightly modified version $q_i^{\tau_i(s)}$ of x^s, it holds that

$$\sum_{s=1}^{T_i} \tilde{f}_i^s \cdot (q_i^{\tau_i(s)} - e_{\phi(i)}) \le \eta \sum_{s=1}^{T_i} (\|\tilde{f}_i^s\|_{x^s}^*)^2 + \eta^{-1} \log(|N_i|) + \sum_{s=1}^{T_i} \tilde{f}_i^s \cdot (q_i^{\tau_i(s)} - x^s) \ .$$

Taking expectations of both sides and upper bounding $|N_i|$ by N,

$$\begin{aligned} &\mathbb{E}\left\{\sum_{s=1}^{T_i} \tilde{f}_i^s \cdot (q_i^{\tau_i(s)} - e_{\phi(i)})\right\} \\ &\le \eta \mathbb{E}\left\{\sum_{s=1}^{T_i} (\|\tilde{f}_i^s\|_{x^s}^*)^2\right\} + \eta^{-1} \log N \\ &+ \mathbb{E}\left\{\sum_{s=1}^{T_i} \tilde{f}_i^s \cdot (q_i^{\tau_i(s)} - x^s)\right\} \ . \end{aligned}$$

A proof identical to that of Lemma 4.1 gives

$$\begin{aligned} &\mathbb{E}\left\{\tilde{f}_i^s \cdot (q_i^{\tau_i(s)} - x^s) \ \middle|\ \mathcal{F}_{\tau_i(s-1)}\right\} \\ &= \mathbb{E}\left\{(q_i^{\tau_i(s)} - x^s)^\intercal L e_{j_{\tau_i(s)}} | \mathcal{F}_{\tau_i(s-1)}\right\} \\ &\le \mathbb{E}\left\{\|q_i^{\tau_i(s)} - x^s\|_1 \cdot \|L e_{j_{\tau_i(s)}}\|_\infty \ \middle|\ \mathcal{F}_{\tau_i(s-1)}\right\} \\ &\le \gamma \bar{\ell} \end{aligned}$$

for the last term, where $\bar{\ell}$ is the upper bound on the magnitude of entries of L. Putting everything together,

$$\begin{aligned} &\mathbb{E}\left\{\sum_{s=1}^{T_i} \tilde{f}_i^s \cdot (q_i^{\tau_i(s)} - e_{\phi(i)})\right\} \\ &\le \eta \mathbb{E}\left\{\sum_{s=1}^{T_i} (\|\tilde{f}_i^s\|_{x^s}^*)^2\right\} + \eta^{-1} \log N + T\gamma\bar{\ell} \end{aligned}$$

where we have upper bounded T_i by T. □

As with many bandit-type problems, effort is required to show that the variance term is controlled. This is the subject of the next lemma.

Lemma 4.3. *The variance term in the bound of Lemma 4.2 is upper bounded as*

$$\sum_{i=1}^{N} \mathbb{E}\left\{\sum_{s=1}^{T_i} (\|\tilde{f}_i^s\|_{x^s}^*)^2\right\} \le 24\bar{v}^2 NT$$

Proof. First, fix an $i \in [N]$ and consider the term $\mathbb{E}\left\{\sum_{s=1}^{T_i} (\|\tilde{f}_i^s\|_{x^s}^*)^2\right\}$. In the proof, we will sometimes omit i from the notation.

We start by observing that $\tilde{f}_i^s$ is a sum of $\tau(s) - \tau(s-1) - 1$ terms of the type $v_{i,j}^\intercal S_i e_{j_r}$ (that is, of constant magnitude) and one term of the type $v_{i,j}^\intercal S_j e_{j_r} / q_i^r(j)$. In controlling $\|\tilde{f}_i^s\|_{x^s}^*$, we therefore have two difficulties: controlling the number of constant-size terms and making sure the last term does not explode due to division by a small probability $q_i^r(j)$. The former is solved below by a careful argument, while the latter problem is solved according to usual bandit-style arguments.

More precisely, we can write $\tilde{f}_i^s = g_{\tau_i(s)}^{\tau_i(s-1)} + h^{\tau_i(s)}$ where the vectors $g_{\tau_i(s)}^{\tau_i(s-1)}, h^{\tau_i(s)} \in \mathbb{R}^N$ are defined as

$$g_{\tau_i(s)}^{\tau_i(s-1)}(j) \triangleq g^{\tau_i(s-1)}(j) \triangleq \sum_{r=\tau_i(s-1)}^{\tau_i(s)-1} \mathbf{I}\{I_r = i\} v_{i,j}^\intercal S_i e_{j_r}$$

and

$$h^{\tau_i(s)}(j) = \mathbf{I}\{I_{\tau_i(s)} = j\} v_{i,I_{\tau_i(s)}}^\intercal S_{I_{\tau_i(s)}} e_{j_{\tau_i(s)}} / q_i^{\tau_i(s)}(I_{\tau_i(s)}),$$

for $j \in N_i$ and zero otherwise. Then

$$\begin{aligned} (\|\tilde{f}_i^s\|_{x^s}^*)^2 &= (\|g^{\tau_i(s-1)} + h^{\tau_i(s)}\|_{x^s}^*)^2 \\ &\le 2(\|g^{\tau_i(s-1)}\|_{x^s}^*)^2 + 2(\|h^{\tau_i(s)}\|_{x^s}^*)^2 \end{aligned}$$

We will bound each of the two terms separately, in expectation. For the second term,

$$\begin{aligned} (\|h^{\tau_i(s)}\|_{x^s}^*)^2 &= x^s(I_\tau)(v_{i,I_\tau}^\intercal S_{I_\tau} e_{j_\tau} / q_i^\tau(I_\tau))^2 \\ &\le x^s(I_\tau)(\bar{v}/q_i^\tau(I_\tau))^2 \end{aligned}$$

where $\tau = \tau_i(s)$. Since $q_i^{\tau_i(s)} = (1-\gamma)x^s + (\gamma/|N_i|)1_{N_i}$, it is easy to verify that $x^s(I_\tau)/q_i^\tau(I_\tau) \le 2$ (whenever $\gamma < 1/2$) and thus

$$(\|h^{\tau_i(s)}\|_{x^s}^*)^2 \le 2\bar{v}^2 / q_i^\tau(I_\tau) \ .$$

The remaining division by the probability disappears under the expectation:

$$\begin{aligned} &\mathbb{E}\left\{(\|h^{\tau_i(s)}\|_{x^s}^*)^2 \ \middle|\ \sigma(k_1, I_1, \dots, k_{\tau_i(s)})\right\} \\ &\le 2\bar{v}^2 \sum_{j=1}^{N} q_i^{\tau_i(s)}(j) / q_i^{\tau_i(s)}(j) = 2N\bar{v}^2 \ . \end{aligned} \tag{3}$$

Consider now the first term. As discussed in the proof of Lemma 4.2, the inverse Hessian of the entropy function shrinks each coordinate i precisely by $x^s(i) \le 1$, implying that the local norm is dominated by the Euclidean norm :

$$\|g^{\tau_i(s-1)}\|_{x^s}^* \le \|g^{\tau_i(s-1)}\|_2.$$

It is therefore enough to upper bound $\mathbb{E}\left\{\sum_{s=1}^{T_i}\|g^{\tau_i(s)}\|_2^2\right\}$. The idea of the proof is the following. Observe that $\mathbb{P}(k_t = i|\mathcal{F}_{t-1}) = \mathbb{P}(I_t = i|\mathcal{F}_{t-1})$. Conditioned on the event that either $k_t = i$ or $I_t = i$, each of the two possibilities has probability 1/2 of occurring. Note that $g^{\tau_i(s-1)}$ inflates every time $k_t \neq i$, yet $I_t = i$ occurs. It is then easy to see that magnitude of $g^{\tau_i(s-1)}$ is unlikely to get large before algorithm $\mathcal{A}_i$ is run again. We now make this intuition precise.

The function g^t is presently defined only for those time steps when $t = \tau_i(s)$ for some s (that is, when the algorithm $\mathcal{A}_i$ is invoked). We extend this definition as follows. Let the jth coordinate of g^t be defined as

$$g^t_{\pi(t+1)}(j) \triangleq g^t(j) \triangleq \sum_{r=t}^{\pi(t+1)-1} \mathbf{I}\left\{I_r = i\right\} v_{(i,j)} S_i e_{j_r}$$

for $j \in N_i$ and 0 otherwise. The function g^t can be thought of as accumulating partial pieces of information on rounds when $I_t = i$ until $k_t = i$ occurs. Let us now define an analogue of τ and π for the event that *either* $I_t = i$ or $k_t = i$:

$$\gamma_i(s) = \min\left\{t \ : \ s = \sum_{r=1}^{t} \mathbf{I}\left\{k_t = i \text{ or } I_t = i\right\}\right\}$$

Further, for any t, let

$$\nu_i(t) = \min\{t' \geq t : k_t = i \text{ or } I_t = i\},$$

the next time occurrence of the event $\{k_\tau = i$ or $I_\tau = i\}$ on or after t. Let

$$\mathcal{I} = \mathbf{I}\left\{\nu_i(t) \neq \pi_i(t)\right\}$$

be the indicator of the event that the first time after t that $\{k_\tau = i$ or $I_\tau = i\}$ occurred it was also the case that the algorithm was not run (i.e. $k_\tau \neq i$). Note that $g^t(j)$ can now be written recursively as

$$g^t(j) = \mathcal{I}\cdot\left[v_{(i,j)} S_i e_{j_{\nu(t)}} + g^{\nu(t)+1}_{\pi(\nu(t)+1)}(j)\right].$$

As argued before, $\mathbb{P}(\mathcal{I} = 1|\mathcal{F}_{t-1}) = 1/2$. We will now show that $\mathbb{E}\left\{g^t(j) \mid \mathcal{F}_{t-1}\right\} \leq 2\bar{v}$ by the following inductive argument, whose base case trivially holds for $t = T$:

$$\begin{aligned}
&\mathbb{E}\left\{g^t(j) \mid \mathcal{F}_{t-1}\right\} \\
&= \mathbb{E}\left\{\mathbb{E}\left\{\mathcal{I}\cdot\left[v_{(i,j)} S_i e_{j_{\nu(t)}} + g^{\nu(t)+1}(j)\right] \Big| \mathcal{F}_{\nu(t)}\right\} \Big| \mathcal{F}_{t-1}\right\} \\
&= \mathbb{E}\left\{\mathcal{I} v_{(i,j)} S_i e_{j_{\nu(t)}} + \mathcal{I}\mathbb{E}\left\{g^{\nu(t)+1}(j) \Big| \mathcal{F}_{\nu(t)}\right\} \Big| \mathcal{F}_{t-1}\right\} \\
&\leq \bar{v} + \mathbb{E}\left\{\mathcal{I} g^{\nu(t)+1}(j) \Big| \mathcal{F}_{t-1}\right\} \\
&= \bar{v} + \mathbb{E}\Big\{\mathcal{I}\,\underbrace{\mathbb{E}\left[g^{\nu(t)+1}(j) \Big| \mathcal{F}_{\nu(t)}\right]}_{\leq 2\bar{v} \text{ by induction}} \Big| \mathcal{F}_{t-1}\Big\}
\end{aligned}$$

The last quantity is upper bounded by

$$\bar{v} + \mathbb{E}\left\{\mathcal{I} \mid \mathcal{F}_{t-1}\right\} 2\bar{v} \leq \bar{v} + (1/2)2\bar{v} = 2\bar{v}\ .$$

The expected value of $(g^t(j))^2$ can be controlled in a similar manner. To ease the notation, let $z = v_{(i,j)} S_i e_{j_{\nu(t)}}$. Using the upper bound for the conditional expectation of $g^t(j)$ calculated above,

$$\begin{aligned}
&\mathbb{E}\left\{(g^t(j))^2 \mid \mathcal{F}_{t-1}\right\} \\
&= \mathbb{E}\left\{\mathcal{I}\cdot\left(z^2 + (g^{\nu(t)+1}(j))^2 + 2zg^{\nu(t)+1}(j)\right) \Big| \mathcal{F}_{t-1}\right\} \\
&= \mathbb{E}\Big\{\mathcal{I}z^2 + \mathcal{I}\mathbb{E}\left\{(g^{\nu(t)+1}(j))^2 \Big| \mathcal{F}_{\nu(t)}\right\} \\
&\qquad + 2\mathcal{I}z\mathbb{E}\left\{g^{\nu(t)+1}(j) \Big| \mathcal{F}_{\nu(t)}\right\} \Big| \mathcal{F}_{t-1}\Big\} \\
&\leq 5\bar{v}^2 + \mathbb{E}\left\{\mathcal{I}\mathbb{E}\left\{(g^{\nu(t)+1}(j))^2 \Big| \mathcal{F}_{\nu(t)}\right\} \Big| \mathcal{F}_{t-1}\right\}
\end{aligned}$$

The argument now proceeds with backward induction exactly as above. We conclude that

$$\mathbb{E}\left\{(g^t(j))^2 \mid \mathcal{F}_{t-1}\right\} \leq 10\bar{v}^2$$

and, hence,

$$\mathbb{E}\left\{\|g^{\tau_i(s-1)}\|_2^2\right\} \leq 10N\bar{v}^2$$

Together with (3), we conclude that

$$\mathbb{E}\left\{(\|\tilde{f}_i^s\|_{x^s}^*)^2\right\} \leq 2(2N\bar{v}^2 + 10N\bar{v}^2) = 24\bar{v}^2 N.$$

Summing over $t = 1, \ldots, T$ and observing that only one algorithm is run at any time t proves the statement. □

Proof of Theorem 4.1. The flow condition $p^t = Q^t p^t$ comes in crucially in several places throughout the proofs, and the next argument is one of them. Observe that

$$\begin{aligned}
\mathbb{E}\left\{e_{\phi(I_t)} \mid \mathcal{F}_{t-1}\right\} &= \sum_{k=1}^{N}\sum_{i=1}^{N} p_k^t q_k^t(i) e_{\phi(i)} \\
&= \sum_{i=1}^{N} e_{\phi(i)} \sum_{k=1}^{N} p_k^t q_k^t(i) = \sum_{i=1}^{N} e_{\phi(i)} p_i^t = \mathbb{E}\left\{e_{\phi(k_t)} \mid \mathcal{F}_{t-1}\right\}
\end{aligned}$$

and thus

$$\begin{aligned}
\mathbb{E}\left\{\sum_{t=1}^{T} e_{\phi(I_t)}^{\intercal} L e_{j_t}\right\} &= \mathbb{E}\left\{\sum_{t=1}^{T} \mathbb{E}\left\{e_{\phi(I_t)} \mid \mathcal{F}_{t-1}\right\}^{\intercal} L e_{j_t}\right\} \\
&= \mathbb{E}\left\{\sum_{t=1}^{T} \mathbb{E}\left\{e_{\phi(k_t)} \mid \mathcal{F}_{t-1}\right\}^{\intercal} L e_{j_t}\right\} \\
&= \mathbb{E}\left\{\sum_{t=1}^{T} e_{\phi(k_t)}^{\intercal} L e_{j_t}\right\}
\end{aligned}$$

It is because of this equality that external regret with respect to the local neighborhood can be turned into local internal regret. We have that

$$\mathbb{E}\left\{\sum_{t=1}^{T}(e_{I_t}-e_{\phi(I_t)})^{\intercal}Le_{j_t}\right\}=\mathbb{E}\left\{\sum_{t=1}^{T}(e_{I_t}-e_{\phi(k_t)})^{\intercal}Le_{j_t}\right\}$$
$$=\mathbb{E}\left\{\sum_{t=1}^{T}(q_{k_t}^t-e_{\phi(k_t)})^{\intercal}Le_{j_t}\right\}$$
$$=\sum_{i=1}^{N}\mathbb{E}\left\{\sum_{t=1}^{T}\mathbf{I}\left\{k_t=i\right\}(q_i^t-e_{\phi(i)})^{\intercal}Le_{j_t}\right\}$$

By Lemma 4.1,

$$\mathbb{E}\left\{(q_i^{\tau_i(s)}-e_{\phi(i)})^{\intercal}Le_{j_{\tau_i(s)}}|\mathcal{F}_{\tau_i(s-1)}\right\}$$
$$=\mathbb{E}\left\{\tilde{f}_i^s\cdot(q_i^{\tau_i(s)}-e_{\phi(i)})\;\middle|\;\mathcal{F}_{\tau_i(s-1)}\right\}$$

and so by Lemma 4.2

$$E\left\{\sum_{t=1}^{T}(e_{I_t}-e_{\phi(I_t)})^{\intercal}Le_{j_t}\right\}$$
$$=\sum_{i=1}^{N}\mathbb{E}\left\{\sum_{s=1}^{T_i}\tilde{f}_i^s\cdot(q_i^{\tau_i(s)}-e_{\phi(i)})\right\}$$
$$\leq\eta\sum_{i=1}^{N}\mathbb{E}\left\{\sum_{s=1}^{T_i}(\|\tilde{f}_i^s\|_{x^s}^*)^2\right\}+N(\eta^{-1}\log N+T\gamma\bar{\ell})$$

With the help of Lemma 4.3,

$$\mathbb{E}\left\{\sum_{t=1}^{T}(e_{I_t}-e_{\phi(I_t)})^{\intercal}Le_{j_t}\right\}$$
$$\leq\eta 24\bar{v}^2NT+N(\eta^{-1}\log N+T\gamma\bar{\ell})$$
$$=4N\bar{v}\sqrt{6(\log N)T}+TN\gamma\bar{\ell}$$

for the setting of $\eta=\sqrt{\frac{\log N}{24\bar{v}^2T}}$.

We remark that for the purposes of "in expectation" bounds, we can simply set $\gamma=0$ and still get $O(\sqrt{T})$ guarantees (see [1]). This point is obscured by the fact that the original algorithm of Auer et al [2] uses the same parameter for the learning rate η and exploration γ. If these are separated, the "in expectation" analysis of [2] can be also done with $\gamma=0$. However, to prove high probability bounds on regret, a setting of $\gamma\propto T^{-1/2}$ is required. Using the techniques in [1], the high-probability extension of results in this paper is straightforward (tails for the terms $\|g^{\tau_i(s-1)}\|_2^2$ in Lemma 4.3 are controlled without much difficulty). □

5 Random Signals

We now briefly consider the setting of partial monitoring with random signals, studied by Rustichini [11], Lugosi, Mannor, and Stoltz [8], and Perchet [9]. Without much modification of the above arguments, the local observability condition yet again yields $O(\sqrt{T})$ internal regret.

Suppose that instead of receiving deterministic feedback $H_{i,j}$, the decision maker now receives a random signal $d_{i,j}$ drawn according to the distribution $H_{i,j}\in\Delta(\Sigma)$ over the signals. In the problem of deterministic feedback studied in the paper so far, the signal $H_{i,j}=\sigma$ was identified with the Dirac distribution δ_σ.

Given the matrix H of distributions on Σ, we can construct, for each row i, a matrix $\Xi_i\in\mathbb{R}^{s_i\times M}$ as $\Xi_i(k,j)\triangleq H_{i,j}(\sigma_k)$ where the set $\sigma_1,\ldots,\sigma_{s_i}$ is the union of supports of $H_{i,1},\ldots,H_{i,M}$. Columns of Ξ_i are now distributions over signals. Given the actions I_t and j_t of the player and the opponent, the feedback provided to the player can be equivalently written as $S_{I_t}^t e_{j_t}$ where each column r of the random matrix $S_{I_t}^t\in\mathbb{R}^{s_i\times M}$ is a standard unit vector drawn independently according to the distribution given by the column r of Ξ_i. Hence, $\mathbb{E}S_i^t=\Xi_i$.

As before, the matrix $\Xi_{(i,j)}$ is constructed by stacking Ξ_i on top of Ξ_j. The local observability condition, adapted to the case of random signals, can now be stated as: $\ell_i-\ell_j\in\operatorname{Im}\Xi_{(i,j)}^{\intercal}$ for all neighboring actions i,j.

Let us specify the few places where the analysis slightly differs from the arguments of the paper. Since we now have an extra (independent) source of randomness, we define $\mathcal{F}_t$ to be the σ-algebra generated by the random variables $\{k_1,I_1,S^1\ldots,k_t,I_t,S^t\}$ where S^t is the random matrix obtained by stacking all S_i^t. We now define the estimates

$$b_{(i,j)}^r\triangleq v_{i,j}^{\intercal}\left[\begin{array}{c}\mathbf{I}\left\{I_r=i\right\}S_i^t\\ \mathbf{I}\left\{k_r=i\right\}\mathbf{I}\left\{I_r=j\right\}S_j^t/q_i^r(j)\end{array}\right]e_{j_r}$$

$\forall r\in\{\tau_i(s-1)+1,\ldots,\tau_i(s)\}$, $\forall j\in N_i$, with the only modification that S_i^t and S_j^t are now random variables. Equation (2) now reads

$$\mathbb{E}\left[b_{(i,j)}^t|\mathcal{F}_{t-1}\right]=\sum_{k=1}^{N}p_k^tq_k^t(i)\cdot v_{i,j}^{\intercal}\left[\begin{array}{c}\Xi_i\\ 0\end{array}\right]e_{j_t}$$
$$+p_i^tq_i^t(j)\cdot v_{i,j}^{\intercal}\left[\begin{array}{c}0\\ \Xi_j/q_i^t(j))\end{array}\right]e_{j_t}$$
$$=p_i^tv_{i,j}^{\intercal}\Xi_{(i,j)}e_{j_t}\tag{4}$$
$$=p_i^t(e_j-e_i)^{\intercal}Le_{j_t}\tag{5}$$

The rest of the analysis follows as in Section 4.3, with Ξ in place of S.

Acknowledgements

We thank Vianney Perchet and Gilles Stoltz for their helpful comments on the first draft of this paper.
A. Rakhlin gratefully acknowledges the support of NSF under grant CAREER DMS-0954737.

References

[1] J. Abernethy and A. Rakhlin. Beating the adaptive bandit with high probability. In *COLT*, 2009.

[2] P. Auer, N. Cesa-Bianchi, Y. Freund, and R.E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM Journal on Computing*, 32(1):48–77, 2003.

[3] G. Bartók, D. Pál, and C. Szepesvári. Toward a classification of finite partial-monitoring games. In *Algorithmic Learning Theory*, pages 224–238. Springer, 2010.

[4] G. Bartók, D. Pál, and C. Szepesvári. Minimax regret of finite partial-monitoring games in stochastic environments. In *Conference on Learning Theory*, 2011.

[5] A. Blum and Y. Mansour. From external to internal regret. *Journal of Machine Learning Research*, 8(1307-1324):3–8, 2007.

[6] N. Cesa-Bianchi, G. Lugosi, and G. Stoltz. Regret minimization under partial monitoring. *Mathematics of Operations Research*, 31(3):562–580, 2006.

[7] D.P. Foster and R.V. Vohra. Calibrated learning and correlated equilibrium. *Games and Economic Behavior*, 21(1-2):40–55, 1997.

[8] G. Lugosi, S. Mannor, and G. Stoltz. Strategies for prediction under imperfect monitoring. *Math. Oper. Res*, 33:513–528, 2008.

[9] V. Perchet. Internal regret with partial monitoring: Calibration-based optimal algorithms. *Journal of Machine Learning Research*, 12:1893–1921, 2011.

[10] A. Piccolboni and C. Schindelhauer. Discrete prediction games with arbitrary feedback and loss. In *Computational Learning Theory*, pages 208–223. Springer, 2001.

[11] A. Rustichini. Minimizing regret: The general case. *Games and Economic Behavior*, 29(1-2):224–243, 1999.