Bandits with Delayed, Aggregated Anonymous Feedback

Ciara Pike-Burke [1] **Shipra Agrawal** [2] **Csaba Szepesvári** [3 4] **Steffen Grünewälder** [1]

Abstract

We study a variant of the stochastic K-armed bandit problem, which we call "bandits with delayed, aggregated anonymous feedback". In this problem, when the player pulls an arm, a reward is generated, however it is not immediately observed. Instead, at the end of each round the player observes only the sum of a number of previously generated rewards which happen to arrive in the given round. The rewards are stochastically delayed and due to the aggregated nature of the observations, the information of which arm led to a particular reward is lost. The question is what is the cost of the information loss due to this delayed, aggregated anonymous feedback? Previous works have studied bandits with stochastic, non-anonymous delays and found that the regret increases only by an additive factor relating to the expected delay. In this paper, we show that this additive regret increase can be maintained in the harder delayed, aggregated anonymous feedback setting when the expected delay (or a bound on it) is known. We provide an algorithm that matches the worst case regret of the non-anonymous problem exactly when the delays are bounded, and up to logarithmic factors or an additive variance term for unbounded delays.

1. Introduction

The stochastic multi-armed bandit (MAB) problem is a prominent framework for capturing the exploration-exploitation tradeoff in online decision making and experiment design. The MAB problem proceeds in discrete sequential rounds, where in each round, the player pulls one of the K possible arms. In the classic stochastic MAB setting, the player immediately observes stochastic feedback from the pulled arm in the form of a 'reward' which can be used to improve the decisions in subsequent rounds. One of the main application areas of MABs is in online advertising. Here, the arms correspond to adverts, and the feedback would correspond to *conversions*, that is users buying a product after seeing an advert. However, in practice, these conversions may not necessarily happen immediately after the advert is shown, and it may not always be possible to assign the credit of a sale to a particular showing of an advert. A similar challenge is encountered in many other applications, e.g., in personalized treatment planning, where the effect of a treatment on a patient's health may be delayed, and it may be difficult to determine which out of several past treatments caused the change in the patient's health; or, in content design applications, where the effects of multiple changes in the website design on website traffic and footfall may be delayed and difficult to distinguish.

In this paper, we propose a new bandit model to handle online problems with such 'delayed, aggregated and anonymous' feedback. In our model, a player interacts with an environment of K actions (or arms) in a sequential fashion. At each time step the player selects an action which leads to a reward generated at random from the underlying reward distribution. At the same time, a nonnegative random integer-valued delay is also generated i.i.d. from an underlying delay distribution. Denoting this delay by $\tau \geq 0$ and the index of the current round by t, the reward generated in round t will arrive at the end of the $(t+\tau)$th round. At the end of each round, the player observes only the *sum* of all the rewards that arrive in that round. Crucially, the player does not know which of the past plays have contributed to this aggregated reward. We call this problem *multi-armed bandits with delayed, aggregated anonymous feedback* (MABDAAF). As in the standard MAB problem, in MABDAAF, the goal is to maximize the cumulative reward from T plays of the bandit, or equivalently to minimize the regret. The regret is the total difference between the reward of the optimal action and the actions taken.

If the delays are all zero, the MABDAAF problem reduces to the standard (stochastic) MAB problem, which has been studied considerably (e.g., Thompson, 1933; Lai & Robbins, 1985; Auer et al., 2002; Bubeck & Cesa-Bianchi,

[1] Department of Mathematics and Statistics, Lancaster University, Lancaster, UK [2] Department of Industrial Engineering and Operations Research, Columbia University, New York, NY, USA [3] DeepMind, London, UK [4] Department of Computing Science, University of Alberta, Edmonton, AB, Canada. Correspondence to: Ciara Pike-Burke <ciara.pikeburke@gmail.com>.

Proceedings of the 35^{th} International Conference on Machine Learning, Stockholm, Sweden, PMLR 80, 2018. Copyright 2018 by the author(s).

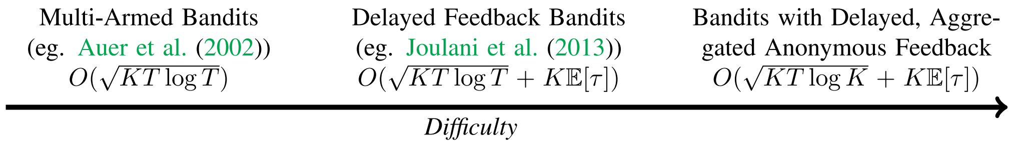

Multi-Armed Bandits
(eg. Auer et al. (2002))
$O(\sqrt{KT\log T})$
Delayed Feedback Bandits
(eg. Joulani et al. (2013))
$O(\sqrt{KT\log T} + K\mathbb{E}[\tau])$
Bandits with Delayed, Aggregated Anonymous Feedback
$O(\sqrt{KT\log K} + K\mathbb{E}[\tau])$
Difficulty

Figure 1: The relative difficulties and problem independent regret bounds of the different problems. For MABDAAF, our algorithm uses knowledge of $\mathbb{E}[\tau]$ and a mild assumption on a delay bound, which is not required by Joulani et al. (2013).

2012). Compared to the MAB problem, the job of the player in our problem appears to be significantly more difficult since the player has to deal with *(i)* that some feedback from the previous pulls may be *missing* due to the delays, and *(ii)* that the feedback takes the form of the sum of an *unknown number* of rewards of *unknown origin*.

An easier problem is when the observations are delayed, but they are *non-aggregated* and *non-anonymous*: that is, the player has to only deal with challenge (i) and not (ii). Here, the player receives delayed feedback in the shape of action-reward pairs that inform the player of both the individual reward and which action generated it. This problem, which we shall call the *(non-anonymous) delayed feedback bandit problem*, has been studied by Joulani et al. (2013), and later followed up by Mandel et al. (2015) (for bounded delays). Remarkably, they show that compared to the standard (non-delayed) stochastic MAB setting, the regret will increase only additively by a factor that scales with the expected delay. For delay distributions with a finite expected delay, $\mathbb{E}[\tau]$, the worst case regret scales with $O(\sqrt{KT\log T} + K\mathbb{E}[\tau])$. Hence, the price to pay for the delay in receiving the observations is negligible. QPM-D of Joulani et al. (2013) and SBD of Mandel et al. (2015) place received rewards into queues for each arm, taking one whenever a base bandit algorithm suggests playing the arm. Throughout, we take UCB1 (Auer et al., 2002) as the base algorithm in QPM-D. Joulani et al. (2013) also present a direct modification of the UCB1 algorithm. All of these algorithms achieve the stated regret. None of them require *any* knowledge of the delay distributions, but they all rely heavily upon the non-anonymous nature of the observations.

While these results are encouraging, the assumption that the rewards are observed individually in a non-anonymous fashion is limiting for most practical applications with delays (e.g., recall the applications discussed earlier). How big is the price to be paid for receiving only aggregated anonymous feedback? Our main result is to prove that essentially there is no extra price to be paid provided that the value of the expected delay (or a bound on it) is available. In particular, this means that detailed knowledge of which action led to a particular delayed reward can be replaced by the much weaker requirement that the expected delay, or a bound on it, is known. Fig. 1 summarizes the relationship between the non-delayed, the delayed and the new problem by showing the leading terms of the regret. In all cases, the dominant term is $\sqrt{KT}$. Hence, asymptotically, the delayed, aggregated anonymous feedback problem is no more difficult than the standard multi-armed bandit problem.

1.1. Our Techniques and Results

We now consider what sort of algorithm will be able to achieve the aforementioned results for the MABDAAF problem. Since the player only observes delayed, aggregated anonymous rewards, the first problem we face is how to even estimate the mean reward of individual actions. Due to the delays and anonymity, it appears that to be able to estimate the mean reward of an action, the player wants to have played it consecutively for long stretches. Indeed, if the stretches are sufficiently long compared to the mean delay, the observations received during the stretch will mostly consist of rewards of the action played in that stretch. This naturally leads to considering algorithms that *switch actions rarely* and this is indeed the basis of our approach.

Several popular MAB algorithms are based on choosing the action with the largest upper confidence bound (UCB) in each round (e.g., Auer et al., 2002; Cappé et al., 2013). UCB-style algorithms tend to switch arms frequently and will only play the optimal arm for long stretches if a unique optimal arm exists. Therefore, for MABDAAF, we will consider alternative algorithms where arm-switching is more tightly controlled. The design of such algorithms goes back at least to the work of Agrawal et al. (1988) where the problem of bandits with switching costs was studied. The general idea of these rarely switching algorithms is to gradually eliminate suboptimal arms by playing arms in phases and comparing each arm's upper confidence bound to the lower confidence bound of a leading arm at the end of each phase. Generally, this sort of rarely switching algorithm switches arms only $O(\log T)$ times. We base our approach on one such algorithm, the so-called Improved UCB[1] algorithm of Auer & Ortner (2010).

Using a rarely switching algorithm alone will not be sufficient for MABDAAF. The remaining problem, and where the bulk of our contribution lies, is to construct appropri-

[1] The adjective "Improved" indicates that the algorithm improves upon the regret bounds achieved by UCB1. The improvement replaces $\log(T)/\Delta_j$ by $\log(T\Delta_j^2)/\Delta_j$ in the regret bound.

ate confidence bounds and adjust the length of the periods of playing each arm to account for the delayed, aggregated anonymous feedback. In particular, in the confidence bounds attention must be paid to fine details: it turns out that unless the variance of the observations is dealt with, there is a blow-up by a multiplicative factor of K. We avoid this by an improved analysis involving Freedman's inequality (Freedman, 1975). Further, to handle the dependencies between the number of plays of each arm and the past rewards, we combine Doob's optimal skipping theorem (Doob, 1953) and Azuma-Hoeffding inequalities. Using a rarely switching algorithm for MABDAAF means we must also consider the dependencies between the elimination of arms in one phase and the corruption of observations in the next phase (ie. past plays can influence both whether an arm is still active and the corruption of its next plays). We deal with this through careful algorithmic design.

Using the above, we provide an algorithm that achieves worst case regret of $O(\sqrt{KT\log K} + K\mathbb{E}[\tau]\log T)$ using only knowledge of the expected delay, $\mathbb{E}[\tau]$. We then show that this regret can be improved by using a more careful martingale argument that exploits the fact that our algorithm is designed to remove most of the dependence between the corruption of future observations and elimination of arms. Particularly, if the delays are bounded with known bound $0 \leq d \leq \sqrt{T/K}$, we can recover worst case regret of $O(\sqrt{KT\log K} + K\mathbb{E}[\tau])$, matching that of Joulani et al. (2013). If the delays are unbounded but have known variance $\mathbb{V}(\tau)$, we show that the problem independent regret can be reduced to $O(\sqrt{KT\log K} + K\mathbb{E}[\tau] + K\mathbb{V}(\tau))$.

1.2. Related Work

We have already discussed several of the most relevant works to our own. However, there has also been other work looking at different flavors of the bandit problem with delayed (non-anonymous) feedback. For example, Neu et al. (2010) and Cesa-Bianchi et al. (2016) consider non-stochastic bandits with fixed constant delays; Dudik et al. (2011) look at stochastic contextual bandits with a constant delay and Desautels et al. (2014) consider Gaussian Process bandits with a bounded stochastic delay. The general observation that delay causes an additive regret penalty in stochastic bandits and a multiplicative one in adversarial bandits is made in Joulani et al. (2013). The empirical performance of K-armed stochastic bandit algorithms in delayed settings was investigated in Chapelle & Li (2011). A further related problem is the 'batched bandit' problem studied by Perchet et al. (2016). Here the player must fix a set of time points at which to collect feedback on all plays leading up to that point. Vernade et al. (2017) consider delayed Bernoulli bandits where some observations could also be censored (e.g., no conversion is ever actually observed if the delay exceeds some threshold) but require complete knowledge of the delay distribution. Crucially, here and in all the aforementioned works, the feedback is always assumed to take the form of arm-reward pairs and knowledge of the assignment of rewards to arms underpins the suggested algorithms, rendering them unsuitable for MABDAAF. To the best of our knowledge, ours is the first work to develop algorithms to deal with delayed, aggregated anonymous feedback in the bandit setting.

1.3. Organization

The reminder of this paper is organized as follows: In the next section (Section 2) we give the formal problem definition. We present our algorithm in Section 3. In Section 4, we discuss the performance of our algorithm under various delay assumptions; known expectation, bounded support with known bound and expectation, and known variance and expectation. This is followed by a numerical illustration of our results in Section 5. We conclude in Section 6.

2. Problem Definition

There are $K > 1$ actions or arms in the set $\mathcal{A}$. Each action $j \in \mathcal{A}$ is associated with a reward distribution ζ_j and a delay distribution δ_j. The reward distribution is supported in $[0, 1]$ and the delay distribution is supported on $\mathbb{N} \doteq \{0, 1, \dots\}$. We denote by μ_j the mean of ζ_j, $\mu^* = \mu_{j^*} = \max_j \mu_j$ and define $\Delta_j = \mu^* - \mu_j$ to be the *reward gap*, that is the expected loss of reward each time action j is chosen instead of an optimal action. Let $(R_{l,j}, \tau_{l,j})_{l\in\mathbb{N}, j\in\mathcal{A}}$ be an infinite array of random variables defined on the probability space (Ω, Σ, P) which are mutually independent. Further, $R_{l,j}$ follows the distribution ζ_j and $\tau_{l,j}$ follows the distribution δ_j. The meaning of these random variables is that if the player plays action j at time l, a payoff of $R_{l,j}$ will be added to the aggregated feedback that the player receives at the end of the $(l + \tau_{l,j})$th play. Formally, if $J_l \in \mathcal{A}$ denotes the action chosen by the player at time $l = 1, 2, \dots$, then the observation received at the end of the tth play is

$$X_t = \sum_{l=1}^{t} \sum_{j=1}^{K} R_{l,j} \times \mathbb{I}\{l + \tau_{l,j} = t, J_l = j\}.$$

For the remainder, we will consider i.i.d. delays across arms. We also assume discrete delay distributions, although most results hold for continuous delays by redefining the event $\{\tau_{l,j} = t - l\}$ as $\{t - l - 1 < \tau_{l,j} \leq t - l\}$ in X_t. In our analysis, we will sum over stochastic index sets. For a stochastic index set I and random variables $\{Z_n\}_{n\in\mathbb{N}}$ we denote such sums as $\sum_{t\in I} Z_t \doteq \sum_{t\in\mathbb{N}} \mathbb{I}\{t \in I\} \times Z_t$.

Regret definition In most bandit problems, the regret is the cumulative loss due to not playing an optimal action.

In the case of delayed feedback, there are several possible ways to define the regret. One option is to consider only the loss of the rewards *received* before horizon T (as in Vernade et al. (2017)). However, we will not use this definition. Instead, as in Joulani et al. (2013), we consider the loss of all *generated* rewards and define the (pseudo-)regret by

$$\mathfrak{R}_T = \sum_{t=1}^{T}(\mu^* - \mu_{J_t}) = T\mu^* - \sum_{t=1}^{T}\mu_{J_t}.$$

This includes the rewards received after the horizon T and does not penalize large delays as long as an optimal action is taken. This definition is natural since, in practice, the player should eventually receive all outstanding reward.

Lai & Robbins (1985) showed that the regret of any algorithm for the standard MAB problem must satisfy,

$$\liminf_{T\to\infty} \frac{\mathbb{E}[\mathfrak{R}_T]}{\log(T)} \geq \sum_{j:\Delta_j>0} \frac{\Delta_j}{KL(\zeta_j,\zeta^*)}, \qquad (1)$$

where $KL(\zeta_j,\zeta^*)$ is the KL-divergence between the reward distributions of arm j and an optimal arm. Theorem 4 of Vernade et al. (2017) shows that the lower bound in (1) also holds for delayed feedback bandits with no censoring and their alternative definition of regret. We therefore suspect (1) should hold for MABDAAF. However, due to the specific problem structure, finding a lower bound for MABDAAF is non-trivial and remains an open problem.

Assumptions on delay distribution For our algorithm for MABDAAF, we need some assumptions on the delay distribution. We assume that the expected delay, $\mathbb{E}[\tau]$, is bounded and known. This quantity is used in the algorithm.

Assumption 1 *The expected delay $\mathbb{E}[\tau]$ is bounded and known to the algorithm.*

We then show that under some further mild assumptions on the delay, we can obtain better algorithms with even more efficient regret guarantees. We consider two settings: delay distributions with bounded support, and bounded variance.

Assumption 2 (Bounded support) *There exists some constant $d > 0$ known to the algorithm such that the support of the delay distribution is bounded by* d.

Assumption 3 (Bounded variance) *The variance, $\mathbb{V}(\tau)$, of the delay is bounded and known to the algorithm.*

In fact the known expected value and known variance assumption can be replaced by a 'known upper bound' on the expected value and variance respectively. However, for simplicity, in the remaining, we use $\mathbb{E}[\tau]$ and $\mathbb{V}(\tau)$ directly. The next sections provide algorithms and regret analysis for different combinations of the above assumptions.

3. Our Algorithm

Our algorithm is a phase-based elimination algorithm based on the Improved UCB algorithm by Auer & Ortner (2010). The general structure is as follows. In each phase, each arm is played multiple times consecutively. At the end of the phase, the observations received are used to update mean estimates, and any arm with an estimated mean below the best estimated mean by a gap larger than a 'separation gap tolerance' is eliminated. This separation tolerance is decreased exponentially over phases, so that it is very small in later phases, eliminating all but the best arm(s) with high probability. An alternative formulation of the algorithm is that at the end of a phase, any arm with an upper confidence bound lower than the best lower confidence bound is eliminated. These confidence bounds are computed so that with high probability they are more (less) than the true mean, but within the separation gap tolerance. The phase lengths are then carefully chosen to ensure that the confidence bounds hold. Here we assume that the horizon T is known, but we expect that this can be relaxed as in Auer & Ortner (2010).

Algorithm overview Our algorithm, ODAAF, is given in Algorithm 1. It operates in phases $m = 1, 2, \ldots$. Define $\mathcal{A}_m$ to be the set of active arms in phase m. The algorithm takes parameter n_m which defines the number of samples of each active arm required by the end of phase m.

In Step 1 of phase m of the algorithm, each active arm j is played repeatedly for $n_m - n_{m-1}$ steps. We record all timesteps where arm j was played in the first m phases (excluding bridge periods) in the set $T_j(m)$. The active arms are played in any arbitrary but fixed order. In Step 2, the n_m observations from timesteps in $T_j(m)$ are averaged to obtain a new estimate $\bar{X}_{m,j}$ of μ_j. Arm j is eliminated if $\bar{X}_{m,j}$ is further than $\tilde{\Delta}_m$ from $\max_{j'\in\mathcal{A}_m} \bar{X}_{m,j'}$.

A further nuance in the algorithm structure is the '*bridge period*' (see Figure 2). The algorithm picks an active arm $j \in \mathcal{A}_{m+1}$ to play in this bridge period for $n_m - n_{m-1}$ steps. The observations received during the bridge period are discarded, and not used for computing confidence intervals. The significance of the bridge period is that it breaks the dependence between confidence intervals calculated in phase m and the delayed payoffs seeping into phase $m+1$. Without the bridge period this dependence would impair the validity of our confidence intervals. However, we suspect that, in practice, it may be possible to remove it.

Choice of $\mathbf{n_m}$ A key element of our algorithm design is the careful choice of n_m. Since n_m determines the number of times each active (possibly suboptimal) arm is played, it clearly has an impact on the regret. Furthermore, n_m needs to be chosen so that the confidence bounds on the estimation error hold with given probability. The main chal-

Algorithm 1 Optimism for Delayed, Aggregated Anonymous Feedback (ODAAF)

```
Input: A set of arms, $\mathcal{A}$; a horizon, $T$; choice of $n_m$ for each phase $m = 1, 2, \ldots$.
Initialization: Set $\tilde{\Delta}_1 = 1/2$ (tolerance), the set of active arms $\mathcal{A}_1 = \mathcal{A}$. Let $T_i(1) = \emptyset, i \in A$, $m = 1$ (phase index), $t = 1$ (round index)
while $t \leq T$ do
    Step 1: Play arms.
    for $j \in \mathcal{A}_m$ do
        Let $T_j(m) = T_j(m-1)$
        while $|T_j(m)| \leq n_m$ and $t \leq T$ do
            Play arm $j$, receive $X_t$. Add $t$ to $T_j(m)$. Increment $t$ by 1.
        end while
    end for
    Step 2: Eliminate sub-optimal arms.
    For every arm in $j \in \mathcal{A}_m$, compute $\bar{X}_{m,j}$ as the average of observations at time steps $t \in T_j(m)$. That is,
        $\bar{X}_{m,j} = \frac{1}{|T_j(m)|} \sum_{t \in T_j(m)} X_t$ .
    Construct $\mathcal{A}_{m+1}$ by eliminating actions $j \in \mathcal{A}_m$ with
        $\bar{X}_{m,j} + \tilde{\Delta}_m < \max_{j' \in \mathcal{A}_m} \bar{X}_{m,j'}$ .
    Step 3: Decrease Tolerance.
    Set $\tilde{\Delta}_{m+1} = \frac{\tilde{\Delta}_m}{2}$.
    Step 4: Bridge period.
    Pick an arm $j \in \mathcal{A}_{m+1}$ and play it $\nu_m = n_m - n_{m-1}$ times while incrementing $t \leq T$. Discard all observations from this period. Do not add $t$ to $T_j(m)$.
    Increment phase index $m$.
end while
```

lenge is developing these confidence bounds from delayed, aggregated anonymous feedback. Handling this form of feedback involves a credit assignment problem of deciding which samples can be used for a given arm's mean estimation, since each sample is an aggregate of rewards from multiple previously played arms. This credit assignment problem would be hopeless in a passive learning setting without further information on how the samples were generated. Our algorithm utilizes the power of active learning to design the phases in such a way that the feedback can be effectively 'decensored' without losing too many samples.

A naive approach to defining the confidence bounds for delays bounded by a constant $d \geq 0$ would be to observe that,

$$\left| \sum_{t \in T_j(m) \setminus T_j(m-1)} X_t - \sum_{t \in T_j(m) \setminus T_j(m-1)} R_{t,j} \right| \leq d,$$

since all rewards are in $[0, 1]$. Then we could use Hoeffding's inequality to bound R_{t,J_t} (see Appendix F) and select

$$n_m = \frac{C_1 \log(T\tilde{\Delta}_m^2)}{\tilde{\Delta}_m^2} + \frac{C_2 md}{\tilde{\Delta}_m}$$

for some constants C_1, C_2. This corresponds to worst case regret of $O(\sqrt{KT \log K} + K \log(T)d)$. For $d \gg \mathbb{E}[\tau]$ and large T, this is significantly worse than that of Joulani et al. (2013). In Section 4, we show that, surprisingly, it is possible to recover the same rate of regret as Joulani et al. (2013), but this requires a significantly more nuanced argument to get tighter confidence bounds and smaller n_m. In the next section, we describe this improved choice of n_m for every phase $m \in \mathbb{N}$ and its implications on the regret, for each of the three cases mentioned previously: (i) Known and bounded expected delay (Assumption 1), (ii) Bounded delay with known bound and expected value (Assumptions 1 and 2), (iii) Delay with known and bounded variance and expectation (Assumptions 1 and 3).

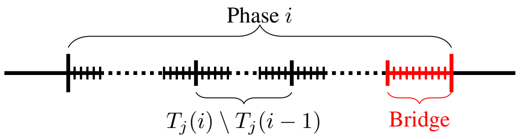


Figure 2: An example of phase i of our algorithm.

4. Regret Analysis

In this section, we specify the choice of parameters n_m and provide regret guarantees for Algorithm 1 for each of the three previously mentioned cases.

4.1. Known and Bounded Expected Delay

First, we consider the setting with the weakest assumption on delay distribution: we only assume that the expected delay, $\mathbb{E}[\tau]$, is bounded and known. No assumption on the support or variance of the delay distribution is made. The regret analysis for this setting will not use the bridge period, so Step 4 of the algorithm could be omitted in this case.

Choice of $\mathbf{n_m}$ Here, we use Algorithm 1 with

$$n_m = \frac{C_1 \log(T\tilde{\Delta}_m^2)}{\tilde{\Delta}_m^2} + \frac{C_2 m\mathbb{E}[\tau]}{\tilde{\Delta}_m} \tag{2}$$

for some large enough constants C_1, C_2. The exact value of n_m is given in Equation (14) in Appendix B.

Estimation of error bounds We bound the error between $\bar{X}_{m,j}$ and μ_j by $\tilde{\Delta}_m/2$. In order to do this we first bound the corruption of the observations received during timesteps $T_j(m)$ due to delays.

Fix a phase m and arm $j \in \mathcal{A}_m$. Then the observations X_t in the period $t \in T_j(m) \setminus T_j(m-1)$ are composed of two types of rewards: a subset of rewards from plays of arm j in this period, and delayed rewards from some of the plays before this period. The expected value of observations from this period would be $(n_m - n_{m-1})\mu_j$ but for the rewards entering and leaving this period due to delay. Since the reward is bounded by 1, a simple observation is that expected discrepancy between the sum of observations in this period and the quantity $(n_m - n_{m-1})\mu_j$ is bounded by the expected delay $\mathbb{E}[\tau]$,

$$\mathbb{E}\left[\sum_{t \in T_j(m)\setminus T_j(m-1)} (X_t - \mu_j)\right] \leq \mathbb{E}[\tau]. \qquad (3)$$

Summing this over phases $\ell = 1, \dots m$ gives a bound

$$|\mathbb{E}[\bar{X}_{m,j}] - \mu_j| \leq \frac{m\mathbb{E}[\tau]}{|T_j(m)|} = \frac{m\mathbb{E}[\tau]}{n_m}. \qquad (4)$$

Note that given the choice of n_m in (2), the above is smaller than $\tilde{\Delta}_m/2$, when large enough constants are used. Using this, along with concentration inequalities and the choice of n_m from (2), we can obtain the following high probability bound. A detailed proof is provided in Appendix B.1.

Lemma 1 *Under Assumption 1 and the choice of n_m given by (2), the estimates $\bar{X}_{m,j}$ constructed by Algorithm 1 satisfy the following: For every fixed arm j and phase m, with probability $1 - \frac{3}{T\tilde{\Delta}_m^2}$, either $j \notin \mathcal{A}_m$, or:*

$$\bar{X}_{m,j} - \mu_j \leq \tilde{\Delta}_m/2\,.$$

Regret bounds Using Lemma 1, we derive the following regret bounds in the current setting.

Theorem 2 *Under Assumption 1, the expected regret of Algorithm 1 is upper bounded as*

$$\mathbb{E}[\mathfrak{R}_T] \leq \sum_{\substack{j=1\\ j\neq j^*}}^{K} O\left(\frac{\log(T\Delta_j^2)}{\Delta_j} + \log(1/\Delta_j)\mathbb{E}[\tau]\right). \qquad (5)$$

Proof: Given Lemma 1, the proof of Theorem 2 closely follows the analysis of the Improved UCB algorithm of Auer & Ortner (2010). Lemma 1 and the elimination condition in Algorithm 1 ensure that, with high probability, any suboptimal arm j will be eliminated by phase $m_j = \log(1/\Delta_j)$, thus incurring regret at most $n_{m_j}\Delta_j$ We then substitute in n_{m_j} from (2), and sum over all suboptimal arms. A detailed proof is in Appendix B.2. As in Auer & Ortner (2010), we avoid a union bound over all arms (which would result in an extra $\log K$) by *(i)* reasoning about the regret of each arm individually, and *(ii)* bounding the regret resulting from erroneously eliminating the optimal arm by carefully controlling the probability it is eliminated in each phase. □

Considering the worst-case values of Δ_j (roughly $\sqrt{K/T}$), we obtain the following problem independent bound.

Corollary 3 *For any problem instance satisfying Assumption 1, the expected regret of Algorithm 1 satisfies*

$$\mathbb{E}[\mathfrak{R}_T] \leq O(\sqrt{KT\log(K)} + K\mathbb{E}[\tau]\log(T)).$$

4.2. Delay with Bounded Support

If the delay is bounded by some constant $d \geq 0$ and a single arm is played repeatedly for long enough, we can restrict the number of arms corrupting the observation X_t at a given time t. In fact, if each arm j is played consecutively for more than d rounds, then at any time $t \in T_j(m)$, the observation X_t will be composed of the rewards from at most two arms: the current arm j, and previous arm j'. Further, from the elimination condition, with high probability, arm j' will have been eliminated if it is clearly suboptimal. We can then recursively use the confidence bounds for arms j and j' from the previous phase to bound $|\mu_j - \mu_{j'}|$. Below, we formalize this intuition to obtain a tighter bound on $|\bar{X}_{m,j} - \mu_j|$ for every arm j and phase m, when each active arm is played a specified number of times per phase.

Choice of $\mathbf{n_m}$ Here, we define,

$$\begin{aligned} n_m =& \frac{C_1\log(T\tilde{\Delta}_m^2)}{\tilde{\Delta}_m^2} + \frac{C_2\mathbb{E}[\tau]}{\tilde{\Delta}_m} \qquad (6)\\ &+ \min\left\{md, \frac{C_3\log(T\tilde{\Delta}_m^2)}{\tilde{\Delta}_m^2} + \frac{C_4 m\mathbb{E}[\tau]}{\tilde{\Delta}_m}\right\} \end{aligned}$$

for some large enough constants C_1, C_2, C_3, C_4 (see Appendix C, Equation (18) for the exact values). This choice of n_m means that for large d, we essentially revert back to the choice of n_m from (2) for the unbounded case, and we gain nothing by using the bound on the delay. However, if d is not large, the choice of n_m in (6) is smaller than (2) since the second term now scales with $\mathbb{E}[\tau]$ rather than $m\mathbb{E}[\tau]$.

Estimation of error bounds In this setting, by the elimination condition and bounded delays, the expectation of each reward entering $T_j(m)$ will be within $\tilde{\Delta}_{m-1}$ of μ_j, with high probability. Then, using knowledge of the upper bound of the support of τ, we can obtain a tighter bound and get an error bound similar to Lemma 1 with the smaller value of n_m in (6). We prove the following proposition. Since $\tilde{\Delta}_m = 2^{-m}$, this is considerably tighter than (3).

Proposition 4 *Assume $n_i - n_{i-1} \geq d$ for phases $i = 1, \dots, m$. Define $\mathcal{E}_{m-1}$ as the event that all arms $j \in \mathcal{A}_m$ satisfy error bounds $|\bar{X}_{m-1,j} - \mu_j| \leq \tilde{\Delta}_{m-1}/2$. Then, for* *every arm $j \in \mathcal{A}_m$,*

$$\mathbb{E}\left[\sum_{t \in T_j(m) \setminus T_j(m-1)} (X_t - \mu_j) \middle| \mathcal{E}_{m-1}\right] \leq \tilde{\Delta}_{m-1}\mathbb{E}[\tau].$$

Proof: (Sketch). Consider a fixed arm $j \in \mathcal{A}_m$. The expected value of the sum of observations X_t for $t \in T_j(m) \setminus T_j(m-1)$ would be $(n_m - n_{m-1})\mu_j$ were it not for some rewards entering and leaving this period due to the delays. Because of the i.i.d. assumption on the delay, in expectation, the number of rewards leaving the period is roughly the same as the number of rewards entering this period, i.e., $\mathbb{E}[\tau]$. (Conditioning on $\mathcal{E}_{m-1}$ does not effect this due to the bridge period). Since $n_m - n_{m-1} \geq d$, the reward coming into the period $T_j(m) \setminus T_j(m-1)$ can only be from the previous arm j'. All rewards leaving the period are from arm j. Therefore the expected difference between rewards entering and leaving the period is $(\mu_j - \mu_{j'})\mathbb{E}[\tau]$. Then, if μ_j is close to $\mu_{j'}$, the total reward leaving the period is compensated by total reward entering. Due to the bridge period, even when j is the first arm played in phase m, $j' \in \mathcal{A}_m$, so it was not eliminated in phase $m-1$. By the elimination condition in Algorithm 1, if the error bounds $|\bar{X}_{m-1,j} - \mu_j| \leq \tilde{\Delta}_{m-1}/2$ are satisfied for all arms in $\mathcal{A}_m$, then $|\mu_j - \mu_{j'}| \leq \tilde{\Delta}_{m-1}$. This gives the result. □

Repeatedly using Proposition 4 we get,

$$\sum_{i=1}^{m} \mathbb{E}\left[\sum_{t \in T_j(i) \setminus T_j(i-1)} (X_t - \mu_j) \middle| \mathcal{E}_{i-1}\right] \leq 2\mathbb{E}[\tau]$$

since $\sum_{i=1}^{m} \tilde{\Delta}_{i-1} = \sum_{i=0}^{m-1} 2^{-i} \leq 2$. Then, observe that $\mathbb{P}(\mathcal{E}_i^C)$ is small. This bound is an improvement of a factor of m compared to (4). For the regret analysis, we derive a high probability version of the above result. Using this, and the choice of $n_m \geq \Omega\left(\frac{\log(T\tilde{\Delta}_m^2)}{\tilde{\Delta}_m^2} + \frac{\mathbb{E}[\tau]}{\tilde{\Delta}_m}\right)$ from (6), for large enough constants, we derive the following lemma. A detailed proof is given in Appendix C.1.

Lemma 5 *Under Assumptions 1 of known expected delay and 2 of bounded delays, and choice of n_m given in (6), the estimates $\bar{X}_{m,j}$ obtained by Algorithm 1 satisfy the following: For any arm j and phase m, with probability at least $1 - \frac{12}{T\tilde{\Delta}_m^2}$, either $j \notin \mathcal{A}_m$ or*

$$\bar{X}_{m,j} - \mu_j \leq \tilde{\Delta}_m/2.$$

Regret bounds We now give regret bounds for this case.

Theorem 6 *Under Assumption 1 and bounded delay Assumption 2, the expected regret of Algorithm 1 satisfies*

$$\mathbb{E}[\mathfrak{R}_T] \leq \sum_{j=1; j \neq j^*}^{K} O\bigg(\frac{\log(T\Delta_j^2)}{\Delta_j} + \mathbb{E}[\tau] + \min\left\{d, \frac{\log(T\Delta_j^2)}{\Delta_j} + \log(\frac{1}{\Delta_j})\mathbb{E}[\tau]\right\}\bigg).$$

Proof: (Sketch). Given Lemma 5, the proof is similar to that of Theorem 2. The full proof is in Appendix C.2. □

Then, if $d \leq \sqrt{\frac{T \log K}{K}} + \mathbb{E}[\tau]$, we get the following problem independent regret bound which matches that of Joulani et al. (2013).

Corollary 7 *For any problem instance satisfying Assumptions 1 and 2 with $d \leq \sqrt{\frac{T \log K}{K}} + \mathbb{E}[\tau]$, the expected regret of Algorithm 1 satisfies*

$$\mathbb{E}[\mathfrak{R}_T] \leq O(\sqrt{KT\log(K)} + K\mathbb{E}[\tau]).$$

4.3. Delay with Bounded Variance

If the delay is unbounded but well behaved in the sense that we know (a bound on) the variance, then we can obtain similar regret bounds to the bounded delay case. Intuitively, delays from the previous phase will only corrupt observations in the current phase if their delays exceed the length of the bridge period. We control this by using the bound on the variance to bound the tails of the delay distributions.

Choice of $\mathbf{n_m}$ Let $\mathbb{V}(\tau)$ be the known variance (or bound on the variance) of the delay, as in Assumption 3. Then, we use Algorithm 1 with the following value of n_m,

$$n_m = C_1 \frac{\log(T\tilde{\Delta}_m^2)}{\tilde{\Delta}_m^2} + C_2 \frac{\mathbb{E}[\tau] + \mathbb{V}(\tau)}{\tilde{\Delta}_m} \tag{7}$$

for some large enough constants C_1, C_2. The exact value of n_m is given in Appendix D, Equation (25).

Regret bounds We get the following instance specific and problem independent regret bound in this case.

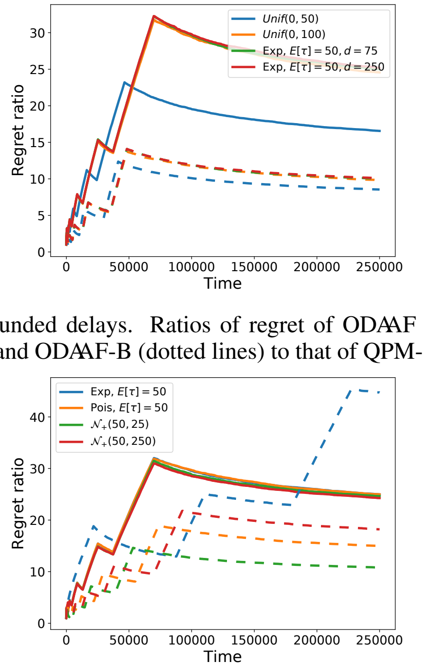
Theorem 8 *Under Assumption 1 and Assumption 3 of known (bound on) the expectation and variance of the delay, and choice of n_m from (7), the expected regret of Algorithm 1 can be upper bounded by,*

$$\mathbb{E}[\mathfrak{R}_T] \leq \sum_{j=1:\mu_j \neq \mu^*}^{K} O\bigg(\frac{\log(T\Delta_j^2)}{\Delta_j} + \mathbb{E}[\tau] + \mathbb{V}(\tau)\bigg).$$

Proof: (Sketch). See Appendix D.2. We use Chebychev's inequality to get a result similar to Lemma 5 and then use a similar argument to the bounded delay case. □

Corollary 9 *For any problem instance satisfying Assumptions 1 and 3, the expected regret of Algorithm 1 satisfies*

$$\mathbb{E}[\mathfrak{R}_T] \leq O(\sqrt{KT\log(K)} + K\mathbb{E}[\tau] + K\mathbb{V}(\tau)).$$



(a) Bounded delays. Ratios of regret of ODAAF (solid lines) and ODAAF-B (dotted lines) to that of QPM-D.

(b) Unbounded delays. Ratios of regret of ODAAF (solid lines) and ODAAF-V (dotted lines) to that of QPM-D.

Figure 3: The ratios of regret of variants of our algorithm to that of QPM-D for different delay distributions.

Remark If $\mathbb{E}[\tau] \geq 1$, then the delay penalty can be reduced to $O(K\mathbb{E}[\tau] + K\mathbb{V}(\tau)/\mathbb{E}[\tau])$ (see Appendix D).

Thus, it is sufficient to know a bound on variance to obtain regret bounds similar to those in bounded delay case. Note that this approach is not possible just using knowledge of the expected delay since we cannot guarantee that the reward entering phase i is from an arm active in phase $i-1$.

5. Experimental Results

We compared the performance of our algorithm (under different assumptions) to QPM-D (Joulani et al., 2013) in various experimental settings. In these experiments, our aim was to investigate the effect of the delay on the performance of the algorithms. In order to focus on this, we used a simple setup of two arms with Bernoulli rewards and $\boldsymbol{\mu} = (0.5, 0.6)$. In every experiment, we ran each algorithm to horizon $T = 250000$ and used UCB1 (Auer et al., 2002) as the base algorithm in QPM-D. The regret was averaged over 200 replications. For ease of reading, we define ODAAF to be our algorithm using only knowledge of the expected delay, with n_m defined as in (2) and run without a bridge period, and ODAAF-B and ODAAF-V to be the versions of Algorithm 1 that use a bridge period and information on the bounded support and the finite variance of the delay to define n_m as in (6) and (7) respectively.

We tested the algorithms with different delay distributions. In the first case, we considered bounded delay distributions whereas in the second case, the delays were unbounded. In Fig. 3a, we plotted the ratios of the regret of ODAAF and ODAAF-B (with knowledge of d, the delay bound) to the regret of QPM-D. We see that in all cases the ratios converge to a constant. This shows that the regret of our algorithm is essentially of the same order as that of QPM-D. Our algorithm predetermines the number of times to play each active arm per phase (the randomness appears in whether an arm is active), so the jumps in the regret are it changing arm. This occurs at the same points in all replications.

Fig. 3b shows a similar story for unbounded delays with mean $\mathbb{E}[\tau] = 50$ (where $\mathcal{N}_+$ denotes the the half normal distribution). The ratios of the regret of ODAAF and ODAAF-V (with knowledge of the delay variance) to the regret of QPM-D again converge to constants. Note that in this case, these constants, and the location of the jumps, vary with the delay distribution and $\mathbb{V}(\tau)$. When the variance of the delay is small, it can be seen that using the variance information leads to improved performance. However, for exponential delays where $\mathbb{V}(\tau) = \mathbb{E}[\tau]^2$, the large variance causes n_m to be large and so the suboptimal arm is played more, increasing the regret. In this case ODAAF-V had only just eliminated the suboptimal arm at time T.

It can also be illustrated experimentally that the regret of our algorithms and that of QPM-D all increase linearly in $\mathbb{E}[\tau]$. This is shown in Appendix E. We also provide an experimental comparison to Vernade et al. (2017) in Appendix E.

6. Conclusion

We have studied an extension of the multi-armed bandit problem to bandits with delayed, aggregated anonymous feedback. Here, a sum of observations is received after some stochastic delay and we do not learn which arms contributed to each observation. In this more difficult setting, we have proven that, surprisingly, it is possible to develop an algorithm that performs comparably to those for the simpler delayed feedback bandits problem, where the assignment of rewards to plays is known. Particularly, using only knowledge of the expected delay, our algorithm matches the worst case regret of Joulani et al. (2013) up to a logarithmic factor. This logarithmic factors can be removed using an improved analysis and slightly more information about the delay; if the delay is bounded, we achieve the same worst case regret as Joulani et al. (2013), and for unbounded delays with known finite variance, we have an extra additive $\mathbb{V}(\tau)$ term. We supported these claims experimentally. Note that while our algorithm matches the order of regret of QPM-D, the constants are worse. Hence, it is an open problem to find algorithms with better constants.

Acknowledgments

CPB would like to thank the EPSRC funded EP/L015692/1 STOR-i centre for doctoral training and Sparx. We would like to thank the reviewers for their helpful comments.

References

Agrawal, R., Hedge, M., and Teneketzis, D. Asymptotically efficient adaptive allocation rules for the multi-armed bandit problem with switching cost. *IEEE Transactions on Automatic Control*, 33(10):899–906, 1988.

Auer, P. and Ortner, R. UCB revisited: Improved regret bounds for the stochastic multi-armed bandit problem. *Periodica Mathematica Hungarica*, 61(1-2):55–65, 2010.

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

Bubeck, S. and Cesa-Bianchi, N. *Regret Analysis of Stochastic and Nonstochastic Multi-armed Bandit Problems*. Foundations and Trends in Machine Learning. 2012.

Cappé, O., Garivier, A., Maillard, O.-A., Munos, R., and Stoltz, G. Kullback–Leibler upper confidence bounds for optimal sequential allocation. *The Annals of Statistics*, 41(3):1516–1541, 2013.

Cesa-Bianchi, N., Gentile, C., Mansour, Y., and Minora, A. Delay and cooperation in nonstochastic bandits. In *Conference on Learning Theory*, pp. 605–622, 2016.

Chapelle, O. and Li, L. An empirical evaluation of Thompson sampling. In *Advances in Neural Information Processing Systems*, pp. 2249–2257, 2011.

Desautels, T., Krause, A., and Burdick, J. Parallelizing exploration-exploitation tradeoffs in Gaussian process bandit optimization. *Journal of Machine Learning Research*, 15(1):3873–3923, 2014.

Doob, J. L. *Stochastic processes*. John Wiley & Sons, 1953.

Dudik, M., Hsu, D., Kale, S., Karampatziakis, N., Langford, J., Reyzin, L., and Zhang, T. Efficient optimal learning for contextual bandits. In *Conference on Uncertainty in Artificial Intelligence*, pp. 169–178, 2011.

Freedman, D. A. On tail probabilities for martingales. *The Annals of Probability*, 3(1):100–118, 1975.

Joulani, P., György, A., and Szepesvári, C. Online learning under delayed feedback. In *International Conference on Machine Learning*, pp. 1453–1461, 2013.

Lai, T. and Robbins, H. Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics*, 6(1): 4–22, 1985.

Mandel, T., Liu, Y.-E., Brunskill, E., and Popovic, Z. The queue method: Handling delay, heuristics, prior data, and evaluation in bandits. In *AAAI*, pp. 2849–2856, 2015.

Neu, G., Antos, A., György, A., and Szepesvári, C. Online Markov decision processes under bandit feedback. In *Advances in Neural Information Processing Systems*, pp. 1804–1812, 2010.

Perchet, V., Rigollet, P., Chassang, S., and Snowberg, E. Batched bandit problems. *The Annals of Statistics*, 44 (2):660–681, 2016.

Szita, I. and Szepesvári, C. Agnostic KWIK learning and efficient approximate reinforcement learning. In *Conference on Learning Theory*, pp. 739–772, July 2011.

Thompson, W. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3–4):285–294, 1933.

Vernade, C., Cappé, O., and Perchet, V. Stochastic bandit models for delayed conversions. In *Conference on Uncertainty in Artificial Intelligence*, 2017.