Nonstochastic Multiarmed Bandits with Unrestricted Delays

Tobias Sommer Thune,[1] **Nicolò Cesa-Bianchi** [2] **& Yevgeny Seldin** [1]

[1]University of Copenhagen, [2]Università degli Studi di Milano

Abstract

We investigate multiarmed bandits with delayed feedback, where the delays need neither be identical nor bounded. We prove three results:

1. The "delayed" version of the standard algorithm `Exp3` achieves the $\mathcal{O}\big(\sqrt{(KT+D)\ln K}\big)$ regret bound conjectured by Cesa-Bianchi et al. [2016], for variable, but bounded delays. Here K is the number of actions and D is the total delay over T rounds.

2. We introduce a new algorithm that skips feedback with excessively large delays. This algorithm maintains the same regret bound but also for unrestricted delays. Tuning requires prior knowledge of T and D.

3. For our new algorithm we then construct a novel doubling scheme that takes care of the tuning under the assumption that the delays are available at action time (rather than at loss observation time). The resulting oracle regret bound is of order $\min_\beta \big(|S_\beta|+\beta\ln K+(KT+D_\beta)/\beta\big)$, where $|S_\beta|$ is the number of observations with delay exceeding β, and D_β is the total delay of observations with delay below β. This relaxes to the conjectured bound but can be polynomially better of which we provide an example.

Setting

Multiarmed Bandits models sequential decision processes as a repeated game, where a learner in each round chooses an action A_t out of K possible actions. The associated loss $\ell_t^{A_t} \in [0,1]$ is suffered by the learner. Normally $\ell_t^{A_t}$ is revealed as feedback to the learner immediately, but we consider a *delayed* variant, where it arrives d_t timesteps later at the end of round $t+d_t$.

The performance of the learner is measured by the *expected regret* of the actions of the learner over T rounds, compared to the best action in hindsight:

$$\bar{\mathcal{R}}_T := \mathbb{E}\left[\sum_{t=1}^T \ell_t^{A_t}\right] - \min_{a\in[K]} \sum_{t=1}^T \ell_t^a. \tag{1}$$

Since only the loss of the chosen action A_t is revealed to the learner, this setting displays a tradeoff between *exploration and exploitation*.

Part 1: Delayed `Exp3`

Our first result concerns the standard `Exp3` algorithm utilising *exponential weights* with *importance weighted loss estimators*. Our variant updates as the delayed feedback becomes available and truncates the learning rate η, such that the probabilities are stable over the *stability-spans* $N_t = |\{s : s+d_s \in [t, t+d_t)\}|$.

Algorithm 1: Delayed exponential weights (DEW)

Input: Learning rate η; Upper bound on stability-spans $N \geq N_t \ \ \forall t$.

Truncate the learning rate: $\eta' = \min\{\eta, (2eN)^{-1}\}$;

Initialize $w_0^a = 1$ for all $a \in [K]$;

for $t = 1, 2, \ldots$ **do**

Let $p_t^a = \frac{w_{t-1}^a}{\sum_b w_{t-1}^b}$ for $a \in [K]$;

Draw an action $A_t \in [K]$ according to the distribution $\boldsymbol{p_t}$ and play it;

Observe feedback $(s, \ell_s^{A_s})$ for all $\{s : s+d_s = t\}$ and construct estimators $\hat{\ell}_s^a = \frac{\ell_s^a \mathbb{1}(a=A_s)}{p_s^a}$;

Update $w_t^a = w_{t-1}^a \exp\left(-\eta' \sum_{s:s+d_s=t} \hat{\ell}_s^a\right)$;

end

Regret Bounds for Delayed `Exp3`

For Algorithm 1 we prove the following result, answering an open problem of Cesa-Bianchi et al. [2016]:

Theorem 1 *Under the assumption that an upper bound $N \geq \max_t N_t$ on the stability-spans is known, the regret of Algorithm 1 with a learning rate η against an oblivious adversary satisfies*

$$\bar{\mathcal{R}}_T \leq \max\left\{\frac{\ln K}{\eta}, 2eN\ln K\right\} + \eta\left(\frac{KTe}{2} + D\right),$$

where $D = \sum_{t=1}^T d_t$. In particular, if T and D are known and $\eta = \sqrt{\frac{\ln K}{\frac{KTe}{2}+D}} \leq \frac{1}{2eN}$, we have

$$\bar{\mathcal{R}}_T \leq \mathcal{O}\left(\sqrt{(KT+D)\ln K}\right).$$

This result generalises the case of the fixed delay considered in Cesa-Bianchi et al. [2016]. In that paper a lower bound is further considered, which also applies as a worst case scaling for our algorithm:

Theorem 2 *[Cesa-Bianchi et al., 2016] For a fixed delay $d_t = d$ such that $D = dT$ the worst case regret is of order*

$$\Omega\left(\sqrt{KT + D\ln K}\right).$$

Part 2: Skipping Large Delays

Theorem 1 only obtains the desired regret scaling when the stability-spans are bounded compared to the learning rate. We remedy this by introducing a wrapper algorithm that disregards feedback from rounds with too large of a delay, tuned by a threshold β:

Algorithm 2: Skipper

Input: Threshold β; Algorithm $\mathcal{A}$.

for $t = 1, 2, \ldots$ **do**

Get prediction A_t from $\mathcal{A}$ and play it;

Observe feedback $(s, \ell_s^{A_s})$ for all $\{s : s+d_s = t\}$, and feed it to $\mathcal{A}$ for each s with $d_s < \beta$;

end

The point of wrapping `DEW` with `Skipper` is that we only have a unit cost of skipping a round, since the loss in each skipped round is bounded by 1.

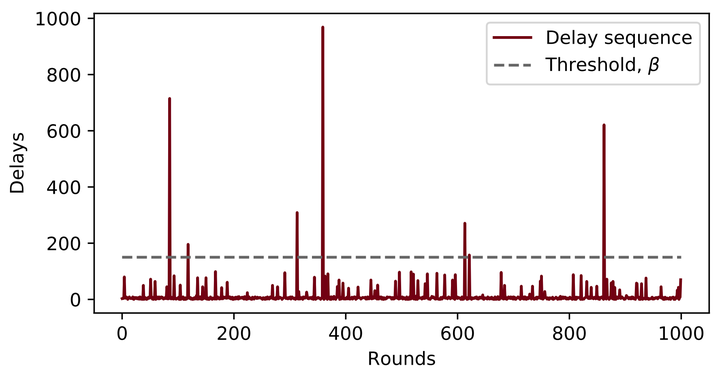

Fig. 1: Illustration of `Skipper` and a delay threshold $\beta = 150$. Skipping incurs a cost $|S_\beta| = 7$ while the maximal stability-span is lowered by an order of magnitude.

Regret Bounds for `Skipper`

The unit cost of skipping a round might be much smaller than the penalty of requiring stability over a large period, giving us the combined regret bound which hold for any stability-spans:

Theorem 3 *The expected regret of* `Skipper`*(β,* `DEW`*(η,2β)) against an oblivious adversary satisfies*

$$\bar{\mathcal{R}}_T \leq |S_\beta| + \max\left\{\frac{\ln K}{\eta}, 4e\beta\ln K\right\} + \eta\left(\frac{KTe}{2} + D_\beta\right),$$

where $D_\beta = \sum_{t\notin S_\beta} d_t$ is the cumulative delay experienced by `DEW`*.*

Corollary 4 *Assume that T and D are known and tune*

$$\eta = \frac{1}{4e\beta}, \quad \beta = \sqrt{\frac{\frac{eKT/2+D}{4e} + D}{4e\ln K}}.$$

Then the expected regret of `Skipper`*(β,* `DEW`*(η,2β)) against an oblivious adversary satisfies*

$$\bar{\mathcal{R}}_T \leq \mathcal{O}\left(\sqrt{(KT+D)\ln K}\right).$$

Part 3: Doubling Trick

In Part 2 we got rid of the need for the bounded stability-spans, meaning we achieve the conjectured regret bound for any sequence of delays. These results still need prior knowledge of T and D in order to tune β and η. We now loosen this requirement to knowing d_t at time t by applying a variation of the *doubling trick*:

We split the game into *epochs* indexed by m and restart the algorithm in each epoch, allowing us to treat them as individual games, where we control both the tuning and the length of the game. In epoch m we set $\eta_m \sim 1/\beta_m$ and

$$\beta_m = \frac{\sqrt{2^m}}{4e\ln K}$$

and we stay in epoch m as long as the following condition holds:

$$\max\left\{|S_{\beta_m}^m|^2, \left(\frac{eK\sigma(m)}{2} + D_{\beta_m}^m\right)\ln K\right\} \leq 2^m,$$

where $\sigma(m)$ is the length of epoch m. We show that not just does this give us the conjectured scaling $\mathcal{O}(\sqrt{(TK+D)\ln K})$ without knowledge of T and D, but we show that the doubling scheme is comparable to the *optimal choice* of β over the entire game:

Theorem 5 *The expected regret of* `Skipper`*(β,* `DEW`*(η,2β)) tuned by* `Doubling` *satisfies for any T*

$$\bar{\mathcal{R}}_T \leq \mathcal{O}\left(\min_\beta\left\{|S_\beta| + \beta\ln K + \frac{KT+D_\beta}{\beta}\right\} + K\ln K\right)$$

Corollary 6 *The expected regret of* `Skipper`*(β,* `DEW`*(η,2β)) tuned by* `Doubling` *can be relaxed for any T to*

$$\bar{\mathcal{R}}_T \leq \mathcal{O}\left(\sqrt{(KT+D)\ln K} + K\ln K\right).$$

Note that the oracle bound (Theorem 5) is always as strong as the explicit bound (Corollary 6). There are, however, cases where it is much tighter. Consider the following example:

Example 7 *For $t < \sqrt{KT/\ln K}$ let $d_t = T - t$ and for $t \geq \sqrt{KT/\ln K}$ let $d_t = 0$. Take $\beta = \sqrt{KT/\ln K}$. Then $D = \Theta(T\sqrt{KT/\ln K})$, but $D_\beta = 0$ (assuming that $T \geq K\ln K$) and $|S_\beta| < \sqrt{KT/\ln K}$. The corresponding regret bounds are*

$$\text{explicit} \quad : \quad \mathcal{O}\left(\sqrt{KT\ln K + T\sqrt{KT}}\right) = \mathcal{O}\big(T^{3/4}\big),$$

$$\text{oracle} \quad : \quad \mathcal{O}\left(\sqrt{KT\ln K}\right) = \mathcal{O}\big(T^{1/2}\big).$$

References

Nicolò Cesa-Bianchi, Claudio Gentile, and Yishay Mansour. Delay and cooperation in nonstochastic bandits. *Journal of Machine Learning Research*, 49:605–622, 2016.