Multi-agent online optimization with delays: Asynchronicity, adaptivity, and optimism

Yu-Guan Hsieh, Franck Iutzeler, Jérôme Malick, Panayotis Mertikopoulos

Multi-Agent Online Optimization with Delays: Asynchronicity, Adaptivity, and Optimism

Yu-Guan Hsieh YU-GUAN.HSIEH@UNIV-GRENOBLE-ALPES.FR
Franck Iutzeler FRANCK.IUTZELER@UNIV-GRENOBLE-ALPES.FR
Univ. Grenoble Alpes, LJK, Grenoble, 38000, France

Jérôme Malick JEROME.MALICK@UNIV-GRENOBLE-ALPES.FR
Univ. Grenoble Alpes, CNRS, Grenoble INP, LJK, 38000 Grenoble, France

Panayotis Mertikopoulos PANAYOTIS.MERTIKOPOULOS@IMAG.FR
Univ. Grenoble Alpes, CNRS, Inria, Grenoble INP, LIG, 38000, Grenoble, France & Criteo AI Lab

Abstract

In this paper, we provide a general framework for studying multi-agent online learning problems in the presence of delays and asynchronicities. Specifically, we propose and analyze a class of adaptive dual averaging schemes in which agents only need to accumulate gradient feedback received from the whole system, without requiring any between-agent coordination. In the single-agent case, the adaptivity of the proposed method allows us to extend a range of existing results to problems with potentially unbounded delays between playing an action and receiving the corresponding feedback. In the multi-agent case, the situation is significantly more complicated because agents may not have access to a global clock to use as a reference point; to overcome this, we focus on the information that is available for producing each prediction rather than the actual delay associated with each feedback. This allows us to derive adaptive learning strategies with optimal regret bounds, even in a fully *decentralized*, *asynchronous* environment. Finally, we also analyze an "optimistic" variant of the proposed algorithm which is capable of exploiting the predictability of problems with a slower variation and leads to improved regret bounds.

Keywords: Online learning; multi-agent systems; delayed feedback; asynchronous methods; adaptive algorithms

1. Introduction

Online learning is a powerful paradigm for sequential decision-making, with a range of diverse applications in portfolio selection, online auctions, recommender systems, and many other fields; for a comprehensive introduction to the topic, see the textbooks by Shalev-Shwartz (2011), Bubeck and Cesa-Bianchi (2012), Hazan (2016), and references therein. In the most basic online learning scenario, the agent (or "learner") chooses an action, the cost of this action is subsequently revealed to the agent (possibly along with some gradient-based feedback), and the process repeats.

In this bare-bones model, the time-varying nature of the problem is reflected in the variability of the cost functions encountered by the agent, and the feedback received by the agent is assumed to be immediately available at the end of each time step. However, in many cases of practical interest, there could be a significant delay between playing an action and receiving the corresponding feedback; for instance, this is typically the case in online ad auctions (Croissant et al., 2020), network traffic routing (Altman et al., 2006), etc.

Our work concerns online learning setups where delays and asynchronicities play a major role; these may be due to the computational overhead involved, the communication latency between

different learners in distributed multi-agent systems, the prediction of long-term effects, or any other reason. In the literature, the specifics of the delay model are often tailored to the targeted application: for instance, in online ad placement problems, delays are caused by the lag between the impression of an ad and its conversion, which data suggests are often exponentially distributed (Chapelle, 2014). Instead of zooming in on a particular application, our paper aims at studying the impact of delays and stimulus-response asynchronicities from a generalist, application-agnostic standpoint. To that end, we propose a flexible framework for distributed online optimization problems in which several agents collaborate asynchronously to enhance their individual/collective performance in an evolving environment with non-zero response times. This allows us to provide a wide range of regret bounds extending existing results in the literature, and to design novel adaptive methods that can be implemented in a fully distributed and decentralized manner.

Our contributions in the context of related work. There are three major underlying themes in our analysis. As we discussed above, the first has to do with **delays:** either due to a computing overhead or an inherent lag between "action" and "reaction", agents may have to update their actions based on feedback that is potentially stale and obsolete. The second has to do with **multi-agent** systems: in a network setting, learners may have to take decisions with very different information at their disposal, and with no realistic means of coordinating their decision-making mechanisms. Expanding further on this point, the third has to do with **adaptivity:** we are interested in learning algorithms that can be run with minimal information prerequisites at the agent end, while still achieving optimal regret bounds.

To take all this into account, we introduce in Section 2 a novel, flexible framework that unifies several models of online learning in the presence of delays – including both single- and multi-agent setups. To achieve no regret in this context, we employ the dual averaging template of Nesterov (2009a) which we combine with adaptive learning rates inspired by the "inverse root sum" blueprint of Auer et al. (2002b), McMahan and Streeter (2010) and Duchi et al. (2011). We show that the resulting policies achieve optimal data- and delay-dependent guarantees even in a fully decentralized environment (Section 4). In the literature, the closest antecedents to our result are the works of McMahan and Streeter (2014) and Joulani et al. (2016, 2019), in which the authors also devised adaptive methods for delayed online learning problems. However, all these papers dealt with the single-agent (shared-memory) setup – and while Joulani et al. (2019) makes the weakest assumptions among these three papers, the derived bounds are only data-dependent, not delay-dependent.

On the technical side, the multi-agent nature of the problem gives rise to two additional challenges that are not present in the single-agent setup: *i*) the non-monotonicity of the total amount of information available to the decision-making agent; and *ii*) the lack of a global counter that indicates the number of updates performed in the entire network so far.[1] In face of these challenges, we introduce in Section 3 the notion of *dependency graph*, a directed acyclic graph (DAG) that encodes how the feedback is actually received and used in the network. Each topological sorting of this DAG represents a *faithful permutation* of time that is compatible with the underlying decision-making process. With help of the dependecy graph we also provide a novel characterization of the key quantities that are involved in the incurred regret. Taken together, these elements allow us to design and analyze an adaptive algorithm that achieves optimal data- and delay-dependent regret bounds

1. To the best of our knowledge, the only work providing a partial answer to these challenges is that of Joulani et al. (2019): this work takes into account the first challenge and can partly address the second challenge, through an approach that is different from ours. Nonetheless, as mentioned in the previous paragraph, they focused on a setup that is fundamentally different from ours, and the obtained results hence also differ considerably.

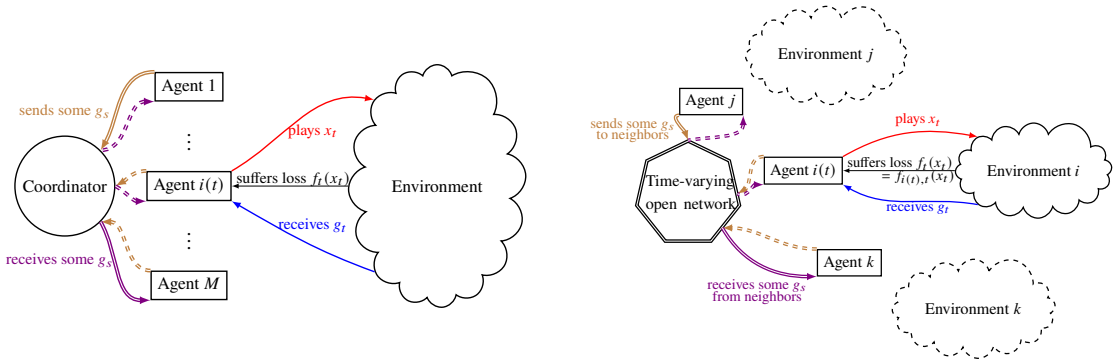


Figure 1: Illustration of the considered setup: a network of agents collaborate to minimize the total regret. We do not put any restriction on how the feedback is actually communicated. This can for example be done either through a coordinator-worker structure (left) or a decentralized open network (right).

in this completely decentralized setting. As a bonus of the new characterizations, we derive for the single-agent setup the first data- and delay-adaptive algorithm that does not require a "bounded delay" assumption.

Finally, in Section 5, we focus on improving these worst-case bounds by introducing a more "optimistic" step-size policy in the spirit of Rakhlin and Sridharan (2013).[2] This approach exploits the slow variation of "predictable" sequences, thereby improving the regret guarantees of online algorithms. However, when gradients arrive out of order, the predictability of a loss sequence may be compromised – and, indeed, in the presence of delays, we show that a crude implementation of optimistic methods cannot yield any obvious benefit. To account for this, we introduce a "separation of timescales" between the "sensing" and "updating" steps of the optimistic dual averaging method, and we show that this variable step-size scaling leads to optimal data-dependent guarantees.

2. A general framework for asynchronous online optimization

2.1 Problem setup

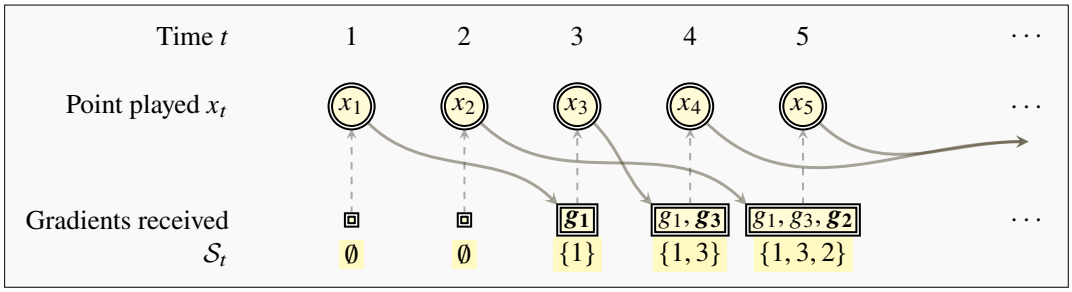
Consider a set of agents $\mathcal{M} = \{1, \dots, M\}$ playing against a sequence of time-varying loss functions, with the goal of minimizing their regret. Formally, at each time slot $t = 1, 2, \dots$, one of the agents becomes *active*, they select an action x_t from the constraint set $\mathcal{X}$, and they incur a loss $f_t(x_t)$.[3] The performance of the agents is then measured by the cumulative regret

$$\mathbf{Reg}_T(u) = \sum_{t=1}^{T} f_t(x_t) - \sum_{t=1}^{T} f_t(u) \tag{1}$$

where $u \in \mathcal{X}$ is an arbitrary comparator action. In the above, $\mathcal{X}$ is assumed to be a closed convex subset of $\mathbb{R}^d$, and each $f_t : \mathbb{R}^d \to \mathbb{R} \cup \{+\infty\}$ is convex and subdifferentiable on $\mathcal{X}$. Unless otherwise stated, we assume that the agents receive first-order feedback $g_t \in \partial f_t(x_t)$ at some moment after x_t

2. The concurrent work Flaspohler et al. (2021) that appears after the initial submission of our manuscript also exploits the same idea and provides empirical evidence of the benefit of optimism in delayed online learning.
3. For simplicity, we assume throughout that only one agent is active at each time step.

Single-agent ($M = 1$)



Multi-agent ($M = 2$)

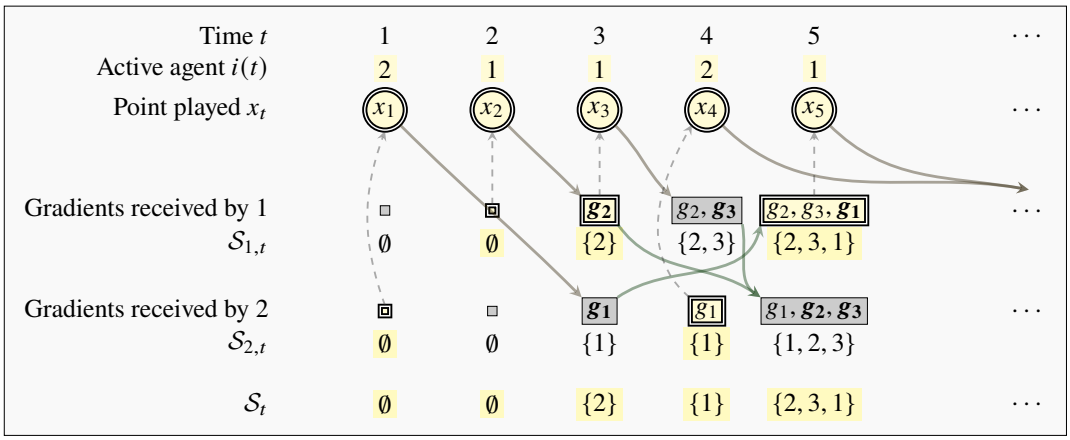


Figure 2: Illustration of the type of feedback sequences that may occur in a multi-agent setting. In the standard single-agent case, the feedback sequence $\mathcal{S}_t$, $t = 1, 2, \dots$, is necessarily non-decreasing: even though the feedback may not arrive with the same order as the corresponding actions, the *number* of available gradients can only grow. This no longer holds when multiple agents are involved in the optimization process.

is played (namely, g_t is a subgradient of f_t at x_t).[4] Irrespective of the nature of the problem, we will refer to x_t interchangeably as the *prediction* made by the active agent or the *action* played by the active agent at time t, and we will write $i(t)$ for the agent that is active at time t.

For visualization purposes, the above setup is illustrated in Fig. 1, where we also highlight the fact that we do not put any restriction are how information is exchanged between agents.

The delay model. In environments with *delayed feedback*, g_t is only received by all the agents $i \in \mathcal{M}$ a certain amount of time after the generating action x_t was played. In this regard, we will focus on the following sources of delay: *i*) *inherent delays* that arise when the effect of a decision requires some time to be observed; *ii*) *computation delays* that arise when processing the action takes time (e.g., due to gradient computations); and *iii*) *communication delays* that arise in network setups where multiple workers share first-order information among themselves.

To express this formally, we write $[t] := \{1, \dots, t\}$ and we write $\mathcal{S}_{i,t} \subseteq [t-1]$ for the set of gradient timestamps that are available to agent i at time t; in other words, at time t, the i-th agent

4. In a slight abuse of terminology, the terms gradient and subgradient will be used interchangeably in the sequel.

only has $\{g_s : s \in \mathcal{S}_{i,t}\}$ at their disposal. Clearly, at each stage $t = 1, 2, \dots$, the active agent $i(t)$ can only compute x_t based on $\{g_s : s \in \mathcal{S}_{i(t),t}\}$, i.e., the set of subgradients available for it at time t. This quantity is of utmost importance in our framework, so we also define

$$\mathcal{S}_t = \mathcal{S}_{i(t),t} \qquad \text{and} \qquad \mathcal{U}_t = [t-1] \setminus \mathcal{S}_t \tag{2}$$

for the set of timestamps that are available (resp. unavailable) to the active agent at time t.

In a slight abuse of terminology, we will refer to both $(\mathcal{S}_{i,t})_{t\in[T]}$ and $(\mathcal{S}_t)_{t\in[T]}$ as feedback sequences although, strictly speaking, they only contain the *timestamps* of the corresponding feedback. Clearly, the non-delayed setting corresponds to the case $\mathcal{S}_t = \mathcal{S}_{i,t} = [t-1]$ and $\mathcal{U}_t = \varnothing$.

2.2 Main challenges: Non-monotonicity of the feedback sequence and lack of synchronization

We now highlight two prominent features of our asynchronous online optimization framework that distinguish it from the large corpus of literature on *single-agent* online learning with delays. First, from the point of view of *any* single agent i, the feedback sequence $(\mathcal{S}_{i,t})_{t\in[T]}$ is non-decreasing by definition, i.e., $\mathcal{S}_{i,t} \subseteq \mathcal{S}_{i,t+1}$ for all $t = 1, 2, \dots$. However, this may not be the case for the *active* feedback sequence $(\mathcal{S}_t)_{t\in[T]}$ which is in general *non-monotone*. In fact, due to communication delays, the same element of feedback may not arrive at each node at the same time. Thus, as the active agent differs from one time slot to another, a timestamp contained in $\mathcal{S}_t$ may not belong to $\mathcal{S}_{t+1}$ (see Fig. 2 for an illustration). This leads to the first challenge we seek to overcome:

Challenge I. Design learning algorithms capable of handling non-monotone feedback sequences.

Remark. We stress here that this issue is inextricably tied to the multi-agent character of our model. In the single-agent case, $\mathcal{S}_t$ is *de facto* monotone, so this problem does not arise.

Second, in our model the agents only communicate when they exchange the received feedback. Without additional coordination, the network does not maintain any global information about the evolution of the learning process. In particular, for reasons of privacy and information security, we do not assume that agents have access to a global counter that indicates how many actions have been played at any given stage (as this could carry sensitive, identification-prone information). Similarly, other quantities of interest, such as the current cumulative unavailability D_t defined below, are also unavailable to each agent. This leads to our second challenge:

Challenge II. Dispense of the need to know t or other non-local information.

As shown above, the lack of network synchronization, along with the non-monotonicity of the active feedback sequence, poses crucial challenges to both the design of the algorithms and the accompanying analysis. In face of these, we introduce in Section 3.2 an appropriate reordering of time that enables us to go beyond the algorithms developed for the single-agent setting.

Quantifying the impact of delays. As illustrated in Fig. 2, having multiple agents also means that we can no longer associate a single delay to each individual feedback element. This explains our choice of focusing on the available subgradients instead of the actual delays, which largely simplifies the description of the framework. The delays, in turn, are still implicitly encoded in the sets $(\mathcal{S}_{i,t})$. To quantify their effect, it will be convenient to consider the following measures:

- The *maximum delay* τ is the longest wait to receive an element of feedback: $\tau = \min\{\tau : [t-\tau-1] \subseteq \mathcal{S}_t \text{ for all } t \in [T]\}$.

- The *maximum unavailability* ν of the feedback is defined as $\nu = \max_{t\in[T]} \operatorname{card}(\mathcal{U}_t)$. This is the maximum number of subgradients that could have – but otherwise *haven't* – been communicated to an active agent at activation time. It is straightforward to see that $\nu \leq \tau$.[5]
- The *cumulative unavailability* D_t is given by $D_t = \sum_{s=1}^{t} \operatorname{card}(\mathcal{U}_s)$. This generalizes the sum of delays to the multi-agent case; clearly, $D_t \leq \nu t$.

3. Delayed dual averaging and faithful permutations

In this section we present the main algorithmic template that we will use to address the limitations identified in the previous section, and which we call *delayed dual averaging*. We also introduce the notion of "faithful permuation", which plays a major role in the analysis to come, as illustrated by the basic regret bound of Theorem 2 below.

3.1 Delayed dual averaging

To begin, recall that at each time t, an agent computes the point x_t using a collection of *previously received subgradients* $\{g_s : s \in \mathcal{S}_t\}$ where $\mathcal{S}_t \subseteq [t-1]$ represents the set of timestamps corresponding to the subgradients used by the active agent to produce x_t. Put differently, if $s \in \mathcal{S}_t$, then $g_s \in \partial f_s(x_s)$ has been used in the computations leading to playing x_t at time t. On the other hand, $\mathcal{U}_t = [t-1] \setminus \mathcal{S}_t$ collects the timestamps of the feedbacks that are missing for the computation of x_t due to delays.

Our candidate algorithm for this asynchronous setup builds on the dual averaging (DA) master template

$$x_t = \arg\min_{x\in\mathcal{X}} \left\{ \sum_{s<t} \langle g_s, x\rangle + \frac{1}{\eta_t} h(x) \right\} \tag{DA}$$

where $\eta_t \geq 0$ is a learning rate parameter and $h\colon \mathcal{X} \to \mathbb{R}$ is the method's *regularizer*, assumed itself to be continuous and 1-strongly convex relative to some ambient norm $\|\cdot\|$ on $\mathbb{R}^d$. This algorithm is a version of "follow the regularized leader" (FTRL) with linearized losses (Shalev-Shwartz and Singer, 2006; Shalev-Shwartz, 2011); our terminology instead follows Nesterov (2009a) and **?** and is meant to clarify that we will be working with first-order feedback.

Examples. The two most popular candidates for h are the squared ℓ_2-norm $h(x) = \|x\|_2^2/2$ for arbitrary closed convex $\mathcal{X}$ and the negative entropy $h(x) = \sum_k x[k] \log(x[k])$ for simplex constraints $\mathcal{X} = \{x : \sum_{k=1}^{d} x[k] = 1\}$ (here $x[k]$ denotes the k^{th} coordinate of x). The first example is 1-strongly convex relative to the Euclidean norm $\|\cdot\| = \|\cdot\|_2$, while the second one is 1-strongly convex relative to the ℓ^1 norm $\|\cdot\|_1$ on the simplex. ♦

Of course, as stated, (DA) is not a practical algorithm because the active agent $i(t)$ only has at their disposal the subgradients $\{g_s : s \in \mathcal{S}_t\}$ at time t. In view of this, we will consider the *delayed dual averaging* (DDA) policy

$$x_t = \arg\min_{x\in\mathcal{X}} \left\{ \sum_{s\in\mathcal{S}_t} \langle g_s, x\rangle + \frac{h(x)}{\eta_t} \right\} = Q\left(-\eta_t \sum_{s\in\mathcal{S}_t} g_s \right). \tag{DDA}$$

5. For any $t \in [T]$, we have $[t-\tau-1] \subseteq \mathcal{S}_t$ and thus $\mathcal{U}_t = [t-1] \setminus \mathcal{S}_t \subseteq \{t-\tau-1, \dots, t-1\}$ which consists of τ elements. On the other hand, if, for some reason, one feedback is *lost*, say the first one, then, the maximum delay is $\tau = T-1$ while the maximum unavailability is $\nu = 1$, in which case $\nu \ll \tau$.

Algorithm 1: (DDA) – from the point of view of agent i

Initialize: $\mathcal{G}_i \leftarrow \emptyset$, $t \leftarrow 1$.
while not stopped **do**
  asynchronously receive feedback g_s from time s: $\mathcal{G}_i \leftarrow \mathcal{G}_i \cup \{s\}$
  if the agent becomes active, i.e., $i(t) = i$ **then**
    $\mathcal{S}_t \leftarrow \mathcal{G}_i$
    Update η_t and play $x_t = \arg\min_{x \in \mathcal{X}} \sum_{s \in \mathcal{S}_t} \langle g_s, x \rangle + \frac{h(x)}{\eta_t}$
    Relay g_s if necessary
  end if
end while

where, for concision, we have now set

$$Q(y) = \arg\max_{x \in \mathcal{X}} \{\langle y, x \rangle - h(x)\}, \quad y \in \mathbb{R}^d,$$

for the *mirror map* induced by h. An intuitive motivation for our algorithmic choice is that every feedback/gradient is put on a equal footing no matter which agent generated the corresponding action or the delay it suffers.[6] Moreover, as long as η_t can be computed locally, (DDA) can indeed be implemented independently by each agent of the network, without requiring a global clock; for a pseudocode implementation, see Algorithm 1.

3.2 Dependencies and faithful permutations

A crucial challenge in (DDA) is the choice of η_t. Indeed, the standard analysis of DA requires the learning rate sequence to be non-increasing, a property that can hardly be ensured in our situation due to the non-monotonicity of the active feedback sequence and the lack of network synchronization. To sidestep this issue, we need to rethink what "time", or the ordering of the timestamps means to (DDA), and how this can be leveraged to construct a valid algorithm.

Our starting point will be to redefine the algorithm's internal clock (and corresponding learning rate) based exclusively on the active timesteamp sets $\mathcal{S}_t$, $t = 1, 2, \dots, T$. To that end, we will start by viewing each timestamp as a node in a "causal graph", and we will include a directed edge from s to t if and only if $s \in \mathcal{S}_t$: this represents a "causal dependency" of t on s in the sense that the gradient g_s has been used to define x_t (cf. Fig. 3). We will refer to this graph as the *dependency graph* associated to the active feedback sequence $\mathcal{S}_t$, $t = 1, 2, \dots, T$, and we will denote it by $\mathcal{G}$; for clarity, we also stress here that we do not assume that this structure is known to the agents.

A first important observation is that the default time ordering $t = 1, 2, \dots$ represents a topological sort of $\mathcal{G}$, i.e., a linear ordering of its vertices such that $s < t$ if there exists a directed edge $s \rightsquigarrow t$ in $\mathcal{G}$.[7] Second, since the update structure of (DDA) is determined entirely by $\mathcal{G}$ and the value of η_t at each vertex of $\mathcal{G}$, it follows that any reshuffling of time that respects the causal structure of $\mathcal{G}$ should be an equally viable alternative for the algorithm. We formalize this idea below via the notion of a *faithful permutation*.

6. See Section 6.2 for more discussion.
7. In particular, this property implies that $\mathcal{G}$ is a directed acyclic graph (DAG).

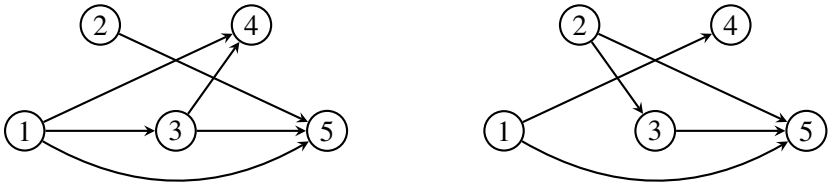


Figure 3: The dependency graphs for the two examples of Fig. 2; the left and right graphs correspond respectively to the single- and multi-agent examples presented therein. The active feedback at time t is exactly the set of in-neighbors of the corresponding vertex.

Definition 1 (Faithful permutations)**.** A permutation σ of $\{1, 2, \dots, T\}$ is *faithful* if and only if, for all $s, t = 1, \dots, T$, we have

$$s \in \mathcal{S}_t \implies \sigma^{-1}(s) < \sigma^{-1}(t). \tag{3}$$

Equivalently, σ is faithful if and only if $\sigma(1), \dots, \sigma(T)$ is a topological ordering of $\mathcal{G}$.

Definition 1 means that the feedback used at time $\sigma(t)$ (whose time indices are in $\mathcal{S}_{\sigma(t)}$) form a subset of $\{g_{\sigma(1)}, \dots, g_{\sigma(t-1)}\}$. Indeed, if $\sigma(s) \in \mathcal{S}_{\sigma(t)}$, then $s = \sigma^{-1}(\sigma(s)) < \sigma^{-1}(\sigma(t)) = t$, i.e., $s \in \{1, \dots, t-1\}$. Thus, a faithful permutation can be seen as a reordering of the time that would still be compatible with the feedback used by each agent at every time. We illustrate this notion with two examples below:

Examples. Clearly, the identity permutation $t \mapsto t$ is always faithful. More interestingly, in the single-agent setting, we can also define the *ordering-by-arrival* as follows: if the k-th received subgradient originates from round t – i.e., $g_t \in \partial f_t(x_t)$ – we set $\sigma(k) = t$, so g_t is the $\sigma^{-1}(t)$-th received gradient.[8] In this notation, the timestamps of all feedback received *before* g_t can be written as $\mathcal{F}_t := \{\sigma(1), \dots, \sigma(\sigma^{-1}(t) - 1)\}$ for that g_t is the $\sigma^{-1}(t)$-th feedback. This shows that σ is indeed a faithful permutation because $\mathcal{S}_t \subseteq \mathcal{F}_t$. ♦

Remark 1. A similar notion was considered by Zimmert and Seldin (2020), but for a completely different purpose. There, the authors aimed to provide optimal algorithms for *single-agent* adversarial bandits with delays. They defined a "dependency-preserving permutation" exactly as the inverse of what we call a faithful permutation, and they used this notion to analyze an algorithm that can "skip" certain rounds of feedback when tuning the algorithm's learning rate. Our definition is motivated by – and tailored to – the multi-agent setting, where the non-monotonicity of the active feedback sequence $\mathcal{S}_t$ plays a major role (we recall that this phenomenon cannot arise in the single-agent case). These elements are altogether absent in the single-agent considerations of Zimmert and Seldin (2020).

3.3 Bounding the regret of delayed dual averaging

We are now in a position to state and prove our main, data-dependent regret guarantee for (DDA) when run with a learning rate that is non-increasing *along a faithful permutation*. For simplicity, we assume throughout the sequel that h is non-negative. This is possible because h is strongly

8. If multiple gradients arrive at a given round, we resolve ties arbitrarily; this ambiguity in the definition of σ plays no role in the analysis.

convex and we can thus always replace h by the non-negative function $h - \min h$ without affecting our algorithms.

Similar to $[t]$ and $\mathcal{U}_t$, for a faithful permutation σ, we also define the set of the first t elements under the new ordering and the set of unavailable elements induced by this ordering as

$$[t]^\sigma = \{\sigma(1), \dots, \sigma(t)\} \quad \text{and} \quad \mathcal{U}_t^\sigma = [t-1]^\sigma \setminus \mathcal{S}_{\sigma(t)}.$$

We have the following theorem concerning the regret of (DDA).

Theorem 2. Let σ be a faithful permutation of $\{1, \dots, T\}$, and assume that (DDA) is run with a learning rate η_t, $t = 1, 2, \dots$, such that $\eta_{\sigma(t+1)} \leq \eta_{\sigma(t)}$ for all t. Then the algorithm enjoys the regret bound

$$\mathbf{Reg}_T(u) \leq \frac{h(u)}{\eta_{\sigma(T)}} + \frac{1}{2}\sum_{t=1}^{T} \eta_{\sigma(t)} \left(\|g_{\sigma(t)}\|_*^2 + 2\|g_{\sigma(t)}\|_* \sum_{s \in \mathcal{U}_t^\sigma} \|g_s\|_* \right). \tag{4}$$

Theorem 2 provides a template regret bound that forms the basis of all the upcoming analysis. To begin, we note that the bound (4) consists of the usual online dual averaging bound (cf. Appendix A) plus a term $\sum_{s \in \mathcal{U}_t^\sigma} \|g_s\|_*$ that reflects the impact of delay. Similar decompositions can be found in McMahan and Streeter (2014), Joulani et al. (2016) and Joulani et al. (2019) respectively for online gradient descent, online mirror descent, and dual averaging.[9] These papers focused on the single-agent (shared-memory) setting and conducted the analysis by either choosing σ as the identity or the ordering by arrival. Theorem 2 thus extends these results by providing a larger class of possible learning rate policies, which enables us to devise efficient and truly implementable learning rate update schemes for the fully decentralized setting in Section 4.

Proof of Theorem 2. As usual, the first step is to bound the algorithm's regret by its linearized counterpart, viz.

$$\mathbf{Reg}_T(u) = \sum_{t=1}^{T} f_t(x_t) - f_t(u) \leq \sum_{t=1}^{T} \langle g_t, x_t - u \rangle.$$

To proceed, we leverage the so-called "perturbed iterate" framework for analyzing asynchronous algorithms in the spirit of Mania et al. (2017) and Joulani et al. (2019). Formally, we define the following virtual iterate sequence

$$\widetilde{x}_t = \arg\min_{x \in \mathcal{X}} \sum_{s=1}^{t-1} \langle g_{\sigma(s)}, x \rangle + \frac{h(x)}{\eta_{\sigma(t)}}.$$

and decompose the sum as:

$$\sum_{t=1}^{T} \langle g_t, x_t - u \rangle = \underbrace{\sum_{t=1}^{T} \langle g_t, \widetilde{x}_{\sigma^{-1}(t)} - u \rangle}_{(a)} + \underbrace{\sum_{t=1}^{T} \langle g_t, x_t - \widetilde{x}_{\sigma^{-1}(t)} \rangle}_{(b)}. \tag{5}$$

We now proceed to bound each term separately.

9. In McMahan and Streeter (2014), the authors work with the specific setting of coordinate-wise unconstrained gradient methods. Therefore, instead of products of norms they have products of scalars in their analysis.

Term (*a*). The first term is exactly the linearized regret of the iterates $\widetilde{x}_1, \dots, \widetilde{x}_T$ that is constructed with the feedback $g_{\sigma(1)}, \dots, g_{\sigma(T)}$. Thus, by analyzing the regret incurred by the dual averaging algorithm (DA) *without* delays, we show in Appendix A that this term can be bounded as

$$\sum_{t=1}^{T} \langle g_t, \widetilde{x}_{\sigma^{-1}(t)} - u \rangle = \sum_{t=1}^{T} \langle g_{\sigma(t)}, \widetilde{x}_t - u \rangle \leq \frac{h(u)}{\eta_{\sigma(T)}} + \frac{1}{2} \sum_{t=1}^{T} \eta_{\sigma(t)} \|g_{\sigma(t)}\|_*^2. \tag{6}$$

Note that the assumption on the learning rate sequence ($\eta_{\sigma(t+1)} \leq \eta_{\sigma(t)}$) is crucial for the derivation of this bound.

Term (*b*). For the second term, we would like to bound the distance between x_t and $\widetilde{x}_{\sigma^{-1}(t)}$, or equivalently, the distance between $x_{\sigma(t)}$ and $\widetilde{x}_t$ (since we shall consider all the $t \in \{1, \dots, T\}$). To that end, we note that

$$x_{\sigma(t)} = Q\left(-\eta_{\sigma(t)} \sum_{s \in \mathcal{S}_{\sigma(t)}} g_s\right) \quad \text{and} \quad \widetilde{x}_t = Q\left(-\eta_{\sigma(t)} \sum_{s \in \mathcal{T}_{t-1}^{\sigma}} g_s\right).$$

Since the permutation σ is faithful, we have $\mathcal{S}_{\sigma(t)} \subseteq \{\sigma(1), .., \sigma(t-1)\} = [t-1]^{\sigma}$. We can then use the non-expansivity of the mirror map (Lemma 21 in Appendix A) to get

$$\|x_{\sigma(t)} - \widetilde{x}_t\| \leq \|\eta_{\sigma(t)} \sum_{s \in \mathcal{U}_t^{\sigma}} g_s\|_* \leq \eta_{\sigma(t)} \sum_{s \in \mathcal{U}_t^{\sigma}} \|g_s\|_*.$$

Subsequently,

$$\begin{aligned} \sum_{t=1}^{T} \langle g_t, x_t - \widetilde{x}_{\sigma^{-1}(t)} \rangle &= \sum_{t=1}^{T} \langle g_{\sigma(t)}, x_{\sigma(t)} - \widetilde{x}_t \rangle \\ &\leq \sum_{t=1}^{T} \|g_{\sigma(t)}\|_* \|x_{\sigma(t)} - \widetilde{x}_t\| \\ &\leq \sum_{t=1}^{T} \eta_{\sigma(t)} \|g_{\sigma(t)}\|_* \sum_{s \in \mathcal{U}_t^{\sigma}} \|g_s\|_*. \end{aligned} \tag{7}$$

Combining (5), (6) and (7), we obtain the desired result. ∎

3.4 Constant learning rate and lag

To get an idea of the optimal regret that the algorithm can achieve, we fix a constant learning rate $\eta_t \equiv \eta$, which we subsequently optimize to minimize the upper-bound on the regret. To proceed, we define the *cumulative lag* as

$$\Lambda_t^{\sigma} = \sum_{s=1}^{t} \left(\|g_{\sigma(s)}\|_*^2 + 2\|g_{\sigma(s)}\|_* \sum_{l \in \mathcal{U}_s^{\sigma}} \|g_l\|_* \right) = \sum_{s \in \mathcal{T}_t^{\sigma}} \|g_s\|_*^2 + 2 \sum_{\{s,l\} \in \mathcal{D}_t^{\sigma}} \|g_s\|_* \|g_l\|_*, \tag{8}$$

where

$$\mathcal{D}_t^{\sigma} = \{\{\sigma(s), l\} : s \in [t], l \in \mathcal{U}_s^{\sigma}\}.$$

In words, $\{s', l\} \in \mathcal{D}_t^\sigma$ if *i)* g_l is not used to define $x_{s'}$; and *ii)* after reordering by σ, l comes before s' and s' comes before $\sigma(t)$. We also write $\Lambda_t = \Lambda_t^{\mathrm{id}}$ for the lag associated to the standard time ordering and define $D_t^\sigma = \mathrm{card}(\mathcal{D}_t^\sigma) = \sum_{t=1}^T \mathrm{card}(\mathcal{U}_t^\sigma)$.

In the above, while D_t^σ captures the "total delay" in terms of the reordering induced by σ, the cumulative lag Λ_t^σ regroups the actual errors caused by the inability of the learners to compensate the missing feedback, and gives the most fine-grained characterization of the effect of delayed feedback on the regret. In the single-agent setting, Joulani et al. (2016) and McMahan and Streeter (2014) also considered the same quantity but in the special case where σ is the ordering-by-arrival permutation discussed in the previous section. In general, it is clear that $\Lambda_t^\sigma \leq (T + 2D_t^\sigma)G^2$ provided that all subgradients are bounded in norm by G; moreover, if σ is the identity permutation, we further have $D_t^\sigma = D_t \leq \nu t$. With all this in mind, a direct application of Theorem 2 gives the following series of more explicit bounds.

Corollary 3. Let σ be a faithful permutation and assume that (DDA) is run with a constant learning rate $\eta > 0$. Then:

- If $\|g_t\|_*$ is uniformly bounded and $\eta = \Theta(1/\sqrt{\nu T})$, then $\mathbf{Reg}_T(u) = \mathcal{O}(\sqrt{\nu T})$.
- If $\|g_t\|_*$ is uniformly bounded and $\eta = \Theta(1/\sqrt{D_T^\sigma})$, then $\mathbf{Reg}_T(u) = \mathcal{O}(\sqrt{D_T^\sigma})$.
- If $\eta = \Theta(1/\sqrt{\Lambda_T^\sigma})$, then $\mathbf{Reg}_T(u) = \mathcal{O}(\sqrt{\Lambda_T^\sigma})$.

Corollary 3 recapitulates several types of regret bound that we can expect from (DDA), depending on the tuning of η_t (either by using a pessimistic upper bound on the delays and the norms of the gradients, or using the actual delays and/or received gradients). In particular, if we focus on the standard time ordering $\sigma = \mathrm{id}$, Corollary 3 allows us to recover the optimal data-dependent bound of $\mathcal{O}(\sqrt{\Lambda_T})$ that was previously obtained for the single-agent setting by Joulani et al. (2016) and McMahan and Streeter (2014). Moreover, if we further assume that $\|g_t\|_* \leq G$ for all $t \in [T]$, we have $\Lambda_T \leq (T + 2D_T)G^2$, which leads to the well-known $\mathcal{O}(\sqrt{D_T})$ bound on the regret (see e.g., Quanrud and Khashabi, 2015).

On the downside, Corollary 3 would seem to suggest that the derived regret bounds depend on the choice of the permutation σ, a concept that is relevant for the analysis, but which is otherwise devoid of physical meaning (at least, relative to the sequence of events as it unfolds in real time). Because of this, the computation of the optimal learning rates required by Corollary 3 seems beyond reach in practice – even if we assume that the various quantities involved are somehow known to the agents. However, as we show below, *this is not the case:* the values of both D_T^σ and Λ_T^σ are independent of σ, and hence, so are the bounds of Corollary 3. To prove this, we first provide a new characterization of the set $\mathcal{D}_t^\sigma$ which is of independent interest:

Proposition 4. Let σ be a faithful permutation. Then

$$\mathcal{D}_t^\sigma = \{\{s, l\} \subseteq [t]^\sigma : s \text{ and } l \text{ are not adjacent in } \mathcal{G}\}. \tag{9}$$

Proof. By definition of the dependency graph, s and l are not adjacent in $\mathcal{G}$ if and only if $\{s \notin \mathcal{S}_l, l \notin \mathcal{S}_s\}$. We will thus show that

$$\mathcal{D}_t^\sigma = \{\{s, l\} \subseteq [t]^\sigma : s \notin \mathcal{S}_l, l \notin \mathcal{S}_s\}.$$

This relies on a two-way inclusion argument.

Inclusion ("$\subseteq$"). Let $s \in [t]$ and $l \in \mathcal{U}_s^\sigma = \mathcal{T}_{s-1}^\sigma \setminus \mathcal{S}_{\sigma(s)}$. By definition of $\mathcal{T}_t^\sigma$ we have $\sigma(s) \in \mathcal{T}_t^\sigma$ and $l \in \mathcal{T}_{s-1}^\sigma \subseteq \mathcal{T}_t^\sigma$. It remains to prove that $\sigma(s) \notin \mathcal{S}_l$. We exploit the equivalence

$$l \in \mathcal{T}_{s-1}^\sigma \iff \sigma^{-1}(l) \leq s-1 \iff \sigma^{-1}(l) < \sigma^{-1}(\sigma(s)) \iff \sigma(s) \notin \mathcal{T}_{\sigma^{-1}(l)}^\sigma. \tag{10}$$

To conclude, we use the fact that σ is a faithful permutation and accordingly $\mathcal{S}_l \subseteq \mathcal{T}_{\sigma^{-1}(l)-1}^\sigma \subseteq \mathcal{T}_{\sigma^{-1}(l)}^\sigma$. Along with (10) we deduce that $\sigma(s) \notin \mathcal{S}_l$.

Containment ("$\supseteq$"). Let $\{s, l\} \subset [t]^\sigma$ such that $s \notin \mathcal{S}_l$ and $l \notin \mathcal{S}_s$. We assume without loss of generality $\sigma^{-1}(l) < \sigma^{-1}(s)$. This is indeed equivalent to $l \in \mathcal{T}_{\sigma^{-1}(s)-1}^\sigma$ and therefore $l \in \mathcal{U}_{\sigma^{-1}(s)}^\sigma$. We complete the proof by noting that $s \in [t]^\sigma$ if and only if $\sigma^{-1}(s) \in [t]$. ∎

In contrast to the original definition of $\mathcal{D}_t^\sigma$, the characterization of Proposition 4 is independent of the ordering of the timestamps. By defining $\mathcal{G}_t^\sigma$ as the subgraph of $\mathcal{G}$ spanned by the vertices of $[t]^\sigma$ in $\mathcal{G}$, the proposition says that $\mathcal{D}_t^\sigma$ contains exactly the non-adjacent vertex pairs of $\mathcal{G}_t^\sigma$. With this in mind, we readily obtain the following important corollary:

Corollary 5. For any two faithful permutations σ and ρ, we have $\mathcal{D}_T^\sigma = \mathcal{D}_T^\rho$, and, a fortiori, $D_T^\sigma = D_T^\rho$ and $\Lambda_T^\sigma = \Lambda_T^\rho$. In other words, the regret bounds of Corollary 3 are independent of σ.

Proof. Simply note that $[T]^\sigma = [T]^\rho = [T]$. ∎

Corollary 5 shows that the regret bounds of Corollary 3 are indeed meaningful, as they do not depend on any "virtual" reordering of time by a faithful permutation. However, given that the quantities Λ_T and D_T cannot be assumed known beforehand, the agents might need to employ a much more conservative learning rate of the order of $\Theta(1/\sqrt{vT})$ to minimize their regret. We address this important issue via the design of suitable adaptive learning methods in the next section.

4. Tuning the learning rate in the presence of delays

In this section, we exploit the template bound of Theorem 2 to design efficient leaning rates that provably achieve low regret. To clarify our objective, we begin by identifying the main desiderata that we seek to achieve:

i) *Anytime / Restart-free:* the algorithm should not require the knowledge of the horizon T and/or include a restart schedule where previous information is discarded.

ii) *Coordination-free:* the learning rate of each agent must be computable based *exclusively* on local information without any need for agent coordination.

iii) *Data-dependent bounds:* the algorithm's regret guarantees should feature the actual gradients observed instead of an upper bound thereof.

iv) *Adaptivity to delays:* the algorithm's regret should depend on the observed delays and not only on a pessimistic, worst-case estimate thereof.

To derive a learning rate with the above properties, we will employ an "inverse-root-sum-square" policy in the spirit of AdaGrad and other adaptive algorithms. (see Lemma 6 in Appendix B for the

details). This is perhaps easiest to illustrate in the case $\sigma = \mathrm{id}$: here, to obtain an $\mathcal{O}(\sqrt{\Lambda_T})$ regret, we could employ the policy $\eta_t = 1/\sqrt{\Lambda_t} = 1/\sqrt{\sum_{s=1}^t \lambda_s}$ where

$$\lambda_s = \|g_s\|_*^2 + 2\|g_s\|_* \sum_{l \in \mathcal{U}_s} \|g_l\|_*.$$

The key in the analysis of this policy is provided by the following standard lemma (dating back at least to Auer et al., 2002b, and proven for completeness in Appendix B):

Lemma 6. For any sequence of real numbers $\lambda_1, \dots, \lambda_T$ with $\sum_{s=1}^t \lambda_s > 0$ for all $t \in [T]$, we have

$$\sum_{t=1}^T \frac{\lambda_t}{\sqrt{\sum_{s=1}^t \lambda_s}} \leq 2\sqrt{\sum_{t=1}^T \lambda_t}.$$

Based on this lemma, it is straightforward to show that (DDA) with learning rate $\eta_t = 1/\sqrt{\Lambda_t}$ incurs at most $\mathcal{O}(\sqrt{\Lambda_T})$ regret. However, this policy is not implementable because it involves unobserved feedback – and hence violates one of our principal desiderata. In the rest of this section, we show how this difficulty can be circumvented in many relevant scenarios.

4.1 Pessimistic non-adaptive learning rate

To set the stage for the analysis to come, we begin by assuming that the agents know an upper bound for the maximum delay τ or the norms of the observed gradients. This leads to $\lambda_s \leq G^2(1+2\nu) \leq G^2(1+2\tau)$ and subsequently $\Lambda_t \leq G^2(1+2\tau)t$. Given this preliminary result, it can be tempting to choose $\eta_t = \Theta(1/G\sqrt{t(1+2\tau)})$. This is however still unrealistic as the agents do not know the exact value of t, and may only estimate it by using $t \leq \mathrm{card}(\mathcal{S}_t) + \tau + 1$. To justify this strategy, we will need to prove that the corresponding learning rate is indeed non-increasing along some faithful permutation in order to apply Theorem 2. For this, we will require the following assumption.

Assumption 1. *If $s \in \mathcal{S}_t$ then* $\mathrm{card}(\mathcal{S}_s) < \mathrm{card}(\mathcal{S}_t)$.

In words, the assumption requires that if g_s is used to compute x_t, then x_s is computed with fewer gradients than x_t. This is a fairly mild requirement which is in turn implied by the upcoming Assumption 2 (see the accompanying discussion). In particular, if the agents also relay the information $\mathrm{card}(\mathcal{S}_t)$ as well, Assumption 1 can be ensured by delaying the actual usage of a received feedback when necessary.[10] Then, when the actual delays are bounded by τ, the gradients $\{g_1, \dots, g_{t-\tau-1}\}$ can always be used for computing x_t. Therefore, introducing this extra delay will not increase the maximum delay and has no effect on the regret bound of the following proposition.

Proposition 7. Suppose that Assumption 1 holds, the maximum delay is bounded by τ, and the norm of the observed gradients is bounded by G. Assume further that (DDA) is run with the learning rate

$$\eta_t = \frac{R}{G\sqrt{(1+2\tau)(\mathrm{card}(\mathcal{S}_t) + \tau + 1)}}.$$

10. In this case, $\mathcal{S}_t$ refers to the timestamps of the gradients that are used for the computation of x_t; however, this does not necessarily contain all the gradients that the active agent $i(t)$ has received by time t.

Then, for any u such that $h(u) \leq R^2$, the generated points $x_1, \dots, x_T$ enjoy the regret bound

$$\mathbf{Reg}_T(u) \leq 2RG\sqrt{(T+\tau)(1+2\tau)}.$$

Proof. We will in fact prove a stronger variant for which it is sufficient to assume that τ is an upper bound of the maximum unavailability. Let $\overline{\Lambda}_t = G^2(1+2\tau)(\mathrm{card}(\mathcal{S}_t)+\tau+1)$ so that $\eta_t = R/\sqrt{\overline{\Lambda}_t}$. We choose a permutation σ that satisfies if $\overline{\Lambda}_s < \overline{\Lambda}_t$ then $\sigma^{-1}(s) < \sigma^{-1}(t)$ (obviously, such a permutation always exists). From Assumption 1 and the definition of $\overline{\Lambda}_t$ we know that σ is a faithful permutation. Moreover, $(\overline{\Lambda}_t)_t$ is non-decreasing along σ: indeed, if this were not the case – that is, if $\overline{\Lambda}_{\sigma(t+1)} < \overline{\Lambda}_{\sigma(t)}$ for some t – we would have $t+1 = \sigma^{-1}(\sigma(t+1)) < \sigma^{-1}(\sigma(t)) = t$, a contradiction.

We now proceed to prove $\mathrm{card}(\mathcal{U}_t^\sigma) \leq \tau$, or equivalently $\mathrm{card}(\mathcal{S}_{\sigma(t)}) \geq t-1-\tau$. For this we show $[t]^\sigma \subseteq [\mathrm{card}(\mathcal{S}_{\sigma(t)})+\tau+1]$. Since $\overline{\Lambda}_t$ is non-decreasing along σ, for $s \leq t$ we have $\mathrm{card}(\mathcal{S}_{\sigma(s)}) \leq \mathrm{card}(\mathcal{S}_{\sigma(t)})$. Using the bounded unavailability assumption we get $\mathrm{card}([\sigma(s)-1] \setminus \mathcal{S}_{\sigma(s)}) \leq \tau$ so that $\sigma(s) - 1 - \mathrm{card}(\mathcal{S}_{\sigma(s)}) \leq \tau$ and subsequently $\sigma(s) \leq \mathrm{card}(\mathcal{S}_{\sigma(t)}) + \tau + 1$. This proves $[t]^\sigma \subseteq [\mathrm{card}(\mathcal{S}_{\sigma(t)})+\tau+1]$.

From $\mathrm{card}(\mathcal{U}_t^\sigma) \leq \tau$ it follows immediately $\lambda_t^\sigma := \|g_{\sigma(t)}\|_*^2 + 2\|g_{\sigma(t)}\|_* \sum_{s\in\mathcal{U}_t^\sigma} \|g_s\|_* \leq G^2(1+2\tau)$ for all t. Along with $t \leq \mathrm{card}(\mathcal{S}_{\sigma(t)})+\tau+1$ we deduce $\Lambda_t^\sigma \leq G^2(1+2\tau)(\mathrm{card}(\mathcal{S}_{\sigma(t)})+\tau+1) = \overline{\Lambda}_{\sigma(t)}$. Applying Theorem 2 and the AdaGrad lemma (Lemma 6), we obtain

$$\begin{aligned}
\mathbf{Reg}_T(u) &\leq \frac{h(u)}{\eta_{\sigma(T)}} + \frac{1}{2}\sum_{t=1}^T \eta_{\sigma(t)} \left(\|g_{\sigma(t)}\|_*^2 + 2\|g_{\sigma(t)}\|_* \sum_{s\in\mathcal{U}_t^\sigma} \|g_s\|_* \right) \\
&\leq R\sqrt{\overline{\Lambda}_{\sigma(T)}} + \frac{R}{2}\sum_{t=1}^T \frac{\lambda_t^\sigma}{\sqrt{\overline{\Lambda}_{\sigma(t)}}} \\
&\leq R\sqrt{\overline{\Lambda}_{\sigma(T)}} + \frac{R}{2}\sum_{t=1}^T \frac{\lambda_t^\sigma}{\sqrt{\Lambda_t^\sigma}} \\
&\leq R\sqrt{\overline{\Lambda}_{\sigma(T)}} + R\sqrt{\Lambda_T^\sigma} \leq 2R\sqrt{\overline{\Lambda}_{\sigma(T)}}.
\end{aligned}$$

Our assertion then follows by noting that $\mathrm{card}(\mathcal{S}_{\sigma(T)}) \leq \sigma(T) - 1 \leq T - 1$. ∎

Proposition 7 shows that, even in the fully decentralized case where no global clock is available, it is *still* possible to design implementable algorithms that retain the optimal $\mathcal{O}(\sqrt{\tau T})$ regret bound. Our next step is to further improve the algorithm so that it can adapt to both the *data* and the *delay* of the feedback. The aforementioned characterization of delay will turn out to be crucial for this task.

4.2 Adaptation to delays in distributed systems

To design a learning rate policy that adapts to both data and delays, we have to find a way to estimate Λ_t by only using local information of each agent. To that end, define for each agent i the *individual* ordering-by-arrival as a permutation σ_i of $\{1, \dots, T\}$ such that the k-th received feedback of i comes from $x_{\sigma_i(k)}$ (played by i or another player), i.e., the k-th received feedback of i is $g_{\sigma_i(k)} \in \partial f_{\sigma_i(k)}(x_{\sigma_i(k)})$. With this notation, we can define the set of all feedback received *before* g_t *by agent*

i; since g_t is the $\sigma_i^{-1}(t)$-th feedback, this set can be defined as $\mathcal{F}_{i,t} \coloneqq \{\sigma_i(1), \sigma_i(2), \dots, \sigma_i(\sigma_i^{-1}(t)-1)\}$.

Using these definitions and looking closely at the definition of the lag (8), we notice that:

1. The quantity $\sum_{s=1}^{t} \|g_{\sigma(s)}\|_*^2$ cannot be known at instant $\sigma(t)$ since the set of gradients available at that time is $\mathcal{S}_{\sigma(t)}$. It is thus natural to consider approximating it by $\sum_{s\in\mathcal{S}_{\sigma(t)}} \|g_s\|_*^2$.
2. For each t the quantity $\sum_{\{s,l\}\in\mathcal{D}_t^\sigma} \|g_s\|_*\|g_l\|_*$, gathering the pairs of feedback of $[t]^\sigma$ satisfying the relation $\{s \notin \mathcal{S}_l, l \notin \mathcal{S}_s\}$ (Proposition 4), is generally unknown. Building on the works of Joulani et al. (2016) and McMahan and Streeter (2014), this sum can be approximated by $\sum_{s\in\mathcal{S}_t}(\|g_s\|_* \sum_{l\in\mathcal{F}_{i(t),s}\setminus\mathcal{S}_s}\|g_l\|_*)$. In words, for all $s \in \mathcal{S}_t$, the worker $i(t)$ aggregates the feedback received before g_s but was not used to generate g_s.

Putting these two points together, a reasonable surrogate for Λ_t^σ would be

$$\Gamma_t = \sum_{s\in\mathcal{S}_t}\left(\|g_s\|_*^2 + 2\|g_s\|_* \sum_{l\in\mathcal{F}_{i(t),s}\setminus\mathcal{S}_s}\|g_l\|_*\right).$$

To make Γ_t a valid approximation, we would need s to satisfy $s \notin \mathcal{S}_l$ whenever $l \in \mathcal{F}_{i(t),s} \setminus \mathcal{S}_s$ given the characterization of Proposition 4. This leads to the following mild assumption: when an agent receives a gradient g_t, they must have already received all the feedback that was used to compute it.

Assumption 2. *For every worker $i \in \mathcal{M}$ and all $t = 1, 2, \dots$, we have $\mathcal{S}_t \subseteq \mathcal{F}_{i,t}$.*

The above assumption is notably verified in the following scenarios: *i*) a coordinator-worker scheme in which the transmission of the gradients occurs *in order*, in first-come, first-serve manner; *ii*) broadcasting of newly received and computed gradient over a fixed communication network; *iii*) whenever two agents communicate their gradient pools are synchronized and the gradients are exchanged in the order they become available to the agents. As a consequence, Assumption 2 is satisfied in many relevant setups and can otherwise be enforced by imposing *iii)* at the price of a slightly higher communication cost.

Now, since the active agent $i(t)$ at time t knows $\mathcal{S}_t$ (by definition) and $\mathcal{F}_{i(t),s}$ for $s \in \mathcal{S}_t$ (by construction), the quantity Γ_t is indeed computable with purely local information. The agents can thus run (DDA) with a learning rate of the form $\eta_t = \Theta(1/\sqrt{\Gamma_t})$. The obtained algorithm, which we call AdaDelay-Dist, is detailed in Algorithm 2; its principal regret guarantee is given below:

Theorem 8. Suppose that the maximum delay is bounded by τ, the norm of the gradients are bounded by G, and that Assumption 2 holds. Assume further that (DDA) is run with the learning rate

$$\eta_t = \frac{R}{\sqrt{\Gamma_t + \beta}}. \tag{AdaDelay–Dist}$$

where $\beta > 0$ is a positive constant. Then, for all u such that $h(u) \le R^2$, the algorithm enjoys the regret bound

$$\mathbf{Reg}_T(u) \le 2R\sqrt{\Lambda_T} + 2R\sqrt{\beta} + \frac{R}{\sqrt{\beta}}G^2(2\tau+1)^2.$$

The bound of Theorem 8 differs from the optimal data-dependent bound by at most a time-independent constant, and this is achieved at the worst-case cost of transmitting an additional scalar

Algorithm 2: AdaDelay–Dist – from the point of view of agent i

Initialize: $\mathcal{G}_i \leftarrow \emptyset, \widetilde{\Gamma}_i \leftarrow \beta > 0$
while not stopped **do**
  asynchronously receive g_t along with $\sum_{s\in\mathcal{S}_t}\|g_s\|_*$ from other agents
    $\widetilde{\Gamma}_i \leftarrow \widetilde{\Gamma}_i + \|g_t\|_*^2 + 2\|g_t\|_*(\sum_{s\in\mathcal{G}_i}\|g_s\|_* - \sum_{s\in\mathcal{S}_t}\|g_s\|_*)$
    $\mathcal{G}_i \leftarrow \mathcal{G}_i \cup \{g_t\}$
    Relay the information if necessary
  asynchronously receive g_t as a feedback
    Retrieve $\sum_{s\in\mathcal{S}_t}\|g_s\|_*$ from the memory
    $\widetilde{\Gamma}_i \leftarrow \widetilde{\Gamma}_i + \|g_t\|_*^2 + 2\|g_t\|_*(\sum_{s\in\mathcal{G}_i}\|g_s\|_* - \sum_{s\in\mathcal{S}_t}\|g_s\|_*)$
    $\mathcal{G}_i \leftarrow \mathcal{G}_i \cup \{g_t\}$
    Send g_t and $\sum_{s\in\mathcal{S}_t}\|g_s\|_*$ to other agents
  if the agent becomes active, i.e., $i(t) = i$ **then**
    $\mathcal{S}_t \leftarrow \mathcal{G}_i$
    $\eta_t \leftarrow R/\sqrt{\widetilde{\Gamma}_i}$
    Play $x_t = \arg\min_{x\in\mathcal{X}} \sum_{s\in\mathcal{S}_t}\langle g_s, x\rangle + \frac{h(x)}{\eta_t}$
  end if
end while

(i.e., $\sum_{s\in\mathcal{S}_t}\|g_s\|_*$) per element of feedback sent. Moreover, we should also stress that the algorithm does not use the global time: as in the case of Proposition 7 time indices are present in Algorithm 2 only for ease of comprehension, notably to highlight the fact that a worker knows (and keeps track) of the feedback used to produce past points (i.e., $\sum_{s\in\mathcal{S}_t}\|g_s\|_*$ for each point x_t played by the worker). Finally, notice that although the theorem assumes the gradients and delays to be bounded, the algorithm itself does *not* require any knowledge of these bounds. A bad estimate of these quantities would only cause the method to suffer from higher regret at the first iterations.

We will now proceed to prove Theorem 8. To that end, let $\mathcal{A}_{i,t} := \{\{s,l\} : s \in \mathcal{S}_t, l \in \mathcal{F}_{i,s} \setminus \mathcal{S}_s\}$ so that

$$\Gamma_t = \sum_{s\in\mathcal{S}_t}\left(\|g_s\|_*^2 + 2\|g_s\|_*\sum_{l\in\mathcal{F}_{i(t),s}\setminus\mathcal{S}_s}\|g_l\|_*\right) = \sum_{s\in\mathcal{S}_t}\|g_s\|_*^2 + 2\sum_{\{s,l\}\in\mathcal{A}_{i(t),t}}\|g_s\|_*\|g_l\|_* \tag{11}$$

To simplify the notation, we will write $\mathcal{A}_t = \mathcal{A}_{i(t),t}$. In the following proposition, we show that $\mathcal{A}_t$ can be characterized in the same way as $\mathcal{D}_t^\sigma$.

Proposition 9. Let σ be a faithful permutation and let Assumption 2 hold. Then

$$\mathcal{A}_t = \{\{s,l\} \subseteq \mathcal{S}_t : s \text{ and } l \text{ are not adjacent in } \mathcal{G}\}$$

Proof. The proof is similar to that of Proposition 4, and we defer it to Appendix B. ∎

Thanks to Proposition 4 and Proposition 9, comparing $\mathcal{D}_t^\sigma$ with $\mathcal{A}_{\sigma(t)}$ amounts to comparing $[t]^\sigma$ with $\mathcal{S}_{\sigma(t)}$. Using the bounded delay assumption, we can prove the following properties on a faithful permutation.

Proposition 10. Let σ be a faithful permutation and assume that the maximum delay is bounded by τ. We have *(a)* $[t]^\sigma \subseteq [\sigma(t)+\tau]$; *(b)* $[t]^\sigma \setminus \mathcal{S}_{\sigma(t)} \subseteq \{\sigma(t)-\tau, \dots, \sigma(t)+\tau\}$; and *(c)* $|\sigma(t)-t| \leq \tau$.

Main idea of the proof. Proving (a) and (b) simply uses the definition of faithful permutations and the maximum delay, while to prove (c) we also leverage the fact that σ is a permutation. To streamline our discussion, the complete proof is again deferred to Appendix B. ∎

Interestingly, Proposition 10(a) shows that when the delays are bounded by τ, a faithful permutation can at most move an element τ steps away from its original position. We are now ready to provide the complete proof of Theorem 8.

Proof of Theorem 8. Let $\overline{\Lambda}_t = \Gamma_t + \beta$ so that $\eta_t = R/\sqrt{\overline{\Lambda}_t}$ and σ be a permutation such that *i)* if $\overline{\Lambda}_s < \overline{\Lambda}_t$ then $\sigma^{-1}(s) < \sigma^{-1}(t)$; *ii)* if $\overline{\Lambda}_s = \overline{\Lambda}_t$ and $s \in \mathcal{S}_t$ then $\sigma^{-1}(s) < \sigma^{-1}(t)$. $(\overline{\Lambda}_t)_t$ is obviously non-decreasing along σ (see proof of Proposition 7). We claim that this is a faithful permutation. For this, let $s \in \mathcal{S}_t$ and we would like to show $\sigma^{-1}(s) < \sigma^{-1}(t)$. By Assumption 2 we have $\mathcal{S}_s \subseteq \mathcal{F}_{i(t),s}$ and from $s \in \mathcal{S}_t$ it holds $\mathcal{F}_{i(t),s} \subseteq \mathcal{S}_t$; accordingly, $\mathcal{S}_s \subseteq \mathcal{S}_t$. Invoking Proposition 9 we deduce $\mathcal{A}_s \subseteq \mathcal{A}_t$. Using (11) we then get $\overline{\Lambda}_s \leq \overline{\Lambda}_t$. This inequality along with $s \in \mathcal{S}_t$ imply $\sigma^{-1}(s) < \sigma^{-1}(t)$.

In the remainder of the proof, we will use the notation $\Gamma_t = \Lambda_T = \Lambda_T^\sigma$ for $t > T$. Let us prove that $\Gamma_{\sigma(t)+2\tau+1} \geq \Lambda_t^\sigma$ for $t \in [T]$. This is the case when $\sigma(t)+2\tau+1 > T$ by the previous definition. Otherwise, with (8), (11), Propositions 4 and 9, this is equivalent to proving that $[t]^\sigma \subseteq \mathcal{S}_{\sigma(t)+2\tau+1}$. The inclusion holds since on one hand, by Proposition 10 we have $[t]^\sigma \subseteq [\sigma(t)+\tau]$ and on the other hand $[\sigma(t)+\tau] \subseteq \mathcal{S}_{\sigma(t)+2\tau+1}$ by the definition of maximum delay. This also shows $\mathcal{S}_{\sigma(t)} \subseteq \mathcal{S}_{\sigma(t)+2\tau+1}$, and accordingly, $\overline{\Lambda}_{\sigma(t)+2\tau+1} \geq \overline{\Lambda}_{\sigma(t)}$. The inequality is still true when $\sigma(t)+2\tau+1 > T$ as $\Gamma_t \leq \Lambda_T$ always holds by Propositions 4 and 9 and $\mathcal{S}_t \subseteq [T]$. Applying Theorem 2 gives

$$\begin{aligned}
\mathbf{Reg}_T(u) &\leq \frac{h(u)}{\eta_{\sigma(T)}} + \frac{1}{2}\sum_{t=1}^T \eta_{\sigma(t)}\left(\|g_{\sigma(t)}\|_*^2 + 2\|g_{\sigma(t)}\|_* \sum_{s\in\mathcal{U}_t^\sigma}\|g_s\|_*\right) \\
&\leq R\sqrt{\overline{\Lambda}_{\sigma(T)}} + \frac{R}{2}\sum_{t=1}^T \frac{\lambda_t^\sigma}{\sqrt{\overline{\Lambda}_{\sigma(t)}}} \\
&= R\sqrt{\overline{\Lambda}_{\sigma(T)}} + \frac{R}{2}\sum_{t=1}^T \left(\frac{1}{\sqrt{\overline{\Lambda}_{\sigma(t)+2\tau+1}}} + \frac{1}{\sqrt{\overline{\Lambda}_{\sigma(t)}}} - \frac{1}{\sqrt{\overline{\Lambda}_{\sigma(t)+2\tau+1}}}\right)\lambda_t^\sigma,
\end{aligned}$$

where we write $\lambda_t^\sigma = \|g_{\sigma(t)}\|_*^2 + 2\|g_{\sigma(t)}\|_* \sum_{s\in\mathcal{U}_t^\sigma}\|g_s\|_*$. From Proposition 10 we know that $[t]^\sigma \setminus \mathcal{S}_{\sigma(t)} \subseteq \{\sigma(t)-\tau, \dots, \sigma(t)+\tau\}$. Since $[t-1]^\sigma = [t]^\sigma \setminus \{\sigma(t)\}$ and $\sigma(t) \notin \mathcal{S}_{\sigma(t)}$, we deduce that $\mathrm{card}(\mathcal{U}_t^\sigma) \leq 2\tau$ and hence $\lambda_t^\sigma \leq G^2(1+4\tau)$. With the non-negativity of $1/\sqrt{\overline{\Lambda}_{\sigma(t)}} - 1/\sqrt{\overline{\Lambda}_{\sigma(t)+2\tau+1}}$ and the fact that $\Lambda_t^\sigma \leq \Gamma_{\sigma(t)+2\tau+1} < \overline{\Lambda}_{\sigma(t)+2\tau+1}$ we then get

$$\mathbf{Reg}_T(u) \leq R\sqrt{\overline{\Lambda}_{\sigma(T)}} + \frac{R}{2}\sum_{t=1}^T \frac{\lambda_t^\sigma}{\sqrt{\Lambda_t^\sigma}} + \frac{R}{2}\sum_{t=1}^T \left(\frac{1}{\sqrt{\overline{\Lambda}_{\sigma(t)}}} - \frac{1}{\sqrt{\overline{\Lambda}_{\sigma(t)+2\tau+1}}}\right) G^2(1+4\tau)$$

$$\leq R\sqrt{\overline{\Lambda}_{\sigma(T)}} + R\sqrt{\Lambda_T^\sigma} + \frac{R}{2}\sum_{t=1}^{T}\left(\frac{1}{\sqrt{\overline{\Lambda}_t}} - \frac{1}{\sqrt{\overline{\Lambda}_{t+2\tau+1}}}\right)G^2(1+4\tau)$$
$$\leq 2R\sqrt{\Lambda_T+\beta} + \frac{R}{2\sqrt{\beta}}(2\tau+1)(4\tau+1)G^2$$
$$\leq 2R\sqrt{\Lambda_T} + 2R\sqrt{\beta} + \frac{R}{\sqrt{\beta}}(2\tau+1)^2G^2$$

The second inequality uses Lemma 6 and reorders the timestamps of the sum; the third inequality upper bounds both $\overline{\Lambda}_{\sigma(T)}$ and $\Lambda_T^\sigma = \Lambda_T$ by $\Lambda_T + \beta$ and lower bounds $\overline{\Lambda}_t$ by β; in the last inequality we employ the fact that $\sqrt{a+b} \leq \sqrt{a} + \sqrt{b}$ for all $a, b \geq 0$. This concludes the proof. ∎

4.3 Adaptation to unbounded delays in the single-agent setting

In this part, we will show that when there is only one agent (i.e., $M = 1$), we can extend the ideas developed in the previous section to cope even with *unbounded* delays. In fact, in this situation the agent knows exactly the delay of each feedback and how each iterate is computed, so they can tune their learning rate accordingly. This is in sharp contrast with the decentralized case in which the agents are in general unable to estimate the number of actions that have been played in the network but for which they have not received the corresponding feedback (i.e., $\mathrm{card}(\mathcal{U}_t)$).

To put all this in motion, let G be an upper bound on the norms of gradients that we assume to be known by the agent, and let $\mathcal{F}_t = \mathcal{F}_{1,t}$ denotes the set of feedback (represented by their timestamps) received before g_t. Our goal is to provide an upper bound of $\Lambda_t = \Lambda_t^{\mathrm{id}}$ that is as tight as possible. As in Section 4.2, this is done in two steps (we write below $\mathcal{D}_t = \mathcal{D}_t^{\mathrm{id}}$ for simplicity)

1. The quantity $\sum_{s=1}^t \|g_s\|_*^2$ can be approximated by $\sum_{s\in\mathcal{S}_t}\|g_s\|_*^2$. Clearly,
$$\sum_{s=1}^{t}\|g_s\|_*^2 \leq \sum_{s\in\mathcal{S}_t}\|g_s\|_*^2 + G^2(\mathrm{card}(\mathcal{U}_t)+1);$$
2. A proxy for $\sum_{\{s,l\}\in\mathcal{D}_t}\|g_s\|_*\|g_l\|_*$, is $\sum_{s\in\mathcal{S}_t}(\|g_s\|_*\sum_{l\in\mathcal{F}_s\setminus\mathcal{S}_s}\|g_l\|_*)$. Thanks to Proposition 4 and Proposition 9, we have indeed
$$\sum_{\{s,l\}\in\mathcal{D}_t}\|g_s\|_*\|g_l\|_* \leq \sum_{s\in\mathcal{S}_t}\left(\|g_s\|_*\sum_{l\in\mathcal{F}_s\setminus\mathcal{S}_s}\|g_l\|_*\right) + 2G^2(\mathrm{card}(\mathcal{D}_t) - \mathrm{card}(\mathcal{A}_t)).$$

In summary, we have shown that $\Lambda_t \leq \Gamma_t + G^2\tilde{\tau}_t$ where $\tilde{\tau}_t := t + 2D_t - \mathrm{card}(\mathcal{S}_t) - 2\,\mathrm{card}(\mathcal{A}_t)$. This has the following immediate consequences:

Proposition 11. Assume that the norms of the gradients are bounded by G and the sequence of active feedback is non-decreasing, i.e., $\mathcal{S}_t \subseteq \mathcal{S}_{t+1}$. Assume further that delayed dual averaging (DDA) is run with the learning rate sequence

$$\eta_t = \min\left(\eta_{t-1}, \frac{R}{\sqrt{\Gamma_t + G^2\tilde{\tau}_t}}\right) \qquad \text{(AdaDelay+)}$$

Algorithm 3: AdaDelay+

Initialize: $\mathcal{G} \leftarrow \emptyset$, $t \leftarrow 1$, $\tilde{\tau} \leftarrow 0$, $\Gamma \leftarrow 0$.
while not stopped **do**
  if receive feedback g_t **then**
    $\tilde{\tau} \leftarrow \tilde{\tau} - 1 - 2(\mathrm{card}(\mathcal{G}) - \mathrm{card}(\mathcal{S}_t))$
    $\Gamma \leftarrow \Gamma + \|g_t\|_*^2 + 2\|g_t\|_*(\sum_{s\in\mathcal{G}}\|g_s\|_* - \sum_{s\in\mathcal{S}_t}\|g_s\|_*)$
    $\mathcal{G} \leftarrow \mathcal{G} \cup \{g_t\}$
  else if requested to play an action x_t **then**
    $\mathcal{S}_t \leftarrow \mathcal{G}$
    $\tilde{\tau} \leftarrow \tilde{\tau} + 1 + 2((t-1) - \mathrm{card}(\mathcal{S}_t))$
    $\widetilde{\Gamma} \leftarrow \max(\widetilde{\Gamma}, \Gamma + G^2\tilde{\tau})$
    $x_t \leftarrow \arg\min_{x\in\mathcal{X}} \sum_{s\in\mathcal{S}_t}\langle g_s, x\rangle + (\sqrt{\widetilde{\Gamma}}/R)h(x)$
    $t \leftarrow t+1$
  end if
end while

where $\tilde{\tau}_t = t + 2D_t - \mathrm{card}(\mathcal{S}_t) - 2\,\mathrm{card}(\mathcal{A}_t)$. Then, for any u such that $h(u) \le R^2$, the generated points $x_1, \dots, x_T$ enjoy the regret bound

$$\mathbf{Reg}_T(u) \le 2R \max_{1\le t\le T}\sqrt{\Gamma_t + G^2\tilde{\tau}_t} \le 2R\min\left(\max_{1\le t\le T}\sqrt{\Lambda_t + G^2\tilde{\tau}_t}, G\sqrt{T+2D_T}\right).$$

Proof. The proof is detailed in Appendix B. We apply Theorem 2 with the choice $\sigma = \mathrm{id}$ and conclude by using the inequality $\Lambda_t \le \Gamma_t + G^2\tilde{\tau}_t$ and the AdaGrad lemma (Lemma 6). ∎

We refer to this new adaptive scheme as AdaDelay+ and we provide one possible pseudocode implementation as Algorithm 3. Notice that we do not use directly $\eta_t = R/\sqrt{\Gamma_t + G^2\tilde{\tau}_t}$ since we want the learning rate to be non-increasing. To the best of our knowledge, AdaDelay+ is the first online algorithm with regret guarantees that are both data- *and* delay-dependent, all the while bypassing the bounded delay assumption. In particular, its regret bound achieves the best of both worlds:

1. When the delays are bounded by τ, we have $\tilde{\tau} \le 2\tau^2 + 3\tau + 1$ (proved in Appendix B), so this worst-case bound still outperforms (by an additive constant) the data-dependent bound of Theorem 8. In the same setting, Joulani et al. (2016) also proposed an adaptive algorithm based on FTRL-Prox with a regret bound of the same order.

2. It also achieves the optimal square-root dependence on the cumulative unavailability D_T no matter whether the delays are bounded or not.

5. An Optimistic Variant

In previous sections, we have established regret guarantees with respect to the worst case scenario. In particular, the losses that we encounter can be arbitrary, and even adversarial. Nonetheless, the environment can have a much more benign nature: there may be patterns of loss functions which can be exploited to achieve a smaller regret (e.g., losses generated by a game mechanism, slowly-varying function sequence). In this spirit, optimistic algorithms exploit the predictability of the loss sequence

to obtain an improved regret bound of the algorithm. In the unconstrained Euclidean setup ($\mathcal{X} = \mathbb{R}^d$, $h = 1/2\|\cdot\|^2$) that we will focus on in the following, the algorithm writes

$$\begin{aligned} x_t &= x_{t-1} - \eta\, g_{t-\frac{1}{2}}, \\ x_{t+\frac{1}{2}} &= x_t - \eta\, \widetilde{g}_{t+\frac{1}{2}}. \end{aligned} \tag{OptGD}$$

The first update $x_t = x_{t-1} - \eta g_{t-\frac{1}{2}}$ is a classical online gradient step. However, for optimistic methods, the point x_t *is not played* at time t; instead, the agent plays $x_{t+\frac{1}{2}} = x_t - \eta\, \widetilde{g}_{t+\frac{1}{2}}$ after sensing the gradient of f_t by designing a gradient *guess* $\widetilde{g}_{t+\frac{1}{2}} = \widetilde{g}_{t+\frac{1}{2}}(x_1, g_{\frac{3}{2}}, \dots, g_{t-\frac{1}{2}})$. This is the *optimistic step*. Following this action, the player suffers a loss $f_t(x_{t+\frac{1}{2}})$ and receives the feedback $g_{t+\frac{1}{2}} \in \partial f_t(x_{t+\frac{1}{2}})$.

The regret of (OptGD) was shown (Chiang et al., 2012; Joulani et al., 2017; Mohri and Yang, 2016; Rakhlin and Sridharan, 2013) to be bounded by

$$\mathbf{Reg}_T(u) \leq \frac{\|u - x_1\|^2}{2\eta} + \sum_{t=1}^{T} \frac{\eta}{2} \|g_{t+\frac{1}{2}} - \widetilde{g}_{t+\frac{1}{2}}\|^2. \tag{12}$$

By optimally choosing η, we attain a regret in $\mathcal{O}\left(\sqrt{\sum_{t=1}^{T} \|g_{t+\frac{1}{2}} - \widetilde{g}_{t+\frac{1}{2}}\|^2}\right)$.

This bound gets smaller as $\widetilde{g}_{t+\frac{1}{2}}$ gets closer to $g_{t+\frac{1}{2}}$ (i.e., when the optimistic guess is good), while we recover the regret of vanilla online gradient descent for $\widetilde{g}_{t+\frac{1}{2}} = 0$ (no optimistic guess). A possible choice in practice is to use the last received feedback as a guess, i.e., $\widetilde{g}_{t+\frac{1}{2}} = g_{t-\frac{1}{2}}$, in which case, favorable guarantees can be derived when the function sequence has a small total variation and when these functions are smooth (see e.g., Chiang et al., 2012; Joulani et al., 2017).

In this section, we present how Delayed Dual Averaging can be extended to incorporate an optimistic step in the unconstrained Euclidean setup. Importantly, we show that the dual averaging step has to be done with a smaller learning rate than the optimistic step.

5.1 Delayed Optimistic Dual averaging

While optimistic gradient descent (OptGD) successfully leverages the predictability of the loss sequence for achieving a smaller regret, the effect of delay on this algorithm remains, as far as we are aware, unknown.

By extending (DDA) to incorporate an optimistic step, delayed optimistic dual averaging can then be stated as follows:[11]

$$\begin{aligned} x_t &= \operatorname*{arg\,min}_{x \in \mathbb{R}^d} \sum_{s \in \mathcal{S}_t} \langle g_{s+\frac{1}{2}}, x\rangle + \frac{\|x - x_1\|^2}{2\eta_t} = x_1 - \eta_t \sum_{s \in \mathcal{S}_t} g_{s+\frac{1}{2}}, \\ x_{t+\frac{1}{2}} &= \operatorname*{arg\,min}_{x \in \mathbb{R}^d} \langle \widetilde{g}_{t+\frac{1}{2}}, x\rangle + \frac{\|x - x_t\|^2}{2\gamma_t} = x_t - \gamma_t\, \widetilde{g}_{t+\frac{1}{2}}. \end{aligned} \tag{DOptDA}$$

Following our delay framework, x_t is computed using gradients from time moments $\mathcal{S}_t$. Similarly, $\widetilde{g}_{t+\frac{1}{2}}$ must be derived solely based on information available to the active agent $i(t)$ at time t.

11. The same algorithm (in a more general form) is called optimistic FTRL in Joulani et al. (2017). We choose to employ the term optimistic dual averaging to maintain consistency with preceding sections.

One key feature of our algorithm is we allow the optimistic step (i.e., the step that leads to $x_{t+\frac{1}{2}}$) of (DOptDA) to use a larger learning rate than the actual update step (i.e., the step that obtains x_{t+1}), i.e., $\gamma_t \geq \eta_t$. This additional flexibility allows us to compensate the missing information that have not arrived due to delays and provides the following regret bound proved in Appendix C.1.

Theorem 12. Assume that the maximum delay is bounded by τ. Let delayed optimistic dual averaging (DOptDA) be run with learning rate sequences $(\eta_t)_{t\in[T]}$, $(\gamma_t)_{t\in[T]}$ satisfying $\eta_{t+1} \leq \eta_t$ and $(2\tau+1)\eta_t \leq \gamma_t$ for all t. Then the regret of the algorithm (evaluated at the points $x_{\frac{3}{2}}, \dots, x_{T+\frac{1}{2}}$) satisfies

$$\mathbf{Reg}_T(u) \leq \frac{\|u - x_1\|^2}{2\eta_T} + \sum_{t=1}^{T} \frac{\gamma_t}{2} \left(\|g_{t+\frac{1}{2}} - \widetilde{g}_{t+\frac{1}{2}}\|^2 - \|\widetilde{g}_{t+\frac{1}{2}}\|^2 \right).$$

In Theorem 12, we successfully show that (DOptDA) retains the desired property of undelayed optimistic gradient descent: the regret of the algorithm is solely determined by the distance between $g_{t+\frac{1}{2}}$ and $\widetilde{g}_{t+\frac{1}{2}}$ (see Eq. (12)). Precisely, the theorem guarantees a regret in $\mathcal{O}\left(\sqrt{\tau \sum_{t=1}^{T} \|g_{t+\frac{1}{2}} - \widetilde{g}_{t+\frac{1}{2}}\|^2}\right)$ for fix learning rate sequences $\eta_t \equiv \eta$, $\gamma_t \equiv (2\tau+1)\eta$ that are optimally chosen. Similar to the case of delayed mirror descent and delayed dual averaging, an additional factor of $\sqrt{\tau}$ appears in the regret bound, and their regret is recovered tightly by setting $\widetilde{g}_{t+\frac{1}{2}} = 0$.

Remark 2. The bounded delay assumption can in fact be relaxed in Theorem 12. Nonetheless, we choose to adopt this assumption for ease of understanding. Otherwise, denoting $d_t = \mathrm{card}(\mathcal{U}_t) + \mathrm{card}(\{s \in [T] : t \in \mathcal{U}_s\}) + 1$ and employing a constant update learning rate $\eta_t \equiv \eta$ and $\gamma_t = d_t\eta$, we achieve a regret in $\mathcal{O}\left(\sqrt{\sum_{t=1}^{T} d_t \|g_{t+\frac{1}{2}} - \widetilde{g}_{t+\frac{1}{2}}\|^2}\right)$. Note that $\sum_{t=1}^{T} d_t = 2D + T$ and when $\widetilde{g}_{t+\frac{1}{2}} = 0$ the bound can be inferred from Theorem 2 with the choice $\sigma = \mathrm{id}$.

5.2 The necessity of scale separation for robustness to delay

In the following, we discuss the *necessity* of having a relatively aggressive optimistic step compared to the update ($\gamma_t \geq \eta_t$) in order to be robust to delay.[12] Note that taking a more aggressive extrapolation update compared to the actual state update was shown to clearly improve the robustness of the extragradient method with respect to both rates and convergence itself in Hsieh et al. (2020).

For this, we consider linear losses $f_t = \langle g_t, \cdot \rangle$ and uniform delay τ (i.e., every feedback becomes available after a delay of τ time steps).[13] We define the τ-*variation* of the loss sequence by $C_T^\tau = \sum_{t=1}^{T} \|g_t - g_{t-\tau}\|^2$ where we set $g_t = 0$ for $t \leq 0$. For ease of notation we further denote $C_T^{\tau^+} = C_T^{\tau+1}$. The following corollary is immediate from Theorem 12.

Corollary 13. In the context of linear losses $f_t = \langle g_t, \cdot \rangle$ and uniform delay τ ($\mathcal{S}_t = [t-\tau-1]$ for all t), running delayed optimistic dual averaging (DOptDA) with $\widetilde{g}_{t+\frac{1}{2}} = g_{t-\tau-1}$ and constant learning rates $\eta = R/\sqrt{(2\tau+1)C_T^{\tau^+}}$ and $\gamma = (2\tau+1)\eta$ where $R \geq \|u - x_1\|$ guarantees the regret bound

$$\mathbf{Reg}_T(u) \leq R\sqrt{(2\tau+1)C_T^{\tau^+}}.$$

12. The optimistic step is also called *extrapolation* step to mirror the vocabulary of the extragradient method Korpelevich (1976).

13. For linear losses, the gradient does not depend on the calling point and thus $g_{t+\frac{1}{2}} = \nabla f_t(x_{t+\frac{1}{2}}) = g_t$.

This results indicates that with an optimistic learning rate γ taken $(2\tau+1)$ times bigger than the update learning rate η, one can guarantee a regret bound of the order of the square root of the $(\tau+1)$-variation. In contrast, we now demonstrate the impossibility to obtain a regret that is sub-linear in $C_T^{\tau^+}$ when $\gamma = \eta$ (or even when $\gamma \leq \tau\eta$).

Theorem 14. Consider the setup of Corollary 13. Let $\eta = \eta(R, T, \tau, C_T^{\tau^+})$ be uniquely determined by $R \geq \|u - x_1\|$, the time horizon T, the uniform delay τ, and the $(\tau+1)$-variation $C_T^{\tau^+}$. If we run delayed optimistic dual averaging (DOptDA) with $\widetilde{g}_{t+\frac{1}{2}} = g_{t-\tau-\frac{1}{2}}$ and $\gamma \leq \tau\eta$, it is impossible to guarantee a regret in $o(\max(C_T^{\tau^+}, \sqrt{T}))$.

Proof. The proof is reported in Appendix C.2; its construction is partially inspired by Chiang et al. (2012), and as a special case, in the undelayed setting, we recover the result that the optimistic step is necessary to guarantee a regret in $\mathcal{O}\left(\sqrt{\sum_{t=1}^T \|g_t - g_{t-1}\|^2}\right)$.

Nonetheless, in the original proof of Chiang et al. (2012), the learning rate was first fixed and then a loss sequence was constructed to yield large regret, which could possibly also prevent optimistic algorithms to achieve low regret. Our approach fixes this fallacy by informing the algorithm of the variation in advance so that optimistic algorithms provably obtain low regrets on these sequences (cf. Corollary 13). ∎

Finally, we also show that among all the online algorithms with the same prior information, the bound achieved in Corollary 13 is tight in its dependence on τ and $C_T^{\tau^+}$.

Proposition 15. For any online learning algorithm with prior knowledge of T, τ and $\overline{C^\tau} \geq C_T^{\tau^+}$, there exists a sequence of linear losses such that if the feedback is subject to constant delay τ, then the regret of the algorithm on this sequence with respect to a vector u with $\|u - x_1\| \leq 1$ is $\Omega(\sqrt{\tau\overline{C^\tau}})$.

Proof. The proof is reported in Appendix C.3. It combines the standard $\Omega(\sqrt{T})$ lower bound of undelayed online learning with idea from Langford et al. (2009). ∎

Thus, in this section we showed that using (DOptDA) *with a double learning rate strategy* enables to achieve a $\mathcal{O}(\sqrt{\tau C_T^{\tau^+}})$ regret which is tight among online learning methods and out of reach of single learning rate (DOptDA).

5.3 Delayed online learning with slow variation

Now that we laid out our main results concerning the optimistic variant of delayed dual averaging, we investigate the choice of $\widetilde{g}_{t+\frac{1}{2}}$ for slowly varying loss functions $(f_t)_{t\in[T]}$.

For this, we consider the case where the full gradient ∇f_t is obtained as a feedback (and not only $g_t = \nabla f_t(x_t)$). Using this kind of feedback, we can compute the gradient of the last received function at the current point immediately[14] and use it as a guess for the current function's gradient. Formally, we make the following assumption.

Assumption 3. *The feedback associated to time step t is the whole vector field $V_t = \nabla f_t$, the evaluation of which at any point $x \in \mathbb{R}^d$ is immediate and does not induce any delay.*

14. i.e., without any delay, the delays considered here are either due to communication between agents or inherent to the feedback mechanism.

The first part of the assumption is sometimes referred as the "full-information" online learning model, and is typically satisfied when the learning system is used for prediction (e.g., classification, regression). In fact, in such problems, the actions of the agents represent the model parameters, for which the whole loss and its gradient can be computed once the corresponding data is observed (Shalev-Shwartz, 2011).

With this assumption, we can set $\widetilde{g}_{t+\frac{1}{2}} = \widetilde{V}_t(x_t)$ where $\widetilde{V}_t$ is some *past* vector field (i.e., $\widetilde{V}_t = V_s$ for some $s \in \mathcal{S}_t$). Now, for smooth losses, the following regret bound can be derived.

Theorem 16. Let the maximum delay be bounded by τ and that Assumption 3 holds. Assume in addition that the vector fields V_t are L-Lipschitz continuous. Take $\widetilde{g}_{t+\frac{1}{2}} = \widetilde{V}_t(x_t)$, $\eta_{t+1} \leq \eta_t$, $(2\tau+1)\eta_t \leq \gamma_t$, and $2\gamma_t^2 L^2 \leq 1$. Then, the regret of delayed optimistic dual averaging (DOptDA) (evaluated at the points $x_{\frac{3}{2}}, \dots, x_{T+\frac{1}{2}}$) satisfies

$$\mathbf{Reg}_T(u) \leq \frac{\|u - x_1\|^2}{2\eta_T} + \sum_{t=1}^{T} \gamma_t \|V_t(x_t) - \widetilde{V}_t(x_t)\|^2.$$

Proof. The proof is immediate from Theorem 12 and is deferred to Appendix C.4. ∎

Theorem 16 reduces the problem of choosing an adequate vector $\widetilde{g}_{t+\frac{1}{2}}$ to that of choosing an operator $\widetilde{V}_t$ which approximates well V_t. In our setup of full gradient feedback with a loss sequence evolving slowly over time, one natural option is reuse some recent function for the constitution of $\widetilde{V}_t$. Since we are in a distributed setting, the evolution of the loss functions may have both global and local components. We discuss these two typical cases below.

Example 1 (Global variation)**.** If the loss functions vary slowly following a global trend, we can timestamp every gradient field which makes it possible to choose $\widetilde{V}_t = V_{\tilde{t}}$ where $\tilde{t} = \max \mathcal{S}_t$, i.e., the active agent $i(t)$ uses the most recent data available at hand (independent of its source) when playing x_t. This would however require the agents to share the whole vector field V_t.

Example 2 (Local variation)**.** If the loss functions vary slowly for all the agents, the active agent $i(t)$ can choose the last feedback corresponding to a point it played, i.e., $\widetilde{V}_t = V_{\tilde{t}}$ where $\tilde{t} = \max\{s \in \mathcal{S}_t : i(s) = i(t)\}$. Compared to Example 1, we gain in terms of both data privacy and communication efficiency since only the gradients $g_{t+\frac{1}{2}}$ need to be shared among the agents in this scenario.

Denoting the total deviation of our approximation by $C_T = \sum_{t=1}^{T} \|V_t(x_t) - \widetilde{V}_t(x_t)\|^2$, Theorem 16 guarantees a regret in $\mathcal{O}(R^2\tau L + R\sqrt{\tau C_T})$ for suitably chosen constant learning rate sequences $\eta_t \equiv \eta$ and $\gamma_t \equiv \gamma$. In both Examples 1 and 2, C_T characterizes some variation of the loss sequence over time. However, the optimal choice of the η and γ allowing us to obtain the aforementioned regret guarantee depends on C_T, which cannot be known in advance. To circumvent this issue, we can again design an adaptive learning rate schedule in the spirit of AdaGrad by assuming knowledge on an universal bound for the difference $\|V_t(x_t) - \widetilde{V}_t(x_t)\|^2$. For the following result, we simply resort to the standard assumption of bounded gradients.

Proposition 17. Let the maximum delay be bounded by τ and let Assumptions 2 and 3 hold. Further suppose that V_t are L-Lipschitz continuous and both $V_t, \widetilde{V}_t$ have their norm bounded by G. Then for

any u such that $\|u - x_1\| \leq R$, running delayed optimistic dual averaging (DOptDA) with $\widetilde{g}_t = \widetilde{V}_t(x_t)$,

$$\gamma_t = \min\left(\frac{R\sqrt{4\tau+1}}{2\sqrt{\left(\sum_{s\in\mathcal{S}_t}\|V_s(x_s)-\widetilde{V}_s(x_s)\|^2 + 4G^2(\tau+1)\right)}}, \frac{1}{\sqrt{2}L}\right),$$

and

$$\eta_t = \min\left(\frac{R}{2\sqrt{(4\tau+1)\left(\sum_{s\in\mathcal{S}_t}\|V_s(x_s)-\widetilde{V}_s(x_s)\|^2 + 4G^2(3\tau+1)\right)}}, \frac{1}{\sqrt{2}L(4\tau+1)}\right)$$

guarantees

$$\mathbf{Reg}_T(u) \leq \max\left(\sqrt{2}R^2L(4\tau+1), 2R\sqrt{(4\tau+1)(C_T + 4G^2(3\tau+1))}\right).$$

Proof. The proof is deferred to Appendix C.5. Notice that the adaptive learning rates are not necessarily non-increasing and therefore Theorem 16 can not be directly applied. To address this challenge, we rely on the use of faithful permutations and adapt both Theorem 12 and Theorem 16 to accommodate more flexible learning rate schedules. ■

Compared to the optimal regret that can be achieved with prior knowledge of C_T, the bound is only degraded by a constant factor. To implement this learning rate schedule, the computation of γ_t and η_t needs to be made possible. This would require the agents to relay $\|V_t(x_t) - \widetilde{V}_t(x_t)\|$ in addition to $g_{t+\frac{1}{2}} = V_t(x_{t+\frac{1}{2}})$ after receiving V_t.

Remark 3. At the price of a worse dependence on the constants, we can use the difference $\|V_t(x_{t+\frac{1}{2}}) - \widetilde{V}_t(x_t)\|$ instead of $\|V_t(x_t) - \widetilde{V}_t(x_t)\|$ in the computation of the learning rates, which prevents us from an extra evaluation of the operator; see e.g., Joulani et al., 2017, Corollary 9.

6. Discussion

In this section, we discuss several links of our work to other, existing results whose detailed presentation would have otherwise interrupted the flow of our paper.

6.1 Related work

Our work lies at the interface between multiple active research areas, each tackling a specific aspect of the general framework considered in this paper. We provide below a more focused view into each of these topics, namely: *i*) online learning with delays; *ii*) multi-agent online learning; *iii*) distributed online optimization; and *iv*) asynchronous optimization.

Online learning with delays. The research on the delayed feedback problem in online learning was pioneered by Weinberger and Ordentlich (2002), in which it was shown that running $\tau+1$ independent learners guaranteed the minimax regret $\mathcal{O}(\sqrt{\tau T})$ when the feedback is uniformly delayed by τ time steps. The same strategy was further analyzed by Joulani et al. (2013) for more complex delay

mechanisms. However, maintaining a pool of learners can be prohibitively resource intensive. Therefore, another line of research focuses on investigating the effect of delays on gradient-based methods.

In Langford et al. (2009), the same $\mathcal{O}(\sqrt{\tau T})$ bound on the regret was first derived for a slowed-down version of online gradient descent (i.e., running the algorithm with smaller learning rates) under the constant delay assumption. Comprehensive studies were later provided by McMahan and Streeter (2014), Quanrud and Khashabi (2015) and Joulani et al. (2016). In more detail, denoting by D the aggregated feedback delay after T rounds, Quanrud and Khashabi (2015) established a regret bound in $\mathcal{O}(\sqrt{D})$ for online gradient descent and dual averaging, and suggested using the classical doubling trick to dynamically adjust the learning rate.[15] Assuming bounded delays, both McMahan and Streeter (2014) and Joulani et al. (2016) devised delay-adaptive methods in order to obtain data-dependent bounds. The former centered on online gradient descent in the unconstrained case while the latter was based on online mirror descent and FTRL-prox. Under the same setting, Joulani et al. (2019) also presented an adaptive method with a data-dependent bound which however has a worst-case dependence on the delay. Very recently, Cao et al. (2020) extended the delayed feedback analysis to an online saddle-point algorithm which handled the constraints through Lagrangian relaxation.

Our work differs from the above in that we consider a multi-agent setup in which feedback does not arrive to the agents at the same time. To the best of our knowledge, this situation has never been considered before and gives rise to extra challenges that call for novel techniques. In fact, even though both McMahan and Streeter (2014) and Joulani et al. (2019) also dealt with *asynchronous* online optimization, they focused on the coordinator-worker setting. It is thus possible there for the agents/workers to exploit a quantity stocked on the server (e.g., an inexact global clock in Joulani et al. (2019)). This is generally impossible in our setup.

The impact of delays has equally been studied in the literature on multi-armed bandits, both stochastic (Pike-Burke et al., 2018; Vernade et al., 2017) and adversarial (Cesa-Bianchi et al., 2018; Li et al., 2019; Cesa-Bianchi et al., 2019). The setting of these papers is quite different from the online optimization problems we consider in our paper, so there is no overlap in results or techniques.

Multi-agent online learning. Multi-agent online learning encompasses a broad spectrum of problems, including distributed online optimization (discussed next), multi-agent bandits (Bar-On and Mansour, 2019; Cesa-Bianchi et al., 2019; Szorenyi et al., 2013; Xu et al., 2015), and games (Cesa-Bianchi and Lugosi, 2006; Héliou et al., 2020). In a very recent paper, Cesa-Bianchi et al. (2020) considered a cooperative online learning problem in which a different set of agents is activated at each round, they encounter the same loss, and they receive immediately the relevant gradient feedback after playing. While this setting is different from our own (there are no feedback delays and a fixed underlying communication graph is assumed), this is the first paper that we are aware of and which considers asynchronous activation in multi-agent online convex optimization problems.

Distributed online optimization. In distributed online convex optimization, the agents cooperatively optimize a sequence of global costs which are defined as the sum of local loss functions, each associated with an agent. Under this setup, consensus-based distributed algorithms were proposed and shown to achieve sublinear regret (Hosseini et al., 2013; Yan et al., 2012). More recently, Shahrampour and Jadbabaie (2017) and Zhang et al. (2019) further modified these algorithms to

15. Due to a lack of consensus in the literature, Quanrud and Khashabi (2015) used the name online mirror descent to refer to dual averaging. See Remark 4 for further discussion.

cope with dynamic regret, whereas the case of a time-varying network topology was examined in Mateos-Núnez and Cortés (2014) and Akbari et al. (2015).

Nonetheless, all of the above works concern the *synchronous* scenario, and this is true for both the activation of the agents (all the agents engage in each iteration) and the communication between the agents (which are performed without any delay). In contrast, our framework allows for asynchronous *activations* as well as asynchronous *communication*. Moreover, the underlying communication topology is not modeled explicitly and it is possible to have agents that leave and join freely during the learning process. For sake of concreteness, we further explain in the next section how our method can be used to solve problems that are studied in this line of research.

Asynchronous optimization. For optimization problems that have a sum structure (e.g., over different parts of some dataset, or over several agents), a large part of the literature is based on a random sampling of one or several of the functions leading to a partial use of the data or of the links between agents. This stems from the study of randomized fixed point operators (Bianchi et al., 2015; Combettes and Pesquet, 2015), later extended to delayed settings (Mania et al., 2017; Peng et al., 2016; Leblond et al., 2017). This kind of randomized algorithms is incompatible with the setup considered in our paper in which the agents are *activated* – not sampled.

In the case when a coordinator uses several workers to gather asynchronously gradient feedback, several variants of the proximal gradient algorithm were shown to be efficient, see Aytekin et al. (2016), Vanli et al. (2018) and Mishchenko et al. (2020), the latter allowing for unbounded delays. However, the analyses of these methods are based on the study of the distance between the iterates and the minimizer of the problem which hinders their extension to the online setting.

Finally, we are aware of very few works on open networks where agents can freely join and leave the system. These exceptions treat the simpler problem of averaging local values and focus on the system's stability (Hendrickx and Martin, 2017; Franceschelli and Frasca, 2020; de Galland et al., 2020). These ideas were recently extended to study the stability of decentralized gradient descent in open networks (Hendrickx and Rabbat, 2020) but, again, there is no overlap with our work.

6.2 Online algorithms are not equally robust to delays

In this paper, we have paid exclusive attention to variants of dual averaging (DA). Another family of algorithms that the agents may follow to minimize their regret is online mirror descent (OMD) and its variants. While these two types of methods achieve the same order of regret in many situations, they are not equally robust to delays in our setup, as explained below.

To define OMD, we make the additional assumption that the subdifferential ∂h admits a continuous selection denoted by ∇h. The *Bregman divergence* induced by h is then written as

$$D_h(x, x') = h(x) - h(x') - \langle \nabla h(x'), x - x' \rangle.$$

Subsequently, the update of OMD is

$$x_t = \arg\min_{x \in \mathcal{X}} \left\{ \langle g_{t-1}, x \rangle + \frac{1}{\eta_{t-1}} D_h(x, x_{t-1}) \right\} = Q\left(\nabla h(x_{t-1}) - \eta_{t-1} g_{t-1}\right). \tag{OMD}$$

The main difference between (OMD) and (DA) is that (OMD) generates a new point by combining the last gradient with the last prediction, while (DA) combines all past gradients and then generates a prediction, without explicitly using the last available prediction.

The two algorithms (OMD) and (DA) are not equally robust to delays. Indeed, if feedback from different rounds arrives out-of-order (due to the presence of delays), the natural extension of the methods would be to use them as if they corresponded to the last played point. The sequence of points generated by the algorithms would then be different than with ordered feedback. However, for (DA), the final output after all feedback has arrived will be *the same* for all agents, in contrast to that of (OMD). This is because, in dual averaging, all gradients enter the model with the *same weight* (Nesterov, 2009a, Sec. 1.2); this is a very appealing feature, especially when trying to incorporate delayed gradients or gradients generated by other agents.

Remark 4. The origins of the above methods can be traced to Nemirovski and Yudin (1983), but there is otherwise no consensus on terminology in the literature. The specific formulation (OMD) is sometimes referred to as "eager" mirror descent, in contrast to the method's "lazy" variant which outputs $x_t \leftarrow Q(-\sum_{s<t} \eta_s g_s)$, see e.g., Nesterov (2009a) or Mertikopoulos and Zhou (2019). These variants coincide when h is infinitely "steep" at the boundary of $\mathcal{X}$, i.e., $\operatorname{dom} \partial h \cap \mathcal{X} = \operatorname{ri} \mathcal{X}$; otherwise, they lead to different sequences of play (Kwon and Mertikopoulos, 2017). The "dual averaging" variant is due to Nesterov (2009a), and differs from the lazy variant of (OMD) in that all gradients enter the algorithm with the same weight. From an online learning viewpoint, (DA) can also be seen as a "linearized" version of the FTRL class of algorithms (Shalev-Shwartz and Singer, 2006), and coincides with FTRL when the loss functions encountered are linear. For a survey, see Juditsky et al. (2019), McMahan (2017), Mertikopoulos (2019), and references therein.

6.3 Multi-agent online learning for minimization of global losses

Throughout the paper, our analysis has focused on the agents' *individual* losses (f_t being the loss of *the active agent $i = i(t)$), and thus lead to regret bounds that characterize how much the whole network actually pays.* While these bounds have an interest, networks of agents may also want to monitor *global* losses over the agents. This is typically the case of distributed online optimization, where the agents cooperate to solve a time-varying global problem.

In this section, we demonstrate the flexibility of our framework by showing that the aforementioned algorithms and analyses can be easily extended to this setup. This, on one hand, bridges the gap between our work and the broad corpus of literature on distributed online optimization, and, on the other hand, provides the occasion to directly address the case of open networks where agents can join and depart the optimization process freely.

6.3.1 FROM EFFECTIVE REGRET TO COLLECTIVE REGRET

In distributed optimization, it is often assumed that multiple predictions are made in a same time slot. Formally, we denote by M_t the number of active agents at time t and identify these agents from 1 to M_t instead of identifying each agent independently. This notation clarifies the fact that the agents are anonymous with respect to the algorithm and each other. The functions and the played points at time t are respectively denoted by $f_{1,t}, \dots, f_{M_t,t}$ and $x_{1,t}, \dots, x_{M_t,t}$.

By directly extending the regret defined by (1) to our current setup, we obtain the following:

$$\mathbf{Reg}_T^{\ell}(u) = \sum_{t=1}^{T} \sum_{i=1}^{M_t} f_{i,t}(x_{i,t}) - \sum_{t=1}^{T} \sum_{i=1}^{M_t} f_{i,t}(u), \qquad \text{(Effective Regret)}$$

where the superscript ℓ means that the regret sums over the *local* costs of the learners. Each agent only pays for the function it serves and the ultimate goal for a single agent is to perform well on

the functions that it encounters. As an example, on-device machine learning aims to equip users' personal devices with intelligent machine features such as conversational understanding and image recognition, for the purposes of providing a satisfying user experience to each individual (Shi et al., 2016; Wang et al., 2020).

In contrast, we can also define *global* loss functions $f_t = \sum_{i=1}^{M_t} f_{i,t}$ at every instant t and evaluate each active agents' action with respect to this function. This leads to the following regret formulation:

$$\mathbf{Reg}_T^g(u) = \sum_{t=1}^{T} \sum_{i=1}^{M_t} f_{i,t}(x_{1,t}) - \sum_{t=1}^{T} \sum_{i=1}^{M_t} f_{i,t}(u), \qquad \text{(Collective Regret)}$$

where, instead of evaluating $f_{i,t}$ at the point $x_{i,t}$ played by learner i, we now evaluate all the $f_{i,t}$ at a single point $x_{1,t}$ independently of the worker i. The choice of the *reference agent* can vary with time; it is however possible to fix its index to 1 in advance given that the attribution of the worker indices at each t is arbitrary.

When the number of agents are fixed, *collective regret* reduces to the usual regret formulation employed in the distributed online optimization literature (Hosseini et al., 2013; Shahrampour and Jadbabaie, 2017; Yan et al., 2012). This performance measure suits better the applications related to wireless sensor networks such as distributed estimation (Rabbat and Nowak, 2004) and data fusion (Nakamura et al., 2007; Raza et al., 2015). In fact, sensor networks are mostly deployed for a common objective shared by all the sensors. To attain this objective, the sensor nodes may need to cooperate to track some unknown variable or to collaborate to learn a global assessment of the situation. The collective regret then measures each agent's performance with respect to this *collective* mission, hence the name thereof.

Finally, our formulation also admits the additional flexibility of involving different sets and numbers of agents at each iteration. This is of particularly interest for open multi-agent systems (Hendrickx and Martin, 2017) and elastic distributed learning (Narayanamurthy et al., 2013).

Now, provided that all the loss functions $f_{i,t}$ are G-Lipschitz, the relation between $\mathbf{Reg}_T^g$ and $\mathbf{Reg}_T^\ell$ is quite direct as formulated in the following lemma.

Lemma 18. Assume that all the loss functions $f_{i,t}$ are G-Lipschitz; then,

$$\mathbf{Reg}_T^g(u) \leq \mathbf{Reg}_T^\ell(u) + \sum_{t=1}^{T} \sum_{i=1}^{M_t} G\|x_{i,t} - x_{1,t}\|.$$

6.3.2 Decentralized Delayed Dual Averaging

Thanks to Lemma 18, a bound on the effective regret can be directly translated into one on the collective regret as long as the distances between the agents' predictions for a same moment can be controlled. To illustrate this idea, we adapt DDA to the current setup and bound its induced collective regret for appropriately chosen learning rates. Let us first slightly extend the previously introduced notations and concepts to the current framework: The set of available gradients at time t for a worker i, $\mathcal{S}_{i,t}$, now represents the set of the (learner, time) indices of the feedback available for playing $x_{i,t}$ so that if $(j,s) \in \mathcal{S}_{i,t}$ then necessarily $s \in [t-1]$. The maximum delay τ is to be understood with respect to the global time index t. That is, for every $s \in [t-\tau-1]$ and $j \in [M_s]$ we must have $(j,s) \in \mathcal{S}_{i,t}$. We also introduce the (root mean square) average number of active agents by $\overline{M} = \sqrt{(1/T)\sum_{t=1}^{T} {M_t}^2}$.

With these notations, the update of decentralized delayed dual averaging (D-DDA) writes at time t for an agent i as

$$x_{i,t} = \arg\min_{x \in \mathcal{X}} \sum_{(j,s) \in \mathcal{S}_{i,t}} \langle g_{j,s}, x \rangle + \frac{h(x)}{\eta_{i,t}}, \tag{D-DDA}$$

where $g_{j,s} \in \partial f_{j,s}(x_{j,s})$. In order to understand the mechanics of collective regret in our setup, we restrict our self to the case of a fixed learning rate $\eta_{i,t} \equiv \eta$.[16] To bound the collective regret, three elements come into play:

- the *effective* regret. For this part, we change the time indices to have exactly one point played at each time. We define $N_t = \sum_{s=1}^{t} M_s$ and $N = N_T$; then, the index of worker i at time t is changed to $\phi(i,t) = N_{t-1} + i$ (so that only one action is performed at that time). This maps our problem to the setting of Theorem 2 with $\eta_t \equiv \eta$ and thus with $\sigma = \mathrm{id}$ we get

$$\mathbf{Reg}_T^{\ell}(u) \leq \frac{h(u)}{\eta} + \frac{1}{2} \sum_{m=1}^{N} \eta \left(\|g'_m\|_*^2 + 2\|g'_m\|_* \sum_{l \in [m-1] \setminus \mathcal{S}'_m} \|g'_l\|_* \right) \tag{13}$$

  where $g'_{\phi(i,t)} = g_{i,t}$ and $\mathcal{S}'_{\phi(i,t)} = \{\phi(j,s) : (j,s) \in \mathcal{S}_{i,t}\}$.

- the maximal delay τ. Bounding from above the number of unavailable gradients for a (learner, time) pair and translating this condition to bound $\mathrm{card}([m-1] \setminus \mathcal{S}'_m)$, we get

$$\mathbf{Reg}_T^{\ell}(u) \leq \frac{h(u)}{\eta} + \eta(\tau+1)G^2 \sum_{t=1}^{T} M_t^2. \tag{14}$$

- the non-expansiveness of the mirror map (Lemma 21). This part enables us to go from the effective regret to the collective regret using Lemma 18.

Putting together these points we manage to show the following bound on the collective regret, the full proof being deferred to Appendix D.

Proposition 19. Assume that the maximum delay is bounded by τ and that all the loss functions are G-Lipschitz. For any u satisfying $h(u) \leq R^2$, running decentralized delayed dual averaging (D-DDA) with constant stepsize

$$\eta_{i,t} \equiv \eta = \frac{R}{G\overline{M}\sqrt{(2\tau+1)T}}$$

guarantees the following upper bound on the collective regret

$$\mathbf{Reg}_T^{g}(u) \leq 2RG\overline{M}\sqrt{(2\tau+1)T} = \mathcal{O}(\overline{M}\sqrt{\tau T}).$$

As a sanity check, we can see that when there is no delay ($\tau = 0$) and a fixed number of agents ($M_t \equiv M$), the proposition ensures a regret in $\mathcal{O}(M\sqrt{T})$. This corresponds to the regret achieved by dual averaging on $f_t = \sum_{i=1}^{M} f_{i,t}$ which is MG-Lipschitz (Hazan, 2016, Section 5.2; **?**; see also Appendix A). Nonetheless, since the network of agents may be evolving, the average number of workers $\overline{M}$ may often not be available in advance; neither is the time horizon T nor the current time index t. Exploiting the ideas of Section 4, we provide in Appendix D an implementable learning rate scheme that achieves a regret in $\mathcal{O}(\sqrt{\tau N M_{\max}})$ where $M_{\max} = \max_{1 \leq t \leq T} M_t$.

16. We bypass this limitation in Appendix D by we providing an implementable variable learning rate strategy that provably achieves small collective regret.

7. Concluding remarks

Our aim in this paper was to design adaptive and non-adaptive learning algorithms that can provably achieve low regret in the presence of delays and asynchronicities in both single- and multi-agent environments. This was achieved by means of a general dual averaging framework for handling delays and deriving regret bounds under various learning rate policies including adaptive and data-dependent ones. In addition, we paid special attention to the decentralized case (which includes open networks of agents collaborating to achieve a low collective regret), and we showed how our analysis can be improved further through the use of optimistic policies in slowly-varying environments.

Our work provides the basis for a number of subsequent extensions of independent interest. One particular direction concerns the case where the agents' gradient feedback is corrupted by noise, either exogenous (e.g., stemming from environmental fluctuations) or endogenous (e.g., from mini-batch sampling in the case of empirical risk objectives). Equally important is the choice of target regret measure: in addition to the agents' effective and collective regret, there is a fair number of network applications in which dynamic regret considerations could be equally relevant. In this regard, it would be important to see if the proposed policies lead to low dynamic regret – or how to modify them to achieve this more demanding benchmark.

Finally, if the agents only have access to their incurred losses at each stage, it is possible to reconstruct a *biased* estimate of the corresponding subgradients using a stochastic approximation estimator – either *single-point* (Flaxman et al., 2005) or *two-point* (Agarwal et al., 2010). However, in addition to the bias introduced by this indirect sampling process, the variance of the single-point estimator also grows unbounded as the process unfolds; moreover, in multi-agent settings, the agent performing an update must have access to both the loss incurred by another agent at a different (known) timestamp *and* the actual sampling perturbation / direction employed by the agent that incurred said loss. Phenomena such as these lead to significant difficulties – both technical and conceptual – in the analysis of adaptive algorithms, and require completely new techniques to handle. We defer work on this fruitful research direction to the future.

Acknowledgments

This research was partially supported by the COST Action CA16228 "European Network for Game Theory" (GAMENET), and the French National Research Agency (ANR) in the framework of the "Investissements d'avenir" program (ANR-15-IDEX-02), the LabEx PERSYVAL (ANR-11-LABX-0025-01), MIAI@Grenoble Alpes (ANR-19-P3IA-0003), and the grants ORACLESS (ANR-16-CE33-0004) and ALIAS (ANR-19-CE48-0018-01).

Appendix A. Undelayed dual averaging

Our paper studies several variants of dual averaging in various delayed/distributed setups. For sake of completeness, we include here an analysis of the vanilla dual averaging algorithm in the basic undelayed online learning setting. For a thorough study of the algorithm the readers can refer to the textbook Hazan, 2016, Section 5 and **?**.

Let us consider a sequence of first-order feedback $g_1, \dots, g_T$. At time t dual averaging computes

$$x_t = \arg\min_{x \in \mathcal{X}} \sum_{s=1}^{t-1} \langle g_s, x \rangle + \frac{h(x)}{\eta_t}. \tag{DA}$$

We recall that the mirror map is defined as $Q : y \mapsto \arg\min_{x \in \mathcal{X}} \ \langle -y, x \rangle + h(x)$. We can thus write $x_t = Q(y_t)$ where $y_t = -\eta_t \sum_{s=1}^{t-1} g_s$ may be viewed as the dual point of x_t. We have the following standard result concerning the (linearize) regret achieved by the algorithm.

Proposition 20. Let online dual averaging (DA) be run with non-increasing learning rates $(\eta_t)_{t \in [T]}$. Then, the generated points $x_1, \dots, x_T$ satisfy

$$\mathbf{LinReg}_T(u) := \sum_{t=1}^{T} \langle g_t, x_t - u \rangle \leq \frac{h(u)}{\eta_T} + \frac{1}{2} \sum_{t=1}^{T} \eta_t \|g_t\|_*^2.$$

Proof. Let us fix $u \in \mathcal{X}$ and define the associated estimate sequence

$$\psi_t(x) = \sum_{s=1}^{t-1} \langle g_s, x - u \rangle + \frac{h(x)}{\eta_t}.$$

We will show that

$$\psi_t(x_t) \leq \psi_{t+1}(x_{t+1}) - \langle g_t, x_{t+1} - u \rangle - \frac{1}{2\eta_t} \|x_{t+1} - x_t\|^2. \tag{15}$$

On one hand, by $\eta_{t+1} \leq \eta_t$ and the non-negativity of h,

$$\psi_{t+1}(x_{t+1}) = \psi_t(x_{t+1}) + \langle g_t, x_{t+1} - u \rangle + \left(\frac{1}{\eta_{t+1}} - \frac{1}{\eta_t} \right) h(x_{t+1}) \geq \psi_t(x_{t+1}) + \langle g_t, x_{t+1} - u \rangle. \tag{16}$$

On the other hand, by the definition of x_t we have $y_t \in \partial h(x_t)$. Thus

$$\begin{aligned} \psi_t(x_{t+1}) - \psi_t(x_t) &= \sum_{s=1}^{t-1} \langle g_s, x_{t+1} - x_t \rangle + \frac{h(x_{t+1})}{\eta_t} - \frac{h(x_t)}{\eta_t} \\ &= -\frac{1}{\eta_t} \langle y_t, x_{t+1} - x_t \rangle + \frac{h(x_{t+1})}{\eta_t} - \frac{h(x_t)}{\eta_t} \geq \frac{1}{2\eta_t} \|x_{t+1} - x_t\|^2. \end{aligned} \tag{17}$$

The inequality holds thanks to the 1-strong convexity of h. Summing (16), (17) and rearranging the terms we obtain (15).

Next, let $\eta_{T+1} = \eta_T$ and define x_{T+1} by (DA) (We can do this since x_{T+1} is not used in the computation of $\mathbf{LinReg}_T$). Leveraging on (15), we bound the regret as follows:

$$\begin{aligned} \mathbf{LinReg}_T(u) &:= \sum_{t=1}^{T} \langle g_t, x_t - u \rangle \\ &= \sum_{t=1}^{T} \left(\langle g_t, x_t - x_{t+1} \rangle + \langle g_t, x_{t+1} - u \rangle \right) \\ &\leq \sum_{t=1}^{T} \left(\frac{\eta_t}{2} \|g_t\|_*^2 + \frac{1}{2\eta_t} \|x_{t+1} - x_t\|^2 + \psi_{t+1}(x_{t+1}) - \psi_t(x_t) - \frac{1}{2\eta_t} \|x_{t+1} - x_t\|^2 \right) \end{aligned}$$

$$= \psi_{T+1}(x_{T+1}) - \psi_1(x_1) + \frac{1}{2}\sum_{t=1}^{T} \eta_t \|g_t\|_*^2$$
$$\leq \frac{h(u)}{\eta_T} + \frac{1}{2}\sum_{t=1}^{T} \eta_t \|g_t\|_*^2. \tag{18}$$

In the last inequality we used

$$\psi_{T+1}(x_{T+1}) = \min_{x\in\mathcal{X}} \psi_{T+1}(x) \leq \psi_{T+1}(u) = \frac{h(u)}{\eta_{T+1}} = \frac{h(u)}{\eta_T}$$

and $\psi_1(x_1) = h(x_1)/\eta_1 \geq 0$. (18) is exactly what we want to prove, so this ends the proof. ∎

We next prove the non-expansiveness of the mirror map which are used multiple times in our analyses (for a reference, see e.g., Hiriart-Urruty and Lemaréchal, 2001, Chapter E, Thm. 4.2.1, or Zalinescu, 2002, Cor. 3.5.11).

Lemma 21. The mirror map is non-expansive, i.e., $\|Q(y) - Q(y')\| \leq \|y - y'\|_*$ for all $y, y' \in \mathbb{R}^d$.[17]

Proof. Let $x = Q(y)$ and $x' = Q(y')$. By definition of the mirror map,

$$x = \arg\min_{\hat{x}\in\mathcal{X}} \langle -y, \hat{x}\rangle + h(\hat{x}), \quad x' = \arg\min_{\hat{x}\in\mathcal{X}} \langle -y', \hat{x}\rangle + h(\hat{x}).$$

The optimality condition implies that $y \in \partial h(x)$ and $y' \in \partial h(x')$. Hence, with the Cauchy–Schwarz inequality and the 1-strong convexity of h with respect to $\|\cdot\|$, we have

$$\|y - y'\|_*\|x' - x\| \geq \langle y' - y, x' - x\rangle \geq \|x - x'\|^2.$$

It follows immediately $\|y - y'\|_* \geq \|x - x'\|$. ∎

Appendix B. Missing proofs for variable learning rate methods

In this part, we complete the proofs of the results presented in Section 4. To begin, the well-known "inverse-root-sum" lemma (see e.g., Auer et al., 2002b, Lemma 3.5) is essential for proving the regret guarantees of these methods.

Lemma 6. For any sequence of real numbers $\lambda_1, \dots, \lambda_T$ with $\sum_{s=1}^{t} \lambda_s > 0$ for all $t \in [T]$, we have

$$\sum_{t=1}^{T} \frac{\lambda_t}{\sqrt{\sum_{s=1}^{t} \lambda_s}} \leq 2\sqrt{\sum_{t=1}^{T} \lambda_t}.$$

Proof. The function $y \in \mathbb{R}^+ \mapsto \sqrt{y}$ being concave and has derivative $y \mapsto 1/(2\sqrt{y})$, it holds for every $z \geq 0$,

$$\sqrt{z} \leq \sqrt{y} + \frac{1}{2\sqrt{y}}(z - y).$$

17. Precisely, Q is non-expansive because we are assuming that the strong convexity constant of h is 1. Otherwise it would just be Lipschitz continuous, and clearly this would only influence our results by a constant factor (that depends on the strong convexity constant of h).

Take $y = \sum_{s=1}^{t} \lambda_s$ and $z = \sum_{s=1}^{t-1} \lambda_s$ gives

$$2\sqrt{\sum_{s=1}^{t-1} \lambda_s} + \frac{\lambda_t}{\sqrt{\sum_{s=1}^{t} \lambda_s}} \leq 2\sqrt{\sum_{s=1}^{t} \lambda_s}.$$

We conclude by summing the inequality from $t = 2$ to $t = T$ and using $\sqrt{\lambda_1} \leq 2\sqrt{\lambda_1}$. ∎

Recall that $\mathcal{A}_{i,t} = \{\{s, l\} : s \in \mathcal{S}_t, l \in \mathcal{F}_{i,s} \setminus \mathcal{S}_s\}$ and $\mathcal{A}_t = \mathcal{A}_{i(t),t}$. The next two propositions were used in the proof of Theorem 8.

Proposition 9. Let σ be a faithful permutation and let Assumption 2 hold. Then

$$\mathcal{A}_t = \{\{s, l\} \subseteq \mathcal{S}_t : s \text{ and } l \text{ are not adjacent in } \mathcal{G}\}$$

Proof. We will prove

$$\mathcal{A}_t = \{\{s, l\} \subseteq \mathcal{S}_t : s \notin \mathcal{S}_l, l \notin \mathcal{S}_s\}$$

by a two-way inclusion argument.

Inclusion ("$\subseteq$"). Let $s \in \mathcal{S}_t$ and $l \in \mathcal{F}_{i(t),s} \setminus \mathcal{S}_s$. The inclusion $l \in \mathcal{F}_{i(t),s}$ means that g_l arrives earlier than g_s on node $i(t)$. As all the available gradients are used when playing x_t and $s \in \mathcal{S}_t$, we deduce $l \in \mathcal{S}_t$. On the other hand, $l \in \mathcal{F}_{i(t),s}$ also implies $s \notin \mathcal{F}_{i(t),l}$. Using Assumption 2 we know that $\mathcal{S}_l \subseteq \mathcal{F}_{i(t),l}$, and consequently $s \notin \mathcal{S}_l$.

Containment ("$\supseteq$"). Let $\{s, l\} \subseteq \mathcal{S}_t$ such that $s \notin \mathcal{S}_l$ and $l \notin \mathcal{S}_s$. Since either $l \in \mathcal{F}_{i(t),s}$ or $s \in \mathcal{F}_{i(t),l}$ (but not both) we conclude immediately $\{s, l\} \in \mathcal{A}_{i(t),t} = \mathcal{A}_t$. ∎

Proposition 10. Let σ be a faithful permutation and assume that the maximum delay is bounded by τ. We have (a) $[t]^\sigma \subseteq [\sigma(t)+\tau]$; (b) $[t]^\sigma \setminus \mathcal{S}_{\sigma(t)} \subseteq \{\sigma(t)-\tau, \ldots, \sigma(t)+\tau\}$; and (c) $|\sigma(t)-t| \leq \tau$.

Proof. (*a*) Let $s, t \in [T]$ such that $s \leq t$. We need to prove $\sigma(s) \leq \sigma(t) + \tau$. Assume the opposite, that is, $\sigma(s) > \sigma(t) + \tau$. Then, from the bounded delay assumption, $\sigma(t) \in \mathcal{S}_{\sigma(s)}$. σ being a faithful permutation, this implies $t = \sigma^{-1}(\sigma(t)) < \sigma^{-1}(\sigma(s)) = s$, a contradiction. Finally, $T_t^\sigma = \{\sigma(1), \ldots, \sigma(t)\} = \{\sigma(s) : s \leq t\}$ and hence $T_t^\sigma \subseteq [\sigma(t) + \tau]$.

(*b*) This is immediate from (*a*) and the inclusion $[\sigma(t) - \tau - 1] \subseteq \mathcal{S}_{\sigma(t)}$ which holds since the maximum delay is assumed to be bounded by τ.

(*c*) Fix $t \in [T]$. For all $s \leq t$, we have $\sigma(s) \leq \sigma(t) + \tau$ and therefore $\max_{s \leq t} \sigma(s) \leq \sigma(t) + \tau$. σ being a permutation of $[T]$, it holds $\max_{s \leq t} \sigma(s) \geq t$ and subsequently $t \leq \sigma(t) + \tau$. Similarly, we also have $\sigma(t) - \tau \leq \min_{t \leq s} \sigma(s)$ and $\min_{t \leq s} \sigma(s) \leq t$. This implies $\sigma(t) - \tau \leq t$. Combining the two we conclude $|\sigma(t) - t| \leq \tau$. ∎

We close this section with the single-agent adaptive algorithm (AdaDelay+).

Proposition 11. Assume that the norms of the gradients are bounded by G and the sequence of active feedback is non-decreasing, i.e., $\mathcal{S}_t \subseteq \mathcal{S}_{t+1}$. Assume further that delayed dual averaging (DDA) is run with the learning rate sequence

$$\eta_t = \min\left(\eta_{t-1}, \frac{R}{\sqrt{\Gamma_t + G^2 \tilde{\tau}_t}}\right) \qquad \text{(AdaDelay+)}$$

where $\tilde{\tau}_t = t + 2D_t - \operatorname{card}(\mathcal{S}_t) - 2\operatorname{card}(\mathcal{A}_t)$. Then, for any u such that $h(u) \leq R^2$, the generated points $x_1, \dots, x_T$ enjoy the regret bound

$$\mathbf{Reg}_T(u) \leq 2R \max_{1\leq t\leq T} \sqrt{\Gamma_t + G^2\tilde{\tau}_t} \leq 2R \min\left(\max_{1\leq t\leq T} \sqrt{\Lambda_t + G^2\tilde{\tau}_t}, G\sqrt{T + 2D_T} \right).$$

Proof. Let $\overline{\Lambda}_t = R^2/\eta_t^2$ so that $\eta_t = R/\sqrt{\overline{\Lambda}_t}$. It holds that $\overline{\Lambda}_t \geq \Gamma_t + \tilde{\tau}_t G^2 \geq \Lambda_t$. The first inequality comes from the definition of η_t and the second inequality was shown in Section 4.3. Applying Theorem 2 with $\sigma = \mathrm{id}$ and Lemma 6 yields

$$\begin{aligned}
\mathbf{Reg}_T(u) &\leq \frac{h(u)}{\eta_T} + \frac{1}{2}\sum_{t=1}^{T} \eta_t \left(\|g_t\|_*^2 + 2\|g_t\|_* \sum_{s\in\mathcal{U}_t} \|g_s\|_* \right) \\
&\leq R\sqrt{\overline{\Lambda}_T} + \frac{R}{2}\sum_{t=1}^{T} \frac{1}{\sqrt{\Lambda_t}} \left(\|g_t\|_*^2 + 2\|g_t\|_* \sum_{s\in\mathcal{U}_t} \|g_s\|_* \right) \\
&\leq R\sqrt{\overline{\Lambda}_T} + R\sqrt{\Lambda_T} \leq 2R\sqrt{\overline{\Lambda}_T}.
\end{aligned}$$

Since $\overline{\Lambda}_T = \max_{1\leq t\leq T} \Gamma_t + \tilde{\tau}_t G^2$, we have already proved the first inequality. For the second inequality, we use both $\Gamma_t \leq \Lambda_t$ and $\Gamma_t \leq (\operatorname{card}(\mathcal{S}_t) + 2\operatorname{card}(\mathcal{A}_t))G^2$ (cf. (11)). ∎

When the delays are bounded by a constant, it is possible to further bound $\tilde{\tau}$ from above, as shown below.

Proposition 22. Assume that the maximum delay is bounded by τ. Then $\tilde{\tau} \leq 2\tau^2 + 3\tau + 1$.

Proof. To begin, we have $t - \operatorname{card}(\mathcal{S}_t) \leq \tau + 1$ as $[t - \tau - 1] \subseteq \mathcal{S}_t$. Next, let us consider a pair $\{s, l\} \in \mathcal{D}_t \setminus \mathcal{A}_t$. From Proposition 9 we know that $\{s, l\} \not\subseteq \mathcal{S}_t$, so we have either $s \in \{t - \tau, \dots, t\}$ or $l \in \{t - \tau, \dots, t\}$. Without loss of generality, we suppose $s < l$, then $l \in \{t - \tau, \dots, t\}$. By Proposition 4 we have $s \notin \mathcal{S}_l$, and thus $s \in \{l - \tau, \dots, l - 1\}$. This shows $\operatorname{card}(\mathcal{D}_t \setminus \mathcal{A}_t) \leq \tau(\tau + 1)$. We can therefore conclude $\tilde{\tau}_t \leq 2\tau(\tau + 1) + \tau + 1 = 2\tau^2 + 3\tau + 1$. ∎

Therefore, the bound of Proposition 11 potentially improves upon the bounds obtained in McMahan and Streeter (2014) and Joulani et al. (2016).

Appendix C. Proofs related to the optimistic variant

C.1 Delayed optimistic dual averaging

Theorem 12. Assume that the maximum delay is bounded by τ. Let delayed optimistic dual averaging (DOptDA) be run with learning rate sequences $(\eta_t)_{t\in[T]}$, $(\gamma_t)_{t\in[T]}$ satisfying $\eta_{t+1} \leq \eta_t$ and $(2\tau + 1)\eta_t \leq \gamma_t$ for all t. Then the regret of the algorithm (evaluated at the points $x_{\frac{3}{2}}, \dots, x_{T+\frac{1}{2}}$) satisfies

$$\mathbf{Reg}_T(u) \leq \frac{\|u - x_1\|^2}{2\eta_T} + \sum_{t=1}^{T} \frac{\gamma_t}{2} \left(\|g_{t+\frac{1}{2}} - \widetilde{g}_{t+\frac{1}{2}}\|^2 - \|\widetilde{g}_{t+\frac{1}{2}}\|^2 \right).$$

Proof. Let us consider the virtual iterates

$$\widetilde{x}_t = x_1 - \eta_t \sum_{s=1}^{t-1} g_{s+\frac{1}{2}}.$$

We define the estimate sequence

$$\psi_t(x) = \sum_{s=1}^{t-1} \langle g_{s+\frac{1}{2}}, x - u \rangle + \frac{\|x - x_1\|^2}{2\eta_t}.$$

Notice that the regret is measured with the leading states

$$f_t(x_{t+\frac{1}{2}}) - f_t(u) \leq \langle g_{t+\frac{1}{2}}, x_{t+\frac{1}{2}} - u \rangle = \langle g_{t+\frac{1}{2}}, x_{t+\frac{1}{2}} - \widetilde{x}_{t+1} \rangle + \langle g_{t+\frac{1}{2}}, \widetilde{x}_{t+1} - u \rangle \tag{19}$$

As shown in the proof of Proposition 20, we have

$$\langle g_{t+\frac{1}{2}}, \widetilde{x}_{t+1} - u \rangle \leq \psi_{t+1}(\widetilde{x}_{t+1}) - \psi_t(\widetilde{x}_t) - \frac{1}{2\eta_t} \|\widetilde{x}_{t+1} - \widetilde{x}_t\|^2. \tag{20}$$

For the other term, we recall the definition $\mathcal{U}_t = [t-1] \setminus \mathcal{S}_t$ and define $\nu_t = \operatorname{card}(\mathcal{U}_t)$. Then,

$$\begin{aligned}
\langle g_{t+\frac{1}{2}}, x_{t+\frac{1}{2}} - \widetilde{x}_{t+1} \rangle &= \langle g_{t+\frac{1}{2}}, x_{t+\frac{1}{2}} - x_t \rangle + \langle g_{t+\frac{1}{2}}, x_t - \widetilde{x}_t \rangle + \langle g_{t+\frac{1}{2}}, \widetilde{x}_t - \widetilde{x}_{t+1} \rangle \\
&= \langle g_{t+\frac{1}{2}}, -\gamma_t \widetilde{g}_{t+\frac{1}{2}} \rangle + \langle g_{t+\frac{1}{2}}, \eta_t \sum_{s \in \mathcal{U}_t} g_{s+\frac{1}{2}} \rangle + \langle g_{t+\frac{1}{2}}, \widetilde{x}_t - \widetilde{x}_{t+1} \rangle \\
&= \frac{\gamma_t}{2} \left(\|g_{t+\frac{1}{2}} - \widetilde{g}_{t+\frac{1}{2}}\|^2 - \|g_{t+\frac{1}{2}}\|^2 - \|\widetilde{g}_{t+\frac{1}{2}}\|^2 \right) \\
&\quad + \eta_t \sum_{s \in \mathcal{U}_t} \langle g_{t+\frac{1}{2}}, g_{s+\frac{1}{2}} \rangle + \langle g_{t+\frac{1}{2}}, \widetilde{x}_t - \widetilde{x}_{t+1} \rangle \\
&\leq \frac{\gamma_t}{2} \left(\|g_{t+\frac{1}{2}} - \widetilde{g}_{t+\frac{1}{2}}\|^2 - \|g_{t+\frac{1}{2}}\|^2 - \|\widetilde{g}_{t+\frac{1}{2}}\|^2 \right) \\
&\quad + \frac{\eta_t}{2} \|g_{t+\frac{1}{2}}\|^2 + \frac{1}{2\eta_t} \|\widetilde{x}_t - \widetilde{x}_{t+1}\|^2 + \frac{\nu_t \eta_t}{2} \|g_{t+\frac{1}{2}}\|^2 + \frac{\eta_t}{2} \sum_{s \in \mathcal{U}_t} \|g_{s+\frac{1}{2}}\|^2.
\end{aligned} \tag{21}$$

Combining (19), (20), (21) and summing from $t = 1$ to T yields

$$\begin{aligned}
\mathbf{Reg}_T(u) &\leq \psi_{T+1}(\widetilde{x}_{T+1}) - \psi_1(\widetilde{x}_1) + \sum_{t=1}^{T} \frac{\gamma_t}{2} \left(\|g_{t+\frac{1}{2}} - \widetilde{g}_{t+\frac{1}{2}}\|^2 - \|\widetilde{g}_{t+\frac{1}{2}}\|^2 \right) \\
&\quad + \left(-\frac{\gamma_t}{2} + \frac{(\nu_t + 1)\eta_t}{2} + \sum_{t \in \mathcal{U}_l} \frac{\eta_l}{2} \right) \|g_{t+\frac{1}{2}}\|^2.
\end{aligned} \tag{22}$$

Since the maximum delay is τ, we have $\nu_t \leq \nu \leq \tau$ and if $t \in \mathcal{U}_l$ it holds $l > t \geq l - \tau$ giving that $\operatorname{card}(\{l : t \in \mathcal{U}_l\}) \leq \tau$. Moreover, as $(\eta_t)_{t \in \mathbb{N}}$ is a decreasing sequence, $t \in \mathcal{U}_l$ also implies $\eta_l \leq \eta_t$. The last term of (22) can thus be bounded as following

$$\left(-\frac{\gamma_t}{2} + \frac{(\nu_t + 1)\eta_t}{2} + \sum_{t \in \mathcal{U}_l} \frac{\eta_l}{2} \right) \|g_{t+\frac{1}{2}}\|^2 \leq \frac{1}{2}((2\tau + 1)\eta_t - \gamma_t)\|g_{t+\frac{1}{2}}\|^2 \leq 0, \tag{23}$$

where the second inequality leverages the condition $\gamma_t \geq (2\tau + 1)\eta_t$.

To conclude, we use $\psi_{T+1}(\widetilde{x}_{T+1}) \leq \psi_{T+1}(u)$ and observe that $\psi_1(\widetilde{x}_1) = \psi_1(x_1) = 0$ by definition, so that

$$\mathbf{Reg}_T(u) \leq \frac{\|u - x_1\|^2}{2\eta_{T+1}} + \sum_{t=1}^{T} \frac{\gamma_t}{2} \left(\|g_{t+\frac{1}{2}} - \widetilde{g}_{t+\frac{1}{2}}\|^2 - \|\widetilde{g}_{t+\frac{1}{2}}\|^2 \right).$$

Let $\eta_{t+1} = \eta_t$ and we get the desired bound. ∎

C.2 The necessity of scale separation

Theorem 14. Consider the setup of Corollary 13. Let $\eta = \eta(R, T, \tau, C_T^{\tau^+})$ be uniquely determined by $R \geq \|u - x_1\|$, the time horizon T, the uniform delay τ, and the $(\tau+1)$-variation $C_T^{\tau^+}$. If we run delayed optimistic dual averaging (DOptDA) with $\widetilde{g}_{t+\frac{1}{2}} = g_{t-\tau-\frac{1}{2}}$ and $\gamma \leq \tau\eta$, it is impossible to guarantee a regret in $o(\max(C_T^{\tau^+}, \sqrt{T}))$.

Proof. Assume, for the sake of contradiction, that there exists $\eta = \eta(R, T, \tau, C_T^{\tau^+})$ and a corresponding γ with $\gamma \leq \tau\eta$ such that (DOptDA) with $\widetilde{g}_{t+\frac{1}{2}} = g_{t-\tau-1}$ guarantees a regret in $o(\max(C_T^{\tau^+}, \sqrt{T}))$. Formally, we define a round of the algorithm as a composition a loss sequence, a delay mechanism, a initial point x_1 and a competing vector u, and denote by $\mathcal{R}(R, T, \tau, C_T^{\tau^+})$ the set of all the rounds with time horizon T, $(\tau+1)$-variation $C_T^{\tau^+}$, uniform delay τ and $\|u - x_1\| \leq R$. Then, fixing R and τ, for every $\varepsilon > 0$, we can find $N > 0$ such that if $\max(C_T^{\tau^+}, \sqrt{T}) \geq N$, the regret achieved by the algorithm for every instance in $\mathcal{R}(R, T, \tau, C_T^{\tau^+})$ is smaller than $\varepsilon \max(C_T^{\tau^+}, \sqrt{T})$. The proof then consists in finding two instances of $\mathcal{R}(R, T, \tau, C_T^{\tau^+})$ such that the regret achieved by the algorithm on these two instances can not be simultaneously smaller than $\varepsilon \max(C_T^{\tau^+}, \sqrt{T})$.

For this, we fix the delay τ, set $R = 1$ without loss of generality and explicit these two instances in the following ($\mathcal{X} = \mathbb{R}$):

1. Let $K, \ell > \tau$ be two positive integers. We first consider a loss sequence of length $2K\ell + \tau + 1$ (i.e., $T = 2K\ell + \tau + 1$) as illustrated below:

$$\underbrace{\underbrace{-1 \ldots -1}_{\ell}\ \underbrace{+1 \ldots +1}_{\ell}\quad \ldots \quad \underbrace{-1 \ldots -1}_{\ell}\ \underbrace{+1 \ldots +1}_{\ell}}_{2K\ell \text{ losses}}\ \underbrace{-1 \ldots -1}_{\tau+1}$$

A period is defined as a subsequence of 2ℓ losses with ℓ consecutive -1s followed by ℓ consecutive $+1$s. The whole loss sequence is then composed of $2K$ periods followed by $\tau+1$ consecutive -1s. We would like to compute the regret achieved by (DOptDA) with $\eta, \gamma, \widetilde{g}_{t+\frac{1}{2}}$ as specified in the statement and $x_1 = u = 0$.

For the first $\tau + 1$ steps, the algorithm stays at $x_1 = u$ so the accumulative regret is 0. For the remaining of the round, the algorithm goes through the same trajectory for each period of delayed feedback vectors it receives and this happens K times. To compute the regret, we just need to match the position of the iterate with the actual loss at each moment, which is done in Fig. 4 (as the loss vectors of a single period sum to 0, after receiving all the vectors from one period it is as if we started again from $x_1 = u = 0$). Notice that the algorithm uses the most recent vector it receives for extrapolation.

The regret for each period of feedback is thus

$$\begin{aligned}
\mathbf{Reg}_{per} &= \frac{-(\ell-\tau-1)(\ell-\tau)\eta}{2} - (\ell-\tau-1)\gamma + \frac{(\tau+1)(2\ell-\tau)\eta}{2} + (\tau+1)\gamma \\
&\quad + \frac{(\ell-\tau-1)(\ell+\tau)\eta}{2} - (\ell-\tau-1)\gamma - \frac{(\tau+1)\tau\eta}{2} + (\tau+1)\gamma \\
&= (\tau+1)(\ell-\tau)\eta + (\ell-\tau-1)\tau\eta + 2(2\tau-\ell+2)\gamma \\
&= (\eta + 2\tau\eta - 2\gamma)\ell - 2\tau(\tau+1)\eta + (4\tau+4)\gamma.
\end{aligned}$$

t	$\tau+2$	$\dots$	ℓ	$\ell+1$	$\cdots$	$\ell+\tau+1$	$\ell+\tau+2$	$\dots$	2ℓ	$2\ell+1$	$\dots$	$2\ell+\tau+1$
x_t	η	$\dots$	$(\ell-\tau-1)\eta$	$(\ell-\tau)\eta$	$\cdots$	$\ell\eta$	$(\ell-1)\eta$	$\dots$	$(\tau+1)\eta$	$\tau\eta$	$\dots$	0
g_t	-1			$+\gamma$			$+1$			$-\gamma$	-1	

Figure 4: Illustration of the evolution of the optimistic algorithm for a period of feedback in the first example of the proof of Theorem 14. The time is taken modulo 2ℓ.

Accordingly, the total regret is

$$\mathbf{Reg}_1 = K((\eta + 2\tau\eta - 2\gamma)\ell - 2\tau(\tau+1)\eta + (4\tau+4)\gamma) \geq K(\ell - 2\tau(\tau+1))\eta,$$

where for the inequality we use the fact that $\gamma \leq \tau\eta$.

Moreover, for every $m \in \mathbb{N}_0$, from time $2m\ell + \tau + 2$ to $2m\ell + 2\ell + \tau + 1$ the $(\tau+1)$-variation increases by $8(\tau+1)$: there are $\tau+1$ switches both from -1 to $+1$ and from $+1$ to -1 with each switch contributing 4 to the variation. Remember also that in the definition of the $C_T^{\tau^+}$ we compare the first $\tau+1$ losses with 0. For the whole sequence we therefore count $C_T^{\tau^+} = (8K+1)(\tau+1)$.

2. We now construct another example with the same $T, C_T^{\tau^+}$ as follows (with $\ell > 4\tau+4$):

$$\underbrace{\underbrace{0\dots0}_{\tau+1}\ \underbrace{1\dots1}_{\tau+1}\ \dots\ \underbrace{0\dots0}_{\tau+1}\ \underbrace{1\dots1}_{\tau+1}}_{8K(\tau+1)\text{ losses}}\ \underbrace{0\dots0}_{2K\ell-8K(\tau+1)}\ \underbrace{1\dots1}_{\tau+1}$$

In particular, $2K\ell - 8K(\tau+1) > 2K > \tau+1$. It follows immediately $C_T^{\tau^+} = (8K+1)(\tau+1)$ and of course $T = 2K\ell + \tau + 1$.

Let $x_1 = 0$ and $u = -1$. In the sequence the loss 1 appears $(4K+1)(\tau+1)$ times while the remaining feedback are all 0s. Given that the last $\tau+1$ losses are never received by the algorithm, we have indeed always $x_t \geq -4K(\tau+1)\eta - \gamma$. The regret can therefore be lower bounded as:

$$\begin{aligned}
\mathbf{Reg}_2 &= \sum_{t=1}^{T} g_t(x_t + 1) \\
&= \sum_{t=1}^{T} g_t x_t + (4K+1)(\tau+1) \\
&\geq (4K+1)(\tau+1) - 4K(4K+1)(\tau+1)^2\eta - (4K+1)(\tau+1)\gamma \\
&\geq (4K+1)(\tau+1) - (4K+1)^2(\tau+1)^2\eta,
\end{aligned}$$

where in the last inequality we use again $\gamma \leq \tau\eta$.

Conclude. We choose K, ℓ so that $\ell = (16K+9)(\tau+1)^2 + 2\tau(\tau+1) > 4\tau+4$. Notice that T and $C_T^{\tau^+}$ can be made arbitrarily large. We run the algorithm in question on the two problem instances described above. We have on one side

$$\mathbf{Reg}_1 \geq K(\ell - 2\tau(\tau+1))\eta = (16K^2 + 9K)(\tau+1)^2\eta.$$

On the other side,

$$\begin{aligned}\mathbf{Reg}_2 &\geq (4K+1)(\tau+1) - (4K+1)^2(\tau+1)^2\eta \\ &\geq (4K+1)(\tau+1) - (16K^2+9K)(\tau+1)^2\eta.\end{aligned}$$

Recalling that $C_T^{\tau^+} = (8K+1)(\tau+1)$, the above shows

$$\mathbf{Reg}_1 + \mathbf{Reg}_2 \geq (4K+1)(\tau+1) \geq C_T^{\tau^+}/2.$$

Similarly, we have $T = 2K\ell + \tau + 1 \leq (32K^2+22K)(\tau+1)^2$. As a consequence

$$\mathbf{Reg}_1 + \mathbf{Reg}_2 \geq (4K+1)(\tau+1) \geq \sqrt{T}/2.$$

To summarize, we have proven for some T and $C_T^{\tau^+}$ arbitrarily large, we can find two instances from $\mathcal{R}(R, T, \tau, C_T^{\tau^+})$ so that the regrets achieved by the algorithm on these two instances satisfy

$$\max(\mathbf{Reg}_1, \mathbf{Reg}_2) \geq \max(C_T^{\tau^+}, \sqrt{T})/2.$$

This is in contradiction with the initial hypothesis by choosing $\varepsilon = 1/2$. ∎

C.3 A lower bound for delayed online learning

Proposition 15. For any online learning algorithm with prior knowledge of T, τ and $\overline{C^\tau} \geq C_T^{\tau^+}$, there exists a sequence of linear losses such that if the feedback is subject to constant delay τ, then the regret of the algorithm on this sequence with respect to a vector u with $\|u - x_1\| \leq 1$ is $\Omega(\sqrt{\tau\overline{C^\tau}})$.

Proof. Let $\ell = \overline{C^\tau}/(4(\tau+1))$ be a positive integer and $T = (\tau+1)\ell$. We consider $\mathfrak{A}$ an arbitrary online algorithm compatible with delayed feedback. From $\mathfrak{A}$ we define $\mathfrak{A}_{/\tau}$ another online algorithm as follows: For any sequence of losses with undelayed feedback, we repeat each loss $\tau+1$ times and only send the feedback after a delay of τ. In other words, for the loss sequence $g_1, g_2, \ldots$, at the end of iteration $k(\tau+1)$ to $k(\tau+1)+\tau$ we receive feedback g_{k-1} (with the convention $g_0 = 0$) while we suffer a loss $\langle g_k, x_t\rangle$ from iteration $p_k = (k-1)(\tau+1)+1$ to $k(\tau+1)$. We then play $\mathfrak{A}$ on this new loss sequence with delayed feedback and after every $\tau+1$ iterations we return $\bar{x}_k = \sum_{t=p_k}^{p_k+\tau} x_t/(\tau+1)$. This is a legitimate online algorithm because the knowledge of g_k is not required for playing $\bar{x}_k$. Moreover, the regret achieved by $\mathfrak{A}$ on the constructed sequence is exactly $\tau+1$ times the regret achieved by $\mathfrak{A}_{/\tau}$ on the original sequence.

We now apply the the well known $\Omega(\sqrt{\ell})$ lower bound for a horizon of ℓ (see e.g., Shalev-Shwartz, 2007), and this proves the existence of a sequence of linear losses of length ℓ and a corresponding u with $\|u - x_1\| \leq 1$ such that the regret achieved by $\mathfrak{A}_{/\tau}$ is $\Omega(\sqrt{\ell})$. Moreover, the loss vectors are either 1 or -1. Let us now considered the loss sequence constructed as in the previous paragraph. The $(\tau+1)$-variation $C_T^{\tau^+}$ is then bounded by $(\tau+1)+4(\tau+1)(\ell-1) < \overline{C^\tau}$ and we have effectively $T = (\tau+1)\ell$. To finish, we observe that the regret achieved by $\mathfrak{A}$ on the constructed sequence is $\Omega((\tau+1)\sqrt{\ell})$ and $(\tau+1)\sqrt{\ell} \sim \sqrt{\tau\overline{C^\tau}}/2$ (where $\sim$ stands for asymptotically equivalent). ∎

C.4 Delayed online learning with slow variation

Theorem 16. Let the maximum delay be bounded by τ and that Assumption 3 holds. Assume in addition that the vector fields V_t are L-Lipschitz continuous. Take $\widetilde{g}_{t+\frac{1}{2}} = \widetilde{V}_t(x_t)$, $\eta_{t+1} \leq \eta_t$,

$(2\tau+1)\eta_t \le \gamma_t$, and $2\gamma_t^2 L^2 \le 1$. Then, the regret of delayed optimistic dual averaging (DOptDA) (evaluated at the points $x_{\frac{3}{2}}, \dots, x_{T+\frac{1}{2}}$) satisfies

$$\mathbf{Reg}_T(u) \le \frac{\|u - x_1\|^2}{2\eta_T} + \sum_{t=1}^{T} \gamma_t \|V_t(x_t) - \widetilde{V}_t(x_t)\|^2.$$

Proof. The proof is immediate from Theorem 12. Indeed,

$$\|V_t(x_{t+\frac{1}{2}}) - \widetilde{V}_t(x_t)\|^2 \le 2\|V_t(x_{t+\frac{1}{2}}) - V_t(x_t)\|^2 + 2\|V_t(x_t) - \widetilde{V}_t(x_t)\|^2.$$

Then, using the Lipschitz continuity of $\widetilde{V}_t$ and the condition $2\gamma_t^2 L^2 \le 1$, we have:

$$2\|V_t(x_{t+\frac{1}{2}}) - V_t(x_t)\|^2 \le 2L^2\|x_{t+\frac{1}{2}} - x_t\|^2 = 2{\gamma_t}^2 L^2 \|\widetilde{V}_t(x_t)\|^2 \le \|\widetilde{V}_t(x_t)\|^2.$$

In other words, we have proven $\|g_{t+\frac{1}{2}} - \widetilde{g}_{t+\frac{1}{2}}\|^2 - \|\widetilde{g}_{t+\frac{1}{2}}\|^2 \le 2\|V_t(x_t) - \widetilde{V}_t(x_t)\|^2$ and the bound follows. ∎

C.5 More flexible learning rates

In order prove Proposition 17, we generalize both Theorem 12 and Theorem 16 to the case where the learning rate is non-increasing along a faithful permutation.

Theorem 12′. Assume that the maximum delay is bounded by τ. Consider a faithful permutation σ and let delayed optimistic dual averaging (DOptDA) be run with learning rate sequences $(\eta_t)_{t\in[T]}$, $(\gamma_t)_{t\in[T]}$ satisfying $\eta_{\sigma(t+1)} \le \eta_{\sigma(t)}$ and $(4\tau+1)\max_{\{s:|s-t|\le\tau\}} \eta_s \le \gamma_t$ for all t. Then, the regret of the algorithm (evaluated at the points $x_{\frac{3}{2}}, \dots, x_{T+\frac{1}{2}}$) satisfies

$$\mathbf{Reg}_T(u) \le \frac{\|u - x_1\|^2}{2\eta_T} + \sum_{t=1}^{T} \frac{\gamma_t}{2} \left(\|g_{t+\frac{1}{2}} - \widetilde{g}_{t+\frac{1}{2}}\|^2 - \|\widetilde{g}_{t+\frac{1}{2}}\|^2 \right).$$

Proof. We define the virtual iterates

$$\widetilde{x}_t = x_1 - \eta_{\sigma(t)} \sum_{s=1}^{t-1} g_{\sigma(s)+\frac{1}{2}}.$$

We then decompose

$$f_t(x_{t+\frac{1}{2}}) - f_t(u) \le \langle g_{t+\frac{1}{2}}, x_{t+\frac{1}{2}} - u\rangle = \langle g_{t+\frac{1}{2}}, x_{\sigma(t)+\frac{1}{2}} - \widetilde{x}_{t+1}\rangle + \langle g_{t+\frac{1}{2}}, \widetilde{x}_{t+1} - u\rangle.$$

Following closely the proof of Theorem 12, we obtain

$$\begin{aligned}\mathbf{Reg}_T(u) \le \frac{\|u - x_1\|^2}{2\eta_{\sigma(T)}} &+ \sum_{t=1}^{T} \frac{\gamma_t}{2} \left(\|g_{t+\frac{1}{2}} - \widetilde{g}_{t+\frac{1}{2}}\|^2 - \|\widetilde{g}_{t+\frac{1}{2}}\|^2 \right) \\ &+ \left(-\frac{\gamma_{\sigma(t)}}{2} + \frac{(\mathrm{card}(\mathcal{U}_t^\sigma)+1)\eta_{\sigma(t)}}{2} + \sum_{\sigma(t)\in\mathcal{U}_t^\sigma} \frac{\eta_{\sigma(l)}}{2} \right) \|g_{\sigma(t)+\frac{1}{2}}\|^2.\end{aligned}$$

Invoking Proposition 10, we know that $[t]^\sigma \setminus \mathcal{S}_{\sigma(t)} \subseteq \{\sigma(t)-\tau, \dots, \sigma(t)+\tau\}$. Given that $\sigma(t) \notin [t-1]^\sigma$, this implies $\mathcal{U}_t^\sigma \subseteq \{\sigma(t)-\tau, \dots, \sigma(t)-1\} \cup \{\sigma(t)+1, \dots, \sigma(t)+\tau\}$. Therefore,

$\mathrm{card}(\mathcal{U}_t^\sigma) \leq 2\tau$ and if $\sigma(t) \in \mathcal{U}_l^\sigma$ then $|\sigma(t) - \sigma(l)| \leq \tau$ while $\sigma(t) \neq \sigma(l)$, which also shows $\mathrm{card}(\{l : \sigma(t) \in \mathcal{U}_l^\sigma\}) \leq 2\tau$. Accordingly,

$$\frac{(\mathrm{card}(\mathcal{U}_t^\sigma)+1)\eta_{\sigma(t)}}{2} + \sum_{\sigma(t)\in\mathcal{U}_l^\sigma} \frac{\eta_{\sigma(l)}}{2} \leq \frac{(4\tau+1)\max_{\{s:|s-\sigma(t)|\leq\tau\}}\eta_s}{2}.$$

With the assumption $\gamma_t \geq (4\tau+1)\max_{\{s:|s-t|\leq\tau\}}\eta_s$, we effectively deduce

$$\mathbf{Reg}_T(u) \leq \frac{\|u-x_1\|^2}{2\eta_{\sigma(T)}} + \sum_{t=1}^{T} \frac{\gamma_t}{2}\left(\|g_{t+\frac{1}{2}} - \widetilde{g}_{t+\frac{1}{2}}\|^2 - \|\widetilde{g}_{t+\frac{1}{2}}\|^2\right).$$

This proves the theorem. ∎

Theorem 16′. Let the maximum delay be bounded by τ and that Assumption 3 holds. Assume in addition that the vector fields V_t are L-Lipschitz continuous. Consider a faithful permutation σ and take $\widetilde{g}_t = \widetilde{V}_t(x_t)$, $\eta_{\sigma(t+1)} \leq \eta_{\sigma(t)}$, $(4\tau+1)\max_{\{s:|s-t|\leq\tau\}}\eta_s \leq \gamma_t$, and $2\gamma_t^2 L^2 \leq 1$. Then, the regret of delayed optimistic dual averaging (DOptDA) (evaluated at the points $x_{\frac{3}{2}}, \dots, x_{T+\frac{1}{2}}$) satisfies

$$\mathbf{Reg}_T(u) \leq \frac{\|u-x_1\|^2}{2\eta_{\sigma(T)}} + \sum_{t=1}^{T} \gamma_t \|V_t(x_t) - \widetilde{V}_t(x_t)\|^2.$$

Proof. Apply Theorem 12′ and bound the term $\|g_{t+\frac{1}{2}} - \widetilde{g}_{t+\frac{1}{2}}\|^2 - \|\widetilde{g}_{t+\frac{1}{2}}\|^2$ as in the proof of Theorem 16. ∎

Proposition 17. Let the maximum delay be bounded by τ and let Assumptions 2 and 3 hold. Further suppose that V_t are L-Lipschitz continuous and both $V_t, \widetilde{V}_t$ have their norm bounded by G. Then for any u such that $\|u - x_1\| \leq R$, running delayed optimistic dual averaging (DOptDA) with $\widetilde{g}_t = \widetilde{V}_t(x_t)$,

$$\gamma_t = \min\left(\frac{R\sqrt{4\tau+1}}{2\sqrt{\left(\sum_{s\in\mathcal{S}_t}\|V_s(x_s) - \widetilde{V}_s(x_s)\|^2 + 4G^2(\tau+1)\right)}}, \frac{1}{\sqrt{2}L}\right),$$

and

$$\eta_t = \min\left(\frac{R}{2\sqrt{(4\tau+1)\left(\sum_{s\in\mathcal{S}_t}\|V_s(x_s) - \widetilde{V}_s(x_s)\|^2 + 4G^2(3\tau+1)\right)}}, \frac{1}{\sqrt{2}L(4\tau+1)}\right)$$

guarantees

$$\mathbf{Reg}_T(u) \leq \max\left(\sqrt{2}R^2L(4\tau+1), 2R\sqrt{(4\tau+1)(C_T + 4G^2(3\tau+1))}\right).$$

Proof. Let $\widetilde{C}_t = \sum_{s\in\mathcal{S}_t}\|V_s(x_s) - \widetilde{V}_s(x_s)\|^2$. We consider a permutation σ such that (i) if $\widetilde{C}_s < \widetilde{C}_t$ then $\sigma^{-1}(s) < \sigma^{-1}(t)$; (ii) if $\widetilde{C}_s = \widetilde{C}_t$ and $s \in \mathcal{S}_t$ then $\sigma^{-1}(s) < \sigma^{-1}(t)$. The sequence $(\widetilde{C}_t)_t$ is non-decreasing along σ (see e.g., proof of Proposition 7) and accordingly the learning rate sequence

$(\eta_t)_t$ is non-decreasing along σ. Moreover, if $s \in \mathcal{S}_t$, we have $\mathcal{S}_s \subseteq \mathcal{F}_{i(t),s} \subseteq \mathcal{S}_t$ thanks to Assumption 2. This implies $\widetilde{C}_s \leq \widetilde{C}_t$; subsequently $\sigma^{-1}(s) < \sigma^{-1}(t)$. The above shows that σ is a faithful permutation. The condition $2\gamma_t^2 L^2 \leq 1$ is automatically satisfied by the definition of γ_t. To apply Theorem 16′, the last missing piece is to prove $(4\tau+1)\max_{\{s:|s-t|\leq\tau\}} \eta_s \leq \gamma_t$. This boils down to showing that

$$\widetilde{C}_s + 4G^2(3\tau+1) \geq \widetilde{C}_t + 4G^2(\tau+1) \tag{24}$$

for all $s \in [T] \cap \{t-\tau, \dots, t+\tau\}$. The maximum delay being bounded by τ, we have $|\operatorname{card}(\mathcal{S}_s) - \operatorname{card}(\mathcal{S}_t)| \leq |s-t| + \tau$. By bounding each $\|V_l(x_l) - \widetilde{V}_l(x_l)\|^2$ by $4G^2$, we indeed prove (24) for s such that $|s-t| \leq \tau$.

With all this at hand, applying Theorem 16′ gives

$$\mathbf{Reg}_T(u) \leq \frac{\|u - x_1\|^2}{2\eta_{\sigma(T)}} + \sum_{t=1}^{T} \gamma_t \|V_t(x_t) - \widetilde{V}_t(x_t)\|^2.$$

As the maximum delay is bounded by τ and the gradients are bounded by G, we have $\widetilde{C}_t + 4G^2(\tau+1) \geq C_t$. Invoking Lemma 6 then gives

$$\begin{aligned} \mathbf{Reg}_T(u) &\leq \frac{\|u - x_1\|^2}{2\eta_{\sigma(T)}} + \frac{R\sqrt{4\tau+1}}{2} \sum_{t=1}^{T} \frac{1}{\sqrt{C_t}} \|V_t(x_t) - \widetilde{V}_t(x_t)\|^2 \\ &\leq \frac{R^2}{2\eta_{\sigma(T)}} + R\sqrt{(4\tau+1)C_T}. \end{aligned} \tag{25}$$

We bound the second term by

$$R\sqrt{(4\tau+1)C_T} \leq R\sqrt{(4\tau+1)(\widetilde{C}_T + 4G^2(3\tau+1))} \leq \frac{R^2}{2\eta_T} \leq \frac{R^2}{2\eta_{\sigma(T)}}. \tag{26}$$

Combining (25) and (26) we get $\mathbf{Reg}_T(u) \leq R^2/\eta_{\sigma(T)}$. We can conlcude by using the definition of $\eta_{\sigma(T)}$ and $\widetilde{C}_{\sigma(T)} \leq C_T$. ∎

Appendix D. Regret bounds for decentralized delayed dual averaging

The proofs in this part leverage on Lemma 18, which we recall below.

Lemma 18. Assume that all the loss functions $f_{i,t}$ are G-Lipschitz; then,

$$\mathbf{Reg}_T^g(u) \leq \mathbf{Reg}_T^\ell(u) + \sum_{t=1}^{T} \sum_{i=1}^{M_t} G\|x_{i,t} - x_{1,t}\|.$$

These proofs can thus be divided into two essential parts: a bound on the effective regret and a bound on the inter-agent discrepancies. For the first part we will utilize the change of index $\phi(i,t) = N_{t-1} + i$ introduced in Section 6.3.2, where $N_t = \sum_{s=1}^{t} M_s$ and $N = N_T$. We also recall the notations $g'_{\phi(i,t)} = g_{i,t}$ and $\mathcal{S}'_{\phi(i,t)} = \{\phi(j,s) : (j,s) \in \mathcal{S}_{i,t}\}$.

D.1 Fixed learning rate

Proposition 19. Assume that the maximum delay is bounded by τ and that all the loss functions are G-Lipschitz. For any u satisfying $h(u) \leq R^2$, running decentralized delayed dual averaging (D-DDA) with constant stepsize

$$\eta_{i,t} \equiv \eta = \frac{R}{G\overline{M}\sqrt{(2\tau+1)T}}$$

guarantees the following upper bound on the collective regret

$$\mathbf{Reg}_T^g(u) \leq 2RG\overline{M}\sqrt{(2\tau+1)T} = \mathcal{O}(\overline{M}\sqrt{\tau T}).$$

Proof. Let us start with (13). Since the loss functions are G-Lipschitz, the subgradients are bounded by G.

$$\begin{aligned}\mathbf{Reg}_T^\ell(u) &\leq \frac{h(u)}{\eta} + \frac{1}{2}\sum_{m=1}^{N}\eta\left(\|g'_m\|_*^2 + 2\|g'_m\|_*\sum_{l\in[m-1]\setminus\mathcal{S}'_m}\|g'_l\|_*\right)\\ &\leq \frac{h(u)}{\eta} + \frac{\eta}{2}\sum_{m=1}^{N}(1+2\operatorname{card}([m-1]\setminus\mathcal{S}'_m))G^2. \end{aligned} \tag{27}$$

To bound $\operatorname{card}([m-1]\setminus\mathcal{S}'_m)$, we write $m = \phi(i,t)$. On one hand, the subgradients

$$\{g_{i-1,t},\dots,g_{1,t}\} = \{g'_{m-1},\dots,g'_{m-i+1}\}$$

of instant t are necessarily unavailable when making the prediction $x_{i,t} = x'_m$. On the other hand, the maximum delay assumption guarantees that all the subgradients received before time $t-\tau$ are used in the computation of $x_{i,t}$. This leads to the inequality

$$\operatorname{card}([m-1]\setminus\mathcal{S}'_m) \leq i - 1 + \sum_{s=1}^{\tau} M_{t-s},$$

with the convention $M_l = 0$ if $l \leq 0$. Subsequently, for any $t \in [T]$,

$$\sum_{m=N_{t-1}+1}^{N_t}\operatorname{card}([m-1]\setminus\mathcal{S}'_m) \leq \frac{M_t(M_t-1)}{2} + M_t\sum_{s=1}^{\tau}M_{t-s} \leq \frac{(\tau+1)}{2}M_t^2 + \frac{1}{2}\sum_{s=1}^{\tau}M_{t-s}^2. \tag{28}$$

Substituting (28) in (27) then yields

$$\mathbf{Reg}_T^\ell(u) \leq \frac{h(u)}{\eta} + \eta(\tau+1)G^2\sum_{t=1}^{T}M_t^2. \tag{29}$$

We proceed to bound the difference $\|x_{i,t} - x_{j,t}\|$ for all $t \in [T]$ and $i, j \in [M_t]$. In fact, we have $x_{i,t} = Q(-y_{i,t})$ and $x_{j,t} = Q(-y_{j,t})$ where $y_{i,t} = \eta\sum_{(k,s)\in\mathcal{S}_{i,t}} g_{k,s}$ and $y_{j,t} = \eta\sum_{(k,s)\in\mathcal{S}_{j,t}} g_{k,s}$. From the maximum delay assumption we know that $\mathcal{S}_{i,t}$ and $\mathcal{S}_{j,t}$ differ by at most $\sum_{s=1}^{\tau}M_{t-s}$ samples. Using the G-Lipshitz continuity of the loss functions and the non-expansiveness of the mirror map (Lemma 21), we obtain

$$\sum_{i=1}^{M_t}G\|x_{i,t} - x_{j,t}\| \leq \eta G^2 M_t\sum_{s=1}^{\tau}M_{t-s} \leq \eta G^2\left(\frac{\tau M_t^2}{2} + \frac{1}{2}\sum_{s=1}^{\tau}M_{t-s}^2\right). \tag{30}$$

With (29) and (30), invoking Lemma 18 gives

$$\mathbf{Reg}_T^g(u) \leq \frac{h(u)}{\eta} + \eta(2\tau+1)G^2 \sum_{t=1}^{T} M_t^2.$$

The theorem follows immediately. ∎

D.2 Learning rates based on the number of received feedbacks

As discussed in Section 6.3.2, the learning rate proposed in Proposition 19 is generally not implementable in practice. We will show below that a learning rate scheme similar to the one considered in Section 4.1 equally guarantees low collective regret. To begin with, we rewrite Assumption 1 to accommodate the new notation.

Assumption 1′. *If* $(j,s) \in \mathcal{S}_{i,t}$ *then* $\mathrm{card}(\mathcal{S}_{j,s}) < \mathrm{card}(\mathcal{S}_{i,t})$.

Under this assumption, we prove the following theorem which further extends the result of Proposition 7.

Proposition 23. Let Assumption 1′ hold. Suppose that the maximum delay is bounded by τ and that all the loss functions are G-Lipschitz. Then, for any u satisfying $h(u) \leq R^2$, decentralized delayed dual averaging (D-DDA) with stepsizes

$$\eta_{i,t} = \frac{R}{G\sqrt{(5\tau+3)(\mathrm{card}(\mathcal{S}_{i,t}) + (\tau+1)M_{\max})M_{\max}}} \tag{31}$$

guarantees a collective regret in

$$\mathbf{Reg}_T^g(u) = \mathcal{O}(\sqrt{\tau N M_{\max}}).$$

Proof. With a slight abuse of notation, we will only work with the (worker, time) index pair in this proof, but it should be understood that the change of index ϕ indeed intervenes implicitly when we apply the arguments of the previous sections (notably when we compare the indices). Compared to Proposition 19, the two additional difficulties here are: *i*) the non-monotonicity of learning rates which are solved by the introduction of a suitable faithful permutation; *ii*) the predictions of a time instant are not generated by the same learning rate, but we still manage to control the deviation since these learning rates are close enough.

To begin, we consider a permutation σ satisfying $\sigma^{-1}(j,s) < \sigma^{-1}(i,t)$ if $\mathrm{card}(\mathcal{S}_{j,s}) < \mathrm{card}(\mathcal{S}_{i,t})$. Such a σ is necessarily faithful according to Assumption 1′. We claim that $\mathrm{card}(\mathcal{U}^{\sigma}_{\sigma^{-1}(i,t)}) \leq (\tau+1)M_{\max}$ (where $\mathcal{U}^{\sigma}_{\sigma^{-1}(i,t)} = [\sigma^{-1}(i,t)-1]^{\sigma} \setminus \mathcal{S}_{i,t}$). Let $s \in \{0,\dots,\tau\}$ such that $N_{t+s-\tau} > \mathrm{card}(\mathcal{S}_{i,t}) \geq N_{t+s-\tau-1}$. Then for any $j \in [M_{t+s+1}]$ it holds $\mathrm{card}(\mathcal{S}_{j,t+s+1}) \geq N_{t+s-\tau} > \mathrm{card}(\mathcal{S}_{i,t})$ and accordingly $\sigma^{-1}(i,t) < \sigma^{-1}(j,t+s+1)$. In other words, if $\sigma^{-1}(k,l) < \sigma^{-1}(i,t)$ for some $l \in [T]$ and $k \in [M_l]$ then $l \leq t+s$, and subsequently $\mathrm{card}([\sigma^{-1}(i,t)-1]^{\sigma}) \leq N_{t+s}$. We have therefore

$$\mathrm{card}([\sigma^{-1}(i,t)-1]^{\sigma} \setminus \mathcal{S}_{i,t}) \leq N_{t+s} - N_{t+s-\tau-1} = \sum_{l=0}^{\tau} M_{t+s-l} \leq (\tau+1)M_{\max}.$$

Since $\eta_{i,t} \leq \eta_{j,s}$ if and only if $\operatorname{card}(\mathcal{S}_{i,t}) \geq \operatorname{card}(\mathcal{S}_{j,s})$, we have indeed $\eta_{\sigma((i,t)+1)} \leq \eta_{\sigma(i,t)}$. Invoking Theorem 2, one has (notice that the sum is ordered differently as stated in the theorem)

$$\begin{aligned}\mathbf{Reg}_T^{\ell}(u) &\leq \frac{h(u)}{\eta_{\sigma(M_T,T)}} + \frac{1}{2}\sum_{t=1}^{T}\sum_{i=1}^{M_t}\eta_{i,t}\left(\|g_{i,t}\|_*^2 + 2\|g_{i,t}\|_* \sum_{s\in\mathcal{U}^{\sigma}_{\sigma^{-1}(i,t)}} \|g_s\|_*\right)\\ &\leq \frac{h(u)}{\min_{t\in[T],i\in[M_t]}\eta_{i,t}} + \frac{1}{2}\sum_{t=1}^{T}\left(\max_{i\in[M_t]}\eta_{i,t}\right) G^2(2\tau+3)M_t M_{\max}. \end{aligned} \tag{32}$$

In the second step we bound the difference $\|x_{i,t} - x_{j,t}\|$ for $i, j \in [M_t]$. Similar to the proof of Proposition 19, we write $x_{i,t} = Q(-y_{i,t})$ and $x_{j,t} = Q(-y_{j,t})$ where $y_{i,t} = \eta_{i,t}\sum_{(k,s)\in\mathcal{S}_{i,t}} g_{k,s}$ and $y_{j,t} = \eta_{j,t}\sum_{(k,s)\in\mathcal{S}_{j,t}} g_{k,s}$. By the non-expansiveness of the mirror map (Lemma 21) it is then sufficient to bound $\|y_{i,t} - y_{j,t}\|$. For ease of notation, in the rest of the proof we will denote by $\mathcal{S}_\cap$ the intersection of $\mathcal{S}_{i,t}$ and $\mathcal{S}_{j,t}$, i.e., $\mathcal{S}_\cap = \mathcal{S}_{i,t}\cap\mathcal{S}_{j,t}$. It follows

$$\begin{aligned}\|y_{i,t} - y_{j,t}\| &= \|(\eta_{i,t}-\eta_{j,t})\sum_{(k,s)\in\mathcal{S}_\cap} g_{k,s} + \eta_{i,t}\sum_{(k,s)\in\mathcal{S}_{i,t}\setminus\mathcal{S}_\cap} g_{k,s} - \eta_{j,t}\sum_{(k,s)\in\mathcal{S}_{j,t}\setminus\mathcal{S}_\cap} g_{k,s}\|\\ &\leq |\eta_{i,t}-\eta_{j,t}|\sum_{(k,s)\in\mathcal{S}_\cap}\|g_{k,s}\| + \eta_{i,t}\sum_{(k,s)\in\mathcal{S}_{i,t}\setminus\mathcal{S}_\cap}\|g_{k,s}\| + \eta_{j,t}\sum_{(k,s)\in\mathcal{S}_{j,t}\setminus\mathcal{S}_\cap}\|g_{k,s}\|\\ &\leq G(|\eta_{i,t}-\eta_{j,t}|\operatorname{card}(\mathcal{S}_\cap) + \max(\eta_{i,t},\eta_{j,t})\operatorname{card}(\mathcal{S}_{i,t}\triangle\mathcal{S}_{j,t}))\\ &\leq G(|\eta_{i,t}-\eta_{j,t}|N_{t-1} + \max(\eta_{i,t},\eta_{j,t})\tau M_{\max}). \end{aligned} \tag{33}$$

In the last inequality we use the fact that if one element belongs to one set but not the other then it must come from the last τ time steps.

To control $|\eta_{i,t} - \eta_{j,t}|$, we note that for any $b > a > 0$, it holds

$$\frac{1}{\sqrt{a}} - \frac{1}{\sqrt{b}} = \frac{b-a}{\sqrt{ab}(\sqrt{a}+\sqrt{b})} \leq \frac{b-a}{2a\sqrt{a}}.$$

For every $k \in [M_t]$, we have $\operatorname{card}(\mathcal{S}_{k,t}) + (\tau+1)M_{\max} \geq N_t > N_{t-1}$. Therefore, with the stepsize rule (31), we get

$$|\eta_{i,t} - \eta_{j,t}| \leq \frac{R\,|\operatorname{card}(\mathcal{S}_{i,t}) - \operatorname{card}(\mathcal{S}_{j,t})|}{2GN_{t-1}\sqrt{(5\tau+3)N_t M_{\max}}} \leq \frac{R\tau M_{\max}}{2GN_{t-1}\sqrt{(5\tau+3)N_t M_{\max}}}. \tag{34}$$

Let us denote $\eta_t = R/(G\sqrt{(5\tau+3)N_t M_{\max}})$; then $\eta_{i,t} \leq \eta_t$ for all $i \in [M_t]$. We also take

$$\underline{\eta} = \frac{R}{G\sqrt{(5\tau+3)(NM_{\max} + (\tau+1)M_{\max}^2)}}$$

so that $\eta_{i,t} \geq \underline{\eta}$ for all $t \in [T], i \in [M_t]$. We conclude with the help of Lemmas 6, 18 and 21, and the inequalities (32), (33) and (34):

$$\begin{aligned}\mathbf{Reg}_T^{g}(u) &\leq \frac{h(u)}{\underline{\eta}} + \frac{1}{2}\sum_{t=1}^{T}\left(\eta_t G^2(4\tau+3)M_t M_{\max} + \frac{RG\tau M_t M_{\max}}{\sqrt{(5\tau+3)N_t M_{\max}}}\right)\\ &= \frac{h(u)}{\underline{\eta}} + \frac{1}{2}\sum_{t=1}^{T}\frac{RG(5\tau+3)M_t M_{\max}}{\sqrt{(5\tau+3)N_t M_{\max}}}\\ &\leq RG\sqrt{(5\tau+3)(NM_{\max}+(\tau+1)M_{\max}^2)} + RG\sqrt{(5\tau+3)NM_{\max}}. \end{aligned}$$

Accordingly, $\mathbf{Reg}_T^g(u) = \mathcal{O}(\sqrt{\tau N M_{\max}})$. ∎

Note that the bound of Proposition 23 directly features the total number of actions taken in the full process; it is thus (at least partly) adaptive to the number of agents. More importantly, since $\mathrm{card}(\mathcal{S}_{i,t})$ is obviously available to each agent at time t, the learning rate (31) is indeed implementable by every single agent as long as the constants G, τ, and $M_{\max}$ are known. We leave the design of fully adaptive methods in the sense of (AdaDelay–Dist) as an open question.

References

Alekh Agarwal, O. Dekel, and L. Xiao. Optimal algorithms for online convex optimization with multi-point bandit feedback. In *COLT '10: Proceedings of the 23rd Annual Conference on Learning Theory*, 2010.

Mohammad Akbari, Bahman Gharesifard, and Tamás Linder. Distributed online convex optimization on time-varying directed graphs. *IEEE Transactions on Control of Network Systems*, 4(3):417–428, 2015.

Eitan Altman, Thomas Boulogne, Rachid el Azouzi, Tania Jiménez, and Laura Wynter. A survey on networking games in telecommunications. *Computers and Operations Research*, 33(2):286–311, 2006.

Peter Auer, Nicolò Cesa-Bianchi, and Claudio Gentile. Adaptive and self-confident on-line learning algorithms. *Journal of Computer and System Sciences*, 64(1):48–75, 2002a.

Peter Auer, Nicolo Cesa-Bianchi, and Claudio Gentile. Adaptive and self-confident on-line learning algorithms. *Journal of Computer and System Sciences*, 64(1):48–75, 2002b.

Arda Aytekin, Hamid Reza Feyzmahdavian, and Mikael Johansson. Analysis and implementation of an asynchronous optimization algorithm for the parameter server. *arXiv preprint arXiv:1610.05507*, 2016.

Yogev Bar-On and Yishay Mansour. Individual regret in cooperative nonstochastic multi-armed bandits. In *Advances in Neural Information Processing Systems*, pages 3116–3126, 2019.

Pascal Bianchi, Walid Hachem, and Franck Iutzeler. A coordinate descent primal-dual algorithm and application to distributed asynchronous optimization. *IEEE Transactions on Automatic Control*, 61(10):2947–2957, 2015.

Sébastien Bubeck and Nicolò Cesa-Bianchi. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends in Machine Learning*, 5(1):1–122, 2012.

Xuanyu Cao, Junshan Zhang, and H Vincent Poor. Constrained online convex optimization with feedback delays. *IEEE Transactions on Automatic Control*, 2020.

Nicolò Cesa-Bianchi and Gábor Lugosi. *Prediction, Learning, and Games*. Cambridge University Press, 2006.

Nicolò Cesa-Bianchi, Claudio Gentile, and Yishay Mansour. Nonstochastic bandits with composite anonymous feedback. In *Conference On Learning Theory*, pages 750–773, 2018.

Nicolò Cesa-Bianchi, Claudio Gentile, and Yishay Mansour. Delay and cooperation in nonstochastic bandits. *The Journal of Machine Learning Research*, 20(1):613–650, 2019.

Nicolò Cesa-Bianchi, Tommaso Cesari, and Claire Monteleoni. Cooperative online learning: Keeping your neighbors updated. In *Algorithmic Learning Theory*, pages 234–250. PMLR, 2020.

Olivier Chapelle. Modeling delayed feedback in display advertising. In *KDD '14: Proceedings of the 20th International ACM Conference on Knowledge Discovery and Data Mining*, 2014.

Chao-Kai Chiang, Tianbao Yang, Chia-Jung Lee, Mehrdad Mahdavi, Chi-Jen Lu, Rong Jin, and Shenghuo Zhu. Online optimization with gradual variations. In *COLT '12: Proceedings of the 25th Annual Conference on Learning Theory*, 2012.

Patrick L Combettes and Jean-Christophe Pesquet. Stochastic quasi-fejér block-coordinate fixed point iterations with random sweeping. *SIAM Journal on Optimization*, 25(2):1221–1248, 2015.

Lorenzo Croissant, Marc Abeille, and Clément Calauzènes. Real-time optimisation for online learning in auctions. In *ICML '20: Proceedings of the 37th International Conference on Machine Learning*, pages 2217–2226. PMLR, 2020.

Charles Monnoyer de Galland, Samuel Martin, and Julien M Hendrickx. Open multi-agent systems with variable size: the case of gossiping. *arXiv preprint arXiv:2009.02970*, 2020.

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *The Journal of Machine Learning Research*, 12:2121–2159, 2011.

Genevieve Flaspohler, Francesco Orabona, Judah Cohen, Soukayna Mouatadid, Miruna Oprescu, Paulo Orenstein, and Lester Mackey. Online learning with optimism and delay. In *International Conference on Machine Learning*, 2021.

Abraham D. Flaxman, Adam Tauman Kalai, and H. Brendan McMahan. Online convex optimization in the bandit setting: gradient descent without a gradient. In *SODA '05: Proceedings of the 16th annual ACM-SIAM Symposium on Discrete Algorithms*, pages 385–394, 2005.

Mauro Franceschelli and Paolo Frasca. Stability of open multi-agent systems and applications to dynamic consensus. *IEEE Transactions on Automatic Control*, 2020.

Elad Hazan. Introduction to online convex optimization. *Foundations and Trends® in Optimization*, 2(3-4):157–325, 2016.

Amélie Héliou, Panayotis Mertikopoulos, and Zhengyuan Zhou. Gradient-free online learning in continuous games with delayed rewards. In *ICML '20: Proceedings of the 37th International Conference on Machine Learning*, 2020.

Julien M Hendrickx and Samuel Martin. Open multi-agent systems: Gossiping with random arrivals and departures. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pages 763–768. IEEE, 2017.

Julien M Hendrickx and Michael G Rabbat. Stability of decentralized gradient descent in open multi-agent systems. *arXiv preprint arXiv:2009.05445*, 2020.

Jean-Baptiste Hiriart-Urruty and Claude Lemaréchal. *Fundamentals of convex analysis*. Springer, 2001.

Saghar Hosseini, Airlie Chapman, and Mehran Mesbahi. Online distributed optimization via dual averaging. In *52nd IEEE Conference on Decision and Control*, pages 1484–1489. IEEE, 2013.

Yu-Guan Hsieh, Franck Iutzeler, Jérôme Malick, and Panayotis Mertikopoulos. Explore aggressively, update conservatively: Stochastic extragradient methods with variable stepsize scaling. In *Neural Information Processing Systems*, 2020.

Pooria Joulani, András György, and Csaba Szepesvári. Online learning under delayed feedback. In *ICML '13: Proceedings of the 30th International Conference on Machine Learning*, 2013.

Pooria Joulani, András György, and Csaba Szepesvári. Delay-tolerant online convex optimization: Unified analysis and adaptive-gradient algorithms. In *AAAI '16: Proceedings of the 30th Conference on Artificial Intelligence*, 2016.

Pooria Joulani, András György, and Csaba Szepesvári. A modular analysis of adaptive (non-) convex optimization: Optimism, composite objectives, and variational bounds. In *International Conference on Algorithmic Learning Theory*, pages 681–720, 2017.

Pooria Joulani, András György, and Csaba Szepesvári. Think out of the "box": Generically-constrained asynchronous composite optimization and hedging. In *Advances in Neural Information Processing Systems*, 2019.

Anatoli Juditsky, Joon Kwon, and Éric Moulines. Unifying mirror descent and dual averaging. *arXiv preprint arXiv:1910.13742*, 2019.

G. M. Korpelevich. The extragradient method for finding saddle points and other problems. *Èkonom. i Mat. Metody*, 12: 747–756, 1976.

Joon Kwon and Panayotis Mertikopoulos. A continuous-time approach to online optimization. *Journal of Dynamics and Games*, 4(2):125–148, April 2017.

John Langford, Alexander J Smola, and Martin Zinkevich. Slow learners are fast. In *Proceedings of the 22nd International Conference on Neural Information Processing Systems*, pages 2331–2339, 2009.

Rémi Leblond, Fabian Pedregosa, and Simon Lacoste-Julien. Asaga: asynchronous parallel saga. In *Artificial Intelligence and Statistics*, pages 46–54. PMLR, 2017.

Bingcong Li, Tianyi Chen, and Georgios B Giannakis. Bandit online learning with unknown delays. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 993–1002. PMLR, 2019.

Horia Mania, Xinghao Pan, Dimitris Papailiopoulos, Benjamin Recht, Kannan Ramchandran, and Michael I Jordan. Perturbed iterate analysis for asynchronous stochastic optimization. *SIAM Journal on Optimization*, 27(4):2202–2229, 2017.

David Mateos-Núnez and Jorge Cortés. Distributed online convex optimization over jointly connected digraphs. *IEEE Transactions on Network Science and Engineering*, 1(1):23–37, 2014.

Brendan McMahan and Matthew Streeter. Delay-tolerant algorithms for asynchronous distributed online learning. In *Advances in Neural Information Processing Systems*, pages 2915–2923, 2014.

H Brendan McMahan. A survey of algorithms and analysis for adaptive online learning. *The Journal of Machine Learning Research*, 18(1):3117–3166, 2017.

H. Brendan McMahan and Matthew Streeter. Adaptive bound optimization for online convex optimization. In *COLT '10: Proceedings of the 23rd Annual Conference on Learning Theory*, 2010.

Panayotis Mertikopoulos. *Online Optimization and Learning in Games: Theory and Applications*. Habilitation à Diriger des Recherches (HDR), Université Grenoble-Alpes, December 2019.

Panayotis Mertikopoulos and Zhengyuan Zhou. Learning in games with continuous action sets and unknown payoff functions. *Mathematical Programming*, 173(1-2):465–507, January 2019.

Konstantin Mishchenko, Franck Iutzeler, and Jérôme Malick. A distributed flexible delay-tolerant proximal gradient algorithm. *SIAM Journal on Optimization*, 30(1):933–959, 2020.

Mehryar Mohri and Scott Yang. Accelerating online convex optimization via adaptive prediction. In *Artificial Intelligence and Statistics*, pages 848–856, 2016.

Eduardo F Nakamura, Antonio AF Loureiro, and Alejandro C Frery. Information fusion for wireless sensor networks: Methods, models, and classifications. *ACM Computing Surveys (CSUR)*, 39(3):9–es, 2007.

Shravan Narayanamurthy, Markus Weimer, Dhruv Mahajan, Tyson Condie, and Sundararajan Sellamanickam. Towards resource-elastic machine learning. In *NIPS 2013 BigLearn Workshop*, 2013.

Arkadi Semen Nemirovski and David Berkovich Yudin. *Problem Complexity and Method Efficiency in Optimization*. Wiley, New York, NY, 1983.

Yurii Nesterov. Primal-dual subgradient methods for convex problems. *Mathematical Programming*, 120(1):221–259, 2009a.

Yurii Nesterov. Primal-dual subgradient methods for convex problems. *Mathematical programming*, 120(1):221–259, 2009b.

Zhimin Peng, Yangyang Xu, Ming Yan, and Wotao Yin. Arock: an algorithmic framework for asynchronous parallel coordinate updates. *SIAM Journal on Scientific Computing*, 38(5):A2851–A2879, 2016.

Ciara Pike-Burke, Shipra Agrawal, Csaba Szepesvári, and Steffen Grunewalder. Bandits with delayed, aggregated anonymous feedback. In *ICML '18: Proceedings of the 35th International Conference on Machine Learning*, 2018.

Kent Quanrud and Daniel Khashabi. Online learning with adversarial delays. In *Advances in Neural Information Processing Systems*, pages 1270–1278, 2015.

Michael Rabbat and Robert Nowak. Distributed optimization in sensor networks. In *Proceedings of the 3rd international symposium on Information processing in sensor networks*, pages 20–27, 2004.

Alexander Rakhlin and Karthik Sridharan. Optimization, learning, and games with predictable sequences. In *NIPS '13: Proceedings of the 27th International Conference on Neural Information Processing Systems*, 2013.

Usman Raza, Alessandro Camerra, Amy L Murphy, Themis Palpanas, and Gian Pietro Picco. Practical data prediction for real-world wireless sensor networks. *IEEE Transactions on Knowledge and Data Engineering*, 27(8):2231–2244, 2015.

Shahin Shahrampour and Ali Jadbabaie. Distributed online optimization in dynamic environments using mirror descent. *IEEE Transactions on Automatic Control*, 63(3):714–725, 2017.

Shai Shalev-Shwartz. *Online learning: Theory, algorithms, and applications*. PhD thesis, Hebrew University of Jerusalem, 2007.

Shai Shalev-Shwartz. Online learning and online convex optimization. *Foundations and Trends in Machine Learning*, 4 (2):107–194, 2011.

Shai Shalev-Shwartz and Yoram Singer. Convex repeated games and Fenchel duality. In *NIPS' 06: Proceedings of the 19th Annual Conference on Neural Information Processing Systems*, pages 1265–1272. MIT Press, 2006.

Shai Shalev-Shwartz et al. Online learning and online convex optimization. *Foundations and trends in Machine Learning*, 4(2):107–194, 2011.

Weisong Shi, Jie Cao, Quan Zhang, Youhuizi Li, and Lanyu Xu. Edge computing: Vision and challenges. *IEEE internet of things journal*, 3(5):637–646, 2016.

Balazs Szorenyi, Róbert Busa-Fekete, István Hegedus, Róbert Ormándi, Márk Jelasity, and Balázs Kégl. Gossip-based distributed stochastic bandit algorithms. In *International Conference on Machine Learning*, pages 19–27. PMLR, 2013.

N Denizcan Vanli, Mert Gurbuzbalaban, and Asuman Ozdaglar. Global convergence rate of proximal incremental aggregated gradient methods. *SIAM Journal on Optimization*, 28(2):1282–1300, 2018.

Claire Vernade, Olivier Cappé, and Vianney Perchet. Stochastic bandit models for delayed conversions. In *UAI' 17: Proceedings of the 33rd Annual Conference on Uncertainty in Artificial Intelligence*, 2017.

Xiaofei Wang, Yiwen Han, Victor CM Leung, Dusit Niyato, Xueqiang Yan, and Xu Chen. Convergence of edge computing and deep learning: A comprehensive survey. *IEEE Communications Surveys & Tutorials*, 22(2):869–904, 2020.

Marcelo J Weinberger and Erik Ordentlich. On delayed prediction of individual sequences. *IEEE Transactions on Information Theory*, 48(7):1959–1976, 2002.

Lin Xiao. Dual averaging method for regularized stochastic learning and online optimization. In *Advances in Neural Information Processing Systems*, pages 2116–2124, 2009.

Lin Xiao. Dual averaging methods for regularized stochastic learning and online optimization. *Journal of Machine Learning Research*, 11:2543–2596, October 2010.

Jie Xu, Cem Tekin, Simpson Zhang, and Mihaela Van Der Schaar. Distributed multi-agent online learning based on global feedback. *IEEE Transactions on Signal Processing*, 63(9):2225–2238, 2015.

Feng Yan, Shreyas Sundaram, SVN Vishwanathan, and Yuan Qi. Distributed autonomous online learning: Regrets and intrinsic privacy-preserving properties. *IEEE Transactions on Knowledge and Data Engineering*, 25(11):2483–2493, 2012.

Constantin Zalinescu. *Convex analysis in general vector spaces*. World scientific, 2002.

Yan Zhang, Robert J Ravier, Michael M Zavlanos, and Vahid Tarokh. A distributed online convex optimization algorithm with improved dynamic regret. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 2449–2454. IEEE, 2019.

Julian Zimmert and Yevgeny Seldin. An optimal algorithm for adversarial bandits with arbitrary delays. In *AISTATS '20: Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics*, 2020.