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

Online learning has been successfully applied to many problems in which data are revealed over time. 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, at both the agent and network levels. 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 applications in portfolio selection, online auctions, recommender systems, and many other fields; for an introduction to the topic, see the textbooks by Shalev-Shwartz et al. (2011), Bubeck and Cesa-Bianchi (2012), Hazan (2016), and references therein. In the most basic version of the problem, the agent (or "learner") chooses an action, the cost of this action is subsequently revealed to the agent along with some feedback (often gradient-based), 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 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. In the rest of this section we detail our specific contributions in the context of related work.

1.1 Our contributions

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 stale and obsolete. The second has to do with *multi-agent* systems: learners may have to take decisions with very different information at their disposal, and with no realistic means of coordinating their decision-making mechanisms. Accordingly, 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 (2009) which we combine with adaptive learning rates inspired by the "inverse square root of the sum" blueprint of McMahan and Streeter (2010) and Duchi et al. (2011). We show that the resulting policies achieve optimal data-dependent guarantees and can automatically adapt to the delays encountered by the agents (cf. Section 3); in particular, under a mild monotonicity requirement for the total amount of information available to the decision-making agent, this policy remains optimal *even if the delays are not bounded.* To the best of our knowledge, this is the first adaptive algorithm that does not require a "bounded delay" assumption in this context.

In the above setting, we do not assume a specific correlation model between the order in which agents perform their updates and the order in which gradients are received; however, we do assume that each agent can observe a global counter that indicates the number of updates performed in the entire network so far. In many cases of practical interest, this assumption is relatively benign; however, in fully decentralized deployments of machine learning algorithms, it is not clear how this information can be made available at every device. For this reason, we also provide in Section 4 a fully distributed version of our adaptive strategy which enjoys the same data-dependent bounds without requiring a global clock. In addition, in Section 5, we take a closer look at decentralized architectures, and we provide bounds for the agents' effective and collective regret to account for two distinct objectives of the learning system: in the former, the goal is to perform well on every upcoming request; in the latter, on the collective, collaborative task undertaken by the agents.

Finally, in Section 6, we focus on improving these worst-case bounds by introducing a more "optimistic" step-size policy in the spirit of Rakhlin and Sridharan (2013). 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.

1.2 Related work

Our work lies at the interface between multiple active research areas, each tackling a special case of the general framework considered in this paper. We provide below an overview of these related 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 online dual averaging, and suggested using the classical doubling trick to dynamically adjust the learning rate.[1] 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. 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 the phrase *asynchronous distributed* appeared in the title of McMahan and Streeter (2014), it falls into the single-agent case in our framework since only one updater was considered there. In particular, in the single-agent setting, we manage to provide a novel adaptive method that does

1. Due to a lack of consensus in the literature, Quanrud and Khashabi (2015) used the name online mirror descent to refer to online dual averaging. See Remark 1 for further discussion.

not require the "bounded delay" assumption which is present in the work of McMahan and Streeter (2014) and Joulani et al. (2016).

The impact of delays has also 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 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. Finally, we analyze both the *effective* and the *collective* regret of the network (see Section 5.2) whereas the works mentioned above only focus on the latter.[2]

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.

2. The effective and collective regrets are also called "network" and "agent" regrets in Mateos-Núnez and Cortés (2014) and Akbari et al. (2015).

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.

2. A general framework for asynchronous online optimization

2.1 Problem and feedback structure

Let us consider a set of M agents playing against a sequence of time-varying loss functions, with the goal to achieve a low regret. Formally, at each time slot $t = 1, 2, \dots$, one of the agents becomes *active*, plays a point x_t from the constraint set $\mathcal{X}$, and incurs 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 $f_t \colon \mathbb{R}^d \to \mathbb{R} \cup \{+\infty\}$ is convex, lower semi-continuous, and has $\mathcal{X} \subset \operatorname{dom} \partial f_t$, where ∂f_t denotes the subdifferential of f_t. 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 is played (namely, g_t is a subgradient of f_t at x_t).[4] We will refer to x_t as either the *prediction* made by the active agent or the *action* played by the active agent at time t regardless of the nature of the problem, and we write $i(t)$ for the agent that is active at time t.

The delay model. In environments with *delayed feedback*, g_t is only received by an agent at a certain amount of time after having played x_t. 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.

To express this formally, we write $[t-1] \coloneqq \{1, \dots, t-1\}$ and we introduce $\mathcal{S}_{i,t} \subset [t-1]$ 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 only has $\{g_s : s \in \mathcal{S}_{i,t}\}$ at their disposal. Clearly, for all $i = 1, \dots, M$, the sequence $(\mathcal{S}_{i,t})_{t \in [T]}$ is non-decreasing, i.e.,

$$\mathcal{S}_{i,t} \subset \mathcal{S}_{i,t+1} \quad \text{for all } t = 1, 2, \dots \tag{2}$$

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

$$\mathcal{S}_t = \mathcal{S}_{i(t),t} \qquad \text{and} \qquad \mathcal{U}_t = [t-1] \setminus \mathcal{S}_t \tag{3}$$

for the set of timestamps that are available (resp. unavailable) to the active agent at time t. As such, in this notation, the non-delayed setting corresponds to the case $\mathcal{S}_t = \mathcal{S}_{i,t} = [t-1]$ and $\mathcal{U}_t = \varnothing$.

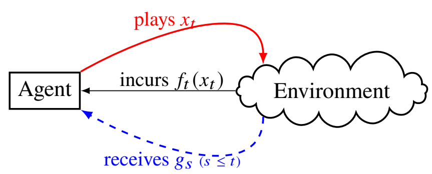
3. For simplicity, we assume throughout that only one agent is active at each time step.

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

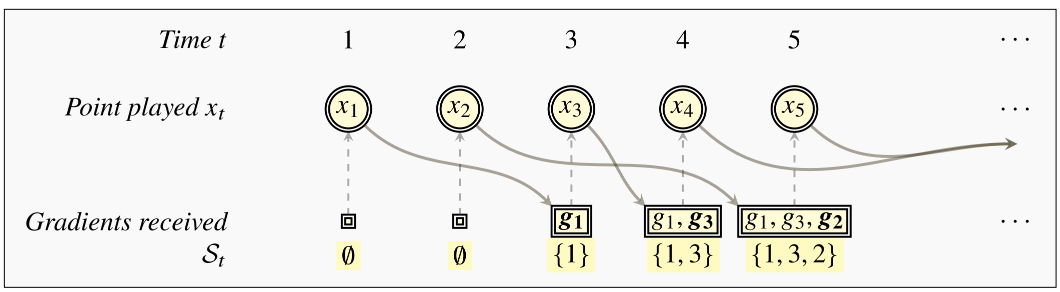
2.2 Running examples

We now present two examples intended to showcase the range of the proposed framework. In order to better describe practical implementations, we separate the single- and multi-agent cases below.

Case 1: Single agent with delayed feedback In the case of a single agent, the available information simply comes from aggregating first-order feedback from previous actions. Since the agent's feedback may be subject to delays, the arrival of gradients may not be sequential or synchronized with the playing times. This is the common *delayed feedback* setup in the literature (see McMahan and Streeter, 2014; Joulani et al., 2013, 2016; Quanrud and Khashabi, 2015). Importantly, while the available feedback $\mathcal{S}_t = \mathcal{S}_{1,t}$ is necessarily monotone in the single-agent case ($\mathcal{S}_t = \mathcal{S}_{1,t} \subset \mathcal{S}_{t+1,1} = \mathcal{S}_{t+1}$ by Eq. (2)), the gradients may be received out-of-order (g_s may be received before g_t even if $t < s$).



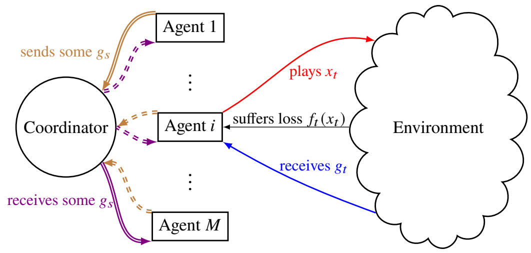
Example 1. *To provide some intuition on the type feedback arrival that this setup can lead to, we give below an example with $T = 5$ rounds.*



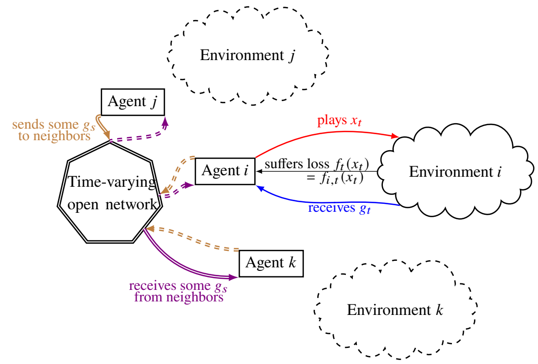
Case 2: Multiple asynchronous agents When there are multiple agents, feedback delays can come from the agents' own delay in generating the feedback but also from the time taken to communicate with other agents. Since there are communications, it is important to define how and why they are done. We exhibit two distinct setups below:

2a) A coordinator-worker asynchronous setup where M agents collaborate in a common time-varying environment. The loss then represents a time-varying global problem with asynchronous interactions between the agents and an individual, agent-specific feedback mechanism (that can result itself in delayed feedback). The agents then collaborate by sending their feedbacks to a coordinator that broadcasts the new gathered information to all participants.[5] This communication naturally induces some delay.

5. One can imagine that in order to reduce communications, the workers wait to gather new feedback and only then send their sum to the coordinator (while still performing updates based on these local feedback), which itself can wait for a certain amount of received feedback to broadcast it back. In addition to reduce communication, such a mechanism can also protect some privacy.



2a) An asynchronous coordinator-worker setup.



2b) A decentralized open network.

2b) A decentralized open network, i.e., a time-varying number M_t of agents connected by a graph, collaborating to improve their individual loss by the help of global information.

For a schematic representation, see the figure above. It is also worth noting that, since the active agent differs from one time slot to another, the sequence $(\mathcal{S}_t)_{t\in\mathbb{N}}$ is not necessarily monotone ($\mathcal{S}_t \not\subset \mathcal{S}_{t+1}$ even though the sequences $\mathcal{S}_{i,t}$ are by Eq. (2)). We make this precise below:

Example 2. *We provide below an example with $T = 5$ rounds and two workers. Note that the situation below can arise in both cases.*

Time t	1	2	3	4	5	$\cdots$
Active agent $i(t)$	2	1	1	2	1	
Point played x_t	x_1	x_2	x_3	x_4	x_5	$\cdots$
Gradients received by 1			$\boldsymbol{g_2}$	$g_2, \boldsymbol{g_3}$	$g_2, g_3, \boldsymbol{g_1}$	$\cdots$
$\mathcal{S}_{1,t}$	$\emptyset$	$\emptyset$	$\{2\}$	$\{2,3\}$	$\{2,3,1\}$	
Gradients received by 2			$\boldsymbol{g_1}$	g_1	$g_1, \boldsymbol{g_2}, \boldsymbol{g_3}$	$\cdots$
$\mathcal{S}_{2,t}$	$\emptyset$	$\emptyset$	$\{1\}$	$\{1\}$	$\{1,2,3\}$	
$\mathcal{S}_t$	$\emptyset$	$\emptyset$	$\{2\}$	$\{1\}$	$\{2,3,1\}$	$\cdots$

2.3 Candidate algorithms

The algorithms that we will examine rely on the use of a suitable regularizer function $h\colon \mathbb{R}^d \to \mathbb{R}\cup\{+\infty\}$ to stabilize the predictions. To define it, we assume that the ambient space $\mathbb{R}^d$ is equipped with a norm $\|\cdot\|$, and we write $\|\cdot\|_*$ for the induced dual norm. Then, we say that h is a *regularizer* on $\mathcal{X}$ if $\mathcal{X} \subset \overline{\operatorname{dom} h}$, the subdifferential ∂h admits a continuous selection denoted by ∇h, and h is 1-strongly convex relative to $\|\cdot\|$ on $\mathcal{X}$. This allows us to define the *Bregman divergence* induced by h as

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

and the corresponding *mirror map*

$$P\colon y \mapsto \operatorname*{arg\,min}_{x\in\mathcal{X}} \ \langle -y, x\rangle + h(x).$$

Two of the most popular candidates of 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. Finally, without loss of generality, we also assume in the sequel that h is non-negative. In fact, as h is strongly convex, $\min h$ is always well-defined and replacing h by $h - \min h$ does not affect any of the algorithms studied in our paper (for example, in the case of entropic gradient descent, we let $h(x) = \sum_k x[k]\log x[k] + d\log d$).

Two families of methods. . . In the non-delayed case, the algorithmic strategies that the agents can follow to optimize their regret using first-order feedback follow two closely related trends:

- *Online mirror descent:*

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

- *Online dual averaging:*

$$x_t = \operatorname*{arg\,min}_{x\in\mathcal{X}} \left\{ \sum_{s<t} \langle g_s, x\rangle + \frac{1}{\eta_t} h(x) \right\} = P\left(-\eta_t \sum_{s<t} g_s\right). \tag{ODA}$$

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

Remark 1. *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 $x_t \leftarrow P(-\sum_{s<t} \eta_s g_s)$, see e.g., Nesterov (2009) 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 (2009), and differs from the lazy variant of* (OMD) *in that all gradients enter the algorithm with the same weight. From an online learning viewpoint,* (ODA) *can also be seen as a "linearized" version of the "follow the regularized leader" (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.*

. . . not equally robust to delays When the feedback to the agents is subject to *delays*, the players have to take decisions with information that is possibly out of order. Trying to extend (OMD) and (ODA) to cope for such a situation sheds considerable light on their fundamental differences in terms of robustness to delays.

Indeed, if feedback arrives out-of-order, 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 (ODA), 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, 2009, Sec. 1.2); this is a very appealing feature, especially when trying to incorporate delayed gradients or gradients generated by other agents.

This feature of (ODA) is due to the aggregation of gradient feedback as it arrives. For instance, if an agent is given a feedback from another agent, it is not immediate to plug it in the (OMD) update whereas it is simply added to the sum in (ODA). In addition, dual averaging strategies were also found to perform better in manifold identification (Lee and Wright, 2012) or in the presence of noise (Flammarion and Bach, 2017). This suggests that the anonymous feedback aggregation properties of (ODA) make it the method of choice for online learning with asynchronous delayed feedback. In the rest of our paper, we will act on this intuition and focus exclusively on (ODA).

3. Delayed dual averaging and asynchronicity

We 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 $\mathcal{S}_t \subset [t-1]$ 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 computation of x_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 delay.

3.1 Delayed dual averaging

Following the discussion of Section 2.3, our candidate algorithm for this asynchronous setup is the delayed dual averaging policy:

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

We start by establishing a data-dependent regret bound of the algorithm.

Theorem 1. *Assume that delayed dual averaging* (DDA) *is run with a non-increasing learning rate sequence* $(\eta_t)_{t\in[T]}$. *Then the generated points* $x_1, \ldots, x_T$ *satisfy*

$$\mathbf{Reg}_T(u) \le \frac{h(u)}{\eta_T} + \frac{1}{2}\sum_{t=1}^{T} \eta_t \left(\|g_t\|_*^2 + 2\|g_t\|_* \sum_{s \in \mathcal{U}_t} \|g_s\|_* \right).$$

This result will be proven as a special case of a more general result in the sequel (Theorem 7). We note the bound consists of the usual online dual averaging bound (cf. Appendix A.1) plus a term $\sum_{s\in\mathcal{U}_t} \|g_s\|_*$ that captures the feedback that *could* have been used (since the corresponding point was played) but was not because it was delayed. In the literature, this kind of regret bound has only been proven for the single-agent scenario. In more detail, both McMahan and Streeter (2014) and Joulani et al. (2016) established a very similar regret guarantee for online gradient/mirror descent when only one agent is involved in the learning process.[6] However, to the best of our knowledge, this result is

6. 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.

not known for dual averaging even in the single-agent case, while a looser bound can be found in Quanrud and Khashabi (2015) for a constant learning rate schedule $\eta_t \equiv \eta$.

3.2 Delays and lag

Delays for the feedback can be measured in several ways. We provide herein four measures that reflect different aspects of delay:

- The *maximum delay* τ is the longest wait to receive a feedback $\tau = \min\{\tau : [t-\tau-1] \subset \mathcal{S}_t \text{ for all } t \in [T]\}$. Notice that as we are in a multi-agent setting, there is not a single delay associated with every individual subgradient (all the agents do not receive a feedback at the same time), while it is still possible to have a bound on the delays of the feedback.

- The *maximum unavailability* ν of the feedback is defined by $\nu = \max_{t\in[T]} \operatorname{card}(\mathcal{U}_t)$. This is the maximum number of subgradients that could have (but have not) been communicated to an active agent when it performs the prediction. It is straightforward to see that $\nu \leq \tau$.[7]

- The *cumulative unavailability* D is given by $D = \sum_{s=1}^{T} \operatorname{card}(\mathcal{U}_s)$. This generalizes the sum of delays to the multi-agent case; it is direct to see that $D \leq \nu T$.

- The *lag* at time t defined by

$$\Lambda_t = \sum_{s=1}^{t} \left(\|g_s\|_*^2 + 2\|g_s\|_* \sum_{l\in\mathcal{U}_s} \|g_l\|_* \right). \tag{4}$$

  It sums over the errors that are caused by the inability of the learners to compensate the missing feedback. It gives the most fine-grained characterization of the effect of delayed feedback on the regret. Similar quantities have also been defined in Joulani et al. (2016); McMahan and Streeter (2014).

Under these notions of delay, a direct application of Theorem 1 with a constant learning rate gives the following result.

Corollary 2. *Let delayed dual averaging* (DDA) *run with constant learning rate* $\eta_t \equiv \eta$*.*

- *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* $\eta = \Theta(1/\sqrt{D})$*, then* $\mathbf{Reg}_T(u) = \mathcal{O}(\sqrt{D})$*.*
- *If* $\eta = \Theta(1/\sqrt{\Lambda_T})$*, then* $\mathbf{Reg}_T(u) = \mathcal{O}(\sqrt{\Lambda_T})$*.*

Comparing Corollary 2 with known results in the literature, we find out that in the single-agent setup, we successfully recover the optimal data-dependent bound in $\mathcal{O}(\sqrt{\Lambda_T})$ as presented in Joulani et al. (2016); McMahan and Streeter (2014). Moreover, if we further assume that $\|g_t\|_* \leq G$ for all $t \in [T]$, we have $\Lambda_T \leq (T+2D)G^2$ which leads to the well-known $\mathcal{O}(\sqrt{D})$ bound on the regret (see e.g., Quanrud and Khashabi, 2015). Finally, since neither Λ_T nor D can be known beforehand, in practice one may need to use a more conservative learning rate in the order of $\Theta(1/\sqrt{\nu T})$. We address this important issue via the design of proper adaptive methods in the next section.

7. For any $t \in [T]$, we have $[t-\tau-1] \subset \mathcal{S}_t$ and thus $\mathcal{U}_t = [t-1] \setminus \mathcal{S}_t \subset \{t-\tau-1, \ldots, t-1\}$ 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 $\nu \ll \tau$.

3.3 Variable learning rate

In this section, we exploit the regret bound of Theorem 1 to design efficient leaning rates that provably achieve low regret. To clarify our objective, we identify here three characteristics that an ideal learning rate schedule would like to have: *i*) arbitrary time horizon: the computation of the learning rate does not require the knowledge of a predetermined time horizon T; *ii*) data-dependant: the regret features the actual feedback instead of an upper bound on the gradient norms; and *iii*) delay-adaptive: the regret depends on the actual delays and not only an worst-case estimation.

To produce a learning rate with the above properties, we rely on the classic trick of "inverse square root of the sum" (see Lemma 22 in Appendix B for mathematical details). To obtain a $\mathcal{O}(\sqrt{\Lambda_T})$ regret, it consists in taking $\eta_t = 1/\sqrt{\Lambda_t} = 1/\sqrt{\sum_{s=1}^t \lambda_s}$ where

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

Obviously this policy is not implementable in our context since it involves unobserved feedback. Nevertheless, it can be approximated in some situations. To streamline our presentation, all the proofs of this subsection are deferred to Appendix B.

3.3.1 Pessimistic non-adaptive learning rate

To set the stage for the more general analysis to come, we begin with the assumption that the maximum delay τ (or an upper bound thereof) is known to the agents, and that the norms of the gradients are bounded by a known constant G. This leads to $\lambda_s \leq G^2(1 + 2\nu) \leq G^2(1 + 2\tau)$. Combining Lemma 22 and Theorem 1, we obtain the following result.

Proposition 3. *Assume that the maximum delay is bounded by τ and the norms of the gradients are bounded by G. Let delayed dual averaging* (DDA) *be run with learning rates*

$$\eta_t = \frac{R}{G\sqrt{t(1 + 2\tau)}}. \tag{Decreasing}$$

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

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

This approximation is crude and can be improved with the actual knowledge of the gradients. Also note that the implementation of this basic strategy requires the knowledge of t, the number of actions that has been played. We will discuss the plausibility of this requirement in Section 3.4.

3.3.2 Adaptation to bounded delays

In addition to the bounded delay and gradients assumptions taken before, we add here another assumption that the sequence of active feedback is non-decreasing, i.e., $\mathcal{S}_t \subset \mathcal{S}_{t+1}$. When this specific assumption is satisfied, a crucial point is that one can define an *arrival order* for the received subgradients. Mathematically, we can define a permutation σ of $\{1, \ldots, T\}$ such that the k-th received subgradient comes from playing $x_{\sigma(k)}$, i.e., the k-th received subgradient is

$g_{\sigma(k)} \in \partial f_{\sigma(k)}(x_{\sigma(k)})$.[8] With this notation, the time index set of all feedback received *before* g_t can be written as $\mathcal{G}_t := \{\sigma(1), \sigma(2), ..\sigma(\sigma^{-1}(t) - 1)\}$ for that g_t is the $\sigma^{-1}(t)$-th feedback.

Using these definitions and looking closely at the definition of the lag (4), we notice that:

- the quantity $\sum_{s=1}^{t} \|g_s\|_*^2$ cannot be known at instant t since the set of gradients available at that time is $\mathcal{S}_t$. It is thus natural to consider approximating it by $\sum_{s\in\mathcal{S}_t} \|g_s\|_*^2$;

- for each received feedback g_s, the quantity $\sum_{l\in\mathcal{U}_s} \|g_l\|_*$, gathering all feedback before s that were not used to compute g_s, is in general unknown. Building on the works of Joulani et al. (2016); McMahan and Streeter (2014), a proxy for this sum is $\sum_{l\in\mathcal{G}_s\setminus\mathcal{S}_s} \|g_s\|_*$. In words, this represents the sum over the feedback received before g_s which was not used to generate it.

Putting these two modifications together, we obtain

$$\Gamma_t = \sum_{s\in\mathcal{S}_t} \left(\|g_s\|_*^2 + 2\|g_s\|_* \sum_{l\in\mathcal{G}_s\setminus\mathcal{S}_s} \|g_l\|_* \right). \tag{5}$$

Now, Γ_t can be computed at time t by the player. Bounding Λ_t using Γ_t and the maximum delay, we obtain the following result.

Proposition 4. *Assume that the maximum delay is bounded by τ, the norm of the gradients are bounded by G, and the sequence of active feedback is non-decreasing, i.e., $\mathcal{S}_t \subset \mathcal{S}_{t+1}$. Let delayed dual averaging* (DDA) *be run with a learning rate sequence*

$$\eta_t = \frac{R}{\sqrt{\Gamma_t + G^2(2\tau^2 + 3\tau + 1)}}. \tag{AdaDelay–O}$$

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

$$\mathbf{Reg}_T(u) \le 2R\sqrt{\Gamma_T + G^2(2\tau^2 + 3\tau + 1)} \le 2R\sqrt{\Lambda_T + G^2(2\tau^2 + 3\tau + 1)}.$$

We call this method AdaDelay–O, where O stands for online. We avoid using the name AdaDelay since it has already been attributed to another algorithm that was designed for stochastic optimization with delays (Sra et al., 2016). We notice that AdaDelay–O is similar to the algorithm proposed by Joulani et al. (2016) and we obtain a regret bound of the same order. Nonetheless, their methods were based on mirror descent and FTRL-Prox, while ours is based on dual averaging.

3.3.3 Adaptation to unbounded delays

Propositions 3 and 4 explicitly use the knowledge of (an upper bound of) the maximum delay τ. However, an investigation into the proof of Proposition 4 shows that by relating Λ_t and Γ_t correctly, it is in reality possible to provide a learning rate policy for which no knowledge about the delay is needed. To that end, let $D_t := \sum_{s=1}^{t} \operatorname{card}(\mathcal{U}_s)$ be the cumulative unavailability at time t (so $D = D_T$) and $\mathcal{A}_t := \{\{s,l\} : s \in \mathcal{S}_t, l \in \mathcal{G}_s \setminus \mathcal{S}_s\}$ be the set of time index pairs of the proxy term Γ_t (here $\{s,l\}$ denotes an *unordered* pair of *distinct* elements). It suffices then to replace the $(2\tau^2 + 3\tau + 1)$ term of AdaDelay–O by $\tilde{\tau}_t := t + 2D_t - \operatorname{card}(\mathcal{S}_t) - 2\operatorname{card}(\mathcal{A}_t)$, which can be computed efficiently on the fly (see the pseudo-code in Appendix B.2).

8. This permutation may not be unique (for instance if several gradients arrive at one time slot) but this plays no role in the subsequent analysis.

Proposition 5. *Assume that the norms of the gradients are bounded by G and the sequence of active feedback is non-decreasing, i.e., $\mathcal{S}_t \subset \mathcal{S}_{t+1}$. Assume further that delayed dual averaging (DDA) is run with the learning rate sequence*

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

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

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

This new algorithm is called AdaDelay–O+. To our knowledge, it is the first adaptive online algorithm that does not require any bounded delay assumption. Furthermore, 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.2), so this bound in the worst case matches the data-dependent bound of Proposition 4.

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

In this regard, AdaDelay–O+ emerges as an appealing "go-to" choice in the presence of delays; accordingly, much of the discussion to follow will be based on extending its properties to more general settings.

3.4 Back to running examples

Now that we have laid out our delayed dual averaging (DDA) method as well as different learning rate policies, we can go back to our examples to evaluate the reach and implementability of the proposed techniques.

Case 1: Single agent with delayed feedback For a single agent, (DDA) can be simply implemented as Algorithm 1. Note here that *i*) the iteration number t is obviously available to the agent; and *ii*) that the sequence of active feedback i.e., $\mathcal{S}_t \subset \mathcal{S}_{t+1}$ (see e.g., Example 1). This means that all the learning rate policies presented before can be used (constant, Decreasing, AdaDelay–O, AdaDelay–O+); note that the data-dependent stepsizes AdaDelay–O (and AdaDelay–O+ for unbounded delays) are in general preferable to the direct Decreasing strategy.

Case 2: Multiple asynchronous agents For multiple agents, (DDA) can be implemented by an agent i as Algorithm 2. Interestingly, this is the same algorithm for both the coordinator-worker and decentralized cases (*2a* and *2b*) since only the set of feedback is essential, not the actual communication scheme. However, to the difference of the single agent case: *i*) the iteration number t may not be available to the agents; *ii*) the sequence of active feedback is not non-decreasing i.e., $\mathcal{S}_t \not\subset \mathcal{S}_{t+1}$ (see e.g., Example 2).

9. The asynchronous reception in line 3 stands both for the reception of a feedback for a point played by the agent or a (bunch of) feedback gathered by communication.

Algorithm 1 (DDA) – Single agent

```
Initialize: 𝒢 ← ∅, t ← 1.
while not stopped do
  asynchronously receive feedback g_s from time s: 𝒢 ← 𝒢 ∪ {s}
  if requested to play an action x_t then
    𝒮_t ← 𝒢
    Play x_t = arg min_{x∈𝒳} Σ_{s∈𝒮_t} ⟨g_s, x⟩ + h(x)/η_t
    t ← t + 1
  end if
end while
```

Algorithm 2 (DDA) – multiple agents – from the point of view of agent i

```
1:  Initialize: 𝒢_i ← ∅, t ← 1.
2:  while not stopped do
3:    asynchronously receive feedback g_s from time s: 𝒢_i ← 𝒢_i ∪ {s}[9]
4:    if the agent becomes active, i.e., i(t) = i then
5:      𝒮_t ← 𝒢_i
6:      Play x_t = arg min_{x∈𝒳} Σ_{s∈𝒮_t} ⟨g_s, x⟩ + h(x)/η_t
7:      Relay g_s if necessary
8:      t ← t + 1
9:    else if another agent plays an action then
10:     t ← t + 1
11:   end if
12: end while
```

- *Case 2a: Coordinator-worker asynchronous.* In this case, Algorithm 2 can be applied quite readily. For example, if the agents are queried cyclically and pool their feedback at the end of every K cycles, Corollary 2 shows that by using a fixed learning rate $\eta = \Theta(1/\sqrt{KMT})$, a regret of $\mathbf{Reg}_T(u) = \mathcal{O}(\sqrt{KMT})$ is obtained under the bounded gradient assumption.

  Here, in order to know the current iteration t, every agent has to know when the other agents play in order to update its time-counter (see lines 9 and 10). When a coordinator is used, this could be a reasonable assumptions which allows taking the decreasing learning rate policy Decreasing. Since the feedback is not non-decreasing, data-dependent strategies are out of grasp.

- *Case 2b: Decentralized open network.* For this setup, applying Algorithm 2 is possible but considerably harder. Since in this very flexible setup, no global information is available (e.g. cumulative unavailability, activity, or presence of other agents), lines 9 and 10 are almost impossible to enforce. Thus, even Decreasing learning rates are out of scope, let alone data-dependent ones.

The limitations of these strategies can be partly attributed to Theorem 1, which is only applicable if one takes a non-increasing learning rate policy *over time* independently of the agents. As we saw above, this is completely impractical in multi-agents settings. This calls for a refined analysis to account for *implementable* and *adaptive* learning rates.

4. Tuning the learning rate in distributed systems

Providing a bona fide asynchronous algorithm and analysis implies paying a special attention to the knowledge of the agents, not only on the available feedback, but also on the implementability of the learning rate strategy. This leads to new definitions, a refined analysis, and revised adaptive algorithms.

4.1 More flexible learning rates for asynchronous dual averaging

As we saw from the running examples, the knowledge of a global time corresponding to the total number of actions played by all agents can be unrealistic in some setting. Thus, implementing a learning rate policy that is non-increasing with respect to time may be impossible. Nonetheless, the non-increasingness of the learning rate is crucial to the analysis of online dual averaging. To marry these two seemingly irreconcilable components, we need to dig into the intricate relation between the delays and the actual updates, which is partially made easy thanks to the framework that we have established. Then, by reordering the time appropriately, the analysis proceeds and we will be able to bound the regret as desired.

Formally, we define a set of permutations of $\{1, \dots, T\}$, called *faithful permutations*, over which a non-increasing learning rate leads to to a regret bound similar to that of Theorem 1.

Definition 6 (faithful permutation). *A permutation σ of the set $\{1, \dots, T\}$ is* faithful *if and only if $s \in \mathcal{S}_t$ (i.e., g_s is available for choosing x_t) implies $\sigma^{-1}(s) < \sigma^{-1}(t)$.*

A permutation σ being faithful 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.

Note that obviously the identity is always a faithful permutation. We also encountered another faithful permutation in Section 3.3.2 when we consider the arrival order of the gradients in the monotonous case $\mathcal{S}_t \subset \mathcal{S}_{t+1}$ (cf. proof of Proposition 24).

Now that we have laid out all the necessary ingredients, the next theorem bounds the regret for a learning rate sequence that is non-increasing *along a faithful permutation*.

Theorem 7. *Let σ be a faithful permutation of $\{1, \dots, T\}$. Assume that delayed dual averaging* (DDA) *is run with a learning rate sequence $(\eta_t)_{t\in[T]}$ such that $\eta_{\sigma(t+1)} \le \eta_{\sigma(t)}$ for all t. Then the generated points $x_1, \dots, x_T$ enjoy the regret bound*

$$\mathbf{Reg}_T(u) \le \frac{h(u)}{\eta_{\sigma(T)}} + \frac{1}{2}\sum_{t=1}^{T} \eta_{\sigma(t)} \left(\|g_{\sigma(t)}\|_*^2 + 2\|g_{\sigma(t)}\|_* \sum_{s\in\{\sigma(1),\dots,\sigma(t-1)\}\setminus\mathcal{S}_{\sigma(t)}} \|g_s\|_* \right).$$

Sketch of proof. The complete proof is presented in Appendix A.3. To show this result, we leverage the so-called "perturbed iterate" framework for asynchronous algorithms in the spirit of Mania et al. (2017) and Stich and Karimireddy (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 bound the difference between the linearized regret achieved by the sequences (x_t) and $(\widetilde{x}_t)$. We then obtain a bound on the algorithm's regret by bounding the regret incurred by the sequence of virtual iterates $\widetilde{x}_t$ and combining the two. ■

This result extends Theorem 1 by providing a larger class of possible learning rate policies (while Theorem 1 is proved by taking σ as the identity permutation). This enables us to devise efficient and truly implementable learning rate update schemes in the next section.

4.2 Variable learning rate

As in Section 3.3, we aim at providing adaptive learning rate policies. However, we here place ourselves in the most general framework where the sequence of active feedback $(\mathcal{S}_t)_{t\in[T]}$ is not necessarily non-decreasing and even the knowledge of the global time (i.e., the total number of actions played) is out of reach. Doing so, we plan on providing efficient methods for cases *2a* and *2b* of our running examples.

4.2.1 Pessimistic non-adaptive learning rate

As discussed previously, in this very general setup, it is in general infeasible to implement a learning rate schedule that is non-increasing along the time sequence. Equipped with the knowledge of (an upper bound of) the maximum delay, it it however possible for an agent to make a pessimistic estimate of the number of actions that has been played, based on the number of the subgradients that it has received. In this regard, we just need to define a faithful permutation such that along this permutation the learning rate is non-increasing in order to apply Theorem 7. To achieve this, we add the following assumption.

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

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 particularly implied by the upcoming Assumption 2 which is itself already a fairly weak assumption (see the accompanying discussion). Furthermore, if the agents relay the information $\operatorname{card}(\mathcal{S}_t)$ 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 8. *Assume that the maximum delay is bounded by τ, the norm of the gradients are bounded by G, and let Assumption 1 hold. Let delayed dual averaging* (DDA) *be run with learning rates*

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

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

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

Sketch of proof. The proof is detailed in Appendix B.3. After introducing an appropriate faithful permutation we bound the cardinality of $\{\sigma(1), \dots, \sigma(t-1)\} \setminus \mathcal{S}_{\sigma(t)}$ and apply the inverse square root of the sum trick (Lemma 22) to conclude. ∎

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

Proposition 8 mirrors Proposition 3, with the only difference that in lieu of the global time index t (which is in general unknown to agents), we use the number of feedbacks that an agent has received. This learning rate scheme can be effectively implemented in a decentralized multi-agent network, and retains the optimal $\mathcal{O}(\sqrt{\tau T})$ regret bound.

4.2.2 AdaGrad-style

The design of data-dependent learning rates in the distributed setting follows closely its counterpart in the single-agent setup. First, as $\mathcal{S}_{i,t}$ the feedback set *at some agent* i is necessarily growing (Eq. (2)), at each agent i, we can define an arrival order per agent in the form of a permutation σ_i of $\{1,\ldots,T\}$ such that the k-th received feedback comes from $x_{\sigma_i(k)}$ (played by i or another player), i.e., the k-th received feedback 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 $\mathcal{G}_{i,t} \coloneqq \{\sigma_i(1), \sigma_i(2), \ldots, \sigma_i(\sigma_i^{-1}(t)-1)\}$.

To proceed, we make also the following mild assumption: when an agent receives a gradient g_t, it has already received all the feedback that was used to compute it (i.e., $\mathcal{S}_t$).

Assumption 2. *For every node i and time t, we have* $\mathcal{S}_t \subset \mathcal{G}_{i,t}$.

The above assumption is notably verified in the following situations: *i*) a coordinator-worker scheme in which the transmission of the gradients are *in order* (first come, first served); *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 higher communication cost.

Now, since at time t, the active agent $i(t)$ has the knowledge of $\mathcal{S}_t$ by definition and of $\mathcal{G}_{i(t),s}$ for $s \in \mathcal{S}_t$ by construction, we can modify the lag approximation of (5) to

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

so that Γ_t^{Dist} is computable by the active worker $i(t)$ at time t. Note that the Γ_t^{Dist} still involves the knowledge of all past $\mathcal{S}_s$ (or rather $\sum_{l \in \mathcal{S}_s} \|g_l\|_*$), which imply some additional communication between the agents (of the order of one scalar per feedback sent). The obtained algorithm is detailed in Algorithm 3. Finally, although the feedback monotonicity is no longer assumed, the regret is very close to the one of Proposition 4 (actually only differs by a time-independent constant).

Proposition 9. *Assume that the maximum delay is bounded by* τ*, the norm of the gradients are bounded by* G*, and let Assumption 2 hold. Let delayed dual averaging* (DDA) *be run with a learning rate sequence*

$$\eta_t = \frac{R}{\sqrt{\Gamma_t^{\text{Dist}} + G^2(2\tau+1)^2}}. \qquad \text{(AdaDelay–Dist)}$$

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

$$\mathbf{Reg}_T(u) \le 2R \max_{1 \le t \le T} \sqrt{\Gamma_t^{\text{Dist}} + G^2(2\tau+1)^2} \le 2R\sqrt{\Lambda_T + G^2(2\tau+1)^2}.$$

Algorithm 3 (DDA) with AdaDelay–Dist – from the point of view of agent i

Initialize: $\mathcal{G}_i \leftarrow \emptyset$, $\widetilde{\Gamma}_i \leftarrow G^2(2\tau+1)^2$
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

Sketch of proof. The proof is reported in Appendix B.3. We first show that Γ_t^{Dist} shares the same characterization as Γ_t thanks to Assumption 2. Then, we prove that $\Gamma_t^{\text{Dist}} + G^2(2\tau+1)^2$ is an upper bound of a modified notion of lag. ∎

In Algorithm 3, the global time is no longer present and feedback are treated separately depending on whether they come from other agents (along with additional information) or come from a point played by the agent. Time indices are still present for ease of comprehension, notably to enhance the fact the a worker knows (and keeps track) of the feedback used to produce past point (i.e., $\sum_{s\in\mathcal{S}_t}\|g_s\|_*$ for each point x_t played by the worker).

Remark 2. *We did not try to design a learning rate strategy that adapts to unbounded delays in this section since in the decentralized case, an agent can hardly make any sensible estimation of the cumulative unavailability without the knowledge of an upper bound on the maximum delay. The impact of this estimate becomes negligible when Λ_T goes to infinity in Proposition 9. However, if the τ used in the algorithm is smaller than the actual maximum delay $\tau_{\max}$, the regret can be deteriorated by a factor of $\tau_{\max}/\tau$.*

5. A closer look at the decentralized case

In this section, we aim at providing a finer analysis of the decentralized case (typically case 2b of Section 2.2). The regret analyses that we presented previously focus on the *individual* losses of the agents (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. Moreover, this section is also the occasion to directly address the case of open networks where agents can join and depart the optimization process freely.

5.1 Decentralized Delayed Dual Averaging

We previously assumed that only one prediction was made at each time t while in many decentralized environments, multiple predictions can be made simultaneously.[11] 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. We also introduce the (root mean square) average number of active agents by $\overline{M} = \sqrt{(1/T)\sum_{t=1}^{T} M_t{}^2}$ and the maximum number of active agents by $M_{\max} = \max_{1 \le t \le T} M_t$.

Before proceeding with the algorithm, we first slightly extend the previously introduced notations and concepts to adapt to the current framework. The functions and the played points at time t are now respectively denoted by $f_{1,t}, \ldots, f_{M_t,t}$ and $x_{1,t}, \ldots, x_{M_t,t}$. 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 $s \in [t-1]$. Lastly, 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 $(j, s) \in \mathcal{S}_{i,t}$.

With these notations, the update of decentralized delayed dual averaging algorithm writes at time t for an agent i

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

where $g_{j,s} \in \partial f_{j,s}(x_{j,s})$.

5.2 Effective and collective regrets

By directly extending the usual notion of regret (Eq. (1)) to our current setup, we obtain the following regret:

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

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:

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

11. In reality, this does not affect the generality of our framework since t does not to have any actual physical meaning. In particular, if multiple events happen at the same moment we may endow them with an arbitrary order.

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.

This definition of *collective regret* recalls the usual regret formulation appearing in the distributed online optimization literature (Hosseini et al., 2013; Shahrampour and Jadbabaie, 2017; Yan et al., 2012) and 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.

Moreover, in a wireless sensor network, the nodes are typically equipped with very limited power supply. The reduction of energy consumption is thus crucial to extend the lifetime of sensor nodes. Transmitting all the sensed data to a sink node or too frequent communication should be avoided in order to achieve improved energy efficiency, and that is why a general framework that accounts for asynchronous communication in a decentralized network would be favorable. 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 10. *Assume that all the loss functions $f_{i,t}$ are G-Lipschitz; then,*

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

5.3 Collective regret with a fixed learning rate

In order to understand the mechanics of collective regret in our setup, we first consider the case of a fixed learning rate $\eta_{i,t} \equiv \eta$. 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 1 with $\eta_t \equiv \eta$ and thus

$$\mathbf{Reg}_T^\ell(u) \leq \frac{h(u)}{\eta} + \frac{1}{2}\sum_{m=1}^{N}\eta\left(\|g'_m\|_*^2 + 2\|g'_m\|_* \sum_{l\in[m-1]\setminus\mathcal{S}'_m}\|g'_l\|_*\right) \tag{6}$$

  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 $\operatorname{card}([m-1] \setminus \mathcal{S}'_m)$, we get

$$\mathbf{Reg}_T^\ell(u) \leq \frac{h(u)}{\eta} + \eta(\tau+1)G^2\sum_{t=1}^{T} M_t^2. \tag{7}$$

- 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 10.

Putting together these points we manage to show the following bound on the collective regret, the full proof being deferred to Appendix C.1.

Theorem 11. *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 theorem ensures a regret in $M\sqrt{T}$ which 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; Xiao, 2009; see also Appendix A.1).

5.4 More practical learning rates

Since the network of agents may be evolving, the average number of workers $\overline{M}$ may often not be available in advance. A first solution can be taking learning rates of the form $\eta_{i,t} = \eta_t = \Theta(1/M_{\max}\sqrt{\tau t})$. However, this still requires the knowledge of the global time t which is typically out of reach in the setup we are considering; in addition, it can be overly pessimistic with the dependence in $M_{\max}\sqrt{\tau}$. To overcome these issues, we base ourselves on Section 4 and show that a learning rate scheme similar to the one considered in Section 4.2.1 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* $\operatorname{card}(\mathcal{S}_{j,s}) < \operatorname{card}(\mathcal{S}_{i,t})$.

Under this assumption, we prove the following theorem which further extends the result of Proposition 8.

Theorem 12. *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*

$$\eta_{i,t} = \frac{R}{G\sqrt{(5\tau+3)(\operatorname{card}(\mathcal{S}_{i,t}) + (\tau+1)M_{\max})M_{\max}}} \tag{8}$$

guarantees a collective regret in

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

Sketch of proof. The proof is reported in Appendix C.2. It follows closely the schema introduced in Section 5.3 while two additional difficulties present: *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. ■

Note that this bound directly features the total number of actions taken in the full process; it is thus (at least partly) adaptive to the number of agents. However, we leave the design of data-dependent adaptive methods in this setting as an open question.

Remark 3. *From our analysis, we notice that all $f_{i,t}$ may not happen exactly at the same time. More generally, the time index t can stand for a time interval in a physical sense. In this case, it is possible to have instantaneous feedback (i.e., $g_{i,t} \in \mathcal{S}_{j,t}$ for some i, j) and a single physical agent can play several times during the period corresponding to t. In such situations, the same proof template can be readily applied.*

6. 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

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

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* $\widetilde{g}_{t+\frac{1}{2}} = \widetilde{g}_{t+\frac{1}{2}}(x_1, g_{\frac{3}{2}}, \ldots, g_{t-\frac{1}{2}})$. 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 (OGD) was shown (Chiang et al., 2012; Joulani et al., 2017; Mohri and Yang, 2016; Rakhlin and Sridharan, 2013) to be bounded by

$$\mathbf{Reg}_T(u) \leq \frac{\|u - x_1\|^2}{2\eta} + \sum_{t=1}^{T} \frac{\eta}{2} \|g_{t+\frac{1}{2}} - \widetilde{g}_{t+\frac{1}{2}}\|^2. \tag{9}$$

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.

6.1 Delayed Optimistic Dual averaging

While optimistic gradient descent (OGD) 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:[12]

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

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.

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 (DODA) 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 D.1.

Theorem 13. *Assume that the maximum delay is bounded by τ. Let delayed optimistic dual averaging (DODA) 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 $x_{\frac{3}{2}}, \ldots, x_{T+\frac{1}{2}}$) 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 13, we successfully show that (DODA) 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. (9)). 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 4. *The bounded delay assumption can in fact be relaxed in Theorem 13. Nonetheless, we choose to adopt this assumption for ease of understanding. Otherwise, denoting $d_t = \operatorname{card}(\mathcal{U}_t) + \operatorname{card}(\{s \in [T] : t \in \mathcal{U}_s\}) + 1$ 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 1.*

12. 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.

6.2 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.[13] 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).[14] 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 13.

Corollary 14. *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* (DODA) *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*

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

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 15. *Consider the setup of Corollary 14. 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* (DODA) *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 $o(\max(C_T^{\tau^+}, \sqrt{T}))$.*

Sketch of proof. The proof is reported in Appendix D.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 14). ∎

Finally, we also show that among all the online algorithms with the same prior information, the bound achieved in Corollary 14 is tight in its dependence on τ and $C_T^{\tau^+}$.

Proposition 16. *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 $\Omega(\sqrt{\tau \overline{C^\tau}})$.*

13. The optimistic step is also called *extrapolation* step to mirror the vocabulary of the extragradient method Korpelevich (1976).
14. For linear losses, the gradient does not depend on the calling point and thus $g_{t+\frac{1}{2}} = \nabla f_t(x_{t+\frac{1}{2}}) = g_t$.

Sketch of proof. The proof is reported in Appendix D.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 (DODA) *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 (DODA).

6.3 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[15] 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.*

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 17. *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* (DODA) *(evaluated at the points $x_{\frac{3}{2}}, \ldots, x_{T+\frac{1}{2}}$) satisfies*

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

Sketch of proof. The proof is immediate from Theorem 13 and is deferred to Appendix D.4. ■

Theorem 17 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.

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

Example 3 (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 4 (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 3, 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 17 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 3 and 4, 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 18. *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* $\|u - x_1\| \le R$*, running delayed optimistic dual averaging* (DODA) *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

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

Sketch of proof. The proof is deferred to Appendix D.5. Notice that the adaptive learning rates are not necessarily non-increasing and therefore Theorem 17 can not be directly applied. To address this challenge, we rely on the notions introduced in Section 4 and adapt both Theorem 13 and Theorem 17 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 5. *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.

7. Conclusion

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. We defer these questions to future work.

Acknowledgments

This work has been partially supported by MIAI Grenoble Alpes (ANR-19-P3IA-0003).

Appendix A. A general regret bound

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 and present several preliminary technical lemmas before going to the proof of the main results. For a thorough study of the algorithm the readers can refer to the textbook Hazan, 2016, Section 5 and Xiao, 2009.

A.1 Undelayed online dual averaging

Considering a sequence of first-order feedback $g_1, \ldots, g_T$, at time t dual averaging computes

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

We recall that the mirror map is defined as $P : y \mapsto \arg\min_{x\in\mathcal{X}} \langle -y, x\rangle + h(x)$. We can thus write $x_t = P(y_t)$ 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 19. *Let online dual averaging* (ODA) *be run with non-increasing learning rates* $(\eta_t)_{t\in[T]}$. *Then, the generated points* $x_1, \dots, x_T$ *satisfy*

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

To prove the above regret bound, we first fix $u \in \mathcal{X}$ and define the associated estimate sequence which is key to our analysis

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

Lemma 20. *Let* $(\eta_t)_{t\in\mathbb{N}}$ *be non-increasing. The functions* $(\psi_t)_{t\in\mathbb{N}}$ *satisfy the following properties:*

(*a*) $\psi_t(x_t) \le \dfrac{h(u)}{\eta_t} - \dfrac{\|x_t - u\|^2}{2\eta_t}$.

(*b*) $\psi_t(x_t) \le \psi_{t+1}(x_{t+1}) - \langle g_t, x_{t+1} - u \rangle - \dfrac{1}{\eta_t} D(x_{t+1}, x_t)$.

Proof. (*a*) The optimality condition of (ODA) implies that for all $x \in \mathcal{X}$, we have

$$\left\langle \sum_{s=1}^{t-1} g_s + \frac{\nabla h(x_t)}{\eta_t}, x - x_t \right\rangle \ge 0. \tag{10}$$

By substituting $x \leftarrow u$, the above gives,

$$\psi_t(x_t) \le \frac{1}{\eta_t}\langle \nabla h(x_t), u - x_t \rangle + \frac{h(x_t)}{\eta_t}. \tag{11}$$

We recall that h is 1-strongly convex with respect to $\|\cdot\|$ and therefore

$$h(u) \ge h(x_t) + \langle \nabla h(x_t), u - x_t \rangle + \frac{\|x_t - u\|^2}{2}. \tag{12}$$

Combining (11) and (12) leads to the desired inequality.

(*b*) On one hand, by $\eta_{t+1} \le \eta_t$ and the non-negativity of h

$$\psi_{t+1}(x_{t+1}) = \psi_t(x_{t+1}) + \langle g_t, x_{t+1} - u \rangle + \left(\frac{1}{\eta_{t+1}} - \frac{1}{\eta_t}\right) h(x_{t+1}) \ge \psi_t(x_{t+1}) + \langle g_t, x_{t+1} - u \rangle. \tag{13}$$

On the other hand, by substituting $x \leftarrow x_{t+1}$ in (10), we can write

$$\begin{aligned}
\psi_t(x_{t+1}) - \psi_t(x_t) &= \sum_{s=1}^{t-1} \langle g_s, x_{t+1} - x_t \rangle + \frac{h(x_{t+1})}{\eta_t} - \frac{h(x_t)}{\eta_t} \\
&\ge \frac{1}{\eta_t}\langle \nabla h(x_t), x_t - x_{t+1} \rangle + \frac{h(x_{t+1})}{\eta_t} - \frac{h(x_t)}{\eta_t} = \frac{1}{\eta_t} D(x_{t+1}, x_t).
\end{aligned} \tag{14}$$

We conclude by summing (13), (14) and rearranging the terms. ■

Proof of Proposition 19. Let $\eta_{T+1} = \eta_T$ and define x_{T+1} by (ODA) (We can do this since x_{T+1} is not used in the computation of $\mathbf{LinReg}_T$). Leveraging on Lemma 20, we bound the regret as follows:

$$
\begin{aligned}
\mathbf{LinReg}_T(u) &:= \sum_{t=1}^{T} \langle g_t, x_t - u \rangle \\
&= \sum_{t=1}^{T} \left(\langle g_t, x_t - x_{t+1} \rangle + \langle g_t, x_{t+1} - u \rangle \right) \\
&\leq \sum_{t=1}^{T} \left(\frac{\eta_t}{2} \|g_t\|^2 + \frac{\|x_t - x_{t+1}\|^2}{2\eta_t} + \psi_{t+1}(x_{t+1}) - \psi_t(x_t) - \frac{1}{\eta_t} D(x_{t+1}, x_t) \right) \\
&\leq \psi_{T+1}(x_{T+1}) - \psi_1(x_1) + \sum_{t=1}^{T} \left(\frac{\eta_t}{2} \|g_t\|^2 + \frac{\|x_t - x_{t+1}\|^2}{2\eta_t} - \frac{\|x_t - x_{t+1}\|^2}{2\eta_t} \right) \\
&\leq \frac{h(u)}{\eta_T} + \frac{1}{2} \sum_{t=1}^{T} \eta_t \|g_t\|_*^2. \qquad (15)
\end{aligned}
$$

In the last inequality we use Lemma 20(a) along with $\eta_{T+1} = \eta_T$ and the fact that $\psi_1(x_1) = h(x_1)/\eta_1 \geq 0$. The second to last inequality holds thanks to the 1-strong convexity of h. (15) is exactly what we want to prove, so this ends the proof. ■

A.2 Technical preliminaries

We prove below the non-expansiveness of the mirror map (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.,* $\|P(y) - P(y')\| \leq \|y - y'\|_*$ *for all* $y, y' \in \mathbb{R}^d$.[16]

Proof. Let $x = P(y)$ and $x' = P(y')$. 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 for all $\hat{x} \in \mathcal{X}$, the following two inequalities hold

$$
\langle -y + \nabla h(x), \hat{x} - x \rangle \geq 0, \quad \langle -y' + \nabla h(x'), \hat{x} - x' \rangle \geq 0.
$$

Substituting respectively $\hat{x} \leftarrow x'$ and $\hat{x} \leftarrow x$ in the two inequalities and summing the resulting formulas leads to

$$
\langle -y + \nabla h(x) + y' - \nabla h(x'), x' - x \rangle \geq 0.
$$

To conclude, we use the Cauchy–Schwarz inequality and the 1-strong convexity of h with respect to $\|\cdot\|$.

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

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

16. Precisely, P 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).

A.3 Proof of Theorem 7

Theorem 7. *Let σ be a faithful permutation of $\{1,\ldots,T\}$. Assume that delayed dual averaging (DDA) is run with a learning rate sequence $(\eta_t)_{t\in[T]}$ such that $\eta_{\sigma(t+1)} \le \eta_{\sigma(t)}$ for all t. Then the generated points $x_1,\ldots,x_T$ enjoy the regret bound*

$$\mathbf{Reg}_T(u) \le \frac{h(u)}{\eta_{\sigma(T)}} + \frac{1}{2}\sum_{t=1}^{T}\eta_{\sigma(t)}\left(\|g_{\sigma(t)}\|_*^2 + 2\|g_{\sigma(t)}\|_* \sum_{s\in\{\sigma(1),\ldots,\sigma(t-1)\}\setminus\mathcal{S}_{\sigma(t)}} \|g_s\|_*\right).$$

Proof. Thanks to the convexity of the loss functions, we first bound our regret by its linearized counterpart

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

To proceed, we introduce the virtual iterate

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

and decompose the sum as:

$$\sum_{t=1}^{T}\langle g_t, x_t - u\rangle = \underbrace{\sum_{t=1}^{T}\langle g_t, \widetilde{x}_{\sigma^{-1}(t)} - u\rangle}_{(a)} + \underbrace{\sum_{t=1}^{T}\langle g_t, x_t - \widetilde{x}_{\sigma^{-1}(t)}\rangle}_{(b)}. \tag{16}$$

(a) The first term can be bounded using Proposition 19 with the virtual time order $\sigma(1),\ldots,\sigma(T)$. Indeed,

$$\sum_{t=1}^{T}\langle g_t, \widetilde{x}_{\sigma^{-1}(t)} - u\rangle = \sum_{t=1}^{T}\langle g_{\sigma(t)}, \widetilde{x}_t - u\rangle \le \frac{h(u)}{\eta_{\sigma(T)}} + \frac{1}{2}\sum_{t=1}^{T}\eta_{\sigma(t)}\|g_{\sigma(t)}\|_*^2 \tag{17}$$

where we used our assumption on the learning rate sequence ($\eta_{\sigma(t+1)} \le \eta_{\sigma(t)}$) to apply Proposition 19.

(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 $t \in \{1,\ldots,T\}$). To this end, we define

$$\mathcal{T}_t^\sigma = \{\sigma(1),\ldots,\sigma(t)\}, \quad \mathcal{U}_t^\sigma = \mathcal{T}_{t-1}^\sigma \setminus \mathcal{S}_{\sigma(t)}.$$

We can then write

$$x_{\sigma(t)} = P(-\eta_{\sigma(t)}\sum_{s\in\mathcal{S}_{\sigma(t)}} g_s), \quad \widetilde{x}_t = P(-\eta_{\sigma(t)}\sum_{s\in\mathcal{T}_{t-1}^\sigma} g_s).$$

By the non-expansivity of the mirror map (Lemma 21), we get

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

Note that since the permutation σ is decent, we have $\mathcal{S}_{\sigma(t)} \subset \{\sigma(1), .., \sigma(t-1)\} = \mathcal{T}_{t-1}^{\sigma}$ so the above formula is indeed verified. Subsequently,

$$\begin{aligned}\sum_{t=1}^{T}\langle g_t, x_t - \widetilde{x}_{\sigma^{-1}(t)}\rangle &= \sum_{t=1}^{T}\langle g_{\sigma(t)}, x_{\sigma(t)} - \widetilde{x}_t\rangle \\ &\leq \sum_{t=1}^{T}\|g_{\sigma(t)}\|_*\|x_{\sigma(t)} - \widetilde{x}_t\| \\ &\leq \sum_{t=1}^{T}\eta_{\sigma(t)}\|g_{\sigma(t)}\|_* \sum_{s\in\mathcal{U}_t^{\sigma}}\|g_s\|_*. \end{aligned} \tag{18}$$

Combining (16), (17) and (18), we obtain the desired result. ■

Appendix B. Proofs for variable learning rate methods

B.1 Introduction

The following standard lemma (see e.g., Auer et al., 2002, Lemma 3.5) is useful for proving the regret guarantees of adaptive methods.

Lemma 22. *For any real numbers $\lambda_1, \ldots, \lambda_T$ such that $\sum_{s=1}^{t}\lambda_s > 0$ for all $t \in [T]$, it holds*

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

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 $z \geq 0$,

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

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 $\sqrt{\lambda_1} \leq 2\sqrt{\lambda_1}$. ■

Lemma 22 is mainly used to produce efficient learning rates by exploiting the structure of the lag. We recall that the lag is defined as

$$\Lambda_t = \sum_{s=1}^{t}\underbrace{\left(\|g_s\|_*^2 + 2\|g_s\|_*\sum_{l\in\mathcal{U}_s}\|g_l\|_*\right)}_{\lambda_s} = \sum_{s=1}^{t}\|g_s\|_*^2 + 2\sum_{\{s,l\}\in\mathcal{D}_t}\|g_s\|_*\|g_l\|_* \tag{19}$$

with $\mathcal{D}_t \coloneqq \{\{s,l\} : s \in [t], l \in \mathcal{U}_s\}$. Combining Lemma 22 and Theorem 1, we have immediately the following result.

Lemma 23. *Let $(\overline{\Lambda}_t)_{t\in[T]}$ be a sequence of non-decreasing numbers satisfying $\overline{\Lambda}_t \geq \Lambda_t$ for all t. We fix $R > 0$ and $u \in \mathcal{X}$ such that $h(u) \leq R^2$. Then, Running* (DDA) *with $\eta_t = R/\sqrt{\overline{\Lambda}_t}$, we have*

$$\mathbf{Reg}_T(u) \leq 2R\sqrt{\overline{\Lambda}_T}\,.$$

Proof. Since $(\overline{\Lambda}_t)_{t\in[T]}$ is non-decreasing, we can apply Theorem 1. Then

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

In particular, the second to last inequality holds thanks to Lemma 22. ■

From the above lemma Proposition 3 follows readily (and in fact we can replace the maximum delay τ by the maximum unavailability ν). As for data-dependent adaptive methods, we need to deepen our understanding of the different quantities that are involved in the regret bound and the algorithms. For the sake of illustration, we make the non-decreasing active feedback assumption so that the set $\mathcal{G}_t$ collecting timestamps of the feedback arriving before g_t can be defined. We also recall the definition $\mathcal{A}_t = \{\{s,l\} : s \in \mathcal{S}_t, l \in \mathcal{G}_s \setminus \mathcal{S}_s\}$. The following characterization of $\mathcal{D}_t$ and $\mathcal{A}_t$ are crucial to our analysis.

Proposition 24. *Both the sets $\mathcal{D}_t$ and $\mathcal{A}_t$ can be uniquely characterized by the condition $\{s \notin \mathcal{S}_l, l \notin \mathcal{S}_s\}$ with an additional constraint on the elements that a pair can take. Precisely,*

$$(a)\ \mathcal{D}_t = \left\{\{s,l\} \in \binom{[t]}{2} : s \notin \mathcal{S}_l, l \notin \mathcal{S}_s\right\}; \quad (b)\ \mathcal{A}_t = \left\{\{s,l\} \in \binom{\mathcal{S}_t}{2} : s \notin \mathcal{S}_l, l \notin \mathcal{S}_s\right\}.$$

For a set $\mathcal{S}$, we denote by $\binom{\mathcal{S}}{2} = \{\{a,b\} \in 2^{\mathcal{S}} : a \neq b\}$ the set of all subsets of size 2 of $\mathcal{S}$.

Proof. (a) For the inclusion we just need to notice that since $l < s$ we necessarily have $l \in [t]$ and $s \notin \mathcal{S}_l$. For the reverse inclusion we can suppose $l < s$ without loss of generality from which we deduce immediately $l \in \mathcal{U}_s$.

(b) $\supset$: We follow the same line of reasoning. Since either $\sigma^{-1}(l) < \sigma^{-1}(s)$ (i.e., $l \in \mathcal{G}_s$) or $\sigma^{-1}(s) < \sigma^{-1}(l)$ (i.e., $s \in \mathcal{G}_l$), we may assume $\sigma^{-1}(l) < \sigma^{-1}(s)$ and it follows $l \in \mathcal{G}_s \setminus \mathcal{S}_s$.

$\subset$: Let $\{s,l\}$ be an unordered pair such that $s \in \mathcal{S}_t$ and $l \in \mathcal{G}_s \setminus \mathcal{S}_s$. By the definition of σ, there exists k such that $\mathcal{S}_t = \{\sigma(1),\ldots,\sigma(k)\}$. From $s \in \mathcal{S}_t$ we get $\sigma^{-1}(s) \leq k$ and from $l \in \mathcal{G}_s$ we know that $\sigma^{-1}(l) < \sigma^{-1}(s)$. Combining the two we deduce $\sigma^{-1}(l) < k$ and hence $l \in \mathcal{S}_t$. Moreover, as $t \notin \mathcal{S}_t$, we have $k < \sigma^{-1}(t)$, or equivalently $\mathcal{S}_t \subset \mathcal{G}_t$ (notice that this indeed proves that σ *is a faithful permutation*). The same applies to $\mathcal{S}_l$, that is, $\mathcal{S}_l \subset \mathcal{G}_l$. We now conclude by arguing that $l \in \mathcal{G}_s \implies s \notin \mathcal{G}_l \implies s \notin \mathcal{S}_l$. ■

Proposition 24 reveals the fundamental relation between $\mathcal{D}_t$ and $\mathcal{A}_t$, at the heart of which is the unique condition $\{s \notin \mathcal{S}_l, l \notin \mathcal{S}_s\}$. This condition, being independent of how the time indices are ordered, is inherent to the underlying learning process as we will see again in Proposition 25.

Algorithm 4 AdaDelay–O+

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

B.2 The single-agent setup: AdaDelay–O and AdaDelay–O+

Thanks to Proposition 24, it is now clear that AdaDelay–O+ consists, in fact, of a simple strategy: we replace the terms appearing in Λ_t by their actual values whenever possible and majorize each of the remaining terms by G^2. Algorithm 4 implements the method in a way that we only need to maintain a minimal number of quantities for the update of η_t (the use of subtraction in line 5 is justified by the inclusion $\mathcal{S}_t \subset \mathcal{G}_t$ proved in the proof of Proposition 24). Let us proceed to prove the regret bound of AdaDelay–O+.

Proposition 5. *Assume that the norms of the gradients are bounded by G and the sequence of active feedback is non-decreasing, i.e., $\mathcal{S}_t \subset \mathcal{S}_{t+1}$. Assume further that delayed dual averaging (DDA) is run with the learning rate sequence*

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

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

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

Proof. Let $\overline{\Lambda}_t = R^2/{\eta_t}^2$ so that $\eta_t = R/\sqrt{\overline{\Lambda}_t}$. $(\overline{\Lambda}_t)_{t\in[T]}$ is non-decreasing by the definition of η_t. Moreover, $\overline{\Lambda}_t \ge \Gamma_t + \tilde{\tau}_t G^2$. We note that

$$\Gamma_t = \sum_{s\in\mathcal{S}_t}\|g_s\|_*^2 + 2\sum_{\{s,l\}\in\mathcal{A}_t}\|g_s\|_*\|g_l\|_*,$$

$$\Lambda_t = \sum_{s=1}^{t}\|g_s\|_*^2 + 2\sum_{\{s,l\}\in\mathcal{D}_t}\|g_s\|_*\|g_l\|_*.$$

From Proposition 24 we have $\mathcal{A}_t \subset \mathcal{D}_t$ (as $\mathcal{S}_t \subset [t]$). Using the definition of $\tilde{\tau}_t$ and the bounded subgradient assumption it is clear that $\Gamma_t + \tilde{\tau}_t G^2 \geq \Lambda_t$ and accordingly $\overline{\Lambda}_t \geq \Lambda_t$. With $\overline{\Lambda}_T = \max_{1 \leq t \leq T} \Gamma_t + \tilde{\tau}_t G^2$, applying Lemma 23 readily gives the first inequality. For the second inequality, we use both $\Gamma_t \leq \Lambda_t$ and $\Gamma_t \leq (\operatorname{card}(\mathcal{S}_t) + 2\operatorname{card}(\mathcal{A}_t))G^2$. ∎

When (an upper bound of) the maximum delay is known, it is possible to further bound $\tilde{\tau}$ from above, and this leads to AdaDelay–O.

Proposition 4. *Assume that the maximum delay is bounded by τ, the norm of the gradients are bounded by G, and the sequence of active feedback is non-decreasing, i.e., $\mathcal{S}_t \subset \mathcal{S}_{t+1}$. Let delayed dual averaging (DDA) be run with a learning rate sequence*

$$\eta_t = \frac{R}{\sqrt{\Gamma_t + G^2(2\tau^2 + 3\tau + 1)}}. \tag{AdaDelay–O}$$

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

$$\mathbf{Reg}_T(u) \leq 2R\sqrt{\Gamma_T + G^2(2\tau^2 + 3\tau + 1)} \leq 2R\sqrt{\Lambda_T + G^2(2\tau^2 + 3\tau + 1)}.$$

Proof. It is sufficient to show that $\tilde{\tau} \leq 2\tau^2 + 3\tau + 1$. To begin, we have $t - \operatorname{card}(\mathcal{S}_t) \leq \tau + 1$ as $[t - \tau - 1] \subset \mathcal{S}_t$. To proceed, let us consider a pair $\{s, l\} \in \mathcal{D}_t \setminus \mathcal{A}_t$. In particular, we know that $\{s, l\} \not\subset \mathcal{S}_t$, so we have either $s \in \{t - \tau, \ldots, t\}$ or $l \in \{t - \tau, \ldots, t\}$. Without loss of generality, we suppose $s < l$, then $l \in \{t - \tau, \ldots, t\}$. By definition of $\mathcal{D}_t$ we have $s \notin \mathcal{S}_l$, and thus $s \in \{l - \tau, \ldots, l - 1\}$. This shows $\operatorname{card}(\mathcal{D}_t \setminus \mathcal{A}_t) \leq \tau(\tau + 1)$. We can therefore conclude $\tilde{\tau}_t \leq 2\tau(\tau + 1) + \tau + 1 = 2\tau^2 + 3\tau + 1$. ∎

As discussed in Section 3.3.3, the bound of AdaDelay–O is always worse than that of AdaDelay–O+. Therefore, we believe that AdaDelay–O+ should always be preferred to AdaDelay–O and it is particularly in the decentralized setup the bounded delay assumption becomes of importance (see also Remark 2).

B.3 The distributed setup

We recall the following notations from Appendix A.3:

$$\mathcal{T}_t^\sigma = \{\sigma(1), \ldots, \sigma(t)\}, \quad \mathcal{U}_t^\sigma = \mathcal{T}_{t-1}^\sigma \setminus \mathcal{S}_{\sigma(t)}.$$

By analogy with Λ_t, we define

$$\Lambda_t^\sigma = \sum_{s=1}^{t} \underbrace{\left(\|g_{\sigma(s)}\|_*^2 + 2\|g_{\sigma(s)}\|_* \sum_{l \in \mathcal{U}_s^\sigma} \|g_l\|_* \right)}_{:=\lambda_s^\sigma} = \sum_{s \in \mathcal{T}_t^\sigma} \|g_s\|_*^2 + 2 \sum_{\{s,l\} \in \mathcal{D}_t^\sigma} \|g_s\|_* \|g_l\|_*, \tag{20}$$

where $\mathcal{D}_t^\sigma := \{\{\sigma(s), l\} : s \in [t], l \in \mathcal{U}_s^\sigma\}$. Using Theorem 7 and Lemma 22, we can adapt Lemma 23 to take into account a faithful permutation σ. The proof is omitted for it being almost the same as the proof of Lemma 23.

Lemma 23′. *Let σ be a faithful permutation and let $(\overline{\Lambda}_t)_{t\in[T]}$ be a sequence of real numbers satisfying $\overline{\Lambda}_{\sigma(t+1)} \geq \overline{\Lambda}_{\sigma(t)}$ and $\overline{\Lambda}_{\sigma(t)} \geq \Lambda^{\sigma}_t$ for all t. We fix $R > 0$ and $u \in \mathcal{X}$ such that $h(u) \leq R^2$. Then, Running (DDA) with $\eta_t = R/\sqrt{\overline{\Lambda}_t}$, we have*

$$\mathbf{Reg}_T(u) \leq 2R\sqrt{\overline{\Lambda}_{\sigma(T)}}.$$

As a first instantiation of Lemma 23′, we prove the following stronger variant of Proposition 8 where we replace the maximum delay by the maximum unavailability (from which Proposition 8 is deduced immediately).

Proposition 8′. *Assume that the maximum unavailability is bounded by ν, the norm of the gradients are bounded by G, and let Assumption 1 hold. Let delayed dual averaging (DDA) be run with a learning rate sequence*

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

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

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

Proof. Let $\overline{\Lambda}_t = G^2(1+2\nu)(\mathrm{card}(\mathcal{S}_t)+\nu+1)$. 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. Assume otherwise, that is, $\overline{\Lambda}_{\sigma(t+1)} < \overline{\Lambda}_{\sigma(t)}$ for some t. Then $t+1 = \sigma^{-1}(\sigma(t+1)) < \sigma^{-1}(\sigma(t)) = t$, a contradiction.

We proceed to prove $\mathrm{card}(\mathcal{U}^{\sigma}_t) \leq \nu$, or equivalently $\mathrm{card}(\mathcal{S}_{\sigma(t)}) \geq t-1-\nu$. For this we show $\mathcal{T}^{\sigma}_t \subset [\mathrm{card}(\mathcal{S}_{\sigma(t)})+\nu+1]$. 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 $\mathrm{card}([\sigma(s)-1] \setminus \mathcal{S}_{\sigma(s)}) \leq \nu$ so that $\sigma(s)-1-\mathrm{card}(\mathcal{S}_{\sigma(s)}) \leq \nu$ and subsequently $\sigma(s) \leq \mathrm{card}(\mathcal{S}_{\sigma(t)})+\nu+1$. This proves $\mathcal{T}^{\sigma}_t \subset [\mathrm{card}(\mathcal{S}_{\sigma(t)})+\nu+1]$.

From $\mathrm{card}(\mathcal{U}^{\sigma}_t) \leq \nu$ it follows immediately $\lambda^{\sigma}_t \leq G^2(1+2\nu)$ for all t. Along with $t \leq \mathrm{card}(\mathcal{S}_{\sigma(t)})+\nu+1$ we deduce $\Lambda^{\sigma}_t \leq G^2(1+2\nu)(\mathrm{card}(\mathcal{S}_{\sigma(t)})+\nu+1) = \overline{\Lambda}_{\sigma(t)}$. Applying Lemma 23′ and using the fact that $\mathrm{card}(\mathcal{S}_t) \leq T-1$ for all t gives the desired result. ■

AdaGrad-style As for the analysis of AdaDelay–Dist, we shall relate the lag Λ^{σ}_t to its estimate Γ^{Dist}_t. To this end, we consider the set $\mathcal{A}_{i,t} \coloneqq \{\{s,l\} : s \in \mathcal{S}_t, l \in \mathcal{G}_{i,s} \setminus \mathcal{S}_s\}$ so that

$$\Gamma^{\mathrm{Dist}}_t = \sum_{s\in\mathcal{S}_t}\left(\|g_s\|_*^2 + 2\|g_s\|_* \sum_{l\in\mathcal{G}_{i(t),s}\setminus\mathcal{S}_s}\|g_l\|_*\right) = \sum_{s\in\mathcal{S}_t}\|g_s\|_*^2 + 2\sum_{\{s,l\}\in\mathcal{A}_{i(t),t}}\|g_s\|_*\|g_l\|_* \tag{21}$$

To simplify the notation, we will write $\mathcal{A}^{\mathrm{Dist}}_t = \mathcal{A}_{i(t),t}$. In the same vein as Proposition 24, we can characterize both $\mathcal{D}^{\sigma}_t$ and $\mathcal{A}^{\mathrm{Dist}}_t$ by the unique condition $\{s \notin \mathcal{S}_l, l \notin \mathcal{S}_s\}$.

Proposition 25. *Let σ be a faithful permutation and let Assumption 2 hold. Then*

$$(a)\ \mathcal{D}^{\sigma}_t = \left\{\{s,l\} \in \binom{\mathcal{T}^{\sigma}_t}{2} : s \notin \mathcal{S}_l, l \notin \mathcal{S}_s\right\}; \quad (b)\ \mathcal{A}^{\mathrm{Dist}}_t = \left\{\{s,l\} \in \binom{\mathcal{S}_t}{2} : s \notin \mathcal{S}_l, l \notin \mathcal{S}_s\right\}.$$

Proof. (a) $\subset$: 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 $l \in \mathcal{T}_{s-1}^\sigma \subset \mathcal{T}_t^\sigma$. It remains to prove that $\sigma(s) \notin \mathcal{S}_l$. We exploit the equivalence

$$l \in \mathcal{T}_{s-1}^\sigma \iff \sigma^{-1}(l) \leq s-1 \iff \sigma^{-1}(l) < \sigma^{-1}(\sigma(s)) \iff \sigma(s) \notin \mathcal{T}_{\sigma^{-1}(l)}^\sigma. \tag{22}$$

To conclude, we use the fact that σ is a faithful permutation and accordingly $\mathcal{S}_l \subset \mathcal{T}_{\sigma^{-1}(l)-1}^\sigma \subset \mathcal{T}_{\sigma^{-1}(l)}^\sigma$. Along with (22) we infer that $\sigma(s) \notin \mathcal{S}_l$.

$\supset$: Let $\{s,l\} \in \binom{\mathcal{T}_t^\sigma}{2}$ 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 $s \in \mathcal{T}_t^\sigma$ if and only if $\sigma^{-1}(s) \in [t]$.

(b) By restricting ourselves to the agent $i(t)$, we can apply the arguments that proved Proposition 24(b). In more detail:

$\subset$: Let $s \in \mathcal{S}_t$ and $l \in \mathcal{G}_{i(t),s} \setminus \mathcal{S}_s$. The inclusion $l \in \mathcal{G}_{i(t),s}$ 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, $l \in \mathcal{G}_{i(t),s}$ also implies $s \notin \mathcal{G}_{i(t),l}$. Using Assumption 2 we know that $\mathcal{S}_l \subset \mathcal{G}_{i(t),l}$, and consequently $s \notin \mathcal{S}_l$.

$\supset$: Let $\{s,l\} \in \binom{\mathcal{S}_t}{2}$ such that $s \notin \mathcal{S}_l$ and $l \notin \mathcal{S}_s$. Since either $l \in \mathcal{G}_{i(t),s}$ or $s \in \mathcal{G}_{i(t),l}$ (but not both) we conclude immediately $\{s,l\} \in \mathcal{A}_{i(t),t} = \mathcal{A}_t^{\text{Dist}}$. ■

Thanks to Proposition 25, comparing $\mathcal{D}_t^\sigma$ with $\mathcal{A}_{\sigma(t)}^{\text{Dist}}$ amounts to comparing $\mathcal{T}_t^\sigma$ with $\mathcal{S}_{\sigma(t)}$. This incites us to study in more detail what the bounded delay assumption would allow us to say on a faithful permutation. We prove the following.

Proposition 26. *Let σ be a faithful permutation and assume that the maximum delay is bounded by τ. We have (a) $|\sigma(t) - t| \leq \tau$ and (b) $\mathcal{T}_t^\sigma \setminus \mathcal{S}_{\sigma(t)} \subset \{\sigma(t)-\tau, \dots, \sigma(t)+\tau\}$.*

Proof. Let $s,t \in [T]$ such that $s \leq t$. We claim that $\sigma(s) \leq \sigma(t) + \tau$. 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. Using the above, we deduce the two properties of σ quite easily.

(a) 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 $t \leq \sigma(t) + \tau$. In the same way, we prove $\sigma(t) - \tau \leq t$. Combining the two we conclude $|\sigma(t) - t| \leq \tau$.

(b) The bounded delay assumption implies $[\sigma(t) - \tau - 1] \subset \mathcal{S}_{\sigma(t)}$. On the other hand, $T_t^\sigma = \{\sigma(1), \dots, \sigma(t)\} = \{\sigma(s) : s \leq t\}$ and hence $T_t^\sigma \subset [\sigma(t) + \tau]$. Putting the two inclusions together we get effectively $\mathcal{T}_t^\sigma \setminus \mathcal{S}_{\sigma(t)} \subset \{\sigma(t) - \tau, \dots, \sigma(t) + \tau\}$. ■

Interestingly, Proposition 26(a) shows that under the bounded delay assumption, a faithful permutation can at most move an element τ steps away from its original position. On the other hand, property (b) forms the last building block of the proof of Proposition 9.

Proposition 9. *Assume that the maximum delay is bounded by τ, the norm of the gradients are bounded by G, and let Assumption 2 hold. Let delayed dual averaging* (DDA) *be run with a learning rate sequence*

$$\eta_t = \frac{R}{\sqrt{\Gamma_t^{\text{Dist}} + G^2(2\tau+1)^2}}. \tag{AdaDelay–Dist}$$

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

$$\mathbf{Reg}_T(u) \leq 2R \max_{1 \leq t \leq T} \sqrt{\Gamma_t^{\text{Dist}} + G^2(2\tau+1)^2} \leq 2R\sqrt{\Lambda_T + G^2(2\tau+1)^2}.$$

Proof. We take $\overline{\Lambda}_t = \Gamma_t^{\text{Dist}} + G^2(2\tau+1)^2$ and define a permutation σ 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 e.g., proof of Proposition 8′). 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 $\mathcal{S}_s \subset \mathcal{G}_{i(t),s}$ and from $s \in \mathcal{S}_t$ it holds $\mathcal{G}_{i(t),s} \subset \mathcal{S}_t$; accordingly, $\mathcal{S}_s \subset \mathcal{S}_t$. Invoking Proposition 25(b) we deduce $\mathcal{A}_s^{\text{Dist}} \subset \mathcal{A}_t^{\text{Dist}}$. Using (21) we then get $\overline{\Lambda}_s \leq \overline{\Lambda}_t$. This inequality along with $s \in \mathcal{S}_t$ imply $\sigma^{-1}(s) < \sigma^{-1}(t)$.

To apply Lemma 23′, we still need to prove $\overline{\Lambda}_{\sigma(t)} \geq \Lambda_t^{\sigma}$. Combing (20), (21), Proposition 25 and $\mathcal{S}_{\sigma(t)} \subset \mathcal{T}_t^{\sigma}$ we know that

$$\Lambda_t^{\sigma} = \Gamma_{\sigma(t)}^{\text{Dist}} + \sum_{s \in \mathcal{T}_t^{\sigma} \setminus \mathcal{S}_{\sigma(t)}} \|g_s\|_*^2 + 2 \sum_{\{s,l\} \in \mathcal{D}_t^{\sigma} \setminus \mathcal{A}_{\sigma(t)}^{\text{Dist}}} \|g_s\|_* \|g_l\|_*. \tag{23}$$

In particular, $\Gamma_{\sigma(t)}^{\text{Dist}} \leq \Lambda_t^{\sigma} \leq \Lambda_T^{\sigma} = \Lambda_T$, which proves the second inequality that appears in the regret bound (for the equality $\Lambda_T^{\sigma} = \Lambda_T$, we use (19), (20), Proposition 24(a), Proposition 25(a) and $\mathcal{T}_T^{\sigma} = [T]$). The end of our proof heavily relies on the use of Proposition 26(b). The first immediate consequence is that $\operatorname{card}(\mathcal{T}_t^{\sigma} \setminus \mathcal{S}_{\sigma(t)}) \leq 2\tau+1$ and therefore the second term of (23) can be bounded by $G^2(2\tau+1)$. As for $\{s,l\} \in \mathcal{D}_t^{\sigma} \setminus \mathcal{A}_{\sigma(t)}^{\text{Dist}}$, invoking Proposition 25 and Proposition 26(b) we know that either $s \in \{\sigma(t)-\tau, \dots, \sigma(t)+\tau\}$ or $l \in \{\sigma(t)-\tau, \dots, \sigma(t)+\tau\}$ and $\max(s,l) \leq \sigma(t)+\tau$. Without loss of generality, we suppose $l < s$, then $s \in \{\sigma(t)-\tau, \dots, \sigma(t)+\tau\}$. From $l \notin \mathcal{S}_s$ and the bounded delay assumption we further deduce $l \in \{s-\tau, \dots, s-1\}$. We can thus conclude $\operatorname{card}(\mathcal{D}_t^{\sigma} \setminus \mathcal{A}_{\sigma(t)}^{\text{Dist}}) \leq (2\tau+1)\tau$ and finally $\Lambda_t^{\rho} \leq \Gamma_{\sigma(t)}^{\text{Dist}} + G^2(2\tau+1)^2$. This proves $\overline{\Lambda}_{\sigma(t)} \geq \Lambda_t^{\sigma}$ and applying Lemma 23′ gives the desired result. ∎

Appendix C. Proofs related to the collective regret

Both the proofs of Theorem 11 and Theorem 12 leverage on Lemma 10, which we recall below.

Lemma 10. *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 5.3, 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 $\mathcal{S}'_{\phi(i,t)} = \{\phi(j,s) : (j,s) \in \mathcal{S}_{i,t}\}$.

C.1 Fixed learning rate

Theorem 11. *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 (6). Since the loss functions are G-Lipschitz, the subgradients are bounded by G.

$$\begin{aligned}\mathbf{Reg}_T^\ell(u) &\leq \frac{h(u)}{\eta} + \frac{1}{2}\sum_{m=1}^{N} \eta \left(\|g'_m\|_*^2 + 2\|g'_m\|_* \sum_{l\in[m-1]\setminus\mathcal{S}'_m} \|g'_l\|_* \right) \\ &\leq \frac{h(u)}{\eta} + \frac{\eta}{2}\sum_{m=1}^{N}(1 + 2\,\mathrm{card}([m-1]\setminus\mathcal{S}'_m))G^2. \end{aligned} \tag{24}$$

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

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

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

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

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

$$\sum_{m=N_{t-1}+1}^{N_t} \mathrm{card}([m-1]\setminus\mathcal{S}'_m) \leq \frac{M_t(M_t-1)}{2} + M_t\sum_{s=1}^{\tau} M_{t-s} \leq \frac{(\tau+1)}{2}M_t^2 + \frac{1}{2}\sum_{s=1}^{\tau} M_{t-s}^2. \tag{25}$$

Substituting (25) in (24) then yields

$$\mathbf{Reg}_T^\ell(u) \leq \frac{h(u)}{\eta} + \eta(\tau+1)G^2\sum_{t=1}^{T} M_t^2. \tag{26}$$

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 $x_{i,t} = P(-y_{i,t})$ and $x_{j,t} = P(-y_{j,t})$ 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

$$\sum_{i=1}^{M_t} G\|x_{i,t} - x_{j,t}\| \leq \eta G^2 M_t \sum_{s=1}^{\tau} M_{t-s} \leq \eta G^2 \left(\frac{\tau M_t^2}{2} + \frac{1}{2}\sum_{s=1}^{\tau} M_{t-s}^2 \right). \tag{27}$$

With (26) and (27), invoking Lemma 10 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. ∎

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

Theorem 12. *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 $h(u) \le R^2$, decentralized delayed dual averaging (D-DDA) with stepsizes*

$$\eta_{i,t} = \frac{R}{G\sqrt{(5\tau+3)(\mathrm{card}(\mathcal{S}_{i,t}) + (\tau+1)M_{\max})M_{\max}}} \tag{8}$$

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 Section 4.

Let us 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 $\mathrm{card}(\mathcal{U}^\sigma_{\sigma^{-1}(i,t)}) \le (\tau+1)M_{\max}$ (where $\mathcal{U}^\sigma_{\sigma^{-1}(i,t)} = \mathcal{T}^\sigma_{\sigma^{-1}(i,t)-1} \setminus \mathcal{S}_{i,t}$). Let $s \in \{0,\dots,\tau\}$ such that $N_{t+s-\tau} > \mathrm{card}(\mathcal{S}_{i,t}) \ge N_{t+s-\tau-1}$. Then for any $j \in [M_{t+s+1}]$ it holds $\mathrm{card}(\mathcal{S}_{j,t+s+1}) \ge N_{t+s-\tau} > \mathrm{card}(\mathcal{S}_{i,t})$ 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 $l \le t+s$, and subsequently $\mathrm{card}(\mathcal{T}^\sigma_{\sigma^{-1}(i,t)-1}) \le N_{t+s}$. We have therefore

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

Since $\eta_{i,t} \le \eta_{j,s}$ if and only if $\mathrm{card}(\mathcal{S}_{i,t}) \ge \mathrm{card}(\mathcal{S}_{j,s})$, we have indeed $\eta_{\sigma((i,t)+1)} \le \eta_{\sigma(i,t)}$. Invoking Theorem 7, one has (notice that the sum is ordered differently as stated in the theorem)

$$\begin{aligned}
\mathbf{Reg}_T^\ell(u) &\le \frac{h(u)}{\eta_{\sigma(M_T,T)}} + \frac{1}{2}\sum_{t=1}^{T}\sum_{i=1}^{M_t}\eta_{i,t}\left(\|g_{i,t}\|_*^2 + 2\|g_{i,t}\|_* \sum_{s\in\mathcal{U}^\sigma_{\sigma^{-1}(i,t)}} \|g_s\|_*\right)\\
&\le \frac{h(u)}{\min_{t\in[T],i\in[M_t]}\eta_{i,t}} + \frac{1}{2}\sum_{t=1}^{T}\left(\max_{i\in[M_t]}\eta_{i,t}\right)G^2(2\tau+3)M_t M_{\max}.
\end{aligned} \tag{28}$$

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 Theorem 11, we write $x_{i,t} = P(-y_{i,t})$ and $x_{j,t} = P(-y_{j,t})$ 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

$$\begin{aligned}
\|y_{i,t} - y_{j,t}\| &= \|(\eta_{i,t} - \eta_{j,t})\sum_{(k,s)\in\mathcal{S}_\cap} g_{k,s} + \eta_{i,t}\sum_{(k,s)\in\mathcal{S}_{i,t}\setminus\mathcal{S}_\cap} g_{k,s} - \eta_{j,t}\sum_{(k,s)\in\mathcal{S}_{j,t}\setminus\mathcal{S}_\cap} g_{k,s}\|\\
&\le |\eta_{i,t} - \eta_{j,t}|\sum_{(k,s)\in\mathcal{S}_\cap}\|g_{k,s}\| + \eta_{i,t}\sum_{(k,s)\in\mathcal{S}_{i,t}\setminus\mathcal{S}_\cap}\|g_{k,s}\| + \eta_{j,t}\sum_{(k,s)\in\mathcal{S}_{j,t}\setminus\mathcal{S}_\cap}\|g_{k,s}\|\\
&\le G(|\eta_{i,t} - \eta_{j,t}|\,\mathrm{card}(\mathcal{S}_\cap) + \max(\eta_{i,t},\eta_{j,t})\,\mathrm{card}(\mathcal{S}_{i,t}\triangle\mathcal{S}_{j,t}))\\
&\le G(|\eta_{i,t} - \eta_{j,t}|N_{t-1} + \max(\eta_{i,t},\eta_{j,t})\tau M_{\max}).
\end{aligned} \tag{29}$$

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 (8), we get

$$|\eta_{i,t} - \eta_{j,t}| \leq \frac{R\,|\operatorname{card}(\mathcal{S}_{i,t}) - \operatorname{card}(\mathcal{S}_{j,t})|}{2GN_{t-1}\sqrt{(5\tau+3)N_t M_{\max}}} \leq \frac{R\tau M_{\max}}{2GN_{t-1}\sqrt{(5\tau+3)N_t M_{\max}}}. \tag{30}$$

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 10, 21 and 22, and the inequalities (28), (29) and (30):

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

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

Appendix D. Proof related to the optimistic variants

D.1 Delayed optimistic dual averaging

Theorem 13. *Assume that the maximum delay is bounded by τ. Let delayed optimistic dual averaging (DODA) 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 $x_{\frac{3}{2}}, \ldots, x_{T+\frac{1}{2}}$) 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

$$f_t(x_{t+\frac{1}{2}}) - f_t(u) \le \langle g_{t+\frac{1}{2}}, x_{t+\frac{1}{2}} - u\rangle = \langle g_{t+\frac{1}{2}}, x_{t+\frac{1}{2}} - \widetilde{x}_{t+1}\rangle + \langle g_{t+\frac{1}{2}}, \widetilde{x}_{t+1} - u\rangle \tag{31}$$

By Lemma 20(*b*), we have

$$\langle g_{t+\frac{1}{2}}, \widetilde{x}_{t+1} - u\rangle \le \psi_{t+1}(\widetilde{x}_{t+1}) - \psi_t(\widetilde{x}_t) - \frac{1}{2\eta_t}\|\widetilde{x}_{t+1} - \widetilde{x}_t\|^2. \tag{32}$$

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

$$\begin{aligned}
\langle g_{t+\frac{1}{2}}, x_{t+\frac{1}{2}} - \widetilde{x}_{t+1}\rangle &= \langle g_{t+\frac{1}{2}}, x_{t+\frac{1}{2}} - x_t\rangle + \langle g_{t+\frac{1}{2}}, x_t - \widetilde{x}_t\rangle + \langle g_{t+\frac{1}{2}}, \widetilde{x}_t - \widetilde{x}_{t+1}\rangle \\
&= \langle g_{t+\frac{1}{2}}, -\gamma_t \widetilde{g}_{t+\frac{1}{2}}\rangle + \langle g_{t+\frac{1}{2}}, \eta_t \sum_{s\in\mathcal{U}_t} g_{s+\frac{1}{2}}\rangle + \langle g_{t+\frac{1}{2}}, \widetilde{x}_t - \widetilde{x}_{t+1}\rangle \\
&= \frac{\gamma_t}{2}\left(\|g_{t+\frac{1}{2}} - \widetilde{g}_{t+\frac{1}{2}}\|^2 - \|g_{t+\frac{1}{2}}\|^2 - \|\widetilde{g}_{t+\frac{1}{2}}\|^2\right) \\
&\quad + \eta_t \sum_{s\in\mathcal{U}_t} \langle g_{t+\frac{1}{2}}, g_{s+\frac{1}{2}}\rangle + \langle g_{t+\frac{1}{2}}, \widetilde{x}_t - \widetilde{x}_{t+1}\rangle \\
&\le \frac{\gamma_t}{2}\left(\|g_{t+\frac{1}{2}} - \widetilde{g}_{t+\frac{1}{2}}\|^2 - \|g_{t+\frac{1}{2}}\|^2 - \|\widetilde{g}_{t+\frac{1}{2}}\|^2\right) \\
&\quad + \frac{\eta_t}{2}\|g_{t+\frac{1}{2}}\|^2 + \frac{1}{2\eta_t}\|\widetilde{x}_t - \widetilde{x}_{t+1}\|^2 + \frac{\nu_t\eta_t}{2}\|g_{t+\frac{1}{2}}\|^2 + \frac{\eta_t}{2}\sum_{s\in\mathcal{U}_t}\|g_{s+\frac{1}{2}}\|^2.
\end{aligned} \tag{33}$$

Combining (31), (32), (33) and summing from $t = 1$ to T yields

$$\begin{aligned}
\mathbf{Reg}_T(u) \le \psi_{T+1}(\widetilde{x}_{T+1}) - \psi_1(\widetilde{x}_1) + \sum_{t=1}^{T} \frac{\gamma_t}{2}\left(\|g_{t+\frac{1}{2}} - \widetilde{g}_{t+\frac{1}{2}}\|^2 - \|\widetilde{g}_{t+\frac{1}{2}}\|^2\right) \\
+ \left(-\frac{\gamma_t}{2} + \frac{(\nu_t+1)\eta_t}{2} + \sum_{t\in\mathcal{U}_l}\frac{\eta_l}{2}\right)\|g_{t+\frac{1}{2}}\|^2.
\end{aligned} \tag{34}$$

Since the maximum delay is τ, we have $\nu_t \le \nu \le \tau$ and if $t \in \mathcal{U}_l$ it holds $l > t \ge l - \tau$ giving that $\mathrm{card}(\{l : t \in \mathcal{U}_l\}) \le \tau$. Moreover, as $(\eta_t)_{t\in\mathbb{N}}$ is a decreasing sequence, $t \in \mathcal{U}_l$ also implies $\eta_l \le \eta_t$. The last term of (34) can thus be bounded as following

$$\left(-\frac{\gamma_t}{2} + \frac{(\nu_t+1)\eta_t}{2} + \sum_{t\in\mathcal{U}_l}\frac{\eta_l}{2}\right)\|g_{t+\frac{1}{2}}\|^2 \le \frac{1}{2}((2\tau+1)\eta_t - \gamma_t)\|g_{t+\frac{1}{2}}\|^2 \le 0, \tag{35}$$

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

To conclude, we use lemma Lemma 20(*a*) and observe that $\psi_1(\widetilde{x}_1) = \psi_1(x_1) = 0$ by definition, so that

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

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

D.2 The necessity of scale separation

Theorem 15. *Consider the setup of Corollary 14. 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 (DODA) 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 (DODA) 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:

$$\underbrace{\underbrace{-1 \dots -1}_{\ell} \;\; \underbrace{+1 \dots +1}_{\ell} \;\; \dots \;\; \underbrace{-1 \dots -1}_{\ell} \;\; \underbrace{+1 \dots +1}_{\ell}}_{2K\ell \text{ losses}} \;\; \underbrace{-1 \dots -1}_{\tau+1}$$

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 (DODA) 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. 1 (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				$+1$					-1	

Figure 1: Illustration of the evolution of the optimistic algorithm for a period of feedback in the first example of the proof of Theorem 15. 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 $\varepsilon = 1/2$. ∎

D.3 A lower bound for delayed online learning

Proposition 16. *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). ∎

D.4 Delayed online learning with slow variation

Theorem 17. *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* (DODA) *(evaluated at the points* $x_{\frac{3}{2}}, \ldots, x_{T+\frac{1}{2}}$*) 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 13. 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. ■

D.5 More flexible learning rates

In order prove Proposition 18, we exploit ideas from Section 4. In particular, we generalize both Theorem 13 and Theorem 17 to the case where the learning rate is non-increasing along a faithful permutation.

Theorem 13′. *Assume that the maximum delay is bounded by* τ*. Consider a faithful permutation* σ *and let delayed optimistic dual averaging* (DODA) *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* $x_{\frac{3}{2}}, \ldots, x_{T+\frac{1}{2}}$*) 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 13, we obtain

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

Invoking Proposition 26, we know that $\mathcal{T}_t^\sigma \setminus \mathcal{S}_{\sigma(t)} \subset \{\sigma(t)-\tau,\ldots,\sigma(t)+\tau\}$. Given that $\sigma(t) \notin \mathcal{T}_{t-1}^\sigma$, this implies $\mathcal{U}_t^\sigma \subset \{\sigma(t)-\tau,\ldots,\sigma(t)-1\} \cup \{\sigma(t)+1,\ldots,\sigma(t)+\tau\}$. Therefore, $\mathrm{card}(\mathcal{U}_t^\sigma) \le 2\tau$ and if $\sigma(t) \in \mathcal{U}_l^\sigma$ then $|\sigma(t)-\sigma(l)| \le \tau$ while $\sigma(t) \neq \sigma(l)$, which also shows $\mathrm{card}(\{l : \sigma(t) \in \mathcal{U}_l^\sigma\}) \le 2\tau$. Accordingly,

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

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

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

This proves the theorem. ∎

Theorem 17′. *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 $\widetilde{g}_t = \widetilde{V}_t(x_t)$, $\eta_{\sigma(t+1)} \le \eta_{\sigma(t)}$, $(4\tau+1)\max_{\{s:|s-t|\le\tau\}}\eta_s \le \gamma_t$, and $2\gamma_t^2L^2 \le 1$. Then, the regret of delayed optimistic dual averaging (DODA) (evaluated at the points $x_{\frac{3}{2}},\ldots,x_{T+\frac{1}{2}}$) satisfies*

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

Proof. Apply Theorem 13′ 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 17. ∎

Proposition 18. *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 $\|u-x_1\| \le R$, running delayed optimistic dual averaging (DODA) 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

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

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 8′) and accordingly the learning rate sequence $(\eta_t)_t$ is non-decreasing along σ. Moreover, if $s \in \mathcal{S}_t$, we have $\mathcal{S}_s \subset \mathcal{G}_{i(t),s} \subset \mathcal{S}_t$ 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 17′, 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

$$\widetilde{C}_s + 4G^2(3\tau+1) \geq \widetilde{C}_t + 4G^2(\tau+1) \tag{36}$$

for all $s \in [T] \cap \{t-\tau, \ldots, t+\tau\}$. 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 (36) for s such that $|s-t| \leq \tau$.

With all this at hand, applying Theorem 17′ 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 22 then gives

$$\begin{aligned}\mathbf{Reg}_T(u) &\leq \frac{\|u - x_1\|^2}{2\eta_{\sigma(T)}} + \frac{R\sqrt{4\tau+1}}{2} \sum_{t=1}^{T} \frac{1}{\sqrt{C_t}} \|V_t(x_t) - \widetilde{V}_t(x_t)\|^2 \\ &\leq \frac{R^2}{2\eta_{\sigma(T)}} + R\sqrt{(4\tau+1)C_T}.\end{aligned} \tag{37}$$

We bound the second term by

$$R\sqrt{(4\tau+1)C_T} \leq R\sqrt{(4\tau+1)(\widetilde{C}_T + 4G^2(3\tau+1))} \leq \frac{R^2}{2\eta_T} \leq \frac{R^2}{2\eta_{\sigma(T)}}. \tag{38}$$

Combining (37) and (38) we get $\mathbf{Reg}_T(u) \leq R^2/\eta_{\sigma(T)}$. We can conlcude by using the definition of $\eta_{\sigma(T)}$ and $\widetilde{C}_{\sigma(T)} \leq C_T$. ∎

References

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

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.

Nicolas Flammarion and Francis Bach. Stochastic composite least-squares regression with convergence rate $o(1/n)$. In *Conference on Learning Theory*, pages 831–875, 2017.

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.

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.

Sangkyun Lee and Stephen J Wright. Manifold identification in dual averaging for regularized stochastic online learning. *The Journal of Machine Learning Research*, 13(1):1705–1744, 2012.

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, 2009.

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.

Suvrit Sra, Adams Wei Yu, Mu Li, and Alex Smola. Adadelay: Delay adaptive distributed stochastic optimization. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, volume 51, pages 957–965, 2016.

Sebastian U Stich and Sai Praneeth Karimireddy. The error-feedback framework: Better rates for sgd with delayed gradients and compressed communication. *arXiv preprint arXiv:1909.05350*, 2019.

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.

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.