



**HAL**  
open science

# Handling Delayed Feedback in Distributed Online Optimization: A Projection-Free Approach

Tuan-Anh Nguyen, Nguyen Kim Thang, Denis Trystram

► **To cite this version:**

Tuan-Anh Nguyen, Nguyen Kim Thang, Denis Trystram. Handling Delayed Feedback in Distributed Online Optimization: A Projection-Free Approach. ECML PKDD 2024 - Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Sep 2024, Vilnius, Lithuania. pp.197-211, 10.1007/978-3-031-70341-6\_12 . hal-04703453

**HAL Id: hal-04703453**

**<https://inria.hal.science/hal-04703453v1>**

Submitted on 20 Sep 2024

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

---

# HANDLING DELAYED FEEDBACK IN DISTRIBUTED ONLINE OPTIMIZATION : A PROJECTION-FREE APPROACH \*

---

**Tuan-Anh Nguyen**

LIG, Inria  
University Grenoble-Alpes  
Saint-Martin-d'Hères, France  
tuan-anh.nguyen@inria.fr

**Nguyen Kim Thang**

LIG, Inria, CNRS, Grenoble INP  
University Grenoble-Alpes  
Saint-Martin-d'Hères, France  
kim-thang.nguyen@univ-grenoble-alpes.fr

**Denis Trystram**

LIG, Inria, CNRS, Grenoble INP  
University Grenoble-Alpes  
Saint-Martin-d'Hères, France  
denis.trystram@imag.fr

## ABSTRACT

Learning at the edges has become increasingly important as large quantities of data are continually generated locally. Among others, this paradigm requires algorithms that are *simple* (so that they can be executed by local devices), *robust* (against uncertainty as data are continually generated), and *reliable* in a distributed manner under network issues, especially delays. In this study, we investigate the problem of online convex optimization (OCO) under adversarial delayed feedback. We propose two projection-free algorithms for centralized and distributed settings in which they are carefully designed to achieve a regret bound of  $\mathcal{O}(\sqrt{B})$  where  $B$  is the sum of delay, which is optimal for the OCO problem in the delay setting while still being projection-free. We provide an extensive theoretical study and experimentally validate the performance of our algorithms by comparing them with existing ones on real-world problems.

**Keywords** Online Learning · Distributed Learning · Delayed Feedback · Frank-Wolfe Algorithm

## 1 Introduction

Many machine learning (ML) applications owe their success to factors such as efficient optimization methods, effective system design, robust computation, and the availability of enormous amounts of data. In a typical situation, ML models are trained in an offline and centralized manner. However, in real-life scenarios, significant portions of data are continuously generated locally at the user level. Learning at the edge naturally emerges as a new paradigm to address such issues. In this new paradigm, the development of suitable learning techniques has become a crucial research objective. Responding to the requirements (of this new paradigm), online learning has been intensively studied in recent years. Its efficient use of computational resources, adaptability to changing environments, scalability, and robustness against uncertainty show promise as an effective approach for edge devices.

However, online learning/online convex optimization (OCO) problems typically assume that the feedback is immediately received after a decision is made, which is too restrictive in many real-world scenarios. For example, a common problem in online advertising is the delay that occurs between clicking on an ad and taking subsequent action, such as buying or selling a product. In distributed systems, the previous assumption is clearly a real issue. Wireless sensor/mobile networks that exchange information sequentially may experience delays in feedback due to several

---

\*The authors express their gratitude for the support provided by the Multidisciplinary Institute in Artificial Intelligence, University Grenoble-Alpes, France (ANR-19-P3IA-0003)

problems: connectivity reliability, varying processing/computation times, heterogeneous data and infrastructures, and unaware-random events. This can lead to difficulties in maintaining coordination and efficient data exchange, eventually affecting network performance and responsiveness. Given these scenarios, the straightforward application of traditional OCO algorithms often results in inefficient resource utilization because one must wait for feedback before starting another round. To address this need, this paper focuses on developing algorithms that can adapt to adversarial delayed feedback in both centralized and distributed settings.

**Model.** We first describe the delay model in a centralized setting. Given a convex set  $\mathcal{K} \subseteq \mathbb{R}^d$ , at every time step  $t$ , the decision maker/agent chooses a decision  $\mathbf{x}_t \in \mathcal{K}$  and suffers from a loss function  $f_t : \mathcal{K} \rightarrow \mathbb{R}$ . We denote by  $d_t \geq 1$  an arbitrary delay value of time  $t$ . In contrast to the classical OCO problem, the feedback of iteration  $t$  is revealed at time  $t + d_t - 1$ . The agent does not know  $d_t$  in advance and is only aware of the feedback of iteration  $t$  at time  $t + d_t - 1$ . Consequently, at time  $t$ , the agent receives feedback from the previous iterations  $s \in \mathcal{F}_t$ , where  $\mathcal{F}_t = \{s : s + d_s - 1 = t\}$ . In other words,  $\mathcal{F}_t$  is the set of moments before time  $t$  such that the corresponding feedbacks are released at time  $t$ . Moreover, the corresponding feedbacks are not necessarily released in the order of their iterations. The goal is to minimize regret, which is defined as:

$$\mathcal{R}_T := \sum_{t=1}^T f_t(\mathbf{x}_t) - \min_{\mathbf{x} \in \mathcal{K}} \sum_{t=1}^T f_t(\mathbf{x})$$

In a distributed setting, we have additionally a set of agents connected over a network, represented by a graph  $\mathcal{G} = (V, E)$  where  $n = |V|$  is the number of agents. Each agent  $i \in V$  can communicate with (and only with) its immediate neighbors, that is, adjacent agents in  $\mathcal{G}$ . At each time  $t \geq 1$ , agent  $i$  takes a decision  $\mathbf{x}_t^i \in \mathcal{K}$  and suffers a partial loss function  $f_t^i : \mathcal{K} \rightarrow \mathbb{R}$ , which is revealed adversarially and locally to the agent at time  $(t + d_t^i - 1)$  — again, that is unknown to the agent. Similarly, denote  $\mathcal{F}_t^i = \{s : s + d_s^i - 1 = t\}$  as the set of feedbacks revealed to agent  $i$  at time  $t$  where  $d_s^i$  is the delay of iteration  $s$  to agent  $i$ . Although the limitation in communication and information, the agent  $i$  is interested in the global loss  $F_t(\cdot)$  where  $F_t(\cdot) = \frac{1}{n} \sum_{i=1}^n f_t^i(\cdot)$ . In particular, at time  $t$ , the loss of agent  $i$  for chosen  $\mathbf{x}_t^i$  is  $F_t(\mathbf{x}_t^i)$ . Note that each agent  $i$  does not know  $F_t$  but has only knowledge of  $f_t^i$  — its observed cost function. The objective here is to minimize regret for all agents:

$$\mathcal{R}_T := \max_i \left( \sum_{t=1}^T F_t(\mathbf{x}_t^i) - \min_{\mathbf{x} \in \mathcal{K}} \sum_{t=1}^T F_t(\mathbf{x}) \right)$$

## 1.1 Our contribution

The challenge in designing robust and efficient algorithms for these problems is to address the following issues simultaneously:

- Uncertainty (online setting, agents observe their loss functions only after selecting their decisions).
- Asynchronous (distributed setting with different delayed feedback between agents)
- Partial information (distributed setting, agents know only its local loss functions while attempting to minimize the cumulative loss).
- Low computation/communication resources of agents (so it is desirable that each agent performs a small number of gradient computations and communications).

We introduce performance-guaranteed algorithms in solving the centralized and distributed constraint online convex optimization problem with adversarial delayed feedback. Our algorithms achieve an *optimal* regret bound for centralized and distributed settings. Specifically, we obtain the regret bound of  $O(\sqrt{B})$  where  $B$  is the total delay in the centralized setting and  $B$  is the average total delay over all agents in the distributed setting. Note that, if  $d$  is a maximum delay of each feedback then our regret bound becomes  $O(\sqrt{dT})$ . This result recovers the regret bound

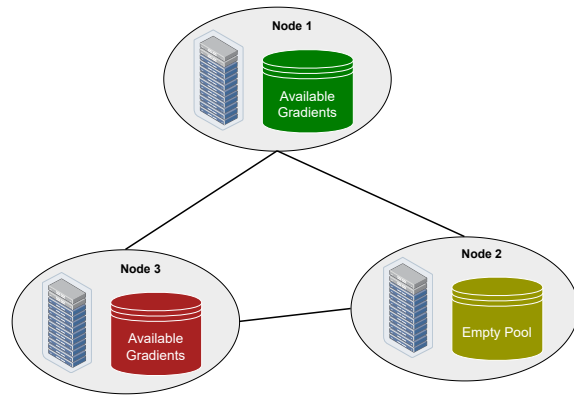


Figure 1: Illustration of delayed feedback in distributed system. Given a time  $t$ , each agent holds a distinct pool of available gradient feedback from  $s < t$  that is ready for computation at the current time. The pool can also be empty if no feedback is provided.

of  $O(\sqrt{T})$  in the classic setting without delay (i.e.,  $d = 1$ ). Additionally, the algorithms can be made projection-free by selecting appropriate oracles, allowing them to be implemented in different contexts based on the computational capacity of local devices. Finally, we illustrate the practical potential of our algorithms and provide a thorough analysis of their performance which is predictably explained by our theoretical results. The experiments demonstrate that our proposed algorithms outperform existing solutions in both synthetic and real-world datasets.

Table 1: Comparisons to previous algorithms DGD [4] and DOFW [5] on centralized online convex optimization with delays bounded by  $d$ . Our algorithms are in bold.

Algorithm	Centralized	Distributed	Adversarial Delay	Projection-free	Regret
DGD	Yes	-	Yes	-	$\mathcal{O}(\sqrt{dT})$
DOFW	Yes	-	Yes	Yes	$\mathcal{O}(T^{3/4} + dT^{1/4})$
<b>DeLMFW</b>	Yes	-	Yes	Yes	$\mathcal{O}(\sqrt{dT})$
<b>De2MFW</b>	-	Yes	Yes	Yes	$\mathcal{O}(\sqrt{dT})$

## 1.2 Related Work

**Online Optimization with delayed feedback** Over the years, studies on online optimization with delayed feedback have undergone a swift evolution. [7] shed light on the field by focusing on the convergence properties of online stochastic gradient descent with delays. They provide a regret bound of  $\mathcal{O}(\sqrt{dT})$  with  $d$  the delay value if  $d^2 \leq T$ . Later on, [4] proposes a centralized (single-agent) gradient descent algorithm under adversarial delays. The theoretical analysis of [4] entails a regret bound of  $\mathcal{O}(\sqrt{B})$ , where  $B$  is the total delay. This bound becomes  $\mathcal{O}(\sqrt{dT})$  if  $d$  is the upper bound of delays. [8] provided a black-box style method to learn under delayed feedback. They showed that for any non-delayed online algorithms, the additional regret in the presence of delayed feedback depends on its prediction drifts. [9] developed an online saddle point algorithm for convex optimization with feedback delays. They achieved a sublinear regret  $\mathcal{O}(\sqrt{dT})$  where  $d$  is a fixed constant delay value. Recently, [5] proposed a first Frank-Wolfe-type online algorithm with delayed feedback. They modified the Online Frank-Wolfe (OFW) for the unknown delays setting and provided a regret bound of  $\mathcal{O}(T^{3/4} + dT^{1/4})$ . This is the current state of the art for projection-free (Frank-Wolfe-type) algorithms with delays. Our bound of  $\mathcal{O}(\sqrt{dT})$  improves over the aforementioned results.

**Distributed Online Optimization.** [10] introduced decentralized online projected subgradient descent and showed vanishing regret for convex and strongly convex functions. In contrast, [11] extended distributed dual averaging technique to the online setting, using a general regularized projection for both unconstrained and constrained optimization. A distributed variant of online conditional gradient [1] was designed and analyzed in [12] that requires linear minimizers and uses exact gradients. Computing exact gradients may be prohibitively expensive for moderately sized data and intractable when a closed form does not exist. [3] proposes a decentralized online algorithm for convex function using stochastic gradient estimate and multiple optimization oracles. This work achieves the optimal regret bound of  $\mathcal{O}(T^{1/2})$  and requires multiple gradient evaluation and communication rounds. Later on, [13] provide a decentralized algorithm that uses stochastic gradient estimate and reduces communication by using only one gradient evaluation. [?] provides a comprehensible survey on recent development of distributed OCO. More recent work on distributed online optimization with feedback delays is proposed in [6]. The authors consider a distributed projected gradient descent algorithm where each agent has a fixed known amount of delay  $d_i$ . They provide a regret bound of  $\mathcal{O}(\sqrt{dT})$  where  $d = \max_i d_i$  but the delays  $d_i$  must be fixed (non-adversarial).

Despite the growing number of studies on decentralized online learning in recent years, there is a lack of research that accounts for the *adversarial/online* delayed feedback. In this paper, we first present a centralized online algorithm and then extend it to a distributed online variant that takes an adversarial delay setting into consideration.

## 2 Projection-Free Algorithms under Delayed Feedback

In this section, we will present our method for addressing delayed feedback in the OCO problem. In Section 2.1, we state some assumptions and results that form the basis of our approach. We then describe our first algorithm DeLMFW for centralized setting in Section 2.2 and extend it to distributed setting in Section 2.3.

## 2.1 Preliminaries

Throughout the paper, we use boldface letter e.g  $\mathbf{x}$  to represent vectors. We denote by  $\mathbf{x}_t$  the final decision of round  $t$  and  $\mathbf{x}_{t,k}$  to be the sub-iterate at round  $k$  of  $t$ . In distributed setting, we add a superscript  $i$  to make distinction between agents. If not specified otherwise, we use Euclidean norm  $\|\cdot\|$  and suppose that the constraint set  $\mathcal{K} \subset \mathbb{R}^m$  is convex. We state the following standard assumptions in OCO.

**Assumption 1.** *The constraint set  $\mathcal{K}$  is a bounded convex set with diameter  $D$  i.e  $D := \sup_{\mathbf{x}, \mathbf{y} \in \mathcal{K}} \|\mathbf{x} - \mathbf{y}\|$*

**Assumption 2** (Lipschitz). *For all  $\mathbf{x} \in \mathcal{K}$ , there exists a constant  $G$  such that,  $\forall t \in [T]$ ,  $\|\nabla f_t(\mathbf{x})\| \leq G$*

**Assumption 3** (Smoothness). *For all  $\mathbf{x}, \mathbf{y} \in \mathcal{K}$ , there exists a constant  $\beta$  such that,  $\forall t \in [T]$  :*

$$f_t(\mathbf{y}) \leq f_t(\mathbf{x}) + \langle \nabla f_t(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{\beta}{2} \|\mathbf{y} - \mathbf{x}\|^2$$

or equivalently  $\|\nabla f_t(\mathbf{x}) - \nabla f_t(\mathbf{y})\| \leq \beta \|\mathbf{x} - \mathbf{y}\|$ .

**Online Linear Optimization Oracles** In the context of the Frank-Wolfe (FW) algorithm, we utilize multiple optimization oracles to approximate the gradient of the upcoming loss function by solving an online linear problem. This approach was first introduced in [2]. Specifically, the online linear problem involves selecting a decision  $\mathbf{v}_t \in \mathcal{K}$  at every time  $t \in [T]$ . The adversary then reveals a vector  $\mathbf{g}_t$  and loss function  $\langle \mathbf{g}_t, \cdot \rangle$  to the oracle. The objective is to minimize the oracle's regret. A possible candidate for an online linear oracle is the Follow the Perturbed Leader algorithm (FTPL) [14]. Given a sequence of historical loss functions  $\langle \mathbf{g}_\ell, \cdot \rangle$ ,  $s \in [1, t]$  and a random vector  $\mathbf{n}$  drawn uniformly from a probability distribution  $\mathcal{D}$ , FTPL makes the following update.

$$\hat{\mathbf{v}}_{t+1} = \arg \min_{\mathbf{v} \in \mathcal{K}} \left\{ \zeta \sum_{\ell=1}^t \langle \mathbf{g}_\ell, \mathbf{v} \rangle + \langle \mathbf{n}, \mathbf{v} \rangle \right\} \quad (1)$$

**Lemma 1** (Theorem 5.8 [1]). *Given a sequence of linear loss function  $f_1, \dots, f_T$ . Suppose that Assumptions 1 to 3 hold true. Let  $\mathcal{D}$  be a the uniform distribution over hypercube  $[0, 1]^m$ . The regret of FTPL is*

$$\mathcal{R}_{T, \mathcal{O}} \leq \zeta D G^2 T + \frac{1}{\zeta} \sqrt{m} D$$

where  $\zeta$  is learning rate of algorithm.

**Delay Mechanism** We consider the following delay mechanism. At round  $t$ , the agent receives a set of delayed gradient  $\nabla f_s(\mathbf{x}_s)$  from previous rounds  $s \leq t$  such that  $s + d_s - 1 = t$ , where  $d_s$  is the delay value of iteration  $s$ . We denote by  $\mathcal{F}_t = \{s : s + d_s - 1 = t\}$  the set of indices released at round  $t$ . Following this setting, the feedback of round  $t$  is released at time  $t + d_t - 1$ , and the case  $d_t = 1$  is considered as no delay. We suppose that the delay value is unknown to the agent and make no assumption about the set  $\mathcal{F}_t$ . Consequently it is possible for the set to be empty at any particular round. We extend the aforementioned mechanism to the distributed setting by assuming that each agent has a unique delay value at each round  $t \in [T]$ . The delay value of agent  $i$  at round  $t$  is denoted by  $d_t^i$ , and the set of delayed feedbacks of agent  $i$  at round  $t$  is denoted by  $\mathcal{F}_t^i = \{s : s + d_s^i - 1 = t\}$ , which is distinct between agents.

## 2.2 Centralized Algorithm

We describe the procedure of Algorithm 1 in details. At each round  $t$ , the agent performs two blocks of operations: prediction and update. During the prediction block, the agent performs  $K$  iterations of FW updates by querying solutions from the oracles  $\mathcal{O}_k$ ,  $k \in [K]$  and updates the sub-iterate vector  $\mathbf{x}_{t,k+1}$  using a convex combination of the previous one and the oracle's output. The agent then plays the final decision  $\mathbf{x}_t = \mathbf{x}_{t,K+1}$  and incurs a loss  $f_t(\mathbf{x}_t)$  which may not be revealed at  $t$  due to delay. From the mechanism described in Section 2.1, there exists a set of gradient feedbacks from the previous rounds revealed at  $t$  whose indices are in  $\mathcal{F}_t$ . The update block involves observing the delayed gradients evaluated at  $K$  sub-iterates of rounds  $s \in \mathcal{F}_t$ , computing surrogate gradients  $\{\mathbf{g}_{t,k}, k \in [K]\}$  by summing the delayed gradients and feeding them back to the oracles  $\{\mathcal{O}_k, k \in [K]\}$ .

In our algorithm, the agent employs a suite of online linear optimization oracles, denoted  $\mathcal{O}_1, \dots, \mathcal{O}_K$ . These oracles utilize feedbacks accumulated from previous rounds to estimate the gradient of the upcoming loss function. However, in the delay setting, these estimations may be perturbed owing to a lack of information. For example, if there is no feedback from rounds  $t$  to  $t'$ , that is,  $\mathcal{F}_s = \emptyset$  for  $s \in [t, t']$ , the oracles will resort to the information available in round  $t - 1$  to estimate the gradient of all rounds from  $t + 1$  to  $t' + 1$ . As a result, the oracle's output remains unchanged for these rounds, and decisions  $\{\mathbf{x}_s : s \in [t + 1, t' + 1]\}$  are not improved. Our analysis for Algorithms 1 and 2 will be focused on assessing the impact of delayed feedback on the oracle's output.

**Lemma 2.** Let  $\hat{\mathbf{v}}_t$  be the FTPL prediction defined in Equation (1) and

$$\mathbf{v}_t = \arg \min_{\mathbf{v} \in \mathcal{K}} \left\{ \zeta \sum_{\ell=1}^{t-1} \left\langle \sum_{s \in \mathcal{F}_\ell} \mathbf{g}_s, \mathbf{v} \right\rangle + \langle \mathbf{n}, \mathbf{v} \rangle \right\}$$

the prediction of FTPL with delayed feedback. For all  $t \in [T]$ , we have:

$$\|\mathbf{v}_t - \hat{\mathbf{v}}_t\| \leq \zeta DG \sum_{s < t} \mathbb{I}_{\{s+d_s > t\}}$$

---

### Algorithm 1 DeLMFW

---

**Input:** Constraint set  $\mathcal{K}$ , number of iterations  $T$ , sub-iteration  $K$ , online oracles  $\{\mathcal{O}_k\}_{k=1}^K$ , step sizes  $\eta_k \in (0, 1]$

```

1: for  $t = 1$  to  $T$  do
2:   Initialize arbitrarily  $\mathbf{x}_{t,1} \in \mathcal{K}$ 
3:   for  $k = 1$  to  $K$  do
4:     Query  $\mathbf{v}_{t,k}$  from oracle  $\mathcal{O}_k$ .
5:      $\mathbf{x}_{t,k+1} \leftarrow (1 - \eta_k)\mathbf{x}_{t,k} + \eta_k\mathbf{v}_{t,k}$ .
6:   end for
7:    $\mathbf{x}_t \leftarrow \mathbf{x}_{t,K+1}$ , play  $\mathbf{x}_t$  and incurs loss  $f_t(\mathbf{x}_t)$ 
8:   Receive  $\mathcal{F}_t = \{s \in [T] : s + d_s - 1 = t\}$ 
9:   if  $\mathcal{F}_t = \emptyset$  then
10:    do nothing
11:  else
12:    for  $k = 1$  to  $K$  do
13:       $\mathbf{g}_{t,k} \leftarrow \sum_{s \in \mathcal{F}_t} \nabla f_s(\mathbf{x}_{s,k})$ 
14:      Feedback  $\langle \mathbf{g}_{t,k}, \cdot \rangle$  to oracles  $\mathcal{O}_k$ .
15:    end for
16:  end if
17: end for

```

---

**Theorem 1.** Given a constraint set  $\mathcal{K}$ . Let  $A = \max\left\{3, \frac{G}{\beta D}\right\}$ ,  $\eta_k = \min\left\{1, \frac{A}{k}\right\}$ , and  $K = \sqrt{T}$ . Suppose that Assumptions 1 to 3 hold true. If we choose FTPL as the underlying oracle and set  $\zeta = \frac{1}{G\sqrt{B}}$ , the regret of Algorithm 1 is

$$\sum_{t=1}^T [f_t(\mathbf{x}_t) - f_t(\mathbf{x}^*)] \leq 2\beta AD^2\sqrt{T} + 3(A+1) \left( DG\sqrt{B} + \mathcal{R}_{T,\mathcal{O}} \right) \quad (2)$$

where  $B = \sum_{t=1}^T d_t$ , the sum of all delay values and  $\mathcal{R}_{T,\mathcal{O}}$  is the regret of FTPL with respect to the current choice of  $\zeta$ .

**Discussion** The regret bound of Theorem 1 differs from that of the non-delayed MFW [2] by the additive term  $DG\sqrt{B}$  which represents the total cost of sending delayed feedback to the oracles over  $T$  rounds (Lemma 2). If we assume that there exists a maximum value  $d$  such that  $d_t \leq d$  for all  $t \in [T]$ . Our regret bound becomes  $\mathcal{O}(\sqrt{dT})$  which coincides with the setting in [5], a delayed-feedback FW algorithm that achieves  $\mathcal{O}(T^{3/4} + dT^{1/4})$ . Another line of work is from [8], a framework that addresses delayed feedback for any base algorithm. By considering MFW as the base algorithm, their theoretical analysis suggests that the algorithm also achieves  $\mathcal{O}(\sqrt{dT})$  regret bound. However, their delay value is not completely unknown to the agent because it is time-stamped by maintaining multiple copies of the base algorithm. We empirically show in Section 3 that this algorithm is highly susceptible to high delay values. Instead of using FTPL, our algorithm has the flexibility to select any online algorithm as an oracle, for example, Online Gradient Descent [1].

## 2.3 Distributed Algorithm

In this section, we extend Algorithm 1 to a distributed setting in which multiple agents collaboratively optimize a global model. Our setting considers a fully distributed framework, characterized by the absence of a server to coordinate the learning process. Let  $\mathbf{W} \in \mathbb{R}_+^{n \times n}$  be the adjacency matrix of communication graph  $\mathcal{G} = (V, E)$ . The entries  $w_{ij}$  are defined as

$$w_{ij} = \begin{cases} \frac{1}{1 + \max\{\tau_i, \tau_j\}} & \text{if } (i, j) \in E \\ 0 & \text{if } (i, j) \notin E, i \neq j \\ 1 - \sum_{j \in N(i)} w_{ij} & \text{if } i = j \end{cases}$$

where  $\tau_i = \#\{j \in V : (i, j) \in E\}$  is the degree of vertex  $i$ . The matrix  $\mathbf{W}$  is doubly stochastic (i.e  $\mathbf{W}\mathbf{1} = \mathbf{W}^T\mathbf{1} = \mathbf{1}$ ) and therefore possesses several useful properties associated with doubly stochastic matrices. We call  $\lambda(\mathbf{W})$  the second-largest eigenvalue of  $\mathbf{W}$  and define  $k_0$  as the smallest integer that verifies  $\lambda(\mathbf{W}) \leq \left(\frac{k_0}{k_0+1}\right)^2$ . Furthermore, we set  $\rho = 1 - \lambda(\mathbf{W})$  to be the spectral gap of matrix  $\mathbf{W}$ .

At a high level, each agent maintains  $K$  copies of the oracles  $\mathcal{O}_1^i, \dots, \mathcal{O}_K^i$  while performing prediction and update at every round  $t$ . The prediction block consists of performing  $K$  FW-steps while incorporating the neighbors' information. Specifically, the agent computes at its local level during the  $K$  steps a local average decision  $\mathbf{y}_{t,k}^i$  representing a weighted aggregation of its neighbor's current sub-iterates. The update vector is convex combination of the local average decision and the oracle's output. The final decision of agent  $\mathbf{x}_t^i$  is disclosed at the end of  $K$  steps. Lemma 3 shows that  $\mathbf{y}_{t,k}^i$  is a local estimation of the global average  $\bar{\mathbf{x}}_{t,k} = \frac{1}{n} \sum_{i=1}^n \mathbf{x}_{t,k}^i$  as  $K$  increases.

Following the  $K$  FW-steps, the update block employs  $K$  gradient updates utilizing the delayed feedback from previous rounds. The agent observes the delayed gradients evaluated on their corresponding subiterates and computes the local average gradient  $\mathbf{d}_{t,k}^i$  through a weighted aggregation of the neighbors' current surrogates (18). The agent updates the surrogate gradient via a gradient-tracking step (19) to ensure that it approaches the global gradient as  $K$  increases. It is worth noting that feedback provided to the oracle contains information about delays experienced by all neighboring agents. Consequently, the oracle  $\mathcal{O}_k^i$  observes delayed feedback from  $\cup_{j \in \mathcal{N}(i)} \mathcal{F}_t^j$  instead of  $\mathcal{F}_t^i$ . This result highlights the dependency on the connectivity of the communication graph when considering the effect of delayed feedback to the oracle's output, as demonstrated in Lemma 4.

**Lemma 3.** Define  $C_d = k_0\sqrt{n}D$ , for all  $t \in [T]$ ,  $k \in [K]$ , we have

$$\max_{i \in [1, n]} \|\mathbf{y}_{t,k}^i - \bar{\mathbf{x}}_{t,k}\| \leq \frac{C_d}{k} \quad (3)$$

**Lemma 4.** For all  $t \in [T]$ ,  $k \in [K]$  and  $i \in [n]$ . Let  $\mathbf{v}_{t,k}^i$  be the output of the oracle  $\mathcal{O}_k^i$  with delayed feedback and  $\hat{\mathbf{v}}_{t,k}^i$  its homologous in non-delay case. Suppose that Assumptions 1 and 2 hold true. Choosing FTPL as the oracle, we have:

$$\|\mathbf{v}_{t,k}^i - \hat{\mathbf{v}}_{t,k}^i\| \leq 2\zeta\sqrt{n}DG \left( \frac{\lambda(\mathbf{W})}{\rho} + 1 \right) \frac{1}{n} \sum_{i=1}^n \sum_{s \leq t} \mathbb{I}_{\{s+d_s^i > t\}} \quad (4)$$

where  $\zeta$  is the learning rate,  $\lambda(\mathbf{W})$  is the second-largest eigenvalue of  $\mathbf{W}$  and  $\rho = 1 - \lambda(\mathbf{W})$  is the spectral gap of matrix  $\mathbf{W}$ .

**Theorem 2.** Given a constraint set  $\mathcal{K}$ . Let  $A = \max \left\{ 3, \frac{3G}{2\beta D}, \frac{2\beta C_d + C_g}{\beta D} \right\}$ ,  $\eta_k = \min \left\{ 1, \frac{A}{k} \right\}$ , and  $K = \sqrt{T}$ . Suppose that Assumptions 1 to 3 hold true. If we choose FTPL as the underlying oracle and set  $\zeta = \frac{1}{G\sqrt{B}}$ , the regret of Algorithm 2 is

$$\sum_{t=1}^T [F_t(\mathbf{x}_t^i) - F_t(\mathbf{x}^*)] \leq (GC_d + 2\beta AD^2) \sqrt{T} + 3(A+1) \left( 2\sqrt{n}DG \left( \frac{\lambda(\mathbf{W})}{\rho} + 1 \right) \sqrt{B} + \mathcal{R}_{T, \mathcal{O}} \right)$$

where  $B = \frac{1}{n} \sum_{i=1}^n B_i$  such that  $B_i$  is the sum of all delay values of agent  $i$ .  $C_d = k_0\sqrt{n}D$  and  $C_g = \sqrt{n} \max \left\{ \lambda_2(\mathbf{W}) \left( G + \frac{\beta D}{\rho} \right), k_0\beta(4C_d + AD) \right\}$  and  $\mathcal{R}_{T, \mathcal{O}}$  is the regret of FTPL with respect to the current choice of  $\zeta$ .

### 3 Numerical Experiments

We evaluated the performance of our algorithms on the online multiclass logistic regression problem using two datasets: MNIST and FashionMNIST. MNIST is a well-known hand digit dataset containing 60000 grayscale images of size  $(28 \times 28)$ , divided into 10 classes, and FashionMNIST includes images of fashion products with the same configuration. We conducted the experiment using Julia 1.7 on MacOS 13.3 with 16GB of memory. The code is available at <https://github.com/tuananhgh/DelayMFW>.

**Centralized Setting** Given an iteration  $t$ , the agent receives a subset  $\mathcal{B}_t$  of the form  $\mathbf{b}_t = \{\mathbf{a}_t, y_t\} \in \mathbb{R}^m \times \{1, \dots, C\}$ , consisting of the features vector  $\mathbf{a}_t$  and the corresponding label  $y_t$ . We define the loss function  $f_t$  as

$$f_t(\mathbf{x}) = - \sum_{\mathbf{b}_t \in \mathcal{B}_t} \sum_{c=1}^C \{y_t^i = c\} \log \frac{\exp \langle \mathbf{x}_c, \mathbf{a}_t^i \rangle}{\sum_{\ell=1}^C \exp \langle \mathbf{x}_\ell, \mathbf{a}_t^i \rangle} \quad (5)$$

---

**Algorithm 2** De2MFW

**Input:** Constraint set  $\mathcal{K}$ , number of iterations  $T$ , sub-iterations  $K$ , online linear optimization oracles  $\{\mathcal{O}_k^i : k \in [K]\}$  for each agent  $i \in [n]$ , step sizes  $\eta_k \in (0, 1]$

```

1: for  $t = 1$  to  $T$  do
2:   for every agent  $i = 1$  to  $n$  do
3:     Initialize arbitrarily  $\mathbf{x}_{t,1}^i \in \mathcal{K}$ 
4:     for  $k = 1$  to  $K$  do
5:       Query  $\mathbf{v}_{t,k}^i$  from oracle  $\mathcal{O}_k^i$ 
6:       Exchange  $\mathbf{x}_{t,k}^i$  with neighbours  $\mathcal{N}(i)$ 
7:        $\mathbf{y}_{t,k}^i \leftarrow \sum_j w_{ij} \mathbf{x}_{t,k}^j$ 
8:        $\mathbf{x}_{t,k+1}^i \leftarrow (1 - \eta_k) \mathbf{y}_{t,k}^i + \eta_k \mathbf{v}_{t,k}^i$ 
9:     end for
10:     $\mathbf{x}_t^i \leftarrow \mathbf{x}_{t,K+1}^i$ , play  $\mathbf{x}_t^i$  and incurs loss  $f_t^i(\mathbf{x}_t^i)$ 
11:    Receive  $\mathcal{F}_t^i = \{s \in [T] : s + d_s^i - 1 = t\}$ 
12:    if  $\mathcal{F}_t^i = \emptyset$  then
13:      do nothing
14:    else
15:       $\mathbf{g}_{t,1}^i \leftarrow \sum_{s \in \mathcal{F}_t^i} \nabla f_s^i(\mathbf{x}_{s,1}^i)$ 
16:      for  $k = 1$  to  $K$  do
17:        Exchange  $\mathbf{g}_{t,k}^i$  with neighbours  $\mathcal{N}(i)$ 
18:         $\mathbf{d}_{t,k}^i \leftarrow \sum_{j \in \mathcal{N}(i)} w_{ij} \mathbf{g}_{t,k}^j$ 
19:         $\mathbf{g}_{t,k+1}^i \leftarrow \sum_{s \in \mathcal{F}_t^i} (\nabla f_s^i(\mathbf{x}_{s,k+1}^i) - \nabla f_s^i(\mathbf{x}_{s,k}^i))$ 
20:           $+ \mathbf{d}_{t,k}^i$ 
21:        Feedback  $\langle \mathbf{d}_{t,k}^i, \cdot \rangle$  to oracles  $\mathcal{O}_{i,k}$ 
22:      end for
23:    end if
24:  end for

```

---

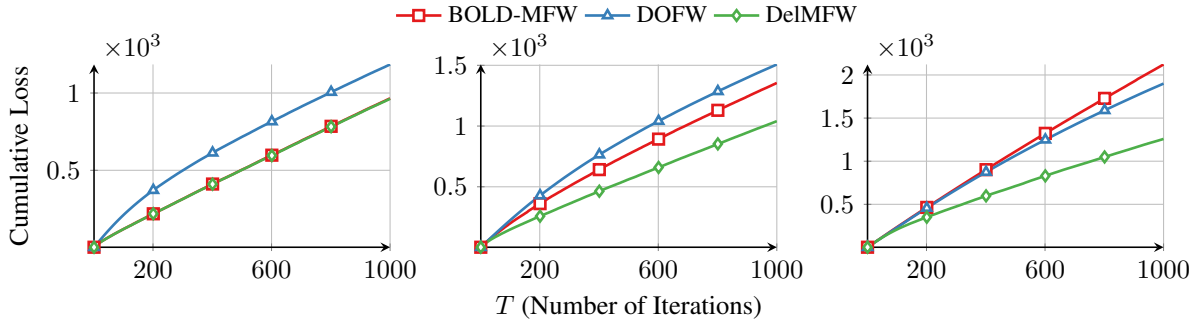


Figure 2: *Cumulative Loss Comparison for Different Delays Regimes. Left : Without delay. Middle : Maximal delay 21. Right : Maximal delay 101*

where  $\mathbf{x}$  must satisfy the constraint  $\mathbf{x} \in \mathcal{K}$  such that  $\mathcal{K} = \{\mathbf{x} \in \mathbb{R}^{m \times C}, \|\mathbf{x}\|_1 \leq r\}$ . Using the MNIST dataset, we note  $m = 784$ ,  $C = 10$ ,  $r = 8$ ,  $|\mathcal{B}_t| = 60$  and a total of  $T = 1000$  rounds. To evaluate the performance of the algorithm under different delay regimes, we generated a random sequence of delays  $d_t$  such that  $d_t \leq d$  for  $d \in \{21, 41, 61, 81, 101\}$ . We compared the performance of DeLMFW against DOFW [5], a projection-free algorithm with adversarial delay, and BOLD-MFW [8], an online learning framework designed to handle delayed feedback. Figure 2 displays the performance of the three algorithms under various delay regimes. In the absence of delay, that is,  $d = 1$  (left figure), DeLMFW and BOLD-MFW have the same performance since both algorithms reduce to MFW [2] with a regret of  $\mathcal{O}(\sqrt{T})$ . Meanwhile, DOFW is the classical OFW [15] that guarantees a regret of  $\mathcal{O}(T^{3/4})$ . The analysis in Theorem 1 suggests that DeLMFW achieves a regret of  $\mathcal{O}(\sqrt{dT})$  when the delay is upper-bounded by  $d$ . In the case where  $d \leq T^{1/2}$  (middle figure,  $d = 21$ ), the dominant term in DOFW is  $T^{3/4}$  whereas DeLMFW takes advantage by incurring a regret of order  $\sqrt{dT} \leq T^{3/4}$ . For  $d \geq T^{1/2}$  (right figure,  $d = 101$ ), DOFW's regret is dominated by the term  $dT^{1/4}$ , which is outperformed by DeLMFW, particularly for high values of  $d$ . This result confirms our theoretical analysis in Section 2.2.



Table 2: Total Loss of the algorithm running on 4 different topology. We randomly select  $f < n$  agents to have delay with maximal value to be 501. In parenthesis, the percentage of total loss compared that of no delayed agents in the network (i.e  $f = 0$ ).

$f$ \ Topology	Erdős-Rényi	Grid	Complete	Cycle
0	809.37	855.62	799.49	925.72
2	820.15 (+1.3%)	852.15 (-0.4%)	798.79 (-0.08%)	932.34 (+0.7%)
5	834.74 (+3.0%)	868.52 (+1.4%)	802.59 (+0.3%)	971.24 (+4.7%)
10	838.74 (+3.5%)	878.04 (+2.5%)	792.45 (-0.8%)	983.89 (+6.0%)
20	850.49 (+4.9%)	902.30 (+5.3%)	812.21 (+1.5%)	1119.24 (+18.9%)

Figure 3 illustrates the total loss of DeLMFW and the other two algorithms when increasing  $d$  to show the sensitivity of each algorithm in the presence of delays. As BOLD is a general framework that can be applied to any base algorithm, it is noticeable that it is susceptible to high levels of delays. This phenomenon has also been observed in [5] when utilizing BOLD with OFW, highlighting the need for a customized design algorithms in the context of delayed feedback.

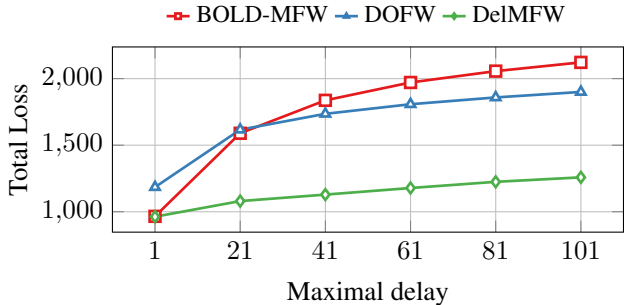


Figure 3: Total loss of BOLD-MFW, DOFW and DeLMFW when varying delay value.

**Distributed Setting** In the second experiment, we examined the distributed online multiclass logistic regression problem on the FashionMNIST dataset, using a network of 30 agents. The algorithm was run on four different topologies, including Erdos-Renyi, Complete, Grid, and Cycle. At each iteration  $t \in [T]$ , each agent  $i$  received a subset  $\mathcal{B}_t^i$  of the form  $\{\mathbf{a}_t^i, y_t^i\} \in \mathbb{R}^d \times \{1, \dots, C\}$ , which consisted of the feature vector  $\mathbf{a}_t^i$  and its corresponding label  $y_t^i$ . The goal was to collaboratively optimize the global loss function  $F_t(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^n f_t^i(\mathbf{x})$ , where the local loss  $f_t^i$  was defined in Equation (5).

Figure 3: Total loss of BOLD-MFW, DOFW and DeLMFW when varying delay value.

For this experiment, we set  $m = 784$ ,  $C = 10$ ,  $r = 32$ ,  $|\mathcal{B}_t^i| = 2$  and  $T = 1000$  rounds. We are interested in examining the effect of delays on network performance, and thus randomly select  $f < n$  agents to have delayed feedback with a maximum value of 501. We compared the total loss on each topology under these conditions, and present the result in Figure 4. We observe that the presence of delayed agents has a significant impact on the network performance of Cycle graph as the number of delayed agents increases, while the Complete graph is less affected. This result is consistent with the analysis in Section 2.3 because the delay term in the regret bound depends on the connectivity of the communication graph.

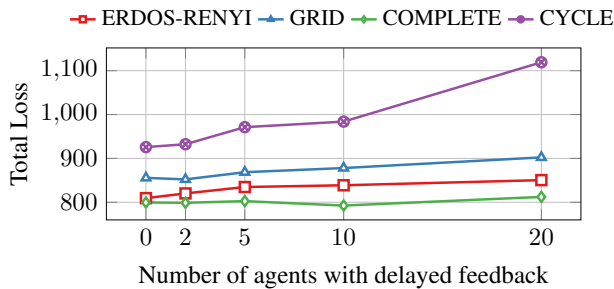


Figure 4: Total Loss with varying numbers of agents experiencing delayed feedback in the network. ( $f = 0$ ) for no delayed-agents.

In Table 2, we report the change in total loss when increasing the number of delayed agents. We observe that the average percentage change is smaller for Grid than for Erdos-Renyi when compared with the network of non-delayed agents ( $f = 0$ ). This result indicate that the generated Erdos-Renyi graph is more sensitive to the presence of delayed agents.

## 4 Concluding Remarks

In this paper, we propose two algorithms for solving the online convex optimization problem with adversarial delayed feedback in both centralized and decentralized settings. These algorithms achieve optimal  $\mathcal{O}(\sqrt{dT})$  regret bounds, where  $d$  is the upper bound of the delays. The experimental results show that our algorithms outperform existing solutions in both centralized and decentralized settings, which are predictable by our theoretical analysis. Although the algorithms achieve good performance guarantees for the online convex optimization problem with adversarial delays, they currently rely on exact gradients, which may not be feasible for many real-world applications. Therefore, future research could explore the use of stochastic gradients with variance reduction techniques. Additionally, in decentralized settings, communication delays can be practically

challenging, and further improvements are needed in this area. Nevertheless, our work demonstrates the potential of using Frank-Wolfe-type algorithms for solving constraint convex optimization problems under adversarial delays, which is beneficial for learning on edge devices.

## References

- [1] Elad Hazan. Introduction to online convex optimization. *Foundations and Trends® in Optimization*, 2(3-4):157–325, 2016.
- [2] Lin Chen, Christopher Harshaw, Hamed Hassani, and Amin Karbasi. Projection-free online optimization with stochastic gradient: From convexity to submodularity. In *Proceedings of the 35th International Conference on Machine Learning*, pages 814–823, 2018.
- [3] Nguyen Kim Thang, Abhinav Srivastav, Denis Trystram, and Paul Youssef. A stochastic conditional gradient algorithm for decentralized online convex optimization. *Journal of Parallel and Distributed Computing*, 169:334–351, 2022.
- [4] Kent Quanrud and Daniel Khashabi. Online learning with adversarial delays. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015.
- [5] Yuanyu Wan, Wei-Wei Tu, and Lijun Zhang. Online frank-wolfe with arbitrary delays. In *Advances in Neural Information Processing Systems*, volume 35, pages 19703–19715. Curran Associates, Inc., 2022.
- [6] Xuanyu Cao and Tamer Başar. Decentralized online convex optimization with feedback delays. *IEEE Transactions on Automatic Control*, 67(6):2889–2904, 2022.
- [7] Martin Zinkevich, John Langford, and Alex Smola. Slow learners are fast. In Y. Bengio, D. Schuurmans, J. Lafferty, C. Williams, and A. Culotta, editors, *Advances in Neural Information Processing Systems*, volume 22. Curran Associates, Inc., 2009.
- [8] Pooria Joulani, Andras Gyorgy, and Csaba Szepesvari. Online learning under delayed feedback. In *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pages 1453–1461, Atlanta, Georgia, USA, 17–19 Jun 2013. PMLR.
- [9] Xuanyu Cao, Junshan Zhang, and H. Vincent Poor. Constrained online convex optimization with feedback delays. *IEEE Transactions on Automatic Control*, 66(11):5049–5064, 2021.
- [10] F. Yan, S. Sundaram, S. V. N. Vishwanathan, and Y. Qi. Distributed autonomous online learning: Regrets and intrinsic privacy-preserving properties. *IEEE Transactions on Knowledge and Data Engineering*, 25(11):2483–2493, 2013.
- [11] S. Hosseini, A. Chapman, and M. Mesbahi. Online distributed optimization via dual averaging. In *52nd IEEE Conference on Decision and Control*, pages 1484–1489, 2013.
- [12] W. Zhang, P. Zhao, W. Zhu, S.C.V. Hoi, and T. Zhang. Projection-free distributed online learning in networks. In *Proceedings of the 34th International Conference on Machine Learning*, pages 4054–4062, 2017.
- [13] Tuan-Anh Nguyen, Nguyen Kim Thang, and Denis Trystram. One gradient frank-wolfe for decentralized online convex and submodular optimization. In *Proceedings of The 14th Asian Conference on Machine Learning*, volume 189 of *Proceedings of Machine Learning Research*, pages 802–815. PMLR, 12–14 Dec 2023.
- [14] Adam Kalai and Santosh Vempala. Efficient algorithms for online decision problems. *Journal of Computer and System Sciences*, 71(3):291–307, 2005. Learning Theory 2003.
- [15] Elad Hazan and Satyen Kale. Projection-free online learning. In *Proceedings of the 29th International Conference on Machine Learning, ICML 2012*, Proceedings of the 29th International Conference on Machine Learning, ICML 2012, pages 521–528, 2012. 29th International Conference on Machine Learning, ICML 2012 ; Conference date: 26-06-2012 Through 01-07-2012.
- [16] Anastasia Koloskova, Sebastian Stich, and Martin Jaggi. Decentralized stochastic optimization and gossip algorithms with compressed communication. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 3478–3487. PMLR, 09–15 Jun 2019.

---

## Supplementary Material

### Handling Delayed Feedback in Distributed Online Optimization : A Projection-Free Approach

---

We start by recalling notations that will be used throughout the proof of Theorem 1 and ???. All notation, are described in the centralized setting. For decentralized setting, if not specify otherwise, we add a superscript  $i$  to indicate the dependant of the variable to agent  $i$ . Following the notation in the main section, we note  $\mathcal{F}_t = \{s \in [T], s + d_s - 1 = t\}$  as the set of feedback arrives at time  $t$  (denoted  $\mathcal{F}_t^i$  in decentralized setting). The sum of total delay is denoted by  $B := \sum_{t=1}^T d_t$  and we call  $d$  the upper bound on all delay value when it is clear from the context. For simplicity of notation, we denote by  $\nabla f_{t,k} = \nabla f_t(\mathbf{x}_{t,k})$ ,  $\zeta$  the learning rate of the oracle and  $\pi_{\mathcal{K}}(\mathbf{x})$  the projection of  $\mathbf{x}$  into the constraint set  $\mathcal{K}$ . If  $t$  is the time the agent plays the decision  $\mathbf{x}_t$ , the feedback is revealed at time  $\sigma(t) := t + d_t - 1$ .

## A Analysis of Algorithm 1

### A.1 Auxiliary Lemmas

**Lemma 2.** Let  $\hat{\mathbf{v}}_t$  be the FTPL prediction defined in Equation (1) and

$$\mathbf{v}_t = \arg \min_{\mathbf{v} \in \mathcal{K}} \left\{ \zeta \sum_{\ell=1}^{t-1} \left\langle \sum_{s \in \mathcal{F}_\ell} \mathbf{g}_s, \mathbf{v} \right\rangle + \langle \mathbf{n}, \mathbf{v} \rangle \right\}$$

the prediction of FTPL with delayed feedback. For all  $t \in [T]$ , we have:

$$\|\mathbf{v}_t - \hat{\mathbf{v}}_t\| \leq \zeta DG \sum_{s < t} \mathbb{I}_{\{s+d_s > t\}}$$

*Proof.* Recall that  $f_t$  is a linear function defined as  $f_t = \langle \mathbf{g}_t, \cdot \rangle$ . For the ease of analysis, we call  $\hat{\mathbf{u}}_t = \zeta \sum_{\ell=1}^{t-1} \mathbf{g}_\ell$  and  $\mathbf{u}_t = \zeta \sum_{\ell=1}^{t-1} \sum_{s \in \mathcal{F}_\ell} \mathbf{g}_s$ . We define  $h_t(\mathbf{n}) = \arg \min_{\mathbf{v} \in \mathcal{K}} \{\langle \mathbf{n}, \mathbf{v} \rangle\}$ . By definition of  $\hat{\mathbf{v}}_t$  and  $\mathbf{v}_t$ , we have

$$\hat{\mathbf{v}}_t = \mathbb{E} [h_t(\mathbf{n} + \hat{\mathbf{u}}_t)] = \int_{\mathbf{n}} h_t(\mathbf{n} + \hat{\mathbf{u}}_t) p(\mathbf{n}) d\mathbf{n} = \int_{\mathbf{n}} h_t(\mathbf{n}) p(\mathbf{n} - \hat{\mathbf{u}}_t) d\mathbf{n} \quad (6)$$

and

$$\mathbf{v}_t = \mathbb{E} [h_t(\mathbf{n} + \mathbf{u})] = \int_{\mathbf{n}} h_t(\mathbf{n}) p(\mathbf{n} - \mathbf{u}) d\mathbf{n} \quad (7)$$

From linearity of expectation and Cauchy-Schwarz inequality,

$$\begin{aligned} \|\mathbf{v}_t - \hat{\mathbf{v}}_t\| &\leq \int_{\mathbf{n}} \|h_t(\mathbf{n})\| |p(\mathbf{n} - \mathbf{u}) - p(\mathbf{n} - \hat{\mathbf{u}}_t)| d\mathbf{n} \\ &= \int_{\mathbf{n}} \|h_t(\mathbf{n}) - h_t(\mathbf{0})\| |p(\mathbf{n} - \mathbf{u}) - p(\mathbf{n} - \hat{\mathbf{u}}_t)| d\mathbf{n} \\ &\leq D \int_{\mathbf{n}} |p(\mathbf{n} - \mathbf{u}) - p(\mathbf{n} - \hat{\mathbf{u}}_t)| d\mathbf{n} \\ &\leq DL \|\mathbf{u} - \hat{\mathbf{u}}_t\| \\ &\leq \zeta DLG \sum_{s < t} \mathbb{I}_{\{s+d_s > t\}} \end{aligned} \quad (8)$$

The first inequality follows from the fact that  $h_t(\mathbf{n})$  and  $h_t(\mathbf{0})$  are in  $\mathcal{K}$ . The second inequality is due to the stability of the distribution of  $\mathbf{n}$ , where  $L$  is the Lipschitz constant of the probability distribution function  $p$ . Since each function is  $G$ -Lipschitz, the distance between  $\mathbf{u}$  and  $\hat{\mathbf{u}}_t$  is bounded by  $G$  multiplied by the number of functions whose feedback is not received at time  $t$ , which leads to the last inequality.  $\square$

**Lemma 5** ([3]). *For every  $t \in [T]$  and  $k \in [K]$ . Define  $h_{t,k} = f_t(\mathbf{x}_{t,k+1}) - f_t(\mathbf{x}^*)$  let  $A = \max(3, \frac{G}{\beta D})$  and  $\eta_k = \min(1, \frac{A}{k})$ , it holds that*

$$h_{t,k} = f_t(\mathbf{x}_{t,k+1}) - f_t(\mathbf{x}^*) \leq \frac{2\beta AD^2}{k} + \sum_{k'=1}^k \eta_{k'} \left( \prod_{\ell=k'+1}^k (1 - \eta_\ell) \right) \langle \nabla f_{t,k'}, \mathbf{v}_{t,k'} - \mathbf{x}^* \rangle \quad (9)$$

*Proof.* We state the proof of [3] for clarity. The proof is based on an induction on  $k$ . For  $k = 1$ ,  $\eta_1 = 1$ , we have  $h_{t,1} = f_t(\mathbf{x}_{t,2}) - f_t(\mathbf{x}^*) \leq GD$  since  $f_t$  is  $G$ -Lipschitz and the constraint set  $\mathcal{K}$  is bounded by  $D$  (Assumptions 1 and 2). More over, we have  $2\beta AD^2 + \langle \nabla f_{t,1}, \mathbf{x}_{t,1} - \mathbf{x}^* \rangle \geq 2\beta AD^2 - \langle \nabla f_{t,1}, \mathbf{x}_{t,1} - \mathbf{x}^* \rangle \geq 2\beta AD^2 - GD \geq GD$  by assuming  $A \geq \frac{G}{\beta D}$ . We have then  $h_{t,1} \leq 2\beta AD^2 + \langle \nabla f_{t,1}, \mathbf{x}_{t,1} - \mathbf{x}^* \rangle$ . Assume that the inequality holds for  $k - 1$ , we now prove for  $k$ . By definition of  $h_{t,k}$ , we have

$$\begin{aligned} h_{t,k} &\leq (1 - \eta_k)h_{t,k-1} + \eta_k \langle \nabla f_{t,k}, \mathbf{v}_{t,k} - \mathbf{x}^* \rangle + \eta_k^2 \frac{\beta D^2}{2} \\ &\leq (1 - \eta_k) \left[ \frac{2\beta AD^2}{k-1} + \sum_{k'=1}^{k-1} \eta_{k'} \left( \prod_{\ell=k'+1}^{k-1} (1 - \eta_\ell) \right) \langle \nabla f_{t,k'}, \mathbf{v}_{t,k'} - \mathbf{x}^* \rangle \right] \\ &\quad + \eta_k \langle \nabla f_{t,k}, \mathbf{v}_{t,k} - \mathbf{x}^* \rangle + \eta_k^2 \frac{\beta D^2}{2} \\ &\leq (1 - \eta_k) \left[ \frac{2\beta AD^2}{k-1} + \sum_{k'=1}^{k-1} \eta_{k'} \left[ \prod_{\ell=k'+1}^{k-1} (1 - \eta_\ell) \right] \langle \nabla f_{t,k'}, \mathbf{v}_{t,k'} - \mathbf{x}^* \rangle \right] \\ &\quad + \eta_k \prod_{\ell=k+1}^k (1 - \eta_\ell) \langle \nabla f_{t,k}, \mathbf{v}_{t,k} - \mathbf{x}^* \rangle + \eta_k^2 \frac{\beta D^2}{2} \\ &\leq (1 - \eta_k) \frac{2\beta AD^2}{k-1} + \sum_{k'=1}^{k-1} \eta_{k'} \left[ \prod_{\ell=k'+1}^k (1 - \eta_\ell) \right] \langle \nabla f_{t,k'}, \mathbf{v}_{t,k'} - \mathbf{x}^* \rangle \\ &\quad + \eta_k \prod_{\ell=k+1}^k (1 - \eta_\ell) \langle \nabla f_{t,k}, \mathbf{v}_{t,k} - \mathbf{x}^* \rangle + \eta_k^2 \frac{\beta D^2}{2} \\ &\leq (1 - \eta_k) \frac{2\beta AD^2}{k-1} + \sum_{k'=1}^k \eta_{k'} \left[ \prod_{\ell=k'+1}^k (1 - \eta_\ell) \right] \langle \nabla f_{t,k'}, \mathbf{v}_{t,k'} - \mathbf{x}^* \rangle + \eta_k^2 \frac{\beta D^2}{2} \\ &\leq \frac{2\beta AD^2}{k-1} - \frac{2\beta A^2 D^2}{k(k-1)} + \frac{\beta A^2 D^2}{2k^2} + \sum_{k'=1}^k \eta_{k'} \left[ \prod_{\ell=k'+1}^k (1 - \eta_\ell) \right] \langle \nabla f_{t,k'}, \mathbf{v}_{t,k'} - \mathbf{x}^* \rangle \\ &\leq \frac{2\beta AD^2}{k-1} - \frac{2\beta A^2 D^2}{k(k-1)} + \frac{\beta A^2 D^2}{2k(k-1)} + \sum_{k'=1}^k \eta_{k'} \left[ \prod_{\ell=k'+1}^k (1 - \eta_\ell) \right] \langle \nabla f_{t,k'}, \mathbf{v}_{t,k'} - \mathbf{x}^* \rangle \\ &\leq \frac{2\beta AD^2}{k-1} - \frac{2\beta A^2 D^2 - \frac{\beta}{2} A^2 D^2}{k(k-1)} + \sum_{k'=1}^k \eta_{k'} \left[ \prod_{\ell=k'+1}^k (1 - \eta_\ell) \right] \langle \nabla f_{t,k'}, \mathbf{v}_{t,k'} - \mathbf{x}^* \rangle \\ &\leq \frac{2\beta AD^2}{k-1} - \frac{\beta A^2 D^2}{k(k-1)} + \sum_{k'=1}^k \eta_{k'} \left[ \prod_{\ell=k'+1}^k (1 - \eta_\ell) \right] \langle \nabla f_{t,k'}, \mathbf{v}_{t,k'} - \mathbf{x}^* \rangle \\ &\leq \frac{2\beta AD^2}{k} + \sum_{k'=1}^k \eta_{k'} \left[ \prod_{\ell=k'+1}^k (1 - \eta_\ell) \right] \langle \nabla f_{t,k'}, \mathbf{v}_{t,k'} - \mathbf{x}^* \rangle \end{aligned}$$

where the last inequality follows from the fact that  $\beta A^2 \geq 2\beta A$  for  $A \geq 3$  and  $\frac{1}{k-1} - \frac{1}{k(k-1)} \leq \frac{1}{k}$ .  $\square$

## A.2 Proof of Theorem 1

**Theorem 1.** Given a constraint set  $\mathcal{K}$ . Let  $A = \max\left\{3, \frac{G}{\beta D}\right\}$ ,  $\eta_k = \min\left\{1, \frac{A}{k}\right\}$ , and  $K = \sqrt{T}$ . Suppose that Assumptions 1 to 3 hold true. If we choose FTPL as the underlying oracle and set  $\zeta = \frac{1}{G\sqrt{B}}$ , the regret of Algorithm 1 is

$$\sum_{t=1}^T [f_t(\mathbf{x}_t) - f_t(\mathbf{x}^*)] \leq 2\beta AD^2\sqrt{T} + 3(A+1) \left( DG\sqrt{B} + \mathcal{R}_{T,\mathcal{O}} \right) \quad (10)$$

where  $B = \sum_{t=1}^T d_t$ , the sum of all delay values and  $\mathcal{R}_{T,\mathcal{O}}$  is the regret of FTPL with respect to the current choice of  $\zeta$

*Proof.* Let  $\mathbf{x}_{t,1}, \dots, \mathbf{x}_{t,K+1}$  be the sequence of sub-iterate for a fixed time step  $t$ . Using Frank-Wolfe updates and smoothness of  $f_t$ , we have

$$\begin{aligned} f_t(\mathbf{x}_{t,k+1}) - f_t(\mathbf{x}^*) &= f_t(\mathbf{x}_{t,k} + \eta_k(\mathbf{v}_{t,k} - \mathbf{x}_{t,k})) - f_t(\mathbf{x}^*) \\ &\leq f_t(\mathbf{x}_{t,k}) - f_t(\mathbf{x}^*) + \eta_k \langle \nabla f_{t,k}, \mathbf{v}_{t,k} - \mathbf{x}_{t,k} \rangle + \eta_k^2 \frac{\beta}{2} \|\mathbf{v}_{t,k} - \mathbf{x}_{t,k}\|^2 \\ &\leq f_t(\mathbf{x}_{t,k}) - f_t(\mathbf{x}^*) + \eta_k \langle \nabla f_{t,k}, \mathbf{v}_{t,k} - \mathbf{x}_{t,k} \rangle + \eta_k^2 \frac{\beta D^2}{2} \quad (\mathcal{K} \text{ is bounded}) \\ &\leq f_t(\mathbf{x}_{t,k}) - f_t(\mathbf{x}^*) + \eta_k [\langle \nabla f_{t,k}, \mathbf{v}_{t,k} - \mathbf{x}^* \rangle + \langle \nabla f_{t,k}, \mathbf{x}^* - \mathbf{x}_{t,k} \rangle] + \eta_k^2 \frac{\beta D^2}{2} \\ &\leq f_t(\mathbf{x}_{t,k}) - f_t(\mathbf{x}^*) + \eta_k [\langle \nabla f_{t,k}, \mathbf{v}_{t,k} - \mathbf{x}^* \rangle + f_t(\mathbf{x}^*) - f_t(\mathbf{x}_{t,k})] + \eta_k^2 \frac{\beta D^2}{2} \quad (\text{by convexity of } f_t) \\ &\leq (1 - \eta_k) [f_t(\mathbf{x}_{t,k}) - f_t(\mathbf{x}^*)] + \eta_k \langle \nabla f_{t,k}, \mathbf{v}_{t,k} - \mathbf{x}^* \rangle + \eta_k^2 \frac{\beta D^2}{2} \end{aligned} \quad (11)$$

Let  $h_{t,k} = f_t(\mathbf{x}_{t,k+1}) - f_t(\mathbf{x}^*)$ , eq. (11) becomes

$$h_{t,k} \leq (1 - \eta_k)h_{t,k-1} + \eta_k \langle \nabla f_{t,k}, \mathbf{v}_{t,k} - \mathbf{x}^* \rangle + \eta_k^2 \frac{\beta D^2}{2} \quad (12)$$

A direct application of Lemma 5 for  $k = K$  yields

$$f_t(\mathbf{x}_{t,K+1}) - f_t(\mathbf{x}^*) \leq \frac{2\beta AD^2}{K} + \sum_{k'=1}^K \eta_{k'} \left[ \prod_{\ell=k'+1}^K (1 - \eta_\ell) \right] \langle \nabla f_{t,k}, \mathbf{v}_{t,k} - \mathbf{x}^* \rangle \quad (13)$$

Following the notation from Algorithm 1 and Lemma 2. For a fixed time  $t$  and any sub-iterate  $k$ ,  $\mathbf{v}_{t,k}$  and  $\hat{\mathbf{v}}_{t,k}$  are respectively the predictions of the oracle  $\mathcal{O}_k$  under delayed and non-delayed feedback, the scalar product of eq. (13) over  $T$ -round is written as

$$\begin{aligned} \sum_{t=1}^T \langle \nabla f_{t,k}, \mathbf{v}_{t,k} - \mathbf{x}^* \rangle &= \sum_{t=1}^T \langle \nabla f_{t,k}, \mathbf{v}_{t,k} - \hat{\mathbf{v}}_{t,k} \rangle + \sum_{t=1}^T \langle \nabla f_{t,k}, \hat{\mathbf{v}}_{t,k} - \mathbf{x}^* \rangle \\ &\leq \sum_{t=1}^T \langle \nabla f_{t,k}, \mathbf{v}_{t,k} - \hat{\mathbf{v}}_{t,k} \rangle + \sum_{t=1}^T \langle \nabla f_{t,k}, \hat{\mathbf{v}}_{t,k} - \mathbf{x}^* \rangle \end{aligned} \quad (14)$$

In the first term on the right hand side of Equation (14), using the Cauchy-Schwartz inequality and Lemma 2, we have

$$\sum_{t=1}^T \langle \nabla f_{t,k}, \mathbf{v}_{t,k} - \hat{\mathbf{v}}_{t,k} \rangle \leq \sum_{t=1}^T \|\nabla f_{t,k}\| \|\mathbf{v}_{t,k} - \hat{\mathbf{v}}_{t,k}\| \leq \zeta DG^2 \sum_{t=1}^T \sum_{s < t} \mathbb{1}_{s+d_s > t} \leq \zeta DG^2 B \quad (15)$$

Recall that the objective function of the oracle  $\mathcal{O}_k$  in non-delay setting is  $\langle \nabla f_{t,k}, \cdot \rangle$  and  $\hat{\mathbf{v}}_{t,k}$  is its prediction at time  $t$ , the second term of Equation (14) is bounded by the regret of  $\mathcal{O}_k$  in non-delay setting,  $\mathcal{R}_{T,\mathcal{O}}$ . Specifically, we have

$$\sum_{t=1}^T \langle \nabla f_{t,k}, \hat{\mathbf{v}}_{t,k} \rangle \leq \min_{\mathbf{x} \in \mathcal{K}} \sum_{t=1}^T \langle \nabla f_{t,k}, \mathbf{x} \rangle + \mathcal{R}_{T,\mathcal{O}} \leq \sum_{t=1}^T \langle \nabla f_{t,k}, \mathbf{x}^* \rangle + \mathcal{R}_{T,\mathcal{O}} \quad (16)$$

Recall that  $\mathbf{x}_t = \mathbf{x}_{t,K+1}$ , combining Equations (14) to (16) and summing Equation (13) over  $T$ -rounds yields

$$\sum_{t=1}^T [f_t(\mathbf{x}_t) - f_t(\mathbf{x}^*)] \leq \frac{2\beta AD^2}{K} T + \sum_{k'=1}^K \eta_{k'} \left[ \prod_{\ell=k'+1}^K (1 - \eta_\ell) \right] [\zeta DG^2 B + \mathcal{R}_{T,\mathcal{O}}] \quad (17)$$

Let  $\eta_k = \frac{A}{k}$ , we have

$$\prod_{k'=k+1}^K (1 - \eta_{k'}) \leq e^{-\sum_{k'=k+1}^K \eta_{k'}} \leq e^{-\sum_{k'=k+1}^K \frac{A}{k'}} \leq e^{-A \int_{k+2}^K ds/s} \leq \left( \frac{k+2}{K} \right)^A \quad (18)$$

We have then,

$$\begin{aligned} \sum_{k=1}^K \eta_k \left[ \prod_{k'=k+1}^K (1 - \eta_{k'}) \right] &\leq \min \left\{ 1, \frac{A}{K} \right\} + \min \left\{ 1, \frac{A}{K-1} \right\} + \min \left\{ 1, \frac{A}{K-2} \right\} + \sum_{k=1}^{K-3} \frac{A}{k} \left[ \frac{k+2}{K} \right]^A \\ &\leq 3 \min \left\{ 1, \frac{A}{K-2} \right\} + \frac{A}{K} \sum_{k=1}^{K-3} \frac{k+2}{k} \left[ \frac{k+2}{K} \right]^{A-1} \\ &\leq 3 + \frac{3A}{K} \sum_{k=1}^{K-3} \left[ \frac{k+2}{K} \right]^{A-1} \\ &\leq 3(A+1) \end{aligned} \quad (19)$$

From Equation (19), we deduce that

$$\begin{aligned} \sum_{t=1}^T [f_t(\mathbf{x}_t) - f_t(\mathbf{x}^*)] &\leq \frac{2\beta AD^2}{K} T + \sum_{k'=1}^K \eta_{k'} \left( \prod_{\ell=k'+1}^K (1 - \eta_\ell) \right) (\zeta DG^2 B + \mathcal{R}_{T,\mathcal{O}}) \\ &\leq \frac{2\beta AD^2}{K} T + 3(A+1) (\zeta DG^2 B + \mathcal{R}_{T,\mathcal{O}}) \end{aligned} \quad (20)$$

The theorem follows by letting  $\zeta = \frac{1}{G\sqrt{B}}$ ,  $K = \sqrt{T}$  and choosing the oracle as FTPL with regret  $\mathcal{R}_{T,\mathcal{O}}$ .  $\square$

## B Analysis of Algorithm 2

### B.1 Additional Notations (only for the analysis)

In the analysis, for any vector  $\mathbf{x}^i \in \mathbb{R}^d, \forall i \in [n]$ , we note  $\mathbf{x}^{cat} \in \mathbb{R}^{dn}$  a column vector defined as  $\mathbf{x}^{cat} := [\mathbf{x}^{1\top}, \dots, \mathbf{x}^{n\top}]^\top$ . Let  $\bar{\mathbf{x}}$  be the average of  $\mathbf{x}^i$  over  $i \in [n]$ , the vector  $\bar{\mathbf{x}}^{cat}$  is a  $dn$ -vector where we stack  $n$ -times  $\bar{\mathbf{x}}$  i.e  $\bar{\mathbf{x}}^{cat} := [\bar{\mathbf{x}}^\top, \dots, \bar{\mathbf{x}}^\top]$ . For simplicity of notation, we note  $\nabla f_t^i(\mathbf{x}_{t,k}^i) := \nabla f_{t,k}^i$  and  $\nabla F_{t,k} := \frac{1}{n} \sum_{i=1}^n \nabla f_{t,k}^i$ . In order to incorporate the delay of agents at each time-step  $t$ , we define  $\nabla f_{t,k}^{cat}$  as described above using the sum of agent's delay feedback, we note then

$$\nabla f_{t,k}^{cat} = \left[ \sum_{s \in \mathcal{F}_t^1} \nabla f_{s,k}^{1\top}, \dots, \sum_{s \in \mathcal{F}_t^n} \nabla f_{s,k}^{n\top} \right]^\top \quad (21)$$

and its homologous in the non-delay setting by  $\nabla \hat{f}_{t,k}^{cat} = [\nabla f_{t,k}^{1\top}, \dots, \nabla f_{t,k}^{n\top}]^\top$

The variables  $\mathbf{d}_{t,k}^{cat}$ ,  $\hat{\mathbf{d}}_{t,k}^{cat}$  and  $\mathbf{g}_{t,k}^{cat}$ ,  $\hat{\mathbf{g}}_{t,k}^{cat}$  are defined similarly as described. Lastly, we define the slack variable  $\delta_{t,k}^i := \nabla f_{t,k}^i - \nabla f_{t,k-1}^i$ , then the definition of  $\bar{\delta}_{t,k}$ ,  $\delta_{t,k}^{cat}$  and  $\bar{\delta}_{t,k}^{cat}$  followed.

### B.2 Auxiliary lemmas

**Claim 1.** *In the analysis, we make use of the following bounds*

$$\|\bar{\mathbf{x}}_{t,k} - \mathbf{x}_{t,k}^i\| \leq \frac{2C_d}{k}$$

$$\|\mathbf{x}_{t,k+1}^i - \mathbf{x}_{t,k}^i\| \leq \frac{4C_d + AD}{k}$$

*Proof of claim.* For the first bound, recall the definition of FW-update in Algorithm 2 and using Lemma 3, we have

$$\begin{aligned} \|\bar{\mathbf{x}}_{t,k} - \mathbf{x}_{t,k}^i\| &= \|(1 - \eta_{k-1})(\bar{\mathbf{x}}_{t,k-1} - \mathbf{y}_{t,k-1}^i) + \eta_{k-1}(\bar{\mathbf{v}}_{t,k-1} - \mathbf{v}_{t,k-1}^i)\| \\ &\leq \frac{C_d}{k-1} - \frac{AC_d}{(k-1)^2} + \frac{AD}{k-1} \leq \frac{C_d}{k-1} - \left[ \frac{AC_d}{(k-1)^2} - \frac{AD}{k-1} \right] \leq \frac{C_d}{k-1} - \left[ \frac{AC_d - AD}{(k-1)^2} \right] \leq \frac{C_d}{k-1} \leq \frac{2C_d}{k} \end{aligned}$$

Applying the first bound on the second one yields

$$\|\mathbf{x}_{t,k+1}^i - \mathbf{x}_{t,k}^i\| \leq \|\mathbf{x}_{t,k+1}^i - \bar{\mathbf{x}}_{t,k+1}\| + \|\bar{\mathbf{x}}_{t,k+1} - \bar{\mathbf{x}}_{t,k}\| + \|\bar{\mathbf{x}}_{t,k} - \mathbf{x}_{t,k}^i\| \leq \frac{2C_d}{k+1} + \frac{AD}{k} + \frac{2C_d}{k} \leq \frac{4C_d + AD}{k}$$

□

**Lemma 3.** Define  $C_d = k_0\sqrt{n}D$ , for all  $t \in [T]$ ,  $k \in [K]$ , we have

$$\max_{i \in [1,n]} \|\mathbf{y}_{t,k}^i - \bar{\mathbf{x}}_{t,k}\| \leq \frac{C_d}{k} \quad (22)$$

*Proof.* We prove the lemma by induction, we first note that

$$\begin{aligned} \|\mathbf{y}_{t,k}^{cat} - \bar{\mathbf{x}}_{t,k}^{cat}\| &= \left\| (\mathbf{W} \otimes I_d) \mathbf{x}_{t,k}^{cat} - \left( \frac{1}{n} \mathbf{1}_n \mathbf{1}_n^T \right) \mathbf{x}_{t,k}^{cat} \right\| \\ &= \left\| \left[ \left( \mathbf{W} - \frac{1}{n} \mathbf{1}_n \mathbf{1}_n^T \right) \otimes I_d \right] \mathbf{x}_{t,k}^{cat} \right\| \\ &= \left\| \left[ \left( \mathbf{W} - \frac{1}{n} \mathbf{1}_n \mathbf{1}_n^T \right) \otimes I_d \right] (\mathbf{x}_{t,k}^{cat} - \bar{\mathbf{x}}_{t,k}^{cat}) \right\| \\ &\leq \lambda(\mathbf{W}) \|\mathbf{x}_{t,k}^{cat} - \bar{\mathbf{x}}_{t,k}^{cat}\| \end{aligned} \quad (23)$$

Let  $C_d = k_0\sqrt{n}D$ , the base case is verified for  $k \in [1, k_0]$  since  $\|\mathbf{x}_{t,k}^{cat} - \bar{\mathbf{x}}_{t,k}^{cat}\| \leq \sqrt{n}D \leq \frac{C_d}{k}$ . Suppose that the hypothesis is verified for  $k-1 \geq k_0$ , we have

$$\begin{aligned} \|\mathbf{y}_{t,k}^{cat} - \bar{\mathbf{x}}_{t,k}^{cat}\| &\leq \lambda(\mathbf{W}) \|\mathbf{x}_{t,k}^{cat} - \bar{\mathbf{x}}_{t,k}^{cat}\| \\ &= \lambda(\mathbf{W}) \|(1 - \eta_{k-1})(\mathbf{y}_{t,k-1}^{cat} - \bar{\mathbf{x}}_{t,k-1}^{cat}) + \eta_{k-1}(\mathbf{v}_{t,k-1}^{cat} - \bar{\mathbf{v}}_{t,k-1}^{cat})\| \\ &\leq \lambda(\mathbf{W}) \|\mathbf{y}_{t,k-1}^{cat} - \bar{\mathbf{x}}_{t,k-1}^{cat}\| + \lambda(\mathbf{W}) \frac{\sqrt{n}D}{k-1} \\ &\leq \lambda(\mathbf{W}) \left( \frac{C_d + \sqrt{n}D}{k-1} \right) \\ &\leq \lambda(\mathbf{W}) C_d \frac{k_0 + 1}{k_0(k-1)} \\ &\leq \frac{C_d}{k} \end{aligned} \quad (24)$$

where we use the induction hypothesis in the third inequality and the last inequality follows the fact that  $\lambda(\mathbf{W}) \frac{k_0+1}{k_0(k-1)} \leq \frac{k-1}{k} \cdot \frac{1}{k-1} \leq \frac{1}{k}$ . We conclude the proof by noting that

$$\max_{i \in [1,n]} \|\mathbf{y}_{t,k}^i - \bar{\mathbf{x}}_{t,k}\| \leq \sqrt{\sum_{i=1}^n \|\mathbf{y}_{t,k}^i - \bar{\mathbf{x}}_{t,k}\|^2} = \|\mathbf{y}_{t,k}^{cat} - \bar{\mathbf{x}}_{t,k}^{cat}\| \leq \frac{C_d}{k}$$

□

**Lemma 6.** Define  $C_g = \sqrt{n} \max \left\{ \lambda(\mathbf{W}) \left( G + \frac{\beta D}{1 - \lambda(\mathbf{W})} \right), k_0 \beta (4C_d + AD) \right\}$  and recall the definition of  $\nabla F_{t,k} := \frac{1}{n} \sum_{i=1}^n \nabla f_{t,k}^i$ . For all  $t \in [T]$ ,  $k \in [K]$ , we have

$$\max_{i \in [1,n]} \|\mathbf{d}_{t,k}^i - \nabla F_{t,k}\| \leq \frac{C_g}{k} \quad (25)$$

*Proof.* We prove the lemma by induction. Following the idea from [?], we have

$$\begin{aligned}
\|\hat{\mathbf{d}}_{t,k}^{cat} - \nabla F_{t,k}^{cat}\| &= \left\| (\mathbf{W} \otimes I_d) \hat{\mathbf{g}}_{t,k}^{cat} - \left( \frac{1}{n} \mathbf{1}_n \mathbf{1}_n^T \right) \hat{\mathbf{g}}_{t,k}^{cat} \right\| \\
&= \left\| \left[ \left( \mathbf{W} - \frac{1}{n} \mathbf{1}_n \mathbf{1}_n^T \right) \otimes I_d \right] \hat{\mathbf{g}}_{t,k}^{cat} \right\| \\
&= \left\| \left[ \left( \mathbf{W} - \frac{1}{n} \mathbf{1}_n \mathbf{1}_n^T \right) \otimes I_d \right] (\hat{\mathbf{g}}_{t,k}^{cat} - \nabla F_{t,k}^{cat}) \right\| \\
&\leq \lambda(\mathbf{W}) \|\hat{\mathbf{g}}_{t,k}^{cat} - \nabla F_{t,k}^{cat}\|
\end{aligned} \tag{26}$$

where the third equality and the last inequality are verified since  $\mathbf{W} \cdot \nabla F_{t,k}^{cat} = \frac{1}{n} \mathbf{1}_n \mathbf{1}_n^T \cdot \nabla F_{t,k}^{cat} = \nabla F_{t,k}^{cat}$  and  $\|\mathbf{W} - \frac{1}{n} \mathbf{1}_n \mathbf{1}_n^T\| \leq \lambda(\mathbf{W})$  [16, Lemma 16]. Using the gradient tracking step and Equation (26), we have

$$\begin{aligned}
\|\hat{\mathbf{d}}_{t,k}^{cat} - \nabla F_{t,k}^{cat}\| &\leq \lambda(\mathbf{W}) \|\hat{\mathbf{g}}_{t,k}^{cat} - \nabla F_{t,k}^{cat}\| \\
&= \lambda(\mathbf{W}) \left\| \delta_{t,k}^{cat} + \hat{\mathbf{d}}_{t,k-1}^{cat} - \nabla F_{t,k}^{cat} + \nabla F_{t,k-1}^{cat} - \nabla F_{t,k-1}^{cat} \right\| \\
&\leq \lambda(\mathbf{W}) \left( \|\hat{\mathbf{d}}_{t,k-1}^{cat} - \nabla F_{t,k-1}^{cat}\| + \|\delta_{t,k}^{cat} - \bar{\delta}_{t,k}^{cat}\| \right) \\
&\leq \lambda(\mathbf{W}) \left( \|\hat{\mathbf{d}}_{t,k-1}^{cat} - \nabla F_{t,k-1}^{cat}\| + \|\delta_{t,k}^{cat}\| \right)
\end{aligned} \tag{27}$$

where the last inequality holds since

$$\|\delta_{t,k}^{cat} - \bar{\delta}_{t,k}^{cat}\|^2 = \sum_{i=1}^n \|\delta_{t,k}^i - \bar{\delta}_{t,k}\|^2 = \sum_{i=1}^n \|\delta_{t,k}^i\|^2 - n \|\bar{\delta}_{t,k}\|^2 \leq \sum_{i=1}^n \|\delta_{t,k}^i\|^2 = \|\delta_{t,k}^{cat}\|^2 \tag{28}$$

Moreover, using the smoothness of  $f_t$  and Claim 1, we have

$$\begin{aligned}
\|\delta_{t,k}^{cat}\|^2 &= \sum_{i=1}^n \|\delta_{t,k}^i\|^2 \leq \sum_{i=1}^n \|\nabla f_{t,k}^i - \nabla f_{t,k-1}^i\|^2 \leq \sum_{i=1}^n \beta^2 \|\mathbf{x}_{t,k}^i - \mathbf{x}_{t,k-1}^i\|^2 \\
&\leq n\beta^2 \left( \frac{4C_d + AD}{k-1} \right)^2
\end{aligned} \tag{29}$$

Thus, we have  $\|\delta_{t,k}^{cat}\| \leq \sqrt{n}\beta \frac{4C_d + AD}{k-1}$ . For the base case  $k = 1$ , we have

$$\|\hat{\mathbf{d}}_{t,1}^{cat} - \nabla F_{t,1}^{cat}\|^2 = \left\| \left[ \left( \mathbf{W} - \frac{1}{n} \mathbf{1}_n \mathbf{1}_n^T \right) \otimes I_d \right] \hat{\mathbf{g}}_{t,1}^{cat} \right\|^2 \leq \lambda^2(\mathbf{W}) \|\hat{\mathbf{g}}_{t,1}^{cat}\|^2 \leq \lambda^2(\mathbf{W}) \sum_{i=1}^n \|\nabla f_{t,1}^i\|^2 \leq n\lambda^2(\mathbf{W})G^2 \tag{30}$$

We have then  $\|\hat{\mathbf{d}}_{t,1}^{cat} - \nabla F_{t,1}^{cat}\| \leq \sqrt{n}\lambda(\mathbf{W})G$ . For  $k \in (1, k_0]$ , by Equation (27)

$$\begin{aligned}
\|\hat{\mathbf{d}}_{t,k}^{cat} - \nabla F_{t,k}^{cat}\| &\leq \lambda(\mathbf{W}) \left( \|\hat{\mathbf{d}}_{t,k-1}^{cat} - \nabla F_{t,k-1}^{cat}\| + \|\delta_{t,k}^{cat}\| \right) \\
&\leq \lambda(\mathbf{W}) \left( \|\hat{\mathbf{d}}_{t,k-1}^{cat} - \nabla F_{t,k-1}^{cat}\| + \sqrt{n\beta^2 D^2} \right) \\
&\leq \lambda^{k-1}(\mathbf{W}) \|\hat{\mathbf{d}}_{t,1}^{cat} - \nabla F_{t,1}^{cat}\| + \sum_{\tau=1}^k \lambda^\tau(\mathbf{W}) \sqrt{n}\beta D \\
&\leq \lambda^k(\mathbf{W}) \sqrt{n}G + \frac{\lambda(\mathbf{W})}{1 - \lambda(\mathbf{W})} \sqrt{n}\beta D \\
&\leq \lambda(\mathbf{W}) \sqrt{n} \left( G + \frac{\beta D}{1 - \lambda(\mathbf{W})} \right)
\end{aligned}$$

where in the second inequality, we use smoothness of  $f_t$  and bound the distance  $\|\mathbf{x}_{t,k}^i - \mathbf{x}_{t,k-1}^i\|$  by the diameters  $D$ . The third inequality resulted from applying the previous one recursively for  $k \in \{1, \dots, k-1\}$ . Using Taylor's expansion of  $\lambda(\mathbf{W})$  and the bound in Equation (30), we obtain the fourth inequality. Let  $C_g =$



$\sqrt{n} \max \left\{ \lambda(\mathbf{W}) \left( G + \frac{\beta D}{1 - \lambda(\mathbf{W})} \right), k_0 \beta (4C_d + AD) \right\}$ . We claim that  $\left\| \hat{\mathbf{d}}_{t,k-1}^{cat} - \nabla F_{t,k-1}^{cat} \right\| \leq \frac{C_g}{k-1}$  for all  $k-1 \geq k_0$ . We prove the claim for round  $k$ . Using Equation (27), we have

$$\begin{aligned}
\left\| \hat{\mathbf{d}}_{t,k}^{cat} - \nabla F_{t,k}^{cat} \right\| &\leq \lambda(\mathbf{W}) \left( \left\| \hat{\mathbf{d}}_{t,k-1}^{cat} - \nabla F_{t,k-1}^{cat} \right\| + \left\| \boldsymbol{\delta}_{t,k}^{cat} \right\| \right) \\
&\leq \lambda(\mathbf{W}) \left( \frac{C_g}{k-1} + \sqrt{n} \beta \frac{4C_d + AD}{k-1} \right) \\
&\leq \lambda(\mathbf{W}) \left( \frac{C_g + \sqrt{n} \beta (4C_d + AD)}{k-1} \right) \\
&\leq \lambda(\mathbf{W}) \left( C_g \frac{k_0 + 1}{k_0(k-1)} \right) \\
&\leq \frac{C_g}{k}
\end{aligned} \tag{31}$$

where the second inequality followed by the induction hypothesis and Equation (29). The fourth inequality is a consequence of the definition of  $C_g$  and the final inequality resulted from the fact that  $\lambda(\mathbf{W}) \frac{k_0+1}{k_0(k-1)} \leq \frac{1}{k}$  as  $k > k_0$ . We conclude the proof by noting that

$$\max_{i \in [1, n]} \left\| \hat{\mathbf{d}}_{t,k}^i - \nabla F_{t,k} \right\| \leq \sqrt{\sum_{i=1}^n \left\| \hat{\mathbf{d}}_{t,k}^i - \nabla F_{t,k} \right\|^2} = \left\| \hat{\mathbf{d}}_{t,k}^{cat} - \nabla F_{t,k}^{cat} \right\| \leq \frac{C_g}{k}$$

□

**Lemma 4.** For all  $t \in [T]$ ,  $k \in [K]$  and  $i \in [n]$ . Let  $\mathbf{v}_{t,k}^i$  be the output of the oracle  $\mathcal{O}_k^i$  with delayed feedback and  $\hat{\mathbf{v}}_{t,k}^i$  its homologous in non-delay case. Suppose that Assumptions 1 and 2 hold true. Choosing FTPL as the oracle, we have:

$$\left\| \mathbf{v}_{t,k}^i - \hat{\mathbf{v}}_{t,k}^i \right\| \leq 2\zeta \sqrt{n} D G \left[ \frac{\lambda(\mathbf{W})}{\rho} + 1 \right] \frac{1}{n} \sum_{i=1}^n \sum_{s \leq t} \mathbb{I}_{\{s + d_s^i > t\}} \tag{32}$$

where  $\zeta$  is the learning rate,  $\lambda(\mathbf{W})$  is the second-largest eigenvalue of  $\mathbf{W}$  and  $\rho = 1 - \lambda(\mathbf{W})$  is the spectral gap of matrix  $\mathbf{W}$ .

*Proof.* We call  $\mathbf{u}_t^i = \zeta \sum_{\ell=1}^t \mathbf{d}_\ell^i$  the accumulated delayed feedback of the oracle of agent  $i$  and  $\hat{\mathbf{u}}_t^i = \zeta \sum_{\ell=1}^t \hat{\mathbf{d}}_\ell^i$  its homologous in non-delay setting. Using the same computation in the proof of Lemma 2, we have

$$\left\| \mathbf{v}_{t,k}^i - \hat{\mathbf{v}}_{t,k}^i \right\| \leq D \left\| \mathbf{u}_{t,k}^i - \hat{\mathbf{u}}_{t,k}^i \right\| \leq D \left\| \mathbf{u}_{t,k}^{cat} - \hat{\mathbf{u}}_{t,k}^{cat} \right\| \leq \zeta D \left\| \sum_{\ell=1}^t \left[ \mathbf{d}_{\ell,k}^{cat} - \hat{\mathbf{d}}_{\ell,k}^{cat} \right] \right\| \tag{33}$$

For  $t \in [T]$ , the concatenation of the gradient local average is expanded as follow

$$\begin{aligned}
\mathbf{d}_{t,k}^{cat} &= (\mathbf{W} \otimes I_d) (\nabla f_{t,k}^{cat} - \nabla f_{t,k-1}^{cat} + \mathbf{d}_{t,k-1}^{cat}) \\
&= (\mathbf{W} \otimes I_d) (\nabla f_{t,k}^{cat} - \nabla f_{t,k-1}^{cat}) + (\mathbf{W} \otimes I_d)^2 (\nabla f_{t,k-1}^{cat} - \nabla f_{t,k-2}^{cat} + \mathbf{d}_{t,k-2}^{cat}) \\
&= \sum_{\tau=1}^{k-1} \left[ (\mathbf{W} \otimes I_d)^{k-\tau} (\nabla f_{t,\tau+1}^{cat} - \nabla f_{t,\tau}^{cat}) \right] + (\mathbf{W} \otimes I_d)^k \nabla f_{t,1}^{cat} \\
&= \sum_{\tau=1}^{k-1} \left[ (\mathbf{W} \otimes I_d)^{k-\tau} (\nabla f_{t,\tau+1}^{cat} - \nabla f_{t,\tau}^{cat}) \right] + (\mathbf{W} \otimes I_d)^k \nabla f_{t,1}^{cat} - \sum_{\tau=1}^{k-1} [\nabla F_{t,\tau+1}^{cat} - \nabla F_{t,\tau}^{cat}] - \nabla F_{t,1}^{cat} + \nabla F_{t,k}^{cat} \\
&= \sum_{\tau=1}^{k-1} \left[ \left[ \left( \mathbf{W}^{k-\tau} - \frac{1}{n} \mathbf{1}_n \mathbf{1}_n^T \right) \otimes I_d \right] (\nabla f_{t,\tau+1}^{cat} - \nabla f_{t,\tau}^{cat}) \right] + \left[ \left( \mathbf{W}^k - \frac{1}{n} \mathbf{1}_n \mathbf{1}_n^T \right) \otimes I_d \right] \nabla f_{t,1}^{cat} + \left( \frac{1}{n} \mathbf{1}_n \mathbf{1}_n^T \otimes I_d \right) \nabla f_{t,k}^{cat}
\end{aligned} \tag{34}$$

Recall the definition of  $\nabla f_{t,k}^{cat} = \left[ \sum_{s \in \mathcal{F}_t^1} \nabla f_{s,k}^{1\top}, \dots, \sum_{s \in \mathcal{F}_t^n} \nabla f_{s,k}^{n\top} \right]^\top$  and  $\nabla \hat{f}_{t,k}^{cat} = \left[ \nabla f_{t,k}^{1\top}, \dots, \nabla f_{t,k}^{n\top} \right]^\top$ . We have

$$\sum_{\ell=1}^t \left[ \nabla f_{\ell,k}^{cat} - \nabla \hat{f}_{\ell,k}^{cat} \right] = \sum_{s < t} \left[ \nabla f_{s,k}^{1\top} \mathbb{I}_{\{s + d_s^1 > t\}}, \dots, \nabla f_{s,k}^{n\top} \mathbb{I}_{\{s + d_s^n > t\}} \right]^\top \tag{35}$$

Using Equation (34), we bound the RHS of Equation (33) as follow

$$\begin{aligned}
& \left\| \sum_{\ell=1}^t \left[ \mathbf{d}_{\ell,k}^{cat} - \hat{\mathbf{d}}_{\ell,k}^{cat} \right] \right\| \leq \sum_{\tau=1}^{k-1} \left\| \left( \mathbf{W}^{k-\tau} - \frac{1}{n} \mathbf{1}_n \mathbf{1}_n^T \right) \otimes I_d \right\| \left\| \sum_{\ell=1}^t \left[ \nabla f_{\ell,\tau+1}^{cat} - \hat{f}_{\ell,\tau+1}^{cat} + \hat{f}_{\ell,\tau}^{cat} - \nabla f_{\ell,\tau}^{cat} \right] \right\| \\
& \quad + \left\| \left( \mathbf{W}^k - \frac{1}{n} \mathbf{1}_n \mathbf{1}_n^T \right) \otimes I_d \right\| \left\| \sum_{\ell=1}^t \left[ \nabla f_{\ell,1}^{cat} - \hat{f}_{\ell,1}^{cat} \right] \right\| + \left( \frac{1}{n} \mathbf{1}_n \mathbf{1}_n^T \otimes I_d \right) \left\| \sum_{\ell=1}^t \left[ \nabla f_{\ell,k}^{cat} - \hat{f}_{\ell,k}^{cat} \right] \right\| \\
& \leq 2 \sum_{\tau=1}^k \lambda^{k-\tau}(\mathbf{W}) \left\| \sum_{s \leq t} \left[ \nabla f_{s,\tau}^{1\top} \mathbb{I}_{\{s+d_s^1 > t\}}, \dots, \nabla f_{s,\tau}^{n\top} \mathbb{I}_{\{s+d_s^n > t\}} \right]^\top \right\| \\
& \quad + \lambda^k(\mathbf{W}) \left\| \sum_{s \leq t} \left[ \nabla f_{s,k}^{1\top} \mathbb{I}_{\{s+d_s^1 > t\}}, \dots, \nabla f_{s,k}^{n\top} \mathbb{I}_{\{s+d_s^n > t\}} \right]^\top \right\| + \left\| \sum_{s \leq t} \left[ \nabla f_{s,1}^{1\top} \mathbb{I}_{\{s+d_s^1 > t\}}, \dots, \nabla f_{s,1}^{n\top} \mathbb{I}_{\{s+d_s^n > t\}} \right]^\top \right\| \\
& \leq 2 \sum_{\tau=1}^k \lambda^{k-\tau}(\mathbf{W}) \sum_{s \leq t} \left\| \left[ \nabla f_{s,\tau}^{1\top} \mathbb{I}_{\{s+d_s^1 > t\}}, \dots, \nabla f_{s,\tau}^{n\top} \mathbb{I}_{\{s+d_s^n > t\}} \right]^\top \right\| \\
& \quad + \lambda^k(\mathbf{W}) \sum_{s \leq t} \left\| \left[ \nabla f_{s,k}^{1\top} \mathbb{I}_{\{s+d_s^1 > t\}}, \dots, \nabla f_{s,k}^{n\top} \mathbb{I}_{\{s+d_s^n > t\}} \right]^\top \right\| + \sum_{s \leq t} \left\| \left[ \nabla f_{s,1}^{1\top} \mathbb{I}_{\{s+d_s^1 > t\}}, \dots, \nabla f_{s,1}^{n\top} \mathbb{I}_{\{s+d_s^n > t\}} \right]^\top \right\| \\
& \leq 2 \sum_{\tau=1}^k \lambda^{k-\tau}(\mathbf{W}) \sum_{s \leq t} \sqrt{\sum_{i=1}^n \left\| \nabla f_{s,\tau}^i \mathbb{I}_{\{s+d_s^i > t\}} \right\|^2} + \lambda^k(\mathbf{W}) \sum_{s \leq t} \sqrt{\sum_{i=1}^n \left\| \nabla f_{s,k}^i \mathbb{I}_{\{s+d_s^i > t\}} \right\|^2} + \sum_{s \leq t} \sqrt{\sum_{i=1}^n \left\| \nabla f_{s,1}^i \mathbb{I}_{\{s+d_s^i > t\}} \right\|^2} \\
& \leq 2G \sum_{\tau=1}^k \lambda^{k-\tau}(\mathbf{W}) \sum_{s \leq t} \sqrt{\sum_{i=1}^n \mathbb{I}_{\{s+d_s^i > t\}}} + G \left[ \lambda^k(\mathbf{W}) + 1 \right] \sum_{s \leq t} \sqrt{\sum_{i=1}^n \mathbb{I}_{\{s+d_s^i > t\}}} \\
& \leq 2G \left[ \frac{\lambda(\mathbf{W})}{1 - \lambda(\mathbf{W})} + 1 \right] \sum_{s \leq t} \sqrt{\sum_{i=1}^n \mathbb{I}_{\{s+d_s^i > t\}}} \\
& \leq 2\sqrt{n}G \left[ \frac{\lambda(\mathbf{W})}{1 - \lambda(\mathbf{W})} + 1 \right] \sum_{s \leq t} \frac{1}{n} \sum_{i=1}^n \sum_{i=1}^n \mathbb{I}_{\{s+d_s^i > t\}}
\end{aligned}$$

where the fifth inequality resulted from Lipschitzness of  $f_t$  and the value of indicator function.  $\square$

### B.3 Proofs of Theorem 2

**Theorem 2.** Given a constraint set  $\mathcal{K}$ . Let  $A = \max \left\{ 3, \frac{3G}{2\beta D}, \frac{2\beta C_d + C_g}{\beta D} \right\}$ ,  $\eta_k = \min \left\{ 1, \frac{A}{k} \right\}$ , and  $K = \sqrt{T}$ . Suppose that Assumptions 1 to 3 hold true. If we choose FTPL as the underlying oracle and set  $\zeta = \frac{1}{G\sqrt{B}}$ , the regret of Algorithm 2 is

$$\sum_{t=1}^T \left[ F_t(\mathbf{x}_t^i) - F_t(\mathbf{x}^*) \right] \leq (GC_d + 2\beta AD^2) \sqrt{T} + 3(A+1) \left[ 2\sqrt{n}DG \left( \frac{\lambda(\mathbf{W})}{\rho} + 1 \right) \sqrt{B} + \mathcal{R}_{T,\mathcal{O}} \right] \quad (36)$$

where  $B = \frac{1}{n} \sum_{i=1}^n B_i$  such that  $B_i$  is the sum of delay values of agent  $i$ .  $C_d = k_0 \sqrt{n}D$  and  $C_g = \sqrt{n} \max \left\{ \lambda(\mathbf{W}) \left( G + \frac{\beta D}{\rho} \right), k_0 \beta (4C_d + AD) \right\}$  are defined in Lemmas 3 and 6 and  $\mathcal{R}_{T,\mathcal{O}}$  is the regret of FTPL with the current choice of  $\zeta$

*Proof.* By smoothness of  $F_t$ , we have

$$\begin{aligned}
F_t(\bar{\mathbf{x}}_{t,k+1}) - F_t(\mathbf{x}^*) &= F_t\left(\bar{\mathbf{x}}_{t,k} + \eta_k \left(\frac{1}{n} \sum_{i=1}^n \mathbf{v}_{t,k}^i - \bar{\mathbf{x}}_{t,k}\right)\right) - F_t(\mathbf{x}^*) \\
&\leq F_t(\bar{\mathbf{x}}_{t,k}) - F_t(\mathbf{x}^*) + \eta_k \left\langle \nabla F_t(\bar{\mathbf{x}}_{t,k}), \frac{1}{n} \sum_{i=1}^n \mathbf{v}_{t,k}^i - \bar{\mathbf{x}}_{t,k} \right\rangle + \eta_k^2 \frac{\beta}{2} \left\| \frac{1}{n} \sum_{i=1}^n \mathbf{v}_{t,k}^i - \bar{\mathbf{x}}_{t,k} \right\|^2 \\
&\leq F_t(\bar{\mathbf{x}}_{t,k}) - F_t(\mathbf{x}^*) + \frac{\eta_k}{n} \langle \nabla F_t(\bar{\mathbf{x}}_{t,k}), \mathbf{v}_{t,k}^i - \bar{\mathbf{x}}_{t,k} \rangle + \eta_k^2 \frac{\beta D^2}{2} \\
&\leq F_t(\bar{\mathbf{x}}_{t,k}) - F_t(\mathbf{x}^*) + \frac{\eta_k}{n} \sum_{i=1}^n [\langle \nabla F_t(\bar{\mathbf{x}}_{t,k}), \mathbf{v}_{t,k}^i - \mathbf{x}^* \rangle + \langle \nabla F_t(\bar{\mathbf{x}}_{t,k}), \mathbf{x}^* - \bar{\mathbf{x}}_{t,k} \rangle] + \eta_k^2 \frac{\beta D^2}{2} \\
&\leq (1 - \eta_k) [F_t(\bar{\mathbf{x}}_{t,k}) - F_t(\mathbf{x}^*)] + \frac{\eta_k}{n} \sum_{i=1}^n [\langle \nabla F_t(\bar{\mathbf{x}}_{t,k}), \mathbf{v}_{t,k}^i - \mathbf{x}^* \rangle] + \eta_k^2 \frac{\beta D^2}{2} \\
&\leq (1 - \eta_k) [F_t(\bar{\mathbf{x}}_{t,k}) - F_t(\mathbf{x}^*)] + \frac{\eta_k}{n} \sum_{i=1}^n [\langle \nabla F_t(\bar{\mathbf{x}}_{t,k}), \mathbf{v}_{t,k}^i - \hat{\mathbf{v}}_{t,k}^i \rangle + \langle \nabla F_t(\bar{\mathbf{x}}_{t,k}), \hat{\mathbf{v}}_{t,k}^i - \mathbf{x}^* \rangle] + \eta_k^2 \frac{\beta D^2}{2} \\
&\leq (1 - \eta_k) [F_t(\bar{\mathbf{x}}_{t,k}) - F_t(\mathbf{x}^*)] + \frac{\eta_k}{n} \sum_{i=1}^n [\langle \nabla F_t(\bar{\mathbf{x}}_{t,k}), \mathbf{v}_{t,k}^i - \hat{\mathbf{v}}_{t,k}^i \rangle + \langle \hat{\mathbf{d}}_{t,k}^i, \hat{\mathbf{v}}_{t,k}^i - \mathbf{x}^* \rangle] + \eta_k \frac{2\beta C_d + C_g}{k} D + \eta_k^2 \frac{\beta D^2}{2}
\end{aligned} \tag{37}$$

where we used convexity of  $F_t$  in the fourth inequality and the last inequality follows by observing that

$$\begin{aligned}
\langle \nabla F_t(\bar{\mathbf{x}}_{t,k}), \hat{\mathbf{v}}_{t,k}^i - \mathbf{x}^* \rangle &= \langle \nabla F_t(\bar{\mathbf{x}}_{t,k}) - \nabla F_{t,k}, \hat{\mathbf{v}}_{t,k}^i - \mathbf{x}^* \rangle + \langle \nabla F_{t,k}, \hat{\mathbf{v}}_{t,k}^i - \mathbf{x}^* \rangle \\
&= \langle \nabla F_t(\bar{\mathbf{x}}_{t,k}) - \nabla F_{t,k}, \hat{\mathbf{v}}_{t,k}^i - \mathbf{x}^* \rangle + \langle \nabla F_{t,k} - \hat{\mathbf{d}}_{t,k}^i, \hat{\mathbf{v}}_{t,k}^i - \mathbf{x}^* \rangle + \langle \hat{\mathbf{d}}_{t,k}^i, \hat{\mathbf{v}}_{t,k}^i - \mathbf{x}^* \rangle \\
&\leq (\beta \|\bar{\mathbf{x}}_{t,k} - \mathbf{x}_{t,k}^i\| + \|\nabla F_{t,k} - \hat{\mathbf{d}}_{t,k}^i\|) D + \langle \hat{\mathbf{d}}_{t,k}^i, \hat{\mathbf{v}}_{t,k}^i - \mathbf{x}^* \rangle \\
&\leq \frac{2\beta C_d + C_g}{k} D + \langle \hat{\mathbf{d}}_{t,k}^i, \hat{\mathbf{v}}_{t,k}^i - \mathbf{x}^* \rangle
\end{aligned}$$

Let  $A = \max\left\{3, \frac{3G}{2\beta D}, \frac{2\beta C_d + C_g}{\beta D}\right\}$  and  $\eta_k = \frac{A}{k}$ . From Lemma 5 we have

$$F_t(\bar{\mathbf{x}}_{t,K+1}) - F_t(\mathbf{x}^*) \leq \frac{2\beta AD^2}{K} + \frac{1}{n} \sum_{i=1}^n \sum_{k=1}^K \eta_k \left[ \prod_{\ell=k+1}^K (1 - \eta_\ell) \right] \left[ \langle \hat{\mathbf{d}}_{t,k}^i, \hat{\mathbf{v}}_{t,k}^i - \mathbf{x}^* \rangle + \langle \nabla F_t(\bar{\mathbf{x}}_{t,k}), \mathbf{v}_{t,k}^i - \hat{\mathbf{v}}_{t,k}^i \rangle \right] \tag{38}$$

Summing Equation (38) over  $T$ -rounds yields,

$$\begin{aligned}
\sum_{t=1}^T [F_t(\bar{\mathbf{x}}_t) - F_t(\mathbf{x}^*)] &\leq \frac{2\beta AD^2 T}{K} + \frac{1}{n} \sum_{i=1}^n \sum_{k=1}^K \eta_k \prod_{\ell=k+1}^K (1 - \eta_\ell) \sum_{t=1}^T \left[ \langle \hat{\mathbf{d}}_{t,k}^i, \hat{\mathbf{v}}_{t,k}^i - \mathbf{x}^* \rangle + \langle \nabla F_t(\bar{\mathbf{x}}_{t,k}), \mathbf{v}_{t,k}^i - \hat{\mathbf{v}}_{t,k}^i \rangle \right] \\
&\leq \frac{2\beta AD^2 T}{K} + \frac{1}{n} \sum_{i=1}^n \sum_{k=1}^K \eta_k \prod_{\ell=k+1}^K (1 - \eta_\ell) \left[ \mathcal{R}_{T,\mathcal{O}} + 2\zeta \sqrt{n} D G^2 \left( \frac{\lambda(\mathbf{W})}{1 - \lambda(\mathbf{W})} + 1 \right) \frac{1}{n} \sum_{i=1}^n \sum_{t=1}^T \sum_{s \leq t} \mathbb{I}_{\{s+d_s^i > t\}} \right] \\
&\leq \frac{2\beta AD^2 T}{K} + 3(A+1) \left[ \mathcal{R}_{T,\mathcal{O}} + 2\zeta \sqrt{n} D G^2 \left( \frac{\lambda(\mathbf{W})}{1 - \lambda(\mathbf{W})} + 1 \right) \frac{1}{n} \sum_{i=1}^n \sum_{t=1}^T \sum_{s \leq t} \mathbb{I}_{\{s+d_s^i > t\}} \right] \tag{39}
\end{aligned}$$

From Equation (39), we deduce that, for all  $i \in [n]$ ,

$$\begin{aligned}
\sum_{t=1}^T [F_t(\mathbf{x}_t^i) - F_t(\mathbf{x}^*)] &\leq \sum_{t=1}^T [F_t(\mathbf{x}_t^i) - F_t(\bar{\mathbf{x}}_t)] + \sum_{t=1}^T [F_t(\bar{\mathbf{x}}_t) - F_t(\mathbf{x}^*)] \\
&\leq \sum_{t=1}^T G \|\mathbf{x}_t^i - \bar{\mathbf{x}}_t\| + \sum_{t=1}^T [F_t(\bar{\mathbf{x}}_t) - F_t(\mathbf{x}^*)] \\
&\leq \frac{GC_d T}{K} + \sum_{t=1}^T [F_t(\bar{\mathbf{x}}_t) - F_t(\mathbf{x}^*)] \\
&\leq \frac{GC_d + 2\beta AD^2}{K} T + 3(A+1) \left[ \mathcal{R}_{T,\mathcal{O}} + 2\zeta \sqrt{n} DG^2 \left( \frac{\lambda(\mathbf{W})}{1 - \lambda(\mathbf{W})} + 1 \right) \frac{1}{n} \sum_{i=1}^n \sum_{t=1}^T \sum_{s < t} \mathbb{I}_{\{s+d_s^i > t\}} \right] \\
&\leq \frac{GC_d + 2\beta AD^2}{K} T + 3(A+1) \left[ \mathcal{R}_{T,\mathcal{O}} + 2\zeta \sqrt{n} DG^2 \left( \frac{\lambda(\mathbf{W})}{1 - \lambda(\mathbf{W})} + 1 \right) \frac{1}{n} \sum_{i=1}^n B_i \right]
\end{aligned} \tag{40}$$

The theorem follows by letting  $\zeta = \frac{1}{G\sqrt{B}}$ ,  $K = \sqrt{T}$  and  $B = \frac{1}{n} \sum_{i=1}^n B_i$ . This concludes the proof.  $\square$