Università degli Studi di Padova

Dipartimento di Ingegneria dell'Informazione

Pushing the Boundaries of Federated Learning:

Super-Linear Convergence and Reinforcement Learning Over Wireless

Ph.D. candidate
Nicoló Dal Fabbro

Advisor
Prof. Luca Schenato

Co-Advisor
Prof. Michele Rossi

Director & Coordinator
Prof. Fabio Vandin

Ph.D. School in
Information Engineering

Department of
Information Engineering

University of Padova
2023

Acknowledgments

I would like to acknowledge my advisor, Prof. Luca Schenato, to whom I owe the most sincere gratitude. Thank you, Luca, for the time you devoted to me, for teaching me a lot, especially on going deeper in the fascinating methodological problems we have addressed together, with passion and commitment. Thank you for your patience, for having confidence in me, for having always provided constructive comments in every circumstance, with a positive and friendly attitude which I will hardly forget. For this and for the countless occasions in which I have treasured of your guidance, thank you.

I would also like to thank Prof. Michele Rossi, to which I owe my decision to start a PhD, as well as my first experience in research, and that has been a precious and helpful co-advisor in my PhD journey. Thank you, Michele, for being there when I needed help and advice, for your dedication, and for teaching me a lot.

I would like to thank with gratitude Prof. Subhrakanti Dey for working with me and advising me from the first year of the PhD onwards. I would also like to thank Prof. Giuseppe Piro for spending a considerable amount of time with me when, at the beginning of the PhD, we were investigating potential research topics related to the 5G technology. I have fond memories of those (sometimes very challenging) days.

When starting the PhD, I had the luck of having the support of wonderful people that I have to mention here. First, I would like to mention how much I appreciated the help and guidance of Francesca Meneghello, that tutored me during my Master thesis and with whom I kept working during the PhD. Thank you for your kindness, gentleness, and devotion to work. You have been very important for my first steps in the research world and it was great working with you. Another person I would like to mention with gratitude is Luca Ballotta. I started the PhD during the Covid-19 shutdowns and I did not have a chance to meet many people at the department. Luca has been there in those days, whenever I needed help. Thank you, Luca, for your kindness and availability. Last but not least, I would like to thank Nicola Lissandrini for his friendship and for being the man I shared more Spritz with in the Department.

I would like to thank the Department of Information Engineering of the University of Padova for being a great and welcoming learning environment along all my undergraduate and graduate studies. Within the department, I would like to thank all the PhD students, postdocs and professors I had a chance to interact with and sharing moments.

I spent a relevant part of my PhD abroad, during the third year. During this time, I had the privilege of being hosted as a visiting scholar at the University of Pennsylvania (Penn), where I met truly amazing people. Firstly, I would like to thank Prof. George Pappas for being my host and supervisor during my period at Penn. George's guidance

and advice has been precious for me. I would like to express the deepest gratitude to Prof. Aritra Mitra, who was a postdoc at Penn at the time of my visit, for involving me in great projects and for guiding me through a theoretical topic that was very new to me, with patience and kindness. Thank you, Aritra, for being not only a great research advisor, but also for welcoming me at Penn and in Philadelphia in the most warm and friendly way. Your passion and dedication to research has been of great inspiration to me. I would like to thank Prof. Lars Lindemann, who was a postdoc at Penn at the time, with whom I shared the office, for being very kind, friendly and helpful. I would also like to thank Claudio Battiloro, who was a visiting scholar at Penn like me, for being an amazing room mate, for his support and for being a true friend. I would like to extend all my gratitude also to all the other beautiful people I met at Penn and in Philadelphia.

Finally, I want to thank with all my heart all the people outside of the University who gave me love. I owe immense gratitude to all of you. Thank you to my parents, my brother, all the members of my family, all my friends, and those whom I loved and who loved me back. I am grateful for all the love I received from all of you. To all these people, with which I shared unforgettable experiences: thank you.

Abstract

In an age defined by explosive growth in information technology, data generation, storage and transmission have increased dramatically. This data fuels the core of machine learning and artificial intelligence. However, we are witnessing increasingly pressing questions raised about data ownership and privacy, given the pivotal role of individuals as data generators. In this context, research efforts in distributed machine learning, particularly Federated Learning (FL), have recently gained momentum. FL enables multiple agents, each with private datasets, to collaborate on machine learning tasks without sharing their data. In recent years, the design of *communication-efficient* FL methods has garnered significant attention, given the inherent need for frequent information exchange among agents to train distributed machine learning algorithms. Given this premise, in this thesis we explore the boundaries of FL, focusing on two aspects. First, we study second-order methods with superlinear convergence rate that can effectively deal with ill-conditioned problems while being communication efficient. Towards this direction, we introduce SHED (Sharing Hessian Eigenvectors for Distributed learning), a novel Newton-type algorithm for FL with state-of-the-art empirical performance that excels in terms of communication efficiency and convergence guarantees. Second, we study the theoretical foundations of Federated Reinforcement Learning (FRL) within the constraints of communication, with special emphasis on wireless networks. In these settings, we provide finite-time convergence rates for FRL, showing the beneficial effect of cooperation even under communication constraints, establishing convergence speedups with the number of agents in different configurations.

Contents

1 Introduction

Thanks to the major information technology advancements of the last fifty years, human civilization has been recently characterized by an extraordinary increase in the capacity to produce, store and telecommunicate information *data*, usually in the form of digital objects, i.e., streams of encoded bits. Together with the phenomena of massive data production and massive connectivity, algorithmic advancements and a dramatic boost in computational power of computing machineries have paved the way towards the development of data-driven algorithms, which nowadays represent the core of the so-called machine learning (ML) paradigm, which in turn is at the foundations of artificial intelligence (AI). The main AI applications, at the time of the writing of this thesis, are not only in prediction, classification and regression, but also in the emerging frameworks of generative algorithms, i.e., algorithms that can generate new data based on past data. In this context, given the high pace of development of AI algorithms, the crucial value of data is becoming apparent together with the relevance of the concepts of data ownership and data privacy. Indeed, in a modern civilization whose destiny seems to be unavoidably intertwined with the development of AI, which relies on data-driven algorithms, data itself becomes a "magic powder" without which the human civilization system does not have the capacity to function and progress. In recent years, major multinational technology companies, such as Google, Meta, Apple, Microsoft, and Amazon, have been aggressively expanding their businesses by leveraging AI, largely built on the vast reserves of human data they manage. However, it is not clear how individuals, private citizens and institutions, should reap benefits from the AI algorithms that rely on the data they generate daily. In this context, western civilization is finding itself in the presence of the following key aspects: (i) the growing awareness of the value and power of the increasing amount of data produced and collected by humans, (ii) the evidence of the benefit that the use of as much data as possible has in obtaining extremely powerful AI data-driven algorithms, and (iii) the growing awareness that the economic value of data, the identity

of the people that should be benefiting from this value, and consequently the regulations to use human data itself, should be seriously reconsidered.
One natural consequence of the above considerations is the increased interest, surged with great momentum in recent years, within both the scientific and industrial communities, for *distributed machine learning algorithms* in which, although participating in training AI models based on the data they produce, individual entities keep their data private.
In addition to the privacy-preserving arguments and the data ownership considerations provided above, there are several reasons why distributed ML solutions are of high interest to the information and communication technology research community. Among these, distributed computation and distributed algorithms are very appealing because they allow a system to distribute the computational effort across multiple entities. In the ML context, this implies the possibility of training AI models on datasets stored in multiple machines, rather than in a single one. Training a global ML model without the need to transfer all the data to a single machine, which, for large datasets, could be undoable, was the original motivation provided by Google researchers when they first proposed the leading distributed ML paradigm of the last years, i.e., Federated Learning (FL) [92]. In addition to the practical storage convenience of ML distributed training, in a general multi-agent setting in which data are stored across multiple machines in multiple locations of a communication network, transferring the data of the individual entities to a central server for ML training could be very expensive from a communication point of view. This latter aspect is particularly crucial when data are generated continuously by devices at the network edge, from which reaching central servers in the core network is well known to be very expensive, from both an energy consumption point of view [116], and a communication resources point of view, given the pace of growth of the network traffic in the present and upcoming years [85]. Distributed computation is indeed also a key design principle for Internet-of-Things and the Multi-access Edge Computing (MEC) paradigms [117], where connected entities collect and produce data, execute algorithms and communicate at the network edge (see Figure 1.1 for an illustration of this scheme[1]). Based on these considerations, we see how distributed computing architectures are not only a requirement for data privacy considerations in the deployment of AI, they are also a key requirement in the modern Internet infrastructure.

In recent years, as mentioned above, the leading paradigm for distributed machine learning has emerged under the name of Federated Learning (FL). In FL, multiple agents, each owning a private dataset, cooperate to solve a common ML problem without sharing their data with each others. FL involves cooperation through the exchange of

[1]Figure source: https://www.wipro.com/infrastructure/edge-computing-understanding-the-user-experience/

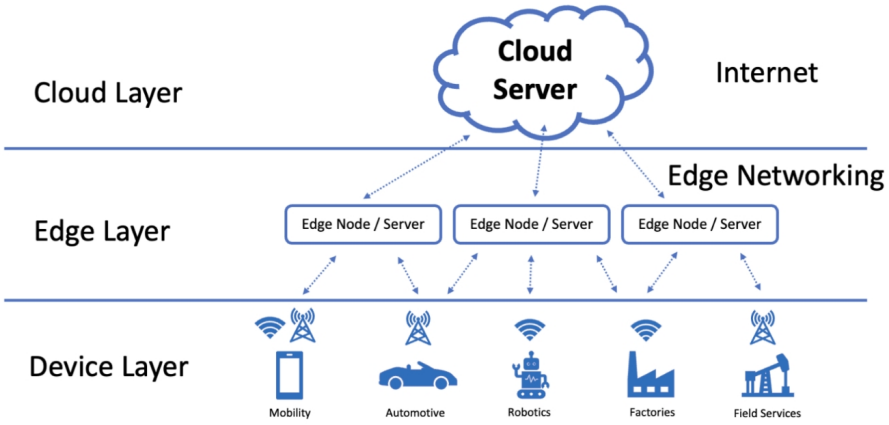

Figure 1.1: Edge Computing architecture.

optimization parameters, such as agents' gradients or ML models, with a central entity known as the Master, Server, or Aggregator. The collective of agents participating in this framework constitutes a "federation" of entities. These entities seek to harness the benefits of cooperation while maintaining the privacy of their data. In essence, solving a machine learning problem often translates to solving a mathematical optimization problem. Likewise, mostly for convergence analysis purposes, an FL problem is often analysed as a particular instance of a distributed optimization (DO) challenge. The realm of DO has garnered significant attention in recent decades, with an extensive body of related literature. From an engineering perspective, one of the primary hurdles in DO and FL lies in achieving communication efficiency. DO and FL algorithms, indeed, necessitate frequent information exchange among agents, demanding coordination and communication. Over the past few years, there has been a growing focus on studying the FL framework under communication constraints, which is motivated also by the fact that in many real-world scenarios, like MEC 5G networks, the bottleneck in time and cost is represented by communicating rather than by performing local distributed computation.

In this thesis, we study some areas in FL that, although very promising, were not very much explored yet during the PhD time span, at the "boundaries" of FL. Specifically, we focus on two aspects: (i) communication-efficient second-order methods in FL, in which local devices computational effort is pushed to the extreme to improve the communication efficiency. In this context, we propose SHED (Sharing Hessian Eigenvectors for Distributed learning) an original Newton-type algorithm for FL with state-of-the-art performance and appealing features like, notably, global convergence guarantees with asymptotic super-linear rate. As a second aspect, (ii) we provide novel theoretical foundation results for the emerging paradigm of federated reinforcement learning (FRL) under communication

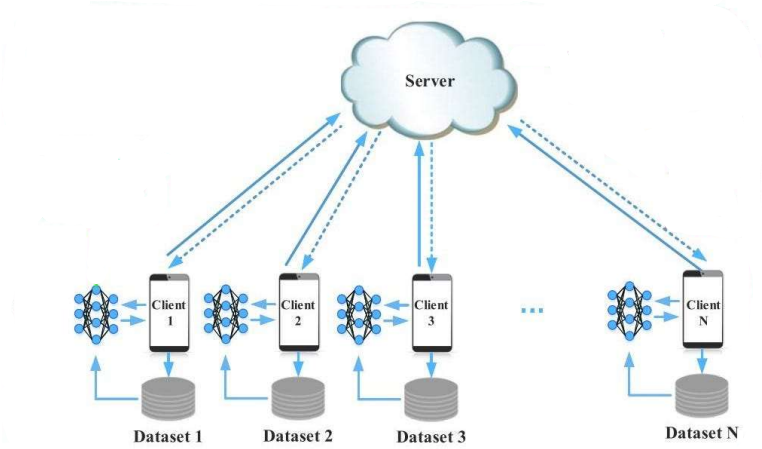

Figure 1.2: Federated Learning framework. N agents communicate with a central entity to cooperatively train an ML model without sharing their private datasets.

constraints, considering different communication models including wireless analog over-the-air computation and settings in which the the Stochastic Approximation operator is computed with delayed parameters and observations.

In the following sections, we will provide a broad overview of FL together with a description of the topics that we investigate in our thesis. We then illustrate our specific contributions and the organization of this manuscript, introducing the key novelties of our findings, and the advancements that we provide with respect to the state-of-the-art.

1.1 Federated Learning

Federated Learning (FL) is a paradigm for distributed machine learning that has been first introduced in [92], [93]. The framework consists of a group of agents aiming to cooperate in training a machine learning (ML) algorithm by communicating with a central entity. Each FL agent holds a private dataset, and is willing to cooperatively training an ML algorithm but without sharing their private data with the other network entities. See Figure 1.2 for an illustration of this scheme. In this thesis, we often refer to the central entity as the Master. Together with the growing concerns for data privacy and data ownership that were mentioned in the first part of the introduction, one of the reasons why the FL framework has gathered a lot of attention lately is the large amount of real-world settings in which multiple agents communicate with some central entity. A relevant example is the client-server type of architecture which pervasively characterizes the modern internet infrastructure and protocols. A further example are wireless cellular

systems in which users are usually connected to a central entity (e.g., a base station), which in turn is usually connected to some server in the core network. From the point of view of distributed optimization (DO), FL represents a special case, being a network configuration with a star topology, where the center of the star is the Master. Compared to peer-to-peer types of frameworks, in which the main building block for DO is usually computing an average, reaching the *consensus*, in most FL instances consensus is reached in one step, thanks to the star topology type of configuration. In mathematical form, the DO and FL problem in a setting with N agents can be written as a minimization of the sum of N cost functions, as follows

$$\text{minimize } f(\boldsymbol{\theta}) = \frac{1}{N}\sum_{i=1}^{N} f_i(\boldsymbol{\theta}),\ \boldsymbol{\theta} \in \mathbb{R}^n, \tag{1.1}$$

where $f_i(\boldsymbol{\theta})$ is the (expected) cost function of agent i, which in turn depends on the local dataset of agent i. The most common methods to train FL models are first-order DO algorithms (e.g., gradient descent). Among the research endeavours on first-order methods for FL, considerable effort has been devoted to reducing the communication load, i.e., in developing *communication efficient* solutions. Indeed, first-order DO requires frequent information exchange between agents and the master. To reach communication efficiency, many techniques have been proposed, including communication *compression* and related approaches [6], [76], [80], [98], local computation [74], [100], [136] and partial participation [58], [60].

1.1.1 Second-Order Methods

Although very effective, first-order methods are usually very sensitive to the problem structure, which is usually captured by the notion of *condition number*. As it is very well known, first-order methods are all extremely sensitive to the condition number of the cost function. In mathematical terms, an example of this dependency can be observed simply inspecting the convergence rate of (centralized) gradient descent (GD). Let us write the GD iterative update rule as follows:

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - \alpha \mathbf{g}(\boldsymbol{\theta}_k), \tag{1.2}$$

where $\boldsymbol{\theta}_k$ is the parameter at iteration k, $\alpha > 0$ is a step size/learning rate and $\mathbf{g}(\boldsymbol{\theta}) = \nabla f(\boldsymbol{\theta})$ is the gradient of the learning surrogate cost function f computed at $\boldsymbol{\theta}$. In the deterministic least squares case, the optimal rate of convergence achievable with a specific

choice of step size is the following:

$$\|\boldsymbol{\theta}_T - \boldsymbol{\theta}^*\| \leq \left(1 - \frac{1}{\kappa}\right)^T \|\boldsymbol{\theta}_0 - \boldsymbol{\theta}^*\|, \tag{1.3}$$

where, denoting by λ_i the i-th eigenvalue (in decreasing order) of the Hessian matrix, $\kappa = \frac{\lambda_1}{\lambda_n}$ is the condition number of the problem. It is well known that all first-order methods have a similar form of dependency on the condition number. On the other hand, it is well-known that Newton-type methods, which use the curvature information provided by the Hessian matrix, can achieve a super-linear convergence rate independent of the condition number [15]. The Newton method update rule has the following form

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - \alpha \left[\mathbf{H}(\boldsymbol{\theta}_k)\right]^{-1} \mathbf{g}(\boldsymbol{\theta}_k) \tag{1.4}$$

where $\mathbf{H}(\boldsymbol{\theta}) = \nabla^2 f(\boldsymbol{\theta})$ is the Hessian matrix of the cost function f computed in $\boldsymbol{\theta}$. To illustrate the aforementioned convergence properties, let us consider a μ-strongly convex cost function with L-Lipschitz continuous Hessian. The Newton method attains a *local* convergence rate of the following form [24]:

$$\|\boldsymbol{\theta}_k - \boldsymbol{\theta}^*\|^2 \leq \frac{4\mu^2}{L^2} \left(\frac{1}{2}\right)^{2^k}. \tag{1.5}$$

Note that (i) the convergence rate is equal to $\frac{1}{2}$, *independent of the condition number* and (ii) the rate is *superlinear*, specifically quadratic.

The extremely appealing convergence properties of Newton-type methods have motivated the research community to use the curvature information, by means of the Hessian matrix, in distributed optimization and federated learning to reduce the number of iterations required to converge, and therefore to reduce the communication load of distributed training. The use of agents second-order information implies, however, an increased computational burden at the agents, as computing the Hessian matrix is computationally intense. In other words, second-order methods have been advocated to attain *communication efficiency* at the price of increasing the *computational effort*. Indeed, in FL, agents are often assumed to have good computing capabilities, like, e.g., in the case of smartphones and laptops in edge networks [85]. Therefore, wisely increasing the computational effort at the agents is an appealing strategy to speedup the convergence. For this reason, Newton-type approaches, characterized by robustness and fast convergence rates, even if computationally demanding, have been recently advocated [63], [125], [139]. In addition to the increased computational burden, it is very impractical to directly apply the Newton method in the FL setting (and in general in DO). Indeed,

given N agents, each computing their own local Hessian matrix, $\mathbf{H}_i(\boldsymbol{\theta}) = \nabla^2 f_i(\boldsymbol{\theta})$ and gradient $\mathbf{g}_i(\boldsymbol{\theta}) = \nabla f_i(\boldsymbol{\theta})$ on their local cost functions $f_i(\boldsymbol{\theta})$, applying the Newton method directly would require performing the following update rule (assuming for simplicity that all agents have the same number of data samples):

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - \alpha \left(\frac{1}{N} \sum_{i=1}^{N} \mathbf{H}_i(\boldsymbol{\theta}_k) \right)^{-1} \left(\frac{1}{N} \sum_{i=1}^{N} \mathbf{g}_i(\boldsymbol{\theta}_k) \right). \tag{1.6}$$

Note that while computing the global cost function gradient $\mathbf{g}(\boldsymbol{\theta}_k) = \frac{1}{N} \sum_{i=1}^{N} \mathbf{g}_i(\boldsymbol{\theta}_k)$ requires each agent to transmit their gradient vector of size $O(n)$ to the master, which is the standard for first-order methods in FL and DO, the computation of the global Hessian $\mathbf{H}(\boldsymbol{\theta}_k) = \frac{1}{N} \sum_{i=1}^{N} \mathbf{H}_i(\boldsymbol{\theta}_k)$ would require each agent to transmit their local Hessian matrix of size $O(n^2)$. The transmission of $O(n^2)$ information at each iteration is prohibitive when the size n of the parameter $\boldsymbol{\theta} \in \mathbb{R}^n$ increases, and this additional communication cost would make pointless the use of second-order information, whose purpose would be to improve the communication efficiency. Therefore, recent research efforts [39], [125], [128], [144] have proposed techniques for DO and FL in which approximations of the Hessian matrices are used in place of the actual Hessian, in order to obtain a communication complexity of $O(n)$ while enjoying faster convergence thanks to the use of second-order information. The update rule of these types of approximate Newton methods, also referred to as Newton-type methods, has usually the following form:

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - \alpha \hat{\mathbf{H}}_k^{-1} \mathbf{g}(\boldsymbol{\theta}_k), \tag{1.7}$$

where $\hat{\mathbf{H}}_k$ is an approximation of the Hessian matrix used at iteration k. In chapter 2 of the thesis, we will illustrate SHED (Sharing Hessian Eigenvectors for Distributed learning), a novel Newton-type method based on eigendecomposition of local agents' Hessian matrices, whose update rule also has the form shown in (1.7).

1.1.2 Federated Reinforcement Learning

In the last years, the FL framework described above has attracted considerable research interest. Major research efforts have been devoted to communication efficiency, i.e., to reap the benefit of cooperation while communicating as few bits as possible. Recently, the need for distributed solutions who preserve data privacy while boosting the training efficiency and effectiveness of ML algorithms has motivated also the study of distributed reinforcement learning (RL) algorithms. Towards this direction, the paradigm of federated reinforcement learning (FRL) has recently emerged [44], [77], [113]. While

empirical studies have already shown the benefits of cooperation in RL training in specific applications [106], there is a lack of theoretical understanding as to whether convergence speedups similar to the ones we obtain in FL also hold for FRL. Furthermore, while the effect of communication compression schemes and, more generally, of communication constraints has been widely studied in FL, little to nothing is known about their effect in FRL. In this regard, an improved theoretical understanding of FRL could pave the way to novel algorithms and effective solutions to boost the performance of FRL in a communication efficient way, which is of major practical relevance. Indeed, RL algorithms are notoriously very data hungry and RL training requires a critical amount of time. A large class of RL algorithms are just instances of Stochastic Approximation. As such, many RL algorithms can be written as iterative update rules as follows:

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k + \alpha \mathbf{g}(\boldsymbol{\theta}_k, o_k), \tag{1.8}$$

where o_k is a "data sample", usually an observation process temporally correlated and modelled as a Markov chain. While it is evident the similarity of this update rule with the typical stochastic gradient descent update rule, the main difference in RL algorithms such as temporal difference (TD) and Q-learning is that (i) the data samples o_k are temporally correlated and (ii) there is not a well-defined cost function associated with the learning problem. Hence, the study of finite-time convergence properties of these algorithms is much more challenging compared to optimization algorithms like stochastic gradient descent. In the context of multi-agent and FRL, one very relevant research question is related to the possibility to obtain a convergence speedup when the RL training is performed in parallel by multiple agents. This question can be formulated mathematically in the following way. Consider a setting in which an agent would achieve $1/T$ convergence precision in T iterations. In a setting in which N agents are training an RL algorithm in parallel, each acting in a replica of the same environment and with the same set of states and rewards, can cooperation - obtained via communication, in FRL with a central aggregator - allow the agents to obtain an overall $1/NT$ convergence precision in T iterations? In addition, would this speedup still hold in the presence of communication constraints such as communication compression, lossy links, noisy wireless channels and asynchronous delayed transmissions? In this thesis, we investigate these research questions, providing novel theoretical results which we believe contribute to build the foundations for an improved understanding and algorithmic advancements of multi-agent cooperative RL.

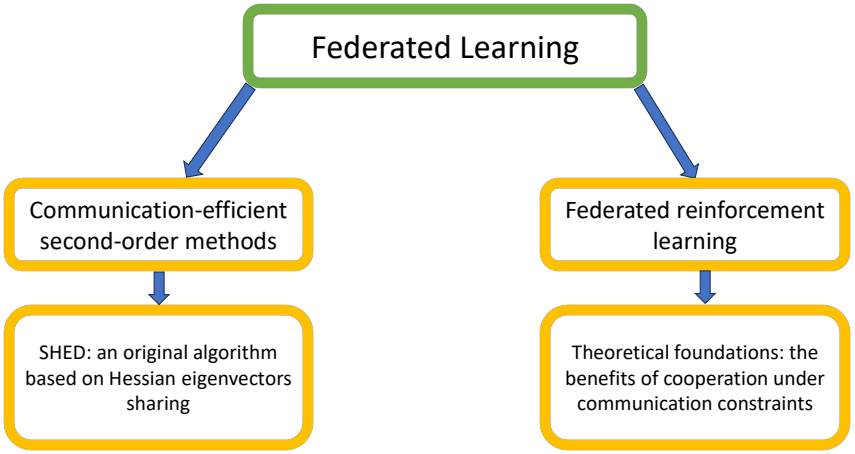

Figure 1.3: Summary of thesis contributions.

1.2 Contributions and Thesis Organization

In this section, we illustrate the main contributions of this thesis. First, we present the two main research questions we have investigated:

- Is it possible to design superlinearly convergent algorithms in FL that effectively combat ill-conditioning while having an $O(n)$ communication complexity per iteration, i.e., the same complexity of first-order methods?

- Is it possible to show that in FRL the cooperation of N agents provides an N-fold linear convergence speedup despite the correlated nature of agents' observation processes and under communication constraints?

In this regard, the contributions of the thesis are mainly as follows: (i) the design and analysis of a novel Newton-type algorithm for FL with state-of-the-art performance, which we briefly summarize in Section 1.2.1; (ii) the theoretical analysis of the finite-time convergence behaviour of FRL under communication constraints, which we summarize next in Section 1.2.2. See Figure 1.3 for a summary illustration of thesis contributions.

1.2.1 SHED: A Novel Newton-Type Algorithm for Federated Learning Based on Hessian Eigendecomposition

In chapter 2, we introduce SHED (Sharing Hessian Eigenvectors for Distributed learning) a novel Newton-type algorithm for FL. The algorithm is based on building approximations of the global Hessian using local Hessian eigenvalue-eigenvector pairs. SHED has several appealing features and state-of-the-art performance. In the chapter, we describe the algorithm in detail. First, we provide the main intuition in the case of linear regression with quadratic cost (least squares). Then, we present the algorithm in the general case of a convex cost function. We provide a rigorous convergence analysis that shows (i) the superlinear convergence of SHED and (ii) how SHED is effective in combatting ill-conditioning thanks to the distributed Hessian approximation technique. We then illustrate an extensive empirical analysis of SHED's performance comparing it against several state-of-the-art approaches. In chapter 3, we present Q-SHED, an extension of SHED based on a quantization scheme in which each agent allocates a bit budget to their local Hessian eigenvectors. The bit allocation strategy is cast as an optimization problem to minimize the difference between the resulting quantized Hessian approximation and the actual Hessian. The empirical results show that Q-SHED can reduce to up to 60% the number of communication rounds required for convergence.

1.2.2 Finite-Time Analysis of Federated Reinforcement Learning under Communication Constraints

In chapter 4, we present a finite-time convergence analysis of FRL frameworks under different communication constrained settings. Specifically, we consider a policy evaluation problem in which N agents cooperate to evaluate a common policy via temporal difference (TD) learning, by communicating with a central aggregator. See Fig. 1.4 for a pictorial representation of the considered federated TD learning framework. We present novel theoretical results for the following three communication settings: (i) a setting in which uplink communications are subject to random dropouts and transmitted gradients are quantized; (ii) a setting in which uplink transmission exploits over-the-air computation, inducing an iterative update rule in which local updates are subject to distortion and additive measurement noise at the receiver; (iii) an asynchronous setting in which local TD update directions are subject to bounded asynchronous delays. For all of these settings, we are able to provide a finite-time convergence analysis and to establish an N-fold linear convergence speedup, i.e., we are able to show the beneficial effect of cooperation even under the considered communication constrained settings. Notably, our results are the first available theoretical groundings providing finite-time convergence

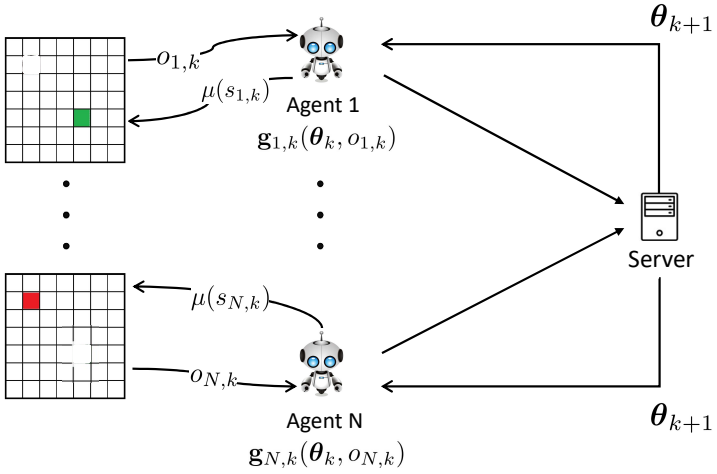

Figure 1.4: Federated TD Learning framework. N agents communicate with a central entity to cooperatively evaluate a common policy μ. Agents compute local TD update directions $\mathbf{g}_{i,k}(\boldsymbol{\theta}_k, o_{i,k})$ based on their observations $o_{i,k}$, collected interacting with their environment.

results for FRL under communication constraints.

In chapter 5, we focus with greater attention on a Stochastic Approximation (SA) setting with delayed updates, in the single agent case. Note that the SA setting under Markovian sampling is a generalization of the TD learning framework considered in chapter 4, and includes also Stochastic gradient descent under Markovian sampling and more complex RL algorithms like Q-learning. Within the considered setting, we are able to provide a finite-time convergence result that enjoys optimal dependencies on both the mixing time of the Markov chain (consistently with the non-delayed case) and on the delay sequence (consistently with the i.i.d. sampling case). Notably, our work is the first to provide finite-time convergence guarantees for SA under Markovian sampling and delayed updates.

2

SHED: A novel Newton-type algorithm for federated learning

In Federated Learning (FL), much attention is being turned to scenarios where the communication network is strongly heterogeneous in terms of communication resources (e.g., bandwidth) and data distribution. In these cases, communication between local machines (*agents*) and the central server (*Master*) is a main consideration. In this chapter, we present SHED, an original communication-constrained Newton-type (NT) algorithm designed to accelerate FL in such heterogeneous scenarios. SHED is by design robust to non independent identically distributed (non i.i.d.) data distributions, handles heterogeneity of agents' communication resources (CRs), only requires sporadic Hessian computations, effectively combats ill-conditioning, and achieves global asymptotic super-linear convergence. This is possible thanks to an *incremental* strategy, based on eigendecomposition of the local Hessian matrices, which exploits (possibly) *outdated* second-order information. The proposed solution is thoroughly validated on real datasets by assessing (i) the number of communication rounds required to achieve ϵ-convergence, (ii) the overall amount of data transmitted and (iii) the number of local Hessian computations. For all these metrics, the proposed approach shows superior performance against state-of-the art techniques like FedNL, GIANT and BFGS.

2.1 Introduction

With the growing computational power of edge devices and the booming increase of data produced and collected by users worldwide, solving machine learning problems without having to collect data at a central server is becoming very appealing [131]. One of the main reasons for not transferring users' data to cloud central servers is due to privacy concerns. Indeed, users, such as individuals or companies, may not want to share their

private data with other network entities, while training their machine learning algorithms. In addition to privacy, distributed processes are by nature more resilient to node/link failures and can be directly implemented on the servers at network edge, i.e., within multi-access edge computing (MEC) scenarios [117].

In recent years, the leading distributed machine learning framework is federated learning (FL) ([85], [94]), which has attracted much research interest in recent years. Direct applications of FL can be found, for example, in the field of healthcare systems [69], [123] or of smartphone utilities [68].

Among the open challenges of FL, a key research question is how to provide efficient distributed optimization algorithms in scenarios with constrained heterogeneous communication links (different bandwidth) and non i.i.d. data distributions [71], [133], [152], [155]. These aspects are found in a variety of applications, such as the so-called *federated edge learning* framework, where learning is moved to the network edge, and which often involves unstable and heterogeneous wireless connections [13], [28], [110], [131], [139]. In addition, the MEC and FL paradigms are of high interest to IoT scenarios such as data retrieval and processing within smart cities, which naturally entail non i.i.d. data distributions due to inherent statistical differences in the underlying spatial processes (e.g., vehicular mobility, user density, etc.) [89]. Within this context, recent works have also considered variants of the original FL framework, such as hierarchical and serverless settings [150], [107], [61].

The bottleneck represented by communication overhead is one of the most critical aspects of FL. In fact, in scenarios with massive number of devices involved, inter-agent communication can be much slower than the local computations performed by the FL agents themselves (e.g., the edge devices), by many orders of magnitude [85]. The problem of reducing the communication overhead of FL becomes even more critical when the system is characterized by non i.i.d. data distributions and heterogeneous communication resources (CRs) [86]. In this setting, where the most critical aspect is inter-node communications, FL agents are often assumed to have good computing capabilities, like, e.g., in the case of smartphones and laptops in edge networks [85]. Therefore, wisely increasing the computational effort at the agents is an appealing strategy to speedup the convergence. For this reason, Newton-type approaches, characterized by robustness and fast convergence rates, even if computationally demanding, have been recently advocated [63], [125], [139].

2.1.1 Main contributions

In this chapter, we present SHED (Sharing Hessian Eigenvectors for Distributed learning), a novel Newton-type algorithm for FL. The main features of SHED and our contributions can be summarized as follows:

- SHED uses an incremental strategy exploiting (possibly) outdated second-order information of the cost function. In particular, local machines (agents) provide the central server (the Master) with Hessian approximations by means of transmitting eigenvalue-eigenvector pairs (EEPs) of their local Hessian matrices together with a carefully computed approximation parameter. We take inspiration from [53] to build Hessian approximations from EEPs.

- SHED is by design robust in dealing with non i.i.d. data distributions and in effectively handling ill-conditioned problems. Furthermore, SHED handles agents' heterogeneous communication resources (CRs) by allowing those with more per-iteration CRs to share more EEPs per communication round. We analytically and empirically show how this improves the convergence rate. Furthermore, in sharp contrast with prior art, SHED, by design, requires agents to locally compute the Hessian matrix only sporadically.

- In the chapter, we first analyse the convergence rate of SHED for least squares problems, and then extend the analysis to the case of general FL problems with convex cost. In the convex case, we show that SHED enjoys global asymptotic super-linear convergence, and we analyse the super-linear convergence rate by studying the Lyapunov exponent of the estimation error dynamical system.

- Our results show that SHED is (i) competitive with state-of-the-art approaches in scenarios with i.i.d. data distributions and (ii) robust to non i.i.d. data distributions and ill-conditioned problems, for which it outperforms competing solutions.

2.1.2 Related work

Next, to put our contribution into context, we review related works on first and second order methods for FL.

First-order methods. To improve the robustness and the convergence properties of FedAvg [94], first-order methods like SCAFFOLD [72], FedProx [86], FedLin [101] and the work in [148] deal with system's heterogeneity, non i.i.d. datasets and communication constraints, like finite-rate channels. Along these lines, the work in [31] leverages the use of outdated first-order information by designing rules to detect slowly varying local

gradients. With respect to the problem of heterogeneous time-varying CRs, [9] presented techniques based on gradient quantization and on analog communication exploiting the additive nature of the wireless channel, while [28] studied a framework to jointly optimize learning and communication for FL in wireless networks.

Second-order methods. Newton-type (NT) methods exploit second-order information of the cost function to provide accelerated optimization, and are therefore appealing candidates to speed up FL. NT methods have been widely investigated for distributed learning purposes: GIANT [144] is an NT approach exploiting the harmonic mean of local Hessian matrices in distributed settings. Other related techniques are LocalNewton [63], DANE [128], AIDE [120], DiSCO [151], DINGO [39] and DANLA [150]. DONE [139] is another technique inspired by GIANT and specifically designed to tackle federated edge learning scenarios. Communication efficient NT methods like GIANT and DONE exploit an extra communication round to obtain estimates of the global Hessian from the harmonic mean of local Hessian matrices. These algorithms, however, were all designed assuming that data is i.i.d. distributed across agents and, as we empirically show in this work, under-perform if such assumption does not hold. A recent study, FedNL [125], proposed algorithms based on matrix compression which use theory developed in [70] to perform distributed training, by iteratively learning the Hessian matrix at the optimum. However, FedNL requires the computation of the local Hessians at each iteration, and does not consider heterogeneity in the CRs. Furthermore, FedNL only offers *local* super-linear convergence guarantees. Our work, instead, provides *global* (asymptotic) super-linear convergence guarantees. Other related NT approaches have been proposed in FLECS [2], FedNew [52] and Quantized Newton [5]. Quasi-Newton methods (i.e., second-order methods that do not explicitly compute the Hessian matrix) have also been recently investigated for FL [50], [90], [134], usually proposing variants of the popular BFGS [87] algorithm. These methods, however, provide usually only *local* convergence analysis or require the knowledge of problem specific constants, and their rate is heavily impacted by the condition number. Although some preliminary NT approaches have been proposed for the FL framework, there is still large space for improvements especially in ill-conditioned setups with heterogeneous CRs and non i.i.d. data distributions.

2.1.3 Organization of the Chapter

The rest of the chapter is organized as follows: in Section 2.2, we detail the problem formulation and the general idea behind the design of SHED. In Section 2.3, we present SHED, and the corresponding theoretical results, in the least squares (LS) case. Starting with LS enables us to provide the core intuition on the working principle of SHED in the

simpler case of constant Hessians. Section 2.4 is instrumental to extend the algorithm to a general strongly convex problem, which is done later on in Section 2.5, and which represents the main contribution of this chapter. At the end of Section 2.5, based on the theoretical results, we propose some heuristic choices for tuning SHED parameters. Finally, in Section 2.6, empirical performance of SHED is shown on real datasets. The proofs of the theoretical results are reported in Appendix A.

Notation: Vectors and matrices are written as lower and upper case bold letters, respectively (e.g., vector $\mathbf{v}$ and matrix $\mathbf{V}$). The operator $\|\cdot\|$ denotes the 2-norm for vectors and the spectral norm for matrices, $diag(\mathbf{v})$ denotes a diagonal matrix with the components of vector $\mathbf{v}$ as entries. We denote by $\mathbf{I}$ the identity matrix.

2.2 Problem Formulation

In Fig. 2.1 we illustrate the typical framework of an FL scenario. In the rest of the chapter, we denote the n-dimensional optimization parameter by $\boldsymbol{\theta} \in \mathbb{R}^n$, and a generic cost function by $f : \mathbb{R}^n \to \mathbb{R}$. We denote a dataset by $\mathcal{D} = \{\mathbf{x}_j, y_j\}_{j=1}^N$, where N is the global number of m-dimensional data samples $\mathbf{x}_j \in \mathbb{R}^m$ and response $y_j \in \mathbb{R}$. In the case of classification problems, y_j would be an integer specifying the class to which sample $\mathbf{x}_j$ belongs. We consider the problem of regularized empirical risk minimization of the form

$$\min_{\boldsymbol{\theta}} f(\boldsymbol{\theta}) := \frac{1}{N}\sum_{j=1}^{N} l_j(\boldsymbol{\theta}) + \frac{\mu}{2}\|\boldsymbol{\theta}\|_2^2, \tag{2.1}$$

where each $l_j(\boldsymbol{\theta})$ is a convex function related to the j-th element of $\mathcal{D}$ and μ is the regularization parameter. Examples of convex cost functions considered in this work, are linear regression with quadratic cost (least squares): $l_j(\boldsymbol{\theta}) = \frac{1}{2}(\mathbf{x}_j^T\boldsymbol{\theta} - y_j)^2$, and logistic regression: $l_j(\boldsymbol{\theta}) = \log(1+e^{-y_j(\mathbf{x}_j^T\boldsymbol{\theta})})$, in which $n = m$. In this chapter, we denote the data matrix by $\mathbf{X} = [\mathbf{x}_1, ..., \mathbf{x}_N] \in \mathbb{R}^{m\times N}$, and the label vector by $\mathbf{Y} = [y_1, ..., y_N] \in \mathbb{R}^{1\times N}$. We refer to the dataset $\mathcal{D}$ as the tuple $\mathcal{D} = (\mathbf{X}, \mathbf{Y})$. We denote by M the number of agents involved in the optimization algorithm, and we can write $\mathbf{X} = [\mathbf{X}_1, ..., \mathbf{X}_M]$, $\mathbf{Y} = [\mathbf{Y}_1, ..., \mathbf{Y}_M]$, where $\mathbf{X}_i \in \mathbb{R}^{m\times N_i}$ and $\mathbf{Y}_i \in \mathbb{R}^{1\times N_i}$, where N_i is the number of data samples of the i-th agent. We denote the local dataset of agent i by $\mathcal{D}_i = (\mathbf{X}_i, \mathbf{Y}_i)$. In this chapter we consider optimization problems in which the following assumptions on the agents' cost functions $f^{(i)}(\boldsymbol{\theta}) = \frac{1}{N_i}\sum_{(\mathbf{x}_j,y_j)\in\mathcal{D}_i} l_j(\boldsymbol{\theta}) + \frac{\mu}{2}\|\boldsymbol{\theta}\|_2^2$ hold:

Assumption 1. Let $\mathbf{H}^{(i)}(\boldsymbol{\theta}) := \nabla^2 f^{(i)}(\boldsymbol{\theta})$ be the local Hessian matrix of agent i. $f^{(i)}(\boldsymbol{\theta})$ is twice continuously differentiable, K_i-smooth, κ_i-strongly convex and $\mathbf{H}^{(i)}(\boldsymbol{\theta})$ is L_i-Lipschitz continuous for all $i = 1, ..., M$.

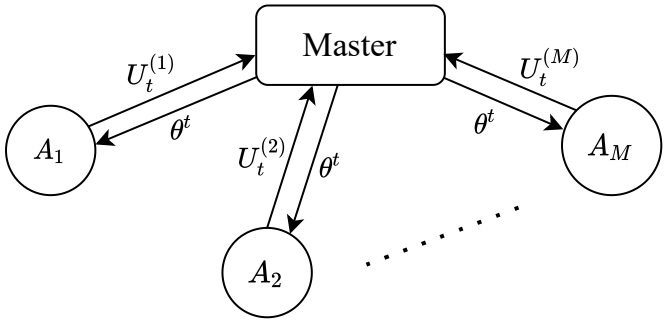

Figure 2.1: Considered FL framework: agents $A_1, ..., A_M$ cooperate to solve a common learning problem. After receiving the global parameter $\boldsymbol{\theta}^t$, at the current iteration t, they share their optimization sets $U_t^{(1)}, ..., U_t^{(M)}$ with the Master.

The above assumptions imply that

$$\kappa_i \mathbf{I} \leq \mathbf{H}^{(i)}(\boldsymbol{\theta}) \leq K_i \mathbf{I}, \ \forall \boldsymbol{\theta}$$

$$\|\mathbf{H}^{(i)}(\boldsymbol{\theta}) - \mathbf{H}^{(i)}(\boldsymbol{\theta}')\| \leq L_i \|\boldsymbol{\theta} - \boldsymbol{\theta}'\|, \ \forall \boldsymbol{\theta}, \boldsymbol{\theta}'$$

Note that K_i-smoothness is also a consequence of the K-smoothness of any global cost function $f(\boldsymbol{\theta})$ for some constant K, while strong convexity of agents' cost functions is always guaranteed, if the functions $l_j(\boldsymbol{\theta})$ are convex, in the presence of a regularization term $\mu > 0$. The results of the next sections all rely either on the existence of a positive regularization term $\mu > 0$, or on the fact that at least one of the cost functions $f^{(i)}(\boldsymbol{\theta}^t)$ is k_i-strongly convex with $k_i > 0$.

2.2.1 An eigendecomposition-based Newton-type method

The Newton method to solve (2.1) works as follows:

$$\boldsymbol{\theta}^{t+1} = \boldsymbol{\theta}^t - \eta_t \mathbf{H}_t^{-1} \mathbf{g}_t,$$

where t denotes the t-th iteration, $\mathbf{g}_t = \mathbf{g}(\boldsymbol{\theta}_t) = \nabla f(\boldsymbol{\theta}^t)$ is the gradient at iteration t and η_t is the step size at iteration t. $\mathbf{H}_t = \nabla^2 f(\boldsymbol{\theta}^t)$ denotes the Hessian matrix at iteration t. Compared to gradient descent, the Newton method exploits the curvature information provided by the Hessian matrix to improve the descent direction. We define $\mathbf{p}_t := \mathbf{H}_t^{-1} \mathbf{g}_t$. In general, Newton-type (NT) methods try to get an approximation of $\mathbf{p}_t$. In an FL scenario, assuming for simplicity that all M agents have the same amount of data, we have that:

$$\mathbf{H}_t = \frac{1}{M} \sum_{i=1}^{M} \mathbf{H}_t^{(i)}, \quad \mathbf{g}_t = \frac{1}{M} \sum_{i=1}^{M} \mathbf{g}_t^{(i)}, \tag{2.2}$$

where $\mathbf{H}_t^{(i)} = \nabla^2 f^{(i)}(\boldsymbol{\theta}^t)$ and $\mathbf{g}_t^{(i)} = \nabla f^{(i)}(\boldsymbol{\theta}^t)$ denote local Hessian and gradient of the local cost $f^{(i)}(\boldsymbol{\theta}^t)$ of agent i, respectively. To get a Newton update at the master, in an FL setting one would need each agent to transfer the whole matrix $\mathbf{H}_t^{(i)}$ of size $O(n^2)$ to the master at each iteration, that is considered a prohibitive communication complexity in a federated learning setting, especially when the size of the feature data vectors, n, increases. Hence, in this work we propose an algorithm in which the Newton-type update takes the form:

$$\boldsymbol{\theta}^{t+1} = \boldsymbol{\theta}^t - \eta_t \hat{\mathbf{H}}_t^{-1} \mathbf{g}_t, \tag{2.3}$$

where $\hat{\mathbf{H}}_t$ is an approximation of $\mathbf{H}_t$. In some previous contributions, like [139], [144], $\hat{\mathbf{H}}_t$ is the harmonic mean of local Hessians, obtained at the master through an intermediate additional communication round to provide each agent with the global gradient, $\mathbf{g}_t$. In the recent FedNL algorithm [125], each agent at each iteration sends a compressed version of a Hessian-related matrix.

Our algorithm incrementally obtains at the master an average of full-rank approximations of local Hessians through the communication of the local Hessian most relevant eigenvalue-eigenvector pairs (EEPs) together with a carefully computed local approximation parameter. In particular, we approximate the Hessian matrix $\mathbf{H}_t$ exploiting eigendecomposition in the following way: the symmetric positive definite Hessian $\mathbf{H}_t$ can be diagonalized as $\mathbf{H}_t = \mathbf{V}_t \boldsymbol{\Lambda}_t \mathbf{V}_t^T$, with $\boldsymbol{\Lambda}_t = diag(\lambda_{1,t}, ..., \lambda_{n,t})$, where $\lambda_{k,t}$ is the eigenvalue corresponding to the k-th eigenvector, $\mathbf{v}_{k,t}$. $\mathbf{H}_t$ can be approximated as

$$\hat{\mathbf{H}}_t = \hat{\mathbf{H}}_t(\rho_t, q_t) = \mathbf{V}_t \hat{\boldsymbol{\Lambda}}_t \mathbf{V}_t^T = \sum_{k=1}^{q_t} (\lambda_{k,t} - \rho_t) \mathbf{v}_{k,t} \mathbf{v}_{k,t}^T + \rho_t \mathbf{I}, \tag{2.4}$$

with $\hat{\boldsymbol{\Lambda}}_t := \hat{\boldsymbol{\Lambda}}_t(\rho_t, q_t) = diag(\lambda_{1,t}, ..., \lambda_{q_t}, \rho_t, ..., \rho_t)$. The scalar $\rho_t > 0$ is the approximation parameter (if $\rho_t = 0$ this becomes a low-rank approximation). The integer $q_t = 1, ..., n$ denotes the number of EEPs $\{\lambda_{k,t}, \mathbf{v}_{k,t}\}_{k=1}^{q_t}$ being used to approximate the Hessian matrix. We always consider eigenvalues ordered so that $\lambda_{1,t} \geq \lambda_{2,t} \geq ... \geq \lambda_{n,t}$. The approximation of Eqn. (2.4) was used in [53] for a sub-sampled centralized optimization problem. Note that the EEPs up to the q-th can be efficiently computed via singular value decomposition [53]. In the FL setting, we use the approximation shown in Eqn. (2.4) to approximate the local Hessian matrices of the agents. In particular, letting $\hat{\mathbf{H}}_t^{(i)}$ be the approximated local Hessian of agent i, the approximated global Hessian is the average of the local approximated Hessian matrices:

$$\hat{\mathbf{H}}_t = \sum_{i=1}^{M} p_i \hat{\mathbf{H}}_t^{(i)}, \quad p_i = N_i/N$$

where $\hat{\mathbf{H}}_t^{(i)} := \hat{\mathbf{H}}_t(\rho_t^{(i)}, q_t^{(i)})$ is a function of the local approximation parameter $\rho_t^{(i)}$ and of the number of EEPs $q_t^{(i)}$ shared by agent i, denoted by $\{\lambda_{k,t}^{(i)}, \mathbf{v}_{k,t}^{(i)}\}_{k=1}^{q_t^{(i)}}$.

2.2.2 The algorithm in a nutshell

The idea of SHED is that agents share with the Master, together with the gradient, some of their Hessian EEPs, according to the available CRs. They share the EEPs in a decreasing order dictated by the value of the positive eigenvalues corresponding to the eigenvectors. At each iteration, they incrementally add new EEPs to the information they have sent to the Master. In a linear regression problem, in which the Hessian does not depend on the current parameter, agents would share their EEPs incrementally up to the n-th. When the n-th EEP is shared, the Master has the full Hessian available and no further second order information needs to be transmitted. In a general convex problem, in which the Hessian matrix changes at each iteration, being a function of the current parameter, SHED is designed in a way in which agents perform a *renewal* operation at certain iterations, i.e., they re-compute the Hessian matrix and re-start sharing the EEPs from the most relevant ones of the new matrix.

2.3 Linear regression (least squares)

In this section we illustrate our algorithm and present the convergence analysis considering the problem of solving (2.1) via Newton-type updates (2.3) in the least squares (LS) case, i.e., in the case of linear regression with quadratic cost. We start by considering LS because, in this case, the Hessian matrix is $\mathbf{H}(\boldsymbol{\theta}) = \mathbf{H}_{LS}$, $\forall \boldsymbol{\theta}$, i.e., it does not depend on the parameter $\boldsymbol{\theta}$. This fact makes the analysis and the algorithm much simpler than the general convex case. This, in turn, allows us to provide the main intuition behind SHED and to illustrate the effect of the incremental EEPs sharing strategy on the convergence rate when the eigenspectrum is constant. When considering LS, we write the eigendecomposition as $\mathbf{H}_{LS} = \mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^T$, with $\boldsymbol{\Lambda} = diag(\lambda_1, ..., \lambda_n)$ without specifying the iteration t when not needed. Let $\boldsymbol{\theta}^*$ denote the solution to (2.1). In the following, we use the fact that the gradient can be written as $\mathbf{g}_t = \mathbf{H}_{LS}(\boldsymbol{\theta}^t - \boldsymbol{\theta}^*)$. In this setup, the update rule of Eqn. (2.3) can be written as a time-varying linear discrete-time system:

$$\boldsymbol{\theta}^{t+1} - \boldsymbol{\theta}^* = \mathbf{A}_t(\boldsymbol{\theta}^t - \boldsymbol{\theta}^*) \tag{2.5}$$

where $\mathbf{A}_t := \mathbf{A}(\rho_t, \eta_t, q_t) = \mathbf{I} - \eta_t \hat{\mathbf{H}}_t^{-1}\mathbf{H}_{LS}$. Indeed,

$$\begin{aligned}\boldsymbol{\theta}^{t+1} - \boldsymbol{\theta}^* &= \boldsymbol{\theta}^t - \boldsymbol{\theta}^* - \eta_t \hat{\mathbf{H}}_t^{-1} \mathbf{g}_t \\ &= (\mathbf{I} - \eta_t \hat{\mathbf{H}}_t^{-1} \mathbf{H}_{LS})(\boldsymbol{\theta}^t - \boldsymbol{\theta}^*).\end{aligned}$$

Before moving to the FL case, we prove some results for the convergence rate of the centralized iterative least squares problem.

2.3.1 Centralized iterative least squares

In this sub-section, we study the optimization problem in the centralized case, so when all the data is kept in a single machine. We provide a range of choices for the approximation parameter ρ_t that are optimal in the convergence rate sense. We denote the convergence factor of the descent algorithm described by Eqn. (2.3) by

$$r_t := r(\rho_t, \eta_t, q_t),$$

making its dependence on the tuple (ρ_t, η_t, q_t) explicit.

Theorem 2.1. Consider solving problem (2.1) via Newton-type updates (2.3) in the least squares case. At iteration t, let the Hessian matrix $\mathbf{H}_{LS}$ be approximated as in Eqn. (2.4) (centralized case). The convergence rate is described by

$$\|\boldsymbol{\theta}^{t+1} - \boldsymbol{\theta}^*\| \leq r_t \|\boldsymbol{\theta}^t - \boldsymbol{\theta}^*\|. \tag{2.6}$$

For a given $q_t \in \{0, 1, ..., n\}$ the best achievable convergence factor is

$$r_t^* = r^*(q_t) := \min_{(\rho_t, \eta_t)} r(\rho_t, \eta_t, q_t) = (1 - \frac{\lambda_n}{\rho_t^*}), \tag{2.7}$$

where

$$\rho_t^* := (\lambda_{q_t+1} + \lambda_n)/2. \tag{2.8}$$

r_t^* is achievable if and only if $(\rho_t, \eta_t) \in \mathcal{S}^*$, with

$$\mathcal{S}^* = \{(\rho_t, \eta_t) : \rho_t \in [\lambda_n, \lambda_{q_t+1}], \eta_t = \frac{2\rho_t}{\lambda_{q_t+1} + \lambda_n}\}$$

Proof. See Appendix A.1. □

In the above Theorem, we have shown that the best convergence rate is achievable, by tuning the step size, as long as $\rho_t \in [\lambda_n, \lambda_{q_t+1}]$. In the following Corollary, we provide an optimal choice for the tuple (ρ_t, η_t) with respect to the estimation error.

Corollary 2.1. Among the tuples $(\rho_t, \eta_t) \in \mathcal{S}^*$, the choice of the tuple $(\rho_t^*, 1)$, with ρ_t^* defined in (2.8), is optimal with respect to the estimation error $\|\boldsymbol{\theta}_{\rho_t, \eta_t}^{t+1} - \boldsymbol{\theta}^*\|$, for any $\boldsymbol{\theta}^t$

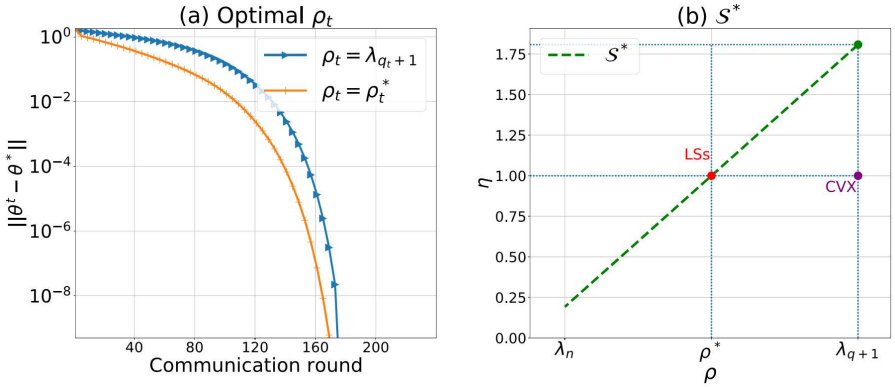

Figure 2.2: In (a) we show the different performance obtained with SHED w.r.t. different choices for the parameter ρ_t: the best choice in terms of estimation error from Corollary 2.1 is compared against the choice that was proposed in [53], that is $\rho_t = \lambda_{q_t+1}$. In the experiment, $q_t = q_{t-1} + 1$. In (b), we show an example of the set $\mathcal{S}^*$, and outline two points: LSs is the choice of the tuple in $\mathcal{S}^*$ providing the optimal convergence factor in LS, while CVX, which does not belong to $\mathcal{S}^*$, is the choice we do in the scenario of FL with convex cost.

and for any t, in the sense that

$$\|\boldsymbol{\theta}^{t+1}_{\rho_t^*,1} - \boldsymbol{\theta}^*\| \leq \|\boldsymbol{\theta}^{t+1}_{\rho_t,\eta_t} - \boldsymbol{\theta}^*\|, \quad \forall (\eta_t, \rho_t) \in \mathcal{S}^*$$

Proof. See Appendix A.2 □

See Fig. 2.2 for an illustration of the set S^* and the impact of the optimal choice in the performance. We remark that the bound in (2.6) is tight for $r_t = r_t^*$. If q_t increases, the convergence factor decreases until it becomes zero, when $q_t = n - 1$, thus we can have convergence in a finite number of steps.

2.3.2 Federated least squares

We now consider the FL scenario described in section 2.2, in which M agents keep their local data and share optimization parameters to contribute to the learning algorithm. For notation convenience, in the rest of the chapter we assume that each agent has the same amount of data samples, $N_i = \frac{N}{M}$. This allows us to express global functions, such as the gradient, as the arithmetic mean of local functions (e.g., $\mathbf{g}_t = (1/M)\sum_{i=1}^{M} \mathbf{g}_t^{(i)}$). This assumption is made only for notation convenience. It is straightforward to show that all the results are valid also when N_i is different for each i. To show this, it is sufficient to replace the arithmetic mean of local functions with the weighted average, weighting each local function with $p_i = N_i/N$.

We now introduce the algorithm for the LS case (Algorithm 1), that is a special case of Algorithm 5, described in Sec. 2.5, which is designed for a general convex cost. We refer to Algorithm 1 as SHED-LS and it works as follows: at iteration t, each agent

Algorithm 1 Least Squares, SHED-LS

Input: $\{\mathcal{D}_i\}_{i=1}^M = \{(\mathbf{X}_i, \mathbf{Y}_i)\}_{i=1}^M$, f, $\boldsymbol{\theta}^1$, $\nabla f(\boldsymbol{\theta}^1)$, $\mathcal{A} = \{agents\}$, $\mathcal{I} = \{1\}$, $\epsilon > 0$
Output: $\boldsymbol{\theta}^t$

$t \leftarrow 1$
while $\|\nabla f(\boldsymbol{\theta}^t)\|_2 \geq \epsilon$ **do**
  for *agent* $i \in \mathcal{A}$ **do**
    when Received $\boldsymbol{\theta}^t$ from the Master **do**
      if $t \in \mathcal{I}$ **then**
        compute $\mathbf{H}_{LS}^{(i)} = \nabla^2 f^{(i)}(\boldsymbol{\theta}^t) = \mathbf{X}_i \mathbf{X}_i^T$
        $\{(\lambda_j^{(i)}, \mathbf{v}_j^{(i)})\}_{j=1}^n \leftarrow \text{eigendecomp}(\mathbf{H}_{LS}^{(i)})$
        $q_0^{(i)} \leftarrow 0$
      end if
      compute $\mathbf{g}_t^{(i)} = \nabla f^{(i)}(\boldsymbol{\theta}^t)$
      set $d_t^{(i)}$ $q_t^{(i)} \leftarrow q_{t-1}^{(i)} + d_t^{(i)}$ // According to CRs
      $\rho_t^{(i)} \leftarrow (\lambda_{q_t^{(i)}+1}^{(i)} + \lambda_n^{(i)})/2$
      $U_t^{(i)} \leftarrow \{\{\mathbf{v}_j^{(i)}, \lambda_j^{(i)}\}_{j=q_{t-1}^{(i)}+1}^{q_t^{(i)}}, \mathbf{g}_t^{(i)}, \rho_t^{(i)}\}$
      Send $U_t^{(i)}$ to the Master.
  end for

  At the Master:
  when Received $U_t^{(i)}$ from all agents **do**
    compute $\hat{\mathbf{H}}_t^{(i)}$, $\forall i$. // see (2.10)
    Compute $\hat{\mathbf{H}}_t$ (as in eq. (2.9)) and $\mathbf{g}_t$.
    Perform Newton-type update (2.3) with $\eta_t = 1$.
    Broadcast $\boldsymbol{\theta}^{t+1}$ to all agents.
  $t \leftarrow t + 1$
end while

$i = 1, ..., M$ shares with the Master some of its EEPs together with its approximation parameter $\rho_t^{(i)}$. The eigenvectors are shared *incrementally*, where the order in which they are shared is given by the corresponding eigenvalues. For example, at $t = 1$ agent i will start by sharing its first Hessian EEPs $\{\mathbf{v}_j^{(i)}, \lambda_j^{(i)}\}_{j=1}^{q_1^{(i)}}$, according to its CRs, and then will incrementally send up to the last EEP $\{\lambda_{n-1}^{(i)}, \mathbf{v}_{n-1}^{(i)}\}$ in the following iterations. To enable the approximation of the local Hessian via a limited number of eigenvectors using (2.4), the parameter $\rho_t^{(i)}$ is sent as well. The Master averages the received information to obtain an estimate of the global Hessian as follows:

$$\hat{\mathbf{H}}_t = \frac{1}{M} \sum_{i=1}^{M} \hat{\mathbf{H}}_t^{(i)}, \tag{2.9}$$

where $\hat{\mathbf{H}}_t^{(i)}$ is

$$\hat{\mathbf{H}}_t^{(i)} := \hat{\mathbf{H}}^{(i)}(\rho_t^{(i)}, q_t^{(i)}) = \sum_{j=1}^{q_t^{(i)}} (\lambda_j^{(i)} - \rho_t^{(i)}) \mathbf{v}_j^{(i)} \mathbf{v}_j^{(i)T} + \rho_t^{(i)} \mathbf{I}, \tag{2.10}$$

in which $q_t^{(i)}$ is the number of the local Hessian EEPs that agent i has already sent to the Master at iteration t. We denote by $d_t^{(i)}$ the *increment*, meaning the number of eigenvectors that agent i can send to the Master at iteration t. Given the results shown in Sec. 2.3.1, in this section we fix the local approximation parameter to be $\rho_t^{(i)} = \rho_t^{(i)*} = (\lambda^{(i)}_{q_t^{(i)}+1} + \lambda_n^{(i)})/2$ and the step size to be $\eta_t = 1$. The following results related to the convergence rate allow the value $q_t^{(i)}$ to be different for each agent i, so we define

$$\mathbf{q}_t = [q_t^{(1)}, ..., q_t^{(M)}]^T, \; q_t^{(i)} \in \{0, ..., n-1\}, \; i = 1, ..., M$$

By construction, the matrix $\hat{\mathbf{H}}_t$ is positive definite, being the sum of positive definite matrices, implying that $-\mathbf{p}_t = -\hat{\mathbf{H}}_t^{-1}\mathbf{g}_t$ is a descent direction.

Theorem 2.2. Consider the problem in (2.1) in the least squares case. Given $\hat{\mathbf{H}}_t$ defined in (2.9), the update rule defined in (2.3) is such that, for $\rho_t^{(i)} \geq (\lambda^{(i)}_{q_t^{(i)}+1} + \lambda_n^{(i)})/2, \quad \forall i$, and $\eta_t = 1$:

$$\|\boldsymbol{\theta}^{t+1} - \boldsymbol{\theta}^*\| \leq c_t \|\boldsymbol{\theta}^t - \boldsymbol{\theta}^*\|, \tag{2.11}$$

with $c_t = (1 - \bar{\lambda}_n/\bar{\rho}_t)$ and

$$\bar{\rho}_t := \bar{\rho}(\mathbf{q}_t) = \frac{1}{M}\sum_{i=1}^{M} \rho_t^{(i)}, \quad \bar{\lambda}_n = \frac{1}{M}\sum_{i=1}^{M} \lambda_n^{(i)}. \tag{2.12}$$

If Algorithm 1 is applied, $c_{t+1} \leq c_t \; \forall t$, and, if for all i it holds that $q_{t'}^{(i)} = n - 1$, at some iteration t', $c_{t'} = 0$.

Proof. See Appendix A.3. □

This theorem shows that SHED-LS provides convergence in a finite number of iterations, if $q_t^{(i)}$ keeps increasing through time for each agent i. Indeed, as in the centralized case, if $q_t^{(i)}$ increases for all i, the factor c_t decreases until it becomes zero. Furthermore, each agent is free to send at each iteration an arbitrary number of EEPs, according to its CRs, and by doing so it can improve the convergence rate. See Fig. 2.2-(a) for an example of SHED-LS performance.

2.4 From least squares to general convex cost

We want to extend the analysis and algorithm presented in the previous sections of the chapter to a general convex cost $f(\boldsymbol{\theta}^t)$. With respect to the proposed approach, the general convex case requires special attention for two main reasons: (i) the update rule defined in (2.3) requires tuning of the step-size η_t, usually via backtracking line search, and (ii) the Hessian matrix is in general a function of the parameter $\boldsymbol{\theta}$. In this section, still focusing on the least squares case, we provide some results that are instrumental to the analysis of the general convex case.

2.4.1 Backtracking line search for step size tuning

We recall the well-known Armijo-Goldstein condition for accepting a step size η_t via backtracking line search:

$$f(\boldsymbol{\theta}^t - \eta_t \mathbf{p}) \leq f(\boldsymbol{\theta}^t) - \alpha \eta_t \mathbf{p}^T \mathbf{g}_t, \tag{2.13}$$

where $\alpha \in (0, 1/2)$. The corresponding line search algorithm is the following:

Algorithm 2 Backtracking line search algorithm

Input: $\alpha \in (0,1/2)$, $\beta \in (0,1)$, $\boldsymbol{\theta}^t$, $\mathbf{p}_t$, $\mathbf{g}_t$, f
Output: $\bar{\eta}_t$
  $\eta_t^0 \leftarrow 1$,
  $k \leftarrow 0$
  while $f(\boldsymbol{\theta}^t - \eta_t^{(k)}\mathbf{p}) > f(\boldsymbol{\theta}^t) - \alpha\eta_t^{(k)}\mathbf{p}^T\mathbf{g}_t$ **do**
    $k \leftarrow k+1$
    $\eta_t^{(k)} = \beta\eta_t^{(k-1)}$
    $\bar{\eta}_t = \eta_t^{(k)}$
  end while

Lemma 2.1. Consider the problem in (2.1) in the least squares case. Let $\mathbf{p}_t = \hat{\mathbf{H}}_t^{-1}\mathbf{g}_t$, with $\hat{\mathbf{H}}_t$ defined in (2.9). A sufficient condition for a step size η_t to satisfy Armijo-Goldstein condition (2.13), for any $\alpha \in (0, 1/2)$, is

$$\eta_t = \min_{i=1,\dots,M} \frac{\rho_t^{(i)}}{\lambda_{q_t^{(i)}+1}^{(i)}} \tag{2.14}$$

Proof. See Appendix A.4. □

Corollary 2.2. In the least squares case (Algorithm 1), when choosing $\rho_t^{(i)} = \rho_t^{(i)*}$, $\forall i$, Armijo backtracking line search (Algorithm 2) would choose a step size $\eta_t \geq \frac{1}{2}$. When choosing $\rho_t^{(i)} = \lambda_{q_t^{(i)}+1}$, $\forall i$, Algorithm 2 would choose a step size $\eta_t = 1$.

Proof. The proof is straightforward from Eqn. (2.14) of Lemma 2.1. □

Remark 2.3. For the choice $\rho^{(i)} = \rho^{(i)*}$, the Armijo-backtracking might not choose a step size $\eta_t = 1$ even for arbitrarily small $\alpha < 0.5$. Indeed, we can easily build a counter-example in the centralized case considering $\hat{\mathbf{H}}_t = \hat{\mathbf{\Lambda}}_t = diag(\lambda_1, ..., \lambda_q, \rho^*, ..., \rho^*)$, a gradient $\mathbf{g}_t$ such that $\mathbf{p}_t = \hat{\mathbf{H}}_t^{-1}\mathbf{g}_t = [0, ..., 0, 1, 0, ..., 0]^\top$ (e.g., $\mathbf{g}_t = [0, ...0, \rho^*, 0, ..., 0]^\top$). We see that with $\eta_t = 1$, Eq. (A.5) becomes $f(\boldsymbol{\theta}^{t+1}) = f(\boldsymbol{\theta}^t) - \lambda_n/2$ and, in order to be satisfied, the Armijo condition would require $\alpha \leq \frac{\lambda_n}{\lambda_{q+1}+\lambda_n}$, where the right hand side can become arbitrarily small depending on the eigenspectrum.

The results of Lemma 2.1 and of Corollary 2.2 and the counter-example of the above Remark are important for the design of the algorithm in the general convex case. Indeed, as illustrated in the next Section (Section 2.5), a requirement for the theoretical results on the convergence rate is that the step size becomes equal to one, which is not guaranteed by the Armijo backtracking line search, even when considering the least squares case, if $\rho_t^{(i)} < \lambda_{q_t^{(i)}+1}^{(i)}$. For this reason, the algorithm in the general convex case is designed with $\rho_t^{(i)} = \lambda_{q_t^{(i)}+1}^{(i)}$

2.4.2 Algorithm with periodic renewals

Now, we introduce a variant of Algorithm 1 that is instrumental to study the convergence rate of the proposed algorithm in the general convex case. The variant is Algorithm 3. The definition of $\mathcal{I} = \{1, T, 2T, ...\}$ implies that every T iterations the incremental strategy is restarted from the first EEPs of $\mathbf{H}_{LS}$, in what we call a periodic *renewal*. Differently

Algorithm 3 Variant of Algorithm 1, SHED-LS-periodic

In Algorithm 1, substitute $\mathcal{I} = \{1\}$ with $\mathcal{I} = \{1, T, 2T, ...\}$, for some input parameter $T < n$.

from Algorithm 1, Algorithm 3 can not guarantee convergence in a finite number of steps, because $T < n$ and thus it could be that $c_t > 0$, $\forall t$ (see Theorem 2.2). We study the convergence rate of the algorithm by focusing on upper bounds on the Lyapunov exponent [91] of the discrete-time dynamical system ruled by the descent algorithm. The Lyapunov exponent characterizes the rate of exponential (linear) convergence and it is defined as the positive constant $a_* > 0$ such that, considering $h(\boldsymbol{\theta}^t) := (\boldsymbol{\theta}^t - \boldsymbol{\theta}^*)a^{-t}$,

if $a > a_*$ then $h(\boldsymbol{\theta}^t)$ vanishes with t, while if $a < a_*$, for some initial condition, $h(\boldsymbol{\theta}^t)$ diverges. The usual definition of Lyapunov exponent for discrete-time linear systems [40] is, considering the system defined in (2.5),

$$a_* := \limsup_{t\to\infty} \|\boldsymbol{\Psi}_t\|^{1/t}, \ \boldsymbol{\Psi}_t = \mathbf{A}_1 \cdots \mathbf{A}_t. \tag{2.15}$$

From (2.11), we have that, for each k, $\|\mathbf{A}_k\| \leq c_k = (1 - \bar{\lambda}_n/\bar{\rho}_k)$. This implies that, defining

$$\begin{aligned} a_t &:= (\prod_{k=1}^{t} c_k)^{1/t}, \\ \bar{a} &:= \limsup_{t\to+\infty} a_t, \end{aligned} \tag{2.16}$$

it is $a_* \leq \bar{a}$. The following Lemma formalizes this bound and provides an upper bound on the Lyapunov exponent obtained by applying Algorithm 3.

Lemma 2.2. Let $c_k = (1 - \bar{\lambda}_n/\bar{\rho}_k)$ (see Eq. (2.12)). Applying Algorithm 3, the Lyapunov exponent of system (2.5) is such that

$$a_* \leq \bar{a} = \limsup_{t\to+\infty} (\prod_{k=1}^{t} c_k)^{1/t}.$$

If $q_t^{(i)} = q_{t-1}^{(i)} + 1, \ \forall i, t,$

$$a_* \leq \bar{a}_T := (\prod_{k=1}^{T} c_k)^{1/T}. \tag{2.17}$$

where $\bar{a}_T$ is such that $\bar{a}_{T+1} \leq \bar{a}_T$ and $\bar{a}_n = 0$.

Proof. See Appendix A.5 □

2.5 Federated learning with convex cost

Given the previous analysis and theoretical results for linear regression with quadratic cost, we are now ready to illustrate our Newton-type algorithm (Algorithm 5) for general convex FL problems, of which Algorithm 1 is a special case. We refer to the this general version of the algorithm simply as SHED. Since in a general convex problem the Hessian depends on the current parameter, $\boldsymbol{\theta}^t$, we denote by $\mathbf{H}(\boldsymbol{\theta}^t)$ the global Hessian at the current iterate, while we denote by $\hat{\mathbf{H}}_t$ the global approximation, defined similarly to (2.9), with the difference that now eigenvalues and eigenvectors depend on the parameter

for which the Hessian was computed.
We therefore write $\hat{\mathbf{H}}_t$ in the following way:

$$\hat{\mathbf{H}}_t = \frac{1}{M}\sum_{i=1}^{M} \hat{\mathbf{H}}_t^{(i)}(\boldsymbol{\theta}^{k_t^{(i)}}), \tag{2.18}$$

where $k_t^{(i)} \leq t$ denotes the iteration in which the local Hessian of agent i was computed. The parameter $\boldsymbol{\theta}^{k_t^{(i)}}$ is the parameter for which agent i computed the local Hessian, that in turn is being used for the update at iteration t.
The idea of the algorithm is to use previous versions of the Hessian rather than always recomputing it. This is motivated by the fact that as we approach the solution of the optimization problem, the second order approximation becomes more accurate and the Hessian changes more slowly. Hence, recomputing the Hessian and restarting the incremental approach provides fewer and fewer advantages as we proceed. From time to time, however, we need to re-compute the Hessian corresponding to the current parameter $\boldsymbol{\theta}^t$, because $\mathbf{H}(\boldsymbol{\theta}^{k_t^{(i)}})$ could have become too different from $\mathbf{H}^{(i)}(\boldsymbol{\theta}^t)$. As in Sec. 2.4.2, we call this operation a *renewal*. We denote by $\mathcal{I}$ the set of iteration indices at which a renewal takes place. In principle, each agent could have its own set of renewal indices, and decide to recompute the Hessian matrix independently. In this work, we consider for simplicity that the set $\mathcal{I}$ is the same for all agents, meaning that all agents use the same parameter for the local Hessian computation, i.e., $k_t^{(i)} = k_t,\ \forall i$. At the end of this section we describe heuristic strategies to choose $\mathcal{I}$ with respect to the theoretical analysis. We remark that in the case of a quadratic cost, in which the Hessian is constant, one chooses $\mathcal{I} = \{1\}$, and so Algorithm 1 is a special case of Algorithm 5.
The eigendecomposition can be applied to the local Hessian as before, we define

$$\hat{\mathbf{v}}_{j,t}^{(i)} = \mathbf{v}_j^{(i)}(\boldsymbol{\theta}^{k_t}), \quad \hat{\lambda}_{j,t}^{(i)} = \lambda_j^{(i)}(\boldsymbol{\theta}^{k_t}). \tag{2.19}$$

For notation convenience we also define

$$\tilde{\mathbf{v}}_{j,t}^{(i)} = (\hat{\lambda}_{j,t}^{(i)} - \hat{\rho}_t^{(i)})^{1/2}\hat{\mathbf{v}}_{j,t}^{(i)}, \tag{2.20}$$

Our theoretical results on the convergence rate hold if, for some $\bar{t} > 0$, the backtracking strategy always chooses $\eta_t = 1$, $\forall t \geq \bar{t}$. To meet this requirement, the analysis requires that $\hat{\mathbf{H}}^{(i)}(\boldsymbol{\theta}^{k_t}) \geq \mathbf{H}^{(i)}(\boldsymbol{\theta}^{k_t}), \forall t, i$ which in turn requires $\rho_t^{(i)}(\boldsymbol{\theta}^{k_t}) \geq \hat{\lambda}^{(i)}_{q_t^{(i)}+1,t}$. Accordingly, we set:

$$\rho_t^{(i)}(\boldsymbol{\theta}^{k_t}) := \hat{\rho}_t^{(i)} = \hat{\lambda}^{(i)}_{q_t+1,t}. \tag{2.21}$$

The local Hessian can be approximated as

$$\hat{\mathbf{H}}_t^{(i)}(\boldsymbol{\theta}^{k_t}) = \sum_{j=1}^{q_t^{(i)}} \tilde{\mathbf{v}}_{j,t}^{(i)} \tilde{\mathbf{v}}_{j,t}^{(i)T} + \hat{\rho}_t^{(i)} \mathbf{I}. \tag{2.22}$$

Clearly, it still holds that $\hat{\mathbf{H}}_t \geq \bar{\rho}_t \mathbf{I}, \ \forall t$, where $\bar{\rho}_t = \frac{1}{M} \sum_{i=1}^{M} \hat{\rho}_t^{(i)}$. Furthermore, it is easy to see that $\hat{\mathbf{H}}_t \leq K\mathbf{I}$, with K the smoothness constant of f. Note that SHED uses the Armijo backtracking strategy, that is recalled in Algorithm 2.

Theorem 2.4. For any initial condition, SHED (Algorithm 5) ensures convergence to the optimum, i.e.,

$$\lim_{t \to +\infty} \|\boldsymbol{\theta}_t - \boldsymbol{\theta}^*\| = 0.$$

Proof. See Appendix A.6. □

In the remainder of this section, we provide the convergence analysis of SHED for the general strongly convex cost case. The results of Theorem 2.5 and 2.6 are the main theoretical contribution of this chapter. In order to provide the convergence guarantees of SHED, we need to impose some constraints on the renewal indices set $\mathcal{I}$. Specifically,

Assumption 2. Denoting $\mathcal{I} = \{I_j\}_{j \in \mathbb{N}}$, there exists a finite positive integer $\bar{l}$ such that $I_j \leq I_{j-1} + \bar{l}, \ \forall j$.

In words, the above assumption implies that writing $k_t = t - \tau$, the 'delay' τ is bounded. The next theorem provides a bound that relates the convergence rate and the increments of outdated Hessians.

Theorem 2.5. Applying SHED (Algorithm 5), for any iteration t, it holds that:

$$\|\boldsymbol{\theta}^{t+1} - \boldsymbol{\theta}^*\| \leq c_{1,t} \|\boldsymbol{\theta}^t - \boldsymbol{\theta}^*\| + c_{2,t} \|\boldsymbol{\theta}^t - \boldsymbol{\theta}^*\|^2 \tag{2.23}$$

where, defining $\bar{\lambda}_{n,t} = \frac{1}{M} \sum_{i=1}^{M} \hat{\lambda}_{n,t}^{(i)}$ and $\bar{\rho}_t = \frac{1}{M} \sum_{i=1}^{M} \hat{\rho}_t^{(i)}$ (see (2.19) and (2.21)),

$$\begin{aligned} c_{1,t} &= (1 - \frac{\bar{\lambda}_{n,t}}{\bar{\rho}_t}) + \frac{L}{\bar{\rho}_t} \|\boldsymbol{\theta}^t - \boldsymbol{\theta}^{k_t}\| + (1 - \eta_t) \frac{\|\mathbf{H}(\boldsymbol{\theta}^t)\|}{\bar{\rho}_t}, \\ c_{2,t} &= \frac{\eta_t L}{2\bar{\rho}_t}. \end{aligned} \tag{2.24}$$

Proof. See Appendix A.7. □

The above theorem is a generalization of Lemma 3.1 in [53] (without sub-sampling). In particular, the difference is that (i) the dataset is distributed and (ii) an outdated Hessian

is used. The next theorem formally establishes linear and super-linear convergence of SHED.

Theorem 2.6. Recall the definition of the average strong convexity constant $\bar{\kappa} = (1/M)\sum_{i=1}^{M}\kappa_i$, with κ_i the strong convexity constant of agent i. Let K be the smoothness constant of f. The following results hold:

1. Applying SHED (Algorithm 5), as soon as

$$3\bar{\kappa}(M(t) + \|\boldsymbol{\theta}^t - \boldsymbol{\theta}^*\|) + K\|\boldsymbol{\theta}^t - \boldsymbol{\theta}^*\| \leq \frac{3\bar{\kappa}^2}{L}(1-2\alpha), \tag{2.25}$$

and

$$\frac{3}{2}L\|\boldsymbol{\theta}^t - \boldsymbol{\theta}^*\| + LM(t) \leq \bar{\kappa}, \tag{2.26}$$

with $M(t) = \max\{\|\boldsymbol{\theta}^t - \boldsymbol{\theta}^*\|, \|\boldsymbol{\theta}^{k_t} - \boldsymbol{\theta}^*\|\}$, the algorithm enjoys at least linear convergence.

2. Define $\mathcal{X}_t := \{k \leq t : \bar{\rho}_k = \bar{\lambda}_{n,k}\}$. Let $|\mathcal{X}_t|$ denote the cardinality of $\mathcal{X}_t$. If $|\mathcal{X}_t| = 0$ then SHED enjoys linear convergence and the Lyapunov exponent of the estimation error can be upper bounded as

$$a_* \leq \limsup_t \Big(\prod_{k=1}^{t} 1 - \frac{\bar{\lambda}_n^o}{\bar{\rho}_k^o}\Big)^{1/t} \tag{2.27}$$

where $\bar{\lambda}_n^o$ and $\bar{\rho}_k^o$ are the average of the n-th eigenvalues and approximation parameters, respectively, computed at the optimum:

$$\bar{\lambda}_n^o = \frac{1}{M}\sum_{i=1}^{M}\lambda_n^{(i)}(\boldsymbol{\theta}^*), \quad \bar{\rho}_k^o = \frac{1}{M}\sum_{i=1}^{M}\rho_k^{(i)o},$$

with $\rho_k^{(i)o} = \lambda^{(i)}_{q_k^{(i)}+1}(\boldsymbol{\theta}^*)$.

3. Let $|\mathcal{X}_t|$ denote the cardinality of $\mathcal{X}_t$. Let $\bar{T} > 0$ be finite. If $|\mathcal{X}_t| \geq t^{1/2}h(t) - \bar{T}$, with $h(t)$ any function such that $h(t) \to \infty$ as $t \to \infty$, then the Lyapunov exponent is $a_* = 0$ and thus SHED enjoys super-linear convergence.

Sketch of proof. For 1), we first show that when condition (2.25) holds, the step size is chosen equal to one by the Armijo backtracking line search. We then show that when also (2.26) holds, then the cost converges at least linearly for any subsequent iteration. For 2), we upper bound the Lyapunov exponent and exploit local Lipschitz continuity to provide the result. For 3), we exploit at least linear convergence proved in

1) together with the assumption on the cardinality of the set $\mathcal{X}_t$. For the complete proof, see Appendix A.8. □

In the above theorem, we have shown that SHED enjoys at least linear convergence and provided a sufficient condition on the choice of renewals indices set to guarantee *global* super-linear convergence. In sharp contrast to existing works, global asymptotic super-linear convergence is guaranteed (i) regardless of the initial condition, and (ii) without requiring the knowledge of problem specific constants, as long as the requirement on the set $\mathcal{X}_t$ is satisfied. The sufficient condition can be easily guaranteed with a choice of periodic renewals with a period such that the cardinality of $\mathcal{X}_t$ is big enough. A simple example is as follows. Let $\mathcal{I} = \{1, n, 2n, ...\}$, and let $d_t^{(i)} = 1$ for each agent i and for each t, meaning that at each iteration, each agent transmits a single EEP together with the gradient. In this case, at iterations $k \in \{n-1, 2n-1, 3n-1, ...\}$ we have $\bar{\rho}_k = \bar{\lambda}_{n,k}$, and thus $|\mathcal{X}_t| = \lfloor t/n \rfloor$. Note that in this case $|\mathcal{X}_t| \geq t/n - 1$, and the condition for super-linearity is satisfied with $h(t) = \frac{t^{1/2}}{n}$ and $\bar{T} = 1$. Note that if $|\mathcal{X}_t| = 0, \ \forall t$, with renewal period T and $d_t^{(i)} = 1 \ \forall i, t$, we can get

$$a_* \leq \bar{a}_T := \prod_{k=1}^{T} (1 - \frac{\bar{\lambda}_n^o}{\bar{\rho}_k^o})^{1/T}. \tag{2.28}$$

2.5.1 Heuristics for the choice of $\mathcal{I}$

From the theoretical results provided above we can do a heuristic design for the renewal indices set $\mathcal{I}$. In particular, we see from the bound (2.23) in Theorem 2.5 that when we are at the first iterations of the optimization we would frequently do the renewal operation, given that the Hessian matrix changes much faster and that the term $\|\boldsymbol{\theta}^{k_t} - \boldsymbol{\theta}^t\|$ is big. As we converge, instead, we would like to reduce the number of renewal operations, to improve the convergence rate, and this is strongly suggested by the result 2) related to the Lyapunov exponent in Theorem 2.6. Furthermore, the super-linear convergence that follows from 3) in the same Theorem suggests to keep performing renewals in order to let the cardinality of $\mathcal{X}_t$ to grow sufficiently fast with t.
To evaluate SHED, we obtain the results of the next section using two different renewal strategies. In the first, we choose the distance between renewals to be determined by the *Fibonacci* sequence, so $\mathcal{I} = \{I_j\}$, where $I_j = \sum_{k=1}^{j} F_k$, F_k being the *Fibonacci* sequence, with $F_0 = 0, F_1 = 1$. When the sequence I_j reaches $n-1$, the next values of the sequence are chosen so that $I_{j+1} = I_j + n - 1$. We call this method Fib-SHED. Note that this strategy is compliant with the requirements for super-linearity of Theorem 2.6. The second strategy is based on the inspection of the value of the gradient norm, $\|\mathbf{g}_t\|$, which

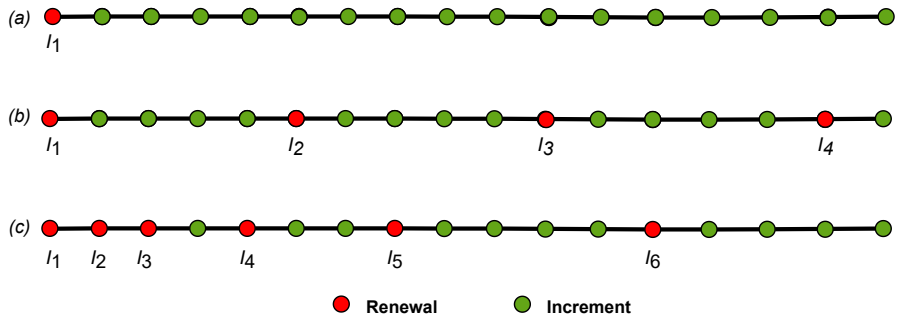

Figure 2.3: Illustration of possible choices of renewal indices set. The set $\mathcal{I} = \{I_j\}$ specifies the iterations at which a renewal takes place. (a) illustrates the least squares case in which renewal is performed only once, see Algorithm 1, (b) the periodic renewals case with $T = 5$, so $\mathcal{I} = \{1, 5, 10, ...\}$, see Algorithm 3, and (c) the set in which the distance between renewals increases according to the Fibonacci sequence.

is directly related to $\|\boldsymbol{\theta}^t - \boldsymbol{\theta}^*\|$. In particular, we make a decision concerning renewals at each iteration by evaluating the empirically observed decrease in the gradient norm. If $\|\mathbf{g}_{t-1}\| - \|\mathbf{g}_t\| < b(\|\mathbf{g}_{t-2}\| - \|\mathbf{g}_{t-1}\|)$ for some constant b, this strategy triggers a renewal. To guarantee at least linear convergence, we impose a renewal after n iterations in which no renewal has been triggered. In the rest of the chapter we call this strategy GN-SHED (Gradient Norm-based SHED). We remark that, contrary to Fib-SHED, GN-SHED does not guarantee super-linear convergence. See Fig. 2.3 for an illustration of some different possible choices for the renewal indices sets.

Algorithm 4 FL with convex cost - SHED

Input: $\{\mathcal{D}_i\}_{i=1}^M$, $\mathcal{I}$, $\boldsymbol{\theta}^1$, f, $\nabla f(\boldsymbol{\theta}^1)$, $\mathcal{A} = \{agents\}$, $\epsilon > 0$
Output: $\boldsymbol{\theta}^t$

$t \leftarrow 1$
while $\|\nabla f(\boldsymbol{\theta}^t)\|_2 \geq \epsilon$ **do**
  for *agent* $i \in \mathcal{A}$ **do**
    when Received $\boldsymbol{\theta}^t$ from the Master **do**
      if $t \in \mathcal{I}$ **then**
        $k_t \leftarrow t$
        compute $\mathbf{H}_t^{(i)} = \nabla^2 f^{(i)}(\boldsymbol{\theta}^t)$ // renewal
        $\{(\hat{\lambda}_{j,t}^{(i)}, \hat{\mathbf{v}}_{j,t}^{(i)})\}_{j=1}^n \leftarrow \text{eigendecomp}(\mathbf{H}_t^{(i)})$
        $q_{t-1}^{(i)} \leftarrow 0$
      end if
      compute $\mathbf{g}_t^{(i)} = \mathbf{g}^{(i)}(\boldsymbol{\theta}^t) = \nabla f^{(i)}(\boldsymbol{\theta}^t)$
      set $d_t^{(i)}$ // according to CRs
      $q_t^{(i)} \leftarrow q_{t-1}^{(i)} + d_t^{(i)}$ // increment
      $\hat{\rho}_t^{(i)} \leftarrow \hat{\lambda}_{q_t^{(i)}+1,t}^{(i)}$ // approximation parameter
      $U_t^{(i)} \leftarrow \{\{\hat{\mathbf{v}}_{j,t}^{(i)}, \hat{\lambda}_{j,t}^{(i)}\}_{j=q_{t-1}^{(i)}+1}^{q_t^{(i)}}, \mathbf{g}_t^{(i)}, \hat{\rho}_t^{(i)}\}$ // see (2.20) and (2.22)
      Send $U_t^{(i)}$ to the Master
  end for

  At the Master:
  when Received $U_t^{(i)}$ from all agents **do**
    compute $\hat{\mathbf{H}}_t^{(i)}$, $\forall i$. // see (2.22)
    Compute $\hat{\mathbf{H}}_t$ (as in eq. (2.18)) and $\mathbf{g}_t$.
    Get η_t via federated backtracking line search.
    Perform Newton-type update (2.3).
    Broadcast $\boldsymbol{\theta}^{t+1}$ to all agents.
  $t \leftarrow t + 1$
end while

2.6 Empirical Results

In this section we present our empirical results with real datasets. We experimented with the Million Song (1M Songs) dataset [16] for LS, and FMNIST [146], EMNIST [37] and 'w8a', available from libSVM [26] for logistic regression. We compare SHED against state-of-the-art approaches in both i.i.d. and non i.i.d. data distributions. We also show the resilience and superiority of SHED under ill-conditioning. Here, we focus mainly on illustrating the results obtained with FMNIST, and we defer the results obtained with the other dasatets to Appendix 2.7 and to the technical report [54]. In Appendix 2.7, we also include additional experiments that illustrate the working principle of SHED. For FMNIST, we consider one-vs-all binary classification with $M = 28$ agents, with a parameter size (after PCA [129]) of $n = 300$. For more details on the datasets setup and on the partitions (i.i.d. and non i.i.d.), see Sec. 6.1 of the technical report [54].
In the convex case, we follow the heuristics discussed at the end of Sec. 2.5, showing the results for both Fib-SHED and GN-SHED. If the value of $d_t^{(i)}$ is not specified, then $d_t^{(i)} = 1$ for each iteration t and agent $i = 1, ..., M$. to show that SHED is effective in the case of heterogeneous (per-iteration) CRs. In this section, we show results with regularizer $\mu = 10^{-5}, 10^{-6}, 10^{-8}$, and we have obtained similar results with $\mu = 10^{-4}$. Note that $10^{-4}, 10^{-6}, 10^{-8}$ were the values considered in [144].

2.6.1 Federated backtracking

To tune the step size η_t when there are no guarantees that $\eta_t = 1$ decreases the cost, we adopt the same strategy adopted in [144]: an additional communication round takes place in which each agent shares with the master the loss obtained when the parameter is updated via the new descent direction for different values of the step size. In this way, we can apply a distributed version of the popular Armijo backtracking line search (see Algorithm 2). When showing the results with respect to communication rounds, we always include also the additional communication round due to backtracking.

2.6.2 Comparison against other algorithms

In the following, SHED+ means that the number of Hessian EEPs is chosen randomly for each agent i, $d_t^{(i)} = d_\gamma$, with an average increment equal to 4 (see Sec. 6.3 of the technical report [54] for more details on the choice of the random variable). We compare the performance of our algorithm with a state-of-the art first-order method, accelerated gradient descent (AGD, with the same implementation of [144]). As benchmark second-order methods we consider a distributed version of the Newton-type method proposed in [53], to which we refer as Mont-Dec, which is the same as Algorithm 5 with the

difference that the renewal occurs at each communication round, so the Hessian is always recomputed and the second-order information is never outdated. We then compare SHED against BFGS [87], GIANT [144] and the recently proposed FedNL [125]. For more details on the implementations, see the technical report (Sec. 6.7 in [54]).

Communication complexity. In Fig. 2.4 we show the results obtained with FMNIST. We show the performance of Fib-SHED, Fib-SHED+ and GN-SHED. For GN-SHED, we fix the constant $b = 0.95$. To provide a complete comparison, in Fig. 2.4-(c)-(d) we show also the relative cost versus the overall amount of data transmitted, in terms of number of vectors in $\mathbb{R}^n$ transmitted. From Fig. 2.4, we can see how the non i.i.d. configuration causes a performance degradation for GIANT, while SHED, BFGS and FedNL are not impacted.
In the i.i.d. case, we see that GN-SHED, Fib-SHED, Fib-SHED+ and GIANT require a similar number of communication rounds to converge, and a similar amount of overall data transmitted per agent. On the other hand, FedNL shows a much slower convergence speed with respect to Fib-SHED and GN-SHED (requiring the same per-iteration communication load). Notice, for FedNL, in Figure 2.4-(c)-(d), the impact that the transmission of the full Hessian matrices at the first round has on the overall communication load. In the considered non i.i.d. case, the large advantage that our approach can provide with respect to the other considered algorithms is strongly evident in both data transmitted and communication rounds. Comparing the SHED approaches against Mont-Dec we see the key role that the *incremental* strategy exploiting *outdated* second order information has on the convergence speed of our approach. Indeed, even though in the first iterations the usage of the current Hessian information provides the same performance of the SHED methods, the performance becomes largely inferior in the following rounds. From a computational point of view, both the Mont-Dec and the FedNL approaches are much more demanding relative to SHED as they require that each agent recomputes the Hessian and SVD at each round. Fib-SHED, instead, requires the agents to compute the Hessian matrix only 12 times out of the 450 rounds needed for convergence. For more details on this, see Figure 2.6.
In Fig. 2.5, we show how SHED is much more resilient to ill-conditioning with respect to competing algorithms, by comparing the convergence performance when the regularization parameter is $\mu = 10^{-5}$ and $\mu = 10^{-8}$. With respect to FedNL, note how Fib-SHED worsens its performance when $\mu = 10^{-8}$ by being around 2.5 times slower compared to the case $\mu = 10^{-5}$, while FedNL worsens much more, being more than 8 times slower when $\mu = 10^{-8}$ compared to the case $\mu = 10^{-5}$.

Computational complexity. In the technical report [54], we show the results

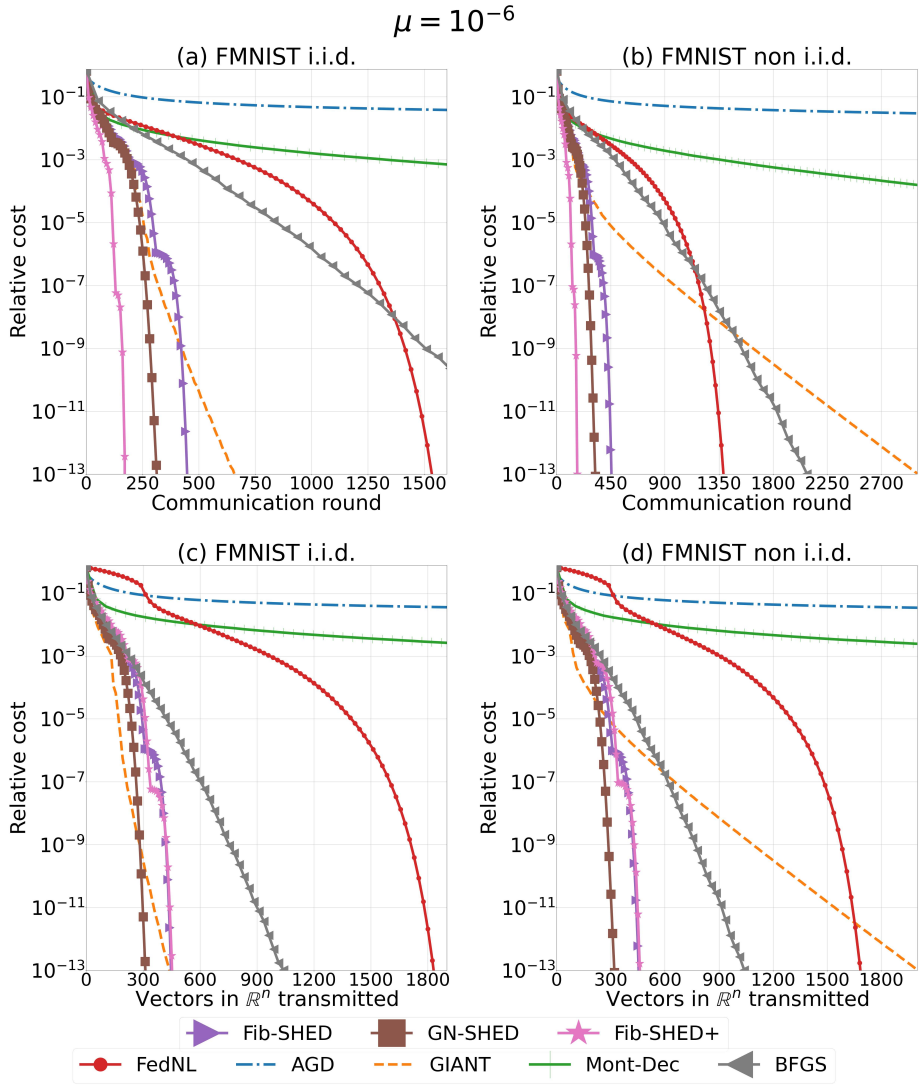

Figure 2.4: Performance comparison of logistic regression on FMNIST when $\mu = 10^{-6}$. Relative cost is $f(\boldsymbol{\theta}^t) - f(\boldsymbol{\theta}^*)$.

obtained with the EMNIST and w8a datasets. In the case of EMNIST, we obtain results similar to FMNIST, except that, when $\mu = 10^{-5}$, GIANT is not much impacted by the considered non i.i.d. configuration. Even if in that case GIANT seems to be the best choice, all the other results show that GIANT and related approaches based on the harmonic mean (like DONE) are strongly sensitive to non i.i.d. data distributions. With image datasets (EMNIST and FMNIST), FedNL is largely outperformed by SHED, while, with the 'w8a' dataset, FedNL is more competitive. However, the SHED approaches have the very appealing feature that they require agents to compute the local Hessian

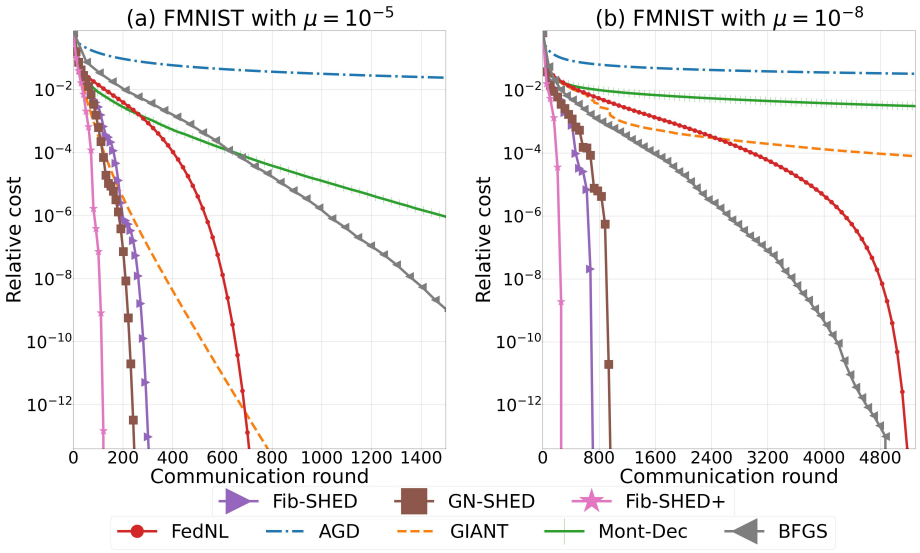

Figure 2.5: Performance of logistic regression on FMNIST, comparing $\mu = 10^{-5}$ and $\mu = 10^{-8}$. Relative cost is $f(\boldsymbol{\theta}^t) - f(\boldsymbol{\theta}^*)$.

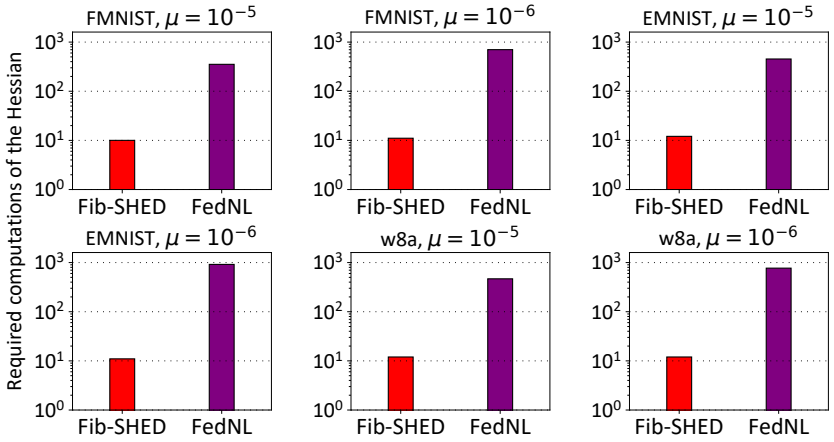

Figure 2.6: In this plots, we show, for three datasets (FMNIST, EMNIST and w8a), the number of times an agent is required to compute the local Hessian matrix in order for an algorithm to converge, comparing the proposed Fib-SHED and the FedNL [125] algorithms.

matrices only *sporadically*. FedNL, instead, requires that the Hessian is recomputed by the agents at each round, thus it is much more computationally demanding. To better illustrate and quantify this advantage, we show, in Figure 2.6, the number of times that an agent is required to compute the local Hessian matrix in order to obtain convergence, comparing Fib-SHED and FedNL, in the cases of the three datasets. In the case of EMNIST and FMNIST, we are showing the non i.i.d. configurations, but similar results

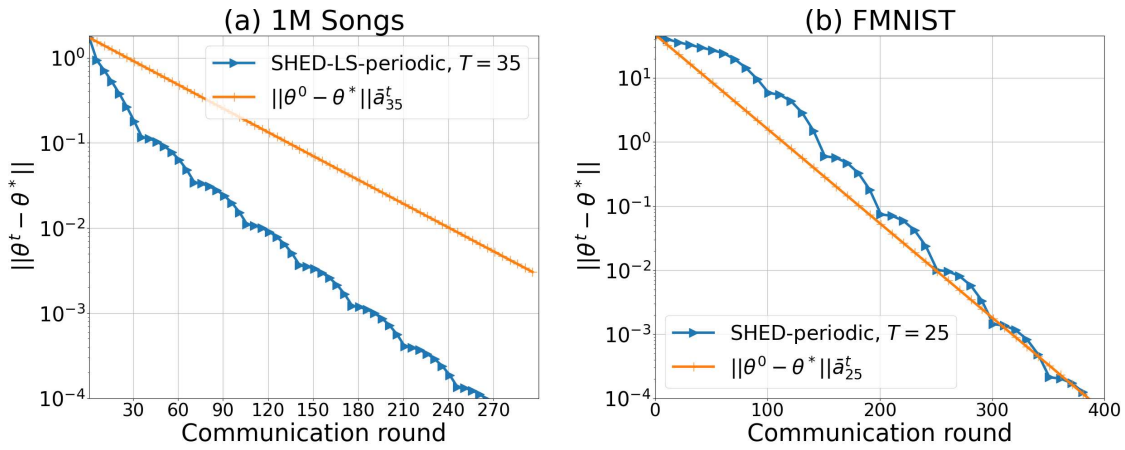

Figure 2.7: Linear convergence with periodic renewals illustrated via the study of the upper bound on the dominant Lyapunov exponent of the estimation error from equation (2.16) and point 2) of Theorem 2.6. $\bar{a}_T$ is as in eqs. (2.16) and (2.28) for (a) and (b), respectively. Note that the upper bound is on the slope of the decreasing cost.

can be obtained with the i.i.d. ones. The results show that, compared with Fib-SHED, the number of times agents are required to compute the Hessian is always at least ten times greater for FedNL to converge.

2.7 Additional Experiments

In this section, we provide some additional experiments that better illustrate the working principles of SHED and corroborate our theoretical analysis. **Lyapunov exponent convergence bound.** In Fig. 2.7, we illustrate how the Lyapunov exponent bounds derived in Sec. 2.4.2 and 2) of Theorem 2.6 characterize the linear convergence rate of the algorithms. In particular, we consider the cases of periodic renewals in both the LSs and logistic regression case, letting $q_t^{(i)} = q_{t-1}^{(i)} + 1$, $\forall i, t$ and renewals are periodic with period T (see (2.28) for the convex case). The plots show how the linear convergence rate is dominated by the Lyapunov exponent bound characterized by $\bar{a}_T$. For illustrative purposes, we show the results for the choice of $T = 35$ and of $T = 25$ for LSs on 1M songs and logistic regression on FMNIST, respectively.

Communication complexity analysis. In Figure 2.8, we show the impact of the number of EEPs transmitted per communication rounds when $d_t^{(i)}$ takes different values. We consider the case when the number is the same and fixed (specifically we consider $d_t^{(i)} \in \{1, 3, 6, 30\}$) for all the agents and the case $d_t^{(i)} = d_\gamma$. In this latter case each agent is able to transmit a different random number of EEPs. This configuration is relevant as our algorithm allows agents to contribute to the optimization according to their specific

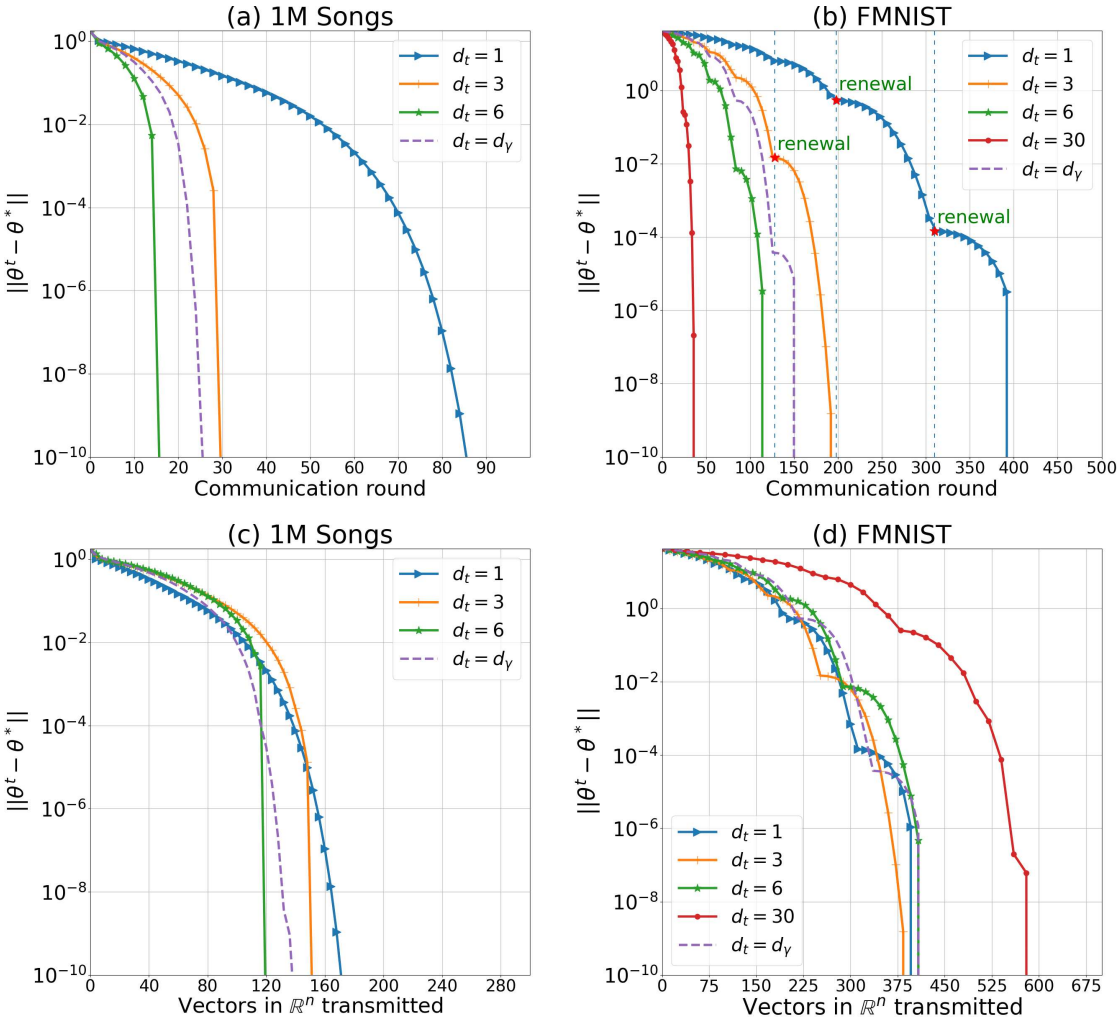

Figure 2.8: Performance comparison for different values of d_t for (i) linear regression on 1M Songs ((a) and (c)) and (ii) logistic regression on FMNIST ((b) and (d)). In (b), we emphasize three points where the renewal operation (see Algorithm 5) takes place.

CRs. In Figures 2.8-(a)-(c) we show the results for LSs on 1M Songs, while in Figures 2.8-(b)-(d) we show the results in the convex case on FMNIST. We can see from Figures 2.8-(a)-(b) how the global number of communication rounds needed for convergence can be significantly reduced by increasing the amount of information transmitted at each round. In particular, at each round, the number of vectors in $\mathbb{R}^n$ being transmitted is $d_t + 1$, since together with the d_t scaled eigenvectors, $\{\tilde{\mathbf{v}}_{j,t}^{(i)}\}_{j=q_{t-1}^{(i)}+1}^{q_t^{(i)}}$ (see Algorithm 5), agents need to transmit also the gradient. In the LSs case, the number of iterations needed for convergence is smaller than the number of EEPs sent: when the $n-1$-th EEP has been shared, convergence occurs. When the number of EEPs is random (d_γ), but

equal to 4 in average, we see that we can still get a significant improvement
For the case of regularized convex cost (logistic regression), in Figures 2.8-(b)-(d) we emphasize the role of the *renewal* operation, showing also how incrementally adding EEPs of the outdated Hessian improves the convergence, as formalized and shown in Theorems 2.5 and 2.6. In particular, from Figure 2.8-(b) it is possible to appreciate the impact of increasing the interval between renewals.
In Figures 2.8-(c)-(d) we plot the error as a function of the amount of data transmitted per agent, where the number of vectors in $\mathbb{R}^n$ is the unit of measure. These plots show that, for small values of d_t (in particular, $d_t \in \{1, 3, 6\}$), the overall data transmitted does not increase for the considered values of d_t, meaning that we can significantly reduce the number of global communication rounds by transmitting more data per round without increasing the overall communication load. This is true in particular also for the case $d_t = d_\gamma$, thus when the agents' channels availability is heterogeneous at each round, showing that our algorithm works even in this relevant scenario without increasing the communication load. On the other hand, in the case of $d_t = 30$, even if we get faster convergence, we pay with a significant increase in the overall communication load that the network has to take care of.

2.8 Conclusions

In this work, we have proposed SHED, an NT algorithm for FL that enjoys global asymptotic super-linear convergence. SHED is versatile with respect to agents' (per-iteration) CRs and operates effectively in the presence of *non i.i.d.* data distributions, outperforming state-of-the-art techniques. SHED achieves better performance with respect to the competing FedNL approach, while involving sporadic Hessian computations. In the case of i.i.d. data statistics, SHED is also competitive with GIANT, even though the latter may perform better under certain conditions. We stress that the key advantage of SHED lies in its robustness under ill-conditioning and non i.i.d. data, and its effectiveness and versatility when CRs differ across nodes and links.
Future work includes the use and extension of the algorithm for more specific scenarios and applications, like, for example, wireless networks, and also the study of new heuristics for the renewal operation in the general convex case.

2.9 Related Publications and Conference Presentations

The content of this chapter is available in ArXiv [41] and has been accepted for publication in *Automatica*. Part of the content has also been accepted as an extended abstract to

the IFAC Conference on Networked Systems, 2022 (NecSys22), and presented in a poster session at the conference in Zurich.

3

Q-SHED: Distributed Optimization at the Edge via Hessian Eigenvectors Quantization

Edge networks call for communication efficient (low overhead) and robust distributed optimization (DO) algorithms. These are, in fact, desirable qualities for DO frameworks, such as federated edge learning techniques, in the presence of data and system heterogeneity, and in scenarios where inter-node communication is the main bottleneck. Although computationally demanding, Newton-type (NT) methods have been recently advocated as enablers of robust convergence rates in challenging DO problems where edge devices have sufficient computational power. Along these lines, in this chapter we propose Q-SHED, an original NT algorithm for DO featuring a novel bit-allocation scheme based on incremental Hessian eigenvectors quantization. The proposed technique is integrated with the SHED algorithm, that we presented in Chapter 2, from which it inherits appealing features like the small number of required Hessian computations, while being bandwidth-versatile at a bit-resolution level. Our empirical evaluation against competing approaches shows that Q-SHED can reduce by up to 60% the number of communication rounds required for ϵ-convergence.

3.1 Introduction

Solving distributed optimization problems in a communication-efficient fashion is one of the main challenges of next generation edge networks [131]. In particular, much attention is being turned to distributed machine learning (ML) settings and applications, and to the distributed training of ML models via *federated learning* (FL) [85]. FL is a distributed optimization (DO) framework motivated by the increasing concerns for data privacy at the user end, and by the convenience of performing distributed processing in multi-access edge computing (MEC) networks. However, DO is particularly

challenging in federated edge learning (FEL) scenarios where communication occurs over unpredictable and heterogeneous wireless links [75]. To tackle these challenges, major research efforts have been conducted in recent years [14], [30], [132]. A common assumption in FEL is that edge devices are equipped with sufficient computing capabilities. Hence, Newton-type (NT) methods, although computationally demanding, have been recently advocated to improve the convergence rate of distributed optimization, while significantly reducing its communication overhead [82], [139]. Communication efficient distributed NT (DNT) algorithms like GIANT [144], and DONE [139] have shown promising results in configurations with i.i.d. data distributions among devices, but underperform when applied to ill-conditioned problems and heterogeneous data configurations [42], which are scenarios of major practical relevance. Some works, like FedNL [125] and SHED (sharing Hessian eigenvectors for distributed learning) [42] have been recently proposed to robustify FL in the presence of non i.i.d. data distributions, system heterogeneity and ill-conditioning. A DNT method with over-the-air aggregation has been studied in [82]. Quantized Newton (QN) [4] has investigated the convergence properties of the distributed Newton method when the Hessian matrix is quantized. However, QN entails a communication load proportional to $O(n^2)$, where n is the problem dimensionality, while a linear per-iteration communication complexity of $O(n)$ is desirable.

In this chapter, we present Q-SHED, a new algorithm that extends the recently proposed SHED [42] via a novel bit-allocation scheme based on incremental Hessian eigenvector quantization. In particular, our main contributions are:

- We propose an original bit-allocation scheme for Hessian approximation based on uniform scalar dithered quantization of Hessian eigenvectors, to improve the efficiency of second-order information transmission in a DNT method.

- We integrate our bit-allocation scheme with the recently proposed SHED technique [42], obtaining a new approach, Q-SHED, based on incremental dithered quantization of Hessian eigenvectors. Q-SHED has a communication complexity of $O(n)$ (inherited by SHED) and handles per-iteration heterogeneity of communication channels of the different edge computers involved in the optimization problem at a bit-resolution (per vector coordinate) level.

- We evaluate Q-SHED on two datasets assessing its performance in a standard distributed optimization setup, as well as in a scenario where the transmission quality of communication links randomly fluctuates over time according to a Rayleigh fading model (a popular model for wireless channels). With respect to competing solutions, Q-SHED shows convergence speed improvements of at least

30% in a non-fading scenario and of up to 60% in the Rayleigh fading case.

3.2 Distributed optimization framework

We consider the typical DO framework where M machines communicate with an aggregator to cooperatively solve an empirical risk minimization problem of the form

$$\min_{\boldsymbol{\theta}} f(\boldsymbol{\theta}) := \frac{1}{N}\sum_{d=1}^{M} N_d f^{(d)}(\boldsymbol{\theta}), \tag{3.1}$$

where $\boldsymbol{\theta} \in \mathbb{R}^n$ is the optimization variable, N_d is the number of data samples of the d-th machine and $N = \sum_{d=1}^{M} N_d$. For the convergence analysis of the algorithm, we make the following standard assumption on the cost function f:

Assumption 3. Let $\mathbf{H}(\boldsymbol{\theta}) := \nabla^2 f(\boldsymbol{\theta})$ be the Hessian matrix of the cost $f(\boldsymbol{\theta})$. $f(\boldsymbol{\theta})$ is twice continuously differentiable, smooth, strongly convex and $\mathbf{H}(\boldsymbol{\theta})$ is Lipschitz continuous.

3.2.1 Distributed Newton method

The Newton method to solve (3.1) is:

$$\boldsymbol{\theta}^{t+1} = \boldsymbol{\theta}^t - \eta_t \mathbf{H}_t^{-1}\mathbf{g}_t,$$

where t denotes the t-th iteration, $\mathbf{g}_t = \mathbf{g}(\boldsymbol{\theta}_t) = \nabla f(\boldsymbol{\theta}^t)$, η_t and $\mathbf{H}_t = \nabla^2 f(\boldsymbol{\theta}^t)$ denote the gradient, the step size and the Hessian matrix at iteration t, respectively. In the considered DO scenario, we have that:

$$\mathbf{H}_t = \frac{1}{N}\sum_{d=1}^{M} N_d \mathbf{H}_t^{(d)}, \quad \mathbf{g}_t = \frac{1}{N}\sum_{d=1}^{M} N_d \mathbf{g}_t^{(d)}, \tag{3.2}$$

where $\mathbf{H}_t^{(d)} = \nabla^2 f^{(d)}(\boldsymbol{\theta}^t)$ and $\mathbf{g}_t^{(d)} = \nabla f^{(d)}(\boldsymbol{\theta}^t)$ denote local Hessian and gradient of the local cost $f^{(d)}(\boldsymbol{\theta}^t)$ of machine d, respectively. To get a Newton update at the aggregator, in a FL setting one would need each agent to transfer the matrix $\mathbf{H}_t^{(i)}$ of size $O(n^2)$ to the aggregator at each iteration, whose communication cost is considered prohibitive in many practical scenarios, especially when n is large. To deal with communication constraints, while still exploiting second-order information, DNT methods use Hessian approximations:

$$\boldsymbol{\theta}^{t+1} = \boldsymbol{\theta}^t - \eta_t \hat{\mathbf{H}}_t^{-1}\mathbf{g}_t, \tag{3.3}$$

where $\hat{\mathbf{H}}_t$ is an approximation of $\mathbf{H}_t$.

3.2.2 The SHED algorithm

In this chapter, we propose a DNT approach built upon SHED [42], a DNT algorithm for FL designed to require few Hessian computations by FL workers, that efficiently shares (low communication overhead) second-order information with the aggregator, see [42] for a detailed description. SHED exploits a full-rank approximation of the workers' Hessians by sending to the aggregator the most relevant eigenvalue-eigenvector pairs (EEPs) of the local Hessian, along with a local approximation parameter. Approximations are incrementally improved across iterations, as machines send additional EEPs to the aggregator. By doing so, the Hessian is computed only sporadically and outdated versions of it are used to incrementally improve the convergence rate. Under Lipschitz Hessians, strong convexity and smoothness assumptions, SHED has super-linear convergence.

3.2.3 Q-SHED: Hessian eigenvectors quantization

Let $\mathbf{H} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^\top$, be the eigendecomposition of a machine (edge computer) Hessian matrix, with $\mathbf{\Lambda} = \text{diag}(\lambda_1, ..., \lambda_n)$, where λ_k is the eigenvalue corresponding to the k-th unitary eigenvector, $\mathbf{v}_k$. In general, the Hessian is a function of the parameter $\boldsymbol{\theta}$, but here we omit this dependence for ease of notation. We always consider eigenvalues ordered so that $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_n$. In SHED, a machine shares with the aggregator a parameter ρ_q together with q EEPs, allowing for a full-rank (q, ρ_q)-approximation of its Hessian, of the form

$$\mathbf{H}_{q,\rho_q} = \sum_{i=1}^{q} (\lambda_i - \rho_q)\mathbf{v}_i\mathbf{v}_i^\top + \rho_q\mathbf{I} = \mathbf{V}\mathbf{\Lambda}_{\rho_q}\mathbf{V}^\top, \tag{3.4}$$

where $\mathbf{V} := [\mathbf{v}_1, ..., \mathbf{v}_n]$, $\mathbf{\Lambda}_{\rho_q} := \text{diag}(\lambda_1, ..., \lambda_q, \rho_q, ..., \rho_q) \in \mathbb{R}^{n\times n}$. In the original SHED algorithm, eigenvectors are transmitted exactly (up to machine-precision). Differently, we here design a quantization scheme for the eigenvectors, obtaining a quantized approximation of the Hessian of the form:

$$\hat{\mathbf{H}}_{q,\rho_q}(b_1, ..., b_q) = \sum_{i=1}^{q} (\lambda_i - \rho_q)\hat{\mathbf{v}}_i(b_i)\hat{\mathbf{v}}_i(b_i)^\top + \rho_q\mathbf{I}, \tag{3.5}$$

where we denote by $\hat{\mathbf{v}}_i(b_i)$ the i-th eigenvector, quantized with b_i bits per vector element. As in [42], we fix $\rho_q = \lambda_{q+1}$. We design the quantization scheme so that if an eigenvector $\mathbf{v}_i$ is quantized and transmitted, then at least one bit is assigned to each of its components. The vectors to which no bit is assigned are all set equal to zero, i.e., $\hat{\mathbf{v}}_i(0) = \mathbf{0}$. We assume that, as in typical machine learning problems, $n \gg 1$. Hence, we design the quantization scheme such that the approximation parameter ρ_q and the eigenvalues $\{\lambda_i\}$ are not quantized and are transmitted exactly (up to machine precision).

3.3 Optimal quantization of eigenvectors

We formulate the design of the quantization scheme as a bit allocation problem, exploiting the specific structure of the Hessian. In particular, as, e.g., in [132], we consider dithered quantization, so that we can model the quantization error as a uniformly distributed (in the lattice) zero mean additive random noise. Let $\mathbf{v}_i$ be a Hessian eigenvector and let $\hat{\mathbf{v}}_i = \hat{\mathbf{v}}_i(b_i)$ be the same eigenvector quantized with b_i bits per vector coordinate (to improve readability, the dependence on b_i is omitted in the following). We write:

$$\hat{\mathbf{v}}_i = \mathbf{v}_i + \boldsymbol{\epsilon}_i, \tag{3.6}$$

where $\boldsymbol{\epsilon}_i$ is a uniformly distributed quantization noise, with $\mathbb{E}[\boldsymbol{\epsilon}_i] = 0$. This is a general and standard model for the quantization noise, widely adopted in the literature, see, e.g., [132].

The aim of the bit allocation is to provide the best possible Hessian approximation given a bit budget. Hence, the quantization scheme design is obtained as the solution of the following problem:

$$\begin{aligned} \min_{b_1,\dots,b_q,q} \quad & \mathbb{E}[\|\mathbf{H} - \hat{\mathbf{H}}_{q,\rho_q}(b_1,\dots,b_q)\|_{\mathcal{F}}^2 | \{\mathbf{v}_i, \lambda_i\}_{i=1}^q] \\ \text{s.t.} \quad & \sum_{i=1}^{q} b_i = B \\ & 0 \leq \ b_i \leq b_{\max}, \forall\, i, \end{aligned} \tag{3.7}$$

where $\hat{\mathbf{H}}_{q,\rho_q}(b_1,\dots,b_q)$ is defined in (3.5). The operator $\|\cdot\|_{\mathcal{F}}$ denotes the Frobenius norm. Note that q is a variable determining the approximation parameter ρ_q. The constant B denotes the bit budget, normalized by n: denoting the total number of available bits by B_{tot}, it holds $B = \lfloor B_{\text{tot}}/n \rfloor$. The integer $b_{\max}$ is the maximum number of bits per vector component. In the following, for ease of notation, we omit the conditioned values from the expectation expression of the squared Frobenius norm introduced in (3.7). For simplicity, we define $\hat{\mathbf{H}}_{q,\rho_q} := \hat{\mathbf{H}}_{q,\rho_q}(b_1,\dots,b_q)$. Denoting by $\text{tr}(\cdot)$ the trace operator, we have that

$$\mathbb{E}[\|\mathbf{H} - \hat{\mathbf{H}}_{q,\rho_q}\|_{\mathcal{F}}^2] = \mathbb{E}[\text{tr}((\mathbf{H} - \hat{\mathbf{H}}_{q,\rho_q})(\mathbf{H} - \hat{\mathbf{H}}_{q,\rho_q}))], \tag{3.8}$$

where we can write, denoting the unitary eigenvector matrix by $\mathbf{V} := [\mathbf{v}_1,\dots,\mathbf{v}_n]$,

$$\mathbf{H} - \hat{\mathbf{H}}_{q,\rho_q} = \mathbf{V}(\boldsymbol{\Lambda} - \boldsymbol{\Lambda}_{\rho_q})\mathbf{V}^\top + \sum_{i=1}^{q}(\lambda_i - \rho_q)\delta\mathbf{V}_i. \tag{3.9}$$

defining $\delta\mathbf{V}_i := (\mathbf{v}_i\mathbf{v}_i^\top - \hat{\mathbf{v}}_i\hat{\mathbf{v}}_i^\top)$. Plugging (3.9) into (3.8):

$$\begin{aligned}\mathbb{E}[\|\mathbf{H} - \hat{\mathbf{H}}_{q,\rho_q}\|_{\mathcal{F}}^2] &= \operatorname{tr}(\mathbf{V}(\mathbf{\Lambda} - \mathbf{\Lambda}_{\rho_q})^2\mathbf{V}^\top) \\ &+ 2\operatorname{tr}\left(\sum_{i=q+1}^{n}(\bar{\lambda}_{q,i})\mathbf{v}_i\mathbf{v}_i^\top\sum_{i=1}^{q}(\bar{\lambda}_{q,i})\mathbb{E}[\delta\mathbf{V}_i]\right) \\ &+ \mathbb{E}\left[\operatorname{tr}\left(\sum_{i=1}^{q}\bar{\lambda}_{q,i}^2\delta\mathbf{V}_i\delta\mathbf{V}_i\right) + \operatorname{tr}\left(\sum_{\substack{i,j=1\\ i\neq j}}^{q}\bar{\lambda}_{q,i}\bar{\lambda}_{q,j}\delta\mathbf{V}_i\delta\mathbf{V}_j\right)\right].\end{aligned} \tag{3.10}$$

where $\bar{\lambda}_{q,i} := \lambda_i - \rho_q$. The first term of the previous expression does not depend on the quantization strategy, but only on the choice of q. The second and third terms, instead, both depend on q and on the quantization strategy through the matrices $\{\delta\mathbf{V}_i\}_{i=1}^q$.

3.3.1 Scalar Uniform Quantization

In the next section, we consider the special case of scalar uniform quantization of the eigenvectors' coordinates. In the case of scalar uniform quantization, each component of vector $\mathbf{v}_i$ is uniformly quantized in the range $[-1, 1]$. Applying dithering, the quantization error vector has i.i.d. uniformly distributed components of known covariance [132]. We can write

$$\mathbb{E}[\boldsymbol{\epsilon}_i\boldsymbol{\epsilon}_i^\top] = \sigma_i^2 I, \text{ with } \sigma_i^2 = \mathbb{E}[\boldsymbol{\epsilon}_{ij}^2] = \Delta_i^2/12, \ \Delta_i = 2^{-(b_i-1)} \tag{3.11}$$

with Δ_i being the quantization interval length, and b_i the number of bits assigned to each coordinate of the i-th eigenvector. After some algebra, we can get

$$\mathbb{E}[\operatorname{tr}(\delta\mathbf{V}_i)(\delta\mathbf{V}_i)] = \Delta_i^2(a_1(n) + a_2(n)\Delta_i^2), \tag{3.12}$$

using the fact that $\alpha_i^4 = \Delta_i^4/80 = \mathbb{E}[\epsilon_{ij}^4]$, and defining $a_1(n) := \frac{1}{12} + \frac{n}{6}$, $a_2(n) := \frac{n}{80} + \frac{n(n-1)}{12^2}$. With similar calculations, one gets

$$\mathbb{E}[\operatorname{tr}(\delta\mathbf{V}_i\delta\mathbf{V}_j)] = n\sigma_i^2\sigma_j^2 = \frac{n\Delta_i^2\Delta_j^2}{12^2} = a_3(n)\Delta_i^2\Delta_j^2, \tag{3.13}$$

with $a_3(n) := \frac{n}{12^2}$. The expectation of the Frobenius norm of the quantization error in (3.10) can then be written as

$$\begin{aligned}\mathbb{E}[\|\mathbf{H} - \hat{\mathbf{H}}_{q,\rho_q}\|_{\mathcal{F}}^2] = \sum_{i=q+1}^{n} \bar{\lambda}_{q,i}^2 + d_q \sum_{i=1}^{q} \bar{\lambda}_{q,i}\Delta_i^2 \\ + \sum_{i=1}^{q} \bar{\lambda}_{q,i}^2 \Delta_i^2 (a_1(n) + a_2(n)\Delta_i^2) + \sum_{\substack{i,j=1 \\ i\neq j}}^{q} \bar{\lambda}_{q,i}\bar{\lambda}_{q,j} a_3(n)\Delta_i^2\Delta_j^2,\end{aligned} \tag{3.14}$$

with $d_q = \frac{1}{6}(\sum_{i=q+1}^{n}(\rho_q - \lambda_i))$. Our objective is to pick the integer parameter q and the quantization intervals $\Delta_1, ..., \Delta_q$ so as to minimize (3.8), with the constraint that $\sum_{i=1}^{q} b_i = B$, with $B = \lfloor B_{\text{tot}}/n \rfloor$, where B_{tot} is the number of available bits. Given that $b_i = -\log \Delta_i + 1$, we see that the constraint becomes $\sum_{i=1}^{q} \log \Delta_i = q - B$, which is equivalent to $\sum_{i=1}^{q} \log \Delta_i^2 = 2(q - B)$. Defining $x_i := \Delta_i^2$ and $\mathbf{x}_q = (x_1, ..., x_q)$, we define the expectation of the quantization error as a cost function ℓ:

$$\ell(\mathbf{x}_q, q) := \mathbb{E}[\|\mathbf{H} - \hat{\mathbf{H}}_{q,\rho_q}\|_{\mathcal{F}}^2], \tag{3.15}$$

and we aim to minimize such cost function over the choice of q and over the choice of $\mathbf{x}_q$. We can rewrite (3.14) as

$$\begin{aligned}\ell(\mathbf{x}_q, q) = \sum_{i=q+1}^{n} \bar{\lambda}_{q,i}^2 + \sum_{i=1}^{q} \gamma_{n,q,i} x_i + a_2(n) \sum_{i=1}^{q} \bar{\lambda}_{q,i}^2 x_i^2 \\ + a_3(n) \sum_{\substack{i,j=1 \\ i\neq j}}^{q} \bar{\lambda}_{q,i}\bar{\lambda}_{q,j} x_i x_j,\end{aligned}$$

where $\gamma_{n,q,i} := d_q \bar{\lambda}_{q,i} + a_1(n)\bar{\lambda}_{q,i}^2$. The optimization problem is thus turned into the following equivalent form:

$$\begin{aligned}\min_{\mathbf{x}_q, q} \quad & \ell(\mathbf{x}_q, q) \\ \text{s.t.} \quad & -\sum_{i=1}^{q} \log x_i \leq 2(B - q) \\ & 0 < \ x_i \leq 4, \ i = 1, ..., q\end{aligned} \tag{3.16}$$

where the last constraint ($x_i \leq 4$) amounts to requiring $b_i \geq 0, i = 1, ..., q$. At optimality, the constraint $-\sum_{i=1}^{q} \log x_i \leq 2(B - q)$ will be satisfied with equality. The solution to the optimization problem (3.16) needs to be converted in a vector of bits. This can be

done by converting each x_i back to b_i using (3.11) and then rounding each b_i to the closest integer, being careful to meet the bit budget $\sum_{i=1}^{q} b_i = B$.

Lemma 3.1. For any $q = 1, ..., n$, the cost function $\ell(\mathbf{x}_q, q)$ is strictly convex in $\mathbf{x}_q = (x_1, \dots, x_q)^\top$.

Proof. Let $\bar{\boldsymbol{\lambda}}_q := (\bar{\lambda}_1, ..., \bar{\lambda}_q)^\top$, $\boldsymbol{\gamma}_{n,q} := (\gamma_{n,q,1}, ..., \gamma_{n,q,q})^\top$, $\bar{\boldsymbol{\Lambda}}_q := \text{diag}(\bar{\lambda}_1^2, ..., \bar{\lambda}_q^2)$, and $\bar{\boldsymbol{\Lambda}}_c \in \mathbb{R}^{q \times q}$ a matrix such that $(\bar{\boldsymbol{\Lambda}}_c)_{i,j} = \bar{\lambda}_i \bar{\lambda}_j (1 - \delta_{ij})$, where $\delta_{ii} = 1$ and $\delta_{ij} = 0$ for $i \neq j$. Note that $a_2(n) = \frac{n}{80} + \frac{n(n-1)}{12^2} > \frac{n}{12^2} = a_3(n)$. Omitting terms that do not depend on $\mathbf{x}_q$, the cost can be rewritten as

$$\begin{aligned} \ell(\mathbf{x}_q, q) &= \boldsymbol{\gamma}_{n,q}^\top \mathbf{x}_q + a_2(n) \mathbf{x}_q^\top \bar{\boldsymbol{\Lambda}}_q \mathbf{x}_q + a_3(n) \mathbf{x}_q^\top \bar{\boldsymbol{\Lambda}}_c \mathbf{x}_q \\ &= \boldsymbol{\gamma}_{n,q}^\top \mathbf{x}_q + \mathbf{x}_q^\top (a_3(n) \bar{\boldsymbol{\Lambda}}_q + (a_2(n) - a_3(n)) \bar{\boldsymbol{\lambda}}_q \bar{\boldsymbol{\lambda}}_q^\top) \mathbf{x}_q \\ &= \boldsymbol{\gamma}_{n,q}^\top \mathbf{x}_q + \mathbf{x}_q^\top \mathbf{A}_q \mathbf{x}_q, \end{aligned} \tag{3.17}$$

and because of the fact that $a_2(n) > a_3(n)$, we have

$$\mathbf{A}_q = a_3(n) \bar{\boldsymbol{\Lambda}}_q + (a_2(n) - a_3(n)) \bar{\boldsymbol{\lambda}}_q \bar{\boldsymbol{\lambda}}_q^\top > 0. \tag{3.18}$$

□

Given the convexity of the constraints, and the strict convexity of the objective function $\ell(\mathbf{x}_q)$ for any $q = 1, ..., n$ the optimization problem can be solved by solving n convex problems whose solution $\mathbf{x}_q^*$ is unique. The optimal solution can be found as the tuple $\{\mathbf{x}_{q^*}^*, q^*\}$, with $q^* = \text{argmin}_q \{\ell(\mathbf{x}_q^*, q)\}$.

3.4 Q-SHED: algorithm design

SHED [42] is designed to make use of Hessian approximations obtained with few Hessian EEPs. In [42], it has been shown that incrementally (per iteration) transmitting additional EEPs improves the converges rate. In this section, we augment SHED with the optimal bit allocation of the previous section, making it suitable to incrementally refine the Hessian approximation at the aggregator. The full technique is illustrated in Algorithm 5, and the details are provided in the following sections.

3.4.1 Uniform scalar quantization with incremental refinements

Let $\mathbf{H}(\boldsymbol{\theta}^{k_t})$ be the Hessian computed for parameter $\boldsymbol{\theta}^{k_t}$ at round k_t. At each round $t \geq k_t$, a number of bits B_t is sent to represent second-order information. At each round, we use newly available bits to incrementally refine the approximation of $\mathbf{H}(\boldsymbol{\theta}^{k_t})$. From now on,

eigenvectors are always denoted by $\mathbf{v}_i = \mathbf{v}_i(\boldsymbol{\theta}^{k_t})$, i.e., they are always the eigenvectors of the most recently computed (and possibly outdated) Hessian. If $t = k_t$, the optimal bit allocation for eigenvectors $\mathbf{v}_1, ..., \mathbf{v}_n$ is provided by the scheme presented in Sec. 3.3.1. Fix $t > k_t$. Let $b_{t-1}(i)$ denote the bits allocated to each coordinate of eigenvector $\mathbf{v}_i$ up to round $t-1$, and let $b_{i,t}$ be the number of bits to be used together with $b_{t-1}(i)$, at round t, to refine the approximation of the coordinates of $\mathbf{v}_i$. We can write

$$b_t(i) := b_{t-1}(i) + b_{i,t}, \quad \Delta_{t,i} := \frac{2}{2^{b_t(i)}} := 2^{-b_{t-1}(i)} 2^{-b_{i,t}+1} \tag{3.19}$$

with $b_t(i)$ the number of bits sent up to round t. The interval $\Delta_{t,i}$ is the quantization interval resulting from adding $b_{i,t}$ bits for the refinement of the i-th eigenvector information, for which $b_i^{(t-1)}$ had been previously allocated. We can plug these intervals into (3.14), and defining $x_{t,i} := 2^{-2(b_{i,t}-1)}$, $\tilde{\gamma}_{n,q_t,i} := 2^{-2b_{t-1}(i)}\gamma_{n,q_t,i}$, $\tilde{\lambda}_{t,q_t,i} := 2^{-2b_{t-1}(i)}\bar{\lambda}_{q_t,i}$, we get a cost $\ell(\mathbf{x}_{q_t}, q_t)$, with $\mathbf{x}_{q_t} = (x_{t,1}, ..., x_{t,q_t})$,

$$\begin{aligned} \ell(\mathbf{x}_{q_t}, q_t) = &\sum_{i=q_t+1}^{n} \bar{\lambda}_{q_t,i}^2 + \sum_{i=1}^{q_t} \tilde{\gamma}_{n,q_t,i} x_{t,i} \\ +b(n) &\sum_{i=1}^{q_t} \tilde{\lambda}_{t,q_t,i}^2 x_{t,i}^2 + c(n) \sum_{\substack{i,j=1 \\ i \neq j}}^{q_t} \tilde{\lambda}_{t,q_t,i} \tilde{\lambda}_{t,q_t,j} x_{t,i} x_{t,j}. \end{aligned} \tag{3.20}$$

Following the same proof technique as for Lemma 3.1, it can be shown that the cost $\ell(\mathbf{x}_{q_t}, q_t)$ is strictly convex in $\mathbf{x}_{q_t}$ for any $q_t = 1, ..., n$. Given that up to round $t-1$, q_{t-1} eigenvectors were considered for bit allocation, it is easy to see that it needs to be $q_t \geq q_{t-1}$. Similarly to Sec. 3.3.1, we formulate the optimal bit allocation of bits $\{b_{i,t}\}_{i=1}^{q_t}$ as

$$\begin{aligned} \min_{\mathbf{x}_{q_t}, q_t \geq q_{t-1}} \quad & \ell(\mathbf{x}_{q_t}, q_t) \\ \text{s.t.} \quad & -\sum_{i=1}^{q_t} \log x_{t,i} \leq 2(B_t - q_t) \\ & 0 < \ x_{t,i} \leq 4, \ i = 1, ..., q_t. \end{aligned} \tag{3.21}$$

The problem can be solved by finding the unique solution to the $n - q_{t-1} + 1$ strictly convex problems corresponding to the different choices of $q_t = q_{t-1}, q_{t-1} + 1, ..., n$. As before, the solution to problem (3.21) needs to be converted to integer numbers, for example by rounding the corresponding allocated number of bits to the closest integer, being careful to retain $\sum_{i=1}^{q} b_{i,t} = B_t$. Sorting the eigenvalues in a decreasing order, we get a monotonically decreasing sequence of allocated bits to the corresponding eigenvectors. To provide an example, with the FMNIST dataset (see Sec.

2.6), at a certain iteration t of the incremental algorithm, an agent allocates bits $b_t = [3, 3, 2, 2, 2, 1, 1, 1, 1, 1, 1]$ to the first 11 eigenvectors, whose corresponding (rounded) eigenvalues are $[0.21, 0.11, 0.06, 0.03, 0.03, 0.02, 0.02, 0.01, 0.01, 0.01, 0.01]$.

3.4.2 Multi-agent setting: notation and definitions

To illustrate the integration of our incremental quantization scheme with SHED, we introduce some definitions for the multi-agent setting. We denote by $B_t^{(d)}$ the bit-budget of device d at iteration t. Let $\rho_t^{(d)} = \lambda^{(d)}_{q_t^{(d)}+1,t}$ be the Hessian approximation parameter of device d at iteration t, function of the $q_t^{(d)}$-th eigenvalue of the d-th device, where the integer $q_t^{(d)}$ is tuned by device d as part of the bit-allocation scheme at iteration t. Let $\mathbf{g}_t^{(d)}$, $\mathbf{H}_t^{(d)}$, $\hat{\mathbf{H}}_t^{(d)}$ be the gradient, Hessian, and Hessian approximation, respectively, of device d. We denote by $\mathbf{v}_i^{(d)}$ and $\hat{\mathbf{v}}_i^{(d)}$ the i-th eigenvector of the d-th device and its quantized version, respectively. Note that eigenvectors always correspond to the last computed Hessian $\mathbf{H}(\boldsymbol{\theta}^{k_t})$, with $k_t \leq t$. The integer $b_t^{(d)}(q)$ denotes the number of bits allocated by device d to the q-th eigenvector coordinates up to iteration t, while $b_{q,t}^{(d)}$ is the per-iteration bits allocated to the q-th eigenvector, i.e., $b_t^{(d)}(q) = b_{t-1}^{(d)}(q) + b_{q,t}^{(d)}$. We define $\mathcal{A}$ to be the set of devices involved in the optimization, $\mathcal{I}$ the set of iteration indices in which each device recomputes its local Hessian, $f^{(d)}$ the cost function of device d, and $\epsilon > 0$ the gradient norm threshold. Hessian approximations are built at the aggregator in the following way:

$$\hat{\mathbf{H}}_t = \frac{1}{N}\sum_{d=1}^{M} N_d \hat{\mathbf{H}}_t^{(d)}, \quad \hat{\mathbf{H}}_t^{(d)} = \sum_{i=1}^{q_t^{(d)}} \bar{\lambda}_i^{(d)} \hat{\mathbf{v}}_i^{(d)} \hat{\mathbf{v}}_i^{(d)\top} + \rho_t^{(d)} \mathbf{I}, \tag{3.22}$$

where $\bar{\lambda}_i^{(d)} = \lambda_i^{(d)} - \rho_t^{(d)}$. Incremental quantization allows devices to refine the previously transmitted quantized version of their eigenvectors by adding information bits, see (3.19). We denote the set of information bits of device d sent to quantize or refine previously sent quantized eigenvectors by $Q_t^{(d)}$.

3.4.3 Heuristic choice of $q_t^{(d)}$

To reduce the computational burden at the edge devices and to solve the bit-allocation problem only once per round, we propose a heuristic strategy for each device to choose $q_t^{(d)}$: at each incremental round t, instead of inspecting all the options corresponding to $q_{t-1}^{(d)}, ..., q_n^{(d)}$, which would provide the exact solution, but would require solving problem (3.21) $n - q_{t-1}^{(d)} + 1$ times. We fix $\bar{q} = q_{t-1}^{(d)} + B_t^{(d)}$: With this choice of $\bar{q}$, we solve problem (3.21), and we subsequently convert the solution to bits obtaining $\{b_{i,t}^{(d)}\}_{i=1}^{\bar{q}}$ and

$\{b_t^{(d)}(i)\}_{i=1}^{\bar{q}}$. We then fix the value

$$q_t^{(d)} = \hat{q}_t^{(d)}(\{b_t^{(d)}(i)\}_{i=1}^{\bar{q}}) := \max_q \{q : b_t^{(d)}(q) > 0\} \tag{3.23}$$

3.4.4 Convergence analysis

The choice of Hessian approximation is positive definite by design (see (3.22)). Hence, the algorithm always provides a descent direction and, with a backtracking strategy like in [144] and [42], convergence is guaranteed (see Theorem 4 of [42]). Empirical results suggest that linear and superlinear convergence of the original SHED may still be guaranteed under some careful quantization design choices. We leave the analysis of the convergence rate as future work, but we provide an intuition on the convergence rate in the least squares case. In the least squares case, for a given choice of q and of the allocated bits $\{b_{i,t}^{(d)}\}_{i=1}^{q}$ of each device d, an easy extension of Theorem 3 in [42] provides the following bound

$$\|\boldsymbol{\theta}^{t+1} - \boldsymbol{\theta}^*\| \leq \kappa_t \|\boldsymbol{\theta}^t - \boldsymbol{\theta}^*\|, \tag{3.24}$$

with $\kappa_t = (1 - (\bar{\lambda}_n - e_t)/\bar{\rho}_t)$, where

$$\bar{\lambda}_n = \frac{1}{N}\sum_{d=1}^{M} N_d \lambda_n^{(d)} \quad \bar{\rho}_t = \frac{1}{N}\sum_{d=1}^{M} N_d \rho_t^{(d)}$$

and

$$e_t = \frac{1}{N}\sum_{d=1}^{M} N_d \sum_{i=1}^{q_t^{(d)}} (\lambda_i^{(d)} - \rho_t^{(d)}) \|\delta \mathbf{V}_i^{(d)}\| \tag{3.25}$$

where $\delta \mathbf{V}_i^{(d)} := (\mathbf{v}_i^{(d)} \mathbf{v}_i^{(d)\top} - \hat{\mathbf{v}}_i^{(d)} \hat{\mathbf{v}}_i^{(d)\top})$. It can be noted how for a sufficiently small quantization error, which can always be achieved by incremental refinements, the convergence rate in the least squares case is at least linear. The extension to the general case is left as a future work.

Algorithm 5 Q-SHED

Input: $\{f^{(d)}\}_{d=1}^{M}$, $\mathcal{I}$, $\boldsymbol{\theta}^1$, $\nabla f(\boldsymbol{\theta}^1)$, $\mathcal{A}$, $\epsilon > 0$
Output: $\boldsymbol{\theta}^t$

$t \leftarrow 1$
while $\|\nabla f(\boldsymbol{\theta}^t)\|_2 \geq \epsilon$ **do**
  for *device* $d \in \mathcal{A}$ **do**
    when received $\boldsymbol{\theta}^t$ from the aggregator **do**
      if $t \in \mathcal{I}$ **then**
        $k_t \leftarrow t$
        compute $\mathbf{H}_t^{(d)} = \nabla^2 f^{(d)}(\boldsymbol{\theta}^t)$ // renewal
        $\{(\lambda_{j,t}^{(d)}, \mathbf{v}_j^{(d)})\}_{j=1}^{n} \leftarrow \text{eigendecomp}(\mathbf{H}_t^{(i)})$
        $q_{t-1}^{(d)} \leftarrow 0$
      end if
      compute $\mathbf{g}_t^{(d)} = \mathbf{g}^{(d)}(\boldsymbol{\theta}^t) = \nabla f^{(d)}(\boldsymbol{\theta}^t)$
      $\bar{q} \leftarrow q_{t-1}^{(d)} + B_t^{(d)}$
      $\mathbf{x}_{\bar{q}}^* \leftarrow$ solve (3.21) for $q_t = \bar{q}$ with budget $B_t^{(d)}$
      $\{b_{i,t}^{(d)}\}_{i=1}^{\bar{q}} \leftarrow \text{convertToBits}(\mathbf{x}_{\bar{q}}^*)$ // back to bits
      $q_t^{(d)} \leftarrow \hat{q}_t^{(d)}(\{b_t^{(d)}(i)\}_{i=1}^{\bar{q}})$ // see (3.23)
      $\rho_t^{(d)} \leftarrow \lambda_{q_t^{(d)}+1,t}^{(d)}$
      $Q_t^{(d)} \leftarrow \text{quantize}(\{\mathbf{v}_i^{(d)}\}_{i=1}^{q_t^{(d)}}, \{b_{i,t}^{(d)}\}_{i=1}^{q_t^{(d)}}, \{b_t^{(d)}(i)\}_{i=1}^{q_t^{(d)}})$ // quantize or refine quantization, see (3.19)
      $U_t^{(d)} \leftarrow \{Q_t^{(d)}, \{\lambda_{j,t}^{(d)}\}_{j=q_{t-1}^{(d)}+1}^{q_t^{(d)}}, \mathbf{g}_t^{(d)}, \rho_t^{(d)}\}$
      send $U_t^{(d)}$ to the aggregator
  end for
  ---
  at the aggregator:
  when received $U_t^{(d)}$ from all devices **do**
    compute $\hat{\mathbf{H}}_t^{(d)}$, $\forall d$ // see (3.22)
    compute $\hat{\mathbf{H}}_t$ (see (3.22)) and $\mathbf{g}_t$
    get η_t via distributed backtracking line search.
    perform Newton-type update (3.3)
    broadcast $\boldsymbol{\theta}^{t+1}$ to all devices.
  $t \leftarrow t + 1$
end while

3.5 Empirical Results

In this section, we provide empirical results obtained with two datasets, FMNIST [146] and w8a [26]. We simulate two configurations for the network: one where every device has the same transmission rate at each communication round, and one where the rate changes randomly for each device based on the widely adopted Rayleigh fading model [114], [142]. For both FMNIST and w8a we build up a binary classification setting with logistic regression (in FMNIST we learn to distinguish class '1' from all the others), simulating a scenario with $M = 8$ devices, each with 500 data samples. We use L2 regularization with parameter $\mu = 10^{-5}$. For FMNIST, we apply PCA [129] to the data to reduce the dimensionality to $n = 90$, while for w8a we keep the original data dimensionality, $n = 300$. To simulate the fading channels, we adopt the following simple model. We consider that all the devices allocate the same bandwidth β for the communication with the aggregator and write the achievable transmission rate as (see, e.g., [114], [142])

$$R^{(d)} = \beta \log_2(1 + \gamma \Gamma^{(d)}) \tag{3.26}$$

where $\Gamma^{(d)}$ is a value related to transmission power and environmental attenuation for user d. For simplicity, we fix $\Gamma^{(d)} = \Gamma = 1$ for all users (in [114], for instance, $\Gamma = 1$ and $\Gamma = 10$ were considered). The only source of variability is then $\gamma \sim \exp(\nu)$, modelling the Rayleigh fading effect. We fix $\nu = 1$. Specifically, to simulate the different bit budgets, we compute the individual bit budget of each device as $B_t^{(d)} = B \log_2(1 + \gamma \Gamma^{(d)})$, setting $B = 2b_{\max}$. We fix $b_{\max} = 16$. In the non-fading case, the bit budget for each device is constant and set to $B_t^{(d)} = 2b_{\max}$. We consider a scenario where the full-quality gradient is always transmitted to the aggregator by the devices. We compare Q-SHED against an ideal version of SHED, dubbed ideal-SHED, where the eigenvectors that are quantized by Q-SHED are transmitted at full quality. We also compare Q-SHED against a naively-quantized counterpart, NQ-SHED, for which all bits are allocated to the first eigenvectors, and the state-of-the-art FedNL [125] with rank-1 compressors. With the exception of ideal-SHED, the per-round bit budget of the considered algorithms is the same. We have experimented with the possibility of quantizing the second-order information of FedNL, but we observed a performance degradation. Hence, when the bit budget of a device is not enough for communicating the rank-1 compression of the Hessian drift at full quality, we only use the device's local gradient. We do the same for NQ-SHED. The results on FMNIST and w8a are shown in Figs. 3.1 and 3.2, respectively. In both cases, it is possible to appreciate the robustness of Q-SHED in terms of iterations required for convergence, while both NQ-SHED and FedNL performance is degraded in

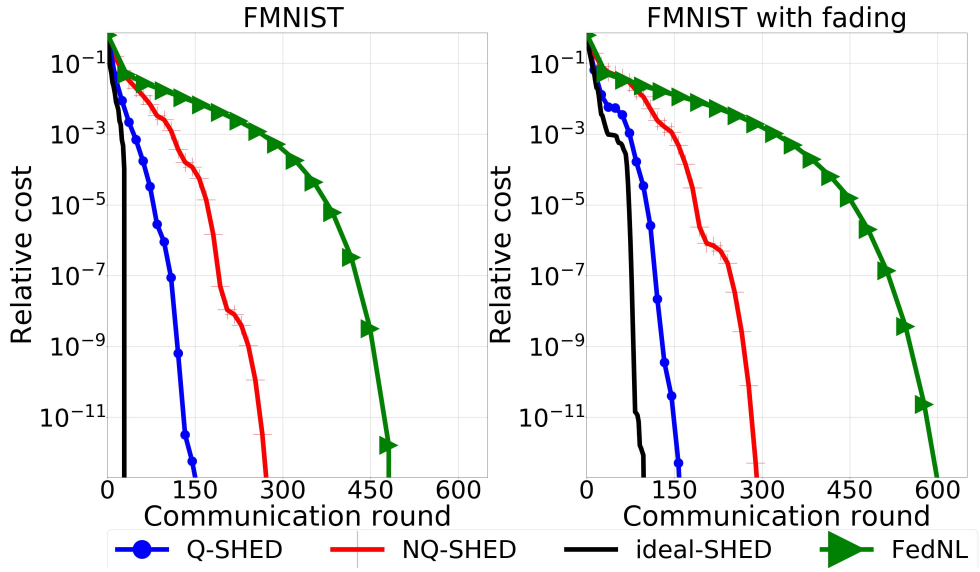

Figure 3.1: Comparison of Q-SHED against NQ-SHED and FedNL with the FMNIST dataset. With the exception of ideal-SHED, for a fair comparison, in each communication round the algorithms use the same number of bits. Relative cost is $f(\boldsymbol{\theta}^t) - f(\boldsymbol{\theta}^*)$.

the presence of fading channels. In terms of convergence speed, the results show that Q-SHED provides performance improvements against the selected competing solutions between 30% and 60%.

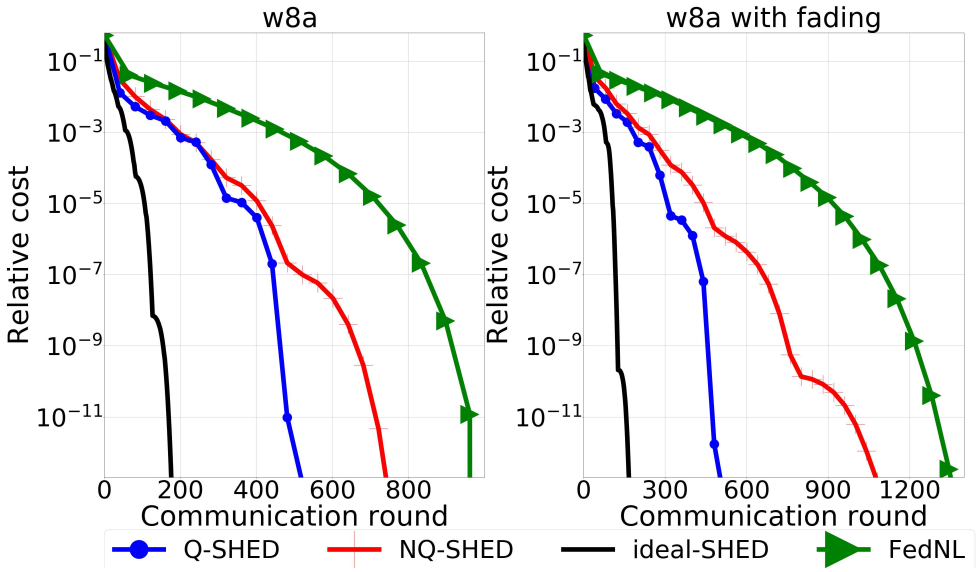

Figure 3.2: Comparison of Q-SHED against NQ-SHED and FedNL with the w8a dataset. With the exception of ideal-SHED, for a fair comparison, in each communication round the algorithms use the same number of bits. Relative cost is $f(\boldsymbol{\theta}^t) - f(\boldsymbol{\theta}^*)$.

3.6 Conclusion and future work

We have empirically shown that Q-SHED outperforms its naively-quantized version as well as state-of-the-art algorithms like FedNL. Future works include an in-depth analysis of the convergence rate, and the adoption of more advanced quantization schemes, like vector quantization techniques.

3.7 Related Publications and Conference Presentations

The content of this chapter is available in ArXiv [55] and has been accepted for presentation at the IEEE International Conference on Communications, 2023 (presented in Rome in May 2023).

4

Federated Reinforcement Learning under Communication Constraints: Finite-Time Rates

Federated learning (FL) has recently gained much attention due to its effectiveness in speeding up supervised learning tasks under communication and privacy constraints. However, whether similar speedups can be established for reinforcement learning remains much less understood theoretically. Towards this direction, we study a federated policy evaluation problem where agents communicate via a central aggregator to expedite the evaluation of a common policy. To capture typical communication constraints in FL, in this chapter we consider several communication models. In particular, we consider (i) finite capacity up-link channels that can drop packets based on a Bernoulli erasure model, (ii) over-the-air computation for bandwidth efficient wireless up-link transmission, and (iii) an asynchronous configuration in which up-link transmissions are subject to time-varying delays. We refer to these three schemes as `QFedTD`, `OACFedTD` and `AsyncFedTD`, respectively. In the following, we present and analyze these algorithms.

4.1 Introduction

Is it possible to obtain statistical models of high accuracy for supervised learning problems (e.g., regression, classification, etc.) by aggregating information from multiple devices while keeping the raw data on these devices private? This is the central question of interest in the popular machine learning paradigm of federated learning (FL) [22], [79], [92]. When the data-generating distributions of the participating devices are identical (or sufficiently similar), several works have shown that one can reap the benefits of collaboration by exchanging locally trained models via a central aggregator (server) [1], [38], [59], [66], [72]–[74], [99], [100], [136], [145]. In practice, these models are typically high-dimensional and need to be exchanged over unreliable communication links of limited

bandwidth. As such, a large body of work in FL has investigated the effects of the communication constraints on the convergence properties of optimization algorithms. Drawing inspiration from this literature, in this chapter, we ask: *Can we establish collaborative performance gains for federated reinforcement learning (FRL) problems subject to communication challenges?* As it turns out, little to nothing is known about this question from a theoretical standpoint.

Towards this direction, we study one of the most basic problems in RL, namely *policy evaluation*, in a federated setting. Specifically, in our problem, N agents, each of whom interacts with the same Markov Decision Process (MDP), communicate via a server to evaluate a fixed policy. While each agent can evaluate the policy on its own using Monte-Carlo sampling or temporal difference (TD) learning algorithms [138], [141], the reason for communicating is the same as in the standard FL setting: *to achieve an N-fold speedup in the sample-complexity of policy evaluation relative to when an agent acts alone.* In the recent survey paper on FRL [118], the authors mention that the goal of the FRL framework is to achieve such speedups while respecting privacy constraints, i.e., without revealing the raw data (states, actions, and rewards) of the agents. Relative to the FL setting, proving finite-time rates for FRL is significantly more challenging since we need to deal with temporally correlated Markovian samples. Indeed, even for the single-agent setting, finite-time rates under Markovian sampling have only recently been established [19], [35], [115], [135]. Works prior to these developments either provided a finite-time analysis under a restrictive i.i.d. sampling assumption [46], [83], or only came with asymptotic guarantees [23], [141]. For the multi-agent setting, almost all the prior works on TD learning make a restrictive i.i.d. sampling assumption [47], [88]. The only two exceptions to this are the very recent papers [77], [143] that establish linear speedups under Markovian sampling; however, none of the above works consider any communication constraints. As such, establishing linear speedups in FRL under Markovian sampling and communication constraints remains largely unexplored. In this regard, we consider three communication models which have many practical motivations and that have been widely investigated in the literature of distributed optimization and distributed machine learning, which we list here:

- `QFedTD`, in which agents upload quantized TD update directions over channels with finite capacity and subject to random packet drops (lossy links). These models [65], [121] have been extensively analyzed in the FL [65], [121], distributed optimization [49], [97], [119], [122], and networked control literature [67], [126] for almost two decades.

- `OACFedTD`, in which agents transmit their local TD update directions in up-link

as analog wireless signals, and the server lets the wireless channel perform the average in the setting of over-the-air computation, that has recently been advocated to provide large-scale, bandwidth- and energy-efficient up-link communication in FL [8], [81]. In particular, OAC exploits the waveform-superposition property of the wireless multiple access channel (MAC) to enable the receiver (server) to obtain the average of the analog signals transmitted by the agents over the same time-frequency block [25]. Compared to standard digital transmission, OAC comes with notable gains in up-link bandwidth efficiency. Furthermore, OAC has intrinsic privacy-preserving features [7], [127]. However, analog signals transmitted over the air are subject to fading channel distortion and additive noise at the receiver [127], [147], [154].

- `AyncFedTD` we consider an *asynchronous* framework in which multiple agents transmit their local TD update directions to a central server via up-link communication channels subject to asynchronous bounded delays. Asynchronous settings of this kind have been theoretically and empirically studied for FL and distributed optimization [51], [78], [108]. On the other hand, although asynchronous multi-agent RL (MARL) implementations have shown promising empirical performance, like in the case of parallel actor-learner frameworks [103], [106], little to nothing is known regarding their non-asymptotic convergence guarantees and multi-agent collaborative gains. Indeed, the only existing study providing finite-sample convergence guarantees for asynchronous MARL [130] establishes collaborative performance gains only under a simplifying i.i.d. sampling assumption on the agents' observations, i.e., considering observations that are not temporally correlated. However, even in the non-delayed single-agent case, the major technical hurdle in the finite-time analysis of RL algorithms (like TD learning) relative to optimization/supervised learning, comes precisely from the fact that the agent's observation sequence is generated by a Markov chain, and, as such, exhibits temporal correlations. For such settings, finite-time convergence bounds have only recently been provided in [19], [135] via some fairly involved analysis. Thus, for the MARL setting we consider with Markovian sampling and asynchronous delays, establishing collaborative performance benefits turns out to be highly non-trivial. Nonetheless, we provide such an analysis as a contribution of this chapter.

For each of the above schemes, our contribution is two-fold: we provide the first non-asymptoptic convergence analysis for the presented communication constrained multi-agent RL schemes while at the same time we establish a linear convergence speedup with the number of agents, i.e., we analytically show the beneficial effect of cooperation

even when agents' trajectories are temporally correlated (Markovian sampling).

We now comment on some of the highlights of our analysis relative to [77] and [143]. Our work crucially departs from both these papers in that, in addition to correlated Markovian samples, we add the constraints of communication. Unlike [143], our work does not require any projection step to ensure the boundedness of iterates. Moreover, compared to [143], and the analysis in [77] that relies on Generalized Moreau Envelopes, our proof is significantly shorter and simpler. As a byproduct of this simpler analysis, for `QFedTD` and `OACFedTD`, we derive bounds that have a tighter linear dependence on the mixing time (consistent with the centralized setting) as opposed to the quadratic dependence in [77], [143]. In this regard, we should point out that [77] and [143] look at somewhat more general updating schemes than us by allowing for the agents to perform multiple local updates in every communication round. Instead, we only consider one local step in our analysis. While performing more than one local step leads to a "client-drift" effect [27], [72], [100], it is not clear to us whether/why such a drift effect should lead to sub-optimal dependencies on the mixing time. In fact, the dependence of $O(\tau)$ in our variance bounds (where τ is the mixing time) is information-theoretically optimal [104]. The other natural advantage of our simple proof template is that one can potentially build on it while trying to establish linear speedups for more involved RL settings.

4.2 System Model and Problem Formulation

We consider a setting involving N agents, where all agents interact with the *same* Markov Decision Process (MDP). Let us denote the shared MDP by $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma)$, where $\mathcal{S}$ is a finite state space of size n, $\mathcal{A}$ is a finite action space, $\mathcal{P}$ is a set of action-dependent Markov transition kernels, $\mathcal{R}$ is a reward function, and $\gamma \in (0, 1)$ is the discount factor. We are interested in a *policy evaluation* (PE) problem where the agents exchange information via a central aggregator (server) to evaluate the value function associated with a policy μ. Here, the policy is a map from the states to the actions, i.e., $\mu : \mathcal{S} \to \mathcal{A}$. In what follows, we first briefly review some key concepts relevant to PE with function approximation. Then, we formally describe our communication model, objectives, and technical challenges.

Policy Evaluation with Linear Function Approximation. The policy μ to be evaluated induces a Markov Reward Process (MRP) with transition matrix $\mathbb{P}_\mu$ and reward function $R_\mu : \mathcal{S} \to \mathbb{R}$. The purpose of PE is to evaluate the value function $\boldsymbol{V}_\mu(s)$ for each $s \in \mathcal{S}$, where $\boldsymbol{V}_\mu(s)$ is the discounted expected cumulative reward obtained by

playing policy μ starting from initial state s. Formally, we have

$$\boldsymbol{V}_\mu(s) = \mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k R_\mu(s_k) | s_0 = s\right], \tag{4.1}$$

where s_k represents the state of the Markov chain at the discrete time-step k under the action of the policy μ. Our particular interest is in the RL setting where the Markov transition kernels and reward functions are *unknown*.

In several large-scale practical settings, the size n of the state space $\mathcal{S}$ is large, thereby creating a major computational challenge. To work around this issue, we will resort to the popular idea of linear function approximation where $\boldsymbol{V}_\mu$ is approximated by vectors in a linear subspace of $\mathbb{R}^n$ spanned by a set of m basis vectors $\{\boldsymbol{\phi}_\ell\}_{\ell \in [m]}$[1]; importantly, $m \ll n$. To be more precise, let us define the feature matrix $\boldsymbol{\Phi} \triangleq [\boldsymbol{\phi}_1, ..., \boldsymbol{\phi}_m] \in \mathbb{R}^{n \times m}$. Given a weight (model) vector $\boldsymbol{\theta} \in \mathbb{R}^m$, the parametric approximation $\hat{\boldsymbol{V}}_{\boldsymbol{\theta}}$ of $\boldsymbol{V}_\mu$ is then given by $\boldsymbol{V}(\boldsymbol{\theta}) := \hat{\boldsymbol{V}}_{\boldsymbol{\theta}} = \boldsymbol{\Phi}\boldsymbol{\theta}$. If we denote the s-th row of $\boldsymbol{\Phi}$ as $\boldsymbol{\phi}'_s$, then the approximation of $\boldsymbol{V}_\mu(s)$, in particular, is given by $\hat{\boldsymbol{V}}_{\boldsymbol{\theta}}(s) = \langle \boldsymbol{\theta}, \boldsymbol{\phi}'_s \rangle$. Throughout, we will make the standard assumption [19] that the columns of $\boldsymbol{\Phi}$ are independent and that the rows are normalized, i.e., $\|\boldsymbol{\phi}'_s\|_2^2 \leq 1, \forall s \in \mathcal{S}$.

Given the above setup, the goal of the server-agent system is to collectively estimate the model vector $\boldsymbol{\theta}^*$ corresponding to the best linear approximation of $\boldsymbol{V}_\mu$ in the span of $\boldsymbol{\Phi}$. To achieve this goal, we now describe a multi-agent variant of the classical `TD`(0) algorithm [138]. All agents start out from a common initial state $s_0 \in \mathcal{S}$ with an initial estimate $\boldsymbol{\theta}_0 \in \mathbb{R}^m$. Subsequently, at each time-step $k \in \mathbb{N}$, a global model vector $\boldsymbol{\theta}_k$ is broadcasted by the server to all agents. Each agent $i \in [N]$ then takes an action $a_{i,k} = \mu(s_{i,k})$, and observes the next state $s_{i,k+1} \sim \mathbb{P}_\mu(\cdot | s_{i,k})$ and instantaneous reward $r_{i,k} = R_\mu(s_{i,k})$; here, $s_{i,k}$ is the state of agent i at time-step k. Using the model vector $\boldsymbol{\theta}_k$ and the observation tuple $o_{i,k} = (s_{i,k}, r_{i,k}, s_{i,k+1})$, agent i computes the following local TD update direction:

$$\mathbf{g}_{i,k}(\boldsymbol{\theta}_k, o_{i,k}) = (r_{i,k} + \gamma \langle \boldsymbol{\phi}'_{s_{i,k+1}}, \boldsymbol{\theta}_k \rangle - \langle \boldsymbol{\phi}'_{s_{i,k}}, \boldsymbol{\theta}_k \rangle) \boldsymbol{\phi}'_{s_{i,k}}.$$

We will often use $\mathbf{g}_{i,k}(\boldsymbol{\theta}_k)$ as a shorthand for $\mathbf{g}_{i,k}(\boldsymbol{\theta}_k, o_{i,k})$, and we sometimes also use $\mathbf{g}(\boldsymbol{\theta}_k, o_{i,k})$, omitting the agent-iteration subscript. Note that although all agents play the same policy μ, and interact with the same MDP, the realizations of the local observation sequences $\{o_{i,k}\}$ can differ across agents. We assume that these observation sequences are *statistically independent* across agents.[2] Intuitively, based on this independence property,

[1] Given a positive integer m, we use the notation $[m] = 1, ..., m$.

[2] Notice that for each agent i, the observations over time are, however, correlated since they are all

one can expect that exchanging agents' local TD update directions should help reduce the variance in the estimate of $\boldsymbol{\theta}^*$.

We now provide some technical preparation and machinery that is needed for each of the provided non-asymptotic analysis that we provide for the considered RL schemes. As is standard, we assume that the rewards are uniformly bounded, i.e., $\exists \bar{r} > 0$ such that $R_\mu(s) \leq \bar{r}, \forall s \in \mathcal{S}$. This ensures that the value function in (4.1) is well-defined. Next, we make a standard assumption that plays a key role in the finite-time analysis of TD learning algorithms [19], [135], [141].

Assumption 4. The Markov chain induced by the policy μ is aperiodic and irreducible.

An immediate consequence of the above assumption is that the Markov chain induced by μ admits a unique stationary distribution π [84]. Let $\boldsymbol{\Sigma} = \boldsymbol{\Phi}^\top \mathbf{D} \boldsymbol{\Phi}$, where $\mathbf{D}$ is a diagonal matrix with entries given by the elements of the stationary distribution π. Since $\boldsymbol{\Phi}$ is assumed to be full column rank, $\boldsymbol{\Sigma}$ is full rank with a strictly positive smallest eigenvalue $\omega < 1$; ω will later show up in our convergence bounds. Next, we define the steady-state local TD update direction as follows:

$$\bar{\mathbf{g}}(\boldsymbol{\theta}) \triangleq \mathbb{E}_{s_{i,k} \sim \pi, s_{i,k+1} \sim \mathbb{P}_\mu(\cdot|s_{i,k})} \left[\mathbf{g}_{i,k}(\boldsymbol{\theta}, o_{i,k}) \right], \forall \boldsymbol{\theta} \in \mathbb{R}^m. \tag{4.2}$$

Essentially, the *deterministic* recursion $\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k + \alpha \bar{\mathbf{g}}(\boldsymbol{\theta}_k)$ captures the limiting behavior of the TD(0) update rule. In [19], it was shown that the iterates generated by this recursion converge exponentially fast to $\boldsymbol{\theta}^*$, where $\boldsymbol{\theta}^*$ is the unique solution of the projected Bellman equation $\Pi_{\mathbf{D}} \mathcal{T}_\mu(\boldsymbol{\Phi}\boldsymbol{\theta}^*) = \boldsymbol{\Phi}\boldsymbol{\theta}^*$. Here, $\Pi_{\mathbf{D}}(\cdot)$ is the projection operator onto the subspace spanned by $\{\boldsymbol{\phi}_\ell\}_{\ell \in [m]}$ with respect to the inner product $\langle \cdot, \cdot \rangle_{\mathbf{D}}$, and $\mathcal{T}_\mu : \mathbb{R}^n \to \mathbb{R}^n$ is the policy-specific Bellman operator [141]. We now define the notion of mixing time τ_ϵ that will play a crucial role in our analysis.

Definition 4.2.1. Let τ_ϵ be the minimum time such that the following holds:

$$\left\| \mathbb{E} \left[\mathbf{g}_{i,k}(\boldsymbol{\theta}, o_{i,k}) | o_{i,0} \right] - \bar{\mathbf{g}}(\boldsymbol{\theta}) \right\| \leq \epsilon \left(\|\boldsymbol{\theta}\| + 1 \right), \forall k \geq \tau_\epsilon, \forall \boldsymbol{\theta} \in \mathbb{R}^m, \forall i \in [N], \forall o_{i,0}.^3$$

Assumption 6 implies that the Markov chain induced by μ mixes at a geometric rate [84], i.e., the total variation distance between $\mathbb{P}\left(s_{i,k} = \cdot | s_{i,0} = s\right)$ and the stationary distribution π decays exponentially fast $\forall k \geq 0, \forall i \in [N], \forall s \in \mathcal{S}$. This immediately implies the existence of some $K \geq 1$ such that τ_ϵ in Definition 5.2.1 satisfies $\tau_\epsilon \leq K \log(\frac{1}{\epsilon})$ [35]. Loosely speaking, this means that for a fixed $\boldsymbol{\theta}$, if we want the noisy TD update direction to be ϵ-close (relative to $\boldsymbol{\theta}$) to the steady-state TD direction (where

part of a single Markov chain.

[3] Unless otherwise specified, we use $\|\cdot\|$ to denote the Euclidean norm.

both these directions are evaluated at $\boldsymbol{\theta}$), then the amount of time we need to wait for this to happen scales logarithmically in the precision ϵ. For our purpose, the precision we will require is $\epsilon = \alpha^q$, where q is an integer satisfying $q \geq 2$. Unlike the centralized case where $q = 1$ suffices [19], [135], to establish the linear speedup property, we will require $q \geq 2$. Henceforth, we will drop the subscript of $\epsilon = \alpha^q$ in τ_ϵ and simply refer to τ as the mixing time. Let us define by $\sigma \triangleq \max\{1, \bar{r}, \|\boldsymbol{\theta}^*\|, \delta_0\}$ the "variance" of the observation model for our problem.

Communication Model and `QFedTD` Algorithm. For the `QFedTD` variant of the multi-agent TD scheme, we model two key aspects of realistic communication channels in large-scale FL settings: finite capacity (due to limited bandwidth) and erasures/packet drops. To account for the first issue, we will employ a simple unbiased quantizer which is a (potentially random) mapping $\mathcal{Q} : \mathbb{R}^m \to \mathbb{R}^m$ satisfying the following constraints [18].

Definition 4.2.2. (**Unbiased Quantizer**) We say that a quantizer $\mathcal{Q}$ is unbiased if the following hold for all $\mathbf{x} \in \mathbb{R}^m$: (i) $\mathbb{E}\left[\mathcal{Q}(\mathbf{x})\right] = \mathbf{x}$, and (ii) there exists some constant $\zeta \geq 0$ such that $\mathbb{E}\left[\|\mathcal{Q}(\mathbf{x}) - \mathbf{x}\|_2^2\right] \leq \zeta \|\mathbf{x}\|_2^2$, where the expectation is w.r.t. the randomness of the quantizer.

The constant ζ captures the amount of distortion introduced by the quantizer. Using *any* quantizer that satisfies Definition 4.2.2, each agent i computes an encoded version $\mathbf{h}_{i,k}(\boldsymbol{\theta}_k) = \mathcal{Q}(\mathbf{g}_{i,k}(\boldsymbol{\theta}_k))$ of $\mathbf{g}_{i,k}(\boldsymbol{\theta}_k)$. Here, we assume that the randomness of the quantizer is independent across agents and also independent of the Markovian observation tuples.

Next, to capture packet drops, we assume that the encoded TD directions are uploaded to the server over Bernoulli erasure channels. Specifically, the transmission of information from an agent i to the server is over a channel whose statistics are governed by an i.i.d. random process $\{b_{i,k}\}$, where for each k, $b_{i,k}$ follows a Bernoulli fading distribution. To be more precise, $b_{i,k} = 0$ with erasure probability $(1 - p)$, and $b_{i,k} = 1$ with probability p. The packet-dropping processes are assumed to be independent of all other sources of randomness in our model.

We are now in a position to describe the global parameter update rule at the server for this Quantized Federated TD learning algorithm, to which we refer to as `QFedTD`:

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k + \alpha \mathbf{v}_k; \;\; \mathbf{v}_k = \frac{1}{N} \sum_{i=1}^{N} b_{i,k} \mathbf{h}_{i,k}(\boldsymbol{\theta}_k), \tag{4.3}$$

where α is a constant step-size/learning rate.

Objective and Challenges. We want to provide a *finite-time analysis* of `QFedTD`. This is non-trivial for several reasons. Even in the single-agent setting, providing a non-asymptotic analysis of `TD`(0) without any projection step is known to be quite challenging

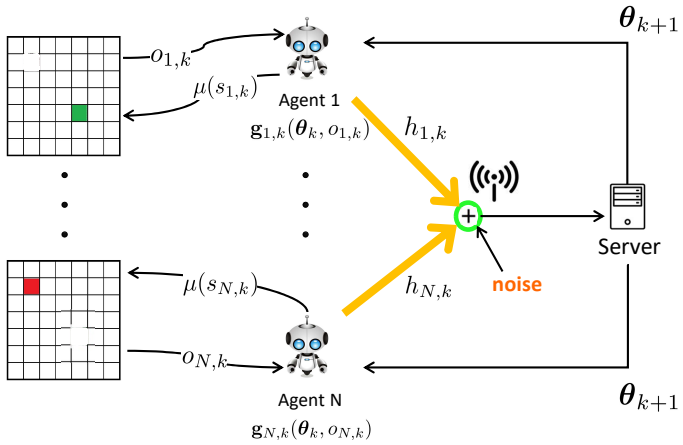

Figure 4.1: Illustration of the `OAC-FedTD` scheme.

due to temporal correlations between the Markov samples. To analyze `QFedTD`, *we need to contend with three distinct sources of randomness*: (i) randomness due to the temporally correlated Markov samples $\{o_{i,k}\}_{i\in[N]}$; (ii) randomness due to the quantization step; and (iii) randomness due to the Bernoulli packet dropping processes $\{b_{i,k}\}_{i\in[N]}$. Each of these sources of randomness influence the evolution of the parameter vector $\boldsymbol{\theta}_k$. Furthermore, unlike a single-agent setting, our goal is to establish a "linear speedup" w.r.t. the number of agents under the different sources of randomness above. This necessitates a very careful analysis that we provide in Appendix B.1.

Remark 4.1. We note here that both the quantization mechanism and the channel model studied for `QFedTD` are quite simple. We have chosen to stick to these models primarily because the focus of our work is on establishing the linear speedup effect under Markovian sampling. That said, we conjecture that the analysis in Section B.1 can potentially be extended to cover more involved encoding schemes (e.g., the use of error-feedback [102]). We reserve these questions for future work.

Over-the-air computation model for `OACFedTD`. We consider the typical OAC channel model that has been adopted, for example, in [7], [25], [127], [154]. In this scheme, N agents, coordinated by a central entity, synchronously transmit their local update directions as analog wireless signals. The central entity then collects the superposition of these signals; hence, the term 'over-the-air.' The analog signals are subject to fading channel distortion and to additive white Gaussian noise at the receiver.

Under the assumptions of synchronization and phase compensation [25], [127], [147], the server at iteration k obtains the following *noisy* and *distorted* global TD direction:

$$\mathbf{v}_k = \frac{1}{N}\sum_{i=1}^{N} h_{i,k}\mathbf{g}_{i,k}(\boldsymbol{\theta}_k) + \mathbf{w}_{k,N}, \tag{4.4}$$

where $\mathbf{w}_{k,N} \sim \mathcal{N}(0, \sigma_{\mathbf{w}}^2 \mathbf{I}_d)$ and $\sigma_{\mathbf{w}}^2 = \tilde{\sigma}_{\mathbf{w}}^2/N^2$, where $\tilde{\sigma}_{\mathbf{w}}^2$ is the additive white noise variance at the receiver. The distortion term $h_{i,k}$ is the *random* channel gain experienced by agent i at iteration k, with mean m_h and variance σ_h^2. We make the standard assumption that the random channel gain process is independent across agents and iterations. We will also assume that the random processes $\{\mathbf{w}_{k,N}\}$ and $\{h_{i,k}\}$ related to the channel effects are independent of the Markovian data tuples $\{o_{i,k}\}$. The model in (4.4) captures different settings of OAC. For example, the model adopted in [25] considers transmitters with adaptive power transmission. In that case, $h_{i,k} = c_{i,k}\sqrt{p_{i,k}}$, where $c_{i,k}$ is the actual channel gain, and $\sqrt{p_{i,k}}$ is the power scaling factor of device i that can be adaptively adjusted to reduce the impact of the channel gain. Due to channel estimation errors [62], even in the case in which channel inversion is performed, $h_{i,k}$ is typically a random object. In general, the model considered in this work captures any OAC framework with phase compensation, as long as the distortion $h_{i,k}$ in (4.4) admits first and second moments.

Once the server receives $\mathbf{v}_k$, it updates the estimate of the parameter $\boldsymbol{\theta}_k$ according to the following update rule, to which we refer to as over-the-air TD learning algorithm, or simply `OAC-FedTD`:

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k + \alpha \mathbf{v}_k, \tag{4.5}$$

where α is a constant step-size/learning rate, and $\mathbf{v}_k$ is as in (4.4).

Objective and Challenges. We aim to provide a *finite-time analysis* of `OAC-FedTD`. This is non-trivial for several reasons. To analyze `OAC-FedTD`, we need to deal with a multi-agent setting where two distinct sources of randomness are concurrently in place: (i) the randomness due to the time-correlated agents' trajectories, and (ii) the randomness due to the wireless fading channel. Furthermore, the final objective of `OAC-FedTD` is to provide a *linear convergence speedup* w.r.t. the number of agents. This requires a careful analysis that we provide in Appendix B.2.

Communication model for `AsyncFedTD`. We now describe the model for `AsyncFedTD`, which is analogous to the models studied, for example, in FL [51], [108] and asynchronous multi-agent RL [130]. At each time-step k, the server updates the model vector $\boldsymbol{\theta}_k$ using the average of asynchronously delayed agents' local TD update directions. Specifically, for each agent i, at iteration k, the corresponding available TD update direction is subject to a *bounded* delay $\tau_{i,k}$. Define $t_{i,k} \triangleq (k - \tau_{i,k})_+$, where, for $x \in \mathbb{R}$, $(x)_+ = \max\{0, x\}$. The server updates the model vector $\boldsymbol{\theta}_k$ according to the following rule, to which we refer

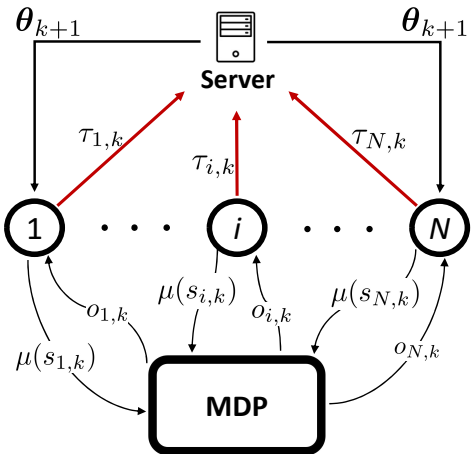

Figure 4.2: System Model for `AsyncFedTD`. Agents $1, \ldots, N$ cooperatively learn a common policy interacting with replicas of the same MDP. At each iteration k, the server uses the available delayed update directions with delays $\tau_{1,k}, \ldots, \tau_{N,k}$.

to as `AsyncFedTD`:

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k + \alpha \mathbf{v}_k, \tag{4.6}$$

where α is a constant step-size/learning rate, and

$$\mathbf{v}_k = \frac{1}{N} \sum_{i=1}^{N} \mathbf{g}(\boldsymbol{\theta}_{t_{i,k}}, o_{i,t_{i,k}}). \tag{4.7}$$

In this work, we assume that the down-link communication from the server to the agents is not subject to delays. Such an assumption is practically motivated by the fact that in most client-server architectures (e.g., wireless networks [29]), the main communication bottleneck comes from up-link transmissions, instead of down-link broadcasting. Note that the update direction $\mathbf{v}_k$ used by the server features iterates $\boldsymbol{\theta}_{t_{i,k}}$ and observations $o_{i,t_{i,k}}$ from potentially stale time-steps. Furthermore, the delays $\tau_{1,k}, \ldots, \tau_{N,k}$ can differ across agents.

We make the following assumption on the delay sequence, which is common in the study of asynchronous distributed optimization and FL [78], [108].

Assumption 5. There exists a positive integer $\tau_{max} > 0$ such that $0 \leq \tau_{i,k} \leq \tau_{max}$, for all i and for all k.

Objective and Challenges. We provide a *finite-time analysis* of `AsyncFedTD`. This poses several challenges. In fact, even in the single-agent setting, providing a non-asymptotic analysis of `TD`(0) without performing intermediate projection steps is known to be challenging due to the temporal correlation between the Markov samples in the

iterative learning process. Crucially, this challenge is absent in asynchronous stochastic optimization where one assumes i.i.d. data, precluding the use of techniques used in this line of work. For the analysis of `AsyncFedTD`, we encounter further obstacles: the update rule involves the use of multiple *correlated iterates* $\boldsymbol{\theta}_{t_{i,k}}$, $i = 1, ..., N$, at which the local TD update directions are asynchronously computed. Indeed, note that, although the observation sequences $o_{i,k}$ are statistically independent across agents, the iterates used to compute the local TD update directions are all correlated. This aspect introduces the need for a much finer analysis when we want to provide finite-time convergence guarantees. Furthermore, unlike a single-agent setting, we aim to establish an N-fold linear convergence speedup while jointly dealing with the challenges outlined above. This necessitates a very careful study, which we detail extensively in Appendix B.3.

4.3 Convergence Results

In this section, we state and discuss our main results pertaining to the non-asymptotic performance of `QFedTD`, `OACFedTD` and `AyncFedTD`. Recall $\delta_k^2 \triangleq \|\boldsymbol{\theta}^* - \boldsymbol{\theta}_k\|^2$. All the proofs of the results of this section are provided in full details in Appendix B.

4.3.1 `QFedTD`: Convergence

We now state the convergence result for `QFedTD`. Let $\zeta' \triangleq \max\{1, \zeta\}$, where ζ is as in Definition 4.2.2.

Theorem 4.2. Consider the update rule of `QFedTD` in (4.3). There exist universal constants $C_0, C_2, C_3 \geq 1$, such that with $\alpha \leq \frac{\omega(1-\gamma)}{C_0 \tau \zeta'}$, the following holds for $T \geq 2\tau$:

$$\mathbb{E}\left[\delta_T^2\right] \leq (1 - \alpha\omega(1-\gamma)p)^T C_1 + \frac{\tau\sigma^2}{\omega(1-\gamma)}\left(\frac{C_2\alpha\zeta'}{N} + C_3\alpha^3\right), \tag{4.8}$$

where $C_1 = 4\delta_0^2 + 2p\sigma^2$.

Discussion: There are several important takeaways from Theorem 4.2. From (4.12), we first note that `QFedTD` guarantees linear convergence (in expectation) to a ball around $\boldsymbol{\theta}^*$ whose radius depends on the variance σ^2 of the noise model. While the linear convergence rate gets slackened by the probability of successful transmission p, the "variance term", namely the second term in (4.8), gets inflated by the quantization parameter ζ. Both of these channel effects are consistent with what one typically observes for analogous settings in FL [121]. Next, compared to the centralized setting [135, Theorem 7], the variance term in (4.8) gets scaled down by a factor of N, up to a

higher-order $O(\alpha^3)$ term that can be dominated by the (α/N) term for small enough α. Note that this α^3 term is obtained thanks to mixing arguments relative to the biasedeness of the TD direction $\mathbf{g}(.)$, and we do not see a way to avoid the presence of this term. Before we make the point on the linear speedup effect explicit, it is instructive to note that our variance bound exhibits a tighter dependence on the mixing time τ relative to [77] and [143], where similar bounds are established. In particular, while this dependence is $O(\tau)$ for us, it is $O(\tau^2)$ in [77, Theorem 4.1] and in [143, Theorem 4]. Notably, the $O(\tau)$ dependence that we establish is consistent with results on centralized TD learning [19], [135], and is in fact the optimal dependence on τ under Markovian data [104]. We have the following immediate corollary of Theorem 4.2.

Corollary 4.1. Consider the update rule of `QFedTD` in (4.3). Let the step-size α and the number of iterations T be chosen to satisfy:

$$\alpha = \frac{\log NT}{\omega(1-\gamma)pT}, \text{ and } T \geq \frac{2C_0 N\tau\zeta' \log NT}{\omega^2(1-\gamma)^2 p}, \tag{4.9}$$

where C_0 is as in Theorem 4.2. We then have the following bound:

$$\mathbb{E}[\delta_T^2] \leq O\left(\left(\frac{\zeta'}{p}\right)\frac{\max\{\delta_0^2, \sigma^2\}\tau \log(NT)}{\omega^2(1-\gamma)^2 \mathrm{tr}\, NT}\right). \tag{4.10}$$

To appreciate the above result, let us compare it to the result for the single agent TD setting in [19]. Under Markovian sampling, part (c) of Theorem 3 in [19] establishes that the mean-square error for single-agent TD decays at the following rate:

$$O\left(\frac{G^2\tau \log(T)}{\omega^2(1-\gamma)^2 T}\right),$$

where G, as defined in [19], captures the effect of both the projection radius (in [19], the authors consider a projected version of TD learning) and the noise variance.[4] The term G^2 can be viewed as the analog of $\max\{\delta_0^2, \sigma^2\}$ in our bound. Comparing the above bound with that in Eq. (4.10), we make two immediate observations. (i) The term T in the centralized bound gets replaced by NT in our bound. This is precisely what we wanted since in our setting, each agent has access to T samples, yielding a total of NT samples. Essentially, this goes on to show that our algorithm is *sample-efficient* in that it makes use of all the samples from all the agents and achieves a linear speedup w.r.t. the number of agents. Second, the effect of channel effects is succinctly captured by the term

[4] Part (c) of Theorem 3 in [19] provides a bound on the error in the value function, and not the iterates (like we do). The bound on the iterates that we report above is derived from the bound on the value functions in Appendix A.2 of [19], where the authors provide a proof of Theorem 3.

in blue in Eq. (4.10). This term essentially inflates the variance $\max\{\delta_0^2, \sigma^2\}$ of our noise model. When the number of agents $N = 1$, the probability of successful transmission $p = 1$, and there is no quantization effect (i.e., $\zeta' = 1$), our bound exactly recovers the bound in the centralized setting (even up to log factors). As far as we are aware, our work is the first to establish such a tight result in multi-agent/federated reinforcement learning under Markovian sampling and communication constraints.

4.3.2 `OACFedTD`: Convergence

We now present the first finite-time result in RL with OAC. Notably, we consider the challenging case in which agents' trajectories follow a Markov process, and show that cooperation between agents provides a linear convergence speedup even under noisy analog communication over wireless fading channels.

Theorem 4.3. Consider the update rule of `OAC-FedTD` in (4.5). There exists a universal constant $C_0 \geq 1$, such that with $\alpha \leq \frac{m_h \omega(1-\gamma)}{C_0 \tau p_h}$, the following holds for $T \geq 2\tau$:

$$\begin{aligned}\mathbb{E}\left[\delta_T^2\right] \leq (1 - m_h \alpha \omega(1-\gamma))^T C_1 + \frac{C_2 p_h \alpha \tau \sigma^2}{m_h \omega(1-\gamma) N} \\ + \frac{C_3 p_h \tau \sigma^2 \alpha^3}{m_h \omega(1-\gamma)} + \frac{C_4 \alpha \tau \tilde{\sigma}_w^2 d}{m_h \omega(1-\gamma) N^2},\end{aligned} \tag{4.11}$$

where $C_1 = 4\delta_0^2 + 2\sigma^2 + 2\frac{\tilde{\sigma}_{\mathbf{w}}^2 d}{N^2}$, and C_2, C_3, C_4 are universal constants.

A detailed proof of Theorem 4.3 is provided in the Appendix, where we outline the key technical challenges relative to the centralized analysis in [135].

Discussion: We now discuss the main takeaways from Theorem 4.3. From (4.11), we first note that `OAC-FedTD` guarantees linear convergence (in the mean-square sense) to a ball around $\boldsymbol{\theta}^*$ whose radius depends on the second, third and fourth terms in (4.11). The linear convergence rate gets slackened by both the mean distortion m_h, and by the choice of the step size, which needs to scale inversely with $p_h \tau$. The term p_h also inflates the dominant "variance term", namely the second term in (4.11). So, given that $\mathbb{E}\left[h_{i,k}^2\right] = m_h^2 + \sigma_h^2$ and recalling that $p_h = \max\{1, m_h^2 + \sigma_h^2\}$, our bound clearly reveals the effect of fading distortion. This channel effect is consistent with what one observes for analogous settings in FL with OAC [127]. Compared with the effect of noise in FL via OAC, we note that the variance term related to the additive noise at the receiver, i.e., the fourth term in (4.11), gets scaled by the mixing time τ. Next, compared to the centralized setting [135, Theorem 7], observe that the second and fourth terms in (4.11) get scaled down by a factor of N. Moreover, the third term is $O(\alpha^3)$, i.e., it is a higher-order term

that is dominated by the second term for small enough α. *Thus, ours is the first work in MARL/FRL over wireless fading channels to establish a variance-reduction effect w.r.t. the number of agents.* With $\alpha = O(\log(NT)/T)$, we can explicitly show that each of the four terms in (4.11) is $O(1/NT)$, yielding the linear speedup effect we had hoped for. Finally, note that, compared to the only other very recent paper [77] that establishes linear speedup under Markovian sampling (albeit, without channel effects), the second, third, and fourth terms in (4.11) have a tighter dependence on the mixing time τ. Indeed, while we achieve a linear dependence of $O(\tau)$, which is consistent with the centralized setting [135], the dependence in [77, Theorem 4.1] is $O(\tau^2)$.

4.3.3 `AsyncFedTD`: Convergence

Let $\delta_k^2 \triangleq \|\boldsymbol{\theta}^* - \boldsymbol{\theta}_k\|^2$ and define by $\sigma \triangleq \max\{1, \bar{r}, \|\boldsymbol{\theta}^*\|, \delta_0\}$ the "variance" of the observation model for our problem. We can now state our convergence result for `AsyncFedTD`.

Theorem 4.4. Consider the update rule of `AsyncFedTD` in (4.6). There exist universal constants $C_0, C_1, C_2, C_3 \geq 1$, such that, for $\alpha \leq \frac{\omega(1-\gamma)}{C_0(\tau+\tau_{max})}$ and $T \geq 2\tau + \tau_{max}$,

$$\begin{aligned}\mathbb{E}\left[\delta_T^2\right] \leq \exp\left(-\frac{\alpha(1-\gamma)\omega T}{2(\tau+\tau_{max})}\right) C_1\sigma^2 \\ + \frac{(\tau+\tau_{max})\sigma^2}{2\omega(1-\gamma)}\left(\frac{C_2\alpha}{N} + C_3\alpha^3\right).\end{aligned} \tag{4.12}$$

Discussion: We now remark on the main takeaways from Theorem 4.4. From the bound in (4.12), we note that `AsyncFedTD` guarantees linear convergence (in mean-square sense) to a ball around $\boldsymbol{\theta}^*$ whose radius depends on the variance σ^2 of the noise model. We now comment on the effect of the asynchronous delays on the convergence bound, and on the linear convergence speedup established by the theorem.

Effect of asynchronous delays. From (4.12), note how both the exponent of the linear convergence term and the radius of the noise ball are impacted by the delay sequence via the maximum delay τ_{max}. Indeed, compared to the centralized TD case [135, Theorem 7], and to the synchronous federated TD case [44, Theorem 1], we see that for `AsyncFedTD`, the noise ball gets multiplied by the sum of mixing time and maximum delay, i.e., $\tau + \tau_{max}$. In essence, our analysis reveals that $\tau + \tau_{max}$ plays the role of an *effective* delay. Interestingly, an immediate implication is that if the underlying Markov chain mixes slowly, i.e., has a larger mixing time τ, then the effect of the delay is less pronounced. This appears to be a novel observation.

Linear convergence speedup. Compared to the centralized setting [135, Theorem 7], the noise variance term in (4.12) gets scaled down by a factor of N up to a higher-order

$O(\alpha^3)$ term that, for small enough α, is dominated by (α/N). To better illustrate the linear speedup effect, consider the following choice of α and T (define $\bar{\tau} \triangleq \tau + \tau_{max}$):

$$\alpha = \frac{\bar{\tau}\log NT}{\omega(1-\gamma)T}, \text{ and } T \geq \frac{2C_0 N\bar{\tau}^2 \log NT}{\omega^2(1-\gamma)^2}. \tag{4.13}$$

With the above choices, and simple manipulations of the bound in (4.12), it can be explicitly shown that

$$\mathbb{E}[\delta_T^2] \leq O\left(\frac{\sigma^2\bar{\tau}^2\log(NT)}{\omega^2(1-\gamma)^2 NT}\right). \tag{4.14}$$

The above bound tells us that `AsyncFedTD` yields a convergence rate of $O(1/(NT))$, which is a factor of N better than the $O(1/T)$ rate in the centralized case [19]. Note the quadratic dependence on both the delay sequence and on the mixing time of the Markov chain, that we can see in the term $\bar{\tau}^2$. Note that in the convergence results for `QFedTD` and `OACFedTD` this dependence was linear which is the optimal dependence from an information theoretic point of view. We further study this aspect in the more general framework of stochastic approximation under Markovian sampling and delayed updates in Chapter 5, where we derive optimal dependencies on τ_{max} and τ for the single-agent case of delayed SA.

We remark that this is the first analysis for asynchronous multi-agent and federated RL that provides finite-time convergence guarantees, while jointly establishing an N-fold linear convergence speedup under Markovian sampling.

4.4 Numerical Simulations

In this section, we provide simulation results to validate our theory. We consider an MDP with $|\mathcal{S}| = 100$ states and a feature space spanned by $d = 10$ orthonormal basis vectors; we set the discount factor to $\gamma = 0.5$ and the step size to $\alpha = 0.05$. For `AsyncFedTD`, to simulate the asynchronous delays in the TD update directions, we generate random delays at each iteration k for each agent i, by generating a uniform random variable $\tau_{i,k}$ in the range $[1, \tau_{max}]$. We set $\tau_{max} = 100$. For `QFedTD`, we generate the erasure channels with Bernoulli random variables, and employ uniform scalar quantization of the TD update directions, assigning a certain number of bits for the quantization of each vector component of each agent, and set the probability of successful transmission to $p = 0.6$, while we quantize the TD update directions assigning 4 bits to each vector component. For `OACFedTD`, we generate the channel distortion $h_{i,k}$ (with mean m_h and variance σ_h^2) as a Rayleigh random variable, which is a widely adopted model for fading channels

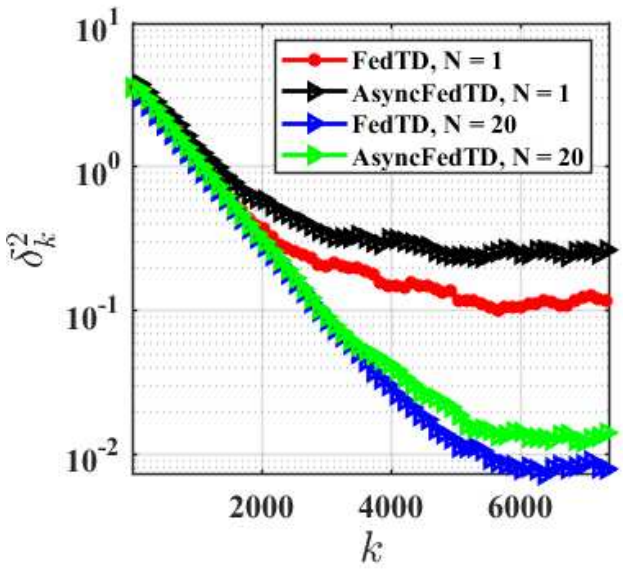

Figure 4.3: Comparison between vanilla `FedTD` and `AsyncFedTD` in single-agent ($N = 1$) and multi-agent ($N = 20$) settings. For `AsyncFedTD`, we set $\tau_{max} = 100$.

[127]. In the multi-agent seting, we experiment with different numbers of agents, like $N = 15, 20, 40$. For each configuration, we plot the average of 20 experiments. From the results, it is apparent how the linear speedup property also holds for `QFedTD` (Figure 4.5), `OACFedTD` (Figure 4.4) and `AsyncFedTD` (Fig 4.3).

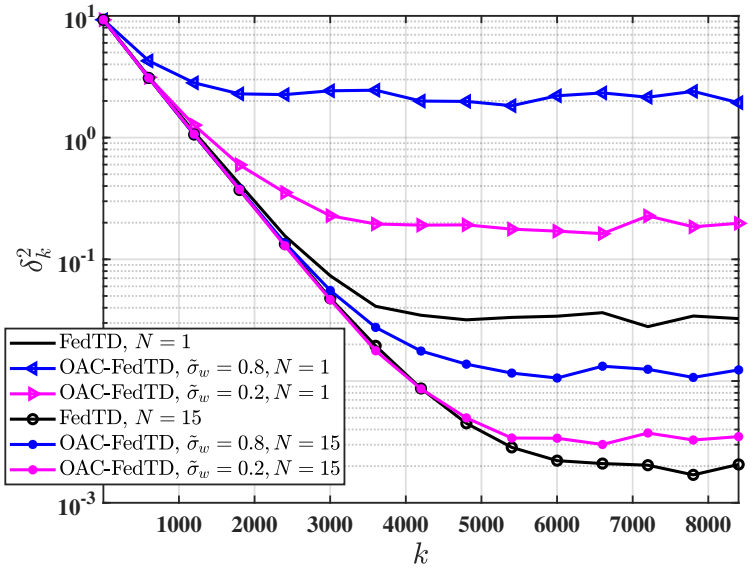

Figure 4.4: Performance of `OAC-FedTD` for different values of the standard deviation of the measurement noise at the receiver ($\tilde{\sigma}_{\mathbf{w}} = 0.2, 0.8$), and for different values of the number of cooperating agents ($N = 1, 15$).

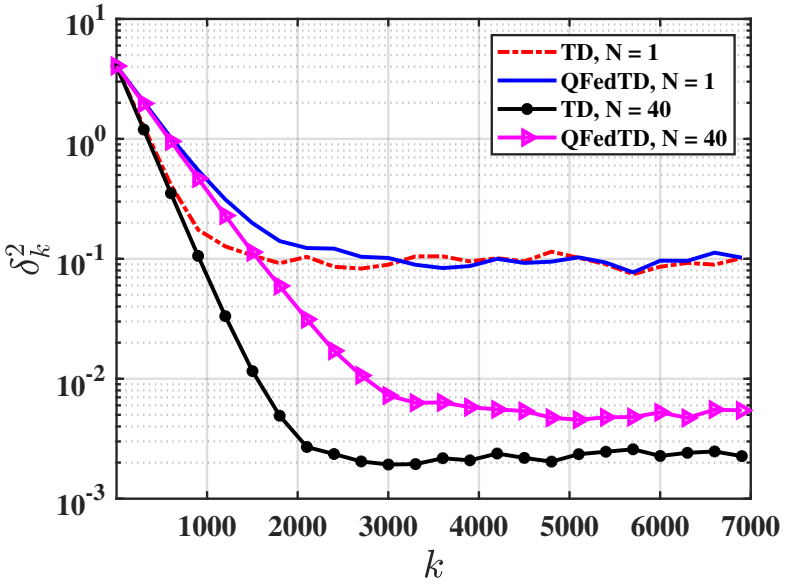

Figure 4.5: Comparison between vanilla `FedTD` and `QFedTD` in single-agent ($N = 1$) and multi-agent ($N = 40$) settings. The number of bits used to quantize the TD update direction is 4 per vector component, and the erasure probability is $p = 0.6$.

4.5 Related Publications and Conference Presentations

The content of this chapter has been published in the IEEE Control Systems Letters [55] and accepted for presentation at the Machine Learning and Systems conference, 2023, in the Workshop on Resource-Constrained Learning in Wireless Networks [43]. Content of this chapter has also be accepted for presentation at the IEEE Conference on Decision and Control in December 2023 and submitted to the American Control Conference, 2024.

5

Stochastic Approximation with Delayed Updates: Finite-Time Rates under Markovian Sampling with Optimal Dependencies

In Chapter 4, we studied a model of asynchronous federated reinforcement learning (FRL). The convergence bound that we derived for the considered model, in which TD update directions are computed at stale iterates and observations from the past, showed a sub-optimal quadratic dependence on the delay sequence and on the mixing time of the Markov chain (see Theorem 4.4 and related discussion). In this chapter, we focus on this specific aspect of the dependency on mixing time and delay sequence, studying a - single-agent - more general stochastic approximation (SA) setting under Markovian sampling with delayed updates. In this setup , iterative updates of SA are based on delayed versions of the SA operator evaluated at stale iterates and samples from the past. We are interested in understanding the finite-time performance of this updating scheme with a focus on characterizing the interplay between the properties of the underlying Markov process and the delay sequence. Our first contribution is to show that, under standard assumptions, the delayed SA update rule guarantees exponentially fast convergence to a ball around the desired fixed point of the operator. We establish that in a constant delay scenario, the optimal convergence rate achieved by the delayed SA algorithm is scaled down by a (information-theoretically optimal) factor of $\max\{\tau, \tau_{mix}\}$, where τ denotes the constant delay, and τ_{mix} represents the mixing time of the Markov process. This result is proven using a technique inspired by previous works that utilizes the weighted average of iterates. This technique that works for the constant delay case cannot be directly generalized to time-varying delay cases. To address this issue, we propose an approach that involves proving the boundedness of the SA iterates for a suitable choice of step size. We then conduct a novel analysis to show that in the case of

time-varying delays, the exponent of convergence for the last iterate is scaled down by a (information-theoretically optimal) factor of $\max\{\tau_{max}, \tau_{mix}\}$. Here, τ_{max} represents the maximum delay. This is the first result to provide finite-time rates for SA under time-varying delayed updates, while establishing a tight bound in both τ_{max} and τ_{mix} on the last iterate. Our theoretical findings apply to various algorithms where the finite-time effects of delays were previously unknown, such as TD learning and Q-learning with function approximation, and stochastic gradient descent under Markovian sampling.

5.1 Introduction

Stochastic Approximation (SA) is an iterative technique used to solve root-finding problems in the presence of noisy information. This method finds its application in various fields such as machine learning and reinforcement learning. In this section, we will provide a brief summary of previous works and then highlight our contributions.

The Stochastic Approximation (SA) framework, was originally introduced in 1951 [124] and it has been extensively studied in the literature, with a focus on understanding its convergence behavior and applications in various domains. Several works have contributed to the finite-time analysis of SA algorithms [20], [32], [33], [135], [149].

In this chapter, we make significant contributions to the field of stochastic approximation (SA) by studying the non-asymptotic convergence rates of SA under Markovian sampling with delayed updates. Our investigation focuses on understanding the finite-time performance of this updating scheme, considering the interplay between the properties of the underlying Markov process and the delay sequence. Our contributions are the following:

1. The first major contribution of this work is the exploration of finite-time analysis for delayed Stochastic Approximation (SA) under Markovian sampling. We delve into the analysis of the joint effect of delayed updates and the correlated Markov observation process on the convergence behaviour of SA algorithms, which has not been studied before.

2. *Optimal dependencies.* Starting from a setting with constant delay, we establish that a carefully weighted average of iterates achieves a convergence rate with optimal dependencies on the delay sequence and on the mixing time of the Markov chain, with the exponent of convergence scaled down by a factor of $\max\{\tau, \tau_{mix}\}$, where τ represents the constant delay, and τ_{mix} denotes the mixing time of the Markov chain. This result sheds light on the trade-off between the delay and mixing time, providing valuable insights into the impact of the delay on convergence performance.

3. *Optimal dependencies.* We then expand our analysis to more general time-varying delays settings. The analysis with constant delay cannot be directly applied to the time-varying delay case, and it also guarantees a bound only on a weighted average of iterates. Therefore, we introduce a novel approach for the time-varying case, proposing an analysis that involves proving the boundedness of SA iterates for a suitable choice of step size. We then show that with this new analysis the exponent of convergence for the last iterate is scaled down by a factor of $\max\{\tau_{max}, \tau_{mix}\}$, where τ_{max} represents the maximum delay. This is the first study to achieve a tight bound on both τ_{max} and τ_{mix}.

In summary, our work provides a comprehensive analysis of the finite-time convergence behaviour of delayed Stochastic Approximation. The insights and results presented in this work pave the way for the development of more efficient and adaptive algorithms that can handle delays effectively in a wide range of applications.

Related Work in Optimization. The impact on the convergence bounds of optimization algorithms under delays and asynchronous settings has been a topic of interest since the seminal work [17], which investigates convergence rates of asynchronous iterative algorithms in parallel or distributed computing systems. In this same area there have been subsequently many efforts, usually focused on stochastic gradient descent and federated learning, like, for example, [3], [36], [78], [137].

5.2 Stochastic Approximation with Delayed Updates

The objective of general SA is to solve a root finding problem of the following form:

$$\text{Find } \boldsymbol{\theta}^* \in \mathbb{R}^m \text{ such that } \bar{\mathbf{g}}(\boldsymbol{\theta}^*) = 0, \tag{5.1}$$

where, for a given approximation parameter $\boldsymbol{\theta} \in \mathbb{R}^m$, the deterministic function $\bar{\mathbf{g}}(\boldsymbol{\theta})$ is the expectation of a noisy operator $\mathbf{g}(\boldsymbol{\theta}, o_t)$, and $\{o_t\}$ denotes a stochastic *observation process*. In this work, we consider SA under Markovian sampling, i.e., the observations $\{o_t\}$ are temporally correlated and form a Markov chain. For this SA scheme, we define (see also [19], [135], [149])

$$\bar{\mathbf{g}}(\boldsymbol{\theta}) \triangleq \mathbb{E}_{o_t \sim \pi}[\mathbf{g}(\boldsymbol{\theta}, o_t)], \tag{5.2}$$

where π is the *stationary distribution* of the Markov chain $\{o_t\}$.

SA consists in finding an approximate solution to (5.1) while having access only to *noisy* instances $\mathbf{g}(\boldsymbol{\theta}, o_t)$ of $\bar{\mathbf{g}}(\boldsymbol{\theta})$. The typical iterative SA update rule with a constant

step size α is as follows [34], [135],

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k + \alpha \mathbf{g}(\boldsymbol{\theta}_k, o_t). \tag{5.3}$$

The asymptotic convergence of SA under Markov randomness method has been thoroughly investigated in prior work [140]. Recently, there is an increased interest in *finite-time convergence guarantees* for SA. Finite-time analysis of SA is relevant because it provides insightful theoretical guarantees about the algorithm's convergence rate.

Several recent works have investigated linear [19], [135] and non-linear [34] SA, and provided finite-time convergence bounds under Markovian sampling. Notably, Finite-time convergence analysis for SA under Markovian sampling are significantly more challenging relative to i.i.d. sampling. Indeed, temporal correlation between samples of $\{o_t\}$, which is also inherited by the iterates $\{\boldsymbol{\theta}_k\}$, prevents the use of some techniques commonly used for the finite-time rates study of SA under i.i.d. sampling, triggering the need for a more elaborate analysis. In many real-world applications, like the FRL framework considered in Chapter 4, the SA operator $\mathbf{g}(\cdot)$ is only available when computed with delayed iterates and/or observations. The main objective of this work is to provide a unified framework to analyse the finite-time convergence of SA under delays. We proceed by formally introducing the setting.

SA with delays. We consider the following stochastic recursion with delayed updates:

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k + \alpha \mathbf{g}(\boldsymbol{\theta}_{k-\tau}, o_{t-\tau_t}), \quad \tau_t \leq t, \tag{5.4}$$

where α is a constant step size and τ_t is the delay with which the operator $\mathbf{g}(\cdot)$ is available to be used at iteration t. This specific update rule is motivated by many scenarios of practical interest. For instance, in distributed machine learning and reinforcement learning, it is often the case that the agents' updates are performed in an asynchronous manner, leading naturally to update rules of the form (5.4).

Update rules of the form (5.4) have been recently studied in the context of SA but with i.i.d. observations (see e.g. [78], [109] for SGD updates with delays). However, to the best of our knowledge, little to nothing is known about the finite-time convergence behaviour of such update rules under Markovian observations. Compared with i.i.d. setting, the Markovian setting introduces major technical challenges, including dealing with the joint effect of (i) the use of a delayed operator $\mathbf{g}(\boldsymbol{\theta}_{k-\tau}, o_{t-\tau_t})$ and (ii) sequences of correlated observation samples $\{o_t\}$. The interplay of update rules based on delayed SA operator instances and the presence of time correlation in the noise process requires a

notably careful analysis, one which we provide as a main contribution of this work. The key features and challenges of the analysis are provided with more details in section 5.4.

We proceed with describing a few assumptions needed for our analysis. First, we make the following natural assumption on the underlying Markov chain $\{o_t\}$ [19], [34], [135].

Assumption 6. The Markov chain $\{o_t\}$ is aperiodic and irreducible.

Next, we state two further assumptions that are common in the analysis of SA algorithms.

Assumption 7. Problem (5.1) admits a solution $\boldsymbol{\theta}^*$, and $\exists\, \mu > 0$ such that for all $\boldsymbol{\theta} \in \mathbb{R}^m$, we have

$$\langle \boldsymbol{\theta} - \boldsymbol{\theta}^*, \bar{\mathbf{g}}(\boldsymbol{\theta}) - \bar{\mathbf{g}}(\boldsymbol{\theta}^*) \rangle \leq -\mu \|\boldsymbol{\theta} - \boldsymbol{\theta}^*\|^2. \tag{5.5}$$

Assumption 8. For every $\boldsymbol{\theta}_1, \boldsymbol{\theta}_2$ and $o \in \{o_t\}$, we have

$$\|\mathbf{g}(\boldsymbol{\theta}_1, o) - \mathbf{g}(\boldsymbol{\theta}_2, o)\| \leq L \|\boldsymbol{\theta}_1 - \boldsymbol{\theta}_2\|. \tag{5.6}$$

Assumption 9. For any $\boldsymbol{\theta} \in \mathbb{R}^m$ and $o \in \{o_t\}$, we have

$$\|\mathbf{g}(\boldsymbol{\theta}, o)\| \leq L(\|\boldsymbol{\theta}\| + \sigma). \tag{5.7}$$

Finally, we introduce an assumption on the time-varying delay sequence $\{\tau_t\}$.

Assumption 10. There exists an integer $\tau_{max} \geq 0$ such that $\tau_t \leq \tau_{max}, \ \ \forall t \geq 0$.

Assumption 7 is a strong monotone property of the map $-\bar{\mathbf{g}}(\boldsymbol{\theta})$ that guarantees that the iterates generated by a "mean-path" version of Eq. (5.1), $\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k + \alpha \bar{\mathbf{g}}(\boldsymbol{\theta}_k)$, converge exponentially fast to $\boldsymbol{\theta}^*$. Assumption 8 states that $\mathbf{g}(\boldsymbol{\theta}, o_t)$ is globally uniformly Lipschitz in the parameter $\boldsymbol{\theta}$. Without loss of generality, we have fixed the Lipschitz constant to be $L = 2$, which is the Lipschitz constant value in the case of TD learning with linear function approximation.

Remark 5.1. Assumption 7 holds for TD learning (Lemma 1 and Lemma 3 in [19]), Q-learning [34], and SGD for strongly convex functions. Similarly, Assumption 8 holds for TD learning [19], and for Q-learning and SGD analysis, it holds up to some constant L [34], [48]. Our proof technique generalizes to this setting easily. Furthermore, Assumption 9 holds for TD learning [19], and for Q-learning, it holds up to some constant [34].

We now introduce the following notion of *mixing time* τ_α, that plays a crucial role in our analysis, as in the analysis of all existing finite-time convergence studies on SA under Markovian sampling.

Table 5.1: Summary of results.

Algorithm	Variance Bound	Bias Bound
Constant Delay (5.8)	$O(\sigma^2)$	$O\left(\exp\left(\frac{-\mu^2 T}{L^2 \max\{\tau,\tau_{mix}\}}\right)\right)$
Time-Varying Delays (5.26)	$O(\sigma^2)$	$O\left(\exp\left(\frac{-\mu^2 T}{L^2 \max\{\tau_{mix},\tau_{max}\}}\right)\right)$

Definition 5.2.1. Let τ_α be such that

$$\|\mathbb{E}\left[\mathbf{g}(\boldsymbol{\theta}, o_t)|o_0\right] - \bar{\mathbf{g}}(\boldsymbol{\theta})\| \leq \alpha\left(\|\boldsymbol{\theta}\| + 1\right), \forall t \geq \tau_\alpha, \forall \boldsymbol{\theta} \in \mathbb{R}^m, \forall o_0.$$

In the rest of the Chapter, we refer to τ_α simply as τ_{mix}.

Remark 5.2. Note that Assumption 6 implies that the Markov chain $\{o_t\}$ mixes at a geometric rate. This, in turn, implies the existence of some $K \geq 1$ such that τ_α in Definition 5.2.1 satisfies $\tau_\alpha \leq K \log(\frac{1}{\alpha})$. In words, this means that for a fixed $\boldsymbol{\theta}$, if we want the noisy operator $\mathbf{g}(\boldsymbol{\theta}, o_t)$ to be α-close (relative to $\boldsymbol{\theta}$) to the expected operator $\bar{\mathbf{g}}(\boldsymbol{\theta})$, then the amount of time we need to wait for this to happen scales logarithmically in the precision α.

5.3 Warm Up: Stochastic Approximation with Constant Delays

In this section, we present the first finite-time convergence analysis of SA with constant delay under Markovian sampling. With respect to the SA with delayed updates introduced in (5.4), we fix $\tau_t = \tau$, with τ the constant delay. Similarly to [137], we define the SA update rule accordingly:

$$\textit{SA with Constant Delay:} \quad \boldsymbol{\theta}_{t+1} = \begin{cases} \boldsymbol{\theta}_0 & \text{if } 0 \leq t < \tau \\ \boldsymbol{\theta}_t + \alpha \mathbf{g}(\boldsymbol{\theta}_{t-\tau}, o_{t-\tau}) & \text{if } t \geq \tau \end{cases} \tag{5.8}$$

The following Theorem provides a finite-time convergence bound for the update rule in (5.8). The form and analysis of the Theorem is inspired by [137] and it is the first contribution of this work.

Theorem 5.3. Let $w_t := (1-0.5\alpha\mu)^{-(t+1)}$ and $W_T = \sum_{t=0}^{T} w_t$. Let $\boldsymbol{\theta}_{out}$ be a randomly chosen iterate from $\{\boldsymbol{\theta}_t\}_{t=0}^{T}$. Specifically, $\boldsymbol{\theta}_{out} = \boldsymbol{\theta}_t$ with probability $\frac{w_t}{W_T}$. Define $r_{out} := \|\boldsymbol{\theta}_{out} - \boldsymbol{\theta}^*\|$ and $\bar{\tau} := \max\{\tau, \tau_{mix}\}$. For $T \geq 0$, there exist universal constants $C_1, C_2, C_3 \geq 2$, such that, for $\alpha \leq \frac{\mu}{C_1 L^2 \bar{\tau}}$, applying the update rule in (5.8), the following holds,

$$\mathbb{E}\left[r_{out}^2\right] \leq C_\alpha \exp\left(-0.5\alpha\mu T\right) r_0^2 + C_2 \frac{\alpha L^2 \bar{\tau} \sigma^2}{\mu}, \tag{5.9}$$

where C_α is inversely proportional to α. Setting $\alpha = \frac{\mu}{C_1 L^2 \bar{\tau}}$, we get

$$\mathbb{E}\left[r_{out}^2\right] \leq C_{\bar{\tau}} \exp\left(-0.5 \frac{\mu^2}{C_1 L^2 \bar{\tau}} T\right) r_0^2 + \frac{C_2 \sigma^2}{C_1}, \tag{5.10}$$

with $C_{\bar{\tau}} = \frac{\bar{\tau}}{\mu}\left(\frac{2C_1 L^2}{\mu} + 4B\right)$, with $B = C_3 \sigma^2$.

Main Takeaways: We now outline the key takeaways of the above Theorem. First, we showed exponential convergence of $\mathbb{E}\left[r_{out}^2\right]$ to a ball around the fixed point $\boldsymbol{\theta}^*$. This latter result represents the first finite-time convergence bound for SA with delayed updates under Markovian sampling. Second, the obtained convergence exponent scales inversely with $\bar{\tau} = \max\{\tau, \tau_{mix}\}$. Hence, if $\tau \geq \tau_{mix}$, we get a dependency on the constant delay τ which is consistent with what is known for SA with delayed updates in the i.i.d. sampling case, specifically in the case of SGD with constant delay [137]. Note that this dependency has been shown to be tight for SGD [10], and, consequently, our rate is optimal in terms of the obtained dependency on the delay τ. If $\tau_{mix} \geq \tau$, the obtained convergence exponent scales inversely with τ_{mix}, which is consistent with the non-delayed case of SA under Markovian sampling [19], [135], and has been shown to be in fact minimax optimal [105]. In summary, in the above Theorem we provide the first finite-time convergence bound for SA with updates subject to constant delays under Markovian sampling, getting a convergence rate that has optimal dependencies on both the delay τ and the mixing time τ_{mix}.

Outline of the Analysis and Challenges. We now provide the main steps of the analysis and underline the key challenges that make each step necessary. First of all, similarly to [137], we define a sequence of virtual iterates, $\tilde{\boldsymbol{\theta}}_t$, which, at each iteration t, are updated with the actual SA update direction $\mathbf{g}(\boldsymbol{\theta}_k, o_t)$:

$$\tilde{\boldsymbol{\theta}}_{t+1} = \tilde{\boldsymbol{\theta}}_t + \alpha \mathbf{g}(\boldsymbol{\theta}_t, o_t). \tag{5.11}$$

Accordingly, we define an error term $\mathbf{d}_t$ which is the gap between $\boldsymbol{\theta}_k$ and $\tilde{\boldsymbol{\theta}}_t$ at each iteration t, i.e., $\tilde{\boldsymbol{\theta}}_t = \boldsymbol{\theta}_k + \mathbf{d}_t$. A key step in the analysis that, as it was the case for [137],

relies on the fact that the delay τ is constant, is that, for any $t \geq 0$, we can write the error term as follows,

$$\mathbf{d}_t = \alpha \sum_{l=t-\tau}^{t-1} \mathbf{g}(\boldsymbol{\theta}_l, o_l). \tag{5.12}$$

In the first part of the proof of Theorem 5.3, we provide a convergence bound for the virtual iterates sequence $\tilde{\boldsymbol{\theta}}_t$, studying $\mathbb{E}\left[\tilde{r}_t^2\right] = \mathbb{E}\left[\|\tilde{\boldsymbol{\theta}}_t - \boldsymbol{\theta}^*\|^2\right]$ and providing a bound that is a function of $\|\mathbf{d}_t\|^2$ and $\tilde{r}_l^2, \|\mathbf{d}_l\|^2$, with $l = t - \bar{\tau}, ..., t-1$. To get this bound, we analyze the following recursion

$$\begin{aligned}
\tilde{r}_{t+1}^2 &= \tilde{r}_t^2 + 2\alpha\langle \mathbf{g}(\boldsymbol{\theta}_k, o_t), \tilde{\boldsymbol{\theta}}_t - \boldsymbol{\theta}^*\rangle + \alpha^2\|\mathbf{g}(\boldsymbol{\theta}_k, o_t)\|^2 \\
&= \tilde{r}_t^2 + 2\alpha\langle \bar{\mathbf{g}}(\tilde{\boldsymbol{\theta}}_t), \tilde{\boldsymbol{\theta}}_t - \boldsymbol{\theta}^\star\rangle + 2\alpha h_t + 2\alpha m_t + \alpha^2\|\mathbf{g}(\boldsymbol{\theta}_k, o_t)\|^2 \\
&\leq (1 - 2\alpha\mu)\tilde{r}_t^2 + 2\alpha h_t + 2\alpha m_t + \alpha^2 n_t,
\end{aligned} \tag{5.13}$$

where the last inequality follows from Assumption 7, and where we have

$$\begin{aligned}
n_t &:= \|\mathbf{g}(\boldsymbol{\theta}_t, o_t)\|^2, \\
h_t &:= \langle \mathbf{g}(\tilde{\boldsymbol{\theta}}_t, o_t) - \bar{\mathbf{g}}(\tilde{\boldsymbol{\theta}}_t), \tilde{\boldsymbol{\theta}}_t - \boldsymbol{\theta}^*\rangle, \\
m_t &:= \langle \mathbf{g}(\boldsymbol{\theta}_k, o_t) - \mathbf{g}(\tilde{\boldsymbol{\theta}}_t, o_t), \tilde{\boldsymbol{\theta}}_t - \boldsymbol{\theta}^*\rangle.
\end{aligned} \tag{5.14}$$

The term h_t is an error term related to Markovian sampling. Indeed, if the process o_t were sampled in an i.i.d. fashion, it would be $\mathbb{E}[h_t] = 0$. However, due to the correlated nature of o_t, this does not hold true, and, consequently, h_t requires careful care in the analysis. On the other hand, the term m_t is an error term related to the delayed nature of the SA algorithm under consideration. In absence of delays, it would be $m_t = 0$. To obtain the convergence bound for $\mathbb{E}\left[\tilde{r}_t^2\right]$, we provide bounds on $\mathbb{E}[h_t]$, m_t and $\|\mathbf{g}(\boldsymbol{\theta}_k, o_t)\|^2$. Obtaining bounds for these terms require some work, that we present as auxiliary Lemmas in the last part of this section. Providing the bound on $\mathbb{E}[h_t]$ is the most challenging part of the analysis, which also requires mixing time arguments. We provide such a bound in Lemma 5.3 - (iii). In order to provide a bound that is a function of $\|\mathbf{d}_t\|^2$ and $\tilde{r}_l^2, \|\mathbf{d}_l\|^2$, with $l = t - \bar{\tau}, ..., t-1$, we need to provide a novel analysis compared to the one used for the non-delayed SA under Markovian sampling in [135]. Specifically, we need to introduce a new way to bound the terms $\|\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t-\tau_{mix}}\|$ and $\|\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t-\tau_{mix}}\|^2$ and use the corresponding bounds accordingly when bounding $\mathbb{E}[h_t]$. The bound obtained

thanks to the auxiliary Lemmas has the following form:

$$\begin{aligned}\mathbb{E}\left[\tilde{r}_{t+1}^2\right] \leq (1 - 2\alpha\mu + 48\alpha^2 L^2 \bar{\tau})\mathbb{E}\left[\tilde{r}_t^2\right] + 128\alpha^2 L^2 \bar{\tau}\sigma^2 \\ + 4\alpha^2 L^2 \mathbb{E}\left[\|\mathbf{d}_t\|^2\right] + 20\alpha^2 L^2 \sum_{l=t-\bar{\tau}}^{t-1} \mathbb{E}\left[\|\mathbf{d}_l\|^2 + 2\tilde{r}_l^2\right] + 2\alpha\bar{B}_t,\end{aligned} \tag{5.15}$$

with

$$\bar{B}_t = \begin{cases} 111\sigma^2 & \text{if } 0 \leq t < \tau_{mix} \\ 0 & \text{otherwise} \end{cases}. \tag{5.16}$$

Starting from this bound, we analyze the weighted average $\sum_{t=0}^{T} w_t \mathbb{E}\left[\tilde{r}_t^2\right]$. Applying the weighted average to both sides of the bounds, applying some manipulations and with the proper choice of upper bound on the step size α, we are able to get the following inequality

$$\begin{aligned}\sum_{t=0}^{T} w_t \mathbb{E}\left[\tilde{r}_{t+1}^2\right] \leq (1 - 0.5\alpha\mu) \sum_{t=0}^{T} w_t \mathbb{E}\left[\tilde{r}_t^2\right] + 150 W_T \alpha^2 L^2 \bar{\tau}\sigma^2 \\ + 2W_{\tau_{mix}-1}\alpha\left(111\sigma^2\right).\end{aligned} \tag{5.17}$$

This last inequality is obtained thanks to Lemma 5.1, which we state in the next paragraph and which establishes a bound on $\sum_{t=0}^{T} w_t \mathbb{E}\left[\|\mathbf{d}_t\|^2\right]$. With some further manipulations and using the fact that $\mathbb{E}\left[r_t^2\right] \leq 2\mathbb{E}\left[\tilde{r}_t^2\right] + 2\mathbb{E}\left[\|\mathbf{d}_t\|^2\right]$, we can derive the final result.

Auxiliary Lemmas. Here, we present the main Lemmas needed to prove Theorem 5.3. We start with three bounds on $\|\mathbf{d}_t\|$, $\|\mathbf{d}_t\|^2$ and $\sum_{t=0}^{T} w_t \|\mathbf{d}_t\|^2$, as follows

Lemma 5.1. The three following inequalities hold:

$$(i) \qquad \|\mathbf{d}_t\| \leq \alpha\tau L\sigma + \alpha L \sum_{l=t-\tau}^{t-1} \|\boldsymbol{\theta}_l\|, \tag{5.18}$$

$$(ii) \qquad \|\mathbf{d}_t\|^2 \leq 2\alpha^2\tau^2 L^2\sigma^2 + 2\alpha^2\tau L^2 \sum_{l=t-\tau}^{t-1} \|\boldsymbol{\theta}_l\|^2, \tag{5.19}$$

$$(iii) \quad \sum_{t=0}^{T} w_t \|\mathbf{d}_t\|^2 \leq 4W_T\alpha^2\tau^2 L^2\sigma^2 + 16\alpha^2\tau^2 L^2 \sum_{t=0}^{T} w_t \|\tilde{\boldsymbol{\theta}}_t\|^2, \tag{5.20}$$

where (iii) holds for $\alpha \leq \frac{1}{4\tau L}$.

Part (iii) of this Lemma is key to obtain the bound in (5.17). In the next Lemma, we provide bounds on the terms $\|\tilde{\boldsymbol{\theta}}_t - \tilde{\boldsymbol{\theta}}_{t-\tau_{mix}}\|$ and $\|\tilde{\boldsymbol{\theta}}_t - \tilde{\boldsymbol{\theta}}_{t-\tau_{mix}}\|^2$.

Lemma 5.2. For any $t \geq \tau_{mix}$, we have

$$(i) \qquad \|\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}} - \tilde{\boldsymbol{\theta}}_k\| \leq L\alpha\sigma\tau_{mix} + L\alpha \sum_{l=t-\tau_{mix}}^{t-1} \|\boldsymbol{\theta}_l\| \tag{5.21}$$

$$(ii) \qquad \|\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}} - \tilde{\boldsymbol{\theta}}_k\|^2 \leq 2L^2\alpha^2\tau_{mix}^2\sigma^2 + 2L^2\alpha^2\tau_{mix} \sum_{l=t-\tau_{mix}}^{t-1} \|\boldsymbol{\theta}_l\|^2 \tag{5.22}$$

Note that this Lemma is a variation of Lemma 3 in [135], which is key to invoke mixing time arguments to get finite-time convergence bounds in existing non-delayed SA analysis. To obtain a bound in the form (5.15), we need to bound $\mathbb{E}[h_t]$ properly, for which, in turn, we need Lemma 5.2. Furthermore, note that, in contrast to [135], the bound is obtained for the sequence of *virtual iterates*. In the next Lemma, we provide bounds for the three key terms of the bound in (5.13), i.e., $\|\mathbf{g}(\boldsymbol{\theta}_k, o_t)\|^2$, m_t and $\mathbb{E}[h_t]$.

Lemma 5.3. For all $t \geq 0$, we have

$$(i) \quad n_t \leq 4L^2\|\mathbf{d}_t\|^2 + 8L^2\tilde{r}_t^2 + 10L^2\sigma^2, \tag{5.23}$$

$$(ii) \quad m_t \leq 6\alpha\tau L^2\sigma^2 + 3\alpha\tau L^2\tilde{r}_t^2 + 2\alpha L^2 \sum_{l=t-\tau}^{t-1} \left(\|\mathbf{d}_l\|^2 + 2\tilde{r}_l^2\right), \tag{5.24}$$

$$(iii) \quad \mathbb{E}[h_t] \leq \begin{cases} 111\sigma^2L^2, \text{ for } 0 \leq t \leq \tau_{mix} \\ 4\alpha\tau_{mix}L^2\left(8\sigma^2 + 3\mathbb{E}[\tilde{r}_t^2]\right) + 8\alpha L^2\sum_{l=t-\tau_{mix}}^{t-1}\mathbb{E}\left[\|\mathbf{d}_l\|^2 + 2\tilde{r}_l^2\right], \text{ for } t \geq \tau_{mix} \end{cases} \tag{5.25}$$

where (iii) holds for $\alpha \leq \frac{1}{36L^2\tau_{mix}}$.

The proof of this last Lemma relies on the bound on $\|\mathbf{d}_t\|$ established in Lemma 5.1. The proof of (iii) relies on the mixing properties of the Markov chain $\{o_t\}$ and on the bounds on $\|\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t-\tau_{mix}}\|$ and $\|\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t-\tau_{mix}}\|^2$ established in Lemma 5.2. Part (iii) is the key and most challenging part of the proof, which allows us to get to the bound in (5.15). Using this last Lemma, in combination with Lemma 5.1, we are able to get the bound in (5.17). The conclusion of the proof is enabled by using $\mathbb{E}[r_t^2] \leq 2\mathbb{E}[\tilde{r}_t^2] + 2\mathbb{E}[\|\mathbf{d}_t\|^2]$ and some further manipulations. The proofs of all the Lemmas in this section and the complete proof of Theorem 5.3 are available in Appendix C.1.

5.4 Stochastic Approximation with Time-Varying Delays

In this section, we present the first finite-time convergence analysis of the delayed SA update rule that was introduced in (5.4), with time-varying delays. Note that, by

Assumption 10, we can re-write (5.4) as:

$$\textbf{\textit{Delayed SA:}} \qquad \boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k + \alpha \mathbf{g}(\boldsymbol{\theta}_{t-\tau_t}, o_{t-\tau_t}), \quad \tau_t \leq \min\{t, \tau_{max}\} \tag{5.26}$$

The following theorem provides a convergence bound for the update rule in (5.26) and it is a major contribution of this work.

Theorem 5.4. Let $r_t := \|\boldsymbol{\theta}_t - \boldsymbol{\theta}^*\|$, $\tau' := 2\tau_{max} + \tau_{mix}$ and $\bar{\tau} := \max\{\tau_{mix}, \tau_{max}\}$. There exist absolute constants $C, C', C'' \geq 2$ such that the iterates generated by the update rule (5.26), for $T \geq \tau'$ and $\alpha \leq \frac{\mu}{CL^2\bar{\tau}}$, satisfy

$$\mathbb{E}\left[r_T^2\right] \leq \exp\left(-2\alpha\mu T\right) 2B + \frac{\alpha C' L^2 (\tau_{mix} + \tau_{max}) B}{\mu}, \tag{5.27}$$

with $B = C''\sigma^2$. Setting $\alpha = \frac{\mu}{CL^2\bar{\tau}}$, we get

$$\mathbb{E}\left[r_T^2\right] \leq \exp\left(-\frac{2\mu^2 T}{CL^2\bar{\tau}}\right) 2B + \frac{2C'B}{C}. \tag{5.28}$$

Main Takeaways: There are many relevant takeaways from this Theorem. We focus on the convergence bound in (5.28), i.e., the case in which the step size matches its upper bound. We note that, (i) with a choice of step size inversely proportional to $\tau_{mix} + \tau_{max}$, the *Delayed SA* update rule (5.26) converges exponentially fast in mean square to a ball around $\boldsymbol{\theta}^*$ whose radius is proportional to the "variance" term $B = 9\sigma^2$, which is consistent with the non-delayed case; (ii) the exponent of convergence gets scaled down by a factor $\bar{\tau} = \max\{\tau_{max}, \tau_{mix}\}$. Remarkably, the dependence on both τ_{max} and τ_{mix} is optimal. With respect to the dependence on the delay sequence, early works on SA with delayed updates and i.i.d. sampling - specifically, gradient descent on a strongly convex smooth cost function - and with time-varying delays [12], [57], [64] showed an exponential convergence with a convergence exponent that gets scaled down by a factor proportional to τ_{max}^2 (see, e.g., [64, Theorem 3.3]). Recent works considering a constant delay τ have shown that the same iterative algorithm can achieve a better convergence rate with a convergence exponent that gets scaled down by a factor proportional to τ [137]. This type of dependence has also been shown to be tight for the same configuration [10]. The works in [10], [137] claim that their analysis can be extended to time-varying delay sequences, with the dependence on τ being replaced by a dependence on τ_{max}. However, they do not provide an explicit derivation of this extension. In Section 5.3, we have shown that, in the constant delay case, an analysis similar to [137] provides the same dependence on τ also for SA under Markovian sampling.

In the analysis, we outlined the critical points that make the extension of the same type of analysis in [137] to the time-varying delays case difficult. In Theorem 5.4, with a different type of analysis, we provide the first convergence guarantee for SA under delayed updates and time-varying delays with an explicit derivation of optimal dependency on the delay sequence. Remarkably, compared to [10], [137], our analysis is done on the much more challenging case of SA under Markovian sampling. Notably, in this configuration, our analysis jointly provides also an optimal dependency on the mixing time τ_{mix}. Indeed, note that this dependence on τ_{mix} of the convergence exponent of the rate of exponential convergence of SA under Markovian sampling has been shown to be minimax optimal in the case without delays [105].

Outline of the Analysis. We now provide insights on the key steps in the analysis. To analyze the convergence of the update rule in (5.26), we consider the delay as a perturbation to the original update. We define the error at iteration t as follows,

$$\mathbf{e}_t \triangleq \mathbf{g}(\boldsymbol{\theta}_k, o_t) - \mathbf{g}(\boldsymbol{\theta}_{t-\tau_t}, o_{t-\tau_t}), \tag{5.29}$$

which we use to rewrite the update rule in (5.26) as

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k + \alpha \mathbf{g}(\boldsymbol{\theta}_k, o_t) - \alpha \mathbf{e}_t. \tag{5.30}$$

We examine $\|\boldsymbol{\theta}_{k+1} - \boldsymbol{\theta}^*\|^2$ using (5.30), which leads us to

$$\|\boldsymbol{\theta}_{k+1} - \boldsymbol{\theta}^*\|^2 = J_{t,1} + \alpha^2 J_{t,2} - 2\alpha J_{t,3}. \tag{5.31}$$

with

$$\begin{aligned} J_{t,1} &:= \|\boldsymbol{\theta}_k - \boldsymbol{\theta}^* + \alpha \mathbf{g}(\boldsymbol{\theta}_k, o_t)\|^2, \\ J_{t,2} &:= \|\mathbf{e}_t\|^2, \\ J_{t,3} &:= \langle \mathbf{e}_t, \boldsymbol{\theta}_k - \boldsymbol{\theta}^* \rangle + \alpha \langle \mathbf{e}_t, \mathbf{g}(\boldsymbol{\theta}_k, o_t) \rangle. \end{aligned} \tag{5.32}$$

Note that the presence of $J_{t,2}$ and $J_{t,3}$ in (5.31) is a consequence of the delay and it would not occur in the case of non-delayed updates. The convergence analysis is built up providing bounds on the expectation of the three terms defined in (5.32).

Main challenges. We now comment on some of the main challenges of the analysis. First, the term $J_{t,1}$ cannot be bounded with the methods used in [135] for non-delayed SA under Markovian sampling. Indeed, Lemma 3 in [135], which is key to prove the finite-time linear convergence rate invoking properties of the geometric mixing of the Markov chain, is not valid when using the delayed operator $\mathbf{g}(\boldsymbol{\theta}_{k-\tau}, o_{t-\tau_t})$. To see why this is the case, note that Lemma 3 in [135] establishes a bound on $\|\boldsymbol{\theta}_k - \boldsymbol{\theta}_{t-\tau}\|$, for any

$0 \leq \tau \leq t$, of the following form

$$\|\boldsymbol{\theta}_k - \boldsymbol{\theta}_{t-\tau}\| \leq O(\alpha\tau)(\|\boldsymbol{\theta}_k\| + \sigma), \tag{5.33}$$

which, however, does not hold true when using the delayed operator $\mathbf{g}(\boldsymbol{\theta}_{k-\tau}, o_{t-\tau_t})$. Indeed, the first key step in proving (5.33) is using the fact that $\|\boldsymbol{\theta}_{k+1} - \boldsymbol{\theta}_k\| \leq O(\alpha)(\|\boldsymbol{\theta}_k\| + \sigma)$, which is not true for the delayed case, where we can only get $\|\boldsymbol{\theta}_{k+1} - \boldsymbol{\theta}_k\| \leq O(\alpha)(\|\boldsymbol{\theta}_{k-\tau}\| + \sigma)$ by using the bound $\|\mathbf{g}(\boldsymbol{\theta}_{k-\tau}, o_{t-\tau_t})\| \leq O(\alpha)(\|\boldsymbol{\theta}_{k-\tau}\| + \sigma)$ on the delayed operator. This fact, that prevents us from applying the analysis of Lemma 3 in [135], forces us to develop a different strategy and to prove a more general result, the statement of which we provide in Lemma 5.4. This new Lemma enables us to deal with $\|\boldsymbol{\theta}_k - \boldsymbol{\theta}_{t-\tau}\|$ in an functional way with respect to the proof of finite-time rates for the considered delayed SA algorithm. Second, note that bounding the term $\langle \mathbf{e}_t, \boldsymbol{\theta}_k - \boldsymbol{\theta}^* \rangle$ is much more challenging compared to the i.i.d. sampling setting considered in the optimization literature with delays [11], [36], [78], [153]. This difficulty arises due to the statistical correlation among the terms in $\mathbf{g}(\boldsymbol{\theta}_k, o_t) - \mathbf{g}(\boldsymbol{\theta}_{t-\tau_t}, o_{t-\tau_t})$ and $\boldsymbol{\theta}_k - \boldsymbol{\theta}^*$, which calls for a more careful analysis. Indeed, the fact that in general, for correlated Markovian samples, $\mathbb{E}[\mathbf{g}(\boldsymbol{\theta}_k, o_t)] \neq \bar{\mathbf{g}}(\boldsymbol{\theta}_k)$, forces us to invoke mixing time arguments to bound this cross term and get the desired finite-time rate, as it is typically done for SA under Markovian sampling. However, the presence of the delay in the operator $\mathbf{g}(\boldsymbol{\theta}_{k-\tau}, o_{t-\tau_t})$ introduces further statistically correlated iterates $\boldsymbol{\theta}_{k-\tau}$ and observations $o_{t-\tau_t}$ in the analysis, whose interplay needs to be carefully taken care of. To do so, we provide a novel analysis, whose results are stated in Lemma 5.5. This analysis is enabled also thanks to the new bound stated in Lemma 5.4 which generalizes Lemma 3 in [135] and which we present next. Another challenge is presented by the presence of time-varying delays. As noted in Section 5.3, with time-varying delays some of the steps of the analysis for the constant delay configuration are not easy to extend to the time-varying delays case. To deal with this challenge, we provide a novel analysis, which is based on the uniform boundedness of the iterates generated by the iterative update rule (5.26), when the constant step-size α is small enough. Specifically, as for the statement of Theorem 5.4, we require $\alpha \leq \frac{\mu}{CL^2\bar{\tau}}$, where C is some absolute constant. The key result establishing the uniform boundedness of the iterates is provided in Lemma 5.6.

Auxiliary Lemmas. We now introduce three Lemmas that are fundamental to prove Theorem 5.4. We start with a result that provides bounds in expectation on quantities of the form $\|\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t-\tau}\|^2$. This result, as mentioned above, represents a generalization of Lemma 3 in [135], being suitable to be applied for the analysis of the delayed case.

Recalling that $\tau' = 2\tau_{max} + \tau_{mix}$, define

$$r_{t,2} := \max_{t-\tau' \le l \le t} \mathbb{E}\left[r_l^2\right]. \tag{5.34}$$

Lemma 5.4. For any $t \ge \tau_{mix}$, we have

$$(i) \qquad \mathbb{E}\left[\|\boldsymbol{\theta}_k - \boldsymbol{\theta}_{t-\tau_{mix}}\|^2\right] \le 2\alpha^2\tau_{mix}^2 L^2(2r_{t,2} + 3\sigma^2).$$

Similarly, for any $t \ge 0$ and $\tau_t \le t$,

$$(ii) \qquad \mathbb{E}\left[\|\boldsymbol{\theta}_k - \boldsymbol{\theta}_{t-\tau_t}\|^2\right] \le 2\alpha^2\tau_{max}^2 L^2(2r_{t,2} + 3\sigma^2).$$

The above Lemma plays a critical role in providing the finite-time rate established in Theorem 5.4. Specifically, the Lemma establishes a bound that enables us to relate $\boldsymbol{\theta}_t$, $\boldsymbol{\theta}_{t-\tau_t}$, and $\boldsymbol{\theta}_{t-\tau_{mix}}$, which is key to establish the finite-time linear rate for (5.26), which is the main result of this section and one of the major contributions of this work. Exploiting these bounds, we can provide bounds on $\mathbb{E}[J_{t,1}]$, $\mathbb{E}[J_{t,2}]$, and $\mathbb{E}[J_{t,3}]$ in terms of $r_{t,2}$, which we do next, in Lemma 5.5.

Lemma 5.5. Let $t \ge \tau' = 2\tau_{max} + \tau_{mix}$, then

$$(i) \quad \mathbb{E}[J_{t,1}] \le (1 - 2\alpha\mu)\mathbb{E}\left[r_t^2\right] + 28\alpha^2\tau_{mix}L^2 r_{t,2} + 34\alpha^2\tau_{mix}L^2\sigma^2,$$

$$(ii) \quad \mathbb{E}[J_{t,2}] \le 8L^2(2r_{t,2} + 3\sigma^2),$$

$$(iii) \quad \mathbb{E}[J_{t,3}] \le 28\alpha L^2(\tau_{mix} + \tau_{max})(r_{t,2} + \sigma^2).$$

The bounds provided in the above Lemma play a central role in the convergence analysis of (5.26). The most challenging part of the proof is part (iii), in which mixing time arguments need to be carefully applied to deal with the joint appearance of the statistically correlated terms $\boldsymbol{\theta}_{k-\tau}$, $o_{t-\tau_t}$ and $\boldsymbol{\theta}_{t-\tau_{mix}}$. Using the bounds established in the above Lemma, we can rewrite equation (5.31) as

$$\begin{aligned}\mathbb{E}\left[r_{t+1}^2\right] &= \mathbb{E}\left[\|\boldsymbol{\theta}_{k+1} - \boldsymbol{\theta}^*\|^2\right] \\ &= \mathbb{E}[J_{t,1}] + \alpha^2\mathbb{E}[J_{t,2}] - 2\alpha\mathbb{E}[J_{t,3}] \\ &\le (1 - 2\alpha\mu)\mathbb{E}\left[r_t^2\right] + 98\alpha^2L^2(\tau_{mix} + \tau_{max})(r_{t,2} + \sigma^2).\end{aligned} \tag{5.35}$$

The above bound is key to get the finite-time convergence rate of Theorem 5.4. The

crucial step to get the optimal dependence on the delay sequence is to use this bound to prove that, for α sufficiently small, the iterates generated by (5.26) are uniformly bounded by $B = 9\sigma^2$. The statement of this key result is provided in the next Lemma, and its proof is obtained by induction.

Lemma 5.6. For all $t \geq 0$, there exists an absolute constant C such that for $\alpha \leq \frac{\mu}{C\bar{\tau}L^2}$,

$$\mathbb{E}\left[r_t^2\right] \leq B, \qquad \text{with } B = 9\sigma^2. \tag{5.36}$$

By applying this Lemma to the recursion illustrated in (5.35), the result stated in Theorem 5.4 follows in few simple steps. The complete proofs of all Lemmas and of Theorem 5.4 is provided in Appendix C.2.

5.5 Related Publications and Presentation at Conferences

Part of the work presented in this chapter has been submitted to the 27th International Conference on Artificial Intelligence and Statistics (AISTATS, 2024).

6

Conclusions and Future Work

In this thesis, we have presented algorithmic and theoretical advancements in FL. The focus, in particular, has been on (i) the design and analysis of super-linearly convergent algorithms, and (ii) the finite-time convergence analysis of federated reinforcement learning algorithms. With respect to super-linear convergence in FL, we have designed and analysed SHED, a Newton-type algorithm for FL based on agents' Hessians eigendecomposition, extensively described in Chapter 2. We extended this algorithm to a version exploiting quantization, Q-SHED, described in Chapter 3, in which the agents carefully allocate the available bits to quantize the eigenvectors used to perform approximate Newton-type updates. SHED provides state-of-the-art theoretical guarantees and empirical performance, while Q-SHED, at the cost of an additional computational burden, can boost the empirical performance of SHED. With respect to (ii), i.e., federated reinforcement learning (FRL), we have provided finite-time explicit convergence bounds for instances of RL under communication constraints, while establishing the beneficial effects of multi-agent cooperation by means of proving N-fold linear convergence speedups (with N the number of agents) under different communication settings and models.

Future work include the following:

- For second-order methods in FL, in particular for the proposed algorithms (SHED and Q-SHED), future efforts include the extension to non-convex cost functions, e.g., via cubic regularization; efforts to reduce the computational burden at the agents, for example by means of analysing sub-sampled Newton methods in the FL setting (see [53] for examples of sub-sampled Newton methods) or by means of using approximated versions of singular value decomposition. Similar efforts to reduce the computational burden of second-order algorithms in FL while maintaining the improved convergence features have been considered, e.g., in [2]. Other potential future works include an improved understanding of the convergence properties of

Q-SHED and the potential application, in practice, of second-order methods in deep learning: similar research works have been recently proposed, for example, see [111], [112]

- For Federated Reinforcement Learning (FRL), the main focus of thesis has been on analysing the convergence of vanilla algorithms under communication constraints common in many applications. Future works include the design and analysis of novel distributed algorithms for FRL, and the evaluation of these algorithms on real-world RL tasks. The focus of future works should be on validating and engineering the beneficial effects of cooperation-via-communication in FRL. Theoretical questions to be addressed also include the possibility of obtaining tight convergence bounds (relative to the dependence on the mixing time) for FRL even when agents perform multiple local steps, updating their local parameter communicating with the central servers/other nodes only at intermittent iterations. Indeed, existing studies (see [77], [143]) only provide suboptimal dependencies on the mixing time. Other research directions include integrating the design of communication schemes with adaptive algorithms relative to the number of local steps performed by the agents, and relative to asynchronous configurations with partial participation.

A

Appendix: Proofs of Chapter 2 and additional experiments

A.1 Proof of Theorem 2.1

From (2.4), writing $\mathbf{H}_{LS} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T$ and $\hat{\mathbf{H}}_t = \mathbf{V}\hat{\mathbf{\Lambda}}_t\mathbf{V}^T$, with $\hat{\mathbf{\Lambda}}_t = diag(\lambda_1, ..., \lambda_{q_t}, \rho_t, ..., \rho_t)$, define $\boldsymbol{\theta}^{t+1}_{\rho_t,\eta_t} := \boldsymbol{\theta}^t - \eta_t\hat{\mathbf{H}}_t^{-1}\mathbf{g}_t$. Recalling that $\mathbf{g}_t = \mathbf{H}_{LS}(\boldsymbol{\theta}^t - \boldsymbol{\theta}^*)$, we have:

$$\begin{aligned}\boldsymbol{\theta}^{t+1}_{\rho_t,\eta_t} - \boldsymbol{\theta}^* = \mathbf{A}_t(\boldsymbol{\theta}^t - \boldsymbol{\theta}^*) &= (\mathbf{I} - \eta_t\hat{\mathbf{H}}_t^{-1}\mathbf{H}_{LS})(\boldsymbol{\theta}^t - \boldsymbol{\theta}^*) \\ &= \mathbf{V}(\mathbf{I} - \eta_t\hat{\mathbf{\Lambda}}_t^{-1}\mathbf{\Lambda})\mathbf{V}^T(\boldsymbol{\theta}^t - \boldsymbol{\theta}^*).\end{aligned} \tag{A.1}$$

For some given $q_t \in \{1, ..., n\}$, r_t is a function of two tunable parameters, i.e., the tuple (η_t, ρ_t). We now prove that r_t^* can be achieved if and only if $\rho_t \in [\lambda_n, \lambda_{q_t+1}]$. The convergence rate is determined by the eigenvalue of $(\mathbf{I} - \eta_t\hat{\mathbf{\Lambda}}_t^{-1}\mathbf{\Lambda})$ with the greatest absolute value. First, we show that $\rho_t \notin [\lambda_n, \lambda_{q_t+1}]$ implies $r_t > r_t^*$, then we show that, if $\rho_t \in [\lambda_n, \lambda_{q_t+1}]$, there exists an optimal η_t^* for which r_t^* is achieved. If $\rho_t < \lambda_n$, the choice of η_t minimizing the maximum absolute value of $(\mathbf{I} - \eta_t\hat{\mathbf{\Lambda}}_t^{-1}\mathbf{\Lambda})$ is the solution of $|1 - \eta_t| = |1 - \eta_t\lambda_{q_t+1}/\rho_t|$, which is $\eta_t^* = 2\rho_t/(\rho_t + \lambda_{q_t+1})$. The corresponding convergence factor is $1 - \eta_t^* > r_t^*$. Similarly, if $\rho_t > \lambda_{q_t+1}$, one gets $\eta_t^* = 2\rho_t/(\rho_t + \lambda_n)$ and convergence factor equal to $1 - 2\lambda_n/(\rho_t + \lambda_n) > r_t^*$. If $\rho_t \in [\lambda_n, \lambda_{q_t+1}]$, the best η_t is such that $|1 - \eta_t\lambda_n/\rho_t| = |1 - \eta_t\lambda_{q_t+1}/\rho_t|$, whose solution is

$$\eta_t^* = \frac{2\rho_t}{\lambda_{q_t+1} + \lambda_n} \tag{A.2}$$

and the achieved factor is $r_t = 1 - \eta_t^*\lambda_n/\rho_t = 1 - \lambda_n/\rho_t^* = r_t^*$. We see that the definition of the set $\mathcal{S}^*$ immediately follows.

A.2 Proof of Corollary 2.1

Define $\mathbf{B}_t^* := (\mathbf{I} - (\hat{\mathbf{\Lambda}}_t(\rho_t^*, q_t))^{-1}\mathbf{\Lambda}) = \text{diag}(0, ..., 0, 1-\lambda_{q_t+1}/\rho_t^*, ..., 1-\lambda_n/\rho_t^*)$. For $\rho_t \neq \rho_t^*$, with $\rho_t \in [\lambda_n, \lambda_{q_t+1}]$, define $\mathbf{B}_t := (\mathbf{I} - \eta_t^*(\hat{\mathbf{\Lambda}}_t(\rho_t, q_t))^{-1}\mathbf{\Lambda}) = \text{diag}(1-\eta_t^*, ..., 1-\eta_t^*, 1-\lambda_{q_t+1}/\rho_t^*, ..., 1-\lambda_n/\rho_t^*) = \mathbf{B}_t^* + \delta\mathbf{B}_t$, with $\delta\mathbf{B}_t = \text{diag}(1-\eta_t^*, ..., 1-\eta_t^*, 0, ..., 0)$, where η_t^* is defined in (A.2) and ρ_t^* in (2.8). Now define $\mathbf{z}^t := \mathbf{V}^T(\boldsymbol{\theta}^t - \boldsymbol{\theta}^*)$ and $\mathbf{z}_{\rho_t,\eta_t}^{t+1} := \mathbf{V}^T(\boldsymbol{\theta}_{\rho_t,\eta_t}^{t+1} - \boldsymbol{\theta}^*)$, where $(\boldsymbol{\theta}_{\rho_t,\eta_t}^{t+1} - \boldsymbol{\theta}^*)$ is defined in (A.1). We have

$$\|\boldsymbol{\theta}_{\rho_t^*,1}^{t+1} - \boldsymbol{\theta}^*\|^2 = \|\mathbf{z}_{\rho_t^*,1}^{t+1}\|^2 = \|\mathbf{B}_t^*\mathbf{z}^t\|^2,$$

$$\|\boldsymbol{\theta}_{\rho_t,\eta_t^*}^{t+1} - \boldsymbol{\theta}^*\|^2 = \|\mathbf{B}_t\mathbf{z}^t\|^2 = \|\mathbf{B}_t^*\mathbf{z}^t\|^2 + \|\delta\mathbf{B}_t\mathbf{z}^t\|^2,$$

because the cross term is $2(\mathbf{B}_t^*\mathbf{z}^t)^T(\delta\mathbf{B}_t\mathbf{z}^t) = 0$. We see that, for any t and for any $\boldsymbol{\theta}^t$,

$$\|\boldsymbol{\theta}_{\rho_t^*,1}^{t+1} - \boldsymbol{\theta}^*\| \leq \|\boldsymbol{\theta}_{\rho_t,\eta_t^*}^{t+1} - \boldsymbol{\theta}^*\|$$

A.3 Proof of Theorem 2.2

Fix $\eta_t = 1$ in (2.5),

$$\boldsymbol{\theta}^{t+1} - \boldsymbol{\theta}^* = \mathbf{A}_t(\boldsymbol{\theta}^t - \boldsymbol{\theta}^*) = (\mathbf{I} - \hat{\mathbf{H}}_t^{-1}\mathbf{H}_{LS})(\boldsymbol{\theta}^t - \boldsymbol{\theta}^*).$$

We have that

$$\begin{aligned}\|\boldsymbol{\theta}^{t+1} - \boldsymbol{\theta}^*\| &\leq \|\mathbf{I} - \hat{\mathbf{H}}_t^{-1}\mathbf{H}_{LS}\|\|\boldsymbol{\theta}^t - \boldsymbol{\theta}^*\| \\ &\leq \|\hat{\mathbf{H}}_t^{-1}\|\|\hat{\mathbf{H}}_t - \mathbf{H}_{LS}\|\|\boldsymbol{\theta}^t - \boldsymbol{\theta}^*\| \\ &\leq \frac{(\bar{\rho}_t - \bar{\lambda}_n)}{\bar{\rho}_t}\|\boldsymbol{\theta}^t - \boldsymbol{\theta}^*\|\end{aligned}$$

The last inequality follows from two inequalities: (i) $\|\hat{\mathbf{H}}_t^{-1}\| \leq 1/\bar{\rho}_t$ and (ii) $\|\hat{\mathbf{H}}_t - \mathbf{H}_{LS}\| \leq \bar{\rho}_t - \bar{\lambda}_n$.
(i) holds because $\|\hat{\mathbf{H}}_t^{-1}\| = (\lambda_{min}(\hat{\mathbf{H}}_t))^{-1}$, and $\lambda_{min}(\hat{\mathbf{H}}_t) \geq \bar{\rho}_t$, thus implying $\|\hat{\mathbf{H}}_t^{-1}\| \leq 1/\bar{\rho}_t$.
(ii) follows recalling that $\mathbf{H}_{LS} = \frac{1}{M}\sum_{i=1}^M \mathbf{H}_{LS}^{(i)}$, with $\mathbf{H}_{LS}^{(i)}$ the local Hessian at agent i. We have

$$\|\hat{\mathbf{H}}_t - \mathbf{H}_{LS}\| = \frac{1}{M}\|\sum_{i=1}^M(\hat{\mathbf{H}}_t^{(i)} - \mathbf{H}_{LS}^{(i)})\| \leq \frac{1}{M}\sum_{i=1}^M\|\hat{\mathbf{H}}_t^{(i)} - \mathbf{H}_{LS}^{(i)}\|.$$

Being $\hat{\mathbf{H}}_t^{(i)} - \mathbf{H}_{LS}^{(i)}$ symmetric, it holds that

$$\|\hat{\mathbf{H}}_t^{(i)} - \mathbf{H}_{LS}^{(i)}\| = \max_j |\lambda_j(\hat{\mathbf{H}}_t^{(i)} - \mathbf{H}_{LS}^{(i)})| = \rho_t^{(i)} - \lambda_n^{(i)},$$

where the last equality holds because

$$\hat{\mathbf{H}}_t^{(i)} - \mathbf{H}_{LS}^{(i)} = \mathbf{V}^{(i)}(\hat{\mathbf{\Lambda}}_t^{(i)} - \mathbf{\Lambda}_t^{(i)})\mathbf{V}^{(i)T} \tag{A.3}$$

where

$$\begin{aligned} \hat{\mathbf{\Lambda}}_t^{(i)} &= diag(\lambda_1^{(i)}, ..., \lambda_{q_t^{(i)}}^{(i)}, \rho_t^{(i)}, ..., \rho_t^{(i)}), \\ \mathbf{\Lambda}_t^{(i)} &= diag(\lambda_1^{(i)}, ..., \lambda_{q_t^{(i)}}^{(i)}, \lambda_{q_t^{(i)}+1}^{(i)}, ...\lambda_n^{(i)}), \end{aligned} \tag{A.4}$$

and because $\rho_t^{(i)} = (\lambda_{q_t^{(i)}+1}^{(i)} + \lambda_n^{(i)})/2$.

A.4 Proof of Lemma 2.1

The quadratic cost in $\boldsymbol{\theta}^t$ can be written as

$$f(\boldsymbol{\theta}^t) = f(\boldsymbol{\theta}^*) + \bar{f}(\boldsymbol{\theta}^t).$$

with $\bar{f}(\boldsymbol{\theta}^t) = \frac{1}{2}(\boldsymbol{\theta}^t - \boldsymbol{\theta}^*)^T\mathbf{H}_{LS}(\boldsymbol{\theta}^t - \boldsymbol{\theta}^*)$. Given that $f(\boldsymbol{\theta}^*)$ does not depend on $\boldsymbol{\theta}^t$, we can focus on $\bar{f}(\boldsymbol{\theta}^t)$.
We have that

$$\begin{aligned} \bar{f}(\boldsymbol{\theta}^t - \eta_t\mathbf{p}_t) &= \frac{1}{2}(\boldsymbol{\theta}^t - \eta_t\mathbf{p}_t - \boldsymbol{\theta}^*)^T\mathbf{H}_{LS}(\boldsymbol{\theta}^t - \eta_t\mathbf{p}_t - \boldsymbol{\theta}^*) \\ &\overset{(1)}{=} \bar{f}(\boldsymbol{\theta}^t) + \frac{1}{2}\eta_t^2\mathbf{p}_t^T\mathbf{H}_{LS}\mathbf{p}_t - \eta_t\mathbf{p}_t^T\mathbf{g}_t \\ &\overset{(2)}{=} \bar{f}(\boldsymbol{\theta}^t) - \eta_t\mathbf{p}_t^T(\hat{\mathbf{H}}_t - \eta_t\frac{\mathbf{H}_{LS}}{2})\mathbf{p}_t \end{aligned} \tag{A.5}$$

where we have used identity $\mathbf{H}_{LS}(\boldsymbol{\theta}_t - \boldsymbol{\theta}^*) = \mathbf{g}_t$ and the fact that $\mathbf{p}_t^T\mathbf{g}_t = \mathbf{p}_t^T\hat{\mathbf{H}}_t\hat{\mathbf{H}}_t^{-1}\mathbf{g}_t = \mathbf{p}_t^T\hat{\mathbf{H}}_t\mathbf{p}_t$ to get equality (1) and (2), respectively. We see that if

$$\hat{\mathbf{H}}_t - \eta_t\mathbf{H}_{LS}/2 \geq \hat{\mathbf{H}}_t/2, \tag{A.6}$$

then Armijo-Goldstein condition (2.13) is satisfied. Indeed, in that case,

$$\begin{aligned}\bar{f}(\boldsymbol{\theta}^t) - \eta_t \mathbf{p}_t^T(\hat{\mathbf{H}}_t - \eta_t \frac{\mathbf{H}_{LS}}{2})\mathbf{p}_t &\leq \bar{f}(\boldsymbol{\theta}^t) - \frac{1}{2}\eta_t \mathbf{p}_t^T \hat{\mathbf{H}}_t \mathbf{p}_t \\ &\leq \bar{f}(\boldsymbol{\theta}^t) - \alpha \eta_t \mathbf{p}_t^T \mathbf{g}_t.\end{aligned}$$

So, we need to find a sufficient condition on η_t for (A.6) to be true. We see that (A.6) is equivalent to $\hat{\mathbf{H}}_t - \eta_t \mathbf{H}_{LS} \geq 0$. We have $\hat{\mathbf{H}}_t - \eta_t \mathbf{H}_{LS} = \frac{1}{M}\sum_{i=1}^{M} \mathbf{V}^{(i)}(\hat{\mathbf{\Lambda}}_t^{(i)} - \eta_t \mathbf{\Lambda}_t^{(i)})\mathbf{V}^{(i)T}$, and, $\forall i$, all the elements of the diagonal matrix $\hat{\mathbf{\Lambda}}_t^{(i)} - \eta_t \mathbf{\Lambda}_t^{(i)}$ are positive if $\eta_t \leq \rho_t^{(i)} / \lambda_{q_t^{(i)}+1}^{(i)}$ (see Eq. (A.4)), and we see that the choice (2.14) provides a sufficient condition to satisfy (A.6), from which we can conclude.

A.5 Proof of Lemma 2.2

We see that $a_* \leq \bar{a}$ from definition (2.15), because for each k, $\|\mathbf{A}_k\| \leq c_k = (1 - \bar{\lambda}_n/\bar{\rho}_k)$. Now, we show that, if $q_t^{(i)} = q_{t-1}^{(i)} + 1$, $\forall i, t$, then $\bar{a} = \bar{a}_T$. To show this latter inequality, let us consider the logarithm of the considered values, so we prove $\log \bar{a} = \log \bar{a}_T$. Indeed, if $q_t^{(i)} = q_{t-1}^{(i)} + 1$ and applying Algorithm 3, c_k is a periodic sequence of the index k ($c_{k+T} = c_k$):

$$\begin{aligned}\log \bar{a} &= \limsup_t \frac{1}{t}\sum_{k=1}^{t} \log c_k \\ &= \limsup_R \frac{1}{RT + T'}(R\sum_{k=1}^{T} \log c_k + \sum_{k=1}^{T'} \log c_k) \\ &= \frac{1}{T}\sum_{k=1}^{T} \log c_k = \log \bar{a}_T\end{aligned} \tag{A.7}$$

where $R = \lfloor t/T \rfloor$ and $T' = t - RT$.

A.6 Proof of Theorem 2.4

First, we need to prove the following Lemma:

Lemma A.1. If $\|\mathbf{g}_t\| > \omega > 0$, for $\hat{\mathbf{H}}_t$ defined in (2.18), $\mathbf{p}_t = \hat{\mathbf{H}}_t^{-1}\mathbf{g}_t$ there are $\gamma_t, \eta_t > 0$ such that

$$f(\boldsymbol{\theta}^t - \eta_t \mathbf{p}_t) \leq f(\boldsymbol{\theta}^t) - \gamma_t, \tag{A.8}$$

in particular, for a backtracking line search with parameters $\alpha \in (0, 0.5), \beta \in (0, 1)$, it holds:

$$\gamma_t = \alpha\beta \frac{\bar{\rho}_t}{K^2}\omega^2 \tag{A.9}$$

Proof. The proof is the same as the one provided in [24] for the damped Newton phase (page 489-490), with the difference that the "Newton decrement", that we denote by σ_t, here is $\sigma_t^2 := \mathbf{g}_t^T \mathbf{p}_t = \mathbf{g}_t^T \hat{\mathbf{H}}_t^{-1} \mathbf{g}_t = \mathbf{p}_t^T \hat{\mathbf{H}}_t \mathbf{p}_t$. Furthermore, the property $\hat{\mathbf{H}}_t \geq \bar{\rho}_t \mathbf{I}, \ \forall t$ is used in place of strong convexity. □

Note that, because of Assumption 3, it always holds that $\rho_t^{(i)} > 0, \quad \forall i, \forall t$, and this implies $\bar{\rho}_t > 0, \forall t$. When $\bar{\rho}_t > 0, \forall t$, Lemma A.1 implies $\|\mathbf{g}_t\| \to 0$. Indeed, if not, there would be some $\epsilon > 0$ such that $\|\mathbf{g}_t\| > \epsilon, \ \forall t$. But then (A.8) immediately implies that $f(\boldsymbol{\theta}^t) \to -\infty$, that contradicts the strong convexity hypothesis. By strong convexity and differentiability of f, $g(\boldsymbol{\theta}) = \nabla f(\boldsymbol{\theta}) = 0 \implies \boldsymbol{\theta} = \boldsymbol{\theta}^*$.

A.7 Proof of Theorem 2.5

The beginning of the proof follows from the proof of Lemma 3.1 in [53], (see page 18). In particular, we can get the same inequality as (A.1) in [53] (the $\mathbf{Q}^t$ here is $\hat{\mathbf{H}}_t$) with the difference that since we are not sub-sampling, we have (using the notation of [53]) $S = [n]$. The following inequality holds:

$$\begin{aligned}\|\boldsymbol{\theta}^{t+1} - \boldsymbol{\theta}^*\| \leq \|\boldsymbol{\theta}^t - \boldsymbol{\theta}^*\| \|I - \eta_t \hat{\mathbf{H}}_t^{-1} \mathbf{H}(\boldsymbol{\theta}^t)\| \\ + \eta_t L \frac{\|\hat{\mathbf{H}}_t^{-1}\|}{2} \|\boldsymbol{\theta}^t - \boldsymbol{\theta}^*\|^2.\end{aligned}$$

Note that, as we have shown in the proof of Theorem 2.2, it holds $\|\hat{\mathbf{H}}_t^{-1}\| \leq 1/\bar{\rho}_t$. We now focus on the first part of the right hand side of the inequality:

$$\begin{aligned}\|I - \eta_t \hat{\mathbf{H}}_t^{-1} \mathbf{H}(\boldsymbol{\theta}^t)\| &\leq \|\hat{\mathbf{H}}_t^{-1}\| (\|\hat{\mathbf{H}}_t - \mathbf{H}(\boldsymbol{\theta}^t)\| \\ &\quad + (1 - \eta_t) \|\mathbf{H}(\boldsymbol{\theta}^t)\|) \\ &\leq \frac{1}{\bar{\rho}_t} \|\hat{\mathbf{H}}_t - \mathbf{H}(\boldsymbol{\theta}^t)\| + \frac{(1 - \eta_t)}{\bar{\rho}_t} \|\mathbf{H}(\boldsymbol{\theta}^t)\|\end{aligned}$$

and focusing now on the first term of the right hand side of the last inequality

$$\begin{aligned}&\frac{1}{\bar{\rho}_t} (\|\hat{\mathbf{H}}_t - \mathbf{H}(\boldsymbol{\theta}^{k_t})\| + \|\mathbf{H}(\boldsymbol{\theta}^t) - \mathbf{H}(\boldsymbol{\theta}^{k_t})\|) \\ \leq &\frac{1}{\bar{\rho}_t} (\frac{1}{M} \sum_{i=1}^{M} \|\hat{\mathbf{H}}_t^{(i)}(\boldsymbol{\theta}^{k_t}) - \mathbf{H}^{(i)}(\boldsymbol{\theta}^{k_t})\| + L \|\boldsymbol{\theta}^{k_t} - \boldsymbol{\theta}^t\|) \\ = &1 - \frac{\bar{\lambda}_{n,t}}{\bar{\rho}_t} + \frac{L}{\bar{\rho}_t} \|\boldsymbol{\theta}^{k_t} - \boldsymbol{\theta}^t\|,\end{aligned}$$

where the last equality holds being $\|\hat{\mathbf{H}}_t^{(i)}(\boldsymbol{\theta}^{k_t}) - \mathbf{H}^{(i)}(\boldsymbol{\theta}^{k_t})\| = \hat{\rho}_t^{(i)} - \lambda_{n,t}^{(i)}$, which in turn is true given that $\hat{\rho}_t^{(i)} = \hat{\lambda}^{(i)}_{q_t^{(i)}+1,t}$.

A.8 Proof of Theorem 2.6

1) We first need the following Lemma:

Lemma A.2. Let $\bar{\kappa} = \sum_{i=1}^{M} \kappa_i$, with κ_i the strong convexity constant of the cost $f^{(i)}$ of agent i, let K be the smoothness constant of f and $M(t) = \max\{\|\boldsymbol{\theta}^t - \boldsymbol{\theta}^*\|, \|\boldsymbol{\theta}^{k_t} - \boldsymbol{\theta}^*\|\}$. Applying Algorithm 5, if

$$3\bar{\kappa}(M(t) + \|\boldsymbol{\theta}^t - \boldsymbol{\theta}^*\|) + K\|\boldsymbol{\theta}^t - \boldsymbol{\theta}^*\| \leq \frac{3\bar{\kappa}^2}{L}(1 - 2\alpha), \tag{A.10}$$

then, for any $\alpha \in (0, 1/2)$, the backtracking algorithm (2) chooses $\eta_t = 1$.

Proof. The following proof is similar to the proof for the beginning of the quadratically convergent phase in centralized Newton method by [24], page 490-491. We start with some definitions: at iteration t, let $f(\boldsymbol{\theta}^t), \mathbf{g}_t = \nabla f(\boldsymbol{\theta}^t), \mathbf{H}_t = \nabla^2 f(\boldsymbol{\theta}^t)$ be the cost, the gradient and the Hessian, respectively, computed at $\boldsymbol{\theta}^t$. Let $\mathbf{H}_{k_t} = \nabla^2 f(\boldsymbol{\theta}^{k_t})$ be the Hessian at $\boldsymbol{\theta}^{k_t}$, and $\hat{\mathbf{H}}_t$ the global Hessian approximation of Algorithm 5. Let the NT descent direction be $\mathbf{p}_t = \hat{\mathbf{H}}_t^{-1}\mathbf{g}_t$. Define

$$\begin{aligned}
\bar{\sigma}_t^2 &:= \mathbf{p}_t^T \mathbf{g}_t = \mathbf{g}_t^T \hat{\mathbf{H}}_t^{-1} \mathbf{g}_t, \\
\sigma_t^2 &:= \mathbf{p}_t^T \nabla^2 f(\boldsymbol{\theta}^t) \mathbf{p}_t = \mathbf{p}_t^T \mathbf{H}_t \mathbf{p}_t, \\
\tilde{f}(\eta) &:= f(\boldsymbol{\theta}^t - \eta \mathbf{p}_t), \quad \tilde{f}(0) = f(\boldsymbol{\theta}^t), \\
\tilde{f}'(\eta) &:= \frac{\partial \tilde{f}(\eta)}{\partial \eta} = -\nabla f(\boldsymbol{\theta}^t - \eta \mathbf{p}_t)^T \mathbf{p}_t, \\
\tilde{f}''(\eta) &:= \frac{\partial^2 \tilde{f}(\eta)}{\partial \eta^2} = \mathbf{p}_t^T \nabla^2 f(\boldsymbol{\theta}^t - \eta \mathbf{p}_t) \mathbf{p}_t.
\end{aligned} \tag{A.11}$$

Note that

$$\begin{aligned}
\tilde{f}'(0) &= -\mathbf{g}_t^T \mathbf{p}_t = -\bar{\sigma}_t^2, \\
\tilde{f}''(0) &= \mathbf{p}_t^T \mathbf{H}_t \mathbf{p}_t = \sigma_t^2.
\end{aligned} \tag{A.12}$$

Note that $\hat{\mathbf{H}}_t \geq \bar{\kappa}, \ \ \forall t$. Thanks to L-Lipschitz continuity, it holds that $\|\nabla^2 f(\boldsymbol{\theta}^t - \eta \mathbf{p}_t) -$

$\nabla^2 f(\boldsymbol{\theta}^t)\| \leq \eta L \|\mathbf{p}_t\|$ and we have that

$$\begin{aligned} |\tilde{f}''(\eta) - \tilde{f}''(0)| &= \mathbf{p}_t^T (\nabla^2 f(\boldsymbol{\theta}^t - \eta \mathbf{p}_t) - \nabla^2 f(\boldsymbol{\theta}^t)) \mathbf{p}_t \\ &\leq \eta L \|\mathbf{p}_t\|^3 \leq \eta L \frac{\bar{\sigma}^3(\boldsymbol{\theta}^t)}{\bar{\kappa}^{3/2}} \end{aligned} \tag{A.13}$$

where the last inequality holds because $\bar{\kappa}\|\mathbf{p}_t\|^2 \leq \mathbf{p}_t^T \hat{\mathbf{H}} \mathbf{p}_t = \bar{\sigma}_t^2$. From (A.13) it follows that

$$\tilde{f}''(\eta) \leq \tilde{f}''(0) + \eta L \frac{\bar{\sigma}^3(\boldsymbol{\theta}^t)}{\bar{\kappa}^{3/2}} = \sigma_t^2 + \eta L \frac{\bar{\sigma}^3(\boldsymbol{\theta}^t)}{\bar{\kappa}^{3/2}}. \tag{A.14}$$

Similarly to [24], page 490-491, we can now integrate both sides of the inequality getting

$$\tilde{f}'(\eta) \leq \tilde{f}'(0) + \eta \sigma_t^2 + \frac{\eta^2}{2} L \frac{\bar{\sigma}_t^3}{\bar{\kappa}^{3/2}} = -\bar{\sigma}_t^2 + \eta \sigma_t^2 + \frac{\eta^2}{2} L \frac{\bar{\sigma}_t^3}{\bar{\kappa}^{3/2}}.$$

By integrating again both sides of the inequality, we get, recalling that $\tilde{f}(0) = f(\boldsymbol{\theta}^t)$,

$$\tilde{f}(\eta) = f(\boldsymbol{\theta}^t - \eta \mathbf{p}_t) \leq f(\boldsymbol{\theta}^t) - \eta \bar{\sigma}_t^2 + \frac{\eta^2}{2} \sigma_t^2 + \frac{\eta^3}{6} L \frac{\bar{\sigma}_t^3}{\bar{\kappa}^{3/2}}. \tag{A.15}$$

Now, recalling $\mathbf{H}(\boldsymbol{\theta}^{k_t}) \leq \hat{\mathbf{H}}_t$, we get that

$$\begin{aligned} \sigma_t^2 &= \mathbf{p}_t^T \mathbf{H}_t \mathbf{p}_t = \mathbf{p}_t^T (\mathbf{H}_{k_t} + \mathbf{H}_t - \mathbf{H}_{k_t}) \mathbf{p}_t \\ &= \mathbf{p}_t^T \mathbf{H}_{k_t} \mathbf{p}_t + \mathbf{p}_t^T (\mathbf{H}_t - \mathbf{H}_{k_t}) \mathbf{p}_t \\ &\leq \mathbf{p}_t^T \hat{\mathbf{H}}_t \mathbf{p}_t + L \|\mathbf{p}_t\|^2 \|\boldsymbol{\theta}^t - \boldsymbol{\theta}^{k_t}\| \leq \bar{\sigma}_t^2 + \frac{L \bar{\sigma}_t^2}{\bar{\kappa}} \|\boldsymbol{\theta}^t - \boldsymbol{\theta}^{k_t}\|, \end{aligned} \tag{A.16}$$

where we have used Lipschitz continuity, and the fact that $\mathbf{p}_t^T \hat{\mathbf{H}}_t \mathbf{p}_t = \bar{\sigma}_t^2$ and $\bar{\kappa}\|\mathbf{p}_t\|^2 \leq \bar{\sigma}_t^2$. Next, setting $\eta = 1$ and plugging (A.16) in (A.15), we get

$$\begin{aligned} f(\boldsymbol{\theta}^t - \mathbf{p}_t) &\leq f(\boldsymbol{\theta}^t) - \bar{\sigma}_t^2 + \frac{\sigma_t^2}{2} + \frac{L}{6} \frac{\bar{\sigma}_t^3}{\bar{\kappa}^{3/2}} \\ &\leq f(\boldsymbol{\theta}^t) - \frac{\bar{\sigma}_t^2}{2} + \frac{L \bar{\sigma}_t^2}{2\bar{\kappa}} \|\boldsymbol{\theta}^t - \boldsymbol{\theta}^{k_t}\| + \frac{L}{6} \frac{\bar{\sigma}_t^3}{\bar{\kappa}^{3/2}} \\ &\leq f(\boldsymbol{\theta}^t) - \bar{\sigma}_t^2 \left(\frac{1}{2} - \frac{L}{2\bar{\kappa}} \|\boldsymbol{\theta}^t - \boldsymbol{\theta}^{k_t}\| - \frac{L}{6} \frac{\bar{\sigma}_t}{\bar{\kappa}^{3/2}}\right) \\ &= f(\boldsymbol{\theta}^t) - \mathbf{p}_t^T \mathbf{g}_t \left(\frac{1}{2} - \frac{L}{2\bar{\kappa}} \|\boldsymbol{\theta}^t - \boldsymbol{\theta}^{k_t}\| - \frac{L}{6} \frac{\bar{\sigma}_t}{\bar{\kappa}^{3/2}}\right). \end{aligned} \tag{A.17}$$

In order for (A.17) to satisfy the Armijo-Goldstein condition (2.13) for any parameter

$\alpha \in (0, 1/2)$ we see that

$$\frac{1}{2} - \frac{L}{2\bar{\kappa}}\|\boldsymbol{\theta}^t - \boldsymbol{\theta}^{k_t}\| - \frac{L}{6}\frac{\bar{\sigma}_t}{\bar{\kappa}^{3/2}} \geq \alpha \tag{A.18}$$

provides a sufficient condition. The above inequality can be written as

$$3\bar{\kappa}\|\boldsymbol{\theta}^t - \boldsymbol{\theta}^{k_t}\| + \bar{\kappa}^{1/2}\bar{\sigma}_t \leq \frac{3\bar{\kappa}^2}{L}(1 - 2\alpha). \tag{A.19}$$

We have that $\bar{\sigma}_t^2 = \mathbf{g}_t^T \hat{\mathbf{H}}^{-1}\mathbf{g}_t \leq \|\mathbf{g}_t\|^2/\bar{\kappa}$, which implies $\bar{\kappa}^{1/2}\|\sigma_t\| \leq \|\mathbf{g}_t\|$. Furthermore, by triangular inequality, we have $\|\boldsymbol{\theta}^t - \boldsymbol{\theta}^{k_t}\| \leq \|\boldsymbol{\theta}^t - \boldsymbol{\theta}^*\| + \|\boldsymbol{\theta}^{k_t} - \boldsymbol{\theta}^*\|$. Therefore, we see that if

$$3\bar{\kappa}(\|\boldsymbol{\theta}^t - \boldsymbol{\theta}^*\| + \|\boldsymbol{\theta}^{k_t} - \boldsymbol{\theta}^*\|) + \|\mathbf{g}_t\| \leq \frac{3\bar{\kappa}^2}{L}(1 - 2\alpha), \tag{A.20}$$

then the Armijo-Goldstein condition is satisfied and $\eta = 1$ is chosen by the backtracking algorithm, proving the Lemma. Indeed, by K-smoothness of the cost function (see Assumption 3) we have $\|\mathbf{g}_t\| \leq K\|\boldsymbol{\theta}^t - \boldsymbol{\theta}^*\|$ and so the condition of the Lemma implies (A.20). □

Let condition (A.10) be satisfied. Then, if

$$(3/2)L\|\boldsymbol{\theta}^t - \boldsymbol{\theta}^*\| + LM(t) \leq \bar{\kappa}, \tag{A.21}$$

the convergence of SHED is at least linear. Indeed, if (A.10) is satisfied, then, from Lemma A.2, the step size is $\eta_t = 1$ and the convergence bound (see Theorem 2.5) becomes $\|\boldsymbol{\theta}^{t+1} - \boldsymbol{\theta}^*\| \leq c_t\|\boldsymbol{\theta}^t - \boldsymbol{\theta}^*\|$, with

$$\begin{aligned} c_t &= (1 - \frac{\bar{\lambda}_{n,t}}{\bar{\rho}_t} + \frac{L}{\bar{\rho}_t}\|\boldsymbol{\theta}^t - \boldsymbol{\theta}^{k_t}\| + \frac{L}{2\bar{\rho}_t}\|\boldsymbol{\theta}^t - \boldsymbol{\theta}^*\|) \\ &\leq (1 - \frac{\bar{\lambda}_{n,t}}{\bar{\rho}_t} + \frac{L}{\bar{\rho}_t}\|\boldsymbol{\theta}^{k_t} - \boldsymbol{\theta}^*\| + \frac{3L}{2\bar{\rho}_t}\|\boldsymbol{\theta}^t - \boldsymbol{\theta}^*\|) \end{aligned} \tag{A.22}$$

and it is easy to see that condition (A.21) implies that $c_t < 1$ and thus we get a contraction in $\|\boldsymbol{\theta}^t - \boldsymbol{\theta}^*\|$. Furthermore, when conditions (A.10) and (A.21) are both satisfied at some iteration $\bar{t}$, they are then satisfied for all $t \geq \bar{t}$ and thus $c_t < 1$ for all $t \geq \bar{t}$. Indeed, $c_t < 1$ implies that $\|\boldsymbol{\theta}^{t+1} - \boldsymbol{\theta}^*\| < \|\boldsymbol{\theta}^t - \boldsymbol{\theta}^*\|$ and $M(t+1) \leq M(t)$ because either $k_{t+1} = k_t$ or $k_{t+1} = t + 1$. Note that Assumption 2 is needed to guarantee that (A.10) and (A.21) are eventually satisfied.

2) From 1), we can write $\|\boldsymbol{\theta}^t - \boldsymbol{\theta}^*\| \leq Ca^t$ for some $a \in (0, 1)$ and some $C > 0$. Considering $t \geq \bar{t}$, with $\bar{t}$ the first iteration for which both conditions (A.10) and (A.21) are satisfied,

we consider c_t as in (A.22), and let $T = t - k_t$

$$\begin{aligned} c_t &\leq 1 - \frac{\bar{\lambda}_{n,t}}{\bar{\rho}_t} + \frac{L}{\bar{\rho}_t}\|\boldsymbol{\theta}^{k_t} - \boldsymbol{\theta}^*\| + \frac{3L}{2\bar{\rho}_t}\|\boldsymbol{\theta}^t - \boldsymbol{\theta}^*\| \\ &\leq 1 - \frac{\bar{\lambda}_{n,t}}{\bar{\rho}_t} + \frac{L}{\bar{\rho}_t}C_1 a^{t-T} + \frac{3L}{2\bar{\rho}_t}C_2 a^t \\ &= 1 - \frac{\bar{\lambda}_{n,t}}{\bar{\rho}_t} + Ba^t, \end{aligned} \tag{A.23}$$

where $B = \frac{L}{\bar{\rho}_t}C_1 a^{-T} + \frac{3L}{2\bar{\rho}_t}C_2$, and C_1, C_2 are some bounded positive constants. Note that T is bounded by Assumption 2. For any iteration t, we can write $\|\boldsymbol{\theta}^{t+1} - \boldsymbol{\theta}^*\| \leq \bar{c}_t\|\boldsymbol{\theta}^t - \boldsymbol{\theta}^*\|$ for some $\bar{c}_t$ that could also be greater than one, if $t < \bar{t}$, but it is easy to see that $\bar{c}_t$ is always bounded. Now, we consider $\bar{a} := \limsup_t (\prod_{k=1}^t \bar{c}_k)^{1/t}$. It is straightforward to see that, as in the least squares case, the Lyapunov exponent of $\|\boldsymbol{\theta}^t - \boldsymbol{\theta}^*\|$ is $a_* \leq \bar{a}$. We can write $\log \bar{a} = \limsup_t \frac{1}{t}(\sum_{k=\bar{t}+1}^t \log c_k)$. We get

$$\begin{aligned} \log \bar{a} &\leq \limsup_t \frac{1}{t}\sum_{k=\bar{t}+1}^{t} \log\left(1 - \frac{\bar{\lambda}_{n,k}}{\bar{\rho}_k} + Ba^k\right) \\ &= \limsup_t \frac{1}{t}\sum_{k=\bar{t}+1}^{t} \log\left(1 - \frac{\bar{\lambda}_{n,k}}{\bar{\rho}_k}\right) + \log\left(1 + \frac{Ba^k}{1 - \frac{\bar{\lambda}_{n,k}}{\bar{\rho}_k}}\right). \end{aligned}$$

We see that the last term is

$$\log\left(1 + \frac{Ba^k}{1 - \frac{\bar{\lambda}_{n,k}}{\bar{\rho}_k}}\right) \leq \frac{Ba^k}{1 - \frac{\bar{\lambda}_{n,k}}{\bar{\rho}_k}} \leq \bar{B}a^k$$

that comes from the identity $\log(1 + x) \leq x$, and where $\bar{B} = \max_k \frac{1}{1 - \frac{\bar{\lambda}_{n,k}}{\bar{\rho}_k}}$, bounded because $|\mathcal{X}_t| = 0, \quad \forall t$, and thus $\bar{\rho}_k > \bar{\lambda}_{n,k}, \forall k$. Now, we see that

$$\limsup_t \frac{1}{t}\sum_{k=1}^{t} \bar{B}a^k = \limsup_t \frac{1}{t}\bar{B}\left(\frac{1 - a^{t+1}}{1 - a} - 1\right) = 0.$$

Hence, we get, using also the finiteness of $\bar{t}$,

$$\log \bar{a} = \limsup_t \frac{1}{t}\sum_{k=1}^{t} \log\left(1 - \frac{\bar{\lambda}_{n,k}}{\bar{\rho}_k}\right). \tag{A.24}$$

Now we use local Lipschitz continuity (Assumption 3) to conclude the proof. Lipschitz continuity of $f^{(i)}(\boldsymbol{\theta})$ implies that, for $\bar{L} = \max_k L_k$ and for any $k \in \{1, ..., n\}$, $|\lambda_k^{(i)}(\boldsymbol{\theta}) -$

$\lambda_k^{(i)}(\boldsymbol{\theta}^*)| \leq \bar{L}\|\boldsymbol{\theta} - \boldsymbol{\theta}^*\|$, which in turn implies $\lambda_k^{(i)}(\boldsymbol{\theta}) \geq \lambda_k^{(i)}(\boldsymbol{\theta}^*) - \bar{L}\|\boldsymbol{\theta} - \boldsymbol{\theta}^*\|$ and $\lambda_k^{(i)}(\boldsymbol{\theta}) \leq \lambda_k^{(i)}(\boldsymbol{\theta}^*) + \bar{L}\|\boldsymbol{\theta} - \boldsymbol{\theta}^*\|$. For a proof of this result, see also [21], page 116, Theorem 4.25. It follows that

$$\begin{aligned}
\log \bar{a} &\leq \limsup_t \frac{1}{t}\sum_{k=1}^{t} \log\left(1 - \frac{\bar{\lambda}_n^o - \bar{L}\|\boldsymbol{\theta}^{k_t} - \boldsymbol{\theta}^*\|}{\bar{\rho}_k^o + \bar{L}\|\boldsymbol{\theta}^{k_t} - \boldsymbol{\theta}^*\|}\right) \\
&\overset{(1)}{=} \limsup_t \frac{1}{t}\sum_{k=1}^{t} \log\left(1 - \frac{\bar{\lambda}_n^o}{\bar{\rho}_k^o + \bar{L}\|\boldsymbol{\theta}^{k_t} - \boldsymbol{\theta}^*\|}\right) \\
&= \limsup_t \frac{1}{t}\sum_{k=1}^{t} \log\left(\frac{\bar{\rho}_k^o - \bar{\lambda}_n^o + \bar{L}\|\boldsymbol{\theta}^{k_t} - \boldsymbol{\theta}^*\|}{\bar{\rho}_k^o + \bar{L}\|\boldsymbol{\theta}^{k_t} - \boldsymbol{\theta}^*\|}\right) \\
&\leq \limsup_t \frac{1}{t}\sum_{k=1}^{t} \log\left(1 - \frac{\bar{\lambda}_n^o}{\bar{\rho}_k^o} + \frac{\bar{L}\|\boldsymbol{\theta}^{k_t} - \boldsymbol{\theta}^*\|}{\bar{\rho}_k^o}\right) \\
&\overset{(2)}{=} \limsup_t \frac{1}{t}\sum_{k=1}^{t} \log\left(1 - \frac{\bar{\lambda}_n^o}{\bar{\rho}_k^o}\right)
\end{aligned}$$

where equalities (1) and (2) follow from calculations equivalent to the ones used to obtain (A.24).

3) Consider c_t as it was defined and bounded in (A.23), $C_3 > 0$ a constant such that $\log c_k \leq C_3$, $\forall k$, and such that $\log B \leq C_3$. Let $\bar{a}$ be defined as before. Let $a \in (0, 1)$. We have

$$\begin{aligned}
\log \bar{a} &= \limsup_t \frac{1}{t}\sum_{k=1}^{t} \log c_k \\
&\leq \limsup_t \frac{1}{t}\sum_{k=1}^{t} \log\left(1 - \frac{\bar{\lambda}_{n,k}}{\bar{\rho}_k} + Ba^k\right) \\
&= \limsup_t \frac{1}{t}\Big(\sum_{k\notin\mathcal{X}_t} \log c_k + \sum_{k\in\mathcal{X}_t} \log Ba^k\Big) \\
&\leq 2C_3 + \limsup_t \frac{1}{t}\log a \sum_{k\in\mathcal{X}_t} k \\
&\leq 2C_3 + \limsup_t \frac{1}{t} C_4 |\mathcal{X}_t|^2 \log a \\
&\leq 2C_3 + \limsup_t \frac{1}{t} C_4 (t^{1/2}h(t) - \bar{T})^2 \log a \\
&\leq 2C_3 + \limsup_t C_4 h(t)^2 \log a = -\infty
\end{aligned} \tag{A.25}$$

where $C_4 > 0$ is some positive constant and the last equality follows because $\log a < 0$ and $\lim_t h(t) = \infty$. We see that $0 \leq a_* \leq \bar{a} \leq 0$, which implies $a_* = 0$.

A.9 Additional Experiments: Results on EMNIST and w8a

In this appendix, we include the results on the EMNIST digits and 'w8a' datasets when comparing the different algorithms. We show results for two values of the regularization parameter μ, specifically $\mu = 10^{-5}$ and $\mu = 10^{-6}$, in Figure A.1 and A.2, respectively. The results obtained on the EMNIST digits dataset confirm the results that were obtained with FMNIST, with the difference that in the case $\mu = 10^{-5}$, GIANT is not much impacted by the considered non i.i.d. configuration. The results obtained with the 'w8a' dataset show that, while GIANT performance is largely degraded because of the non i.i.d. configuration, also in this case Fib-SHED and Fib-SHED+ significantly outperform FedNL in both communication rounds and communication load required for convergence.

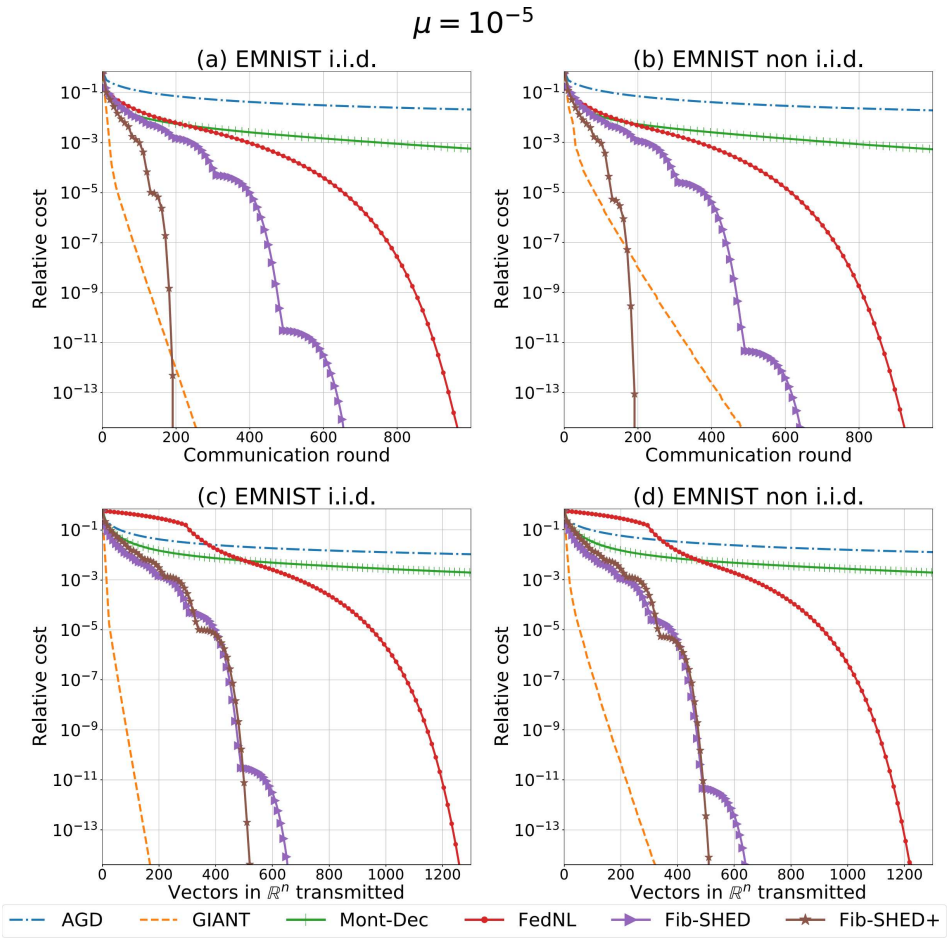

Figure A.1: Performance comparison of logistic regression on EMNIST when $\mu = 10^{-5}$. Relative cost is $f(\theta^t) - f(\theta^*)$.

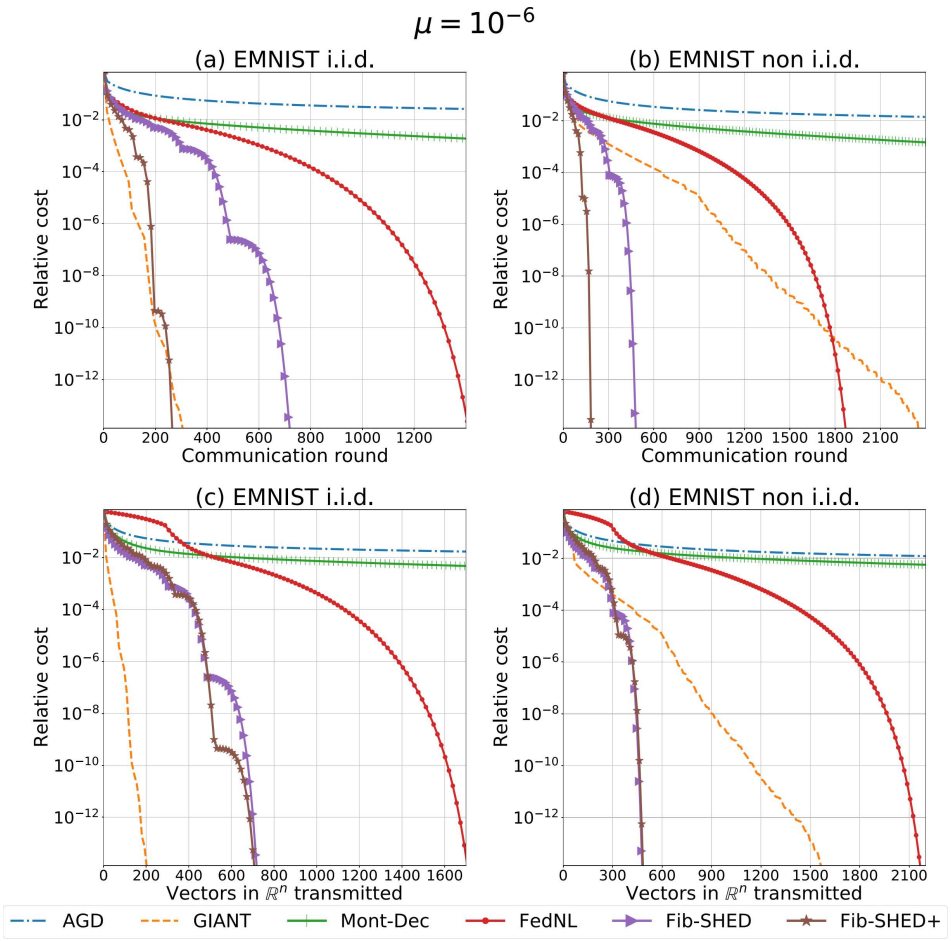

Figure A.2: Performance comparison of logistic regression on EMNIST when $\mu = 10^{-6}$. Relative cost is $f(\theta^t) - f(\theta^*)$.

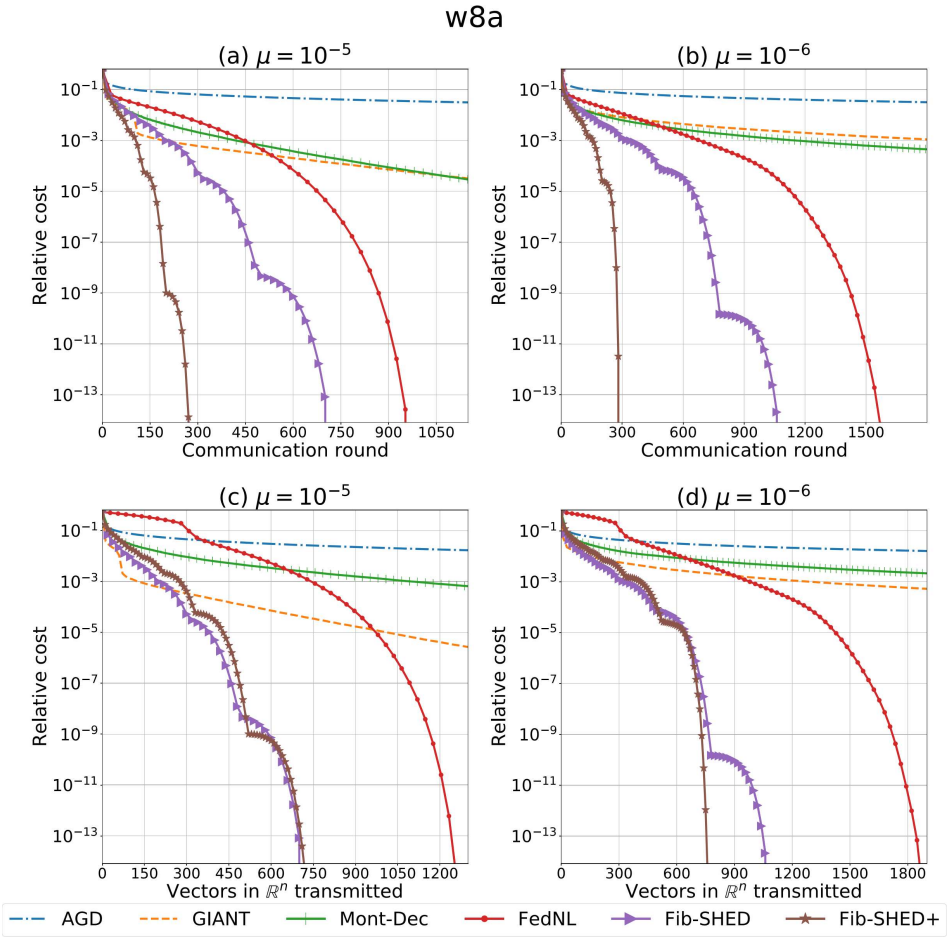

Figure A.3: Performance comparison of logistic regression on w8a when $\mu = 10^{-5}$ and $\mu = 10^{-6}$. Relative cost is $f(\theta^t) - f(\theta^*)$.

B

Appendix: Proofs of Chapter 4

B.1 Proof of Theorem 4.2

In this section, we will prove Theorem 4.2. We start by introducing some definitions to lighten the notation, and by recalling some basic results from prior work. Let us define:

$$\begin{aligned} \eta_{k,\tau}^{(i)}(\boldsymbol{\theta}) &\triangleq \|\mathbb{E}\left[\mathbf{g}_{i,k}(\boldsymbol{\theta}, o_{i,k})|o_{i,k-\tau}\right] - \bar{\mathbf{g}}(\boldsymbol{\theta})\|, \ k \geq \tau, \\ \delta_{k,\tau} &\triangleq \|\boldsymbol{\theta}_k - \boldsymbol{\theta}_{k-\tau}\|, \ k \geq \tau. \end{aligned} \tag{B.1}$$

For our analysis, we will need the following result from [19].

Lemma B.1. The following holds $\forall \boldsymbol{\theta} \in \mathbb{R}^m$:

$$\langle \boldsymbol{\theta}^* - \boldsymbol{\theta}, \bar{\mathbf{g}}(\boldsymbol{\theta}) \rangle \geq \omega(1-\gamma)\|\boldsymbol{\theta}^* - \boldsymbol{\theta}\|^2.$$

We will also use the fact that the random TD update directions and their steady-state versions are 2-Lipschitz [19], i.e., $\forall i \in [N], \forall k \in \mathbb{N}$, and $\forall \boldsymbol{\theta}, \boldsymbol{\theta}' \in \mathbb{R}^m$, we have:

$$\max\{\|\mathbf{g}_{i,k}(\boldsymbol{\theta}) - \mathbf{g}_{i,k}(\boldsymbol{\theta}')\|, \|\bar{\mathbf{g}}(\boldsymbol{\theta}) - \bar{\mathbf{g}}(\boldsymbol{\theta}')\|\} \leq 2\|\boldsymbol{\theta} - \boldsymbol{\theta}'\|. \tag{B.2}$$

Finally, we will use the following bound from [135]:

$$\|\mathbf{g}_{i,k}(\boldsymbol{\theta}, o_{i,k})\| \leq 2\|\boldsymbol{\theta}\| + 2\bar{r}, \forall i \in [N], \forall k \in \mathbb{N}, \forall \boldsymbol{\theta} \in \mathbb{R}^m. \tag{B.3}$$

Equipped with the above basic results, we now provide an outline of our proof before delving into the technical details.

Outline of the proof. We start by defining:

$$\begin{aligned} \bar{\mathbf{g}}_N(\boldsymbol{\theta}_k) &\triangleq \frac{1}{N}\sum_{i=1}^{N} b_{i,k}\bar{\mathbf{g}}(\boldsymbol{\theta}_k), \text{ and} \\ \psi_k &\triangleq \langle \mathbf{v}_k - \bar{\mathbf{g}}_N(\boldsymbol{\theta}_k), \boldsymbol{\theta}_k - \boldsymbol{\theta}^* \rangle. \end{aligned} \tag{B.4}$$

Since for all $i \in [N]$, $b_{i,k}$ is independent of $\boldsymbol{\theta}_k$, we have $\mathbb{E}\left[\langle \bar{\mathbf{g}}_N(\boldsymbol{\theta}_k), \boldsymbol{\theta}_k - \boldsymbol{\theta}^* \rangle\right] = p\mathbb{E}\left[\langle \bar{\mathbf{g}}(\boldsymbol{\theta}_k), \boldsymbol{\theta}_k - \boldsymbol{\theta}^* \rangle\right]$. Thus, recalling that $\delta_k^2 \triangleq \|\boldsymbol{\theta}^* - \boldsymbol{\theta}_k\|^2$, and using (4.3), we obtain

$$\begin{aligned} \mathbb{E}\left[\delta_{k+1}^2\right] &= \mathbb{E}\left[\delta_k^2\right] - 2\alpha\mathbb{E}\left[\langle \boldsymbol{\theta}^* - \boldsymbol{\theta}_k, \mathbf{v}_k \rangle\right] + \alpha^2\mathbb{E}\left[\|\mathbf{v}_k\|^2\right] \\ &= \mathbb{E}\left[\delta_k^2\right] - 2\alpha p\mathbb{E}\left[\langle \boldsymbol{\theta}^* - \boldsymbol{\theta}_k, \bar{\mathbf{g}}(\boldsymbol{\theta}_k) \rangle\right] \\ &\quad + 2\alpha\mathbb{E}\left[\psi_k\right] + \alpha^2\mathbb{E}\left[\|\mathbf{v}_k\|^2\right]. \end{aligned} \tag{B.5}$$

The main technical burden in proving Theorem 4.2 is in bounding $\mathbb{E}\left[\|\mathbf{v}_k\|^2\right]$ and $\mathbb{E}\left[\psi_k\right]$ in the above recursion. Following the centralized analysis in [19], [135], one can easily bound $\mathbb{E}\left[\|\mathbf{v}_k\|^2\right]$ using (C.4). However, this approach will fall short of yielding the desired linear speedup property. Hence, to bound $\mathbb{E}\left[\|\mathbf{v}_k\|^2\right]$, we need a much finer analysis, one that we provide in Lemma B.2. Leveraging Lemma B.2, we then establish an intermediate result in Lemma B.3 that bounds $\mathbb{E}\left[\|\boldsymbol{\theta}_k - \boldsymbol{\theta}_{k-\tau}\|\right]$. This result, in turn, helps us bound $\mathbb{E}\left[\psi_k\right]$ in Lemma B.4. We now proceed to flesh out these steps. In what follows, $\tau = \tau_\epsilon$ with $\epsilon = \alpha^q$, $q \geq 2$.

Lemma B.2. (**Key Technical Result**) For $k \geq \tau$, we have

$$\mathbb{E}\left[\|\mathbf{v}_k\|^2\right] \leq 60\zeta' p\mathbb{E}\left[\delta_k^2\right] + 12\sigma^2 p\left(10\frac{\zeta'}{N} + \alpha^{2q}\right). \tag{B.6}$$

Proof. Note that $\|\mathbf{v}_k\|^2 \leq \frac{3}{N^2}(T_1 + T_2 + T_3)$, with

$$\begin{aligned} T_1 &= \|\sum_{i=1}^{N} b_{i,k}\mathbf{g}_{i,k}(\boldsymbol{\theta}^*)\|^2, \\ T_2 &= \|\sum_{i=1}^{N} b_{i,k}(\mathbf{g}_{i,k}(\boldsymbol{\theta}_k) - \mathbf{g}_{i,k}(\boldsymbol{\theta}^*))\|^2, \text{ and} \\ T_3 &= \|\sum_{i=1}^{N} b_{i,k}(\mathbf{g}_{i,k}(\boldsymbol{\theta}_k) - \mathbf{h}_{i,k}(\boldsymbol{\theta}_k))\|^2. \end{aligned} \tag{B.7}$$

We now proceed to bound $T_1 - T_3$. To that end, we first write T_1 as

$$\begin{aligned} T_1 &= T_{11} + T_{12}, \text{ with} \\ T_{11} &= \sum_{i=1}^{N} b_{i,k}^2 \|\mathbf{g}_{i,k}(\boldsymbol{\theta}^*)\|^2, \text{ and} \\ T_{12} &= \sum_{\substack{i,j=1 \\ i\neq j}}^{N} b_{i,k} b_{j,k} \langle \mathbf{g}_{i,k}(\boldsymbol{\theta}^*), \mathbf{g}_{j,k}(\boldsymbol{\theta}^*) \rangle. \end{aligned} \tag{B.8}$$

Now using (C.4), we obtain $T_{11} \leq 8(\|\boldsymbol{\theta}^*\|^2 + \bar{r}^2) \sum_{i=1}^{N} b_{i,k}^2$. Recalling that $\sigma \triangleq \max\{1, \bar{r}, \|\boldsymbol{\theta}^*\|\}$, we then have $\mathbb{E}[T_{11}] \leq 16\sigma^2 \mathbb{E}\left[\sum_{i=1}^{N} b_{i,k}^2\right] = 16\sigma^2 Np$. Next, to bound the cross-terms in T_{12}, we will exploit the mixing property in Definition 5.2.1. To that end, we note that since (i) $\bar{\mathbf{g}}(\boldsymbol{\theta}^*) = \mathbf{0}$ [19], (ii) the packet-dropping processes are independent of the Markovian tuples, and (iii) $\mathbf{g}_{i,k}(\boldsymbol{\theta}^*)$ and $\mathbf{g}_{j,k}(\boldsymbol{\theta}^*)$ are independent for $i \neq j$,

$$\mathbb{E}[T_{12}] = \sum_{\substack{i,j=1 \\ i\neq j}}^{N} \mathbb{E}[b_{i,k} b_{j,k}] \left\langle \mathbb{E}\left[\mathbb{E}\left[\mathbf{g}_{i,k}(\boldsymbol{\theta}^*) | o_{i,k-\tau}\right] - \bar{\mathbf{g}}(\boldsymbol{\theta}^*)\right], \mathbb{E}\left[\mathbb{E}\left[\mathbf{g}_{j,k}(\boldsymbol{\theta}^*) | o_{j,k-\tau}\right] - \bar{\mathbf{g}}(\boldsymbol{\theta}^*)\right] \right\rangle.$$

Using the Cauchy-Schwarz inequality followed by Jensen's inequality, we can further bound the above inner-product via $\mathbb{E}\left[\eta_{k,\tau}^{(i)}(\boldsymbol{\theta}^*)\right] \times \mathbb{E}\left[\eta_{k,\tau}^{(j)}(\boldsymbol{\theta}^*)\right] \leq 4\sigma^2\alpha^{2q}$. For the last inequality, we used the mixing property by noting that $k \geq \tau$. Specifically, appealing to Definition 5.2.1, and recalling that $\sigma \triangleq \max\{1, \bar{r}, \|\boldsymbol{\theta}^*\|\}$, we have

$$\eta_{k,\tau}^{(i)}(\boldsymbol{\theta}^*) \leq \alpha^q(\|\boldsymbol{\theta}^*\| + 1) \leq 2\sigma\alpha^q.$$

Clearly, the same bound also applies to $\eta_{k,\tau}^{(j)}(\boldsymbol{\theta}^*)$ via an identical reasoning. Combining this analysis with the fact that $\mathbb{E}[b_{i,k} b_{j,k}] = \mathbb{E}[b_{i,k}]\,\mathbb{E}[b_{j,k}] = p^2$, we obtain that $\mathbb{E}[T_{12}] \leq 4N^2p^2\sigma^2\alpha^{2q}$. Combining the bounds for $\mathbb{E}[T_{11}]$ and $\mathbb{E}[T_{12}]$ thus yields:

$$\mathbb{E}[T_1] \leq 16\sigma^2 Np + 4N^2p^2\sigma^2\alpha^{2q}. \tag{B.9}$$

Now, using (C.3), we see that

$$\begin{aligned} \mathbb{E}[T_2] &\leq N \sum_{i=1}^{N} \mathbb{E}\left[b_{i,k}^2 \|\mathbf{g}_{i,k}(\boldsymbol{\theta}_k) - \mathbf{g}_{i,k}(\boldsymbol{\theta}^*)\|^2\right] \\ &\leq 4N\mathbb{E}\left[\delta_k^2\right] \sum_{i=1}^{N} \mathbb{E}\left[b_{i,k}^2\right] = 4pN^2\mathbb{E}\left[\delta_k^2\right]. \end{aligned} \tag{B.10}$$

Defining $\boldsymbol{\lambda}_{i,k}(\boldsymbol{\theta}_k) \triangleq \mathbf{h}_{i,k}(\boldsymbol{\theta}_k) - \mathbf{g}_{i,k}(\boldsymbol{\theta}_k)$, we now turn to bounding T_3 by writing it as

$$\begin{aligned} T_3 &= T_{31} + T_{32}, \text{ with} \\ T_{31} &= \sum_{i=1}^{N} b_{i,k}^2 \|\boldsymbol{\lambda}_{i,k}(\boldsymbol{\theta}_k)\|^2, \text{ and} \\ T_{32} &= \sum_{\substack{i,j \\ i \neq j}}^{N} b_{i,k} b_{j,k} \langle \boldsymbol{\lambda}_{i,k}(\boldsymbol{\theta}_k), \boldsymbol{\lambda}_{j,k}(\boldsymbol{\theta}_k) \rangle. \end{aligned} \tag{B.11}$$

We now proceed to bound $\mathbb{E}[T_{31}]$ and $\mathbb{E}[T_{32}]$ as follows:

$$\begin{aligned} \mathbb{E}[T_{31}] &= \sum_{i=1}^{N} \mathbb{E}\left[b_{i,k}^2\right] \mathbb{E}\left[\mathbb{E}\left[\|\boldsymbol{\lambda}_{i,k}(\boldsymbol{\theta}_k)\|^2 | o_{i,k}, \boldsymbol{\theta}_k\right]\right] \\ &\overset{(a)}{\leq} \sum_{i=1}^{N} p\zeta \mathbb{E}\left[\|\mathbf{g}_{i,k}(\boldsymbol{\theta}_k)\|^2\right] \\ &\overset{(b)}{\leq} 8Np\zeta(\mathbb{E}\left[\|\boldsymbol{\theta}_k\|^2\right] + \sigma^2) \\ &\leq 16Np\zeta\mathbb{E}\left[\|\boldsymbol{\theta}_k - \boldsymbol{\theta}^*\|^2\right] + 24Np\zeta\sigma^2, \end{aligned}$$

where (a) follows from the variance bound of the quantizer map $\mathcal{Q}(\cdot)$, and (b) follows from (C.4). Next, observe that:

$$\mathbb{E}[T_{32}] = p^2 \sum_{\substack{i,j=1 \\ i \neq j}}^{N} \mathbb{E}\left[\mathbb{E}\left[\langle \boldsymbol{\lambda}_{i,k}(\boldsymbol{\theta}_k), \boldsymbol{\lambda}_{j,k}(\boldsymbol{\theta}_k) \rangle | o_{i,k}, o_{j,k}, \boldsymbol{\theta}_k\right]\right].$$

Using the fact that the randomness of the quantization map is independent across agents, and the unbiasedness of $\mathcal{Q}(\cdot)$, we conclude that $\mathbb{E}[T_{32}] = 0$. Combining the bounds on $\mathbb{E}[T_1]$, $\mathbb{E}[T_2]$, and $\mathbb{E}[T_3]$ above yields the desired result. □

Remark B.1. As the rest of our analysis will reveal, Lemma B.2 is really the key technical result that will help us establish the desired linear speedup effect under Markovian sampling. One important takeaway from the proof of this result is that we do not need to exploit the fact that the TD update direction is an affine function of the parameter $\boldsymbol{\theta}_k$. As such, Lemma B.2 should essentially be applicable (with potentially minor modifications) to more general stochastic approximation schemes where the operator under consideration satisfies basic smoothness properties.

Later in the analysis, we will once again need to invoke a mixing time argument by conditioning on $\boldsymbol{\theta}_{k-\tau}$. This will give rise to the $\delta_{k,\tau} = \|\boldsymbol{\theta}_k - \boldsymbol{\theta}_{k-\tau}\|$ term that we proceed

to bound below by leveraging Lemma B.2.

Lemma B.3. Let $\alpha \leq \frac{1}{484\tau\zeta'}$ and $k \geq 2\tau$. Then, we have

$$\mathbb{E}\left[\delta_{k,\tau}^2\right] \leq 480\alpha^2\tau^2 p\zeta' \mathbb{E}\left[\delta_k^2\right] + \alpha^2\tau^2 p\sigma^2\left(\frac{360\zeta'}{N} + 4\alpha^q\right).$$

Proof. We start with a bound on δ_{k+1}^2:

$$\begin{aligned}
\delta_{k+1}^2 &= \delta_k^2 - 2\alpha\langle \mathbf{v}_k, \boldsymbol{\theta}^* - \boldsymbol{\theta}_k\rangle + \alpha^2\|\mathbf{v}_k\|^2 \\
&\overset{(a)}{\leq} \delta_k^2 + 2\alpha\|\mathbf{v}_k\|\delta_k + \alpha^2\|\mathbf{v}_k\|^2 \\
&\overset{(b)}{\leq} (1+\alpha)\delta_k^2 + (\alpha+\alpha^2)\|\mathbf{v}_k\|^2 \\
&\overset{(c)}{\leq} (1+\alpha)\delta_k^2 + 2\alpha\|\mathbf{v}_k\|^2.
\end{aligned} \tag{B.12}$$

In the above steps, (a) follows from the Cauchy-Schwarz inequality. For (b), we note that given any two positive numbers x and y, it holds that

$$xy \leq \frac{1}{2}x^2 + \frac{1}{2}y^2.$$

For (c), we simply used the fact that since $\alpha \in (0,1)$, it holds that $\alpha^2 \leq \alpha$. Hence, $\alpha + \alpha^2 \leq 2\alpha$. Using Lemma B.2 and the fact that $p < 1$, we obtain

$$\mathbb{E}\left[\delta_{k+1}^2\right] \leq (1 + 121\alpha\zeta')\mathbb{E}\left[\delta_k^2\right] + \underbrace{24\alpha p\sigma^2\left(\frac{10\zeta'}{N} + \alpha^{2q}\right)}_{B}.$$

Iterating this inequality, we get for any $k - \tau \leq k' \leq k$,

$$\mathbb{E}\left[\delta_{k'}^2\right] \leq (1 + 121\alpha\zeta')^\tau \mathbb{E}\left[\delta_{k-\tau}^2\right] + B\sum_{\ell=0}^{\tau-1}(1 + 121\alpha\zeta')^\ell. \tag{B.13}$$

Now using $(1+x) \leq e^x, \forall x \in \mathbb{R}$, observe that $(1 + 121\alpha\zeta')^\ell \leq (1 + 121\alpha\zeta')^\tau \leq e^{0.25} \leq 2$, for $\alpha \leq 1/(484\tau\zeta')$. Thus, $\sum_{\ell=0}^{\tau-1}(1 + 121\alpha\zeta')^\ell \leq 2\tau$. Plugging this bound in (B.13), we obtain

$$\mathbb{E}\left[\delta_{k'}^2\right] \leq 2\mathbb{E}\left[\delta_{k-\tau}^2\right] + 2\tau B. \tag{B.14}$$

Next, observe that

$$\delta_{k,\tau}^2 \leq \tau\sum_{\ell=k-\tau}^{k-1}\|\boldsymbol{\theta}_{\ell+1} - \boldsymbol{\theta}_\ell\|^2 = \tau\alpha^2\sum_{\ell=k-\tau}^{k-1}\|\mathbf{v}_\ell\|^2.$$

Since $k \geq 2\tau$, we have $\ell \geq \tau$. Hence, we can invoke Lemma B.2 to bound $\mathbb{E}\left[\|\mathbf{v}_\ell\|^2\right]$. This yields

$$\mathbb{E}\left[\delta_{k,\tau}^2\right] \leq \alpha^2\tau \sum_{\ell=k-\tau}^{k-1} 60\zeta' p \mathbb{E}\left[\delta_\ell^2\right] + 0.5\alpha\tau^2 B. \tag{B.15}$$

Using (B.14) to bound $\mathbb{E}\left[\delta_\ell^2\right]$ above, we further obtain

$$\mathbb{E}\left[\delta_{k,\tau}^2\right] \leq \alpha^2\tau \sum_{\ell=k-\tau}^{k-1} 120\zeta' p \left(\mathbb{E}\left[\delta_{k-\tau}^2\right] + \tau B\right) + \frac{1}{2}\alpha\tau^2 B.$$

Simplifying using $\alpha \leq 1/484\zeta'\tau$, $p < 1$, and $q \geq 2$ yields

$$\mathbb{E}\left[\delta_{k,\tau}^2\right] \leq 120\alpha^2\tau^2 p\zeta' \mathbb{E}\left[\delta_{k-\tau}^2\right] + \alpha^2\tau^2\sigma^2 p\left(\frac{180\zeta'}{N} + 2\alpha^q\right).$$

Using $\delta_{k-\tau}^2 \leq 2\delta_k^2 + 2\delta_{k,\tau}^2$ and $240\alpha^2\tau^2\zeta' \leq 1/2$ to simplify the above inequality, we arrive at the desired result. □

Our next result is the final ingredient needed to prove Theorem 4.2.

Lemma B.4. Define

$$\mathbf{g}_N(\boldsymbol{\theta}_k) \triangleq \frac{1}{N}\sum_{i=1}^{N} b_{i,k}\mathbf{g}_{i,k}(\boldsymbol{\theta}_k),$$

and let $\alpha \leq 1/(484\zeta'\tau)$ and $k \geq 2\tau$. We have

$$\mathbb{E}\left[\psi_k\right] \leq \alpha\tau p\left(3191\zeta'\mathbb{E}\left[\delta_k^2\right] + \sigma^2\left(\frac{2461\zeta'}{N} + 30\alpha^q\right)\right).$$

Proof. We can write $\psi_k = T_1 + T_2 + T_3 + T_4 + T_5$, with

$$\begin{aligned}
T_1 &= \langle \boldsymbol{\theta}_k - \boldsymbol{\theta}_{k-\tau}, \mathbf{g}_N(\boldsymbol{\theta}_k) - \bar{\mathbf{g}}_N(\boldsymbol{\theta}_k)\rangle,\\
T_2 &= \langle \boldsymbol{\theta}_{k-\tau} - \boldsymbol{\theta}^*, \mathbf{g}_N(\boldsymbol{\theta}_{k-\tau}) - \bar{\mathbf{g}}_N(\boldsymbol{\theta}_{k-\tau})\rangle,\\
T_3 &= \langle \boldsymbol{\theta}_{k-\tau} - \boldsymbol{\theta}^*, \mathbf{g}_N(\boldsymbol{\theta}_k) - \mathbf{g}_N(\boldsymbol{\theta}_{k-\tau})\rangle,\\
T_4 &= \langle \boldsymbol{\theta}_{k-\tau} - \boldsymbol{\theta}^*, \bar{\mathbf{g}}_N(\boldsymbol{\theta}_{k-\tau}) - \bar{\mathbf{g}}_N(\boldsymbol{\theta}_k)\rangle,\\
T_5 &= \langle \boldsymbol{\theta}_k - \boldsymbol{\theta}^*, \mathbf{v}_k - \mathbf{g}_N(\boldsymbol{\theta}_k)\rangle.
\end{aligned} \tag{B.16}$$

To bound T_1, observe the following inequalities:

$$\begin{aligned} T_1 &= \langle \boldsymbol{\theta}_k - \boldsymbol{\theta}_{k-\tau}, \mathbf{g}_N(\boldsymbol{\theta}_k) - \bar{\mathbf{g}}_N(\boldsymbol{\theta}_k) \rangle \\ &\overset{(a)}{\leq} \|\boldsymbol{\theta}_k - \boldsymbol{\theta}_{k-\tau}\| \|\mathbf{g}_N(\boldsymbol{\theta}_k) - \bar{\mathbf{g}}_N(\boldsymbol{\theta}_k)\| \\ &\overset{(b)}{\leq} \frac{1}{2\alpha\tau}\|\boldsymbol{\theta}_k - \boldsymbol{\theta}_{k-\tau}\|^2 + \frac{\alpha\tau}{2}\|\mathbf{g}_N(\boldsymbol{\theta}_k) - \bar{\mathbf{g}}_N(\boldsymbol{\theta}_k)\|^2 \\ &\overset{(c)}{\leq} \underbrace{\frac{1}{2\alpha\tau}\|\boldsymbol{\theta}_k - \boldsymbol{\theta}_{k-\tau}\|^2}_{S_1} + \underbrace{\alpha\tau\|\mathbf{g}_N(\boldsymbol{\theta}_k)\|^2}_{S_2} + \underbrace{\alpha\tau\|\bar{\mathbf{g}}_N(\boldsymbol{\theta}_k) - \bar{\mathbf{g}}_N(\boldsymbol{\theta}^*)\|^2}_{S_3}. \end{aligned} \tag{B.17}$$

In the above steps, (a) follows from the Cauchy-Schwarz inequality. For (b), we used the fact that given any two positive numbers x and y, the following holds for any $\eta > 0$:

$$xy \leq \frac{1}{2\eta}x^2 + \frac{\eta}{2}y^2.$$

We used the above inequality with $\eta = \alpha\tau$ to arrive at (b). Finally, for (c), we used the fact that $\bar{\mathbf{g}}(\boldsymbol{\theta}^*) = 0$; hence, $\bar{\mathbf{g}}_N(\boldsymbol{\theta}^*) = 0$. We now proceed to bound the expectations of each of the terms $S_1 - S_3$, starting with S_3. Note that using (C.3), i.e., the Lipschitz property of the TD update directions, we get:

$$\begin{aligned} \|\bar{\mathbf{g}}_N(\boldsymbol{\theta}_k) - \bar{\mathbf{g}}_N(\boldsymbol{\theta}^*)\|^2 &\leq \|\frac{1}{N}\sum_{i=1}^{N} b_{i,k}(\bar{\mathbf{g}}_N(\boldsymbol{\theta}_k) - \bar{\mathbf{g}}_N(\boldsymbol{\theta}^*))\|^2 \\ &\leq \frac{1}{N}\sum_{i=1}^{N} b_{i,k}^2\|\bar{\mathbf{g}}_N(\boldsymbol{\theta}_k) - \bar{\mathbf{g}}_N(\boldsymbol{\theta}^*)\|^2 \\ &\leq \frac{4}{N}\sum_{i=1}^{N} b_{i,k}^2\|\boldsymbol{\theta}_k - \boldsymbol{\theta}^*\|^2. \end{aligned}$$

Taking expectations on each side of the above inequality then yields:

$$\mathbb{E}\left[\alpha\tau\|\bar{\mathbf{g}}_N(\boldsymbol{\theta}_k) - \bar{\mathbf{g}}_N(\boldsymbol{\theta}^*)\|^2\right] \leq 4\alpha\tau p\mathbb{E}\left[\|\boldsymbol{\theta}_k - \boldsymbol{\theta}^*\|^2\right]. \tag{B.18}$$

In arriving at the above inequality, we used the following facts: (i) the randomness in $\boldsymbol{\theta}_k$ depends on all the sources of randomness in our model up to time $k-1$; (ii) the Bernoulli packet-drop random variables $\{b_{i,k}\}_{i\in[N]}$ are independent of all the sources of randomness up to time $k-1$. Hence, for each $i \in [N]$, $\mathbb{E}\left[b_{i,k}^2\|\boldsymbol{\theta}_k - \boldsymbol{\theta}^*\|^2\right] = \mathbb{E}\left[b_{i,k}^2\right]\mathbb{E}\left[\|\boldsymbol{\theta}_k - \boldsymbol{\theta}^*\|^2\right] = p\mathbb{E}\left[\|\boldsymbol{\theta}_k - \boldsymbol{\theta}^*\|^2\right]$.

Next, to bound $\mathbb{E}[S_1]$, note that $\mathbb{E}\left[\frac{1}{2\alpha\tau}\|\boldsymbol{\theta}_k - \boldsymbol{\theta}_{k-\tau}\|^2\right]$ can be directly bounded using

Lemma B.3 in the following way:

$$\frac{1}{2\alpha\tau}\mathbb{E}\left[\|\boldsymbol{\theta}_k - \boldsymbol{\theta}_{k-\tau}\|^2\right] \leq 240\alpha\tau p\zeta'\mathbb{E}\left[\|\boldsymbol{\theta}_k - \boldsymbol{\theta}^*\|^2\right] + \alpha\tau p\sigma^2\left(\frac{180\zeta'}{N} + 2\alpha^q\right). \tag{B.19}$$

Finally, the only term that remains to be bounded is $\mathbb{E}\left[\|\mathbf{g}_N(\boldsymbol{\theta}_k)\|^2\right]$. Note that we can write:

$$\begin{aligned}\|\mathbf{g}_N(\boldsymbol{\theta}_k)\|^2 &\leq \frac{2}{N^2}(T_1' + T_2') \text{ with}\\ T_1' &= \|\sum_{i=1}^N b_{i,k}\mathbf{g}_{i,k}(\boldsymbol{\theta}^*)\|^2, \text{ and}\\ T_2' &= \|\sum_{i=1}^N b_{i,k}(\mathbf{g}_{i,k}(\boldsymbol{\theta}_k) - \mathbf{g}_{i,k}(\boldsymbol{\theta}^*))\|^2.\end{aligned} \tag{B.20}$$

Observe that T_1' and T_2' above correspond exactly to the terms T_1 and T_2 in the proof of Lemma B.2. Thus, they can be bounded as follows:

$$\begin{aligned}\mathbb{E}\left[T_1'\right] &\leq 16\sigma^2 Np + 4N^2p^2\sigma^2\alpha^{2q}.\\ \mathbb{E}\left[T_2'\right] &\leq 4pN^2\mathbb{E}\left[\|\boldsymbol{\theta}_k - \boldsymbol{\theta}^*\|^2\right].\end{aligned} \tag{B.21}$$

So, plugging (B.18), (B.19), (B.20), and the above bound into (B.17), we get the final bound on $\mathbb{E}\left[T_1\right]$ as follows:

$$\mathbb{E}\left[T_1\right] \leq 304\alpha\tau\zeta' p\mathbb{E}\left[\delta_k^2\right] + \alpha\tau p\sigma^2\left(\frac{300\zeta'}{N} + 3\alpha^q\right).$$

Next we bound $\mathbb{E}\left[T_3\right]$ and $\mathbb{E}\left[T_4\right]$. Observe that:

$$\begin{aligned}\mathbb{E}\left[T_3\right] &= \frac{1}{N}\sum_{i=1}^N \mathbb{E}\left[b_{i,k}\langle\boldsymbol{\theta}_{k-\tau} - \boldsymbol{\theta}^*, (\mathbf{g}_{i,k}(\boldsymbol{\theta}_k) - \mathbf{g}_{i,k}(\boldsymbol{\theta}_{k-\tau}))\rangle\right]\\ &\leq p\mathbb{E}\left[\delta_{k-\tau}\frac{1}{N}\sum_{i=1}^N\|\mathbf{g}_{i,k}(\boldsymbol{\theta}_k) - \mathbf{g}_{i,k}(\boldsymbol{\theta}_{k-\tau})\|\right]\\ &\overset{(C.3)}{\leq} 2p\mathbb{E}\left[\delta_{k-\tau}\delta_{k,\tau}\right]\\ &\leq \frac{\alpha\tau p}{2}\mathbb{E}\left[\delta_{k-\tau}^2\right] + \frac{2p}{\alpha\tau}\mathbb{E}\left[\delta_{k,\tau}^2\right].\end{aligned}$$

Using $\delta_{k-\tau}^2 \leq 2\delta_k^2 + 2\delta_{k,\tau}^2$ and Lemma B.3, we then obtain:

$$\mathbb{E}\left[T_3\right] \leq 1441\alpha\tau p\zeta'\mathbb{E}\left[\delta_k^2\right] + 6\alpha\tau p\sigma^2\left(\frac{180\zeta'}{N} + 2\alpha^q\right).$$

Using the same process, we can derive the exact same bound for $\mathbb{E}[T_4]$. We now bound $\mathbb{E}[T_2]$. For ease of notation, let us define $\mathcal{F}_{k,\tau} = (\{o_{i,k-\tau}\}_{i=1}^N, \boldsymbol{\theta}_{k-\tau})$. Observe:

$$\begin{aligned}
\mathbb{E}[T_2] &= \mathbb{E}[\mathbb{E}[T_2|\mathcal{F}_{k,\tau}]] \\
&= \mathbb{E}[\langle \boldsymbol{\theta}_{k-\tau} - \boldsymbol{\theta}^*, \frac{p}{N}\sum_{i=1}^N (\mathbb{E}[\mathbf{g}_{i,k}(\boldsymbol{\theta}_{k-\tau}, o_{i,k})|\mathcal{F}_{k,\tau}] - \bar{\mathbf{g}}(\boldsymbol{\theta}_{k-\tau}))\rangle] \\
&\leq \mathbb{E}\left[\delta_{k-\tau}\frac{p}{N}\sum_{i=1}^N \eta_{k,\tau}^{(i)}(\boldsymbol{\theta}_{k-\tau})\right] \\
&\leq p\alpha^q \mathbb{E}[\delta_{k-\tau}(1 + \|\boldsymbol{\theta}_{k-\tau}\|)],
\end{aligned}$$

where in the last step, we made use of the mixing property. Since $\alpha < 1$, we have $\delta_{k-\tau}(\delta_{k-\tau} + 2\sigma) \leq \frac{\delta_{k-\tau}^2}{\alpha} + 2\sigma\delta_{k-\tau} + \alpha\sigma^2 = \left(\frac{\delta_{k-\tau}}{\sqrt{\alpha}} + \sqrt{\alpha}\sigma\right)^2 \leq 2\left(\frac{\delta_{k-\tau}^2}{\alpha} + \alpha\sigma^2\right)$. Using $q \geq 2$, we obtain:

$$\begin{aligned}
\mathbb{E}[T_2] &\leq 2p\alpha^q \mathbb{E}\left[\frac{1}{\alpha}\delta_{k-\tau}^2 + \alpha\sigma^2\right] \\
&\leq 2p\alpha\mathbb{E}\left[\delta_{k-\tau}^2\right] + 2p\alpha^{q+1}\sigma^2.
\end{aligned} \tag{B.22}$$

Using $\delta_{k-\tau}^2 \leq 2\delta_k^2 + 2\delta_{k,\tau}^2$ and Lemma B.3, and then simplifying yields:

$$\mathbb{E}[T_2] \leq 5\alpha\tau p\zeta'\mathbb{E}\left[\delta_k^2\right] + \alpha\tau p\sigma^2\left(\frac{\zeta'}{N} + 3\alpha^q\right). \tag{B.23}$$

Finally, to bound T_5, let $\mathcal{F}_k = \{\{o_{i,k}\}_{i=1}^N, \boldsymbol{\theta}_k\}$. We have

$$\mathbb{E}[T_5] = \mathbb{E}\left[\langle \boldsymbol{\theta}_k - \boldsymbol{\theta}^*, \underbrace{\mathbb{E}[\mathbf{v}_k - \mathbf{g}_N(\boldsymbol{\theta}_k)|\mathcal{F}_k]}_{T_{51}}\rangle\right]. \tag{B.24}$$

Note that $T_{51} = \frac{p}{N}\sum_{i=1}^N \mathbb{E}[\mathbf{h}_{i,k}(\boldsymbol{\theta}_k) - \mathbf{g}_{i,k}(\boldsymbol{\theta}_k)|\mathcal{F}_k] = 0$, based on the unbiasedness of $\mathcal{Q}(\cdot)$. Thus, $\mathbb{E}[T_5] = 0$. Collecting the bounds on $T_1 - T_5$ concludes the proof. □

With the help of the auxiliary lemmas provided above, we are now ready to prove our main result, i.e., Theorem 4.2.

Proof of Theorem 4.2. Setting $\alpha \leq \frac{1}{484\zeta'\tau}$, we can apply the bounds in Lemmas C.1, B.2, and B.4 to (B.5). This yields:

$$\mathbb{E}\left[\delta_{k+1}^2\right] \leq \mathbb{E}\left[\delta_k^2\right] - \alpha p(2(1-\gamma)\omega - 6446\alpha\tau\zeta')\mathbb{E}\left[\delta_k^2\right] + 5162\frac{\alpha^2\tau p\sigma^2\zeta'}{N} + 61\alpha^{(2+q)}\tau p\sigma^2. \tag{B.25}$$

For $\alpha \leq \frac{\omega(1-\gamma)}{C_0\tau\zeta'}$ with $C_0 = 6446$, we then obtain:

$$\mathbb{E}\left[\delta_{k+1}^2\right] \leq (1-\alpha\omega(1-\gamma)p)\mathbb{E}\left[\delta_k^2\right] + 5162\frac{\alpha^2\tau p\sigma^2\zeta}{N} + 61\alpha^{(2+q)}\tau p\sigma^2. \tag{B.26}$$

Iterating the last inequality, we have $\forall k \geq 2\tau$:

$$\mathbb{E}\left[\delta_k^2\right] \leq \rho^{k-2\tau}\mathbb{E}\left[\delta_{2\tau}^2\right] + \frac{\tau\sigma^2}{\omega(1-\gamma)}\left(\frac{C_2\alpha\zeta'}{N} + C_3\alpha^3\right),$$

where $\rho = (1-\alpha\omega(1-\gamma)p)$, $C_2 = 5162$, $C_3 = 61$, and we set $q = 2$. It only remains to show that with our choice of α, $\mathbb{E}\left[\delta_{2\tau}^2\right] = O(\delta_0^2 + \sigma^2)$. This follows from some simple algebra and steps similar to those in the proof of Lemma B.3. We provide these steps below for completeness. Note that, defining $T' = \|\sum_{i=1}^N b_{i,k}\mathbf{g}_{i,k}(\boldsymbol{\theta}_k)\|^2$, and using (C.4),

$$\mathbb{E}\left[T'\right] \leq 8N^2p\mathbb{E}\left[2\delta_k^2 + 3\sigma^2\right] \leq N^2(16p\mathbb{E}\left[\delta_k^2\right] + 24p\sigma^2).$$

Letting T_3 be as defined in (B.7), note that

$$\mathbb{E}\left[\|\mathbf{v}_k\|^2\right] \leq \frac{2}{N^2}\mathbb{E}\left[T' + T_3\right] \leq 64p\zeta'\mathbb{E}\left[\delta_k^2\right] + 96p\zeta'\sigma^2.$$

Plugging this inequality into (B.53) and iterating,

$$\mathbb{E}\left[\delta_k^2\right] \leq (1+129\alpha\zeta')^k\delta_0^2 + 192\alpha p\sigma^2\sum_{j=0}^{k-1}(1+129\alpha\zeta')^j.$$

Using the same arguments used to arrive at (B.14), we have

$$\begin{aligned}\mathbb{E}\left[\delta_{2\tau}^2\right] &\leq (1+129\alpha\zeta')^{2\tau}\delta_0^2 + 768\alpha\tau p\zeta'\sigma^2 \\ &\leq 2\delta_0^2 + p\sigma^2,\end{aligned} \tag{B.27}$$

where we used the fact that $\alpha\tau \leq \frac{\omega(1-\gamma)}{6446\zeta'} \leq \frac{1}{1032\zeta'}$. Given that $\alpha\tau \leq \frac{1}{C_0} \leq \frac{1}{4}$, from Bernoulli's inequality, we have that $(1-\alpha)^{2\tau} \geq 1 - 2\alpha\tau \geq \frac{1}{2}$. Thus, observe that $(1-\alpha\omega(1-\gamma)p)^{-2\tau} \leq (1-\alpha)^{-2\tau} \leq 2$. This concludes the proof. □

We now provide the proof of Corollary 4.1.

Proof of Corollary 4.1. We first recall the main result of Theorem 4.2, i.e., the

following bound:

$$\mathbb{E}\left[\delta_T^2\right] \leq \underbrace{(1-\alpha\omega(1-\gamma)p)^T C_1}_{T_1} + \underbrace{\frac{C_2\alpha\zeta'\tau\sigma^2}{\omega(1-\gamma)N}}_{T_2} + \underbrace{\frac{C_3\alpha^3\tau\sigma^2}{\omega(1-\gamma)}}_{T_3}. \tag{B.28}$$

Let us also recall the choice of step-size α and number of iterations T from Corollary 4.1:

$$\alpha = \frac{\log NT}{\omega(1-\gamma)pT}, \quad \text{and} \quad T \geq \frac{2C_0N\tau\zeta'\log NT}{\omega^2(1-\gamma)^2p}. \tag{B.29}$$

To simplify the first term in Eq. (B.28), we use the fact that for all $x \in (0,1)$, it holds that $(1-x) \leq e^{-x}$. Using this in conjunction with the choice of α in (B.29) yields the following bound on T_1 in Eq. (B.28):

$$T_1 = O\left(\frac{\max\{\delta_0^2, \sigma^2\}}{NT}\right).$$

To bound T_2, we simply substitute the choice of α in Eq. (B.29). For T_3, we first substitute the choice of α to obtain:

$$T_3 = \frac{C_3\tau\sigma^2(\log NT)^3}{\omega^4(1-\gamma)^4p^3T^3}.$$

From our choice of T in Eq. (B.29), the following hold:

$$\frac{\tau\log(NT)}{p\omega^2(1-\gamma)^2T} \leq 1, \quad \frac{N\log(NT)}{Tp} \leq 1.$$

Using these two inequalities, we immediately note that:

$$T_3 = O\left(\frac{\sigma^2\log(NT)}{p\omega^2(1-\gamma)^2NT}\right).$$

Combining the individual bounds on T_1, T_2, and T_3 leads to Eq. (4.14). Let us complete our derivation with a couple of other points. First, straightforward calculations suffice to check that the choice of α and T in Eq. (B.29) meet the requirement on α in the statement of Theorem 4.2. Finally, recall from the discussion following Definition 5.2.1 that the mixing time τ_ϵ satisfies:

$$\tau_\epsilon \leq K\log(1/\epsilon),$$

for some constant $K \geq 1$. Throughout our analysis, we set $\epsilon = \alpha^2$, and then dropped

the dependence of τ on ϵ for notational convenience. Plugging in the choice of α from Eq. (B.29), we obtain:

$$\tau \leq 2K \log\left(\frac{\omega(1-\gamma)pT}{\log(NT)}\right) \leq 2K \log\left(\omega(1-\gamma)pT\right),$$

for $NT \geq e$. The point of the above calculation is to explicitly demonstrate that one can indeed meet the requirement on T in Eq. (B.29) for large enough T.

B.2 Proof of Theorem 4.3

In this appendix, we will provide the detailed proof of Theorem 4.3. We start by introducing some definitions and preliminary results. To lighten the notation, let us define

$$\begin{aligned} \eta_{k,\tau}^{(i)}(\boldsymbol{\theta}) &\triangleq \|\mathbb{E}\left[\mathbf{g}_{i,k}(\boldsymbol{\theta}, o_{i,k})|o_{i,k-\tau}\right] - \bar{\mathbf{g}}(\boldsymbol{\theta})\|, \forall k \geq \tau, \forall \boldsymbol{\theta} \in \mathbb{R}^d, \forall i \in [N], \\ \delta_{k,\tau} &\triangleq \|\boldsymbol{\theta}_k - \boldsymbol{\theta}_{k-\tau}\|, \forall k \geq \tau. \end{aligned} \tag{B.30}$$

Next, we summarize in one lemma a result from [19] that we will use in our analysis.

Lemma B.5. The following holds $\forall \boldsymbol{\theta} \in \mathbb{R}^d$:

$$\langle \boldsymbol{\theta}^* - \boldsymbol{\theta}, \bar{\mathbf{g}}(\boldsymbol{\theta}) \rangle \geq \omega(1-\gamma)\|\boldsymbol{\theta}^* - \boldsymbol{\theta}\|^2.$$

We will also use the fact that the random TD update directions and their steady-state versions are 2-Lipschitz [19], i.e., $\forall i \in [N], \forall k \in \mathbb{N}$, and $\forall \boldsymbol{\theta}, \boldsymbol{\theta}' \in \mathbb{R}^d$, we have:

$$\begin{aligned} &\|\bar{\mathbf{g}}(\boldsymbol{\theta}) - \bar{\mathbf{g}}(\boldsymbol{\theta}')\| \leq 2\|\boldsymbol{\theta} - \boldsymbol{\theta}'\|, \text{ and} \\ &\|\mathbf{g}_{i,k}(\boldsymbol{\theta}) - \mathbf{g}_{i,k}(\boldsymbol{\theta}')\| \leq 2\|\boldsymbol{\theta} - \boldsymbol{\theta}'\|. \end{aligned} \tag{B.31}$$

From [135], we further have

$$\|\mathbf{g}_{i,k}(\boldsymbol{\theta})\| \leq 2\|\boldsymbol{\theta}\| + 2\bar{r}, \forall i \in [N], \forall k \in \mathbb{N}, \forall \boldsymbol{\theta} \in \mathbb{R}^d. \tag{B.32}$$

Given that $(x+y)^2 \leq 2(x^2+y^2), \forall x, y \in \mathbb{R}$, and the definition of σ, we will often use the following inequality:

$$\|\mathbf{g}_{i,k}(\boldsymbol{\theta})\|^2 \leq 4(\|\boldsymbol{\theta}\| + \bar{r})^2 \leq 8(\|\boldsymbol{\theta}\|^2 + \bar{r}^2) \leq 8(\|\boldsymbol{\theta}\|^2 + \sigma^2). \tag{B.33}$$

In what follows, $\tau = \tau_\epsilon$ with $\epsilon = \alpha^2$. We now provide an intuitive outline of the proof,

highlighting the challenges and the key technical steps in establishing Theorem 4.3.

Outline of the Proof

The proof relies on analyzing the following recursion, which, in turn, follows directly from the update rule of `OAC-FedTD`:

$$\delta_{k+1}^2 = \delta_k^2 - 2\alpha\langle \mathbf{v}_k, \boldsymbol{\theta}^* - \boldsymbol{\theta}_k\rangle + \alpha^2\|\mathbf{v}_k\|^2. \tag{B.34}$$

Let $\bar{\mathbf{g}}_N(\boldsymbol{\theta}_k) \triangleq \frac{1}{N}\sum_{i=1}^N h_{i,k}\bar{\mathbf{g}}(\boldsymbol{\theta}_k)$ and $\mathbf{g}_{h,k}(\boldsymbol{\theta}_k) \triangleq \frac{1}{N}\sum_{i=1}^N h_{i,k}\mathbf{g}_{i,k}(\boldsymbol{\theta}_k)$. Taking expectation on both sides of (B.34),

$$\begin{aligned}\mathbb{E}\left[\delta_{k+1}^2\right] = \mathbb{E}\left[\delta_k^2\right] &- 2\alpha\mathbb{E}\left[\langle \bar{\mathbf{g}}_N(\boldsymbol{\theta}_k), \boldsymbol{\theta}^* - \boldsymbol{\theta}_k\rangle\right] \\ &- 2\alpha\mathbb{E}\left[\langle \mathbf{g}_{h,k}(\boldsymbol{\theta}_k) - \bar{\mathbf{g}}_N(\boldsymbol{\theta}_k), \boldsymbol{\theta}^* - \boldsymbol{\theta}_k\rangle\right] \\ &- 2\alpha\mathbb{E}\left[\langle \mathbf{w}_k, \boldsymbol{\theta}^* - \boldsymbol{\theta}_k\rangle\right] + \alpha^2\|\mathbf{v}_k\|^2.\end{aligned} \tag{B.35}$$

Now note that $\mathbb{E}\left[\langle \mathbf{w}_k, \boldsymbol{\theta}_k - \boldsymbol{\theta}^*\rangle\right] = \langle \mathbb{E}\left[\mathbf{w}_k\right], \mathbb{E}\left[\boldsymbol{\theta}_k - \boldsymbol{\theta}^*\right]\rangle = 0$, using the fact that the measurement noise at iteration k and the iterate $\boldsymbol{\theta}_k$ are independent, and $\mathbb{E}\left[\mathbf{w}_k\right] = \mathbf{0}$. Moreover, using the fact that the distortion $h_{i,k}$ of agent i at iteration k and the parameter $\boldsymbol{\theta}_k$ are independent, we obtain

$$\mathbb{E}\left[\langle \bar{\mathbf{g}}_N(\boldsymbol{\theta}_k), \boldsymbol{\theta}^* - \boldsymbol{\theta}_k\rangle\right] = \frac{1}{N}\sum_{i=1}^N \mathbb{E}\left[h_{i,k}\right]\mathbb{E}\left[\langle \bar{\mathbf{g}}(\boldsymbol{\theta}_k), \boldsymbol{\theta}^* - \boldsymbol{\theta}_k\rangle\right] = m_h\mathbb{E}\left[\langle \bar{\mathbf{g}}(\boldsymbol{\theta}_k), \boldsymbol{\theta}^* - \boldsymbol{\theta}_k\rangle\right]. \tag{B.36}$$

Based on the above discussion, we can write

$$\mathbb{E}\left[\delta_{k+1}^2\right] = \mathbb{E}\left[\delta_k^2\right] - 2\alpha m_h\langle \bar{\mathbf{g}}(\boldsymbol{\theta}_k), \boldsymbol{\theta}^* - \boldsymbol{\theta}_k\rangle + 2\alpha\mathbb{E}\left[\langle \mathbf{g}_{h,k}(\boldsymbol{\theta}_k) - \bar{\mathbf{g}}_N(\boldsymbol{\theta}_k), \boldsymbol{\theta}_k - \boldsymbol{\theta}^*\rangle\right] + \alpha^2\|\mathbf{v}_k\|^2. \tag{B.37}$$

Now define

$$\psi_k \triangleq \langle \mathbf{g}_{h,k}(\boldsymbol{\theta}_k) - \bar{\mathbf{g}}_N(\boldsymbol{\theta}_k), \boldsymbol{\theta}_k - \boldsymbol{\theta}^*\rangle. \tag{B.38}$$

Using Lemma C.1, we then obtain

$$\mathbb{E}\left[\delta_{k+1}^2\right] \leq \mathbb{E}\left[\delta_k^2\right] - 2\alpha m_h(1-\gamma)\omega\mathbb{E}\left[\delta_k^2\right] + 2\alpha\mathbb{E}\left[\psi_k\right] + \alpha^2\mathbb{E}\left[\|\mathbf{v}_k\|^2\right]. \tag{B.39}$$

The most challenging part of the analysis is in bounding $\mathbb{E}\left[\|\mathbf{v}_k\|^2\right]$ and $\mathbb{E}\left[\psi_k\right]$ while guaranteeing a convergence speedup w.r.t. the number of agents. In fact, even without the channel effects, this is highly non-trivial. Let us elaborate on this point. First, in standard stochastic optimization analyses, $\mathbb{E}\left[\psi_k\right]$ would vanish under the unbiasedness

assumption of the stochastic gradient oracle. However, in our case, since the Markovian observations are temporally coupled, $\mathbb{E}[\psi_k]$ does not vanish. To work around this difficulty, the bounding techniques in the centralized setting, like the ones in [19] and [135], use mixing-time arguments in conjunction with equation (C.5). Unfortunately, directly appealing to such techniques will fail to provide the desired convergence speedup that we seek in our multi-agent setting. The key technical step of our proof is providing a bound for $\mathbb{E}\left[\|\mathbf{v}_k\|^2\right]$ of the following form:

$$\mathbb{E}\left[\|\mathbf{v}_k\|^2\right] \leq O\left(p_h\right)\mathbb{E}\left[\delta_k^2\right] + O\left(\frac{\sigma^2 p_h}{N}\right) + O\left(\sigma^2 m_h^2 \alpha^4\right) + O\left(\frac{\tilde{\sigma}_{\mathbf{w}}^2 d}{N^2}\right). \tag{B.40}$$

We derive this bound by appealing to the Lipschitz properties of $\mathbf{g}_{i,k}(\boldsymbol{\theta}_k)$ and performing some careful manipulations that allow us to exploit the mixing property of the Markov chain. Leveraging this key result, our next main step is to obtain a bound on $\mathbb{E}\left[\delta_{k,\tau}^2\right]$ of the following form:

$$\mathbb{E}\left[\delta_{k,\tau}^2\right] \leq O\left(\alpha^2\tau^2 p_h\right)\mathbb{E}\left[\delta_k^2\right] + O\left(\alpha^2\tau^2\frac{p_h\sigma^2}{N}\right) + O\left(\tau^2\sigma^2\alpha^4\right) + O\left(\alpha^2\tau^2\frac{\tilde{\sigma}_{\mathbf{w}}^2 d}{N^2}\right). \tag{B.41}$$

This result, derived in Lemma B.7, turns out to play an essential role in bounding $\mathbb{E}[\psi_k]$. In particular, using Lemma B.7, we show that

$$\mathbb{E}[\psi_k] \leq O\left(\alpha\tau p_h\right)\mathbb{E}\left[\delta_k^2\right] + O\left(\frac{\alpha\tau p_h\sigma^2}{N}\right) + O\left(\tau p_h\sigma^2\alpha^3\right) + O\left(\frac{\alpha\tau\tilde{\sigma}_{\mathbf{w}}^2 d}{N^2}\right).$$

This final ingredient is established in Lemma B.8. Combining these bounds leads to Theorem 4.3. In what follows, we flesh out the above argument.

Auxiliary Lemmas

We state and prove three lemmas that are instrumental to the proof of Theorem 4.3. In particular, these three results allow us to bound the terms $\mathbb{E}\left[\|\mathbf{v}_k\|^2\right]$ and $\mathbb{E}[\psi_k]$ in (B.39). We start by providing a bound on $\mathbb{E}\left[\|\mathbf{v}_k\|^2\right]$ of the form illustrated in (B.40). To that end, we state and prove the following lemma.

Lemma B.6. For $k \geq \tau$, we have

$$\mathbb{E}\left[\|\mathbf{v}_k\|^2\right] \leq 8p_h\mathbb{E}\left[\delta_k^2\right] + 32\frac{\sigma^2 p_h}{N} + 8\sigma^2 m_h^2\alpha^4 + \frac{\tilde{\sigma}_{\mathbf{w}}^2 d}{N^2}. \tag{B.42}$$

Proof. Let us start by noting that the randomness in $\boldsymbol{\theta}_k$ is induced by $\{h_{i,\ell}\}_{i\in[N],\ell\in[k-1]}$, $\{o_{i,\ell}\}_{i\in[N],\ell\in[k-1]}$,

and $\{\mathbf{w}_\ell\}_{\ell\in[k-1]}$. Based on our assumptions on the noise process, $\mathbf{w}_k$ is independent of each of these random variables and also independent of $\{h_{i,k}\}_{i\in[N]}$ and $\{o_{i,k}\}_{i\in[N]}$. Using these observations with the fact that $\mathbb{E}[\mathbf{w}_k] = \mathbf{0}$, we immediately obtain $\mathbb{E}[\langle \mathbf{g}_{h,k}(\boldsymbol{\theta}_k), \mathbf{w}_k\rangle] = \langle \mathbb{E}[\mathbf{g}_{h,k}(\boldsymbol{\theta}_k)], \mathbb{E}[\mathbf{w}_k]\rangle = 0$. This yields:

$$\mathbb{E}\left[\|\mathbf{v}_k\|^2\right] = \mathbb{E}\left[\|\mathbf{g}_{h,k}(\boldsymbol{\theta}_k)\|^2\right] + \mathbb{E}\left[\|\mathbf{w}_k\|^2\right]. \tag{B.43}$$

Now, note that in the centralized/single-agent TD analysis, $\|\mathbf{g}_{h,k}(\boldsymbol{\theta}_k)\|^2$ could be bounded using (C.4), and this would provide a term of the form $O(\delta_k^2) + O(\sigma^2)$. This approach would, however, fail to provide a linear convergence speedup with the number of agents, N. We will show how through a finer analysis, we can establish a tighter bound. We start by writing

$$\begin{aligned}\|\mathbf{g}_{h,k}(\boldsymbol{\theta}_k)\|^2 &= \|\mathbf{g}_{h,k}(\boldsymbol{\theta}_k) - \mathbf{g}_{h,k}(\boldsymbol{\theta}^*) + \mathbf{g}_{h,k}(\boldsymbol{\theta}^*)\|^2 \\ &\leq \frac{2}{N^2}\left(T_1 + T_2\right),\end{aligned} \tag{B.44}$$

where T_1 and T_2 are as follows:

$$T_1 = \|\sum_{i=1}^N h_{i,k}\mathbf{g}_{i,k}(\boldsymbol{\theta}^*)\|^2, \quad T_2 = \|\sum_{i=1}^N h_{i,k}(\mathbf{g}_{i,k}(\boldsymbol{\theta}_k) - \mathbf{g}_{i,k}(\boldsymbol{\theta}^*))\|^2. \tag{B.45}$$

We proceed to bound T_1 first. We express $T_1 = T_{11} + T_{12}$, with

$$\begin{aligned}T_{11} &= \sum_{i=1}^N h_{i,k}^2\|\mathbf{g}_{i,k}(\boldsymbol{\theta}^*)\|^2, \text{ and} \\ T_{12} &= \sum_{\substack{i,j=1\\ i\neq j}}^N h_{i,k}h_{j,k}\langle \mathbf{g}_{i,k}(\boldsymbol{\theta}^*), \mathbf{g}_{j,k}(\boldsymbol{\theta}^*)\rangle.\end{aligned} \tag{B.46}$$

Using (C.5) and the fact that $\|\boldsymbol{\theta}^*\| \leq \sigma$, we obtain

$$\|\mathbf{g}_{i,k}(\boldsymbol{\theta}^*)\|^2 \leq 16\sigma^2, \tag{B.47}$$

and hence, $T_{11} \leq 16\sigma^2\sum_{i=1}^N h_{i,k}^2$. Taking expectations, we thus obtain

$$\mathbb{E}[T_{11}] \leq 16\sigma^2\mathbb{E}\left[\sum_{i=1}^N h_{i,k}^2\right] = 16\sigma^2 N(m_h^2 + \sigma_h^2) \leq 16N\sigma^2 p_h.$$

Next, to bound the cross-terms in T_{12}, we will exploit the mixing property in Defini-

tion 5.2.1. To that end, we write

$$\begin{aligned}
\mathbb{E}\left[T_{12}\right] &= \sum_{\substack{i,j=1\\ i\neq j}}^{N} \mathbb{E}\left[h_{i,k}h_{j,k}\langle \mathbf{g}_{i,k}(\boldsymbol{\theta}^*), \mathbf{g}_{j,k}(\boldsymbol{\theta}^*)\rangle\right] \\
&\overset{(a)}{=} \sum_{\substack{i,j=1\\ i\neq j}}^{N} \mathbb{E}\left[h_{i,k}h_{j,k}\right] \mathbb{E}\left[\langle \mathbf{g}_{i,k}(\boldsymbol{\theta}^*), \mathbf{g}_{j,k}(\boldsymbol{\theta}^*)\rangle\right] \\
&\overset{(b)}{=} \sum_{\substack{i,j=1\\ i\neq j}}^{N} \mathbb{E}\left[h_{i,k}\right] \mathbb{E}\left[h_{j,k}\right] \left\langle \mathbb{E}\left[\mathbf{g}_{i,k}(\boldsymbol{\theta}^*)\right], \mathbb{E}\left[\mathbf{g}_{j,k}(\boldsymbol{\theta}^*)\right]\right\rangle \\
&\overset{(c)}{=} m_h^2 \sum_{\substack{i,j=1\\ i\neq j}}^{N} \left\langle \mathbb{E}\left[\mathbb{E}\left[\mathbf{g}_{i,k}(\boldsymbol{\theta}^*)|o_{i,k-\tau}\right] - \bar{\mathbf{g}}(\boldsymbol{\theta}^*)\right], \mathbb{E}\left[\mathbb{E}\left[\mathbf{g}_{j,k}(\boldsymbol{\theta}^*)|o_{j,k-\tau}\right] - \bar{\mathbf{g}}(\boldsymbol{\theta}^*)\right]\right\rangle \\
&\overset{(d)}{\leq} m_h^2 \sum_{\substack{i,j=1\\ i\neq j}}^{N} \left\|\mathbb{E}\left[\mathbb{E}\left[\mathbf{g}_{i,k}(\boldsymbol{\theta}^*)|o_{i,k-\tau}\right] - \bar{\mathbf{g}}(\boldsymbol{\theta}^*)\right]\right\| \left\|\mathbb{E}\left[\mathbb{E}\left[\mathbf{g}_{j,k}(\boldsymbol{\theta}^*)|o_{j,k-\tau}\right] - \bar{\mathbf{g}}(\boldsymbol{\theta}^*)\right]\right\| \\
&\overset{(e)}{\leq} m_h^2 \sum_{\substack{i,j=1\\ i\neq j}}^{N} \mathbb{E}\left[\underbrace{\left\|\mathbb{E}\left[\mathbf{g}_{i,k}(\boldsymbol{\theta}^*)|o_{i,k-\tau}\right] - \bar{\mathbf{g}}(\boldsymbol{\theta}^*)\right\|}_{\eta_{k,\tau}^{(i)}(\boldsymbol{\theta}^*)}\right] \mathbb{E}\left[\underbrace{\left\|\mathbb{E}\left[\mathbf{g}_{j,k}(\boldsymbol{\theta}^*)|o_{j,k-\tau}\right] - \bar{\mathbf{g}}(\boldsymbol{\theta}^*)\right\|}_{\eta_{k,\tau}^{(j)}(\boldsymbol{\theta}^*)}\right],
\end{aligned}$$

where (a) follows from the independence between the channel distortion gains and the Markovian tuples; (b) follows from the independence between $h_{i,k}$ and $h_{j,k}$ for $i \neq j$, and between $o_{i,k}$ and $o_{j,k}$ for $i \neq j$; (c) follows from the fact that $\bar{\mathbf{g}}(\boldsymbol{\theta}^*) = \mathbf{0}$ [19]; (d) is a consequence of the Cauchy-Schwarz inequality; and (e) follows from Jensen's inequality. Now observe that:

$$\mathbb{E}\left[\eta_{k,\tau}^{(i)}(\boldsymbol{\theta}^*)\right] \times \mathbb{E}\left[\eta_{k,\tau}^{(j)}(\boldsymbol{\theta}^*)\right] \leq \left(\alpha^2(1 + \|\boldsymbol{\theta}^*\|)\right)^2 \leq 4\sigma^2\alpha^4. \tag{B.48}$$

In the step above, we used the mixing property by noting that $k \geq \tau$. We therefore obtain that $\mathbb{E}\left[T_{12}\right] \leq 4N^2 m_h^2 \sigma^2 \alpha^4$. Combining the bounds for $\mathbb{E}\left[T_{11}\right]$ and $\mathbb{E}\left[T_{12}\right]$ thus yields:

$$\mathbb{E}\left[T_1\right] \leq 16\sigma^2 N p_h + 4N^2 m_h^2 \sigma^2 \alpha^4. \tag{B.49}$$

Now, using (C.3), we see that

$$\begin{aligned}\mathbb{E}\left[T_2\right] &\leq N\sum_{i=1}^{N}\mathbb{E}\left[h_{i,k}^2\|\mathbf{g}_{i,k}(\boldsymbol{\theta}_k)-\mathbf{g}_{i,k}(\boldsymbol{\theta}^*)\|^2\right]\\ &\leq 4N\mathbb{E}\left[\delta_k^2\right]\sum_{i=1}^{N}\mathbb{E}\left[h_{i,k}^2\right]=4p_hN^2\mathbb{E}\left[\delta_k^2\right].\end{aligned} \tag{B.50}$$

Combining all the bounds above, we conclude that

$$\mathbb{E}\left[\|\mathbf{g}_{h,k}(\boldsymbol{\theta}_k)\|^2\right]\leq 8p_h\mathbb{E}\left[\delta_k^2\right]+32\frac{\sigma^2p_h}{N}+8\sigma^2m_h^2\alpha^4. \tag{B.51}$$

The claim of the lemma then follows from the above bound and by noting that $\mathbb{E}\left[\|\mathbf{w}_k\|^2\right]=\frac{\tilde{\sigma}_{\mathbf{w}}^2d}{N^2}$. □

Our next key result is the following.

Lemma B.7. Let $k\geq 2\tau$ and $\alpha\leq\frac{1}{68\tau p_h}$. We then have

$$\mathbb{E}\left[\delta_{k,\tau}^2\right]\leq 64\alpha^2\tau^2p_h\mathbb{E}\left[\delta_k^2\right]+96\alpha^2\tau^2\frac{p_h\sigma^2}{N}+4\alpha^4\tau^2\sigma^2+4\alpha^2\tau^2\frac{\tilde{\sigma}_{\mathbf{w}}^2d}{N^2}. \tag{B.52}$$

Proof. We start by writing

$$\begin{aligned}\delta_{k+1}^2=\delta_k^2-2\alpha\langle\mathbf{v}_k,\boldsymbol{\theta}^*-\boldsymbol{\theta}_k\rangle+\alpha^2\|\mathbf{v}_k\|^2&\leq(1+\alpha)\delta_k^2+(\alpha+\alpha^2)\|\mathbf{v}_k\|^2\\ &\leq(1+\alpha)\delta_k^2+2\alpha\|\mathbf{v}_k\|^2.\end{aligned} \tag{B.53}$$

Now using Lemma B.6, we have

$$\begin{aligned}\mathbb{E}\left[\delta_{k+1}^2\right]&\leq(1+\alpha)\mathbb{E}\left[\delta_k^2\right]+2\alpha\left(8p_h\mathbb{E}\left[\delta_k^2\right]+32\frac{\sigma^2p_h}{N}+8\sigma^2m_h^2\alpha^4+\frac{\tilde{\sigma}_{\mathbf{w}}^2d}{N^2}\right)\\ &\leq(1+17\alpha p_h)\mathbb{E}\left[\delta_k^2\right]+\underbrace{64\alpha\frac{\sigma^2p_h}{N}+16\sigma^2m_h^2\alpha^5+2\alpha\frac{\tilde{\sigma}_{\mathbf{w}}^2d}{N^2}}_{B}.\end{aligned} \tag{B.54}$$

Iterating this inequality, we can obtain for any $k-\tau\leq k'\leq k$,

$$\mathbb{E}\left[\delta_{k'}^2\right]\leq(1+17\alpha p_h)^{\tau}\mathbb{E}\left[\delta_{k-\tau}^2\right]+B\sum_{\ell=0}^{\tau-1}(1+17\alpha p_h)^{\ell}. \tag{B.55}$$

Now using the fact that $(1+x)\leq e^x,\forall x\in\mathbb{R}$, observe that $(1+17\alpha p_h)^{\ell}\leq(1+17\alpha p_h)^{\tau}\leq e^{0.25}\leq 2$, for $\alpha\leq 1/(68p_h\tau)$. Using the same argument, we also have

$\sum_{\ell=0}^{\tau-1}(1+17\alpha p_h)^\ell \leq 2\tau$. This yields:

$$\mathbb{E}\left[\delta_{k'}^2\right] \leq 2\mathbb{E}\left[\delta_{k-\tau}^2\right] + 2B\tau. \tag{B.56}$$

Next, note that

$$\delta_{k,\tau}^2 \leq \tau \sum_{\ell=k-\tau}^{k-1} \|\boldsymbol{\theta}_{\ell+1} - \boldsymbol{\theta}_\ell\|^2 = \tau\alpha^2 \sum_{\ell=k-\tau}^{k-1} \|\mathbf{v}_\ell\|^2. \tag{B.57}$$

Taking expectations on both sides of the above equation and applying Lemma B.6 and (B.56), we get

$$\begin{aligned}
\mathbb{E}\left[\delta_{k,\tau}^2\right] &\leq \alpha^2\tau \sum_{\ell=k-\tau}^{k-1} \left(8p_h\mathbb{E}\left[\delta_\ell^2\right] + 32\frac{\sigma^2 p_h}{N} + 8\sigma^2 m_h^2\alpha^4 + \frac{\tilde{\sigma}_{\mathbf{w}}^2 d}{N^2}\right) \\
&= 8p_h\alpha^2\tau \sum_{\ell=k-\tau}^{k-1} \mathbb{E}\left[\delta_\ell^2\right] + \frac{1}{2}\alpha\tau^2 B \\
&\leq 8p_h\alpha^2\tau \sum_{\ell=k-\tau}^{k-1} \left(2\mathbb{E}\left[\delta_{k-\tau}^2\right] + 2B\tau\right) + \frac{1}{2}\alpha\tau^2 B \\
&= 16\alpha^2\tau^2 p_h\mathbb{E}\left[\delta_{k-\tau}^2\right] + 16\alpha^2\tau^3 B p_h + \frac{1}{2}\alpha\tau^2 B.
\end{aligned} \tag{B.58}$$

In the above steps, we used the fact that $\ell \geq \tau$ since $k \geq 2\tau$. We now proceed to simplify the resulting inequality above as follows:

$$\begin{aligned}
\mathbb{E}\left[\delta_{k,\tau}^2\right] &\leq 16\alpha^2\tau^2 p_h\mathbb{E}\left[\delta_{k-\tau}^2\right] \\
&\quad + 16\alpha^2\tau^3 p_h \left(64\alpha\frac{\sigma^2 p_h}{N} + 16\sigma^2 m_h^2\alpha^5 + 2\alpha\frac{\tilde{\sigma}_{\mathbf{w}}^2 d}{N^2}\right) \\
&\quad + \frac{1}{2}\alpha\tau^2 \left(64\alpha\frac{\sigma^2 p_h}{N} + 16\sigma^2 m_h^2\alpha^5 + 2\alpha\frac{\tilde{\sigma}_{\mathbf{w}}^2 d}{N^2}\right) \\
&= 16\alpha^2\tau^2 p_h\mathbb{E}\left[\delta_{k-\tau}^2\right] \\
&\quad + 16\alpha^3\tau^3 p_h \left(64\frac{\sigma^2 p_h}{N} + 16\sigma^2 m_h^2\alpha^4 + 2\frac{\tilde{\sigma}_{\mathbf{w}}^2 d}{N^2}\right) \\
&\quad + \frac{1}{2}\alpha^2\tau^2 \left(64\frac{\sigma^2 p_h}{N} + 16\sigma^2 m_h^2\alpha^4 + 2\frac{\tilde{\sigma}_{\mathbf{w}}^2 d}{N^2}\right) \\
&\overset{(a)}{\leq} 16\alpha^2\tau^2 p_h\mathbb{E}\left[\delta_{k-\tau}^2\right] + 48\alpha^2\tau^2\frac{\sigma^2 p_h}{N} + 2\alpha^4\tau^2\sigma^2 + 2\alpha^2\tau^2\frac{\tilde{\sigma}_{\mathbf{w}}^2 d}{N^2},
\end{aligned} \tag{B.59}$$

where for (a), we used the fact that $\alpha\tau \leq \frac{1}{68p_h}$, and that $\frac{m_h^2}{p_h} \leq 1$, implying $m_h^2\alpha \leq \frac{1}{68\tau}$.

Now noting that $\delta_{k-\tau}^2 \leq 2\delta_k^2 + 2\delta_{k,\tau}^2$, we obtain

$$\mathbb{E}\left[\delta_{k,\tau}^2\right](1 - 32\alpha^2\tau^2 p_h) \leq 32\alpha^2\tau^2 p_h \mathbb{E}\left[\delta_k^2\right] + 48\alpha^2\tau^2\frac{\sigma^2 p_h}{N} + 2\alpha^4\tau^2\sigma^2 + 2\alpha^2\tau^2\frac{\tilde{\sigma}_{\mathbf{w}}^2 d}{N^2}. \tag{B.60}$$

Since $\alpha\tau \leq \frac{1}{68p_h}$, we have that $1 - 32\alpha^2\tau^2 p_h \leq \frac{1}{2}$, and hence

$$\mathbb{E}\left[\delta_{k,\tau}^2\right] \leq 64\alpha^2\tau^2 p_h \mathbb{E}\left[\delta_k^2\right] + 96\alpha^2\tau^2\frac{\sigma^2 p_h}{N} + 4\alpha^4\tau^2\sigma^2 + 4\alpha^2\tau^2\frac{\tilde{\sigma}_{\mathbf{w}}^2 d}{N^2}. \tag{B.61}$$

□

Using the above lemma, we are now able to provide a bound for $\mathbb{E}[\psi_k]$, which is the last ingredient we need to prove Theorem 4.3.

Lemma B.8. Let $k \geq 2\tau$ and $\alpha \leq \frac{1}{68\tau p_h}$. We then have

$$\mathbb{E}[\psi_k] \leq 435\alpha\tau p_h \mathbb{E}\left[\delta_k^2\right] + 657\alpha\tau\frac{p_h\sigma^2}{N} + 30\tau p_h\sigma^2\alpha^3 + 27\alpha\tau\frac{\tilde{\sigma}_{\mathbf{w}}^2 d}{N^2}. \tag{B.62}$$

Proof. Recall the definition of $\bar{\mathbf{g}}_N(\boldsymbol{\theta}_k) \triangleq \frac{1}{N}\sum_{i=1}^N h_{i,k}\bar{\mathbf{g}}(\boldsymbol{\theta}_k)$ and $\mathbf{g}_{h,k}(\boldsymbol{\theta}_k) \triangleq \frac{1}{N}\sum_{i=1}^N h_{i,k}\mathbf{g}_{i,k}(\boldsymbol{\theta}_k)$. We write ψ_k as $\psi_k = T_1 + T_2 + T_3 + T_4$, where

$$\begin{aligned}
T_1 &= \langle \boldsymbol{\theta}_k - \boldsymbol{\theta}_{k-\tau}, \mathbf{g}_{h,k}(\boldsymbol{\theta}_k) - \bar{\mathbf{g}}_N(\boldsymbol{\theta}_k)\rangle, \\
T_2 &= \langle \boldsymbol{\theta}_{k-\tau} - \boldsymbol{\theta}^*, \mathbf{g}_{h,k}(\boldsymbol{\theta}_{k-\tau}) - \bar{\mathbf{g}}_N(\boldsymbol{\theta}_{k-\tau})\rangle, \\
T_3 &= \langle \boldsymbol{\theta}_{k-\tau} - \boldsymbol{\theta}^*, \mathbf{g}_{h,k}(\boldsymbol{\theta}_k) - \mathbf{g}_{h,k}(\boldsymbol{\theta}_{k-\tau})\rangle, \\
T_4 &= \langle \boldsymbol{\theta}_{k-\tau} - \boldsymbol{\theta}^*, \bar{\mathbf{g}}_N(\boldsymbol{\theta}_{k-\tau}) - \bar{\mathbf{g}}_N(\boldsymbol{\theta}_k)\rangle.
\end{aligned} \tag{B.63}$$

We now bound each of the terms $\mathbb{E}[T_1] - \mathbb{E}[T_4]$ individually. We start by observing that

$$\begin{aligned}
\mathbb{E}[T_1] &= \langle \boldsymbol{\theta}_k - \boldsymbol{\theta}_{k-\tau}, \mathbf{g}_{h,k}(\boldsymbol{\theta}_k) - \bar{\mathbf{g}}_N(\boldsymbol{\theta}_k)\rangle \\
&\leq \frac{1}{2\alpha\tau}\mathbb{E}\left[\delta_{k,\tau}^2\right] + \frac{1}{2}\alpha\tau\mathbb{E}\left[\|\mathbf{g}_{h,k}(\boldsymbol{\theta}_k) - \bar{\mathbf{g}}_N(\boldsymbol{\theta}_k)\|^2\right] \\
&\leq \frac{1}{2\alpha\tau}\mathbb{E}\left[\delta_{k,\tau}^2\right] + \alpha\tau\mathbb{E}\left[\|\mathbf{g}_{h,k}(\boldsymbol{\theta}_k)\|^2\right] + \alpha\tau\mathbb{E}\left[\|\bar{\mathbf{g}}_N(\boldsymbol{\theta}_k) - \bar{\mathbf{g}}_N(\boldsymbol{\theta}^*)\|^2\right].
\end{aligned} \tag{B.64}$$

Now note that $\mathbb{E}\left[\|\mathbf{g}_{h,k}(\boldsymbol{\theta}_k)\|^2\right]$ can be bounded using the same procedure we used in

(B.44), while for $\mathbb{E}\left[\delta_{k,\tau}^2\right]$ we can invoke Lemma B.7. We also have

$$\begin{aligned}\mathbb{E}\left[\|\bar{\mathbf{g}}_N(\boldsymbol{\theta}_k) - \bar{\mathbf{g}}_N(\boldsymbol{\theta}^*)\|^2\right] &\leq \frac{N}{N^2}\sum_{i=1}^{N}\mathbb{E}\left[h_{i,k}^2\|\bar{\mathbf{g}}(\boldsymbol{\theta}_k) - \bar{\mathbf{g}}(\boldsymbol{\theta}^*)\|^2\right] \\ &\leq \frac{4}{N}\mathbb{E}\left[\delta_k^2\right]\sum_{i=1}^{N}\mathbb{E}\left[h_{i,k}^2\right] = 4p_h\mathbb{E}\left[\delta_k^2\right].\end{aligned} \tag{B.65}$$

Now, combining the bounds on these three terms and simplifying, we can obtain

$$\mathbb{E}\left[T_1\right] \leq 44\alpha\tau p_h\mathbb{E}\left[\delta_k^2\right] + 80\alpha\tau\frac{\sigma^2 p_h}{N} + 3\tau\sigma^2\alpha^3 + 2\alpha\tau\frac{\tilde{\sigma}_{\mathbf{w}}^2 d}{N^2}. \tag{B.66}$$

We now proceed to bound $\mathbb{E}\left[T_3\right]$. We will again use the fact that $\delta_{k-\tau}^2 \leq 2\delta_k^2 + 2\delta_{k,\tau}^2$.

$$\begin{aligned}\mathbb{E}\left[T_3\right] &= \mathbb{E}\left[\langle\boldsymbol{\theta}_{k-\tau} - \boldsymbol{\theta}^*, \mathbf{g}_{h,k}(\boldsymbol{\theta}_k) - \mathbf{g}_{h,k}(\boldsymbol{\theta}_{k-\tau})\rangle\right] \\ &= \mathbb{E}\left[\langle\boldsymbol{\theta}_{k-\tau} - \boldsymbol{\theta}^*, \frac{1}{N}\sum_{i=1}^{N}h_{i,k}(\mathbf{g}_{i,k}(\boldsymbol{\theta}_k) - \mathbf{g}_{i,k}(\boldsymbol{\theta}_{k-\tau}))\rangle\right] \\ &= \mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N}h_{i,k}\langle\boldsymbol{\theta}_{k-\tau} - \boldsymbol{\theta}^*, \mathbf{g}_{i,k}(\boldsymbol{\theta}_k) - \mathbf{g}_{i,k}(\boldsymbol{\theta}_{k-\tau})\rangle\right] \\ &\leq m_h\mathbb{E}\left[\delta_{k-\tau}\frac{1}{N}\sum_{i=1}^{N}\|\mathbf{g}_{i,k}(\boldsymbol{\theta}_k) - \mathbf{g}_{i,k}(\boldsymbol{\theta}_{k-\tau})\|\right] \\ &\leq \frac{\alpha\tau}{2}m_h^2\mathbb{E}\left[\delta_{k-\tau}^2\right] + \frac{2}{\alpha\tau}\mathbb{E}\left[\delta_{k,\tau}^2\right] \\ &\leq \alpha\tau m_h^2\mathbb{E}\left[\delta_k^2\right] + \alpha\tau m_h^2\mathbb{E}\left[\delta_{k,\tau}^2\right] + \frac{2}{\alpha\tau}\mathbb{E}\left[\delta_{k,\tau}^2\right] \\ &\leq \alpha\tau m_h^2\mathbb{E}\left[\delta_k^2\right] + \frac{3}{\alpha\tau}\mathbb{E}\left[\delta_{k,\tau}^2\right],\end{aligned} \tag{B.67}$$

where we have used that $\alpha\tau \leq \frac{1}{68p_h}$ and $\frac{m_h^2}{p_h} \leq 1$, which imply $m_h^2\alpha\tau \leq 1$. Applying Lemma B.7, we can then get

$$\mathbb{E}\left[T_3\right] \leq \alpha\tau p_h\mathbb{E}\left[\delta_k^2\right] + \frac{3}{\alpha\tau}\left(64\alpha^2\tau^2 p_h\mathbb{E}\left[\delta_k^2\right] + 96\alpha^2\tau^2\frac{p_h\sigma^2}{N} + 4\alpha^2\tau^2\sigma^2\alpha^2 + 4\alpha^2\tau^2\frac{\tilde{\sigma}_{\mathbf{w}}^2 d}{N^2}\right). \tag{B.68}$$

Simplifying the above bound yields:

$$\mathbb{E}\left[T_3\right] \leq 193\alpha\tau p_h\mathbb{E}\left[\delta_k^2\right] + 288\alpha\tau\frac{p_h\sigma^2}{N} + 12\tau\sigma^2\alpha^3 + 12\alpha\tau\frac{\tilde{\sigma}_{\mathbf{w}}^2 d}{N^2}. \tag{B.69}$$

With analogous calculations, we can derive exactly the same bound for $\mathbb{E}\left[T_4\right]$.

We now proceed to bound $\mathbb{E}[T_2]$. For ease of notation, let us define $\mathcal{F}_{k,\tau} = (\{o_{i,k-\tau}\}_{i=1}^N, \boldsymbol{\theta}_{k-\tau})$. Observe:

$$\mathbb{E}[T_2] = \mathbb{E}[\mathbb{E}[T_2|\mathcal{F}_{k,\tau}]] = \mathbb{E}[\langle \boldsymbol{\theta}_{k-\tau} - \boldsymbol{\theta}^*, \frac{m_h}{N}\sum_{i=1}^{N}(\mathbb{E}[\mathbf{g}_{i,k}(\boldsymbol{\theta}_{k-\tau}, o_{i,k})|\mathcal{F}_{k,\tau}] - \bar{\mathbf{g}}(\boldsymbol{\theta}_{k-\tau}))\rangle]$$
$$\leq \mathbb{E}\left[\delta_{k-\tau}\frac{m_h}{N}\sum_{i=1}^{N}\eta_{k,\tau}^{(i)}(\boldsymbol{\theta}_{k-\tau})\right] \leq m_h\alpha^2\mathbb{E}[\delta_{k-\tau}(1+\|\boldsymbol{\theta}_{k-\tau}\|)].$$

Since $\alpha < 1$, we have $\delta_{k-\tau}(\delta_{k-\tau}+2\sigma) \leq \frac{\delta_{k-\tau}^2}{\alpha} + 2\sigma\delta_{k-\tau} + \alpha\sigma^2 = \left(\frac{\delta_{k-\tau}}{\sqrt{\alpha}} + \sqrt{\alpha}\sigma\right)^2 \leq 2\left(\frac{\delta_{k-\tau}^2}{\alpha} + \alpha\sigma^2\right)$. Based on this observation, Lemma B.7, and the fact that $m_h \leq p_h$, we obtain

$$\begin{aligned}\mathbb{E}[T_2] &\leq 2m_h\alpha^2\left(\frac{\delta_{k-\tau}^2}{\alpha} + \alpha\sigma^2\right)\\ &= 2m_h\alpha\delta_{k-\tau}^2 + 2m_h\alpha^3\sigma^2\\ &\leq 4m_h\alpha\delta_k^2 + 4m_h\alpha\delta_{k,\tau}^2 + 2m_h\alpha^3\sigma^2\\ &\leq 5p_h\alpha\tau\mathbb{E}\left[\delta_k^2\right] + \alpha\tau\frac{\sigma^2 p_h}{N} + 3\tau\sigma^2 p_h\alpha^3 + \alpha\tau\frac{\tilde{\sigma}_{\mathbf{w}}^2 d}{N^2}.\end{aligned} \tag{B.70}$$

Combining all the terms, we can conclude the proof. □

We are now in position to prove Theorem 4.3.

Proof of Theorem 4.3

Consider the inequality that we derived in (B.34). For $k \geq 2\tau$, plugging in the inequality the bounds derived in Lemma B.6 and in Lemma B.8, we get

$$\begin{aligned}
\mathbb{E}\left[\delta_{k+1}^2\right] &\leq \mathbb{E}\left[\delta_k^2\right] - 2\alpha m_h(1-\gamma)\omega\mathbb{E}\left[\delta_k^2\right] + 2\alpha\mathbb{E}\left[\psi_k\right] + \alpha^2\mathbb{E}\left[\|\mathbf{v}_k\|^2\right] \\
&\leq \mathbb{E}\left[\delta_k^2\right] - 2\alpha m_h(1-\gamma)\omega\mathbb{E}\left[\delta_k^2\right] \\
&\quad + 2\alpha\left(435\alpha\tau p_h\mathbb{E}\left[\delta_k^2\right] + 657\alpha\tau\frac{p_h\sigma^2}{N} + 30\tau p_h\sigma^2\alpha^3 + 27\alpha\tau\frac{\tilde{\sigma}_{\mathbf{w}}^2 d}{N^2}\right) \\
&\quad + \alpha^2\left(8p_h\mathbb{E}\left[\delta_k^2\right] + 32\frac{\sigma^2 p_h}{N} + 8\sigma^2 m_h^2\alpha^4 + \frac{\tilde{\sigma}_{\mathbf{w}}^2 d}{N^2}\right) \\
&= \mathbb{E}\left[\delta_k^2\right] - 2\alpha m_h(1-\gamma)\omega\mathbb{E}\left[\delta_k^2\right] \\
&\quad + 878\alpha^2\tau p_h\mathbb{E}\left[\delta_k^2\right] + 1346\alpha^2\tau\frac{p_h\sigma^2}{N} + 61\tau p_h\sigma^2\alpha^4 + 55\alpha^2\tau\frac{\tilde{\sigma}_{\mathbf{w}}^2 d}{N^2} \\
&= \mathbb{E}\left[\delta_k^2\right] - \alpha\left(2m_h(1-\gamma)\omega - 878\alpha\tau p_h\right)\mathbb{E}\left[\delta_k^2\right] \\
&\quad + 1346\alpha^2\tau\frac{p_h\sigma^2}{N} + 61\tau p_h\sigma^2\alpha^4 + 55\alpha^2\tau\frac{\tilde{\sigma}_{\mathbf{w}}^2 d}{N^2}.
\end{aligned} \tag{B.71}$$

Hence, for $\alpha \leq \frac{m_h(1-\gamma)\omega}{C_0\tau p_h}$, with $C_0 = 878$, we get

$$\mathbb{E}\left[\delta_{k+1}^2\right] \leq (1 - \alpha m_h(1-\gamma)\omega)\mathbb{E}\left[\delta_k^2\right] + 1346\alpha^2\tau\frac{p_h\sigma^2}{N} + 61\tau p_h\sigma^2\alpha^4 + 55\alpha^2\tau\frac{\tilde{\sigma}_{\mathbf{w}}^2 d}{N^2}. \tag{B.72}$$

Unrolling this inequality, we obtain

$$\begin{aligned}
\mathbb{E}\left[\delta_T^2\right] \leq (1 - \alpha m_h(1-\gamma)\omega)^{T-2\tau}\mathbb{E}\left[\delta_{2\tau}^2\right] + C_2\frac{\alpha\tau p_h\sigma^2}{m_h(1-\gamma)\omega N} \\
+ \frac{C_3\tau p_h\sigma^2\alpha^3}{m_h(1-\gamma)\omega} + \frac{C_4\alpha\tau\tilde{\sigma}_{\mathbf{w}}^2 d}{m_h(1-\gamma)\omega N^2},
\end{aligned} \tag{B.73}$$

with $C_2 = 1346$, $C_3 = 61$ and $C_4 = 55$. To conclude, we proceed to bound $\mathbb{E}\left[\delta_{2\tau}^2\right]$. Note that, for any $k \geq 0$,

$$\mathbb{E}\left[\delta_{k+1}^2\right] \leq (1+\alpha)\mathbb{E}\left[\delta_k^2\right] + 2\alpha\mathbb{E}\left[\|\mathbf{v}_k\|^2\right]. \tag{B.74}$$

Observe as before (see (B.43)):

$$\mathbb{E}\left[\|\mathbf{v}_k\|^2\right] = \mathbb{E}\left[\|\mathbf{g}_{h,k}(\boldsymbol{\theta}_k)\|^2\right] + \mathbb{E}\left[\|\mathbf{w}_k\|^2\right]. \tag{B.75}$$

Note that $\mathbb{E}\left[\|\mathbf{w}_k\|^2\right] = \frac{\tilde{\sigma}_{\mathbf{w}}^2 d}{N^2}$ and that we can bound $\mathbb{E}\left[\|\mathbf{g}_{h,k}(\boldsymbol{\theta}_k)\|^2\right]$ as follows:

$$\begin{aligned}\mathbb{E}\left[\|\mathbf{g}_{h,k}(\boldsymbol{\theta}_k)\|^2\right] &= \mathbb{E}\left[\|\frac{1}{N}\sum_{i=1}^{N} h_{i,k}\mathbf{g}_{i,k}(\boldsymbol{\theta}_k)\|^2\right] \leq \frac{N}{N^2}\sum_{i=1}^{N} h_{i,k}^2\|\mathbf{g}_{i,k}(\boldsymbol{\theta}_k)\|^2 \\ &\leq \frac{1}{N}\left(8(\|\boldsymbol{\theta}_k\|^2+\sigma^2)\sum_{i=1}^{N}\mathbb{E}\left[h_{i,k}^2\right]\right) \leq \left(8(2\delta_k^2+3\sigma^2)\right)p_h \\ &= 16p_h\mathbb{E}\left[\delta_k^2\right] + 24p_h\sigma^2.\end{aligned} \tag{B.76}$$

Hence,

$$\mathbb{E}\left[\|\mathbf{v}_k\|^2\right] \leq 16p_h\mathbb{E}\left[\delta_k^2\right] + 24p_h\sigma^2 + \frac{\tilde{\sigma}_{\mathbf{w}}^2 d}{N^2}. \tag{B.77}$$

We thus have

$$\begin{aligned}\mathbb{E}\left[\delta_{k+1}^2\right] &\leq (1+\alpha)\mathbb{E}\left[\delta_k^2\right] + 2\alpha\left(16p_h\mathbb{E}\left[\delta_k^2\right] + 24p_h\sigma^2 + \frac{\tilde{\sigma}_{\mathbf{w}}^2 d}{N^2}\right) \\ &\leq (1+33p_h\alpha)\mathbb{E}\left[\delta_k^2\right] + 48\alpha p_h\sigma^2 + 2\alpha\frac{\tilde{\sigma}_{\mathbf{w}}^2 d}{N^2}.\end{aligned} \tag{B.78}$$

Iterating this inequality, we obtain

$$\mathbb{E}\left[\delta_{2\tau}^2\right] \leq (1+33\alpha p_h)^{2\tau}\delta_0^2 + \left(48\alpha p_h\sigma^2 + 2\alpha\frac{\tilde{\sigma}_{\mathbf{w}}^2 d}{N^2}\right)\sum_{j=0}^{2\tau-1}(1+33\alpha p_h)^j. \tag{B.79}$$

Now with the same procedure used to obtain (B.56), we see that if $66\alpha p_h\tau \leq \frac{1}{4}$, then

$$\begin{aligned}\mathbb{E}\left[\delta_{2\tau}^2\right] &\leq 2\mathbb{E}\left[\delta_0^2\right] + 4\tau\left(48\alpha p_h\sigma^2 + 2\alpha\frac{\tilde{\sigma}_{\mathbf{w}}^2 d}{N^2}\right) \\ &\leq 2\mathbb{E}\left[\delta_0^2\right] + 192\alpha\tau\sigma^2 p_h + \frac{8\alpha\tilde{\sigma}_{\mathbf{w}}^2 d\tau}{N^2} \\ &\leq 2\mathbb{E}\left[\delta_0^2\right] + \sigma^2 + \frac{\tilde{\sigma}_{\mathbf{w}}^2 d}{N^2},\end{aligned} \tag{B.80}$$

where we have used that $\alpha\tau \leq \frac{1}{878p_h}$. With the choice of step size in the statement of the theorem, we have $\alpha\omega(1-\gamma)m_h \leq 1$. This then yields $(1-\alpha\omega(1-\gamma)m_h)^{-2\tau} \leq (1-\alpha m_h)^{-2\tau}$. Finally note that since $\alpha\tau \leq \frac{1}{C_0 p_h} \leq \frac{1}{4p_h}$, we have $\alpha\tau m_h \leq \frac{m_h}{4p_h} \leq \frac{1}{4}$, where we used the fact that $\frac{m_h}{p_h} \leq 1$. Based on this discussion and using Bernoulli's inequality, we obtain $(1-\alpha m_h)^{2\tau} \geq 1 - 2\alpha\tau m_h \geq \frac{1}{2}$; hence, $(1-\alpha m_h)^{-2\tau} \leq 2$. We thus have $(1-\alpha\omega(1-\gamma)m_h)^{-2\tau} \leq 2$. Plugging this bound back in (B.73) completes the proof.

B.3 Proof of Theorem 4.4

In this section, we prove Theorem 4.4. We start by introducing the following definitions to lighten the notation:

$$\begin{aligned} \eta_{k,\tau}^{(i)}(\boldsymbol{\theta}) &\triangleq \|\mathbb{E}\left[\mathbf{g}_{i,k}(\boldsymbol{\theta}, o_{i,k})|o_{i,k-\tau}\right] - \bar{\mathbf{g}}(\boldsymbol{\theta})\|, \ k \geq \tau, \\ \delta_{k,h} &\triangleq \|\boldsymbol{\theta}_k - \boldsymbol{\theta}_{k-h}\|, \ k \geq h \geq 0, \\ d_k &\triangleq \max_{k-2\tau_{max}-\tau \leq j \leq k} \mathbb{E}\left[\delta_j^2\right], \ k \geq \tau + 2\tau_{max}. \end{aligned} \tag{B.81}$$

For our analysis, we will need the following result from [19].

Lemma B.9. The following holds $\forall \boldsymbol{\theta} \in \mathbb{R}^m$:

$$\langle \boldsymbol{\theta}^* - \boldsymbol{\theta}, \bar{\mathbf{g}}(\boldsymbol{\theta}) \rangle \geq \omega(1-\gamma)\|\boldsymbol{\theta}^* - \boldsymbol{\theta}\|^2.$$

We will also use the fact that the random TD update directions and their steady-state versions are 2-Lipschitz [19], i.e., $\forall i \in [N], \forall k \in \mathbb{N}$, and $\forall \boldsymbol{\theta}, \boldsymbol{\theta}' \in \mathbb{R}^m$, we have:

$$\max\{\|\mathbf{g}(\boldsymbol{\theta}) - \mathbf{g}(\boldsymbol{\theta}')\|, \|\bar{\mathbf{g}}(\boldsymbol{\theta}) - \bar{\mathbf{g}}(\boldsymbol{\theta}')\|\} \leq 2\|\boldsymbol{\theta} - \boldsymbol{\theta}'\|. \tag{B.82}$$

From [135], we also have that $\forall i \in [N], \forall k \in \mathbb{N}, \forall \boldsymbol{\theta} \in \mathbb{R}^m$:

$$\|\mathbf{g}(\boldsymbol{\theta}, o_{i,k})\| \leq 2\|\boldsymbol{\theta}\| + 2\bar{r}, \tag{B.83}$$

which, squared, using $\bar{r} \leq \sigma$, yields

$$\|\mathbf{g}(\boldsymbol{\theta}, o_{i,k})\|^2 \leq 8(\|\boldsymbol{\theta}\|^2 + \sigma^2). \tag{B.84}$$

We will often use the fact that, from the definition of the mixing time in Definition 5.2.1, we have, for a given iteration $k \geq \tau$, defining $\Theta_{i,k} \triangleq \{\boldsymbol{\theta}_{k-\tau}, o_{i,k-\tau}\}$,

$$\|\mathbb{E}\left[\mathbf{g}(\boldsymbol{\theta}_{k-\tau}, o_{i,k})|\Theta_{i,k}\right] - \bar{\mathbf{g}}(\boldsymbol{\theta}_{k-\tau})\| \leq \alpha^q \left(\|\boldsymbol{\theta}_{k-\tau}\| + 1\right) \tag{B.85}$$

The proof also relies on the following result from [56]:

Lemma B.10. Let V_k be non-negative real numbers that satisfy

$$V_{k+1} \leq pV_k + q \max_{(k-d(k))_+ \leq \ell \leq k} V_\ell + \beta,$$

for $\beta, p, q > 0$. Here, $k \geq 0$ and $0 \leq d(k) \leq d_{\max}$ for some $d_{\max} \geq 0$. If $p + q < 1$, then

we have

$$V_k \leq \rho^k V_0 + \epsilon,$$

where $\rho = (p+q)^{1/(1+d_{\max})}$ and $\epsilon = \frac{\beta}{1-p-q}$.

We will also use the fact that, for any $a, b \in \mathbb{R}, c \geq 0$,

$$ab = a\sqrt{c}\frac{b}{\sqrt{c}} \leq \frac{1}{2}\left(ca^2 + \frac{b^2}{c}\right), \tag{B.86}$$

and also the fact that, for $a_i \in \mathbb{R}, i = 1, ..., N$,

$$\left(\sum_{i=1}^{N} a_i\right)^2 \leq N \sum_{i=1}^{N} a_i^2. \tag{B.87}$$

Equipped with the above basic results, we now provide an outline of our proof before illustrating the technical details.

Outline of the proof. Recall $t_{i,k} \triangleq (k - \tau_{i,k})_+$. We write the update rule (**??**) as

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k + \alpha \mathbf{v}_k = \boldsymbol{\theta}_k + \alpha \bar{\mathbf{g}}(\boldsymbol{\theta}_k) - \alpha \mathbf{e}_k, \tag{B.88}$$

with $\mathbf{e}_k \triangleq \bar{\mathbf{g}}(\boldsymbol{\theta}_k) - \mathbf{v}_k$. Thus,

$$\mathbf{e}_k = \frac{1}{N}\sum_{i=1}^{N}\left(\bar{\mathbf{g}}(\boldsymbol{\theta}_k) - \mathbf{g}(\boldsymbol{\theta}_{t_{i,k}}, o_{i,t_{i,k}})\right). \tag{B.89}$$

We analyze the following recursion:

$$\begin{aligned} \delta_{k+1}^2 &= T_1 + \alpha^2 T_2 - 2\alpha T_3, \text{ with} \\ T_1 &= \|\boldsymbol{\theta}_k - \boldsymbol{\theta}^* + \alpha \bar{\mathbf{g}}(\boldsymbol{\theta}_k)\|^2 \\ T_2 &= \|\mathbf{e}_k\|^2 \\ T_3 &= \langle \boldsymbol{\theta}_k - \boldsymbol{\theta}^* + \alpha \bar{\mathbf{g}}(\boldsymbol{\theta}_k), \mathbf{e}_k \rangle. \end{aligned} \tag{B.90}$$

The most important part of the proof consists of obtaining a bound of the following form:

$$\mathbb{E}\left[\delta_{k+1}^2\right] \leq p\mathbb{E}\left[\delta_k^2\right] + O(\alpha^2(\tau + \tau_{max}))d_k + B_{\alpha,N}, \tag{B.91}$$

where d_k was defined in (B.81), $p < 1$ is a contraction factor and

$$B_{\alpha,N} = O(\alpha^2(\tau + \tau_{max}))\frac{\sigma^2}{N} + O(\alpha^4)\sigma^2, \tag{B.92}$$

which guarantees the linear speedup effect with N. The bound in (B.91) allows us to obtain the desired result, by picking a step size small enough and applying Lemma B.10. Given this outline, in the following we provide bounds for $\mathbb{E}[T_1]$, $\mathbb{E}[T_2]$ and $\mathbb{E}[T_3]$.

Bounding $\mathbb{E}[T_1]$. Note that

$$\begin{aligned} T_1 &= \|\boldsymbol{\theta}_k - \boldsymbol{\theta}^* + \alpha\bar{\mathbf{g}}(\boldsymbol{\theta}_k)\|^2 \\ &= \delta_k^2 + 2\alpha\langle\boldsymbol{\theta}_k - \boldsymbol{\theta}^*, \bar{\mathbf{g}}(\boldsymbol{\theta}_k)\rangle + \alpha^2\|\bar{\mathbf{g}}(\boldsymbol{\theta}_k)\|^2. \end{aligned} \tag{B.93}$$

Note that, using Lemma C.1, we get

$$\langle\boldsymbol{\theta}_k - \boldsymbol{\theta}^*, \bar{\mathbf{g}}(\boldsymbol{\theta}_k)\rangle \leq -(1-\gamma)\omega\delta_k^2, \tag{B.94}$$

and using (C.3) we get

$$\|\bar{\mathbf{g}}(\boldsymbol{\theta}_k)\|^2 = \|\bar{\mathbf{g}}(\boldsymbol{\theta}_k) - \bar{\mathbf{g}}(\boldsymbol{\theta}^*)\|^2 \leq 4\delta_k^2. \tag{B.95}$$

Combining the two bounds above and taking the expectation,

$$\mathbb{E}[T_1] \leq (1 - 2\alpha(1-\gamma)\omega)\mathbb{E}\left[\delta_k^2\right] + 4\alpha^2\mathbb{E}\left[\delta_k^2\right]. \tag{B.96}$$

Bounding $\mathbb{E}[T_2]$. We need the following result.

Lemma B.11. For $k \geq \tau + \tau_{max}$, we have

$$\mathbb{E}\left[\|\mathbf{v}_k\|^2\right] \leq 8 \max_{k-\tau_{max}\leq j\leq k} \mathbb{E}\left[\delta_j^2\right] + 32\frac{\sigma^2}{N} + 8\sigma^2\alpha^{2q} \tag{B.97}$$

Proof. We write

$$\begin{aligned} \|\mathbf{v}_k\|^2 &\leq \frac{2}{N^2}(V_1 + V_2), \quad \text{with} \\ V_1 &= \|\sum_{i=1}^{N} \mathbf{g}(\boldsymbol{\theta}_{t_{i,k}}, o_{i,t_{i,k}}) - \mathbf{g}(\boldsymbol{\theta}^*, o_{i,t_{i,k}})\|^2, \\ V_2 &= \|\sum_{i=1}^{N} \mathbf{g}(\boldsymbol{\theta}^*, o_{i,t_{i,k}})\|^2. \end{aligned} \tag{B.98}$$

We now bound V_1.

$$\begin{aligned} V_1 &\leq N \sum_{i=1}^{N} \|\mathbf{g}(\boldsymbol{\theta}_{t_{i,k}}, o_{i,t_{i,k}}) - \mathbf{g}(\boldsymbol{\theta}^*, o_{i,t_{i,k}})\|^2 \\ &\overset{(C.3)}{\leq} 4N \sum_{i=1}^{N} \delta^2_{t_{i,k}}. \quad \text{Thus,} \\ \mathbb{E}[V_1] &\leq 4N \sum_{i=1}^{N} \mathbb{E}\left[\delta^2_{t_{i,k}}\right] \leq 4N^2 \max_{k-\tau_{max} \leq j \leq k} \mathbb{E}\left[\delta^2_j\right] \end{aligned} \tag{B.99}$$

We now proceed to bound V_2.

$$\begin{aligned} V_2 &= V_{21} + V_{22}, \quad \text{with} \\ V_{21} &= \sum_{i=1}^{N} \|\mathbf{g}(\boldsymbol{\theta}^*, o_{i,t_{i,k}})\|^2 \\ V_{22} &= \sum_{\substack{i,j=1 \\ i \neq j}}^{N} \langle \mathbf{g}(\boldsymbol{\theta}^*, o_{i,t_{i,k}}), \mathbf{g}(\boldsymbol{\theta}^*, o_{j,t_{j,k}}) \rangle. \end{aligned} \tag{B.100}$$

We see that, using (B.84), we get

$$V_{21} \leq 8 \sum_{i=1}^{N} (\|\boldsymbol{\theta}^*\|^2 + \sigma^2) \quad \leq 16 N \sigma^2. \tag{B.101}$$

Now, using the fact that the observations $o_{i,k}$ and $o_{j,k'}$ are independent for $i \neq j$ and for any $k, k' \geq 0$,

$$\begin{aligned} \mathbb{E}[V_{22}] = \sum_{\substack{i,j=1 \\ i \neq j}}^{N} \langle &\mathbb{E}\left[\mathbb{E}\left[\mathbf{g}(\boldsymbol{\theta}^*, o_{i,t_{i,k}}) | o_{i,t_{i,k}-\tau}\right]\right], \\ &\mathbb{E}\left[\mathbb{E}\left[\mathbf{g}(\boldsymbol{\theta}^*, o_{j,t_{j,k}}) | o_{j,t_{j,k}-\tau}\right]\right] \rangle. \end{aligned} \tag{B.102}$$

Using the fact that $\bar{\mathbf{g}}(\boldsymbol{\theta}^*) = 0$, and Cauchy-Schwarz inequality followed by Jensen's inequality, we can write

$$\begin{aligned} \mathbb{E}[V_{22}] &\leq \sum_{\substack{i,j=1 \\ i \neq j}}^{N} \mathbb{E}\left[\eta^{(i)}_{t_{i,k},\tau}(\boldsymbol{\theta}^*)\right] \times \mathbb{E}\left[\eta^{(j)}_{t_{j,k},\tau}(\boldsymbol{\theta}^*)\right] \\ &\leq N^2 \alpha^{2q} (\|\boldsymbol{\theta}^*\| + \sigma)^2 \leq 4N^2 \alpha^{2q} \sigma^2. \end{aligned} \tag{B.103}$$

Plugging the above bounds on $\mathbb{E}[V_1]$ and $\mathbb{E}[V_2]$ in (B.98), we can conclude the proof of the lemma. □

We are now in the position to proceed bounding $\mathbb{E}[T_2]$.

$$\begin{aligned}\mathbb{E}[T_2] = \mathbb{E}\left[\|\mathbf{e}_k\|^2\right] &= \mathbb{E}\left[\|\bar{\mathbf{g}}(\boldsymbol{\theta}_k) - \mathbf{v}_k\|^2\right] \\ &\leq 2\mathbb{E}\left[\|\bar{\mathbf{g}}(\boldsymbol{\theta}_k)\|^2 + \|\mathbf{v}_k\|^2\right].\end{aligned} \tag{B.104}$$

Note that $\|\bar{\mathbf{g}}(\boldsymbol{\theta}_k)\|^2 = \|\bar{\mathbf{g}}(\boldsymbol{\theta}_k) - \bar{\mathbf{g}}(\boldsymbol{\theta}^*)\|^2 \leq 4\delta_k^2$, and so using Lemma B.11 we get

$$\mathbb{E}[T_2] \leq 24 \max_{k-\tau_{max}\leq j\leq k} \mathbb{E}\left[\delta_j^2\right] + 64\frac{\sigma^2}{N} + 16\sigma^2\alpha^{2q}. \tag{B.105}$$

Bounding $\mathbb{E}[T_3]$. We now bound $\mathbb{E}[T_3]$, which represents the major technical burden of the proof. We need the following result.

Lemma B.12. Let $k \geq \tau_{max} + h$. Then,

$$\mathbb{E}\left[\delta_{k,h}^2\right] \leq 8\alpha^2 h^2 \left(d_k + 4\frac{\sigma^2}{N} + \sigma^2\alpha^{2q}\right) \tag{B.106}$$

Proof. Note that, using Lemma B.11,

$$\begin{aligned}\mathbb{E}\left[\delta_{k,h}^2\right] &= \mathbb{E}\left[\|\boldsymbol{\theta}_k - \boldsymbol{\theta}_{k-h}\|^2\right] \\ &\leq h\sum_{l=k-h}^{k-1} \mathbb{E}\left[\|\boldsymbol{\theta}_{l+1} - \boldsymbol{\theta}_l\|^2\right] \\ &\leq \alpha^2 h \sum_{l=k-h}^{k-1} \mathbb{E}\left[\|\mathbf{v}_l\|^2\right] \\ &\leq \alpha^2 h \sum_{l=k-h}^{k-1} (8 \max_{l-\tau_{max}\leq j\leq l} \mathbb{E}\left[\delta_j^2\right] \\ &+ 32\frac{\sigma^2}{N} + 8\sigma^2\alpha^{2q}) \\ &\leq 8\alpha^2 h^2 \left(d_k + 4\frac{\sigma^2}{N} + \sigma^2\alpha^{2q}\right).\end{aligned} \tag{B.107}$$

□

Now, we can write

$$\begin{aligned}T_3 &= K + T_{32}, \text{ with} \\ K &= \langle \boldsymbol{\theta}_k - \boldsymbol{\theta}^*, \mathbf{e}_k\rangle, \\ T_{32} &= \alpha\langle \bar{\mathbf{g}}(\boldsymbol{\theta}_k), \mathbf{e}_k\rangle.\end{aligned} \tag{B.108}$$

Note that, using Cauchy-Schwarz and (C.7),

$$\begin{aligned} T_{32} &\leq \frac{\alpha}{2}\left(\|\bar{\mathbf{g}}(\boldsymbol{\theta}_k)\|^2 + \|\mathbf{e}_k\|^2\right) \\ &\leq 2\alpha\delta_k^2 + \frac{\alpha}{2}\|\mathbf{e}_k\|^2. \end{aligned} \tag{B.109}$$

Taking the expectation and using the bound on $\mathbb{E}[T_2]$,

$$\mathbb{E}[T_{32}] \leq \alpha(14d_k + 32\frac{\sigma^2}{N} + 8\sigma^2\alpha^{2q}). \tag{B.110}$$

Now we bound K. Define $\bar{\mathbf{g}}_{N,k} \triangleq \frac{1}{N}\sum_{i=1}^{N}\bar{\mathbf{g}}(\boldsymbol{\theta}_{k-\tau_{i,k}})$. Adding and subtracting $\bar{\mathbf{g}}_{N,k}$, we write

$$\begin{aligned} K &= K_1 + K_2, \quad \text{with} \\ K_1 &= \langle \boldsymbol{\theta}_k - \boldsymbol{\theta}^*, \bar{\mathbf{g}}(\boldsymbol{\theta}_k) - \bar{\mathbf{g}}_{N,k}\rangle, \\ K_2 &= \langle \boldsymbol{\theta}_k - \boldsymbol{\theta}^*, \bar{\mathbf{g}}_{N,k} - \mathbf{v}_k\rangle. \end{aligned} \tag{B.111}$$

Now note that, taking the sum of the second term outside of the inner product in K_1,

$$\begin{aligned} K_1 &= \frac{1}{N}\sum_{i=1}^{N} K_{1,i}, \quad \text{with} \\ K_{1,i} &= \langle \boldsymbol{\theta}_k - \boldsymbol{\theta}^*, \bar{\mathbf{g}}(\boldsymbol{\theta}_k) - \bar{\mathbf{g}}(\boldsymbol{\theta}_{t_{i,k}})\rangle. \end{aligned} \tag{B.112}$$

We now bound $\mathbb{E}[K_{1,i}]$. Using (C.7) and (C.3),

$$\begin{aligned} K_{1,i} &\leq \alpha\tau_{max}\delta_k^2 + \frac{1}{4\alpha\tau_{max}}\|\bar{\mathbf{g}}(\boldsymbol{\theta}_k) - \bar{\mathbf{g}}(\boldsymbol{\theta}_{t_{i,k}})\|^2 \\ &\leq \alpha\tau_{max}\delta_k^2 + \frac{1}{\alpha\tau_{max}}\delta_{k,\tau_{i,k}}^2. \end{aligned} \tag{B.113}$$

Note that, using Lemma B.12, which requires $t_{i,k} = k - \tau_{i,k} \geq \tau$, which holds for $k \geq \tau_{max} + \tau$,

$$\mathbb{E}\left[\delta_{t_{i,k},\tau}^2\right] \leq 8\alpha^2\tau_{max}^2\left(d_k + 4\frac{\sigma^2}{N} + \sigma^2\alpha^{2q}\right). \tag{B.114}$$

Taking the expectation and applying (B.114), we get

$$\mathbb{E}[K_{1,i}] \leq \alpha\tau_{max}\left(9d_k + 32\frac{\sigma^2}{N} + 8\sigma^2\alpha^{2q}\right), \tag{B.115}$$

and note that $\mathbb{E}[K_1]$ is bounded by the same quantity. We now proceed to bound $\mathbb{E}[K_2]$.

$$\begin{aligned} K_2 &= \frac{1}{N}\sum_{i=1}^{N} K_{2,i}, \quad \text{with} \\ K_{2,i} &= \langle \boldsymbol{\theta}_k - \boldsymbol{\theta}^*, \bar{\mathbf{g}}(\boldsymbol{\theta}_{t_{i,k}}) - \mathbf{g}(\boldsymbol{\theta}_{t_{i,k}}, o_{i,t_{i,k}})\rangle \\ &= \Delta_{1,i} + \Delta_{2,i} + \Delta_{3,i}, \quad \text{where} \\ \Delta_{1,i} &= \langle \boldsymbol{\theta}_k - \boldsymbol{\theta}^*, \bar{\mathbf{g}}(\boldsymbol{\theta}_{t_{i,k}}) - \bar{\mathbf{g}}(\boldsymbol{\theta}_{t_{i,k}-\tau})\rangle, \\ \Delta_{2,i} &= \langle \boldsymbol{\theta}_k - \boldsymbol{\theta}^*, \bar{\mathbf{g}}(\boldsymbol{\theta}_{t_{i,k}-\tau}) - \mathbf{g}(\boldsymbol{\theta}_{t_{i,k}-\tau}, o_{i,t_{i,k}})\rangle \\ \Delta_{3,i} &= \langle \boldsymbol{\theta}_k - \boldsymbol{\theta}^*, \mathbf{g}(\boldsymbol{\theta}_{t_{i,k}-\tau}, o_{i,t_{i,k}}) - \mathbf{g}(\boldsymbol{\theta}_{t_{i,k}}, o_{i,t_{i,k}})\rangle. \end{aligned} \tag{B.116}$$

Note that, using Cauchy-Schwarz inequality and (C.3),

$$\begin{aligned} \Delta_{1,i} &\leq \delta_k \|\bar{\mathbf{g}}(\boldsymbol{\theta}_{t_{i,k}}) - \bar{\mathbf{g}}(\boldsymbol{\theta}_{t_{i,k}-\tau})\| \\ &\leq 2\delta_k \delta_{t_{i,k},\tau} \\ &\overset{(C.7)}{\leq} \left(\alpha\tau\delta_k^2 + \frac{\delta_{t_{i,k},\tau}^2}{\alpha\tau}\right). \end{aligned} \tag{B.117}$$

For $k \geq \tau + \tau_{max}$, we can apply Lemma B.12 and get

$$\mathbb{E}\left[\delta_{t_{i,k},\tau}^2\right] \leq \alpha^2\tau^2\left(8d_k + 32\frac{\sigma^2}{N} + 8\sigma^2\alpha^{2q}\right). \tag{B.118}$$

Thus, taking the expectation, we can get

$$\mathbb{E}[\Delta_{1,i}] \leq 9\alpha\tau d_k + 32\alpha\tau\frac{\sigma^2}{N} + 8\alpha\tau\sigma^2\alpha^{2q}. \tag{B.119}$$

Note that, using the Lipschitz property (see (C.3)), we get the exact same bound for $\Delta_{3,i}$. Now note that

$$\begin{aligned} K_2 &= \frac{1}{N}\sum_{i=1}^{N}(\Delta_{1,i} + \Delta_{3,i}) + \bar{\Delta}, \quad \text{with} \\ \bar{\Delta} &= \frac{1}{N}\sum_{i=1}^{N}\Delta_{2,i}. \end{aligned} \tag{B.120}$$

To bound $\bar{\Delta}$, we want to use the geometric mixing property of the Markov chain, that follows from Assumption 6. However, due to the correlations existing between the iterates $\boldsymbol{\theta}_{t_{i,k}}$, this needs to be done with special care. Defining $k' \triangleq k - \tau_{max} - \tau$, we start by

adding and subtracting $\boldsymbol{\theta}_{k'}$ from the first term in the inner product of $\Delta_{2,i}$, getting

$$\begin{aligned}
\bar{\Delta} &= \bar{\Delta}_1 + \bar{\Delta}_2, \quad \text{with} \\
\bar{\Delta}_1 &= \langle \boldsymbol{\theta}_k - \boldsymbol{\theta}_{k'}, \frac{1}{N}\sum_{i=1}^{N} \bar{\mathbf{g}}(\boldsymbol{\theta}_{t_{i,k}-\tau}) - \mathbf{g}(\boldsymbol{\theta}_{t_{i,k}-\tau}, o_{i,t_{i,k}}) \rangle, \\
\bar{\Delta}_2 &= \langle \boldsymbol{\theta}_{k'} - \boldsymbol{\theta}^*, \frac{1}{N}\sum_{i=1}^{N} \bar{\mathbf{g}}(\boldsymbol{\theta}_{t_{i,k}-\tau}) - \mathbf{g}(\boldsymbol{\theta}_{t_{i,k}-\tau}, o_{i,t_{i,k}}) \rangle.
\end{aligned} \tag{B.121}$$

Note that we can write, using (C.7) and (B.87),

$$\begin{aligned}
\bar{\Delta}_1 \leq & \frac{1}{2\alpha\left(\tau_{max}+\tau\right)} \delta^2_{k,\tau+\tau_{max}} \\
& + \frac{\alpha\left(\tau_{max}+\tau\right)}{2N^2} \underbrace{\| \sum_{i=1}^{N} \bar{\mathbf{g}}(\boldsymbol{\theta}_{t_{i,k}-\tau}) - \mathbf{g}(\boldsymbol{\theta}_{t_{i,k}-\tau}, o_{i,t_{i,k}}) \|^2}_{G}
\end{aligned} \tag{B.122}$$

For $k \geq \tau + 2\tau_{max}$, we can apply Lemma B.12 and get

$$\mathbb{E}\left[\delta^2_{k,\tau+\tau_{max}}\right] \leq 8\alpha^2\left(\tau+\tau_{max}\right)^2\left(d_k + 4\frac{\sigma^2}{N} + \sigma^2\alpha^{2q}\right). \tag{B.123}$$

Now note that

$$G \leq 2 \underbrace{\| \sum_{i=1}^{N} \bar{\mathbf{g}}(\boldsymbol{\theta}_{t_{i,k}-\tau}) \|^2}_{G_1} + 2 \underbrace{\| \sum_{i=1}^{N} \mathbf{g}(\boldsymbol{\theta}_{t_{i,k}-\tau}, o_{i,t_{i,k}}) \|^2}_{G_2}, \tag{B.124}$$

with $\mathbb{E}\left[G_1\right] \leq 4N^2\mathbb{E}\left[\delta^2_{t_{i,k}-\tau}\right] \leq 4N^2 d_k$. Also note that

$$\begin{aligned}
G_2 &\leq 2G_{21} + 2G_{22}, \quad \text{with} \\
G_{21} &= \| \sum_{i=1}^{N} \mathbf{g}(\boldsymbol{\theta}_{t_{i,k}-\tau}, o_{i,t_{i,k}}) - \mathbf{g}(\boldsymbol{\theta}^*, o_{i,k-\tau_{i,k}}) \|^2, \\
G_{22} &= \| \sum_{i=1}^{N} \mathbf{g}(\boldsymbol{\theta}^*, o_{i,t_{i,k}}) \|^2.
\end{aligned} \tag{B.125}$$

Note that $\mathbb{E}\left[G_{21}\right]$ and $\mathbb{E}\left[G_{22}\right]$ can be bounded in the same way as $\mathbb{E}\left[V_1\right]$ and $\mathbb{E}\left[V_2\right]$ in the proof of Lemma B.11. We get

$$\mathbb{E}\left[G_2\right] \leq 8\left(N^2 d_k + 4N\sigma^2 + N^2\sigma^2\alpha^{2q}\right). \tag{B.126}$$

Hence, we get

$$\mathbb{E}\left[\bar{\Delta}_1\right] \leq 16\alpha\left(\tau_{max}+\tau\right)\left(d_k + 3\frac{\sigma^2}{N} + \sigma^2\alpha^{2q}\right). \tag{B.127}$$

Defining $\bar{\Delta}_{2,i} \triangleq \langle \boldsymbol{\theta}_{k'} - \boldsymbol{\theta}^*, \bar{\mathbf{g}}(\boldsymbol{\theta}_{t_{i,k}-\tau}) - \mathbf{g}(\boldsymbol{\theta}_{t_{i,k}-\tau}, o_{i,t_{i,k}})\rangle$,

$$\mathbb{E}\left[\bar{\Delta}_2\right] = \frac{1}{N}\sum_{i=1}^{N}\mathbb{E}\left[\bar{\Delta}_{2,i}\right]. \tag{B.128}$$

Now, defining $\bar{\Theta}_{i,k} \triangleq \{\boldsymbol{\theta}_{k'}, \boldsymbol{\theta}_{t_{i,k}-\tau}, o_{t_{i,k}-\tau}\}$,

$$\begin{aligned}
\mathbb{E}\left[\bar{\Delta}_{2,i}\right] =& \mathbb{E}[\langle \boldsymbol{\theta}_{k'} - \boldsymbol{\theta}^*, \bar{\mathbf{g}}(\boldsymbol{\theta}_{t_{i,k}-\tau}) - \mathbf{g}(\boldsymbol{\theta}_{t_{i,k}-\tau}, o_{i,t_{i,k}})\rangle] \\
=& \mathbb{E}[\langle \boldsymbol{\theta}_{k'} - \boldsymbol{\theta}^*, \\
& \bar{\mathbf{g}}(\boldsymbol{\theta}_{t_{i,k}-\tau}) - \mathbb{E}\left[\mathbf{g}(\boldsymbol{\theta}_{t_{i,k}-\tau}, o_{i,t_{i,k}})|\bar{\Theta}_{i,k}\right]\rangle] \\
\leq& \mathbb{E}[\delta_{k'} \\
& \cdot \underbrace{\|\bar{\mathbf{g}}(\boldsymbol{\theta}_{t_{i,k}-\tau}) - \mathbb{E}\left[\mathbf{g}(\boldsymbol{\theta}_{t_{i,k}-\tau}, o_{i,t_{i,k}})|\bar{\Theta}_{i,k}\right]\|}_{\bar{\eta}_{k,i}}].
\end{aligned} \tag{B.129}$$

Now, recall $\Theta_{i,k} = \{\boldsymbol{\theta}_{t_{i,k}-\tau}, o_{i,t_{i,k}-\tau}\}$. Note that, for the memoryless property of the Markov chain $\{o_{i,k}\}$, we have

$$\mathbb{E}\left[\mathbf{g}(\boldsymbol{\theta}_{t_{i,k}-\tau}, o_{i,t_{i,k}})|\bar{\Theta}_{i,k}\right] = \mathbb{E}\left[\mathbf{g}(\boldsymbol{\theta}_{t_{i,k}-\tau}, o_{i,t_{i,k}})|\Theta_{i,k}\right]. \tag{B.130}$$

Indeed, by inspecting the update rule (4.6) we see that the parameter $\boldsymbol{\theta}_{k'}$ is a function of agent i observations only up to time step $k'-1$, so up to observation $o_{i,k'-1}$. Hence, all the statistical information contained in $\boldsymbol{\theta}_{k'}$ has no influence on the random variable $o_{i,t_{i,k}}$ once we condition on $o_{i,t_{i,k}-\tau}$, because $t_{i,k} - \tau = k - \tau_{i,k} - \tau > k' - 1 = k - \tau_{max} - \tau - 1$. Therefore,

$$\begin{aligned}
\bar{\eta}_{k,i} &= \|\bar{\mathbf{g}}(\boldsymbol{\theta}_{t_{i,k}-\tau}) - \mathbb{E}\left[\mathbf{g}(\boldsymbol{\theta}_{t_{i,k}-\tau}, o_{i,t_{i,k}})|\Theta_{i,k}\right]\| \\
&\overset{(B.85)}{\leq} \alpha^q\left(\|\boldsymbol{\theta}_{t_{i,k}-\tau}\| + \sigma\right).
\end{aligned} \tag{B.131}$$

Hence, we get

$$\begin{aligned}\mathbb{E}\left[\bar{\Delta}_{2,i}\right] &\leq \mathbb{E}\left[\delta_{k'}\alpha^q\left(\|\boldsymbol{\theta}_{t_{i,k}-\tau}\|+\sigma\right)\right]\\ &\leq \mathbb{E}\left[\delta_{k'}\alpha^q\left(\delta_{t_{i,k}-\tau}+2\sigma\right)\right]\\ &\leq \mathbb{E}\left[\frac{\alpha}{2}\delta_{k'}^2+\alpha^{2q-1}\left(\delta_{t_{i,k}-\tau}^2+4\sigma^2\right)\right]\\ &\leq \alpha d_k + 4\alpha\sigma^2\alpha^{2(q-1)}\\ &\leq \alpha d_k + 4\alpha\sigma^2\alpha^q\end{aligned} \tag{B.132}$$

where we used the fact that $\alpha \leq \frac{1}{2}$ and the fact that $2(q-1) \geq q$ for $q \geq 2$. Putting all the above bounds together, and also the fact that $\alpha \leq \frac{1}{16(\tau_{max}+\tau)}$ we get

$$\mathbb{E}\left[T_3\right] \leq \alpha\left(\tau_{max}+\tau\right)\left(44d_k + 144\frac{\sigma^2}{N}\right) + 7\alpha\sigma^2\alpha^q. \tag{B.133}$$

Concluding the proof. Using the bounds on $\mathbb{E}\left[T_1\right], \mathbb{E}\left[T_2\right]$ and $\mathbb{E}\left[T_3\right]$, we obtain, for $k \geq \tau' \triangleq 2\tau_{max}+\tau$,

$$\begin{aligned}\mathbb{E}\left[\delta_{k+1}^2\right] \leq (1-2\alpha(1-\gamma)\omega)\mathbb{E}\left[\delta_k^2\right] + 15\alpha^2\sigma^2\alpha^q\\ + \alpha^2\left(\tau+\tau_{max}\right)\left(112d_k + 352\frac{\sigma^2}{N}\right).\end{aligned} \tag{B.134}$$

We can now write the above bound in the following way:

$$\begin{aligned}V_{k+1} &= pV_k + q\max_{k-\tau'\leq j\leq k} V_j + \beta, \quad \text{with}\\ V_k &= \mathbb{E}\left[\delta_k^2\right],\\ p &= (1-2\alpha(1-\gamma)\omega),\\ q &= 112\alpha^2\left(\tau_{max}+\tau\right),\\ \beta &= 352\alpha^2\left(\tau_{max}+\tau\right)\left(\frac{\sigma^2}{N}\right) + 15\alpha^2\sigma^2\alpha^q.\end{aligned} \tag{B.135}$$

Imposing $\alpha \leq \frac{(1-\gamma)\omega}{112(\tau+\tau_{max})}$, we get $p+q \leq 1-\alpha(1-\gamma)\omega$, and we can apply Lemma B.10 getting

$$\mathbb{E}\left[\delta_T^2\right] \leq \rho^{T-\tau'}\mathbb{E}\left[\delta_{\tau'}^2\right] + \epsilon, \tag{B.136}$$

with $\rho = (1-\alpha(1-\gamma)\omega)^{\frac{1}{1+\tau'}}$ and $\epsilon = \frac{\beta}{1-p-q}$. Note that we can easily show that $\rho^{-\tau'} \leq 2$ for $\alpha \leq \frac{1}{16(\tau_{max}+\tau)}$. Furthermore, we can show that $\mathbb{E}\left[\delta_{\tau'}^2\right] \leq 3\sigma^2$, as we do next. Note

that, using (B.84),

$$\begin{aligned}\|\mathbf{v}_k\|^2 &\leq \frac{1}{N}\sum_{i=1}^{N} 8(2\delta_{t_{i,k}}^2 + 3\sigma^2) \\ &\leq 16 \max_{i=1,\dots,N} \delta_{t_{i,k}}^2 + 24\sigma^2.\end{aligned} \tag{B.137}$$

Using this bound, note that, for any $k \geq 0$

$$\begin{aligned}\delta_{k+1}^2 &= \delta_k^2 + 2\alpha\langle \boldsymbol{\theta}_t - \boldsymbol{\theta}^*, \mathbf{v}_k\rangle + \alpha^2\|\mathbf{v}_k\|^2 \\ &\overset{(C.7)}{\leq} \delta_k^2 + \alpha\delta_k^2 + \alpha\|\mathbf{v}_k\|^2 + \alpha^2\|\mathbf{v}_k\|^2 \\ &\leq (1+\alpha)\delta_k^2 + 2\alpha\|\mathbf{v}_k\|^2 \\ &\leq (1+\alpha)\delta_k^2 + 32\alpha \max_{i=1,\dots,N} \delta_{t_{i,k}}^2 + 48\alpha\sigma^2,\end{aligned} \tag{B.138}$$

where recall that $t_{i,k} \leq k$. Now define $\bar{p} \triangleq 1+\alpha$, $\bar{q} \triangleq 32\alpha$ and $\nu \triangleq \bar{p} + \bar{q}$ and $\bar{\beta} \triangleq 48\alpha\sigma^2$. We now prove by induction that, for all $k \geq 0$,

$$\delta_k^2 \leq \nu^k \delta_0^2 + \epsilon_k, \tag{B.139}$$

where $\epsilon_k = \nu\epsilon_{k-1} + \bar{\beta}$ for $k \geq 1$ and $\epsilon_0 = 0$. The base case is trivially satisfied, because $\delta_0^2 \leq \delta_0^2$. As an inductive step, suppose that (C.137) is true for $0 \leq s \leq k$, for some $k \geq 0$, so

$$\delta_s^2 \leq \nu^s \delta_0^2 + \epsilon_s, \quad 0 \leq s \leq k. \tag{B.140}$$

Now, we check the property for $k+1$, using (C.136), and noting that ϵ_k is an increasing sequence:

$$\begin{aligned}\delta_{k+1}^2 &\leq \bar{p}\delta_k^2 + \bar{q} \max_{i=1,\dots,N} \delta_{t_{i,k}}^2 + \bar{\beta}, \\ &\leq \bar{p}(\nu^k\delta_0^2 + \epsilon_k) + \bar{q}\left(\max_{i=1,\dots,N} \nu^{t_{i,k}}\delta_0^2 + \epsilon_{t_{i,k}}\right) + \bar{\beta} \\ &\leq \bar{p}(\nu^k\delta_0^2 + \epsilon_k) + \bar{q}(\nu^k\delta_0^2 + \epsilon_k) + \bar{\beta} \\ &\leq (\bar{p}+\bar{q})\nu^k\delta_0^2 + (\bar{p}+\bar{q})\epsilon_k + \bar{\beta} \\ &= \nu^{k+1}\delta_0^2 + \epsilon_{k+1}.\end{aligned} \tag{B.141}$$

From which we can conclude the proof of (C.137). Now, note that $\epsilon_k = \bar{\beta}\sum_{j=0}^{k-1}\nu^j$, and that for $0 \leq k \leq \tau'$,

$$\nu^k \leq \nu^{\tau'} \leq (1+33\alpha)^{\tau'} \leq e^{33\alpha\tau'} \leq e^{0.25} \leq 2, \tag{B.142}$$

imposing $\alpha \leq \frac{1}{132\tau'}$. Hence, for $0 \leq k \leq \tau'$,

$$\begin{aligned} \delta_k^2 &\leq \nu^k \delta_0^2 + \epsilon_k \leq 2\delta_0^2 + \bar{\beta} \sum_{j=0}^{\tau'-1} \nu^j \leq 2\delta_0^2 + 2\bar{\beta}\tau' \\ &= 2\delta_0^2 + 2(48\alpha\sigma^2)\tau' \leq 2\delta_0^2 + \sigma^2 \leq 3\sigma^2, \end{aligned} \tag{B.143}$$

where we used the fact that $\alpha \leq \frac{1}{100\tau'}$ and that $\delta_0^2 \leq \sigma^2$. We can therefore conclude, writing the bound in (B.136) as

$$\begin{aligned} \mathbb{E}\left[\delta_T^2\right] \leq \exp\left(-\frac{\alpha(1-\gamma)\omega T}{2\left(\tau+\tau_{max}\right)}\right) 6\sigma^2 \\ + 352\frac{\alpha\left(\tau+\tau_{max}\right)\sigma^2}{(1-\gamma)\,\omega N} + 15\frac{\alpha^3\sigma^2}{(1-\gamma)\omega}. \end{aligned} \tag{B.144}$$

C

Appendix: Proofs of Chapter 5

In this Appendix, we provide the proofs for the theoretical results stated in Chapter 5. In particular, we provide the proofs for all the Theorems and Lemmas. We start by recalling some implications of the Assumptions of Section 5.2 in the following.

Preliminaries

First, recall that from Assumption 7 we have, $\forall \boldsymbol{\theta} \in \mathbb{R}^d$:

$$\langle \boldsymbol{\theta}^* - \boldsymbol{\theta}, \bar{\mathbf{g}}(\boldsymbol{\theta}) \rangle \geq \mu \|\boldsymbol{\theta}^* - \boldsymbol{\theta}\|^2. \tag{C.1}$$

Throughout the proof, we will often invoke the mixing property (see Definition 5.2.1), which implies that, for a fixed $\boldsymbol{\theta}$,

$$\|\mathbb{E}\left[\mathbf{g}(\boldsymbol{\theta}, o_t) | o_{t-\tau_{mix}}\right] - \bar{\mathbf{g}}(\boldsymbol{\theta})\| \leq \alpha \left(\|\boldsymbol{\theta}\| + \sigma\right) \tag{C.2}$$

We will also use the fact that the SA update directions and their steady-state versions are L-Lipschitz (Assumption 8), i.e., $\forall o \in \{o_t\}_{t\in\mathbb{N}}$, and $\forall \boldsymbol{\theta}, \boldsymbol{\theta}' \in \mathbb{R}^d$, we have:

$$\begin{aligned} &\|\bar{\mathbf{g}}(\boldsymbol{\theta}) - \bar{\mathbf{g}}(\boldsymbol{\theta}')\| \leq L\|\boldsymbol{\theta} - \boldsymbol{\theta}'\|, \text{ and} \\ &\|\mathbf{g}(\boldsymbol{\theta}, o_t) - \mathbf{g}(\boldsymbol{\theta}', o_t)\| \leq L\|\boldsymbol{\theta} - \boldsymbol{\theta}'\|. \end{aligned} \tag{C.3}$$

We further have

$$\|\mathbf{g}(\boldsymbol{\theta}, o)\| \leq L(\|\boldsymbol{\theta}\| + \sigma), \forall o \in \{o_t\}_{t\in\mathbb{N}}, \forall \boldsymbol{\theta} \in \mathbb{R}^d. \tag{C.4}$$

Given that $(x+y)^2 \leq 2(x^2+y^2), \forall x, y \in \mathbb{R}$, we will often use the following inequality:

$$\|\mathbf{g}(\boldsymbol{\theta}, o_t)\|^2 \leq L^2(\|\boldsymbol{\theta}\| + \sigma)^2 \leq 2L^2(\|\boldsymbol{\theta}\|^2 + \sigma^2). \tag{C.5}$$

Without loss of generality, we assume that

$$L \geq 1, \quad \sigma \geq \max\{\|\boldsymbol{\theta}_0\|, \|\boldsymbol{\theta}^*\|\}, \quad \mu < 1. \tag{C.6}$$

We will often use the fact that, for any $x, y \in \mathbb{R}$, we have

$$xy \leq \frac{1}{2}(x^2 + y^2). \tag{C.7}$$

In addition, we will often use the fact that, for $t \geq 2$, $a_i \in \mathbb{R}, i = 0, ..., t-1$, it holds

$$\left(\sum_{i=0}^{t-1} a_i\right)^2 \leq t \sum_{i=0}^{t-1} a_i^2 \tag{C.8}$$

C.1 Proof of Theorem 5.3

First, we recall the definition of the SA recursion with constant delay:

$$\boldsymbol{\theta}_{t+1} = \begin{cases} \boldsymbol{\theta}_0 & \text{if } 0 \leq t < \tau \\ \boldsymbol{\theta}_t + \alpha \mathbf{g}(\boldsymbol{\theta}_{t-\tau}, o_{t-\tau}) & \text{if } t \geq \tau \end{cases} \tag{C.9}$$

For analysis purposes, we define a virtual iterate, $\tilde{\boldsymbol{\theta}}_t$. This virtual iterate is updated with the SA update direction without delays, and it is defined as follows:

$$\tilde{\boldsymbol{\theta}}_{t+1} = \tilde{\boldsymbol{\theta}}_t + \alpha \mathbf{g}(\boldsymbol{\theta}_t, o_t), \ \tilde{\boldsymbol{\theta}}_0 = \boldsymbol{\theta}_0. \tag{C.10}$$

We also introduce the related error term $\mathbf{d}_t$, which is the gap between the virtual iterate and the actual iterate.

$$\tilde{\boldsymbol{\theta}}_t = \boldsymbol{\theta}_t + \mathbf{d}_t, \quad \text{with } \mathbf{d}_0 = \mathbf{0}. \tag{C.11}$$

From the definition of $\tilde{\boldsymbol{\theta}}_t$, we can write the following recursions for $\mathbf{d}_t$, for $t \geq 0$:

$$\mathbf{d}_{t+1} = \mathbf{d}_t + \alpha(\mathbf{g}(\boldsymbol{\theta}_t, o_t) - \mathbf{g}(\boldsymbol{\theta}_{t-\tau}, o_{t-\tau})). \tag{C.12}$$

We define $\mathbf{g}(\boldsymbol{\theta}_l, o_l) = \boldsymbol{\theta}_l = \mathbf{d}_l = \mathbf{0}$ for $l < 0$. We also define $\tilde{r}_t = \|\tilde{\boldsymbol{\theta}}_t - \boldsymbol{\theta}^*\|$. For convenience, we define $\tilde{r}_t = 0$ for $t < 0$, which is equivalent to setting $\tilde{\boldsymbol{\theta}}_t = \boldsymbol{\theta}^*$ for $t < 0$.

C.1.1 Proofs of Auxiliary Lemmas

We first state and prove the following Lemma, which we will use in the proof of Theorem 5.3.

Lemma C.1. For $w_t := (1 - 0.5\mu\alpha)^{-(t+1)}$ with $\alpha \leq \frac{\mu}{C\bar{\tau}}$, $C \geq 2$, the following inequality holds for $0 \leq i \leq 2\bar{\tau}$, and for any t,

$$w_t \leq 2w_{t-i}. \tag{C.13}$$

Proof.

$$\begin{aligned}
w_t &= w_{t-i}\left(1 - \frac{\mu\alpha}{2}\right)^{-i} \\
&\overset{(a)}{\leq} w_{t-i}\left(1 - \frac{\mu^2}{2C\bar{\tau}}\right)^{-i} \\
&\overset{(b)}{\leq} w_{t-i}\left(1 - \frac{\mu^2}{2C\bar{\tau}}\right)^{-\bar{\tau}} \\
&\overset{(c)}{\leq} w_{t-i}\left(1 - \frac{1}{4\bar{\tau}}\right)^{-\bar{\tau}} \\
&\overset{(d)}{\leq} w_{t-i}\left(1 + \frac{1}{2\bar{\tau}}\right)^{\bar{\tau}} \\
&\overset{(e)}{\leq} w_{t-i}\exp\left(\frac{1}{2}\right) \\
&\leq 2w_{t-i},
\end{aligned} \tag{C.14}$$

in (a), we used the bound on α, in (b), we used the bound on i, in (c), we used $\mu < 1$ and $C \geq 2$, in (d), we used

$$(1 - x)^{-1} \leq (1 + 2x) \text{ for } \quad 0 \leq x \leq \frac{1}{2}, \tag{C.15}$$

and for (e) we used $(1 + x)^k \leq \exp(xk)$ for $k \geq 0$. □

We defined $\mathbf{g}(\boldsymbol{\theta}_i, o_i) = \mathbf{0}$ for $i < 0$, $\boldsymbol{\theta}_t = \mathbf{0}$ for $t < 0$, and $\mathbf{d}_t = \mathbf{0}$ for $t < 0$. First, note

that, starting from the definition of $\mathbf{d}_t$ in (C.12),

$$\begin{aligned}
\mathbf{d}_{t+1} &= \mathbf{d}_t + \alpha\left(\mathbf{g}(\boldsymbol{\theta}_k, o_t) - \mathbf{g}(\boldsymbol{\theta}_{t-\tau}, o_{t-\tau})\right. \\
&= \mathbf{d}_{t-1} + \alpha\left(\mathbf{g}(\boldsymbol{\theta}_{t-1}, o_{t-1}) - \mathbf{g}(\boldsymbol{\theta}_{t-1-\tau}, o_{t-1-\tau})\right) \\
&\qquad + \alpha\left(\mathbf{g}(\boldsymbol{\theta}_k, o_t) - \mathbf{g}(\boldsymbol{\theta}_{t-\tau}, o_{t-\tau})\right) \\
&= \mathbf{d}_0 \quad + \alpha \sum_{l=0}^{t} \left(\mathbf{g}(\boldsymbol{\theta}_l, o_l) - \mathbf{g}(\boldsymbol{\theta}_{l-\tau}, o_{l-\tau})\right) \\
&\overset{(*)}{=} \mathbf{0} \quad + \alpha \sum_{l=t-\tau+1}^{t} \mathbf{g}(\boldsymbol{\theta}_l, o_l),
\end{aligned} \tag{C.16}$$

where $(*)$ follows because the overlapping terms in the sum cancel out. So, we obtain, for all $t \geq 0$,

$$\mathbf{d}_t = \alpha \sum_{l=t-\tau}^{t-1} \mathbf{g}(\boldsymbol{\theta}_l, o_l). \tag{C.17}$$

We can now prove Lemma 5.1, which is key to prove Theorem 5.3.

Proof of Lemma 5.1 - (i), (ii). From (C.17), using the triangle inequality and the bound on the update direction (C.4), we get, recalling that $\sigma \geq \|\boldsymbol{\theta}_0\|$,

$$\begin{aligned}
\|\mathbf{d}_t\| &= \|\alpha \sum_{l=t-\tau}^{t-1} \mathbf{g}(\boldsymbol{\theta}_l, o_l)\| \\
&\overset{(C.4)}{\leq} \alpha L \sum_{l=t-\tau}^{t-1} (\|\boldsymbol{\theta}_l\| + \sigma) \\
&\leq \alpha\tau L\sigma + \alpha L \sum_{l=t-\tau}^{t-1} \|\boldsymbol{\theta}_l\|,
\end{aligned} \tag{C.18}$$

which proves (i). We now prove (ii). Using the triangle inequality and (C.8),

$$\begin{aligned}
\|\mathbf{d}_t\|^2 &= \|\alpha \sum_{l=t-\tau}^{t-1} \mathbf{g}(\boldsymbol{\theta}_l, o_l)\|^2 \\
&\overset{(C.8)}{\leq} \alpha^2\tau \sum_{l=t-\tau}^{t-1} \|\mathbf{g}(\boldsymbol{\theta}_l, o_l)\|^2.
\end{aligned} \tag{C.19}$$

Now, using the upper bound on the squared gradient norm (C.5),

$$\begin{aligned}\|\mathbf{d}_t\|^2 &\leq \alpha^2\tau \sum_{l=t-\tau}^{t-1} \|\mathbf{g}(\boldsymbol{\theta}_l, o_l)\|^2 \\ &\leq 2\alpha^2\tau L^2 \sum_{l=t-\tau}^{t-1} (\|\boldsymbol{\theta}_l\|^2 + \sigma^2) \\ &\leq 2\alpha^2\tau^2 L^2\sigma^2 + 2\alpha^2\tau L^2 \sum_{l=t-\tau}^{t-1} \|\boldsymbol{\theta}_l\|^2.\end{aligned} \tag{C.20}$$

which concludes the proof. □

Using the above inequalities, we can now prove part (*iii*) of Lemma 5.1.

Proof of Lemma 5.1 - (iii). First, recall that, from Lemma 5.1, we have

$$\|\mathbf{d}_t\|^2 \leq 2\alpha^2\tau^2 L^2\sigma^2 + 2\alpha^2\tau L^2 \sum_{l=t-\tau}^{t-1} \|\boldsymbol{\theta}_l\|^2. \tag{C.21}$$

Based on Lemma C.1, for $0 \leq i \leq 2\bar{\tau}$, we have $w_t \leq 2w_{t-i}$ (see (C.14)). Using (C.21),

$$\begin{aligned}\sum_{t=0}^{T} w_t\|\mathbf{d}_t\|^2 &\leq \sum_{t=0}^{T} w_t \left(2\alpha^2\tau^2 L^2\sigma^2 + 2\alpha^2\tau L^2 \sum_{l=t-\tau}^{t-1} \|\boldsymbol{\theta}_l\|^2\right) \\ &\leq 2W_T\alpha^2\tau^2 L^2\sigma^2 + 2\alpha^2\tau L^2 \sum_{t=0}^{T} w_t \sum_{l=t-\tau}^{t-1} \|\boldsymbol{\theta}_l\|^2 \\ &\overset{(*)}{\leq} 2W_T\alpha^2\tau^2 L^2\sigma^2 + 4\alpha^2\tau L^2 \sum_{t=0}^{T} \sum_{l=t-\tau}^{t-1} w_l\|\boldsymbol{\theta}_l\|^2 \\ &\overset{(**)}{\leq} 2W_T\alpha^2\tau^2 L^2\sigma^2 + 4\alpha^2\tau^2 L^2 \sum_{t=0}^{T} w_t\|\boldsymbol{\theta}_t\|^2 \\ &\leq 2W_T\alpha^2\tau^2 L^2\sigma^2 + 8\alpha^2\tau^2 L^2 \sum_{t=0}^{T} w_t \left(\|\tilde{\boldsymbol{\theta}}_t\|^2 + \|\mathbf{d}_t\|^2\right) \\ &\leq 2W_T\alpha^2\tau^2 L^2\sigma^2 + 8\alpha^2\tau^2 L^2 \sum_{t=0}^{T} w_t\|\tilde{\boldsymbol{\theta}}_t\|^2 + \frac{1}{2}\sum_{t=0}^{T} w_t\|\mathbf{d}_t\|^2,\end{aligned} \tag{C.22}$$

where for $(*)$ we used the fact that $w_t \leq 2w_l$ for $t - 2\bar{\tau} \leq l \leq t-1$, and for $(**)$ we used the fact that each element $w_l\|\boldsymbol{\theta}_l\|^2$ appears at most τ times in the sum, for $l = 0, ..., T-1$ (note that, by definition, $\boldsymbol{\theta}_l = 0$ for $l < 0$). In the last inequality, we used $\alpha \leq \frac{1}{4\tau L}$. We

can conclude getting

$$\sum_{t=0}^{T} w_t \|\mathbf{d}_t\|^2 \leq 4W_T \alpha^2 \tau^2 L^2 \sigma^2 + 16\alpha^2 \tau^2 L^2 \sum_{t=0}^{T} w_t \|\tilde{\boldsymbol{\theta}}_t\|^2. \tag{C.23}$$

□

We now prove Lemma 5.2, that provides a bound on the norm of the gap $\|\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}} - \tilde{\boldsymbol{\theta}}_k\|$ and its squared version $\|\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}} - \tilde{\boldsymbol{\theta}}_k\|^2$.

Proof of Lemma 5.2. Inequality (i) of the Lemma can be easily proved by applying the definition of the recursion (C.10),

$$\begin{aligned}
\|\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}} - \tilde{\boldsymbol{\theta}}_k\| &\leq \sum_{l=t-\tau_{mix}}^{t-1} \|\tilde{\boldsymbol{\theta}}_{l+1} - \tilde{\boldsymbol{\theta}}_l\| \\
&\leq \alpha \sum_{l=t-\tau_{mix}}^{t-1} \|\mathbf{g}(\boldsymbol{\theta}_l, o_l)\| \\
&\leq L\alpha \sum_{l=t-\tau_{mix}}^{t-1} (\|\boldsymbol{\theta}_l\| + \sigma) \\
&= L\alpha\sigma\tau_{mix} + L\alpha \sum_{l=t-\tau_{mix}}^{t-1} \|\boldsymbol{\theta}_l\|.
\end{aligned} \tag{C.24}$$

Similarly, for inequality (ii), note that, squaring equation (C.24),

$$\|\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}} - \tilde{\boldsymbol{\theta}}_k\|^2 \leq 2L^2\alpha^2\tau_{mix}^2\sigma^2 + 2L^2\alpha^2\tau_{mix} \sum_{l=t-\tau_{mix}}^{t-1} \|\boldsymbol{\theta}_l\|^2. \tag{C.25}$$

□

We now prove Lemma 5.3, which provide bounds for $\|\mathbf{g}(\boldsymbol{\theta}_k, o_t)\|^2$, m_t and $\mathbb{E}[h_t]$, respectively.

Proof of Lemma 5.3 - (i). From (C.5), we have $\|\mathbf{g}(\boldsymbol{\theta}_t, o_t)\|^2 \leq 2L^2(\|\boldsymbol{\theta}_t\|^2 + \sigma^2)$,

and so

$$\begin{aligned} n_t = \|\mathbf{g}(\boldsymbol{\theta}_t, o_t)\|^2 &\leq 2L^2(\|\boldsymbol{\theta}_t\|^2 + \sigma^2) \\ &\leq 2L^2\|\boldsymbol{\theta}_t - \tilde{\boldsymbol{\theta}}_t + \tilde{\boldsymbol{\theta}}_t\|^2 + 2L^2\sigma^2 \\ &\leq 4L^2\|\mathbf{d}_t\|^2 + 4L^2\|\tilde{\boldsymbol{\theta}}_t\|^2 + 2L^2\sigma^2 \\ &\leq 4L^2\|\mathbf{d}_t\|^2 + 4L^2\|\tilde{\boldsymbol{\theta}}_t - \boldsymbol{\theta}^* + \boldsymbol{\theta}^*\|^2 + 2L^2\sigma^2 \\ &\leq 4L^2\|\mathbf{d}_t\|^2 + 8L^2\tilde{r}_t^2 + 8L^2\|\boldsymbol{\theta}^*\|^2 + 2L^2\sigma^2 \\ &\leq 4L^2\|\mathbf{d}_t\|^2 + 8L^2\tilde{r}_t^2 + 10L^2\sigma^2 \end{aligned} \tag{C.26}$$

where we used $\|\boldsymbol{\theta}^*\| \leq \sigma$ and from which we can conclude. □

Proof of Lemma 5.3 - (ii). By Cauchy-Schwarz inequality, Lipschitz continuity of $\mathbf{g}(\boldsymbol{\theta}, o_t)$ in $\boldsymbol{\theta}$ (see (C.3)), and from the definition of $\mathbf{d}_t$, we get

$$\begin{aligned} m_t &= \langle \mathbf{g}(\boldsymbol{\theta}_t, o_t) - \mathbf{g}(\tilde{\boldsymbol{\theta}}_t, o_t), \tilde{\boldsymbol{\theta}}_t - \boldsymbol{\theta}^\star \rangle \\ &\leq \|\mathbf{g}(\boldsymbol{\theta}_t, o_t) - \mathbf{g}(\tilde{\boldsymbol{\theta}}_t, o_t)\| \|\tilde{\boldsymbol{\theta}}_t - \boldsymbol{\theta}^*\| \\ &\leq L\|\tilde{\boldsymbol{\theta}}_t - \boldsymbol{\theta}_k\| \|\tilde{\boldsymbol{\theta}}_t - \boldsymbol{\theta}^*\| \\ &= L\|\mathbf{d}_t\|\tilde{r}_t. \end{aligned} \tag{C.27}$$

Applying Lemma 5.1 to bound $\|\mathbf{d}_t\|$, we get

$$\begin{aligned} m_t &\leq L\left(\alpha\tau L\sigma + \alpha L \sum_{l=t-\tau}^{t-1} \|\boldsymbol{\theta}_l\|\right)\tilde{r}_t \\ &= \alpha\tau L^2\sigma\tilde{r}_t + \alpha L^2 \sum_{l=t-\tau}^{t-1} \|\boldsymbol{\theta}_l\|\tilde{r}_t \\ &\overset{(C.7)}{\leq} 2\alpha\tau L^2\sigma^2 + 2\alpha\tau L^2\tilde{r}_t^2 + \alpha L^2 \sum_{l=t-\tau}^{t-1} \left(\|\boldsymbol{\theta}_l\|^2 + \tilde{r}_t^2\right) \\ &= 2\alpha\tau L^2\sigma^2 + 3\alpha\tau L^2\tilde{r}_t^2 + \alpha L^2 \sum_{l=t-\tau}^{t-1} \|\boldsymbol{\theta}_l\|^2 \\ &\overset{(C.8)}{\leq} 2\alpha\tau L^2\sigma^2 + 3\alpha\tau L^2\tilde{r}_t^2 + 2\alpha L^2 \sum_{l=t-\tau}^{t-1} \left(\|\mathbf{d}_l\|^2 + \|\tilde{\boldsymbol{\theta}}_l\|^2\right) \\ &\leq 6\alpha\tau L^2\sigma^2 + 3\alpha\tau L^2\tilde{r}_t^2 + 2\alpha L^2 \sum_{l=t-\tau}^{t-1} \|\mathbf{d}_l\|^2 + 4\alpha L^2 \sum_{l=t-\tau}^{t-1} \tilde{r}_l^2. \end{aligned} \tag{C.28}$$

□

Next, we provide the proof of Lemma 5.3, which provides a bound for $\mathbb{E}[h_t]$, which is

the term related to the Markovian sampling and whose analysis requires special care and mixing time arguments.

Proof of Lemma 5.3 - (iii). We start with the case $0 \leq t \leq \tau_{mix}$ Note that, using (C.5),

$$\begin{aligned} h_t &= \langle \tilde{\boldsymbol{\theta}}_t - \boldsymbol{\theta}^*, \mathbf{g}(\tilde{\boldsymbol{\theta}}_t, o_t) - \bar{\mathbf{g}}(\tilde{\boldsymbol{\theta}}_t) \rangle \\ &\leq \|\tilde{\boldsymbol{\theta}}_t - \boldsymbol{\theta}^*\| \|\mathbf{g}(\tilde{\boldsymbol{\theta}}_t, o_t) - \bar{\mathbf{g}}(\tilde{\boldsymbol{\theta}}_t)\| \\ &\overset{(C.7)}{\leq} \frac{1}{2}\tilde{r}_t^2 + \frac{1}{2}\|\mathbf{g}(\tilde{\boldsymbol{\theta}}_t, o_t) - \bar{\mathbf{g}}(\tilde{\boldsymbol{\theta}}_t)\|^2 \\ &\overset{(C.8)}{\leq} \frac{1}{2}\tilde{r}_t^2 + \|\mathbf{g}(\tilde{\boldsymbol{\theta}}_t, o_t)\|^2 + \|\bar{\mathbf{g}}(\tilde{\boldsymbol{\theta}}_t)\|^2 \\ &\overset{(C.5)}{\leq} \frac{\tilde{r}_t^2}{2} + 2L^2\|\tilde{\boldsymbol{\theta}}_t\|^2 + 2L^2\sigma^2 + 2L^2\|\tilde{\boldsymbol{\theta}}_t\|^2 + 2L^2\sigma^2 \\ &\leq \frac{\tilde{r}_t^2}{2} + 8L^2\tilde{r}_t^2 + 12L^2\sigma^2 \\ &\leq 9L^2\tilde{r}_t^2 + 12L^2\sigma^2. \end{aligned} \tag{C.29}$$

Recall that

$$\boldsymbol{\theta}_{t+1} = \begin{cases} \boldsymbol{\theta}_0 & \text{if } \ 0 \leq t < \tau \\ \boldsymbol{\theta}_t + \alpha \mathbf{g}(\boldsymbol{\theta}_{t-\tau}, o_{t-\tau}) & \text{if } \ t \geq \tau \end{cases}, \tag{C.30}$$

from which we can write, for $t \geq \tau$,

$$\begin{aligned} r_{t+1}^2 &= r_t^2 + 2\alpha\langle \boldsymbol{\theta}_t - \boldsymbol{\theta}^*, \mathbf{g}(\boldsymbol{\theta}_{t-\tau}, o_{t-\tau})\rangle + \alpha^2\|\mathbf{g}(\boldsymbol{\theta}_{t-\tau}, o_{t-\tau})\|^2 \\ &\overset{(C.7)}{\leq} r_t^2 + \alpha r_t^2 + \alpha\|\mathbf{g}(\boldsymbol{\theta}_{t-\tau}, o_{t-\tau})\|^2 + \alpha^2\|\mathbf{g}(\boldsymbol{\theta}_{t-\tau}, o_{t-\tau})\|^2 \\ &\overset{\alpha<1}{\leq} (1+\alpha)r_t^2 + 2\alpha\|\mathbf{g}(\boldsymbol{\theta}_{t-\tau}, o_{t-\tau})\|^2, \end{aligned} \tag{C.31}$$

and note that, for $t < \tau$, $r_{t+1}^2 = r_t^2$, and hence (C.31) holds true for all $t \geq 0$. Now note that

$$\|\mathbf{g}(\boldsymbol{\theta}_{t-\tau}, o_{t-\tau})\|^2 \leq 2L^2(\|\boldsymbol{\theta}_{t-\tau}\|^2 + \sigma^2) \leq 4L^2 r_{t-\tau}^2 + 6L^2\sigma^2. \tag{C.32}$$

Therefore, we can write

$$r_{t+1}^2 \leq (1+\alpha)r_t^2 + 8\alpha L^2 r_{t-\tau}^2 + 12\alpha L^2\sigma^2. \tag{C.33}$$

Now, we show that, for $k < \tau_{mix}$,

$$r_k^2 \leq \rho^k r_0^2 + \epsilon_k, \tag{C.34}$$

with $\epsilon_k = \rho\epsilon_{k-1} + \beta$, $\epsilon_0 = 0$, where $\rho = 1 + \alpha + 8\alpha L^2 > 1$, and $\beta = 12\alpha L^2\sigma^2$. We show it

by induction. The base case $k = 0$ is trivially true. Now suppose that inequality (C.34) is true up to some $k \geq 0$, thus

$$r_s^2 \leq \rho^s r_0^2 + \epsilon_s, \quad \forall s \leq k. \tag{C.35}$$

We can get, noting that, for all k, $0 \leq \epsilon_k \leq \epsilon_{k+1}$,

$$\begin{aligned} r_{k+1}^2 &\leq (1+\alpha) r_k^2 + 8\alpha L^2 r_{k-\tau} + 12\alpha L^2 \sigma^2 \\ &\leq (1+\alpha)(\rho^k r_0^2 + \epsilon_k) + 8\alpha L^2 (\rho^k r_0^2 + \epsilon_k) + 12\alpha L^2 \sigma^2 \\ &= (1 + \alpha + 8\alpha L^2)\rho^k r_0^2 + (1 + \alpha + 8\alpha L^2)\epsilon_k + 12\alpha L^2 \sigma^2 \\ &= \rho^{k+1} r_0^2 + \rho \epsilon_k + \beta \\ &= \rho^{k+1} r_0^2 + \epsilon_{k+1}, \end{aligned} \tag{C.36}$$

which concludes the induction proof of (C.34). Now note that, given that $L \geq 1$, $\rho \leq 1 + 9\alpha L^2$, and, for $\alpha \leq \frac{1}{36 L^2 \tau_{mix}}$

$$\rho^k \leq (1 + 9\alpha L^2)^k \leq (1 + 9\alpha L^2)_{mix}^{\tau} \leq e^{9\alpha L^2 \tau_{mix}} \leq e^{0.25} \leq 2. \tag{C.37}$$

Also note that, for all $k \leq \tau_{mix}$,

$$\epsilon_k = \beta \sum_{j=0}^{k-1} (1 + 9\alpha L^2)^j \leq \beta \sum_{j=0}^{\tau_{mix}-1} (1 + 9\alpha L^2)_{mix}^{\tau} \leq 2\beta \tau_{mix}, \tag{C.38}$$

and we can get, for all $k \leq \tau_{mix}$, noting that $r_0^2 \leq 4\sigma^2$,

$$r_k^2 \leq 2 r_0^2 + 2\beta \tau_{mix} = 2 r_0^2 + 24 \alpha L^2 \sigma^2 \tau_{mix} \leq 9\sigma^2. \tag{C.39}$$

Now note that, similarly to the calculations performed above, for $t < \tau_{mix}$,

$$\begin{aligned} \tilde{r}_{t+1}^2 &= \tilde{r}_t^2 + 2\alpha \langle \boldsymbol{\theta}_t - \boldsymbol{\theta}^*, \mathbf{g}(\boldsymbol{\theta}_t, o_t) \rangle + \alpha^2 \|\mathbf{g}(\boldsymbol{\theta}_t, o_t)\|^2 \\ &\leq (1+\alpha)\tilde{r}_t^2 + 2\alpha \|\mathbf{g}(\boldsymbol{\theta}_t, o_t)\|^2 \\ &\overset{(C.5)}{\leq} (1+\alpha)\tilde{r}_t^2 + 4\alpha L^2 (\|\boldsymbol{\theta}_t\|^2 + \sigma^2) \\ &\leq (1+\alpha)\tilde{r}_t^2 + 4\alpha L^2 (2 r_t^2 + 3\sigma^2). \end{aligned} \tag{C.40}$$

Using the bound established in (C.39), we can get

$$\begin{aligned} \tilde{r}_{t+1}^2 &\leq (1+\alpha)\tilde{r}_t^2 + 8\alpha L^2 r_t^2 + 12\alpha L^2 \sigma^2 \\ &\leq (1+\alpha)\tilde{r}_t^2 + 84\alpha L^2 \sigma^2. \end{aligned} \tag{C.41}$$

From this, we can proceed as follows:

$$\begin{aligned}\tilde{r}_{t+1}^2 &\leq (1+\alpha)\tilde{r}_t^2 + 84\alpha L^2\sigma^2 \\ &\leq (1+\alpha)^2\tilde{r}_{t-1}^2 + (1+\alpha)84\alpha L^2\sigma^2 + 84\alpha L^2\sigma^2 \\ &\leq (1+\alpha)^{t+1}\tilde{r}_0^2 + 84\alpha L^2\sigma^2\sum_{j=0}^{t}(1+\alpha)^j.\end{aligned} \tag{C.42}$$

So, for $0 \leq t < \tau_{mix}$,

$$\tilde{r}_{t+1}^2 \leq (1+\alpha)^{\tau_{mix}}\tilde{r}_0^2 + 84\alpha L^2\sigma^2\sum_{j=0}^{\tau_{mix}}(1+\alpha)^j. \tag{C.43}$$

Now, given that $L \geq 1$, note that, for $\alpha\tau_{mix} \leq \frac{1}{36L^2}$ and $j = 0, ..., \tau_{mix} - 1$, we have $(1+\alpha)^j \leq (1+\alpha)_{mix}^{\tau} \leq e^{\alpha\tau_{mix}} \leq e^{0.25} \leq 2$. Thus, we get

$$\tilde{r}_t^2 \leq 2\tilde{r}_0^2 + 84\alpha L^2\sigma^2\tau_{mix} \leq 11\sigma^2. \tag{C.44}$$

Finally,

$$\begin{aligned}h_t &\leq 9L^2\tilde{r}_t^2 + 12L^2\sigma^2 \\ &\leq 9L^2(11\sigma^2) + 12L^2\sigma^2 \\ &\leq 111L^2\sigma^2.\end{aligned} \tag{C.45}$$

.

We now analyze the case in which $t \geq \tau_{mix}$. Adding and subtracting $\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}}$ in the left hand side of the inner product, we have

$$\begin{aligned}h_t &= \langle\tilde{\boldsymbol{\theta}}_t - \boldsymbol{\theta}^*, \mathbf{g}(\tilde{\boldsymbol{\theta}}_t, o_t) - \bar{\mathbf{g}}(\tilde{\boldsymbol{\theta}}_t)\rangle \\ &= \underbrace{\langle\tilde{\boldsymbol{\theta}}_k - \tilde{\boldsymbol{\theta}}_{t-\tau_{mix}}, \mathbf{g}(\tilde{\boldsymbol{\theta}}_k, o_t) - \bar{\mathbf{g}}(\tilde{\boldsymbol{\theta}}_k)\rangle}_{T_1} + \underbrace{\langle\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}} - \boldsymbol{\theta}^*, \mathbf{g}(\tilde{\boldsymbol{\theta}}_k, o_t) - \bar{\mathbf{g}}(\tilde{\boldsymbol{\theta}}_k)\rangle}_{T_2},\end{aligned} \tag{C.46}$$

where, using (C.4), Cauchy-Schwarz inequality and Lemma 5.2,

$$
\begin{aligned}
T_1 &\leq \|\tilde{\boldsymbol{\theta}}_k - \tilde{\boldsymbol{\theta}}_{t-\tau_{mix}}\|(\|\mathbf{g}(\tilde{\boldsymbol{\theta}}_k, o_t)\| + \|\bar{\mathbf{g}}(\tilde{\boldsymbol{\theta}}_k)\|) \\
&\overset{(C.4)}{\leq} \|\tilde{\boldsymbol{\theta}}_k - \tilde{\boldsymbol{\theta}}_{t-\tau_{mix}}\| 2L(\|\tilde{\boldsymbol{\theta}}_k\| + \sigma) \\
&\leq 2\alpha L^2 \left(\sigma\tau_{mix} + \sum_{l=t-\tau_{mix}}^{t-1} \|\boldsymbol{\theta}_l\|\right)(\|\tilde{\boldsymbol{\theta}}_k\| + \sigma) \\
&\leq 2\alpha L^2 \sigma\tau_{mix}(\|\tilde{\boldsymbol{\theta}}_k\| + \sigma) + 2\alpha L^2 \sum_{l=t-\tau_{mix}}^{t-1} \|\boldsymbol{\theta}_l\|(\|\tilde{\boldsymbol{\theta}}_k\| + \sigma) \\
&\overset{(C.7)}{\leq} 2\alpha L^2\sigma^2\tau_{mix} + 2\alpha L^2\tau_{mix}\sigma\|\tilde{\boldsymbol{\theta}}_k\| \\
&+ 2\alpha L^2 \sum_{l=t-\tau_{mix}}^{t-1} \left(\frac{1}{2}\|\boldsymbol{\theta}_l\|^2 + \frac{1}{2}(\|\tilde{\boldsymbol{\theta}}_k\| + \sigma)^2\right) \\
&\overset{(C.8)}{\leq} 2\alpha L^2\sigma^2\tau_{mix} + \alpha L^2\tau_{mix}\sigma^2 + \alpha L^2\tau_{mix}\|\tilde{\boldsymbol{\theta}}_k\|^2 \\
&+ 2\alpha L^2 \sum_{l=t-\tau_{mix}}^{t-1} \left(\frac{1}{2}\|\boldsymbol{\theta}_l\|^2 + \|\tilde{\boldsymbol{\theta}}_k\|^2 + \sigma^2\right) \\
&\leq 11\alpha L^2\sigma^2\tau_{mix} + 6\alpha L^2\tau_{mix}\tilde{r}_t^2 + \alpha L^2 \sum_{l=t-\tau_{mix}}^{t-1} \|\boldsymbol{\theta}_l\|^2.
\end{aligned}
\tag{C.47}
$$

So, taking the expectation,

$$
\mathbb{E}[T_1] \leq 11\alpha L^2\sigma^2\tau_{mix} + 6\alpha L^2\tau_{mix}\mathbb{E}\left[\tilde{r}_t^2\right] + \alpha L^2 \sum_{l=t-\tau_{mix}}^{t-1} \mathbb{E}\left[\|\boldsymbol{\theta}_l\|^2\right]. \tag{C.48}
$$

Now, we focus on T_2. Note that, adding and subtracting $\mathbf{g}(\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}}, o_t)$ and $\bar{\mathbf{g}}(\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}})$ to the right hand side of the inner product, we can write

$$
\begin{aligned}
T_2 &= \langle \tilde{\boldsymbol{\theta}}_{t-\tau_{mix}} - \boldsymbol{\theta}^*, \mathbf{g}(\tilde{\boldsymbol{\theta}}_k, o_t) - \bar{\mathbf{g}}(\tilde{\boldsymbol{\theta}}_k)\rangle \\
&= \bar{T}_1 + \bar{T}_2 + \bar{T}_3
\end{aligned}
\tag{C.49}
$$

with

$$
\begin{aligned}
\bar{T}_1 &= \langle \tilde{\boldsymbol{\theta}}_{t-\tau_{mix}} - \boldsymbol{\theta}^*, \mathbf{g}(\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}}, o_t) - \bar{\mathbf{g}}(\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}})\rangle \\
\bar{T}_2 &= \langle \tilde{\boldsymbol{\theta}}_{t-\tau_{mix}} - \boldsymbol{\theta}^*, \mathbf{g}(\tilde{\boldsymbol{\theta}}_t, o_t) - \mathbf{g}(\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}}, o_t)\rangle \\
\bar{T}_3 &= \langle \tilde{\boldsymbol{\theta}}_{t-\tau_{mix}} - \boldsymbol{\theta}^*, \bar{\mathbf{g}}(\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}}) - \bar{\mathbf{g}}(\tilde{\boldsymbol{\theta}}_t)\rangle.
\end{aligned}
\tag{C.50}
$$

We first bound $\bar{T}_2$ and $\bar{T}_3$. Note that, using the Lipschitz property of the TD update

direction (C.3) and Lemma 5.2,

$$\begin{aligned}
\bar{T}_2 &\leq \|\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}} - \boldsymbol{\theta}^*\| \|\mathbf{g}(\tilde{\boldsymbol{\theta}}_k, o_t) - \mathbf{g}(\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}}, o_t)\| \\
&\leq L\|\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}} - \boldsymbol{\theta}^*\| \|\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}} - \tilde{\boldsymbol{\theta}}_k\| \\
&\leq L\|\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}} - \tilde{\boldsymbol{\theta}}_t + \tilde{\boldsymbol{\theta}}_t - \boldsymbol{\theta}^*\| \|\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}} - \tilde{\boldsymbol{\theta}}_k\| \\
&\leq L\tilde{r}_t \|\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}} - \tilde{\boldsymbol{\theta}}_k\| + L\|\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}} - \tilde{\boldsymbol{\theta}}_k\|^2 \\
&\leq L^2\alpha \left(\sigma\tau_{mix} + \sum_{l=t-\tau_{mix}}^{t-1} \|\boldsymbol{\theta}_l\|\right) \tilde{r}_t + L\left(2L^2\alpha^2\tau_{mix}^2\sigma^2 + 2L^2\alpha^2\tau_{mix} \sum_{l=t-\tau_{mix}}^{t-1} \|\boldsymbol{\theta}_l\|^2\right) \\
&= L^2\alpha\tau_{mix}\frac{1}{2}\left(\sigma^2 + \tilde{r}_t^2\right) + \frac{1}{2}L^2\alpha \sum_{l=t-\tau_{mix}}^{t-1} \left(\|\boldsymbol{\theta}_l\|^2 + \tilde{r}_t^2\right) \\
&+ 2L^3\alpha^2\tau_{mix}^2\sigma^2 + 2L^3\alpha^2\tau_{mix} \sum_{l=t-\tau_{mix}}^{t-1} \|\boldsymbol{\theta}_l\|^2 \\
&\leq \alpha\tau_{mix}L^2\sigma^2 + \alpha\tau_{mix}L^2\tilde{r}_t^2 + \alpha L^2 \sum_{l=t-\tau_{mix}}^{t-1} \|\boldsymbol{\theta}_l\|^2,
\end{aligned} \tag{C.51}$$

where in the last inequality we used $\alpha \leq \frac{1}{8\tau_{mix}L}$. Taking the expectation,

$$\mathbb{E}\left[\bar{T}_2\right] \leq \alpha\tau_{mix}L^2\sigma^2 + \alpha\tau_{mix}L^2\mathbb{E}\left[\tilde{r}_t^2\right] + \alpha L^2 \sum_{l=t-\tau_{mix}}^{t-1} \mathbb{E}\left[\|\boldsymbol{\theta}_l\|^2\right]. \tag{C.52}$$

With the same calculations, we can get

$$\mathbb{E}\left[\bar{T}_3\right] \leq \alpha\tau_{mix}L^2\sigma^2 + \alpha\tau_{mix}L^2\mathbb{E}\left[\tilde{r}_t^2\right] + \alpha L^2 \sum_{l=t-\tau_{mix}}^{t-1} \mathbb{E}\left[\|\boldsymbol{\theta}_l\|^2\right]. \tag{C.53}$$

We now proceed to bound $\bar{T}_1$.

$$\begin{aligned}
\mathbb{E}\left[\bar{T}_1\right] &= \mathbb{E}\left[\langle \tilde{\boldsymbol{\theta}}_{t-\tau_{mix}} - \boldsymbol{\theta}^*, \mathbf{g}(\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}}, o_t) - \bar{\mathbf{g}}(\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}})\rangle\right] \\
&= \mathbb{E}\left[\langle \tilde{\boldsymbol{\theta}}_{t-\tau_{mix}} - \boldsymbol{\theta}^*, \mathbb{E}\left[\mathbf{g}(\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}}, o_t)|o_{t-\tau_{mix}}, \tilde{\boldsymbol{\theta}}_{t-\tau_{mix}}\right] - \bar{\mathbf{g}}(\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}})\rangle\right] \\
&\leq \mathbb{E}\left[\|\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}} - \boldsymbol{\theta}^*\| \|\mathbb{E}\left[\mathbf{g}(\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}}, o_t)|o_{t-\tau_{mix}}, \tilde{\boldsymbol{\theta}}_{t-\tau_{mix}}\right] - \bar{\mathbf{g}}(\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}})\|\right] \\
&\overset{(*)}{\leq} \alpha\mathbb{E}\left[\|\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}} - \boldsymbol{\theta}^*\|(\|\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}}\| + \sigma)\right] \\
&\leq \alpha\mathbb{E}\left[\|\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}} - \boldsymbol{\theta}^*\|(\|\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}} - \boldsymbol{\theta}^*\| + 2\sigma)\right] \\
&\leq \alpha\mathbb{E}\left[\frac{1}{2}\|\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}} - \boldsymbol{\theta}^*\|^2 + \frac{1}{2}(\|\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}} - \boldsymbol{\theta}^*\| + 2\sigma)^2\right] \\
&\leq \alpha\mathbb{E}\left[\frac{1}{2}\|\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}} - \boldsymbol{\theta}^*\|^2 + \|\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}} - \boldsymbol{\theta}^*\|^2 + 2\sigma^2\right] \\
&\leq 2\alpha\mathbb{E}\left[\|\tilde{\boldsymbol{\theta}}_{t-\tau_{mix}} - \boldsymbol{\theta}^*\|^2 + \sigma^2\right] \\
&\leq 2\alpha\mathbb{E}\left[2\|\tilde{\boldsymbol{\theta}}_t - \boldsymbol{\theta}^*\|^2 + 2\|\tilde{\boldsymbol{\theta}}_t - \tilde{\boldsymbol{\theta}}_{t-\tau_{mix}}\|^2 + \sigma^2\right] \\
&\leq 2\alpha\mathbb{E}\left[2\tilde{r}_t^2 + 2(2L^2\alpha^2\tau_{mix}^2\sigma^2 + 2L^2\alpha^2\tau_{mix}\sum_{l=t-\tau_{mix}}^{t-1}\|\boldsymbol{\theta}_l\|^2) + \sigma^2\right] \\
&\leq 4\alpha\mathbb{E}\left[\tilde{r}_t^2\right] + 3\alpha\sigma^2 + \alpha\sum_{l=t-\tau_{mix}}^{t-1}\mathbb{E}\left[\|\boldsymbol{\theta}_l\|^2\right],
\end{aligned} \tag{C.54}$$

where for $(*)$ we used Definition 5.2.1 of mixing time and the fact that $\sigma \geq 1$, and in the last inequality we used $\alpha \leq \frac{1}{8\tau_{mix}L}$. So, we get

$$\begin{aligned}
\mathbb{E}\left[T_2\right] &= \mathbb{E}\left[\bar{T}_1\right] + \mathbb{E}\left[\bar{T}_2\right] + \mathbb{E}\left[\bar{T}_3\right] \\
&\leq 6\alpha\tau_{mix}L^2\mathbb{E}\left[\tilde{r}_t^2\right] + 5\alpha\tau_{mix}L^2\sigma^2 + 3\alpha L^2\sum_{l=t-\tau_{mix}}^{t-1}\mathbb{E}\left[\|\boldsymbol{\theta}_l\|^2\right].
\end{aligned} \tag{C.55}$$

Finally, we get

$$\begin{aligned}
\mathbb{E}\left[h_t\right] &= \mathbb{E}\left[T_1\right] + \mathbb{E}\left[T_2\right] \\
&\leq 16\alpha\tau_{mix}L^2\sigma^2 + 12\alpha\tau_{mix}L^2\mathbb{E}\left[\tilde{r}_t^2\right] + 4\alpha L^2 \sum_{l=t-\tau_{mix}}^{t-1} \mathbb{E}\left[\|\boldsymbol{\theta}_l\|^2\right] \\
&\leq 16\alpha\tau_{mix}L^2\sigma^2 + 12\alpha\tau_{mix}L^2\mathbb{E}\left[\tilde{r}_t^2\right] \\
&+ 8\alpha L^2 \sum_{l=t-\tau_{mix}}^{t-1} \left(\mathbb{E}\left[\|\mathbf{d}_l\|^2\right] + \mathbb{E}\left[\|\tilde{\boldsymbol{\theta}}_l\|^2\right]\right) \\
&\leq 32\alpha\tau_{mix}L^2\sigma^2 + 12\alpha\tau_{mix}L^2\mathbb{E}\left[\tilde{r}_t^2\right] + 8\alpha L^2 \sum_{l=t-\tau_{mix}}^{t-1} \mathbb{E}\left[\|\mathbf{d}_l\|^2 + 2\tilde{r}_l^2\right].
\end{aligned} \tag{C.56}$$

□

C.1.2 Proof of Theorem 5.3

First, we have

$$\tilde{r}_{t+1}^2 = \tilde{r}_t^2 + 2\alpha\langle \mathbf{g}(\boldsymbol{\theta}_k, o_t), \tilde{\boldsymbol{\theta}}_t - \boldsymbol{\theta}^\star\rangle + \alpha^2\|\mathbf{g}(\boldsymbol{\theta}_k, o_t)\|^2 \tag{C.57}$$

Then, using (C.1), i.e., $\langle \bar{\mathbf{g}}(\tilde{\boldsymbol{\theta}}_t), \tilde{\boldsymbol{\theta}}_t - \boldsymbol{\theta}^\star\rangle \leq -\mu\tilde{r}_t^2$,

$$\begin{aligned}
\tilde{r}_{t+1}^2 &= \tilde{r}_t^2 + 2\alpha\langle \mathbf{g}(\boldsymbol{\theta}_k, o_t), \tilde{\boldsymbol{\theta}}_t - \boldsymbol{\theta}^*\rangle + \alpha^2\|\mathbf{g}(\boldsymbol{\theta}_k, o_t)\|^2 \\
&= \tilde{r}_t^2 + 2\alpha\langle \bar{\mathbf{g}}(\tilde{\boldsymbol{\theta}}_t), \tilde{\boldsymbol{\theta}}_t - \boldsymbol{\theta}^\star\rangle + 2\alpha h_t + 2\alpha m_t + \alpha^2\|\mathbf{g}(\boldsymbol{\theta}_k, o_t)\|^2 \\
&\leq (1 - 2\alpha\mu)\tilde{r}_t^2 + 2\alpha h_t + 2\alpha m_t + \alpha^2 n_t.
\end{aligned} \tag{C.58}$$

We now apply the inequalities obtained in Lemma 5.3 to bound $\mathbb{E}\left[h_t\right]$, m_t and n_t. Recall that $\bar{\tau} = \max\{\tau, \tau_{mix}\}$. Note that, from Lemma 5.3 - (iii), we can write $\mathbb{E}\left[h_t\right] \leq \bar{h}_t$, defining

$$\bar{h}_t = \begin{cases} B & \text{if } 0 \leq t < \tau_{mix} \\ q_t & \text{if } t \geq \tau_{mix} \end{cases}, \tag{C.59}$$

with $B = 111\sigma^2$, and

$$q_t = \alpha\tau_{mix}L^2\left(32\sigma^2 + 12\mathbb{E}\left[\tilde{r}_t^2\right]\right) + 8\alpha L^2 \sum_{l=t-\tau_{mix}}^{t-1} \mathbb{E}\left[\|\mathbf{d}_l\|^2 + 2\tilde{r}_l^2\right]. \tag{C.60}$$

As a consequence, we can write, for every $t \geq 0$,

$$\mathbb{E}\left[h_t\right] \leq q_t + \bar{B}_t \tag{C.61}$$

where, in turn,

$$\bar{B}_t = \begin{cases} B & \text{if } 0 \leq t < \tau_{mix} \\ 0 & \text{otherwise} \end{cases}. \tag{C.62}$$

Also, recall that, from Lemma 5.3, we have

$$\begin{aligned} n_t &\leq 4L^2\|\mathbf{d}_t\|^2 + 8L^2\tilde{r}_t^2 + 10L^2\sigma^2, \\ m_t &\leq 6\alpha\tau L^2\sigma^2 + 3\alpha\tau L^2\tilde{r}_t^2 + 2\alpha L^2 \sum_{l=t-\tau}^{t-1} \left(\|\mathbf{d}_l\|^2 + 2\tilde{r}_l^2\right) \end{aligned} \tag{C.63}$$

Combining these inequalities together, we have, for $t \geq 0$,

$$\begin{aligned} \mathbb{E}\left[\tilde{r}_{t+1}^2\right] &\leq (1-2\alpha\mu)\mathbb{E}\left[\tilde{r}_t^2\right] + 2\alpha\mathbb{E}\left[h_t\right] + 2\alpha\mathbb{E}\left[m_t\right] + \alpha^2\mathbb{E}\left[\|\mathbf{g}(\boldsymbol{\theta}_k, o_t)\|^2\right] \\ &\leq (1-2\alpha\mu)\mathbb{E}\left[\tilde{r}_t^2\right] + 2\alpha^2\tau_{mix}L^2\left(32\sigma^2 + 12\mathbb{E}\left[\tilde{r}_t^2\right]\right) \\ &\quad + 16\alpha^2L^2 \sum_{l=t-\tau_{mix}}^{t-1} \mathbb{E}\left[\|\mathbf{d}_l\|^2 + 2\tilde{r}_l^2\right] \\ &\quad + 12\alpha^2\tau L^2\sigma^2 + 6\alpha^2\tau L^2\mathbb{E}\left[\tilde{r}_t^2\right] + 4\alpha^2L^2 \sum_{l=t-\tau}^{t-1} \mathbb{E}\left[\|\mathbf{d}_l\|^2 + 2\tilde{r}_l^2\right] \\ &\quad + 4\alpha^2L^2\mathbb{E}\left[\|\mathbf{d}_t\|^2 + 2\tilde{r}_t^2\right] + 10\alpha^2L^2\sigma^2 + 2\alpha\bar{B}_t. \end{aligned} \tag{C.64}$$

Combining terms, we can get

$$\begin{aligned} \mathbb{E}\left[\tilde{r}_{t+1}^2\right] &\leq (1 - 2\alpha\mu + 48\alpha^2L^2\bar{\tau})\mathbb{E}\left[\tilde{r}_t^2\right] + 128\alpha^2L^2\bar{\tau}\sigma^2 \\ &\quad + 4\alpha^2L^2\mathbb{E}\left[\|\mathbf{d}_t\|^2\right] + 20\alpha^2L^2 \sum_{l=t-\bar{\tau}}^{t-1} \mathbb{E}\left[\|\mathbf{d}_l\|^2 + 2\tilde{r}_l^2\right] + 2\alpha\bar{B}_t, \end{aligned} \tag{C.65}$$

where we have used $\tau + \tau_{mix} \leq 2\bar{\tau}$. Now, using the fact that $r_t^2 \leq 2\tilde{r}_t^2 + 2\|\mathbf{d}_t\|^2$, which implies $-\tilde{r}_t^2 \leq -\frac{r_t^2}{2} + \|\mathbf{d}_t\|^2$, we have

$$\begin{aligned} (1 - 2\alpha\mu + 48\alpha^2L^2\bar{\tau})\mathbb{E}\left[\tilde{r}_t^2\right] &= (1 - \alpha\mu + 48\alpha^2L^2\bar{\tau})\mathbb{E}\left[\tilde{r}_t^2\right] - \alpha\mu\mathbb{E}\left[\tilde{r}_t^2\right] \\ &\leq (1 - \alpha\mu + 48\alpha^2L^2\bar{\tau})\mathbb{E}\left[\tilde{r}_t^2\right] - \alpha\mu\frac{\mathbb{E}\left[r_t^2\right]}{2} + \alpha\mathbb{E}\left[\|\mathbf{d}_t\|^2\right], \end{aligned} \tag{C.66}$$

and, using $\mu \leq 1$, we can re-write (C.65) as

$$\begin{aligned}\mathbb{E}\left[\tilde{r}_{t+1}^2\right] \leq (1-\alpha\mu+48\alpha^2L^2\bar{\tau})\mathbb{E}\left[\tilde{r}_t^2\right] - \alpha\mu\frac{\mathbb{E}\left[r_t^2\right]}{2} + 128\alpha^2L^2\bar{\tau}\sigma^2 \\ + \alpha(1+4\alpha L^2)\mathbb{E}\left[\|\mathbf{d}_t\|^2\right] + 20\alpha^2L^2\sum_{l=t-\bar{\tau}}^{t-1}\mathbb{E}\left[\|\mathbf{d}_l\|^2 + 2\tilde{r}_l^2\right] + 2\alpha\bar{B}_t,\end{aligned} \tag{C.67}$$

Multiplying both sides by w_t, we have

$$\begin{aligned}w_t\mathbb{E}\left[\tilde{r}_{t+1}^2\right] \leq (1-\alpha\mu+48\alpha^2L^2\bar{\tau})w_t\mathbb{E}\left[\tilde{r}_t^2\right] - \alpha\mu\frac{w_t\mathbb{E}\left[r_t^2\right]}{2} + 128w_t\alpha^2L^2\bar{\tau}\sigma^2 \\ + \alpha(1+4\alpha L^2)w_t\mathbb{E}\left[\|\mathbf{d}_t\|^2\right] + 20\alpha^2L^2w_t\sum_{l=t-\bar{\tau}}^{t-1}\mathbb{E}\left[\|\mathbf{d}_l\|^2 + 2\tilde{r}_l^2\right] + 2\alpha w_t\bar{B}_t,\end{aligned} \tag{C.68}$$

by summing over $t = 0, ..., T$, we get, with $W_T = \sum_{t=0}^{T} w_t$,

$$\begin{aligned}\sum_{t=0}^{T} w_t\mathbb{E}\left[\tilde{r}_{t+1}^2\right] \leq (1-\alpha\mu+48\alpha^2L^2\bar{\tau})\sum_{t=0}^{T} w_t\mathbb{E}\left[\tilde{r}_t^2\right] - \frac{\alpha\mu}{2}\sum_{t=0}^{T} w_t\mathbb{E}\left[r_t^2\right] \\ + 128W_T\alpha^2L^2\bar{\tau}\sigma^2 + \alpha(1+4\alpha L^2)\underbrace{\sum_{t=0}^{T} w_t\mathbb{E}\left[\|\mathbf{d}_t\|^2\right]}_{p_1} \\ + 20\alpha^2L^2\underbrace{\sum_{t=0}^{T} w_t\sum_{l=t-\bar{\tau}}^{t-1}\mathbb{E}\left[\|\mathbf{d}_l\|^2 + 2\tilde{r}_l^2\right]}_{p_2} + 2W_{\tau_{mix}-1}\alpha B\end{aligned} \tag{C.69}$$

Note that, from Lemma 5.1 - (iii), we have, picking $\alpha \leq \frac{1}{72\tau L^2}$,

$$\begin{aligned}p_1 = \sum_{t=0}^{T} w_t\mathbb{E}\left[\|\mathbf{d}_t\|^2\right] &\leq 4W_T\alpha^2\tau^2L^2\sigma^2 + 16\alpha^2\tau^2L^2\sum_{t=0}^{T} w_t\mathbb{E}\left[\|\tilde{\boldsymbol{\theta}}_t\|^2\right] \\ &\leq 36W_T\alpha^2\tau^2L^2\sigma^2 + 32\alpha^2\tau^2L^2\sum_{t=0}^{T} w_t\mathbb{E}\left[\tilde{r}_t^2\right] \\ &\leq \frac{\alpha\tau W_T\sigma^2}{2} + \frac{\alpha\tau}{2}\sum_{t=0}^{T} w_t\mathbb{E}\left[\tilde{r}_t^2\right].\end{aligned} \tag{C.70}$$

Furthermore, using the fact that $w_t \leq 2w_l$ for $l = t - \bar{\tau}, ..., t-1$, we can bound p_2 as follows, using also the above bound on p_1, and picking $\alpha \leq \frac{1}{72\tau L^2}$,

$$\begin{aligned} p_2 &= \sum_{t=0}^{T} w_t \sum_{l=t-\bar{\tau}}^{t-1} \mathbb{E}\left[\|\mathbf{d}_l\|^2 + 2\tilde{r}_l^2\right] \\ &\overset{(a)}{\leq} 2\sum_{t=0}^{T}\sum_{l=t-\bar{\tau}}^{t-1} w_l \mathbb{E}\left[\|\mathbf{d}_l\|^2 + 2\tilde{r}_l^2\right] \\ &\overset{(b)}{\leq} 2\bar{\tau}\sum_{t=0}^{T} w_t \mathbb{E}\left[\|\mathbf{d}_t\|^2 + 2\tilde{r}_t^2\right]. \\ &\leq 2\bar{\tau}\sum_{t=0}^{T} w_t \mathbb{E}\left[\|\mathbf{d}_t\|^2\right] + 4\bar{\tau}\sum_{t=0}^{T} w_t \mathbb{E}\left[\tilde{r}_t^2\right] \\ &\overset{(c)}{\leq} 2\bar{\tau}\left(\frac{\alpha\tau W_T \sigma^2}{2} + \frac{\alpha\tau}{2}\sum_{t=0}^{T} w_t \mathbb{E}\left[\tilde{r}_t^2\right]\right) + 4\bar{\tau}\sum_{t=0}^{T} w_t \mathbb{E}\left[\tilde{r}_t^2\right] \\ &\leq 5\bar{\tau}\sum_{t=0}^{T} w_t \mathbb{E}\left[\tilde{r}_t^2\right] + W_T\sigma^2\bar{\tau}, \end{aligned} \tag{C.71}$$

where for (a) we used Lemma C.1, for (b) we used the fact that each element $w_l\|\boldsymbol{\theta}_l\|^2$ appears at most τ times in the sum, for $l = 0, ..., T-1$ (note that, by definition, $\mathbf{d}_l = \tilde{r}_l = 0$ for $l < 0$) and for (c) we used the bound on p_1. In the last inequality we simply used $\alpha\tau \leq 1$. Plugging the two bounds on p_1 and p_2 in (C.69), we get

$$\begin{aligned} \sum_{t=0}^{T} w_t \mathbb{E}\left[\tilde{r}_{t+1}^2\right] \leq (1 - \alpha\mu + 150\alpha^2 L^2\bar{\tau})\sum_{t=0}^{T} w_t \mathbb{E}\left[\tilde{r}_t^2\right] - \frac{\alpha\mu}{2}\sum_{t=0}^{T} w_t \mathbb{E}\left[r_t^2\right] \\ + 150 W_T \alpha^2 L^2 \bar{\tau}\sigma^2 + 2W_{\tau_{mix}-1}\alpha B. \end{aligned} \tag{C.72}$$

Now, note that for $\alpha \leq \frac{\mu}{100L^2\bar{\tau}}$, which is such that $(1 - 2\alpha\mu + 150\alpha^2 L^2\bar{\tau}) \leq (1 - 0.5\alpha\mu)$, we can re-write (C.72) as

$$\begin{aligned} \sum_{t=0}^{T} w_t \mathbb{E}\left[\tilde{r}_{t+1}^2\right] \leq (1 - 0.5\alpha\mu)\sum_{t=0}^{T} w_t \mathbb{E}\left[\tilde{r}_t^2\right] - \frac{\alpha\mu}{2}\sum_{t=0}^{T} w_t \mathbb{E}\left[r_t^2\right] \\ + 150 W_T \alpha^2 L^2 \bar{\tau}\sigma^2 + 2W_{\tau_{mix}-1}\alpha B. \end{aligned} \tag{C.73}$$

Now, dividing by W_T both sides of (C.73), bringing $\sum_{t=0}^{T} w_t \mathbb{E}\left[\tilde{r}_{t+1}^2\right]$ to the right hand

side of the inequality and $-\frac{\alpha\mu}{2}\sum_{t=0}^{T} w_t \mathbb{E}\left[r_t^2\right]$ to the left side, we get

$$\begin{aligned}\frac{\alpha\mu}{2}\sum_{t=0}^{T}\frac{w_t}{W_T}\mathbb{E}\left[r_t^2\right] \leq \frac{1}{W_T}\sum_{t=0}^{T}\left(w_t(1-0.5\alpha\mu)\mathbb{E}\left[\tilde{r}_t^2\right] - w_t\mathbb{E}\left[\tilde{r}_{t+1}^2\right]\right) \\ +150\alpha^2L^2\bar{\tau}\sigma^2 + \frac{2W_{\tau_{mix}-1}\alpha B}{W_T}.\end{aligned} \tag{C.74}$$

Now, recalling that $w_t = (1-0.5\alpha\mu)^{-(t+1)}$, note that $w_t(1-0.5\alpha\mu) = w_{t-1}$, and we can get, noting that $w_{-1} = 1$,

$$\begin{aligned}\sum_{t=0}^{T}\left(w_t(1-0.5\alpha\mu)\mathbb{E}\left[\tilde{r}_t^2\right] - w_t\mathbb{E}\left[\tilde{r}_{t+1}^2\right]\right) = \sum_{t=0}^{T}\left(w_{t-1}\mathbb{E}\left[\tilde{r}_t^2\right] - w_t\mathbb{E}\left[\tilde{r}_{t+1}^2\right]\right) \\ \leq \mathbb{E}\left[\tilde{r}_0^2\right] - w_T\mathbb{E}\left[\tilde{r}_{T+1}^2\right] \leq \tilde{r}_0^2.\end{aligned} \tag{C.75}$$

Hence, we can write (C.74) as

$$\frac{\alpha\mu}{2}\sum_{t=0}^{T}\frac{w_t}{W_T}\mathbb{E}\left[r_t^2\right] \leq \frac{\tilde{r}_0^2}{W_T} + 150\alpha^2L^2\bar{\tau}\sigma^2 + \frac{2W_{\tau_{mix}-1}\alpha B}{W_T}. \tag{C.76}$$

Now note that

$$W_{\tau_{mix}-1} = \sum_{t=0}^{\tau_{mix}-1} w_t = \sum_{t=0}^{\tau_{mix}-1}(1-0.5\alpha\mu)^{-(t+1)} \leq \sum_{t=0}^{\tau_{mix}-1}(1+\alpha\mu)^{t+1} \leq 2\tau_{mix} \tag{C.77}$$

and that

$$\frac{1}{W_T} \leq \frac{1}{w_T} = (1-0.5\alpha\mu)^{T+1} \tag{C.78}$$

from which we can obtain, re-arranging the different terms in (C.76),

$$\begin{aligned}\frac{1}{W_T}\sum_{t=0}^{T} w_t\mathbb{E}\left[r_{t+1}^2\right] &\leq (1-0.5\alpha\mu)^{T+1}\tilde{r}_0^2\left(\frac{2}{\alpha\mu} + \frac{4\bar{\tau}B}{\mu}\right) + 300\frac{\alpha L^2\bar{\tau}\sigma^2}{\mu} \\ &= C_\alpha(1-0.5\alpha\mu)^{T+1}\tilde{r}_0^2 + C_2\frac{\alpha L^2\bar{\tau}\sigma^2}{\mu},\end{aligned} \tag{C.79}$$

where we define $C_\alpha = \left(\frac{2}{\alpha\mu} + \frac{4\bar{\tau}B}{\mu}\right)$ and $C_2 = 300$. By plugging the maximum value for the step size $\alpha = \frac{\mu}{150L^2\bar{\tau}}$, we can get, defining $C_{\bar{\tau}} = \frac{\bar{\tau}}{\mu}\left(\frac{2C_1L^2}{\mu} + 4B\right)$,

$$\frac{1}{W_T}\sum_{t=0}^{T} w_t\mathbb{E}\left[r_{t+1}^2\right] \leq C_2(1-0.5\alpha\mu)^{T+1}\tilde{r}_0^2 + 2\sigma^2. \tag{C.80}$$

Indeed, for $\alpha = \frac{\mu}{150L^2\bar{\tau}}$, it holds $C_2\frac{\alpha L^2\bar{\tau}}{\mu} = 2$. Finally, by definition of $\boldsymbol{\theta}_{out}$ in Theorem 5.3, note that we have

$$\mathbb{E}\left[\|\boldsymbol{\theta}_{out} - \boldsymbol{\theta}^*\|^2\right] = \frac{1}{W_T}\sum_{t=0}^{T} w_t \mathbb{E}\left[r_t^2\right] \tag{C.81}$$

and we can conclude the proof of Theorem 5.3. □

C.2 Proof of Theorem 5.4

Let $r_t := \|\boldsymbol{\theta}_k - \boldsymbol{\theta}^*\|$. Define $\tau' = 2\tau_{max} + \tau_{mix}$, and recall

$$r_{t,2} := \max_{t-\tau' \le l \le t} \mathbb{E}\left[r_l^2\right]. \tag{C.82}$$

C.2.1 Proofs of Auxliary Lemmas

We start by proving Lemma 5.4, i.e., the bounds on terms of the form $\|\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t-\tau}\|^2$, for some $0 \le \tau \le t$.

Proof of Lemma 5.4. To prove (i), note that we can get

$$\begin{aligned}
\|\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t-\tau_{mix}}\|^2 &\le \left(\sum_{l=t-\tau_{mix}}^{t-1} \|\boldsymbol{\theta}_{l+1} - \boldsymbol{\theta}_l\|\right)^2 \\
&\overset{(C.8)}{\le} \tau_{mix} \sum_{l=t-\tau_{mix}}^{t-1} \|\boldsymbol{\theta}_{l+1} - \boldsymbol{\theta}_l\|^2 \\
&= \tau_{mix}\alpha^2 \sum_{l=t-\tau_{mix}}^{t-1} \|\mathbf{g}(\boldsymbol{\theta}_{l-\tau_l}, o_{l-\tau_l})\|^2 \\
&\overset{(C.5)}{\le} 2\alpha^2\tau_{mix}L^2 \sum_{l=t-\tau_{mix}}^{t-1} (\|\boldsymbol{\theta}_{l-\tau_l}\|^2 + \sigma^2) \\
&\le 2\alpha^2\tau_{mix}L^2 \sum_{l=t-\tau_{mix}}^{t-1} (2r_{l-\tau_l}^2 + 3\sigma^2).
\end{aligned} \tag{C.83}$$

Taking the expectation on both sides of the inequality, we get

$$\begin{aligned}\mathbb{E}\left[\|\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t-\tau_{mix}}\|^2\right] &\leq 2\alpha^2\tau_{mix}L^2 \sum_{l=t-\tau_{mix}}^{t-1} (2\mathbb{E}\left[r_{l-\tau_l}^2\right] + 3\sigma^2) \\ &\leq 2\alpha^2\tau_{mix}L^2 \sum_{l=t-\tau_{mix}}^{t-1} (2 \max_{t-\tau_{mix}-\tau_{max}\leq j\leq t} \mathbb{E}\left[r_j^2\right] + 3\sigma^2) \\ &\leq 4\tau_{mix}^2\alpha^2L^2r_{t,2} + 6\alpha^2\tau_{mix}^2L^2\sigma^2 \\ &= 2\alpha^2\tau_{mix}^2L^2(2r_{t,2} + 3\sigma^2).\end{aligned} \tag{C.84}$$

With analogous computations, we can get part (ii) of the Lemma, i.e.

$$\mathbb{E}\left[\|\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t-\tau_t}\|^2\right] \leq 2\alpha^2\tau_{max}^2L^2(2r_{t,2} + 3\sigma^2). \tag{C.85}$$

□

Recall the definition of $\mathbf{e}_t$,

$$\mathbf{e}_t := \mathbf{g}(\boldsymbol{\theta}_k, o_t) - \mathbf{g}(\boldsymbol{\theta}_{t-\tau_t}, o_{t-\tau_t}). \tag{C.86}$$

As illustrated in the outline of the analysis in Section 5.4, for the purpose of the analysis, we write the update rule as follows,

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k + \alpha\mathbf{g}(\boldsymbol{\theta}_k, o_t) - \alpha\mathbf{e}_t, \tag{C.87}$$

from which we can write

$$\|\boldsymbol{\theta}_{k+1} - \boldsymbol{\theta}^*\|^2 = J_{t,1} + \alpha^2 J_{t,2} - 2\alpha J_{t,3}, \tag{C.88}$$

with

$$\begin{aligned}J_{t,1} &:= \|\boldsymbol{\theta}_k - \boldsymbol{\theta}^* + \alpha\mathbf{g}(\boldsymbol{\theta}_k, o_t)\|^2 \\ J_{t,2} &:= \|\mathbf{e}_t\|^2 \\ J_{t,3} &:= \langle \mathbf{e}_t, \boldsymbol{\theta}_k - \boldsymbol{\theta}^* + \alpha\mathbf{g}(\boldsymbol{\theta}_k, o_t)\rangle.\end{aligned} \tag{C.89}$$

Proof of Lemma 5.5 - (i). Note that

$$J_{t,1} = \|\boldsymbol{\theta}_k - \boldsymbol{\theta}^* + \alpha \mathbf{g}(\boldsymbol{\theta}_k, o_t)\|^2 = r_t^2 + 2\alpha \underbrace{\langle \boldsymbol{\theta}_k - \boldsymbol{\theta}^*, \mathbf{g}(\boldsymbol{\theta}_k, o_t)\rangle}_{J_{t,11}} + \alpha^2 \underbrace{\|\mathbf{g}(\boldsymbol{\theta}_k, o_t)\|^2}_{J_{t,12}}. \tag{C.90}$$

Note that

$$\begin{aligned} \mathbb{E}\left[J_{t,12}\right] &= \mathbb{E}\left[\|\mathbf{g}(\boldsymbol{\theta}_k, o_t)\|^2\right] \\ &\leq \mathbb{E}\left[2L^2\left(\|\boldsymbol{\theta}_k\|^2 + \sigma^2\right)\right] \\ &\leq 2L^2\left(2\mathbb{E}\left[r_t^2\right] + 3\sigma^2\right) \\ &\leq 2L^2\left(2r_{t,2} + 3\sigma^2\right) \end{aligned} \tag{C.91}$$

Now note that, using (C.1),

$$\begin{aligned} J_{t,11} &= \langle \boldsymbol{\theta}_k - \boldsymbol{\theta}^*, \mathbf{g}(\boldsymbol{\theta}_k, o_t)\rangle = -\langle \boldsymbol{\theta}^* - \boldsymbol{\theta}_k, \bar{\mathbf{g}}(\boldsymbol{\theta}_k)\rangle \\ &\quad + \langle \boldsymbol{\theta}_k - \boldsymbol{\theta}^*, \mathbf{g}(\boldsymbol{\theta}_k, o_t) - \bar{\mathbf{g}}(\boldsymbol{\theta}_k)\rangle \\ &\leq -\mu r_t^2 + \underbrace{\langle \boldsymbol{\theta}_k - \boldsymbol{\theta}^*, \mathbf{g}(\boldsymbol{\theta}_k, o_t) - \bar{\mathbf{g}}(\boldsymbol{\theta}_k)\rangle}_{T_1'}, \end{aligned} \tag{C.92}$$

where we now omit the dependence on the iterate t in the terms we bound, for notation convenience. Now, note that

$$T_1' = \underbrace{\langle \boldsymbol{\theta}_k - \boldsymbol{\theta}_{t-\tau_{mix}}, \mathbf{g}(\boldsymbol{\theta}_k, o_t) - \bar{\mathbf{g}}(\boldsymbol{\theta}_k)\rangle}_{T_{11}'} + \underbrace{\langle \boldsymbol{\theta}_{t-\tau_{mix}} - \boldsymbol{\theta}^*, \mathbf{g}(\boldsymbol{\theta}_k, o_t) - \bar{\mathbf{g}}(\boldsymbol{\theta}_k)\rangle}_{T_{12}'}, \tag{C.93}$$

where, using the Cauchy-Schwarz inequality and the triangle inequality,

$$\begin{aligned} T_{11}' &\leq \|\boldsymbol{\theta}_k - \boldsymbol{\theta}_{t-\tau_{mix}}\|(\|\mathbf{g}(\boldsymbol{\theta}_k, o_t)\| + \|\bar{\mathbf{g}}(\boldsymbol{\theta}_k)\|) \\ &\overset{(C.4)}{\leq} 2L(\|\boldsymbol{\theta}_k - \boldsymbol{\theta}_{t-\tau_{mix}}\|(\|\boldsymbol{\theta}_k\| + \sigma)) \\ &\overset{(*)}{\leq} L\left(\frac{1}{\alpha\tau_{mix}L}\|\boldsymbol{\theta}_k - \boldsymbol{\theta}_{t-\tau_{mix}}\|^2 + \alpha\tau_{mix}L(\|\boldsymbol{\theta}_k\| + \sigma)^2\right) \\ &\overset{(C.8)}{\leq} L\left(\frac{1}{\alpha\tau_{mix}L}\|\boldsymbol{\theta}_k - \boldsymbol{\theta}_{t-\tau_{mix}}\|^2 + 2\alpha\tau_{mix}L(\|\boldsymbol{\theta}_k\|^2 + \sigma^2)\right), \end{aligned} \tag{C.94}$$

where for $(*)$ we used the fact that, from (C.7), we have

$$ab = (\frac{1}{\sqrt{c}}a)(\sqrt{c}b) \leq \frac{1}{2c}a^2 + \frac{cb^2}{2}, \tag{C.95}$$

specifically with $c = \alpha\tau_{mix}L$. Taking the expectation on both sides and applying (ii) of Lemma 5.4, we get

$$\begin{aligned}\mathbb{E}\left[T'_{11}\right] &\leq L\left(\frac{1}{2\alpha\tau_{mix}L}\mathbb{E}\left[\|\boldsymbol{\theta}_k - \boldsymbol{\theta}_{t-\tau_{mix}}\|^2\right] + \alpha\tau_{mix}L(2\mathbb{E}\left[r_t^2\right] + 3\sigma^2)\right)\\ &\leq L\left(\frac{2\alpha^2\tau_{mix}^2L^2}{2\alpha\tau_{mix}L}(2r_{t,2} + 3\sigma^2) + \alpha\tau_{mix}L(2r_{t,2} + 3\sigma^2)\right)\\ &= 4\alpha\tau_{mix}L^2r_{t,2} + 6\alpha\tau_{mix}L^2\sigma^2.\end{aligned} \tag{C.96}$$

Now, we proceed to bound $\mathbb{E}\left[T'_{12}\right]$. Note that

$$\begin{aligned}T'_{12} &= \langle\boldsymbol{\theta}_{t-\tau_{mix}} - \boldsymbol{\theta}^*, \mathbf{g}(\boldsymbol{\theta}_k, o_t) - \bar{\mathbf{g}}(\boldsymbol{\theta}_k)\rangle\\ &= \bar{T}_1 + \bar{T}_2 + \bar{T}_3\end{aligned} \tag{C.97}$$

with

$$\begin{aligned}\bar{T}_1 &= \langle\boldsymbol{\theta}_{t-\tau_{mix}} - \boldsymbol{\theta}^*, \mathbf{g}(\boldsymbol{\theta}_{t-\tau_{mix}}, o_t) - \bar{\mathbf{g}}(\boldsymbol{\theta}_{t-\tau})\rangle\\ \bar{T}_2 &= \langle\boldsymbol{\theta}_{t-\tau_{mix}} - \boldsymbol{\theta}^*, \mathbf{g}(\boldsymbol{\theta}_t, o_t) - \mathbf{g}(\boldsymbol{\theta}_{t-\tau_{mix}}, o_t)\rangle\\ \bar{T}_3 &= \langle\boldsymbol{\theta}_{t-\tau_{mix}} - \boldsymbol{\theta}^*, \bar{\mathbf{g}}(\boldsymbol{\theta}_{t-\tau_{mix}}) - \bar{\mathbf{g}}(\boldsymbol{\theta}_t)\rangle.\end{aligned} \tag{C.98}$$

We first bound $\bar{T}_2$ and $\bar{T}_3$. Note that, using Lipschitz property of the TD update direction (C.3), and calculations similar to the ones used to bound $\mathbb{E}\left[T'_{11}\right]$, we get

$$\begin{aligned}\bar{T}_2 &\leq \|\boldsymbol{\theta}_{t-\tau_{mix}} - \boldsymbol{\theta}^*\|\|\mathbf{g}(\boldsymbol{\theta}_k, o_t) - \mathbf{g}(\boldsymbol{\theta}_{t-\tau_{mix}}, o_t)\|\\ &\leq L\|\boldsymbol{\theta}_{t-\tau_{mix}} - \boldsymbol{\theta}^*\|\|\boldsymbol{\theta}_{t-\tau_{mix}} - \boldsymbol{\theta}_k\|\\ &\overset{(C.95)}{\leq} \frac{L^2\alpha\tau_{mix}}{2}r_{t-\tau_{mix}}^2 + \frac{\|\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t-\tau_{mix}}\|^2}{2\alpha\tau_{mix}}.\end{aligned} \tag{C.99}$$

Taking the expectation and applying (ii) of Lemma 5.4, we can get

$$\begin{aligned}\mathbb{E}\left[\bar{T}_2\right] &\leq \frac{\alpha\tau_{mix}L^2r_{t,2}}{2} + 2\alpha\tau_{mix}L^2r_{t,2} + 3\alpha\tau_{mix}L^2\sigma^2\\ &\leq 3\alpha\tau_{mix}L^2(r_{t,2} + \sigma^2)\end{aligned} \tag{C.100}$$

With the same calculations, we can get

$$\mathbb{E}\left[\bar{T}_3\right] \leq 3\alpha\tau_{mix}L^2(r_{t,2}^2 + \sigma^2). \tag{C.101}$$

We now proceed to bound $\bar{T}_1$.

$$\begin{aligned}
\mathbb{E}\left[\bar{T}_1\right] &= \mathbb{E}\left[\langle \boldsymbol{\theta}_{t-\tau_{mix}} - \boldsymbol{\theta}^*, \mathbf{g}(\boldsymbol{\theta}_{t-\tau_{mix}}, o_t) - \bar{\mathbf{g}}(\boldsymbol{\theta}_{t-\tau_{mix}})\rangle\right] \\
&= \mathbb{E}\left[\langle \boldsymbol{\theta}_{t-\tau_{mix}} - \boldsymbol{\theta}^*, \mathbb{E}\left[\mathbf{g}(\boldsymbol{\theta}_{t-\tau_{mix}}, o_t)|o_{t-\tau}, \boldsymbol{\theta}_{t-\tau_{mix}}\right] - \bar{\mathbf{g}}(\boldsymbol{\theta}_{t-\tau_{mix}})\rangle\right] \\
&\leq \mathbb{E}\left[\|\boldsymbol{\theta}_{t-\tau_{mix}} - \boldsymbol{\theta}^*\| \|\mathbb{E}\left[\mathbf{g}(\boldsymbol{\theta}_{t-\tau_{mix}}, o_t)|o_{t-\tau_{mix}}, \boldsymbol{\theta}_{t-\tau_{mix}}\right] - \bar{\mathbf{g}}(\boldsymbol{\theta}_{t-\tau_{mix}})\|\right] \\
&\overset{(*)}{\leq} \alpha\mathbb{E}\left[\|\boldsymbol{\theta}_{t-\tau_{mix}} - \boldsymbol{\theta}^*\|(\|\boldsymbol{\theta}_{t-\tau_{mix}}\| + \sigma)\right] \\
&\leq \alpha\mathbb{E}\left[\|\boldsymbol{\theta}_{t-\tau_{mix}} - \boldsymbol{\theta}^*\|(\|\boldsymbol{\theta}_{t-\tau_{mix}} - \boldsymbol{\theta}^*\| + 2\sigma)\right] \\
&\leq \alpha\mathbb{E}\left[\frac{1}{2}\left(r^2_{t-\tau_{mix}} + 2r^2_{t-\tau_{mix}} + 4\sigma^2\right)\right] \\
&\leq 2\alpha(r_{t,2} + \sigma^2),
\end{aligned} \tag{C.102}$$

where for $(*)$ we used Definition 5.2.1 of mixing time and the fact that $\sigma \geq 1$. So, putting the above bounds together, we get

$$\mathbb{E}\left[T'_{12}\right] = \mathbb{E}\left[\bar{T}_1\right] + \mathbb{E}\left[\bar{T}_2\right] + \mathbb{E}\left[\bar{T}_3\right] \leq 8\alpha\tau_{mix}L^2(r^2_{t,2} + \sigma^2). \tag{C.103}$$

So, we get

$$\begin{aligned}
\mathbb{E}\left[T'_1\right] &= \mathbb{E}\left[T'_{11}\right] + \mathbb{E}\left[T'_{12}\right] \\
&\leq 4\alpha\tau_{mix}L^2 r_{t,2} + 6\alpha\tau_{mix}L^2\sigma^2 + 8\alpha\tau_{mix}L^2(r_{t,2} + \sigma^2) \\
&\leq 12\alpha\tau_{mix}L^2 r_{t,2} + 14\alpha\tau_{mix}L^2\sigma^2
\end{aligned} \tag{C.104}$$

so,

$$\mathbb{E}\left[J_{t,11}\right] \leq -\mu\mathbb{E}\left[r_t^2\right] + \mathbb{E}\left[T'_1\right]. \tag{C.105}$$

Hence,

$$\begin{aligned}
\mathbb{E}\left[J_{t,1}\right] &= \mathbb{E}\left[r_t^2\right] + 2\alpha\mathbb{E}\left[J_{t,11}\right] + \alpha^2\mathbb{E}\left[J_{t,12}\right] \\
&\leq (1 - 2\alpha\mu)\mathbb{E}\left[r_t^2\right] + 28\alpha^2\tau_{mix}L^2 r_{t,2} + 34\alpha^2\tau_{mix}L^2\sigma^2,
\end{aligned} \tag{C.106}$$

which concludes the proof of the Lemma. □

Proof of Lemma 5.5 - (ii). Note that

$$\begin{aligned} J_{t,2} = \|\mathbf{e}_t\|^2 &= \|\mathbf{g}(\boldsymbol{\theta}_t, o_t) - \mathbf{g}(\boldsymbol{\theta}_{t-\tau_t}, o_{t-\tau_t})\|^2 \\ &\overset{(C.8)}{\leq} 2\left(\|\mathbf{g}(\boldsymbol{\theta}_t, o_t)\|^2 + \|\mathbf{g}(\boldsymbol{\theta}_{t-\tau_t}, o_{t-\tau_t})\|^2\right) \\ &\overset{(C.5)}{\leq} 2\left(2L^2(\|\boldsymbol{\theta}_t\|^2 + \sigma^2) + 2L^2(\|\boldsymbol{\theta}_{t-\tau_t}\|^2 + \sigma^2)\right) \\ &\leq 4L^2(2r_t^2 + 3\sigma^2 + 2r_{t-\tau_t} + 3\sigma^2). \end{aligned} \tag{C.107}$$

Taking the expectation, we conclude getting

$$\mathbb{E}\left[J_{t,2}\right] = \mathbb{E}\left[\|\mathbf{e}_t\|^2\right] \leq 8L^2(2r_{t,2} + 3\sigma^2) \tag{C.108}$$

Proof of Lemma 5.5 - (iii). In the following, we drop the dependence on the iteration t in the terms we bound. We write

$$\begin{aligned} J_{t,3} &= \langle \mathbf{e}_t, \boldsymbol{\theta}_k - \boldsymbol{\theta}^* + \alpha \mathbf{g}(\boldsymbol{\theta}_k, o_t)\rangle \\ &= \underbrace{\langle \mathbf{e}_t, \boldsymbol{\theta}_k - \boldsymbol{\theta}^*\rangle}_{\Delta} + \underbrace{\alpha\langle \mathbf{e}_t, \mathbf{g}(\boldsymbol{\theta}_k, o_t)\rangle}_{\bar{\Delta}}. \end{aligned} \tag{C.109}$$

Note that

$$\begin{aligned} \bar{\Delta} = \alpha\langle \mathbf{e}_t, \mathbf{g}(\boldsymbol{\theta}_k, o_t)\rangle &\leq \alpha\|\mathbf{e}_t\|\|\mathbf{g}(\boldsymbol{\theta}_k, o_t)\| \\ &\leq \frac{\alpha}{2}\left(\|\mathbf{e}_t\|^2 + \|\mathbf{g}(\boldsymbol{\theta}_t, o_t)\|^2\right). \end{aligned} \tag{C.110}$$

Using (C.108) and (C.5) to bound $\mathbb{E}\left[\|\mathbf{e}_t\|^2\right]$ and $\mathbb{E}\left[\|\mathbf{g}(\boldsymbol{\theta}_t, o_t)\|^2\right]$, respectively, we get

$$\begin{aligned} \mathbb{E}\left[\bar{\Delta}\right] &\leq \frac{\alpha}{2}\left(8L^2\left(2r_{t,2} + 3\sigma^2\right) + 2L^2\left(2r_{t,2} + 3\sigma^2\right)\right) \\ &= 10\alpha L^2 r_{t,2} + 15\alpha L^2\sigma^2. \end{aligned} \tag{C.111}$$

We now proceed to bound Δ.

$$\begin{aligned} \Delta = \langle \mathbf{e}_t, \boldsymbol{\theta}_k - \boldsymbol{\theta}^*\rangle &= \langle \mathbf{g}(\boldsymbol{\theta}_k, o_t) - \mathbf{g}(\boldsymbol{\theta}_{t-\tau_t}, o_{t-\tau_t}), \boldsymbol{\theta}_k - \boldsymbol{\theta}^*\rangle \\ &= \underbrace{\langle \mathbf{g}(\boldsymbol{\theta}_k, o_t) - \mathbf{g}(\boldsymbol{\theta}_k, o_{t-\tau_t}), \boldsymbol{\theta}_k - \boldsymbol{\theta}^*\rangle}_{\Delta_1} \\ &+ \underbrace{\langle \mathbf{g}(\boldsymbol{\theta}_k, o_{t-\tau_t}) - \mathbf{g}(\boldsymbol{\theta}_{t-\tau_t}, o_{t-\tau_t}), \boldsymbol{\theta}_k - \boldsymbol{\theta}^*\rangle}_{\Delta_2} \end{aligned} \tag{C.112}$$

Note that, thanks to the Lipschitz property of the TD direction and with calculations analogous to the ones performed to obtain the bound on $\mathbb{E}\left[\bar{T}_2\right]$ (see (C.99) and (C.100)),

we get

$$\mathbb{E}\left[\Delta_2\right] \leq \mathbb{E}\left[L\|\boldsymbol{\theta}_k - \boldsymbol{\theta}_{t-\tau_t}\| r_t\right] \leq 3\alpha\tau_{mix} L^2 (r_{t,2} + \sigma^2). \tag{C.113}$$

We now bound Δ_1.

$$\begin{aligned}\Delta_1 = &\underbrace{\langle \mathbf{g}(\boldsymbol{\theta}_k, o_t) - \mathbf{g}(\boldsymbol{\theta}_{t-\tau_{mix}}, o_t), \boldsymbol{\theta}_k, -\boldsymbol{\theta}^* \rangle}_{\Delta_{11}} \\ &+ \underbrace{\langle \mathbf{g}(\boldsymbol{\theta}_{t-\tau_{mix}}, o_t) - \mathbf{g}(\boldsymbol{\theta}_k, o_{t-\tau_t}), \boldsymbol{\theta}_k - \boldsymbol{\theta}^* \rangle}_{\Delta_{12}}.\end{aligned} \tag{C.114}$$

With calculations analogous to the ones performed to obtain the bound on $\mathbb{E}\left[\bar{T}_2\right]$ (see (C.99) and (C.100)), we get

$$\mathbb{E}\left[\Delta_{11}\right] \leq \mathbb{E}\left[L\|\boldsymbol{\theta}_k - \boldsymbol{\theta}_{t-\tau_t}\| r_t\right] \leq 3\alpha\tau_{mix} L^2 (r_{t,2} + \sigma^2). \tag{C.115}$$

We now proceed to bound Δ_{12}.

$$\begin{aligned}\Delta_{12} = &\underbrace{\langle \mathbf{g}(\boldsymbol{\theta}_{t-\tau_{mix}}, o_t) - \bar{\mathbf{g}}(\boldsymbol{\theta}_{t-\tau_{mix}}), \boldsymbol{\theta}_k - \boldsymbol{\theta}^* \rangle}_{\Delta_1'} \\ &+ \underbrace{\langle \bar{\mathbf{g}}(\boldsymbol{\theta}_{t-\tau_{mix}}) - \mathbf{g}(\boldsymbol{\theta}_k, o_{t-\tau_t}), \boldsymbol{\theta}_k - \boldsymbol{\theta}^* \rangle}_{\Delta_2'}\end{aligned} \tag{C.116}$$

We have

$$\begin{aligned}\Delta_1' = &\underbrace{\langle \mathbf{g}(\boldsymbol{\theta}_{t-\tau_{mix}}, o_t) - \bar{\mathbf{g}}(\boldsymbol{\theta}_{t-\tau_{mix}}), \boldsymbol{\theta}_{t-\tau_{mix}} - \boldsymbol{\theta}^* \rangle}_{\Delta_{11}'} \\ &+ \underbrace{\langle \mathbf{g}(\boldsymbol{\theta}_{t-\tau_{mix}}, o_t) - \bar{\mathbf{g}}(\boldsymbol{\theta}_{t-\tau_{mix}}), \boldsymbol{\theta}_k - \boldsymbol{\theta}_{t-\tau_{mix}} \rangle}_{\Delta_{12}'}\end{aligned} \tag{C.117}$$

Note that

$$\begin{aligned}\Delta_{12}' &\leq \|\mathbf{g}(\boldsymbol{\theta}_{t-\tau_{mix}}, o_t) - \bar{\mathbf{g}}(\boldsymbol{\theta}_{t-\tau_{mix}})\| \|\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t-\tau_{mix}}\| \\ &\leq \left(\|\mathbf{g}(\boldsymbol{\theta}_{t-\tau_{mix}}, o_t)\| + \|\bar{\mathbf{g}}(\boldsymbol{\theta}_{t-\tau_{mix}})\|\right) \|\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t-\tau_{mix}}\| \\ &\overset{(C.4)}{\leq} 2L \left(\|\boldsymbol{\theta}_{t-\tau_{mix}}\| + \sigma\right) \|\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t-\tau_{mix}}\| \\ &\leq 2L \left(r_{t-\tau_{mix}}^2 + 2\sigma\right) \|\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t-\tau_{mix}}\| \\ &\leq 2\alpha L^2 \tau_{mix} (r_{t-\tau_{mix}}^2 + 2\sigma^2) + \frac{1}{2\alpha\tau_{mix}} \|\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t-\tau_{mix}}\|^2.\end{aligned} \tag{C.118}$$

Taking expectation on both sides and applying Lemma 5.4, we get

$$\begin{aligned}\mathbb{E}\left[\Delta'_{12}\right] &\leq 2\alpha\tau_{mix}L^2\left(\mathbb{E}\left[r^2_{t-\tau_{mix}}\right]+2\sigma^2\right)+\frac{1}{2\alpha\tau_{mix}}\mathbb{E}\left[\|\boldsymbol{\theta}_t-\boldsymbol{\theta}_{t-\tau_{mix}}\|^2\right]\\ &\leq 2\alpha\tau_{mix}L^2\left(r_{t,2}+2\sigma^2\right)+\alpha\tau_{mix}L^2\left(2r_{t,2}+3\sigma^2\right)\\ &= 4\alpha\tau_{mix}L^2r_{t,2}+7\alpha\tau_{mix}L^2\sigma^2.\end{aligned} \tag{C.119}$$

Now note that Δ'_{11} can be bounded using the same calculations used for $\bar{T}_1$ in (C.102),

$$\mathbb{E}\left[\Delta'_{11}\right]\leq 2\alpha(r_{t,2}+\sigma^2). \tag{C.120}$$

Now note that

$$\begin{aligned}\Delta'_2 &= \langle\bar{\mathbf{g}}(\boldsymbol{\theta}_{t-\tau_{mix}})-\mathbf{g}(\boldsymbol{\theta}_k,o_{t-\tau_t}),\boldsymbol{\theta}_k-\boldsymbol{\theta}^*\rangle\\ &= \underbrace{\langle\bar{\mathbf{g}}(\boldsymbol{\theta}_{t-\tau_{mix}})-\mathbf{g}(\boldsymbol{\theta}_{t-\tau_{mix}},o_{t-\tau_t}),\boldsymbol{\theta}_k-\boldsymbol{\theta}^*\rangle}_{\Delta'_{21}}\\ &+\underbrace{\langle\mathbf{g}(\boldsymbol{\theta}_{t-\tau_{mix}},o_{t-\tau_t})-\mathbf{g}(\boldsymbol{\theta}_k,o_{t-\tau_t}),\boldsymbol{\theta}_k-\boldsymbol{\theta}^*\rangle}_{\Delta'_{22}}\end{aligned} \tag{C.121}$$

To bound $\mathbb{E}\left[\Delta'_{22}\right]$, we can proceed with calculations analogous to the ones performed to obtain the bound on $\mathbb{E}\left[\bar{T}_2\right]$ (see (C.99) and (C.100)), getting

$$\mathbb{E}\left[\Delta'_{22}\right]\leq 3\alpha\tau_{mix}L^2(r_{t,2}+\sigma^2). \tag{C.122}$$

Now, we write

$$\begin{aligned}\Delta'_{21} &= \underbrace{\langle\bar{\mathbf{g}}(\boldsymbol{\theta}_{t-\tau_{mix}})-\bar{\mathbf{g}}(\boldsymbol{\theta}_{t-\tau_{mix}-\tau_t}),\boldsymbol{\theta}_k-\boldsymbol{\theta}^*\rangle}_{\bar{\Delta}_1}\\ &+\underbrace{\langle\bar{\mathbf{g}}(\boldsymbol{\theta}_{t-\tau_{mix}-\tau_t})-\mathbf{g}(\boldsymbol{\theta}_{t-\tau_{mix}},o_{t-\tau_t}),\boldsymbol{\theta}_k-\boldsymbol{\theta}^*\rangle}_{\bar{\Delta}_2}.\end{aligned} \tag{C.123}$$

We see that, as before, we can bound $\mathbb{E}\left[\bar{\Delta}_1\right]$ with the same procedure we used to bound $\mathbb{E}\left[\Delta'_{22}\right]$:

$$\mathbb{E}\left[\bar{\Delta}_1\right]\leq 3\alpha\tau_{mix}L^2(r_{t,2}+\sigma^2). \tag{C.124}$$

We write

$$\begin{aligned}\bar{\Delta}_2 &= \underbrace{\langle\bar{\mathbf{g}}(\boldsymbol{\theta}_{t-\tau_{mix}-\tau_t})-\mathbf{g}(\boldsymbol{\theta}_{t-\tau_{mix}-\tau_t},o_{t-\tau_t}),\boldsymbol{\theta}_k-\boldsymbol{\theta}^*\rangle}_{\bar{\Delta}_{21}}\\ &+\underbrace{\langle\mathbf{g}(\boldsymbol{\theta}_{t-\tau_{mix}-\tau_t},o_{t-\tau_t})-\mathbf{g}(\boldsymbol{\theta}_{t-\tau_{mix}},o_{t-\tau_t}),\boldsymbol{\theta}_k-\boldsymbol{\theta}^*\rangle}_{\bar{\Delta}_{22}}.\end{aligned} \tag{C.125}$$

Now note that $\mathbb{E}\left[\bar{\Delta}_{22}\right]$ can be bounded with calculations analogous to the ones performed to obtain the bound on $\bar{\Delta}_1$,

$$\begin{aligned}\mathbb{E}\left[\bar{\Delta}_{22}\right] &\leq \mathbb{E}\left[L(\|\boldsymbol{\theta}_{t-\tau_{mix}-\tau_t} - \boldsymbol{\theta}_{t-\tau_{mix}}\|\|\boldsymbol{\theta}_k - \boldsymbol{\theta}^*\|)\right] \\ &\leq 3\alpha\tau_{mix}L^2(r_{t,2} + \sigma^2)\end{aligned} \tag{C.126}$$

Now note that

$$\begin{aligned}\bar{\Delta}_{21} = &\underbrace{\langle \bar{\mathbf{g}}(\boldsymbol{\theta}_{t-\tau_{mix}-\tau_t}) - \mathbf{g}(\boldsymbol{\theta}_{t-\tau_{mix}-\tau_t}, o_{t-\tau_t}), \boldsymbol{\theta}_{t-\tau_{mix}-\tau_t} - \boldsymbol{\theta}^*\rangle}_{\bar{\Delta}_{211}} \\ &+ \underbrace{\langle \bar{\mathbf{g}}(\boldsymbol{\theta}_{t-\tau_{mix}-\tau_t}) - \mathbf{g}(\boldsymbol{\theta}_{t-\tau_{mix}-\tau_t}, o_{t-\tau_t}), \boldsymbol{\theta}_k - \boldsymbol{\theta}_{t-\tau_{mix}-\tau_t}\rangle}_{\bar{\Delta}_{212}}.\end{aligned} \tag{C.127}$$

With calculations analogous to the ones performed to obtain the bound on $\mathbb{E}\left[\Delta'_{12}\right]$ (see (C.119)), we get

$$\begin{aligned}\mathbb{E}\left[\bar{\Delta}_{212}\right] &\leq \mathbb{E}\left[\|\boldsymbol{\theta}_k - \boldsymbol{\theta}_{t-\tau_{mix}-\tau_t}\|L\left(\|\bar{\mathbf{g}}(\boldsymbol{\theta}_{t-\tau_{mix}-\tau_t})\| + \|\mathbf{g}(\boldsymbol{\theta}_{t-\tau_{mix}-\tau_t}, o_{t-\tau_t})\|\right)\right] \\ &\leq 4\alpha(\tau_{mix} + \tau_{max})L^2 r_{t,2} + 7\alpha(\tau_{mix} + \tau_{max})L^2\sigma^2.\end{aligned} \tag{C.128}$$

Now note that $\bar{\Delta}_{211}$ can be bounded using the same calculations used for $\bar{T}_1$ in (C.102), getting

$$\mathbb{E}\left[\bar{\Delta}_{211}\right] \leq 2\alpha(r_{t,2} + \sigma^2). \tag{C.129}$$

So, $\mathbb{E}\left[T_3\right]$ can be upper bounded by a sum of terms that are upper bounded by either $O(\alpha)(r_{t,2}^2 + \sigma^2)$, $O(\alpha\tau_{max})(r_{t,2}^2 + \sigma^2)$, $O(\alpha\tau_{mix})(r_{t,2}^2 + \sigma^2)$ or $O(\alpha)(\tau_{mix} + \tau_{max})(r_{t,2}^2 + \sigma^2)$. Putting all the terms together, we can get

$$\mathbb{E}\left[J_{t,3}\right] \leq 28\alpha L^2(\tau_{mix} + \tau_{max})(r_{t,2} + \sigma^2), \tag{C.130}$$

which concludes the proof. □

Now, recall the definition of the update rule for delayed SA with time-varying delay under Assumption 10:

$$\textit{Delayed SA:} \qquad \boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k + \alpha\mathbf{g}(\boldsymbol{\theta}_{t-\tau_t}, o_{t-\tau_t}), \quad \tau_t \leq \min\{t, \tau_{max}\}. \tag{C.131}$$

Consider the mean squared error term $\mathbb{E}\left[r_t^2\right] = \mathbb{E}\left[\|\boldsymbol{\theta}_t - \boldsymbol{\theta}^*\|^2\right]$, and its expression derived in (C.88). The bounds on $\mathbb{E}\left[J_{t,1}\right]$, $\mathbb{E}\left[J_{t,2}\right]$ and $\mathbb{E}\left[J_{t,3}\right]$ provided in the previous section, defining $\tau' = 2\tau_{max} + \tau_{mix}$, for $t \geq \tau'$, are such that the update rule (C.131) satisfies the

following:

$$\begin{aligned}\mathbb{E}\left[r_{t+1}^2\right] &= \mathbb{E}\left[\|\boldsymbol{\theta}_{k+1} - \boldsymbol{\theta}^*\|^2\right] \\ &= \mathbb{E}\left[J_{t,1}\right] + \alpha^2\mathbb{E}\left[J_{t,2}\right] - 2\alpha\mathbb{E}\left[J_{t,3}\right] \\ &\leq (1 - 2\alpha\mu)\mathbb{E}\left[r_t^2\right] + 98\alpha^2L^2(\tau_{mix} + \tau_{max})(r_{t,2} + \sigma^2),\end{aligned} \tag{C.132}$$

with

$$r_{t,2} = \max_{t-\tau' \leq l \leq t} \mathbb{E}\left[r_l^2\right].$$

As mentioned in the outline of the analysis in Section 5.4, the final part of the proof of Theorem 5.4 is based on a crucial argument that shows that, for a sufficiently small step size, the iterates generated by (C.131) are uniformly bounded, which is shown in Lemma 5.6. To prove the result, we first provide the following Lemma, which proves the base case of the induction proof on which the proof of Lemma 5.6 relies on.

Lemma C.2. Consider the update rule in (C.131) and let $B = 9\sigma^2$. For $0 \leq t \leq \tau'$ and $\alpha \leq \frac{1}{24L^2\tau'}$, we have

$$\mathbb{E}\left[r_t^2\right] \leq B, \quad 0 \leq t \leq \tau' \tag{C.133}$$

Proof. Note that

$$\begin{aligned}r_{t+1}^2 &= r_t^2 + 2\alpha\langle\boldsymbol{\theta}_t - \boldsymbol{\theta}^*, \mathbf{g}(\boldsymbol{\theta}_{t-\tau_t}, o_{t-\tau_t})\rangle + \alpha^2\|\mathbf{g}(\boldsymbol{\theta}_{t-\tau_t}, o_{t-\tau_t})\|^2 \\ &\overset{(C.7)}{\leq} r_t^2 + \alpha r_t^2 + \alpha\|\mathbf{g}(\boldsymbol{\theta}_{t-\tau_t}, o_{t-\tau_t})\|^2 + \alpha^2\|\mathbf{g}(\boldsymbol{\theta}_{t-\tau_t}, o_{t-\tau_t})\|^2 \\ &\leq (1+\alpha)r_t^2 + 2\alpha\|\mathbf{g}(\boldsymbol{\theta}_{t-\tau_t}, o_{t-\tau_t})\|^2 \\ &\overset{(C.5)}{\leq} (1+\alpha)r_t^2 + 4\alpha L^2(\|\boldsymbol{\theta}_{t-\tau_t}\|^2 + \sigma^2) \\ &\leq (1+\alpha)r_t^2 + 4\alpha L^2(2r_{t-\tau_t}^2 + 3\sigma^2) \\ &= (1+\alpha)r_t^2 + 8\alpha L^2 r_{t-\tau_t}^2 + 12\alpha L^2\sigma^2.\end{aligned} \tag{C.134}$$

Taking the expectation on both sides, we get

$$\mathbb{E}\left[r_{t+1}^2\right] \leq (1+\alpha)\mathbb{E}\left[r_t^2\right] + 8\alpha L^2\mathbb{E}\left[r_{t-\tau_t}^2\right] + 12\alpha L^2\sigma^2. \tag{C.135}$$

Hence, we get an inequality of the following form:

$$V_{t+1} \leq pV_t + qV_{t-\tau_t} + \beta, \quad 0 \leq \tau_t \leq \min\{t, \tau_{max}\}, \tag{C.136}$$

with $V_t = \mathbb{E}\left[r_t^2\right]$, $p = 1 + \alpha$, $q = 8\alpha L^2\tau_{max}$, and $\beta = 12\alpha L^2\sigma^2$. We define $\rho = p + q$,

noting that $\rho > 1$. We now prove by induction that, for all $t \geq 0$,

$$V_t \leq \rho^t V_0 + \epsilon_t, \tag{C.137}$$

where

$$\epsilon_t = \begin{cases} \rho\epsilon_{t-1} + \beta \text{ for } t \geq 1 \\ 0 \text{ for } t = 0 \end{cases} \tag{C.138}$$

The base case is trivially satisfied, because $V_0 \leq V_0$. As an inductive step, suppose that (C.137) is true for $0 \leq s \leq k$, for some $k \geq 0$, so

$$V_s \leq \rho^s V_0 + \epsilon_s, \quad 0 \leq s \leq k. \tag{C.139}$$

Now, we check the property for $k+1$, using (C.136), and noting that ϵ_k is an increasing sequence

$$\begin{aligned} V_{k+1} &\leq pV_k + qV_{k-\tau_k} + \beta, \\ &\leq p(\rho^k V_0 + \epsilon_k) + q(\rho^{k-\tau_k} V_0 + \epsilon_{k-\tau_k}) + \beta \\ &\leq p(\rho^k V_0 + \epsilon_k) + q(\rho^k V_0 + \epsilon_k) + \beta \\ &\leq (p+q)\rho^k V_0 + (p+q)\epsilon_k + \beta \\ &= \rho^{k+1} V_0 + \epsilon_{k+1}. \end{aligned} \tag{C.140}$$

From which we can conclude the proof of (C.137). Now, note that

$$\epsilon_t = \beta \sum_{j=0}^{t-1} \rho^j, \tag{C.141}$$

and so we can write, for $0 \leq t \leq \tau'$,

$$\rho^t \leq \rho^{\tau'} \leq (1+\alpha)^{\tau'} \leq e^{\alpha\tau'} \leq e^{0.25} \leq 2, \tag{C.142}$$

using the fact that $\alpha \leq \frac{1}{4\tau'}$. Hence, we can get, for $0 \leq t \leq \tau'$, using the above results,

$$\begin{aligned} \mathbb{E}\left[r_t^2\right] \leq \rho^t r_0^2 + \epsilon_t \leq 2r_0^2 + \beta \sum_{j=0}^{\tau'-1} \rho^j \leq 2r_0^2 + 2\beta\tau' = 2r_0^2 + 2(12\alpha L^2\sigma^2)\tau' \\ \leq 2r_0^2 + \sigma^2 \leq 9\sigma^2, \end{aligned} \tag{C.143}$$

where we used the fact that $\alpha \leq \frac{1}{24L^2\tau'}$ and that $r_0^2 = \|\boldsymbol{\theta}_0 - \boldsymbol{\theta}^*\|^2 \leq 2\|\boldsymbol{\theta}_0\|^2 + 2\|\boldsymbol{\theta}^*\|^2 \leq 4\sigma^2$.

We have just shown that, for $0 \leq t \leq \tau'$, it holds

$$\mathbb{E}\left[r_t^2\right] \leq 9\sigma^2 \tag{C.144}$$

□

Now, we provide the proof of Lemma 5.6, which relies on Lemma C.2.

Proof of Lemma 5.6. We know from Lemma C.2 that for $t = 0, ..., \tau'$, with $\tau' = 2\tau_{max} + \tau_{mix}$, and $\alpha \leq \frac{1}{24L^2\tau'}$, we have

$$\mathbb{E}\left[r_t^2\right] \leq B. \tag{C.145}$$

We now proceed by induction to show that the bound holds true also for any $t \geq \tau'$, thus for all $t \geq 0$. We use (C.145) as the base case for the induction proof. As an induction step, assume that the property is true for some $t \geq \tau'$ and for $\tau' \leq s \leq t$, so

$$\mathbb{E}\left[r_s^2\right] \leq B \quad \forall \tau' \leq s \leq t. \tag{C.146}$$

Now, from (C.132) we can write

$$\mathbb{E}\left[r_{t+1}^2\right] \leq (1 - 2\alpha\mu)\mathbb{E}\left[r_t^2\right] + 98\alpha^2L^2(\tau_{mix} + \tau_{max})(r_{t,2} + \sigma^2). \tag{C.147}$$

Now note that by the inductive step and induction base case, it holds

$$r_{t,2} = \max_{t-\tau' \leq l \leq t} \mathbb{E}\left[r_l^2\right] \leq B. \tag{C.148}$$

Hence, we can write, recalling that $B = 9\sigma^2 \geq \sigma^2$,

$$\begin{aligned}
\mathbb{E}\left[r_{t+1}^2\right] &\leq (1 - 2\alpha\mu)\mathbb{E}\left[r_t^2\right] + 98\alpha^2L^2(\tau_{mix} + \tau_{max})(r_{t,2} + \sigma^2) \\
&\leq (1 - 2\alpha\mu)B + 98\alpha^2L^2(\tau_{mix} + \tau_{max})(B + \sigma^2) \\
&\leq (1 - 2\alpha\mu)B + 2B98\alpha^2L^2(\tau_{mix} + \tau_{max}) \\
&\leq (1 - 2\alpha\mu + 196\alpha^2L^2(\tau_{mix} + \tau_{max}))B,
\end{aligned} \tag{C.149}$$

and so for $\alpha \leq \frac{\mu}{196L^2\bar{\tau}} \leq \frac{\mu}{98L^2\tau'} \leq \frac{\mu}{98L^2(\tau_{mix}+\tau_{max})}$, we get $1-2\alpha\mu+196\alpha^2L^2(\tau_{mix}+\tau_{max}) \leq 1$ and thus

$$\mathbb{E}\left[r_{t+1}^2\right] \leq B, \tag{C.150}$$

implying that the absolute constant is $C = 196$, from which we can conclude the proof.

□

Using this last Lemma in conjunction with (C.132), we can get the following proof of Theorem 5.4.

C.2.2 Conclusion of the Proof

Note that, from (C.132), we can write, for $T \geq \tau' = 2\tau_{max} + \tau_{mix}$,

$$\begin{aligned}\mathbb{E}\left[r_{t+1}^2\right] &\leq (1-2\alpha\mu)\mathbb{E}\left[r_t^2\right] + 98\alpha^2L^2(\tau_{mix}+\tau_{max})(r_{t,2}+\sigma^2) \\ &\overset{(*)}{\leq} (1-2\alpha\mu)\mathbb{E}\left[r_t^2\right] + 2B98\alpha^2L^2(\tau_{mix}+\tau_{max}),\end{aligned} \tag{C.151}$$

where for $(*)$ we used the fact that, for $\alpha \leq \frac{1}{196L^2\bar{\tau}}$, it holds $r_{t,2} \leq B = 9\sigma^2$, as established by Lemma 5.6. Iterating the inequality, we get

$$\begin{aligned}\mathbb{E}\left[r_{t+1}^2\right] &\leq (1-2\alpha\mu)^{t+1-\tau'}r_{\tau'}^2 + 298L^2\alpha^2(\tau_{mix}+\tau_{max})B\sum_{j=0}^{\infty}(1-2\alpha\mu)^j \\ &\leq (1-2\alpha\mu)^{t+1-\tau'}r_{\tau'}^2 + \frac{98L^2\alpha(\tau_{mix}+\tau_{max})B}{\mu} \\ &\leq (1-2\alpha\mu)^{t+1-\tau'}B + \frac{98L^2\alpha(\tau_{mix}+\tau_{max})B}{\mu} \\ &\leq (1-2\alpha\mu)^{t+1}2B + \frac{98L^2\alpha(\tau_{mix}+\tau_{max})B}{\mu} \\ &\leq e^{-2\alpha\mu(t+1)}2B + \frac{98L^2\alpha(\tau_{mix}+\tau_{max})B}{\mu},\end{aligned} \tag{C.152}$$

where for the last inequality we used the fact that

$$(1-2\alpha\mu)^{-\tau'} \leq e^{2\alpha\mu\tau'} \leq e^{0.25} \leq 2, \tag{C.153}$$

where the inequality follows because $\alpha\mu \leq \alpha \leq \frac{1}{196L^2\bar{\tau}} \leq \frac{1}{8\tau'}$. Hence, for $\alpha \leq \frac{1}{CL^2\bar{\tau}}$, with $C = 196$, we get the result. Setting $\alpha = \frac{1}{CL^2\bar{\tau}}$, with $C \geq 196$ and $C' = 98$, we can also get

$$\mathbb{E}\left[r_T^2\right] \leq \exp\left(-\frac{2\mu^2T}{CL^2\bar{\tau}}\right)2B + \frac{C'B}{C}. \tag{C.154}$$

□

References

[1] D. A. E. Acar, Y. Zhao, R. M. Navarro, M. Mattina, P. N. Whatmough, and V. Saligrama, "Federated learning based on dynamic regularization," *arXiv preprint arXiv:2111.04263*, 2021 (Cited in page 59).

[2] A. Agafonov, D. Kamzolov, R. Tappenden, A. Gasnikov, and M. Takáč, "Flecs: A federated learning second-order framework via compression and sketching," *arXiv preprint arXiv:2206.02009*, 2022 (Cited in pages 16, 93).

[3] A. Agarwal and J. C. Duchi, "Distributed delayed stochastic optimization," *Advances in neural information processing systems*, vol. 24, 2011 (Cited in page 79).

[4] F. Alimisis, P. Davies, and D. Alistarh, "Communication-efficient distributed optimization with quantized preconditioners," in *Proceedings of the 38th International Conference on Machine Learning*, 2021 (Cited in page 44).

[5] ——, "Communication-efficient distributed optimization with quantized preconditioners," in *Proceedings of the 38th International Conference on Machine Learning*, M. Meila and T. Zhang, Eds., ser. Proceedings of Machine Learning Research, vol. 139, PMLR, 2021, pp. 196–206 (Cited in page 16).

[6] D. Alistarh, D. Grubic, J. Li, R. Tomioka, and M. Vojnovic, "Qsgd: Communication-efficient sgd via gradient quantization and encoding," *Advances in neural information processing systems*, vol. 30, 2017 (Cited in page 5).

[7] M. M. Amiri and D. Gündüz, "Federated learning over wireless fading channels," *IEEE Transactions on Wireless Communications*, vol. 19, no. 5, pp. 3546–3557, 2020 (Cited in pages 61, 66).

[8] ——, "Machine learning at the wireless edge: Distributed stochastic gradient descent over-the-air," *IEEE Transactions on Signal Processing*, vol. 68, pp. 2155–2169, 2020 (Cited in page 61).

[9] M. M. Amiri and D. Gündüz, "Federated learning over wireless fading channels," *IEEE Transactions on Wireless Communications*, vol. 19, no. 5, pp. 3546–3557, 2020 (Cited in page 16).

[10] Y. Arjevani, O. Shamir, and N. Srebro, "A tight convergence analysis for stochastic gradient descent with delayed updates," in *Algorithmic Learning Theory*, PMLR, 2020, pp. 111–132 (Cited in pages 83, 87, 88).

[11] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *International conference on machine learning*, PMLR, 2017, pp. 214–223 (Cited in page 89).

[12] M. Assran, A. Aytekin, H. R. Feyzmahdavian, M. Johansson, and M. G. Rabbat, "Advances in asynchronous parallel and distributed optimization," *Proceedings of the IEEE*, vol. 108, no. 11, pp. 2013–2031, 2020 (Cited in page 87).

[13] C. Battiloro, P. Di Lorenzo, M. Merluzzi, and S. Barbarossa, "Lyapunov-based optimization of edge resources for energy-efficient adaptive federated learning," *IEEE Transactions on Green Communications and Networking*, 2022 (Cited in page 14).

[14] C. Battiloro, P. D. Lorenzo, M. Merluzzi, and S. Barbarossa, "Lyapunov-based optimization of edge resources for energy-efficient adaptive federated learning," *IEEE Transactions on Green Communications and Networking*, 2022 (Cited in page 44).

[15] A. Beck, *Introduction to nonlinear optimization: Theory, algorithms, and applications with MATLAB*. SIAM, 2014 (Cited in page 6).

[16] T. Bertin-Mahieux, D. P. Ellis, B. Whitman, and P. Lamere, "The million song dataset," in *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR 2011)*, 2011 (Cited in page 34).

[17] D. P. Bertsekas and J. N. Tsitsiklis, "Convergence rate and termination of asynchronous iterative algorithms," in *Proceedings of the 3rd International Conference on Supercomputing*, 1989, pp. 461–470 (Cited in page 79).

[18] A. Beznosikov, S. Horváth, P. Richtárik, and M. Safaryan, "On biased compression for distributed learning," *arXiv:2002.12410*, 2020 (Cited in page 65).

[19] J. Bhandari, D. Russo, and R. Singal, "A finite time analysis of temporal difference learning with linear function approximation," in *Conference on learning theory*, PMLR, 2018, pp. 1691–1692 (Cited in pages 60, 61, 63–65, 70, 73, 79–81, 83, 109–111, 120, 122, 124, 132).

[20] ——, "A finite time analysis of temporal difference learning with linear function approximation," in *Conference on learning theory*, PMLR, 2018, pp. 1691–1692 (Cited in page 78).

[21] J. Bisgard, *Analysis and Linear Algebra: The Singular Value Decomposition and Applications*. American Mathematical Soc., 2020, vol. 94 (Cited in page 104).

[22] K. Bonawitz, H. Eichner, W. Grieskamp, D. Huba, A. Ingerman, V. Ivanov, C. Kiddon, J. Konečnỳ, S. Mazzocchi, H. B. McMahan, *et al.*, "Towards federated learning at scale: System design," *arXiv preprint arXiv:1902.01046*, 2019 (Cited in page 59).

[23] V. S. Borkar and S. P. Meyn, "The ode method for convergence of stochastic approximation and reinforcement learning," *SIAM Journal on Control and Optimization*, vol. 38, no. 2, pp. 447–469, 2000 (Cited in page 60).

[24] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004 (Cited in pages 6, 99–101).

[25] X. Cao, G. Zhu, J. Xu, Z. Wang, and S. Cui, "Optimized power control design for over-the-air federated edge learning," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 1, pp. 342–358, 2021 (Cited in pages 61, 66, 67).

[26] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, 27:1–27:27, 3 2011, Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm (Cited in pages 34, 55).

[27] Z. Charles and J. Konečnỳ, "On the outsized importance of learning rates in local update methods," *arXiv preprint arXiv:2007.00878*, 2020 (Cited in page 62).

[28] M. Chen, Z. Yang, W. Saad, C. Yin, H. V. Poor, and S. Cui, "A joint learning and communications framework for federated learning over wireless networks," *IEEE Transactions on Wireless Communications*, vol. 20, no. 1, pp. 269–283, 2020 (Cited in pages 14, 16).

[29] M. Chen, Z. Yang, W. Saad, C. Yin, H. V. Poor, and S. Cui, "A joint learning and communications framework for federated learning over wireless networks," *IEEE Transactions on Wireless Communications*, vol. 20, no. 1, pp. 269–283, 2020 (Cited in page 68).

[30] M. Chen, Z. Yang, W. Saad, C. Yin, H. V. Poor, and S. Cui, "A joint learning and communications framework for federated learning over wireless networks," *IEEE Transactions on Wireless Communications*, vol. 20, no. 1, pp. 269–283, 2021 (Cited in page 44).

[31] T. Chen, G. Giannakis, T. Sun, and W. Yin, "Lag: Lazily aggregated gradient for communication-efficient distributed learning," *Advances in neural information processing systems*, 2018 (Cited in page 15).

[32] Z. Chen, S. T. Maguluri, S. Shakkottai, and K. Shanmugam, "A lyapunov theory for finite-sample guarantees of asynchronous q-learning and td-learning variants," *arXiv preprint arXiv:2102.01567*, 2021 (Cited in page 78).

[33] Z. Chen, S. T. Maguluri, and M. Zubeldia, "Concentration of contractive stochastic approximation: Additive and multiplicative noise," *arXiv preprint arXiv:2303.15740*, 2023 (Cited in page 78).

[34] Z. Chen, S. Zhang, T. T. Doan, J.-P. Clarke, and S. T. Maguluri, "Finite-sample analysis of nonlinear stochastic approximation with applications in reinforcement learning," *Automatica*, vol. 146, p. 110 623, 2022 (Cited in pages 80, 81).

[35] Z. Chen, S. Zhang, T. T. Doan, S. T. Maguluri, and J.-P. Clarke, "Performance of q-learning with linear function approximation: Stability and finite-time analysis," *arXiv preprint arXiv:1905.11425*, p. 4, 2019 (Cited in pages 60, 64).

[36] A. Cohen, A. Daniely, Y. Drori, T. Koren, and M. Schain, "Asynchronous stochastic optimization robust to arbitrary delays," *Advances in Neural Information Processing Systems*, vol. 34, pp. 9024–9035, 2021 (Cited in pages 79, 89).

[37] G. Cohen, S. Afshar, J. Tapson, and A. van Schaik, "Emnist: Extending mnist to handwritten letters," in *2017 International Joint Conference on Neural Networks (IJCNN)*, 2017, pp. 2921–2926 (Cited in page 34).

[38] L. Collins, H. Hassani, A. Mokhtari, and S. Shakkottai, "Fedavg with fine tuning: Local updates lead to representation learning," in *Advances in Neural Information Processing Systems* (Cited in page 59).

[39] R. Crane and F. Roosta, "DINGO: Distributed newton-type method for gradient-norm optimization," *Advances in Neural Information Processing Systems*, vol. 32, 2019 (Cited in pages 7, 16).

[40] A. Czornik, A. Nawrat, M. Niezabitowski, and A. Szyda, "On the lyapunov and bohl exponent of time-varying discrete linear system," in *2012 20th Mediterranean Conference on Control Automation (MED)*, 2012 (Cited in page 27).

[41] N. Dal Fabbro, S. Dey, M. Rossi, and L. Schenato, "Shed: A newton-type algorithm for federated learning based on incremental hessian eigenvector sharing," *arXiv e-prints*, arXiv–2202, 2022 (Cited in page 40).

[42] ——, "SHED: A Newton-type algorithm for federated learning based on incremental hessian eigenvector sharing," *arXiv preprint arXiv:2202.05800*, 2022 (Cited in pages 44, 46, 50, 53).

[43] N. Dal Fabbro, A. Mitra, R. Heath, L. Schenato, and G. J. Pappas, "Over-the-air federated td learning," 2023 (Cited in page 76).

[44] N. Dal Fabbro, A. Mitra, and G. J. Pappas, "Federated TD learning over finite-rate erasure channels: Linear speedup under markovian sampling," *IEEE Control Systems Letters*, 2023 (Cited in pages 7, 72).

[46] G. Dalal, B. Szörényi, G. Thoppe, and S. Mannor, "Finite sample analyses for TD (0) with function approximation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, 2018 (Cited in page 60).

[47] T. Doan, S. Maguluri, and J. Romberg, "Finite-time analysis of distributed TD (0) with linear function approximation on multi-agent reinforcement learning," in *International Conference on Machine Learning*, PMLR, 2019, pp. 1626–1635 (Cited in page 60).

[48] T. T. Doan, "Finite-time analysis of markov gradient descent," *IEEE Transactions on Automatic Control*, 2022 (Cited in page 81).

[49] T. T. Doan, S. T. Maguluri, and J. Romberg, "Fast convergence rates of distributed subgradient methods with adaptive quantization," *IEEE Transactions on Automatic Control*, vol. 66, no. 5, pp. 2191–2205, 2020 (Cited in page 60).

[50] Y. Du and K. You, "Adaptive greedy quasi-newton with superlinear rate and global convergence guarantee," in *2022 IEEE 61st Conference on Decision and Control (CDC)*, 2022, pp. 7606–7611 (Cited in page 16).

[51] S. Dutta, G. Joshi, S. Ghosh, P. Dube, and P. Nagpurkar, "Slow and stale gradients can win the race: Error-runtime trade-offs in distributed sgd," in *International conference on artificial intelligence and statistics*, PMLR, 2018, pp. 803–812 (Cited in pages 61, 67).

[52] A. Elgabli, C. B. Issaid, A. S. Bedi, K. Rajawat, M. Bennis, and V. Aggarwal, "Fednew: A communication-efficient and privacy-preserving newton-type method for federated learning," in *International Conference on Machine Learning*, PMLR, 2022, pp. 5861–5877 (Cited in page 16).

[53] M. A. Erdogdu and A. Montanari, "Convergence rates of sub-sampled newton methods," *Advances in Neural Information Processing Systems*, vol. 28, 2015 (Cited in pages 15, 19, 22, 29, 34, 93, 99).

[54] N. D. Fabbro, S. Dey, M. Rossi, and L. Schenato, "SHED: A Newton-type algorithm for federated learning based on incremental hessian eigenvector sharing," *arXiv preprint arXiv:2202.05800*, 2022 (Cited in pages 34, 35).

[55] N. D. Fabbro, M. Rossi, L. Schenato, and S. Dey, "Q-shed: Distributed optimization at the edge via hessian eigenvectors quantization," *arXiv preprint arXiv:2305.10852*, 2023 (Cited in pages 57, 76).

[56] H. R. Feyzmahdavian, A. Aytekin, and M. Johansson, "A delayed proximal gradient method with linear convergence rate," in *2014 IEEE international workshop on machine learning for signal processing (MLSP)*, IEEE, 2014, pp. 1–6 (Cited in page 132).

[57] ——, "An asynchronous mini-batch algorithm for regularized stochastic optimization," *IEEE Transactions on Automatic Control*, vol. 61, no. 12, pp. 3740–3754, 2016 (Cited in page 87).

[58] E. Gorbunov, F. Hanzely, and P. Richtárik, "A unified theory of sgd: Variance reduction, sampling, quantization and coordinate descent," in *International Conference on Artificial Intelligence and Statistics*, PMLR, 2020, pp. 680–690 (Cited in page 5).

[59] ——, "Local SGD: Unified theory and new efficient methods," in *International Conference on Artificial Intelligence and Statistics*, PMLR, 2021, pp. 3556–3564 (Cited in page 59).

[60] R. M. Gower, N. Loizou, X. Qian, A. Sailanbayev, E. Shulgin, and P. Richtárik, "Sgd: General analysis and improved rates," in *International conference on machine learning*, PMLR, 2019, pp. 5200–5209 (Cited in page 5).

[61] A. Grafberger, M. Chadha, A. Jindal, J. Gu, and M. Gerndt, "Fedless: Secure and scalable federated learning using serverless computing," in *2021 IEEE International Conference on Big Data (Big Data)*, IEEE, 2021, pp. 164–173 (Cited in page 14).

[62] H. Guo, Y. Zhu, H. Ma, V. K. Lau, K. Huang, X. Li, H. Nong, and M. Zhou, "Over-the-air aggregation for federated learning: Waveform superposition and prototype validation," *Journal of Communications and Information Networks*, vol. 6, no. 4, pp. 429–442, 2021 (Cited in page 67).

[63] V. Gupta, A. Ghosh, M. Dereziński, R. Khanna, *et al.*, "LocalNewton: Reducing communication rounds for distributed learning," in *Uncertainty in Artificial Intelligence*, PMLR, 2021 (Cited in pages 6, 14, 16).

[64] M. Gurbuzbalaban, A. Ozdaglar, and P. A. Parrilo, "On the convergence rate of incremental aggregated gradient algorithms," *SIAM Journal on Optimization*, vol. 27, no. 2, pp. 1035–1048, 2017 (Cited in page 87).

[65] F. Haddadpour, M. M. Kamani, A. Mokhtari, and M. Mahdavi, "Federated learning with compression: Unified analysis and sharp guarantees," in *International Conference on Artificial Intelligence and Statistics*, PMLR, 2021, pp. 2350–2358 (Cited in page 60).

[66] F. Haddadpour and M. Mahdavi, "On the convergence of local descent methods in federated learning," *arXiv preprint arXiv:1910.14425*, 2019 (Cited in page 59).

[67] C. N. Hadjicostis and R. Touri, "Feedback control utilizing packet dropping network links," in *Proceedings of the 41st IEEE Conference on Decision and Control, 2002.*, IEEE, vol. 2, 2002, pp. 1205–1210 (Cited in page 60).

[68] A. Hard, K. Rao, R. Mathews, S. Ramaswamy, F. Beaufays, S. Augenstein, H. Eichner, C. Kiddon, and D. Ramage, "Federated learning for mobile keyboard prediction," *arXiv preprint arXiv:1811.03604*, 2018 (Cited in page 14).

[69] L. Huang, Y. Yin, Z. Fu, S. Zhang, H. Deng, and D. Liu, "Loadaboost: Loss-based adaboost federated machine learning with reduced computational complexity on iid and non-iid intensive care data," *PLOS ONE*, vol. 15, no. 4, pp. 1–16, Apr. 2020 (Cited in page 14).

[70] R. Islamov, X. Qian, and P. Richtárik, "Distributed second order methods with fast rates and compressed communication," in *Proceedings of the 38th International Conference on Machine Learning*, PMLR, 2021 (Cited in page 16).

[71] P. Kairouz, H. B. McMahan, *et al.*, "Advances and open problems in federated learning," *Foundations and Trends® in Machine Learning*, vol. 14, no. 1–2, pp. 1–210, 2021 (Cited in page 14).

[72] S. P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, and A. T. Suresh, "SCAFFOLD: Stochastic controlled averaging for federated learning," in *Proceedings of the 37th International Conference on Machine Learning*, PMLR, 2020 (Cited in pages 15, 59, 62).

[73] A. Khaled, K. Mishchenko, and P. Richtárik, "First analysis of local gd on heterogeneous data," *arXiv preprint arXiv:1909.04715*, 2019 (Cited in page 59).

[74] ——, "Tighter theory for local SGD on identical and heterogeneous data," in *International Conference on Artificial Intelligence and Statistics*, PMLR, 2020, pp. 4519–4529 (Cited in pages 5, 59).

[75] L. U. Khan, S. R. Pandey, N. H. Tran, W. Saad, Z. Han, M. N. H. Nguyen, and C. S. Hong, "Federated learning for edge networks: Resource optimization and incentive mechanism," *IEEE Communications Magazine*, vol. 58, no. 10, pp. 88–93, 2020 (Cited in page 44).

[76] S. Khirirat, H. R. Feyzmahdavian, and M. Johansson, "Distributed learning with compressed gradients," *arXiv preprint arXiv:1806.06573*, 2018 (Cited in page 5).

[77] S. Khodadadian, P. Sharma, G. Joshi, and S. T. Maguluri, "Federated reinforcement learning: Linear speedup under markovian sampling," in *ICML*, PMLR, 2022, pp. 10 997–11 057 (Cited in pages 7, 60, 62, 70, 72, 94).

[78] A. Koloskova, S. U. Stich, and M. Jaggi, "Sharper convergence guarantees for asynchronous sgd for distributed and federated learning," *Advances in Neural Information Processing Systems*, vol. 35, pp. 17 202–17 215, 2022 (Cited in pages 61, 68, 79, 80, 89).

[79] J. Konečnỳ, H. B. McMahan, D. Ramage, and P. Richtárik, "Federated optimization: Distributed machine learning for on-device intelligence," *arXiv preprint arXiv:1610.02527*, 2016 (Cited in page 59).

[80] J. Konečnỳ, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," *arXiv preprint arXiv:1610.05492*, 2016 (Cited in page 5).

[81] M. Krouka, A. Elgabli, C. B. Issaid, and M. Bennis, "Communication-efficient and federated multi-agent reinforcement learning," *IEEE Transactions on Cognitive Communications and Networking*, vol. 8, no. 1, pp. 311–320, 2022 (Cited in page 61).

[82] ——, "Communication-efficient federated learning: A second order Newton-type method with analog over-the-air aggregation," *IEEE Transactions on Green Communications and Networking*, vol. 6, no. 3, pp. 1862–1874, 2022 (Cited in page 44).

[83] C. Lakshminarayanan and C. Szepesvári, "Linear stochastic approximation: Constant step-size and iterate averaging," *arXiv preprint arXiv:1709.04073*, 2017 (Cited in page 60).

[84] D. A. Levin and Y. Peres, *Markov chains and mixing times*. American Mathematical Soc., 2017, vol. 107 (Cited in page 64).

[85] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," *IEEE Signal Processing Magazine*, vol. 37, no. 3, pp. 50–60, 2020 (Cited in pages 2, 6, 14, 43).

[86] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," in *Proceedings of Machine Learning and Systems*, I. Dhillon, D. Papailiopoulos, and V. Sze, Eds., 2020 (Cited in pages 14, 15).

[87] D. C. Liu and J. Nocedal, "On the limited memory bfgs method for large scale optimization," *Mathematical programming*, vol. 45, no. 1-3, pp. 503–528, 1989 (Cited in pages 16, 35).

[88] R. Liu and A. Olshevsky, "Distributed td (0) with almost no communication," *IEEE Control Systems Letters*, 2023 (Cited in page 60).

[89] Y. Liu, M. Peng, G. Shou, Y. Chen, and S. Chen, "Toward Edge Intelligence: Multiaccess Edge Computing for 5G and Internet of Things," *IEEE Internet of Things Journal*, vol. 7, 2020 (Cited in page 14).

[90] Y. Liu, Y. Zhu, and J. James, "Resource-constrained federated edge learning with heterogeneous data: Formulation and analysis," *IEEE Transactions on Network Science and Engineering*, vol. 9, no. 5, pp. 3166–3178, 2021 (Cited in page 16).

[91] A. M. Lyapunov, "The general problem of the stability of motion," *International journal of control*, vol. 55, no. 3, pp. 531–534, 1992 (Cited in page 26).

[92] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial Intelligence and Statistics*, PMLR, 2017, pp. 1273–1282 (Cited in pages 2, 4, 59).

[93] ——, "Communication-efficient learning of deep networks from decentralized data," in *Artificial intelligence and statistics*, PMLR, 2017, pp. 1273–1282 (Cited in page 4).

[94] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. Y. Arcas, "Communication-efficient learning of deep networks from decentralized data," in *AISTATS*, 2017 (Cited in pages 14, 15).

[97] N. Michelusi, G. Scutari, and C.-S. Lee, "Finite-bit quantization for distributed algorithms with linear convergence," *IEEE Transactions on Information Theory*, vol. 68, no. 11, pp. 7254–7280, 2022 (Cited in page 60).

[98] K. Mishchenko, E. Gorbunov, M. Takáč, and P. Richtárik, "Distributed learning with compressed gradient differences," *arXiv preprint arXiv:1901.09269*, 2019 (Cited in page 5).

[99] K. Mishchenko, G. Malinovsky, S. Stich, and P. Richtárik, "ProxSkip: Yes! Local Gradient Steps Provably Lead to Communication Acceleration! Finally!" *arXiv preprint arXiv:2202.09357*, 2022 (Cited in page 59).

[100] A. Mitra, R. Jaafar, G. J. Pappas, and H. Hassani, "Linear convergence in federated learning: Tackling client heterogeneity and sparse gradients," *Advances in Neural Information Processing Systems*, vol. 34, pp. 14 606–14 619, 2021 (Cited in pages 5, 59, 62).

[101] ——, "Linear convergence in federated learning: Tackling client heterogeneity and sparse gradients," *Advances in Neural Information Processing Systems*, vol. 34, pp. 14 606–14 619, 2021 (Cited in page 15).

[102] A. Mitra, G. J. Pappas, and H. Hassani, "Temporal difference learning with compressed updates: Error-feedback meets reinforcement learning," *arXiv preprint arXiv:2301.00944*, 2023 (Cited in page 66).

[103] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu, "Asynchronous methods for deep reinforcement learning," in *International conference on machine learning*, PMLR, 2016, pp. 1928–1937 (Cited in page 61).

[104] D. Nagaraj, X. Wu, G. Bresler, P. Jain, and P. Netrapalli, "Least squares regression with markovian data: Fundamental limits and algorithms," *Advances in neural information processing systems*, vol. 33, pp. 16 666–16 676, 2020 (Cited in pages 62, 70).

[105] ——, "Least squares regression with markovian data: Fundamental limits and algorithms," *Advances in neural information processing systems*, vol. 33, pp. 16 666–16 676, 2020 (Cited in pages 83, 88).

[106] A. Nair, P. Srinivasan, S. Blackwell, C. Alcicek, R. Fearon, A. De Maria, V. Panneershelvam, M. Suleyman, C. Beattie, S. Petersen, *et al.*, "Massively parallel methods for deep reinforcement learning," *arXiv preprint arXiv:1507.04296*, 2015 (Cited in pages 8, 61).

[107] J. S. Ng, W. Y. B. Lim, Z. Xiong, X. Cao, J. Jin, D. Niyato, C. Leung, and C. Miao, "Reputation-aware hedonic coalition formation for efficient serverless hierarchical federated learning," *IEEE Transactions on Parallel and Distributed Systems*, vol. 33, no. 11, pp. 2675–2686, 2021 (Cited in page 14).

[108] J. Nguyen, K. Malik, H. Zhan, A. Yousefpour, M. Rabbat, M. Malek, and D. Huba, "Federated learning with buffered asynchronous aggregation," in *International Conference on Artificial Intelligence and Statistics*, PMLR, 2022, pp. 3581–3607 (Cited in pages 61, 67, 68).

[109] ——, "Federated learning with buffered asynchronous aggregation," in *International Conference on Artificial Intelligence and Statistics*, PMLR, 2022, pp. 3581–3607 (Cited in page 80).

[110] T. D. Nguyen, A. R. Balef, C. T. Dinh, N. H. Tran, D. T. Ngo, T. Anh Le, and P. L. Vo, "Accelerating federated edge learning," *IEEE Communications Letters*, vol. 25, no. 10, pp. 3282–3286, 2021 (Cited in page 14).

[111] Y. Niu, Z. Fabian, S. Lee, M. Soltanolkotabi, and S. Avestimehr, "Ml-bfgs: A momentum-based l-bfgs for distributed large-scale neural network optimization," *Transactions on Machine Learning Research*, 2023 (Cited in page 94).

[112] K. Osawa, S. Li, and T. Hoefler, "Pipefisher: Efficient training of large language models using pipelining and fisher information matrices," *Proceedings of Machine Learning and Systems*, vol. 5, 2023 (Cited in page 94).

[113] C. Pan, Z. Wang, H. Liao, Z. Zhou, X. Wang, M. Tariq, and S. Al-Otaibi, "Asynchronous federated deep reinforcement learning-based urllc-aware computation offloading in space-assisted vehicular networks," *IEEE Transactions on Intelligent Transportation Systems*, 2022 (Cited in page 7).

[114] F. Pase, M. Giordani, and M. Zorzi, "On the convergence time of federated learning over wireless networks under imperfect CSI," *IEEE International Conference on Communications Workshops (ICC WKSHPS)*, pp. 1–6, 2021 (Cited in page 55).

[115] G. Patil, L. Prashanth, D. Nagaraj, and D. Precup, "Finite time analysis of temporal difference learning with linear function approximation: Tail averaging and regularisation," in *International Conference on Artificial Intelligence and Statistics*, PMLR, 2023, pp. 5438–5448 (Cited in page 60).

[116] G. Perin *et al.*, "Optimizing edge computing resources towards greener networks and services," 2023 (Cited in page 2).

[117] Q.-V. Pham, F. Fang, V. N. Ha, M. J. Piran, M. Les, L. B. Le, W.-J. Hwang, and Z. Ding, "A Survey of Multi-Access Edge Computing in 5G and Beyond: Fundamentals, Technology Integration, and State-of-the-Art," *IEEE Access*, vol. 8, 2020 (Cited in pages 2, 14).

[118] J. Qi, Q. Zhou, L. Lei, and K. Zheng, "Federated reinforcement learning: techniques, applications, and open challenges," *arXiv preprint arXiv:2108.11887*, 2021 (Cited in page 60).

[119] M. G. Rabbat and R. D. Nowak, "Quantized incremental algorithms for distributed optimization," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 4, pp. 798–808, 2005 (Cited in page 60).

[120] S. J. Reddi, J. Konečnỳ, P. Richtárik, B. Póczós, and A. Smola, "Aide: Fast and communication efficient distributed optimization," *arXiv preprint arXiv:1608.06879*, 2016 (Cited in page 16).

[121] A. Reisizadeh, A. Mokhtari, H. Hassani, A. Jadbabaie, and R. Pedarsani, "Fedpaq: A communication-efficient federated learning method with periodic averaging and quantization," in *International Conference on Artificial Intelligence and Statistics*, PMLR, 2020, pp. 2021–2031 (Cited in pages 60, 69).

[122] A. Reisizadeh, A. Mokhtari, H. Hassani, and R. Pedarsani, "An exact quantized decentralized gradient descent algorithm," *IEEE Transactions on Signal Processing*, vol. 67, no. 19, pp. 4934–4947, 2019 (Cited in page 60).

[123] N. Rieke, J. Hancox, W. Li, F. Milletari, H. R. Roth, S. Albarqouni, S. Bakas, M. N. Galtier, B. A. Landman, K. Maier-Hein, *et al.*, "The future of digital health with federated learning," *NPJ digital medicine*, vol. 3, no. 1, pp. 1–7, 2020 (Cited in page 14).

[124] H. Robbins and S. Monro, "A stochastic approximation method," *The Annals of Mathematical Statistics*, vol. 22, no. 3, pp. 400–407, 1951 (Cited in page 78).

[125] M. Safaryan, R. Islamov, X. Qian, and P. Richtárik, "FedNL: Making Newton-type methods applicable to federated learning," in *Proceedings of the 39th International Conference on Machine Learning*, 2022 (Cited in pages 6, 7, 14, 16, 19, 35, 37, 44, 55).

[126] L. Schenato, B. Sinopoli, M. Franceschetti, K. Poolla, and S. S. Sastry, "Foundations of control and estimation over lossy networks," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 163–187, 2007 (Cited in page 60).

[127] T. Sery and K. Cohen, "On analog gradient descent learning over multiple access fading channels," *IEEE Transactions on Signal Processing*, vol. 68, pp. 2897–2911, 2020 (Cited in pages 61, 66, 71, 74).

[128] O. Shamir, N. Srebro, and T. Zhang, "Communication-efficient distributed optimization using an approximate newton-type method," in *Proceedings of the 31st International Conference on Machine Learning*, 2014, pp. 1000–1008 (Cited in pages 7, 16).

[129] R. Sheikh, M. Patel, and A. Sinhal, "Recognizing MNIST handwritten data set using PCA and LDA," in *International Conference on Artificial Intelligence: Advances and Applications*, 2020 (Cited in pages 34, 55).

[130] H. Shen, K. Zhang, M. Hong, and T. Chen, "Towards understanding asynchronous advantage actor-critic: Convergence and linear speedup," *IEEE Transactions on Signal Processing*, vol. 71, pp. 2579–2594, 2023 (Cited in pages 61, 67).

[131] Y. Shi, K. Yang, T. Jiang, J. Zhang, and K. B. Letaief, "Communication-efficient edge AI: Algorithms and systems," *IEEE Communications Surveys Tutorials*, vol. 22, no. 4, pp. 2167–2191, 2020 (Cited in pages 13, 14, 43).

[132] N. Shlezinger, M. Chen, Y. C. Eldar, H. V. Poor, and S. Cui, "UVeQFed: Universal vector quantization for federated learning," *IEEE Transactions on Signal Processing*, vol. 69, pp. 500–514, 2020 (Cited in pages 44, 47, 48).

[133] V. Smith, C.-K. Chiang, M. Sanjabi, and A. Talwalkar, "Federated multi-task learning," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017, pp. 4427–4437 (Cited in page 14).

[134] S. Soori, K. Mishchenko, A. Mokhtari, M. M. Dehnavi, and M. Gurbuzbalaban, "Dave-qn: A distributed averaged quasi-newton method with local superlinear convergence rate," in *International Conference on Artificial Intelligence and Statistics*, PMLR, 2020, pp. 1965–1976 (Cited in page 16).

[135] R. Srikant and L. Ying, "Finite-time error bounds for linear stochastic approximation and TD learning," in *Conference on Learning Theory*, PMLR, 2019, pp. 2803–2830 (Cited in pages 60, 61, 64, 65, 69–72, 78–81, 83, 84, 86, 88, 89, 109, 110, 120, 122, 132).

[136] S. U. Stich, "Local SGD converges fast and communicates little," *arXiv preprint arXiv:1805.09767*, 2018 (Cited in pages 5, 59).

[137] S. U. Stich and S. P. Karimireddy, "The error-feedback framework: Better rates for sgd with delayed gradients and compressed updates," *The Journal of Machine Learning Research*, vol. 21, no. 1, pp. 9613–9648, 2020 (Cited in pages 79, 82, 83, 87, 88).

[138] R. S. Sutton, "Learning to predict by the methods of temporal differences," *Machine learning*, vol. 3, no. 1, pp. 9–44, 1988 (Cited in pages 60, 63).

[139] C. T. Dinh, N. H. Tran, T. D. Nguyen, W. Bao, A. Rezaei Balef, B. B. Zhou, and A. Zomaya, "Done: Distributed approximate newton-type method for federated edge learning," *IEEE Transactions on Parallel and Distributed Systems*, 2022 (Cited in pages 6, 14, 16, 19, 44).

[140] J. Tsitsiklis and B Vanroy, "An analysis of temporal-difference learning with function approximation," *IEEE Transactions on Automatic Control*, vol. 42, no. 5, pp. 674–690, 1997 (Cited in page 80).

[141] J. N. Tsitsiklis and B. Van Roy, "An analysis of temporal-difference learning with function approximation," in *IEEE Transactions on Automatic Control*, 1997 (Cited in pages 60, 64).

[142] M. M. Wadu, S. Samarakoon, and M. Bennis, "Federated learning under channel uncertainty: Joint client scheduling and resource allocation," in *2020 IEEE Wireless Communications and Networking Conference (WCNC)*, IEEE, 2020 (Cited in page 55).

[143] H. Wang, A. Mitra, H. Hassani, G. J. Pappas, and J. Anderson, "Federated temporal difference learning with linear function approximation under environmental heterogeneity," *arXiv:2302.02212*, 2023 (Cited in pages 60, 62, 70, 94).

[144] S. Wang, F. Roosta-Khorasani, P. Xu, and M. W. Mahoney, "GIANT: Globally improved approximate newton method for distributed optimization," in *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, Montréal, Canada, 2018 (Cited in pages 7, 16, 19, 34, 35, 44, 53).

[145] B. Woodworth, K. K. Patel, S. U. Stich, Z. Dai, B. Bullins, H. B. McMahan, O. Shamir, and N. Srebro, "Is Local SGD Better than Minibatch SGD?" *arXiv preprint arXiv:2002.07839*, 2020 (Cited in page 59).

[146] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-mnist: A novel image dataset for benchmarking machine learning algorithms," *arXiv preprint arXiv:1708.07747*, 2017 (Cited in pages 34, 55).

[147] H. H. Yang, Z. Chen, T. Q. Quek, and H. V. Poor, "Revisiting analog over-the-air machine learning: The blessing and curse of interference," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 3, pp. 406–419, 2021 (Cited in pages 61, 66).

[148] H. Yu, S. Yang, and S. Zhu, "Parallel restarted sgd with faster convergence and less communication: Demystifying why model averaging works for deep learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 5693–5700 (Cited in page 15).

[149] S. Zeng, T. T. Doan, and J. Romberg, "Finite-time convergence rates of decentralized stochastic approximation with applications in multi-agent and multi-task learning," *IEEE Transactions on Automatic Control*, 2022 (Cited in pages 78, 79).

[150] J. Zhang, K. You, and T. Başar, "Distributed adaptive newton methods with global superlinear convergence," *Automatica*, vol. 138, p. 110 156, 2022 (Cited in pages 14, 16).

[151] Y. Zhang and X. Lin, "Disco: Distributed optimization for self-concordant empirical loss," in *International conference on machine learning*, PMLR, 2015, pp. 362–370 (Cited in page 16).

[152] Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, and V. Chandra, "Federated learning with non-iid data," *arXiv preprint arXiv:1806.00582*, 2018 (Cited in page 14).

[153] Z. Zhou, P. Mertikopoulos, N. Bambos, P. Glynn, Y. Ye, L.-J. Li, and L. Fei-Fei, "Distributed asynchronous optimization with unbounded delays: How slow can you go?" In *International Conference on Machine Learning*, PMLR, 2018, pp. 5970–5979 (Cited in page 89).

[154] G. Zhu, Y. Du, D. Gündüz, and K. Huang, "One-bit over-the-air aggregation for communication-efficient federated edge learning: Design and convergence analysis," *IEEE Transactions on Wireless Communications*, vol. 20, no. 3, pp. 2120–2135, 2020 (Cited in pages 61, 66).

[155] H. Zhu, J. Xu, S. Liu, and Y. Jin, "Federated learning on non-iid data: A survey," *Neurocomputing*, vol. 465, pp. 371–390, 2021 (Cited in page 14).