MARINA-P: SUPERIOR PERFORMANCE IN NON-SMOOTH FEDERATED OPTIMIZATION WITH ADAPTIVE STEPSIZES

Anonymous authors
Paper under double-blind review

ABSTRACT

Non-smooth communication-efficient federated optimization remains largely unexplored theoretically, despite its importance in machine learning applications. We consider a setup focusing on optimizing downlink communication by improving state-of-the-art schemes like EF21-P (Gruntkowska et al., 2023) and MARINA-P (Gruntkowska et al., 2024) in the non-smooth convex setting. Our key contributions include extending the non-smooth convex theory of EF21-P from single-node to distributed settings and generalizing MARINA-P to non-smooth convex optimization. For both algorithms, we prove optimal $\mathcal{O}\left({}^{1}/\sqrt{T}\right)$ convergence rates under standard assumptions and establish matching communication complexity bounds with classical subgradient methods. We provide theoretical guarantees under constant, decreasing, and adaptive (Polyak-type) stepsizes. Our experiments demonstrate MARINA-P's superior performance with correlated compressors in both smooth non-convex and non-smooth convex settings. This work presents the first theoretical analysis of distributed non-smooth optimization with server-to-worker compression, including comprehensive analysis for various stepsize schemes.

1 INTRODUCTION

In recent years, the machine learning community has witnessed a paradigm shift toward larger models and datasets, spurring major performance gains but also posing new hardware, algorithmic, and software challenges (LeCun et al., 2015; Bottou et al., 2018; Kaplan et al., 2020; Deng et al., 2009).

The Rise of Big Data and Distributed Systems. The sheer volume of data needed for cutting-edge models has driven the adoption of distributed systems (Dean et al., 2012; Khirirat et al., 2018; Lin et al., 2018), since single-machine setups can no longer handle the storage and computational demands. This approach is particularly relevant in supervised learning (Hastie et al., 2009; Shalev-Shwartz & Ben-David, 2014; Vapnik, 2013), often formulated as:

$$\min_{x\in\mathbb{R}^d}\left\{f(x):=\frac{1}{n}\sum_{i=1}^{n}f_i(x)\right\}, \tag{1}$$

where n denotes the number of clients, $x\in\mathbb{R}^d$ is the model's parameter vector, and $f_i(x)$ is the local loss on client i. Throughout, we assume each f_i is convex (possibly non-smooth).

Federated learning (FL) (McMahan et al., 2016; Konečný et al., 2016b;a; McMahan et al., 2017) extends the distributed paradigm to heterogeneous clients with decentralized data, seeking to avoid central data aggregation and preserve privacy. In FL, devices connect to a central server that orchestrates training (Konečný et al., 2016b; Kairouz et al., 2021): each device locally updates parameters using its data, then sends these updates to the server. The server aggregates them, performs global calculations, and broadcasts new parameters back to devices. This process continues until convergence or acceptable performance is reached.

Communication Challenges in Large-scale Model Training. Although distributing data alleviates storage and compute constraints, it introduces substantial communication overhead. Modern gradient-based methods (Bottou, 2012; Kingma & Ba, 2014; Demidovich et al., 2023; Duchi et al., 2011; Robbins & Monro, 1951) require iterative updates for all d parameters, making frequent transmission of high-dimensional gradients expensive. Two broad approaches reduce this burden: (i) performing multiple local gradient steps before communicating, as in LocalSGD (Stich, 2020; Khaled et al., 2020; Woodworth et al., 2020; Yi et al., 2024; Sadiev et al., 2022; Richtárik et al., 2024), and (ii) compressing gradients via lossy transformations (Khirirat et al., 2018; Alistarh et al., 2018b; Mishchenko et al., 2020; 2019; Li et al., 2020; Li & Richtárik, 2021; Richtárik et al., 2021; Fatkhullin et al., 2021; Richtárik et al., 2022; Seide et al., 2014; Alistarh et al., 2017; Panferov et al., 2024). Moreover, studies of 4G LTE and 5G networks (Huang et al., 2012; Narayanan et al., 2021) show that upload/download speeds are often comparable, emphasizing that both server-to-worker and worker-to-server communication must be optimized.

Prevalence of Non-smooth Objectives in Machine Learning Applications. Despite notable advances in distributed optimization, theoretical work has primarily targeted smooth objectives, leaving non-smooth problems less explored in federated contexts. Non-smoothness arises in various ML scenarios: ReLU activations (Glorot et al., 2011; Nair & Hinton, 2010), L1 regularization for sparsity (Tibshirani, 1996; Zou & Hastie, 2005), hinge loss (Cortes, 1995), total variation (Rudin et al., 1992; Chambolle, 2004), quantile regression (Koenker & Bassett Jr, 1978), max-pooling (Scherer et al., 2010), submodular minimization (Bach, 2013), Huber loss (Huber, 1964), and graph-based learning (Hallac et al., 2015).

Adaptive Stepsizes are Widely Used in Practice. Because constants like L-Lipschitz continuity or smoothness parameters are difficult to determine in deep learning, practitioners rely on adaptive learning rates. Popular methods include AdaGrad (Duchi et al., 2011), RMSProp, Adam (Kingma & Ba, 2014), and AMSGrad (Reddi et al., 2018), all of which adjust per-parameter stepsizes based on observed gradients.

1.1 Notation and Assumptions

We denote the set $\{1,2,\cdots,n\}$ by $[n]$. For vectors, $\|\cdot\|_2$ represents the Euclidean norm, while for matrices, it denotes the spectral norm. The inner product of vectors u and v is denoted by $\langle u, v\rangle$. We use $\mathcal{O}(\cdot)$ to hide absolute constants. We denote $R_0 := \left\|x^0 - x^*\right\|_2$.

Our analysis relies on the following standard assumptions:

Assumption 1. *The function f has at least one minimizer, denoted by x^*.*

Assumption 2. *The functions f_i are convex for all $i \in [n]$.*

In the distributed setting, assuming convexity for individual functions f_i is sufficient, as it implies convexity for f itself.

Assumption 3 (Lipschitz continuity of f_i)**.** *Functions f_i are $L_{0,i}$-Lipschitz continuous for all $i \in [n]$. That is, for all $i \in [n]$, there exists $L_{0,i} > 0$ such that $|f_i(x) - f_i(y)| \leq L_{0,i}\|x-y\|_2, \quad \forall x, y \in \mathbb{R}^d$.*

If each f_i is Lipschitz continuous, then by Jensen's inequality, f is L_0-Lipschitz with $L_0 := \frac{1}{n}\sum_{i=1}^n L_{0,i}$ (Nesterov, 2013).

Both convexity and Lipschitz continuity of f are standard assumptions in non-smooth optimization (Vorontsova et al., 2021; Nesterov, 2013; Bubeck, 2015; Beck, 2017; Duchi, 2018; Lan, 2020; Drusvyatskiy, 2020). Moreover, L_0 and $L_{0,i}$-Lipschitz continuity imply uniformly bounded subgradients (Beck, 2017), a property that will be useful in our proofs:

$$\|\partial f(x)\|_2 \quad \leq \quad L_0 \quad \forall x \in \mathbb{R}^d, \tag{2}$$

$$\|\partial f_i(x)\|_2 \quad \leq \quad L_{0,i} \quad \forall x \in \mathbb{R}^d \text{ and } \forall i \in [n]. \tag{3}$$

We define $\widetilde{L}_0 := \sqrt{\frac{1}{n}\sum_{i=1}^n L_{0,i}^2}$ and $\overline{L}_0 := \frac{1}{n}\sum_{i=1}^n L_{0,i}$. By the arithmetic-quadratic mean inequality, we have $\overline{L}_0 \leq \widetilde{L}_0$.

Following classical optimization literature (Nemirovski et al., 2009; Beck, 2017; Duchi, 2018; Lan, 2020; Drusvyatskiy, 2020), for non-smooth convex objectives, we aim to find an ε-suboptimal solution: a random vector $\hat{x} \in \mathbb{R}^d$ satisfying $\mathbb{E}\left[f(\hat{x}) - f(x^*)\right] \leq \varepsilon$, where $\mathbb{E}\left[\cdot\right]$ denotes the expectation over algorithmic randomness.

To assess the efficiency of distributed subgradient-based algorithms, we primarily use two metrics:

1. Communication complexity (alternatively, communication cost): The expected total number of floats per worker required to communicate to reach an ε-suboptimal solution. In this paper, we focus on server-to-worker communication compression.

2. Iteration complexity: The number of communication rounds needed to achieve an ε-suboptimal solution.

1.2 Related work

Subgradient Methods in Non-smooth Convex Optimization. Subgradient methods, pioneered in the 1960s (Shor et al., 1985; Polyak, 1987), remain central to non-smooth convex optimization. Classic theory establishes $\mathcal{O}\left(1/\sqrt{T}\right)$ rates for general convex objectives (Nesterov, 2013; Vorontsova et al., 2021; Bubeck, 2015; Beck, 2017; Duchi, 2018; Lan, 2020; Drusvyatskiy, 2020) and $\mathcal{O}\left(1/T\right)$ for strongly convex problems (Beck, 2017; Drusvyatskiy, 2020). For unknown T, decreasing stepsizes of order $\mathcal{O}\left(1/\sqrt{t}\right)$ or $\mathcal{O}\left(1/t\right)$ add a logarithmic factor, yielding $\mathcal{O}\left(\log T/\sqrt{T}\right)$ (Nesterov, 2013) and $\mathcal{O}\left(\log T/T\right)$ (Hazan et al., 2007; Hazan & Kale, 2014). Nevertheless, recent works (Zhu et al., 2024; Lacoste-Julien et al., 2012; Rakhlin et al., 2011) have removed these factors, attaining optimal rates in convex and strongly convex settings. In the stochastic regime, mirror-descent methods also achieve $\mathcal{O}\left(1/\sqrt{T}\right)$ (Nemirovski et al., 2009). Beyond averaged-iterate convergence, tighter last-iterate analyses (Jain et al., 2019; Zamani & Glineur, 2023) provide stronger guarantees. Subgradient methods remain crucial for large-scale machine learning tasks, including support vector machines and structured prediction (Shalev-Shwartz et al., 2007; Ratliff et al., 2007).

Communication Compression. Before discussing more advanced optimization methods, let us consider the simplest baseline: the standard subgradient method (SM) [1], which iteratively performs updates [2]

$$x^{t+1} = x^t - \tfrac{\gamma_t}{n} \textstyle\sum_{i=1}^n g_i^t, \tag{4}$$

where $g_i^t = \partial f_i(x^t)$ is a subgradient of f_i at x^t. In the distributed setting, the method can be implemented as follows: each worker calculates g_i^t and sends it to the server, where the subgradients are aggregated. The server takes the step and broadcasts x^{t+1} back to the workers. With stepsize $\gamma_t := R_0/L_0\sqrt{T}$, where $R_0 := \left\|x^0 - x^*\right\|_2$ and T is the total number of iterations, SM finds an ε-approximate solution after $\mathcal{O}\left(L_0^2 R_0^2/\varepsilon^2\right)$ steps (Nesterov, 2013; Drusvyatskiy, 2020). Since at each step the workers and the server send $\Theta(d)$ coordinates/floats, the worker-to-server and server-to-worker communication costs are $\mathcal{O}\left(dL_0^2 R_0^2/\varepsilon^2\right)$. To formally quantify communication costs, we introduce the following definition.

Definition 1. *The worker-to-server (w2s, uplink) and server-to-worker (s2w, downlink) communication complexities of a method are the expected number of coordinates/floats that a worker sends to the server and that the server sends to a worker, respectively, to find an ε–solution.*

Communication compression techniques, such as sparsification (Wang et al., 2018; Mishchenko et al., 2020; Alistarh et al., 2018b; Wangni et al., 2018; Konečný & Richtárik, 2018) and quantization (Alistarh et al., 2017; Wen et al., 2017; Zhang et al., 2016; Horváth et al., 2022; Wu et al., 2018; Mishchenko et al., 2019), are known to be immensely powerful for reducing the communication overhead of gradient-type methods. Existing literature primarily considers two main classes of compression operators: *unbiased* and *biased (contractive)* compressors.

Definition 2. *(Unbiased compressor). A stochastic mapping $\mathcal{Q} : \mathbb{R}^d \to \mathbb{R}^d$ is called an unbiased compressor/compression operator if there exists $\omega \geq 0$ such that for any $x \in \mathbb{R}^d$:*

$$\mathbb{E}[\mathcal{Q}(x)] = x, \quad \mathbb{E}\left[\|\mathcal{Q}(x) - x\|_2^2\right] \leq \omega \|x\|_2^2 . \tag{5}$$

This definition encompasses a wide range of well-known compression techniques, including Rand$K$ sparsification (Stich et al., 2018), random dithering (Roberts, 1962; Goodall, 1951), and natural

[1] In this paper, we use the non-normalized form (4) of the subgradient method studied in (Vorontsova et al., 2021; Bubeck, 2015; Beck, 2017; Duchi, 2018; Lan, 2020; Drusvyatskiy, 2020; Nemirovski et al., 2009). Earlier works (Shor et al., 1985; Polyak, 1987) typically employed SM in the form $x^{t+1} = x^t - \frac{\gamma_t}{\|\partial f(x^t)\|}\partial f(x^t)$, which includes an additional normalization term $\left\|\partial f(x^t)\right\|$.

[2] For constrained optimization problems, the subgradient method typically operates through projections onto a convex set $\mathcal{X}$ (see (Bubeck, 2015; Lacoste-Julien et al., 2012; Beck, 2017; Duchi, 2018)). However, when optimizing over an unbounded domain, i.e., $\mathcal{X} = \mathbb{R}^d$, projections are not needed.

compression (Horváth et al., 2022). Notable examples of methods employing compressor (5) are QSGD (Alistarh et al., 2017), DCGD (Khirirat et al., 2018), MARINA (Gorbunov et al., 2021), DIANA (Mishchenko et al., 2019), VR-DIANA (Horváth et al., 2019), DASHA (Tyurin & Richtárik, 2023), FedCOMGATE (Haddadpour et al., 2021), FedPAQ (Reisizadeh et al., 2020), FedSTEPH (Das et al., 2020), FedCOM (Haddadpour et al., 2021), ADIANA (Li et al., 2020), NEOLITHIC (Huang et al., 2022a), ACGD (Li et al., 2020), and CANITA (Li & Richtárik, 2021). However, Definition 2 does not cover another important class of practically more favorable compressors, called *contractive*, which are usually biased.

Definition 3. *(Contractive compressor). A stochastic mapping $\mathcal{C}: \mathbb{R}^d \to \mathbb{R}^d$ is called a contractive compressor/compression operator if there exists $\alpha \in (0,1]$ such that for any $x \in \mathbb{R}^d$:*

$$\mathbb{E}\left[\|\mathcal{C}(x) - x\|_2^2\right] \leq (1-\alpha)\|x\|_2^2 . \tag{6}$$

We denote the families of compressors satisfying Definitions 2 and 3 by $\mathbb{U}(\omega)$ and $\mathbb{B}(\alpha)$, respectively.[3]

Inequality (6) is satisfied by many compressors, including Top$K$ (Ström, 2015; Dryden et al., 2016; Aji & Heafield, 2017; Alistarh et al., 2018b), quantization (Alistarh et al., 2017; Horváth et al., 2022), low-rank approximations (Vogels et al., 2019; 2020; Safaryan et al., 2021), and count-sketches (Ivkin et al., 2019; Rothchild et al., 2020). For broader surveys, see (Beznosikov et al., 2023; Demidovich et al., 2023; Safaryan et al., 2022). However, naive distributed SGD with biased compression (e.g., Top$K$) can diverge (Beznosikov et al., 2023). Error Feedback (EF14), introduced by Seide et al. (2014), emerged as a key technique to avert such divergence. Early theory of EF14 was confined to single-node settings (Stich et al., 2018; Alistarh et al., 2018a; Stich & Karimireddy, 2019), then expanded to distributed setting (Cordonnier, 2018; Beznosikov et al., 2023; Koloskova et al., 2020). Richtárik et al. (2021) reformulated EF14 into EF21, achieving optimal $\mathcal{O}\left(1/T\right)$ convergence for smooth non-convex problems, surpassing the previous $\mathcal{O}\left(1/T^{2/3}\right)$ rate (Koloskova et al., 2020).

The EF21 framework led to multiple variants (Richtárik et al., 2022; Fatkhullin et al., 2021), including bidirectional (s2w and w2s) biased compression. Gruntkowska et al. (2023) introduced EF21-P, combining biased s2w and unbiased w2s to improve complexity in smooth strongly convex settings. Later, Gruntkowska et al. (2024) proposed MARINA-P for smooth non-convex problems, delivering sharper bounds than both EF21 and EF21-P. Concurrently, Anonymous (2024) provided the first non-smooth convergence guarantees for EF21-P, albeit restricted to single-node scenarios.

In order to express communication complexities, we will further need the following quantities.

Definition 4 (Expected density)**.** *For the given compression operators $\mathcal{Q}(x)$ and $\mathcal{C}(x)$, we define the expected density as $\zeta_{\mathcal{Q}} = \sup_{x\in\mathbb{R}^d} \mathbb{E}\left[\|\mathcal{Q}(x)\|_0\right]$ and $\zeta_{\mathcal{C}} = \sup_{x\in\mathbb{R}^d} \mathbb{E}\left[\|\mathcal{C}(x)\|_0\right]$, where $\|y\|_0$ is the number of non-zero components of $y \in \mathbb{R}^d$.*

Notice that the expected density is well-defined for any compression operator since $\|\mathcal{Q}(x)\|_0 \leq d$ and $\|\mathcal{C}(x)\|_0 \leq d$.

The landscape of communication-efficient federated methods for non-smooth optimization is largely unexplored, with most research targeting smooth objectives or single-node settings. Below, we highlight open challenges and gaps.

Numerous works study s2w compression (Zheng et al., 2019; Gruntkowska et al., 2023; Fatkhullin et al., 2021; Philippenko & Dieuleveut, 2021; Liu et al., 2020; Gorbunov et al., 2020; Safaryan et al., 2022; Huang et al., 2022b; Horváth et al., 2022; Tang et al., 2019; Tyurin & Richtarik, 2023; Gruntkowska et al., 2024), yet almost all focus on smooth objectives. To our knowledge, only Karimireddy et al. (2019) and Anonymous (2024) offer non-smooth convex guarantees with s2w compression, and both are limited to single-node settings with minimal relevance to federated learning.

Distributed subgradient methods are well-studied, but either lack compression (Nedic & Ozdaglar, 2009; Kiwiel & Lindberg, 2001; Hishinuma & Iiduka, 2015; Zheng et al., 2022) or focus on specific operators like quantization (Xia et al., 2023; Doan et al., 2020; 2018; Xia et al., 2022; Emiola & Enyioha, 2022), covering only the w2s direction. No comprehensive treatments address s2w compression in non-smooth distributed optimization.

[3] Notably, it can easily be verified (see Lemma 8 in (Richtárik et al., 2021)) that if $\mathcal{Q} \in \mathbb{U}(\omega)$, then $(\omega+1)^{-1}\mathcal{Q} \in \mathbb{B}\left((\omega+1)^{-1}\right)$, indicating that the family of biased compressors is wider.

Method	Non-smooth	Distributed	Compressed communications	Compression type	Adaptive stepsizes
EF14 (Karimireddy et al., 2019)	✓	✗	✓	w2s	✗
EF21-P (Anonymous, 2024)	✓	✗	✓	s2w	✓
MARINA-P (Gruntkowska et al., 2024)	✗	✓	✓	s2w	✗
SM with Polyak Stepsize (Hazan & Kakade, 2019)	✓	✗	✗	-	✓
SM with Quantization (Xia et al., 2023)	✓	✓	✓	w2s	✗
EF21-P [OURS]	✓	✓	✓	s2w	✓
MARINA-P [OURS]	✓	✓	✓	s2w	✓

Table 1: Summary of optimization methods employing worker-to-server (w2s) or server-to-worker (s2w) compression schemes.

Recent works on adaptive stepsizes (Khaled et al., 2023; Defazio et al., 2023; 2024; Mishchenko & Defazio; Defazio & Mishchenko, 2023) primarily handle single-node problems. Polyak stepsizes (Polyak, 1987; Hazan & Kakade, 2019) remain popular, but current studies (Loizou et al., 2021; Oikonomou & Loizou, 2024; Jiang & Stich, 2024) often assume smoothness or lack distributed analysis. Even existing non-smooth convex results (Hazan & Kakade, 2019; Schaipp et al., 2023) remain restricted to single-node contexts.

In summary, the intersection of non-smooth optimization, communication efficiency, and federated learning remains underexplored. Our work addresses this gap by providing the first comprehensive study of distributed non-smooth optimization with s2w compression and adaptive stepsizes, while maintaining optimal convergence rates.

2 Contributions

We now summarize our main contributions: • **Extension of EF21-P to distributed non-smooth settings.** We generalize EF21-P (Anonymous, 2024) from single-node to distributed architectures, proving optimal $\mathcal{O}\left(1/\sqrt{T}\right)$ rates for both Polyak and constant stepsizes, and a suboptimal $\mathcal{O}\left(\log T/\sqrt{T}\right)$ rate for decreasing stepsizes. Our communication complexity matches classical distributed subgradient methods, addressing a longstanding gap in Error Feedback theory for non-smooth problems.

• **Introduction of MARINA-P for non-smooth objectives.** Building on Gruntkowska et al. (2024), we extend MARINA-P from smooth non-convex to non-smooth convex settings, establishing optimal $\mathcal{O}\left(1/\sqrt{T}\right)$ rates for constant and Polyak stepsizes, along with a suboptimal $\mathcal{O}\left(\log T/\sqrt{T}\right)$ rate under decreasing steps.

• **Superior performance of MARINA-P with correlated compressors.** Empirical results show that MARINA-P, when paired with correlated compressors, surpasses EF21-P in non-smooth settings. This extends the known benefits of correlated compressors, previously shown for smooth non-convex objectives, to non-smooth convex federated tasks.

• **Support for diverse stepsize schedules.** We provide theoretical guarantees for both algorithms under constant, decreasing, and Polyak stepsizes, bridging the gap between foundational theory and practical deep learning use cases.

To our knowledge, these are the first theoretical results for distributed non-smooth optimization incorporating s2w compression and adaptive stepsizes, while achieving provably optimal convergence rates.

Algorithm 1 EF21-P (distributed version)

Input: initial points $w^0 = x^0 \in \mathbb{R}^d$, stepsize $\gamma_0 > 0$
for $t = 0, 1, 2, \ldots, T$ **do**
  for $i = 1, \ldots, n$ **on Workers in parallel do**
    Receive compressed difference Δ^t from server
    Compute local subgradient $g_i^t = \partial f_i(w^t)$ and send it to server
  end for
  On Server:
  Receive g_i^t from workers
  Choose stepsize γ_t (can be set according to (9), (10), or (11))
  $x^{t+1} = x^t - \gamma_t \frac{1}{n} \sum_{i=1}^n g_i^t$
  Compute $\Delta^{t+1} = \mathcal{C}(x^{t+1} - w^t)$ and broadcast it to workers
  $w^{t+1} = w^t + \Delta^{t+1}$
  for $i = 1, \ldots, n$ **on Workers in parallel do**
    $w^{t+1} = w^t + \Delta^{t+1}$
  end for
end for
Output: x^T

3 EF21-P

We now present the first major contribution of our paper: a distributed version of EF21-P for the non-smooth setting.

Let us first recap the standard single-node EF21-P algorithm, which aims to solve (1) via the iterative process:

$$\begin{aligned} x^{t+1} &= x^t - \gamma_t \nabla f(w^t) \\ w^{t+1} &= w^t + \mathcal{C}^t \left(x^{t+1} - w^t\right), \end{aligned} \tag{7}$$

where $\gamma_t > 0$ is a stepsize, $x^0 \in \mathbb{R}^d$ is the initial iterate, $w^0 = x^0 \in \mathbb{R}^d$ is the initial iterate shift, and $\mathcal{C}^t$ is an instantiation of a randomized contractive compressor $\mathcal{C}$ sampled at time t. This method was proposed as a primal[4] counterpart to the standard EF21. It has proven particularly useful in bidirectional settings where primal compression is performed on the server side, allowing for the decoupling of primal and dual compression parameter constants. For more details, we refer the reader to the original paper (Gruntkowska et al., 2023). Anonymous (2024) first extended EF21-P to the non-smooth setting. Their key modification was replacing the ”smooth” update step with a ”non-smooth” one: $x^{t+1} = x^t - \gamma_t \partial f(w^t)$.

They proved an optimal rates of $\mathcal{O}\left({}^1/\sqrt{T}\right)$ for Polyak and constant stepsizes, and a suboptimal rate of $\mathcal{O}\left({}^{\log T}/\sqrt{T}\right)$ for decreasing stepsizes, but only for the single-node regime. In Algorithm 1, we extend these results to the distributed setting, allowing for parallel computation of subgradients $\partial f(w^t)$.

At each iteration of distributed EF21-P, the workers calculate $\partial f_i(w^t)$ and transmit it to the server. The server then averages the subgradients and updates the global model x^t. Subsequently, it computes the compressed difference $\Delta^{t+1} = \mathcal{C}_i^t(x^{t+1} - w^t)$ and broadcasts the same vector Δ^{t+1} to all workers. Both the server and workers then use Δ^{t+1} to update their internal states w^t. Note that this procedure ensures that the states w^t remain synchronized between workers and the server.

We now present the convergence result for our distributed EF21-P algorithm.

Theorem 1. *Let Assumptions 1, 2 and 3 hold. Define a Lyapunov function* $V^t := \|x^t - x^*\|_2^2 + \frac{1}{\lambda_* \theta}\|w^t - x^t\|_2^2$, *where* $\lambda_* := \frac{\sqrt{1-\alpha}}{1-\sqrt{1-\alpha}}$ *and* $\theta := 1 - \sqrt{1-\alpha}$. *Define also a constant* $B_* := 1 + 2\frac{\sqrt{1-\alpha}}{1-\sqrt{1-\alpha}}$. *Let* $\{w^t\}$ *be the sequence produced by* EF21-P *(Algorithm 1). Define* $\overline{w}^T := \frac{1}{T}\sum_{t=0}^{T-1} w^t$ *and* $\widehat{w}^T := \frac{1}{\sum_{t=0}^{T-1}\gamma_t}\sum_{t=0}^{T-1}\gamma_t w^t$.

[4] Since it operates in the primal space of model parameters

1. (Constant stepsize). *If $\gamma_t := \gamma > 0$, then*

$$\mathbb{E}\left[f(\overline{w}^T) - f(x^*)\right] \leq \frac{V^0}{2\gamma T} + \frac{B_* L_0^2 \gamma}{2}. \tag{8}$$

If, moreover, optimal γ is chosen i.e.

$$\gamma := \frac{1}{\sqrt{T}} \sqrt{\frac{V^0}{B_* L_0^2}}, \tag{9}$$

then $\mathbb{E}\left[f(\overline{w}^T) - f(x^*)\right] \leq \frac{\sqrt{B_* L_0^2 V^0}}{\sqrt{T}}$.

2. Polyak stepsize. *If γ_t is chosen as*

$$\gamma_t \quad := \quad \frac{f(w^t) - f(x^*)}{B_* \left\|\partial f\left(w^t\right)\right\|_2^2}, \tag{10}$$

then $\mathbb{E}\left[f(\overline{w}^T) - f(x^*)\right] \leq \frac{\sqrt{B_* L_0^2 V^0}}{\sqrt{T}}$.

3. (Decreasing stepsize). *If γ_t is chosen as*

$$\gamma_t := \frac{\gamma_0}{\sqrt{t+1}}, \tag{11}$$

then $\mathbb{E}\left[f(\widehat{w}^T) - f(x^*)\right] \leq \frac{V^0 + 2\gamma_0^2 B_* L_0^2 \log(T+1)}{\gamma_0 \sqrt{T}}$.

If, moreover, optimal γ_0 is chosen i.e.

$$\gamma_0 := \sqrt{\frac{V_0}{2B_* L_0^2 \log(T+1)}}, \tag{12}$$

then $\mathbb{E}\left[f(\widehat{w}^T) - f(x^*)\right] \leq 2\sqrt{2B_* L_0^2 V_0}\sqrt{\frac{\log(T+1)}{T}}$.

Let us analyze the obtained results. The constant $B_* := 1 + 2\frac{\sqrt{1-\alpha}}{1-\sqrt{1-\alpha}} \leq \frac{4}{\alpha} - 1$ is a decreasing function in α, which aligns with intuition since larger values of α correspond to less aggressive compression regimes. For both constant (9) and Polyak (10) stepsizes, we achieve the optimal rate of $\mathcal{O}\left({}^{1}/\sqrt{T}\right)$ known for uncompressed subgradient methods (Nesterov, 2013; Arjevani & Shamir, 2015). However, achieving this rate requires either knowing the total number of iterations T in advance (for constant stepsize) or knowing the optimal value $f(x^*)$ (for Polyak stepsize), which may be impractical in many applications. When neither T nor $f(x^*)$ is known, one can employ the decreasing stepsize strategy (11). This approach leads to a suboptimal convergence rate of $\mathcal{O}\left({}^{\log T}/\sqrt{T}\right)$ – a well-known limitation of subgradient methods (Nesterov, 2013; Anonymous, 2024).

For both constant and Polyak stepsizes, the following corollary provides explicit complexity bounds, characterizing both the number of iterations and the total communication cost needed to obtain an ε-approximate solution.

Corollary 1. *Let the conditions of the Theorem 1 are met. If γ_t is set according to* (9) *or* (10) *(constant or Polyak stepsizes) then* EF21-P *(Algorithm 1) requires $T = \mathcal{O}\left(\frac{L_0^2 R_0^2}{\alpha \varepsilon^2}\right)$ iterations/communication rounds in order to achieve $\mathbb{E}\left[f(\overline{w}^T) - f(x^*)\right] \leq \varepsilon$, and the expected total communication cost per worker is $\mathcal{O}\left(d + \zeta_{\mathcal{C}} T\right)$.*

Let us analyze the implications of Corollary 1. In the uncompressed case ($\alpha = 1$), our algorithm achieves the optimal rate of standard Subgradient Methods (SM) (Nesterov, 2013) for first-order non-smooth optimization. With Top$K$ compression ($\zeta_{\mathcal{C}} = K$), the communication complexity becomes $\mathcal{O}\left({}^{dL_0^2 R_0^2}/{\varepsilon^2}\right)$, matching the worst-case complexity of distributed SM. This indicates that EF21-P with Top$K$ compression cannot improve upon SM's complexity regardless of the compression parameter α – a fundamental limitation in communication-compressed non-smooth optimization. Our findings align with Balkanski & Singer (2018), who demonstrated that parallelization provides no worst-case benefits for non-smooth optimization.

From a practical perspective, EF21-P's main limitation stems from broadcasting identical compressed differences Δ_t to all workers, potentially leading to poor approximations of x^{t+1} by $w^t + \Delta_t$. The MARINA-P algorithm (Gruntkowska et al., 2024), originally developed for smooth non-convex problems, addresses this limitation. In the following section, we extend MARINA-P to the non-smooth setting.

Algorithm 2 MARINA-P

Input: initial point $x^0 \in \mathbb{R}^d$, initial model shifts $w_i^0 = x^0$ for all $i \in [n]$, stepsize $\gamma_0 > 0$, probability $0 < p \leq 1$, compressors $\mathcal{Q}_i^t \in \mathbb{U}(\omega)$ for all $i \in [n]$
for $t = 0, 1, \ldots, T$ **do**
  for $i = 1, \ldots, n$ **on Workers in parallel do**
    Compute local subgradient $g_i^t = \partial f_i(w_i^t)$ and send it to server
  end for
  On Server:
  Receive g_i^t from workers
  Choose stepsize γ_t (can be set according to (13), (14), or (15))
  $x^{t+1} = x^t - \gamma_t \frac{1}{n} \sum_{i=1}^n g_i^t$
  Sample $c^t \sim \text{Bernoulli}(p)$
  if $c^t = 0$ **then**
    Send $\mathcal{Q}_i^t(x^{t+1} - x^t)$ to worker i for $i \in [n]$
  else
    Send x^{t+1} to all workers
  end if
  for $i = 1, \ldots, n$ **on Workers in parallel do**
    $w_i^{t+1} = \begin{cases} x^{t+1} & \text{if } c^t = 1 \\ w_i^t + \mathcal{Q}_i^t(x^{t+1} - x^t) & \text{if } c^t = 0 \end{cases}$
  end for
end for
Output: x^T

4 MARINA-P

Building upon the foundations of the standard MARINA algorithm (Gorbunov et al., 2021; Szlendak et al., 2022), Gruntkowska et al. (2024) introduced MARINA-P, a primal counterpart designed to operate in the model parameter space. This section presents an extension of MARINA-P to the non-smooth convex setting. At each iteration, workers compute subgradients $\partial f_i(w_i^t)$ and transmit them to the server. The server aggregates these subgradients and updates the global model x^t. With probability p (typically small), the server sends the uncompressed updated model x^{t+1} to all workers. Otherwise, each worker i receives a compressed vector $\mathcal{Q}_i^t(x^{t+1} - x^t)$. Workers then update their local models w_i^{t+1} based on the received information. A key feature of MARINA-P is that the compressed vectors $\mathcal{Q}_1^t(x^{t+1} - x^t), \ldots, \mathcal{Q}_n^t(x^{t+1} - x^t)$ can differ across workers. This distinction is crucial for the algorithm's practical superiority, as it allows for potentially better approximations of x^{t+1} compared to methods like EF21-P, especially when the compressors $\mathcal{Q}_1, \ldots, \mathcal{Q}_n$ are correlated.

We now present the main convergence results for MARINA-P in the non-smooth convex setting.

Theorem 2. *Let Assumptions 1, 2 and 3 hold. Define a Lyapunov function* $V^t := \|x^t - x^*\|_2^2 + \frac{1}{\lambda_* p} \frac{1}{n} \sum_{i=1}^n \|w_i^t - x^t\|_2^2$*, where* $\lambda_* := \frac{\overline{L}_0}{\widetilde{L}_0} \sqrt{\frac{(1-p)\omega}{p}}$*. Define also a constant* $\widetilde{B}_* := \overline{L}_0^2 + 2\overline{L}_0 \widetilde{L}_0 \sqrt{\frac{(1-p)\omega}{p}}$*. Let* $\{w_i^t\}$ *be the sequence produced by* MARINA-P *(Algorithm 2). Define* $\overline{w}_i^T := \frac{1}{T} \sum_{t=0}^{T-1} w_i^t$ *and* $\widehat{w}_i^T := \frac{1}{\sum_{t=0}^{T-1} \gamma_t} \sum_{t=0}^{T-1} \gamma_t w_i^t$ *for all* $i \in [n]$.

1. (Constant stepsize). *If* $\gamma_t := \gamma > 0$*, then* $\mathbb{E}\left[\frac{1}{n} \sum_{i=1}^n f_i(\overline{w}_i^T) - f(x^*)\right] \leq \frac{V^0}{2\gamma T} + \frac{\widetilde{B}_* \gamma}{2}$.

If, moreover, optimal γ *is chosen i.e.*

$$\gamma := \frac{1}{\sqrt{T}} \sqrt{\frac{V^0}{\widetilde{B}_*}}, \tag{13}$$

then $\mathbb{E}\left[\frac{1}{n} \sum_{i=1}^n f_i(\overline{w}_i^T) - f(x^*)\right] \leq \frac{\sqrt{\widetilde{B}_* V^0}}{\sqrt{T}}$.

2. Polyak stepsize. *If* γ_t *is chosen as*

$$\gamma_t := \frac{\frac{1}{n} \sum_{i=1}^n f_i(w_i^t) - f(x^*)}{\left\|\frac{1}{n} \sum_{i=1}^n \partial f_i\left(w_i^t\right)\right\|_2^2 \left(1 + 2\frac{\sqrt{\frac{1}{n} \sum_{i=1}^n \|\partial f_i(w_i^t)\|_2^2}}{\left\|\frac{1}{n} \sum_{i=1}^n \partial f_i\left(w_i^t\right)\right\|_2} \sqrt{\frac{(1-p)\omega}{p}}\right)}, \tag{14}$$

then $\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^n f_i(\overline{w}_i^T) - f(x^*)\right] \le \frac{\sqrt{\widetilde{B}_* V^0}}{\sqrt{T}}$.

3. (Decreasing stepsize). *If γ_t is chosen as*

$$\gamma_t := \frac{\gamma_0}{\sqrt{t+1}}, \tag{15}$$

then $\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^n f_i(\widehat{w}_i^T) - f(x^*)\right] \le \frac{V^0 + 2\gamma_0^2 \widetilde{B}_* \log(T+1)}{\gamma_0\sqrt{T}}$.

If, moreover, optimal γ_0 is chosen i.e.

$$\gamma_0 := \sqrt{\frac{V_0}{2\widetilde{B}_* \log(T+1)}}, \tag{16}$$

then $\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^n f_i(\widehat{w}_i^T) - f(x^*)\right] \le 2\sqrt{2\widetilde{B}_* V_0}\sqrt{\frac{\log(T+1)}{T}}$.

Remark 1. *For both EF21-P and MARINA-P, the Polyak stepsize can be efficiently implemented in the distributed setting without additional per-iteration communication overhead. This is because the subgradient values $\partial f_i(w^t)$ (for EF21-P) and $\partial f_i(w_i^t)$ (for MARINA-P) are already computed by the clients and transmitted to the server as part of the algorithm's regular operations.*

Let us analyze these results. The constant $\widetilde{B}_* := \overline{L}_0^2 + 2\overline{L}_0\widetilde{L}_0\sqrt{\frac{(1-p)\omega}{p}}$ depends on both compression parameters ω and p. Smaller values of ω correspond to less aggressive compression, while larger values of p indicate more frequent uncompressed communication – both cases lead to smaller $\widetilde{B}_*$ and consequently faster convergence. For both constant (13) and Polyak (14) stepsizes, we obtain the optimal rate of $\mathcal{O}\left(1/\sqrt{T}\right)$ (Nesterov, 2013; Arjevani & Shamir, 2015). As with EF21-P, achieving this rate requires either knowing the total iterations T (for constant stepsize) or the optimal value $f(x^*)$ (for Polyak stepsize) in advance. When such knowledge is unavailable, the decreasing stepsize strategy offers a practical alternative, though it results in a suboptimal $\mathcal{O}\left(\log T/\sqrt{T}\right)$ convergence rate – a characteristic limitation of subgradient methods (Nesterov, 2013). It is worth noting that implementing the Polyak stepsize only requires an estimate of $f(x^*)$, rather than knowledge of the Lipschitz constant L_0. This characteristic is common among Polyak stepsizes (Loizou et al., 2021).

For the constant and Polyak stepsize regimes, the following corollary establishes complexity bounds and characterizes the communication costs required to achieve an ε-approximate solution.

Corollary 2. *Let the conditions of the Theorem 2 are met and $p = \zeta_{\mathcal{Q}}/d$. If γ_t is set according to* (13) *or* (14) *(constant or Polyak stepsizes) then MARINA-P (Algorithm 2) requires $T = \mathcal{O}\left(\frac{R_0^2}{\varepsilon^2}\left(\overline{L}_0^2 + \overline{L}_0\widetilde{L}_0\sqrt{\omega\left(\frac{d}{\zeta_{\mathcal{Q}}} - 1\right)}\right)\right)$ iterations/communication rounds in order to achieve $\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^n f_i(\overline{w}_i^T) - f(x^*)\right] \le \varepsilon$, and the expected total communication cost per worker is $\mathcal{O}(d + \zeta_{\mathcal{Q}} T)$.*

This corollary reveals several important properties. With Rand$K$ compression ($\zeta_{\mathcal{Q}} = K, \omega = d/K - 1$) (Beznosikov et al., 2023), MARINA-P achieves communication complexity $\mathcal{O}\left(d\widetilde{L}_0^2 R_0^2/\varepsilon^2\right)$. Under the condition $\widetilde{L}_0^2 = \mathcal{O}(L_0)$, this matches the optimal per-worker complexity of standard SM, up to constant factors (Nesterov, 2013). A notable feature of our complexity result is its independence from the number of workers n in the non-smooth setting – a known phenomenon in subgradient methods (Arjevani & Shamir, 2015; Balkanski & Singer, 2018). This contrasts with MARINA-P's behavior in smooth non-convex settings (Gruntkowska et al., 2024), where complexity scales as $\mathcal{O}(1/n)$. The absence of theoretical bounds predicting such scaling behavior in non-smooth distributed settings presents an interesting direction for future research.

MARINA-P's primary advantage over EF21-P lies in its ability to employ worker-specific compression operators $\mathcal{Q}_i$, enabling more accurate approximations of the global model, particularly when using correlated compressors. The following section examines various constructions of $\mathcal{Q}_i$ that leverage this flexibility to enhance practical performance.

5 Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

REFERENCES

Alham Fikri Aji and Kenneth Heafield. Sparse communication for distributed gradient descent. *arXiv preprint arXiv:1704.05021*, 2017.

Dan Alistarh, Demjan Grubic, Jerry Li, Ryota Tomioka, and Milan Vojnovic. QSGD: Communication-efficient SGD via gradient quantization and encoding. *Advances in neural information processing systems*, 30, 2017.

Dan Alistarh, Torsten Hoefler, Mikael Johansson, Sarit Khirirat, Nikola Konstantinov, and Cédric Renggli. The convergence of sparsified gradient methods. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018a.

Dan Alistarh, Torsten Hoefler, Mikael Johansson, Nikola Konstantinov, Sarit Khirirat, and Cédric Renggli. The convergence of sparsified gradient methods. *Advances in Neural Information Processing Systems*, 31, 2018b.

Anonymous. Error feedback for smooth and nonsmooth convex optimization with constant, decreasing and polyak stepsizes. In *Submitted to The Thirteenth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=Qv9TG9yDG0. under review.

Yossi Arjevani and Ohad Shamir. Communication complexity of distributed convex learning and optimization. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015. URL https://proceedings.neurips.cc/paper_files/paper/2015/file/7fec306d1e665bc9c748b5d2b99a6e97-[]Paper.pdf.

Francis Bach. Learning with submodular functions: A convex optimization perspective. *Foundations and Trends® in machine learning*, 6(2-3):145–373, 2013.

Eric Balkanski and Yaron Singer. Parallelization does not accelerate convex optimization: Adaptivity lower bounds for non-smooth convex minimization. *arXiv preprint arXiv:1808.03880*, 2018.

Amir Beck. *First-order methods in optimization*. SIAM, 2017.

Aleksandr Beznosikov, Samuel Horváth, Peter Richtárik, and Mher Safaryan. On biased compression for distributed learning. *Journal of Machine Learning Research*, 24(276):1–50, 2023.

Leon Bottou. *Stochastic Gradient Descent Tricks*, volume 7700 of *Lecture Notes in Computer Science (LNCS)*, pp. 430–445. Springer, neural networks, tricks of the trade, reloaded edition, January 2012. URL https://www.microsoft.com/en-[]us/research/publication/stochastic-[]gradient-[]tricks/.

Léon Bottou, Frank Curtis, and Jorge Nocedal. Optimization methods for large-scale machine learning. *SIAM Review*, 60:223–311, 2018.

Sébastien Bubeck. Convex optimization: Algorithms and complexity. *Foundations and Trends® in Machine Learning*, 8(3-4):231–357, 2015.

Antonin Chambolle. An algorithm for total variation minimization and applications. *Journal of Mathematical imaging and vision*, 20:89–97, 2004.

Jean-Baptiste Cordonnier. Convex optimization using sparsified stochastic gradient descent with memory. 2018.

Corinna Cortes. Support-vector networks. *Machine Learning*, 1995.

Rudrajit Das, Abolfazl Hashemi, Sujay Sanghavi, and Inderjit S Dhillon. Improved convergence rates for non-convex federated learning with compression. *arXiv preprint arXiv:2012.04061*, 2020.

Jeffrey Dean, Greg Corrado, Rajat Monga, Kai Chen, Matthieu Devin, Mark Mao, Andrew Senior, Paul Tucker, Ke Yang, Quoc V Le, and et al. Large scale distributed deep networks. In *Advances in Neural Information Processing Systems*, pp. 1223–1231, 2012.

Aaron Defazio and Konstantin Mishchenko. Learning-rate-free learning by d-adaptation. In *International Conference on Machine Learning*, pp. 7449–7479. PMLR, 2023.

Aaron Defazio, Ashok Cutkosky, Harsh Mehta, and Konstantin Mishchenko. When, why and how much? adaptive learning rate scheduling by refinement. *arXiv preprint arXiv:2310.07831*, 2023.

Aaron Defazio, Xingyu Alice Yang, Harsh Mehta, Konstantin Mishchenko, Ahmed Khaled, and Ashok Cutkosky. The road less scheduled. *arXiv preprint arXiv:2405.15682*, 2024.

Yury Demidovich, Grigory Malinovsky, Igor Sokolov, and Peter Richtárik. A guide through the zoo of biased SGD. *Advances in Neural Information Processing Systems*, 36:23158–23171, 2023.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Thinh T Doan, Siva Theja Maguluri, and Justin Romberg. On the convergence of distributed subgradient methods under quantization. In *2018 56th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 567–574. IEEE, 2018.

Thinh T Doan, Siva Theja Maguluri, and Justin Romberg. Fast convergence rates of distributed subgradient methods with adaptive quantization. *IEEE Transactions on Automatic Control*, 66(5): 2191–2205, 2020.

Dmitriy Drusvyatskiy. Convex analysis and nonsmooth optimization. *University Lecture*, 2020.

Nikoli Dryden, Tim Moon, Sam Ade Jacobs, and Brian Van Essen. Communication quantization for data-parallel training of deep neural networks. In *2016 2nd Workshop on Machine Learning in HPC Environments (MLHPC)*, pp. 1–8, 2016. doi: 10.1109/MLHPC.2016.004.

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7), 2011.

John C Duchi. Introductory lectures on stochastic optimization. *The mathematics of data*, 25:99–186, 2018.

Iyanuoluwa Emiola and Chinwendu Enyioha. Quantized and distributed subgradient optimization method with malicious attack. *IEEE Control Systems Letters*, 7:181–186, 2022.

Ilyas Fatkhullin, Igor Sokolov, Eduard Gorbunov, Zhize Li, and Peter Richtárik. EF21 with bells & whistles: Practical algorithmic extensions of modern error feedback. *arXiv preprint arXiv:2110.03294*, 2021.

Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pp. 315–323. JMLR Workshop and Conference Proceedings, 2011.

WM Goodall. Television by pulse code modulation. *Bell System Technical Journal*, 30(1):33–49, 1951.

Eduard Gorbunov, Dmitry Kovalev, Dmitry Makarenko, and Peter Richtárik. Linearly converging error compensated SGD. *Advances in Neural Information Processing Systems*, 33:20889–20900, 2020.

Eduard Gorbunov, Konstantin P Burlachenko, Zhize Li, and Peter Richtárik. MARINA: Faster non-convex distributed learning with compression. In *International Conference on Machine Learning*, pp. 3788–3798. PMLR, 2021.

Kaja Gruntkowska, Alexander Tyurin, and Peter Richtárik. EF21-P and friends: Improved theoretical communication complexity for distributed optimization with bidirectional compression. In *International Conference on Machine Learning*, pp. 11761–11807. PMLR, 2023.

Kaja Gruntkowska, Alexander Tyurin, and Peter Richtárik. Improving the worst-case bidirectional communication complexity for nonconvex distributed optimization under function similarity. *arXiv preprint arXiv:2402.06412*, 2024.

Farzin Haddadpour, Mohammad Mahdi Kamani, Aryan Mokhtari, and Mehrdad Mahdavi. Federated learning with compression: Unified analysis and sharp guarantees. In *International Conference on Artificial Intelligence and Statistics*, pp. 2350–2358. PMLR, 2021.

David Hallac, Jure Leskovec, and Stephen Boyd. Network lasso: Clustering and optimization in large graphs. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 387–396, 2015.

Trevor Hastie, Robert Tibshirani, Jerome H Friedman, and Jerome H Friedman. *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer, 2009.

Elad Hazan and Sham Kakade. Revisiting the Polyak step size. *arXiv preprint arXiv:1905.00313*, 2019.

Elad Hazan and Satyen Kale. Beyond the regret minimization barrier: optimal algorithms for stochastic strongly-convex optimization. *The Journal of Machine Learning Research*, 15(1): 2489–2512, 2014.

Elad Hazan, Amit Agarwal, and Satyen Kale. Logarithmic regret algorithms for online convex optimization. *Machine Learning*, 69(2):169–192, 2007.

Kazuhiro Hishinuma and Hideaki Iiduka. Parallel subgradient method for nonsmooth convex optimization with a simple constraint. *Linear Nonlinear Anal*, 1:67–77, 2015.

Samuel Horváth, Dmitry Kovalev, Konstantin Mishchenko, Sebastian Stich, and Peter Richtárik. Stochastic distributed learning with gradient quantization and variance reduction. *arXiv preprint arXiv:1904.05115*, 2019.

Samuel Horváth, Chen-Yu Ho, Ludovit Horváth, Atal Narayan Sahu, Marco Canini, and Peter Richtárik. Natural compression for distributed deep learning. In *Mathematical and Scientific Machine Learning*, pp. 129–141. PMLR, 2022.

Junxian Huang, Feng Qian, Alexandre Gerber, Z Morley Mao, Subhabrata Sen, and Oliver Spatscheck. A close examination of performance and power characteristics of 4g lte networks. In *Proceedings of the 10th international conference on Mobile systems, applications, and services*, pp. 225–238, 2012.

Xinmeng Huang, Yiming Chen, Wotao Yin, and Kun Yuan. Lower bounds and nearly optimal algorithms in distributed learning with communication compression. *Advances in Neural Information Processing Systems*, 35:18955–18969, 2022a.

Xinmeng Huang, Yiming Chen, Wotao Yin, and Kun Yuan. Lower bounds and nearly optimal algorithms in distributed learning with communication compression. *Advances in Neural Information Processing Systems*, 35:18955–18969, 2022b.

Peter J Huber. Robust estimation of a location parameter. *The Annals of Mathematical Statistics*, pp. 73–101, 1964.

Nikita Ivkin, Daniel Rothchild, Enayat Ullah, Vladimir braverman, Ion Stoica, and Raman Arora. Communication-efficient distributed SGD with sketching. *Advances in Neural Information Processing Systems*, 32, 2019.

Prateek Jain, Dheeraj Nagaraj, and Praneeth Netrapalli. Making the last iterate of SGD information theoretically optimal. In *Conference on Learning Theory*, pp. 1752–1755. PMLR, 2019.

Xiaowen Jiang and Sebastian U Stich. Adaptive SGD with polyak stepsize and line-search: Robust convergence and variance reduction. *Advances in Neural Information Processing Systems*, 36, 2024.

Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *Foundations and trends® in machine learning*, 14(1–2):1–210, 2021.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

Sai Praneeth Karimireddy, Quentin Rebjock, Sebastian Stich, and Martin Jaggi. Error feedback fixes signsgd and other gradient compression schemes. In *International Conference on Machine Learning*, pp. 3252–3261. PMLR, 2019.

Ahmed Khaled, Konstantin Mishchenko, and Peter Richtárik. Tighter theory for local SGD on identical and heterogeneous data. In *The 23rd International Conference on Artificial Intelligence and Statistics (AISTATS 2020)*, 2020.

Ahmed Khaled, Konstantin Mishchenko, and Chi Jin. Dowg unleashed: An efficient universal parameter-free gradient descent method. *Advances in Neural Information Processing Systems*, 36: 6748–6769, 2023.

Sarit Khirirat, Hamid Reza Feyzmahdavian, and Mikael Johansson. Distributed learning with compressed gradients. *arXiv preprint arXiv:1806.06573*, 2018.

Diederik P. Kingma and Jimmy Ba. Adam: a method for stochastic optimization. In *The 3rd International Conference on Learning Representations*, 2014. URL https://arxiv.org/pdf/1412.6980.pdf.

KC Kiwiel and PO Lindberg. Parallel subgradient methods for convex optimization. In *Studies in Computational Mathematics*, volume 8, pp. 335–344. Elsevier, 2001.

Roger Koenker and Gilbert Bassett Jr. Regression quantiles. *Econometrica: journal of the Econometric Society*, pp. 33–50, 1978.

Anastasia Koloskova, Nicolas Loizou, Sadra Boreiri, Martin Jaggi, and Sebastian Stich. A unified theory of decentralized SGD with changing topology and local updates. In *International Conference on Machine Learning*, pp. 5381–5393. PMLR, 2020.

Jakub Konečnỳ and Peter Richtárik. Randomized distributed mean estimation: Accuracy vs. communication. *Frontiers in Applied Mathematics and Statistics*, 4:62, 2018.

Jakub Konečný, H. Brendan McMahan, Daniel Ramage, and Peter Richtárik. Federated optimization: distributed machine learning for on-device intelligence. *arXiv:1610.02527*, 2016a.

Jakub Konečný, H. Brendan McMahan, Felix Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon. Federated learning: strategies for improving communication efficiency. In *NIPS Private Multi-Party Machine Learning Workshop*, 2016b.

Simon Lacoste-Julien, Mark Schmidt, and Francis Bach. A simpler approach to obtaining an o (1/t) convergence rate for the projected stochastic subgradient method. *arXiv preprint arXiv:1212.2002*, 2012.

Guanghui Lan. *First-order and stochastic optimization methods for machine learning*, volume 1. Springer, 2020.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

Zhize Li and Peter Richtárik. CANITA: Faster rates for distributed convex optimization with communication compression. *Advances in Neural Information Processing Systems*, 34:13770–13781, 2021.

Zhize Li, Dmitry Kovalev, Xun Qian, and Peter Richtárik. Acceleration for compressed gradient descent in distributed and federated optimization. In *International Conference on Machine Learning (ICML)*, pp. 5895–5904. PMLR, 2020.

Yujun Lin, Song Han, Huizi Mao, Yu Wang, and Bill Dally. Deep gradient compression: Reducing the communication bandwidth for distributed training. In *International Conference on Learning Representations*, 2018.

Xiaorui Liu, Yao Li, Jiliang Tang, and Ming Yan. A double residual compression algorithm for efficient distributed learning. In *International Conference on Artificial Intelligence and Statistics*, pp. 133–143. PMLR, 2020.

Nicolas Loizou, Sharan Vaswani, Issam Hadj Laradji, and Simon Lacoste-Julien. Stochastic Polyak step-size for SGD: An adaptive learning rate for fast convergence. In *International Conference on Artificial Intelligence and Statistics*, pp. 1306–1314. PMLR, 2021.

Brendan McMahan, Eider Moore, Daniel Ramage, and Blaise Agüera y Arcas. Federated learning of deep networks using model averaging. *arXiv preprint arXiv:1602.05629*, 2016.

H Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2017.

Konstantin Mishchenko and Aaron Defazio. Prodigy: An expeditiously adaptive parameter-free learner. In *Forty-first International Conference on Machine Learning*.

Konstantin Mishchenko, Eduard Gorbunov, Martin Takáč, and Peter Richtárik. Distributed learning with compressed gradient differences. *arXiv preprint arXiv:1901.09269*, 2019.

Konstantin Mishchenko, Filip Hanzely, and Peter Richtárik. 99% of worker-master communication in distributed optimization is not needed. In *Conference on Uncertainty in Artificial Intelligence*, pp. 979–988. PMLR, 2020.

Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pp. 807–814, 2010.

Arvind Narayanan, Xumiao Zhang, Ruiyang Zhu, Ahmad Hassan, Shuowei Jin, Xiao Zhu, Xiaoxuan Zhang, Denis Rybkin, Zhengxuan Yang, Zhuoqing Morley Mao, et al. A variegated look at 5g in the wild: performance, power, and qoe implications. In *Proceedings of the 2021 ACM SIGCOMM 2021 Conference*, pp. 610–625, 2021.

Angelia Nedic and Asuman Ozdaglar. Distributed subgradient methods for multi-agent optimization. *IEEE Transactions on Automatic Control*, 54(1):48–61, 2009.

Arkadi Nemirovski, Anatoli Juditsky, Guanghui Lan, and Alexander Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on optimization*, 19(4):1574–1609, 2009.

Yurii Nesterov. *Introductory lectures on convex optimization: A basic course*, volume 87. Springer Science & Business Media, 2013.

Dimitris Oikonomou and Nicolas Loizou. Stochastic polyak step-sizes and momentum: Convergence guarantees and practical performance. *arXiv preprint arXiv:2406.04142*, 2024.

Andrei Panferov, Yury Demidovich, Ahmad Rammal, and Peter Richtárik. Correlated quantization for faster nonconvex distributed optimization. *arXiv preprint arXiv:2401.05518*, 2024.

Constantin Philippenko and Aymeric Dieuleveut. Preserved central model for faster bidirectional compression in distributed settings. *Advances in Neural Information Processing Systems*, 34: 2387–2399, 2021.

Boris T Polyak. Introduction to optimization. 1987.

Alexander Rakhlin, Ohad Shamir, and Karthik Sridharan. Making gradient descent optimal for strongly convex stochastic optimization. *arXiv preprint arXiv:1109.5647*, 2011.

Nathan D Ratliff, J Andrew Bagnell, and Martin A Zinkevich. (approximate) subgradient methods for structured prediction. In *Artificial Intelligence and Statistics*, pp. 380–387. PMLR, 2007.

Sashank J Reddi, Satyen Kale, and Sanjiv Kumar. On the convergence of adam and beyond. In *International Conference on Learning Representations*, 2018.

Amirhossein Reisizadeh, Aryan Mokhtari, Hamed Hassani, Ali Jadbabaie, and Ramtin Pedarsani. Fedpaq: A communication-efficient federated learning method with periodic averaging and quantization. In *International conference on artificial intelligence and statistics*, pp. 2021–2031. PMLR, 2020.

Peter Richtárik, Igor Sokolov, and Ilyas Fatkhullin. EF21: A new, simpler, theoretically better, and practically faster error feedback. *Advances in Neural Information Processing Systems*, 34: 4384–4396, 2021.

Peter Richtárik, Igor Sokolov, Elnur Gasanov, Ilyas Fatkhullin, Zhize Li, and Eduard Gorbunov. 3PC: Three point compressors for communication-efficient distributed training and a better theory for lazy aggregation. In *International Conference on Machine Learning*, pp. 18596–18648. PMLR, 2022.

Peter Richtárik, Abdurakhmon Sadiev, and Yury Demidovich. A unified theory of stochastic proximal point methods without smoothness. *arXiv preprint arXiv:2405.15941*, 2024.

Herbert Robbins and Sutton Monro. A stochastic approximation method. *The annals of mathematical statistics*, pp. 400–407, 1951.

Lawrence Roberts. Picture coding using pseudo-random noise. *IRE Transactions on Information Theory*, 8(2):145–154, 1962.

Daniel Rothchild, Ashwinee Panda, Enayat Ullah, Nikita Ivkin, Ion Stoica, Vladimir Braverman, Joseph Gonzalez, and Raman Arora. FetchSGD: Communication-efficient federated learning with sketching. In *International Conference on Machine Learning*, pp. 8253–8265. PMLR, 2020.

Leonid I Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: nonlinear phenomena*, 60(1-4):259–268, 1992.

Abdurakhmon Sadiev, Grigory Malinovsky, Eduard Gorbunov, Igor Sokolov, Ahmed Khaled, Konstantin Burlachenko, and Peter Richtárik. Federated optimization algorithms with random reshuffling and gradient compression. *arXiv preprint arXiv:2206.07021*, 2022.

Mher Safaryan, Rustem Islamov, Xun Qian, and Peter Richtárik. FedNL: Making Newton-type methods applicable to federated learning. *arXiv preprint arXiv:2106.02969*, 2021.

Mher Safaryan, Egor Shulgin, and Peter Richtárik. Uncertainty principle for communication compression in distributed and federated learning and the search for an optimal compressor. *Information and Inference: A Journal of the IMA*, 11(2):557–580, 2022.

Fabian Schaipp, Robert M. Gower, and Michael Ulbrich. A stochastic proximal polyak step size. *Transactions on Machine Learning Research*, 2023.

Dominik Scherer, Andreas Müller, and Sven Behnke. Evaluation of pooling operations in convolutional architectures for object recognition. In *International conference on artificial neural networks*, pp. 92–101. Springer, 2010.

Frank Seide, Hao Fu, Jasha Droppo, Gang Li, and Dong Yu. 1-bit stochastic gradient descent and its application to data-parallel distributed training of speech DNNs. In *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.

Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

Shai Shalev-Shwartz, Yoram Singer, and Nathan Srebro. Pegasos: Primal estimated sub-gradient solver for svm. In *Proceedings of the 24th international conference on Machine learning*, pp. 807–814, 2007.

N. Z. Shor, Krzysztof C. Kiwiel, and Andrzej Ruszczyński. Minimization methods for non-differentiable functions, 1985.

Sebastian U. Stich. Local SGD converges fast and communicates little. In *International Conference on Learning Representations*, 2020.

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

Sebastian U Stich, Jean-Baptiste Cordonnier, and Martin Jaggi. Sparsified SGD with memory. *Advances in neural information processing systems*, 31, 2018.

Nikko Ström. Scalable distributed DNN training using commodity GPU cloud computing. In *Interspeech 2015*, 2015.

Rafał Szlendak, Alexander Tyurin, and Peter Richtárik. Permutation compressors for provably faster distributed nonconvex optimization. In *International Conference on Learning Representations*, 2022.

Hanlin Tang, Chen Yu, Xiangru Lian, Tong Zhang, and Ji Liu. Doublesqueeze: Parallel stochastic gradient descent with double-pass error-compensated compression. In *International Conference on Machine Learning*, pp. 6155–6165. PMLR, 2019.

Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 58(1):267–288, 1996.

Alexander Tyurin and Peter Richtárik. DASHA: Distributed nonconvex optimization with communication compression and optimal oracle complexity. In *The Eleventh International Conference on Learning Representations*, 2023.

Alexander Tyurin and Peter Richtarik. 2Direction: Theoretically faster distributed training with bidirectional communication compression. *Advances in Neural Information Processing Systems*, 36:11737–11808, 2023.

Vladimir Vapnik. *The nature of statistical learning theory*. Springer science & business media, 2013.

Thijs Vogels, Sai Praneeth Karimireddy, and Martin Jaggi. PowerSGD: Practical low-rank gradient compression for distributed optimization. *Advances in Neural Information Processing Systems*, 32, 2019.

Thijs Vogels, Sai Praneeth Karimireddy, and Martin Jaggi. Practical low-rank communication compression in decentralized deep learning. *Advances in Neural Information Processing Systems*, 33:14171–14181, 2020.

Evgeniya Vorontsova, Roland Hildebrand, Alexander Gasnikov, and Fedor Stonyakin. Convex optimization. *arXiv preprint arXiv:2106.01946*, 2021.

Hongyi Wang, Scott Sievert, Shengchao Liu, Zachary Charles, Dimitris Papailiopoulos, and Stephen Wright. Atomo: Communication-efficient learning via atomic sparsification. *Advances in neural information processing systems*, 31, 2018.

Jianqiao Wangni, Jialei Wang, Ji Liu, and Tong Zhang. Gradient sparsification for communication-efficient distributed optimization. *Advances in Neural Information Processing Systems*, 31, 2018.

Wei Wen, Cong Xu, Feng Yan, Chunpeng Wu, Yandan Wang, Yiran Chen, and Hai Li. Terngrad: Ternary gradients to reduce communication in distributed deep learning. *Advances in neural information processing systems*, 30, 2017.

Blake Woodworth, Kumar Kshitij Patel, Sebastian Stich, Zhen Dai, Brian Bullins, Brendan Mcmahan, Ohad Shamir, and Nathan Srebro. Is local SGD better than minibatch SGD? In *International Conference on Machine Learning*, pp. 10334–10343. PMLR, 2020.

Jiaxiang Wu, Weidong Huang, Junzhou Huang, and Tong Zhang. Error compensated quantized SGD and its applications to large-scale distributed optimization. In *International conference on machine learning*, pp. 5325–5333. PMLR, 2018.

Zhaoyue Xia, Jun Du, and Yong Ren. Convergence theory of generalized distributed subgradient method with random quantization. *arXiv preprint arXiv:2207.10969*, 2022.

Zhaoyue Xia, Jun Du, Chunxiao Jiang, H Vincent Poor, Zhu Han, and Yong Ren. Distributed subgradient method with random quantization and flexible weights: Convergence analysis. *IEEE Transactions on Cybernetics*, 2023.

Kai Yi, Timur Kharisov, Igor Sokolov, and Peter Richtárik. Cohort squeeze: Beyond a single communication round per cohort in cross-device federated learning. *arXiv preprint arXiv:2406.01115*, 2024.

Moslem Zamani and François Glineur. Exact convergence rate of the last iterate in subgradient methods. *arXiv preprint arXiv:2307.11134*, 2023.

Hantian Zhang, Jerry Li, Kaan Kara, Dan Alistarh, Ji Liu, and Ce Zhang. The ZipML framework for training models with end-to-end low precision: The cans, the cannots, and a little bit of deep learning. *arXiv preprint arXiv:1611.05402*, 2016.

Shuai Zheng, Ziyue Huang, and James Kwok. Communication-efficient distributed blockwise momentum SGD with error-feedback. *Advances in Neural Information Processing Systems*, 32, 2019.

Yuchen Zheng, Yujia Xie, Ilbin Lee, Amin Dehghanian, and Nicoleta Serban. Parallel subgradient algorithm with block dual decomposition for large-scale optimization. *European journal of operational research*, 299(1):60–74, 2022.

Zhihan Zhu, Yanhao Zhang, and Yong Xia. Convergence rate of projected subgradient method with time-varying step-sizes. *Optimization Letters*, pp. 1–5, 2024.

Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 67(2):301–320, 2005.

A Appendix

Contents

B Three Ways to Compress: A Recap

In our experiments, we will examine three distinct approaches to constructing the compressors $\{\mathcal{Q}_i\}$ in MARINA-P, as outlined in (Gruntkowska et al., 2024):

1. Same Compressor. The conventional method where the server broadcasts an identical compressed message to all workers. Using a single Rand$K$ compressor $\mathcal{Q}$, we have $\mathcal{Q}_1^t(x^{t+1} - x^t) = \cdots = \mathcal{Q}_n^t(x^{t+1} - x^t) = \mathcal{Q}^t(x^{t+1} - x^t)$ for all workers $i \in [n]$. This approach, while simple, limits the amount of information conveyed.

2. Independent Compressors. This strategy employs a set of independent Rand$K$ compressors $\mathcal{Q}_i, i \in [n]$, generating distinct, independent sparse vectors $\mathcal{Q}_1(x), \ldots, \mathcal{Q}_n(x)$ for input $x \in \mathbb{R}^d$. This method allows for more diverse information transmission but lacks coordination between compressors.

3. Correlated Compressors. Introduced by Szlendak et al. (2022), this approach uses coordinated compressors, with Perm$K$ being a key example. For $d \geq n$ and $d = qn$, $q \in \mathbb{N}_{>0}$, Perm$K$ is defined as:

Definition 5 (Perm$K$). *Let $\pi = (\pi_1, \dots, \pi_d)$ be a random permutation of $\{1, \dots, d\}$. For $x \in \mathbb{R}^d$ and $i \in [n]$:*

$$\mathcal{Q}_i(x) := n \times \sum_{j=q(i-1)+1}^{qi} x_{\pi_j} e_{\pi_j}. \tag{17}$$

Perm$K$ ensures $\frac{1}{n}\sum_{i=1}^{n} \mathcal{Q}_i(x) = x$ deterministically, offering superior theoretical and practical performance in MARINA-P compared to the other two approaches. This method exploits correlation between compressors to achieve better approximation of the global model and improved communication efficiency.

C Experiments

To verify our theoretical results, we conducted experiments comparing MARINA-P with different compressor configurations from Section B (sameRand$K$, indRand$K$, and Perm$K$) against EF21-P with Top$K$ compression.

Hardware and Software. All algorithms were implemented in Python 3.10. We utilized three different CPU cluster node types:

1. AMD EPYC 7702 64-Core;
2. Intel(R) Xeon(R) Gold 6148 CPU @ 2.40GHz;
3. Intel(R) Xeon(R) Gold 6248 CPU @ 2.50GHz.

Algorithm 3 Synthetic datasets generation routine

Parameters: number nodes n, dimension d, parameter $\mu = 10^{-6}$, and noise scale s.
for $i = 1, \dots, n$ **do**
  Generate random noises $\nu_i^s = 1 + s\xi_i^s$, i.i.d. $\xi_i^s \sim \mathcal{N}(0, 1)$
  Take the initial tridiagonal matrix

$$\mathbf{A}_i = \frac{\nu_i^s}{4}\begin{pmatrix} 2 & -1 & & 0 \\ -1 & \ddots & \ddots & \\ & \ddots & \ddots & -1 \\ 0 & & -1 & 2 \end{pmatrix} \in \mathbb{R}^{d \times d}$$

end for
Take the mean of matrices $\mathbf{A} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{A}_i$
Find the minimum eigenvalue $\lambda_{\min}(\mathbf{A})$
for $i = 1, \dots, n$ **do**
  Update matrix $\mathbf{A}_i = \mathbf{A}_i + (\mu - \lambda_{\min}(\mathbf{A}))\mathbf{I}$
end for
Sample starting point $x^0 \sim \mathcal{N}(0, \mathbf{I})$
Output: matrices $\mathbf{A}_1, \cdots, \mathbf{A}_n$, starting point x^0

Objective and Datasets. The primary goal of these numerical experiments is to illustrate our theoretical findings and motivate further practical comparisons of MARINA-P against other baselines.

We consider a finite sum function $f(x) = \frac{1}{n}\sum_{i=1}^{n} f_i(x)$, consisting of synthetic non-smooth convex functions

$$f_i(x) := \|\mathbf{A}_i x\|_1,$$

where $\mathbf{A}_i \in \mathbb{R}^{d \times d}$ and $\mathbf{A}_i = \mathbf{A}_i^\top$ is the training data that belongs to the device/worker i. This objective was chosen for its simplicity to synthetically emulate the behavior of distributed training and to collect all required theoretical metrics, such as function suboptimality $f(w^t) - f(x^*)$. For this function, it is known that $x^* = (0,0,\ldots,0)^\top$, $f(x^*) = 0$. Each subgradient $\partial f_i\,(x)$ can be explicitly written (and computed) as $\partial f_i\,(x) = \mathbf{A}_i^\top\,\mathrm{sign}\,(\mathbf{A}_i x)$ (see Example 3.44 of the book (Beck, 2017) for proof details), where sign is the componentwise sign operator, i.e.

$$\mathrm{sign}(x)_i = \begin{cases} 1, & x_i \geq 0 \\ -1, & x_i < 0 \end{cases}. \tag{18}$$

Note, that $\partial f\,(x)$ can be computed as $\partial f\,(x) = \frac{1}{n}\sum_{i=1}^{n} \partial f_i\,(x)$. In all experiments of this section, we have $d = 1000$.

We generated synthetic matrices $\{\mathbf{A}_i\}_{i=1}^n$ (training data) via Algorithm 3 This data generation routine was inspired by a similar one used for solving synthetic quadratic problems (see Algorithm 11 in (Richtárik et al., 2022)). However, we introduced several minor modifications to the original algorithm for the needs of this project. We generated optimization problems having different numbers of nodes $n \in \{10, 100\}$ and different data heterogeneity regimes, controlled by the empirically proposed data dissimilarity measure

$$\sigma_A := \sqrt{\frac{1}{n}\sum_{i=1}^{n} \|\mathbf{A}_i\|_2^2 - \left(\frac{1}{n}\sum_{j=1}^{n} \|\mathbf{A}_j\|_2\right)^2}. \tag{19}$$

From the definition, it follows that the case of similar (or even identical) functions f_i relates to the small (or even 0) value of σ_A, whereas in the case of completely different f_i (which relate to heterogeneous data regime) σ_A can be large. In our experiments, homogeneity of each optimization task is controlled by noise scale s introduced in Algorithm 3. Indeed, for the noise scale $s = 0$, all matrices $\mathbf{A}_i$ are equal, whereas with the increase of the noise scale, functions become less "similar" and σ_A rises. We take noise scales $s \in \{0.1, 1.0, 10.0\}$. Table 2 summarizes the σ_A values corresponding to these noise scales for $n \in \{10, 100\}$.

n \ s	0.1	1.0	10.0
10	0.09	0.88	5.60
100	0.10	0.83	5.91

Table 2: Summary of the data heterogeneity σ_A values for different number of nodes n and various noise scales s.

Baselines and Hyperparameters. For each dataset (determined by values n and σ_A), we run the following baselines:

1. EF21-P with Top$K$ compressor;
2. MARINA-P with sameRand$K$ compressor;
3. MARINA-P with indRand$K$ compressors;
4. MARINA-P with Perm$K$ compressors.

where sameRand$K$, indRand$K$, and Perm$K$ are defined as described in subsection B.

In all experiments, we set $K = {}^{d}/_{n}$ and for MARINA-P we additionally choose $p = {}^{K}/_{d}$ to ensure a fair comparison of communication costs. Indeed, whereas for EF21-P with Top$K$, parameter K (and therefore $\zeta_{\mathcal{Q}} = K$) is deterministic and fixed throughout the optimization process, in the case of MARINA-P, K is random, but $\zeta_{\mathcal{Q}} = dp + (1-p)K = d\,({}^{K}/_{d}) + (1 - {}^{K}/_{d})\,K \leq 2K = \mathcal{O}(K)$, meaning that the choice of $p = {}^{K}/_{d}$ on expectation guarantees similar communication costs for EF21-P and MARINA-P.

For all algorithms, at each iteration (communication round) we updated the following metrics being tracked throughout the whole optimization procedure:

1. Function suboptimality $f(x^t) - f(x^*)$;
2. Number of bits per worker send from server to clients (titled as "bits/n" on corresponding Figure 6).

Method \ Stepsize type	Constant	Decreasing	Polyak	Reference
EF21-P	$\frac{1}{\sqrt{T}}\sqrt{\frac{V^0}{B_* L_0^2}}$	$\frac{\gamma_0}{\sqrt{t+1}}$	$\frac{f(w^t)-f(x^*)}{B_*\|\partial f(w^t)\|_2^2}$	(9), (10)
MARINA-P	$\frac{1}{\sqrt{T}}\sqrt{\frac{V^0}{\widetilde{B}_*}}$	$\frac{\gamma_0}{\sqrt{t+1}}$	$\frac{\frac{1}{n}\sum_{i=1}^n f_i(w_i^t)-f(x^*)}{\left\|\frac{1}{n}\sum_{i=1}^n \partial f_i\left(w_i^t\right)\right\|_2^2\left(1+2\frac{\sqrt{\frac{1}{n}\sum_{i=1}^n\left\|\partial f_i\left(w_i^t\right)\right\|_2^2}}{\left\|\frac{1}{n}\sum_{i=1}^n \partial f_i\left(w_i^t\right)\right\|_2}\sqrt{\frac{(1-p)\omega}{p}}\right)}$	(13), (14)

[1] For the decreasing stepsize, optimal $\gamma_0 = \sqrt{\frac{V_0}{2B_* L_0^2 \log(T+1)}}$ for EF21-P and $\gamma_0 = \sqrt{\frac{V_0}{2\widetilde{B}_* \log(T+1)}}$ for MARINA-P;
[2] B_*, $\widetilde{B}_*$, and other constants are defined in the respective theorems.

Table 3: Summary of theoretical stepsize formulas for EF21-P and MARINA-P algorithms.

We employed 64-bit precision in our experiments. Our communication model assumes that the server transfers $(65 + \log_2(d))q$ bits to each worker, where q represents the number of non-zero entries retained after sparsification. This total is broken down as follows:

- 64 bits allocated for each non-zero value;
- 1 bit for the sign of each entry;
- $\log_2(d)$ bits to encode the position of each non-zero entry.

The same communication model was also used in (Horváth et al., 2022).

For each value of $n \in \{10, 100\}$, we allocated an individual communication budget: $3.5 \cdot 10^8$ bits for $n = 10$ and $3.5 \cdot 10^7$ bits for $n = 100$. Each algorithm was terminated upon reaching its respective budget.

In experiments utilizing constant stepsizes, we set the stepsize to the largest theoretically acceptable value, multiplied by an individually tuned factor. This factor was selected from the set $\{2^{-9}, 2^{-8}, \ldots, 2^7\}$. For experiments employing adaptive stepsizes, we similarly tuned a constant factor. During the optimization procedure, this factor was multiplied by the theoretically defined adaptive stepsize at each iteration. Tables 3 and 5 summarize the theoretical stepsize formulas and optimal tuned stepsize multiplicative factors.

Figure 1: Constant stepsize; $n = 10$.

Method \ s	0.1	1.0	10.0
EF21-P with Top$K$	0.5	1.0	1.0
MARINA-P sameRand$K$	0.03125	0.03125	0.03125
MARINA-P indRand$K$	0.03125	0.03125	0.03125
MARINA-P Perm$K$	0.03125	0.03125	0.03125

Figure 2: Constant stepsize; $n = 100$.

Method \ s	0.1	1.0	10.0
EF21-P with Top$K$	4.0	4.0	8.0
MARINA-P sameRand$K$	0.03125	0.03125	0.03125
MARINA-P indRand$K$	0.03125	0.0625	0.0625
MARINA-P Perm$K$	0.03125	0.0625	0.0625

Figure 3: Polyak stepsize; $n = 10$.

Method \ s	0.1	1.0	10.0
EF21-P with Top$K$	16.0	16.0	16.0
MARINA-P sameRand$K$	2.0	2.0	2.0
MARINA-P indRand$K$	2.0	2.0	2.0
MARINA-P Perm$K$	2.0	2.0	2.0

Figure 4: Polyak stepsize; $n = 100$.

Method \ s	0.1	1.0	10.0
EF21-P with Top$K$	16.0	16.0	16.0
MARINA-P sameRand$K$	2.0	2.0	2.0
MARINA-P indRand$K$	2.0	2.0	2.0
MARINA-P Perm$K$	2.0	2.0	2.0

Figure 5: Optimal stepsize multiplicative factors for different methods, number of nodes, and heterogeneity levels.

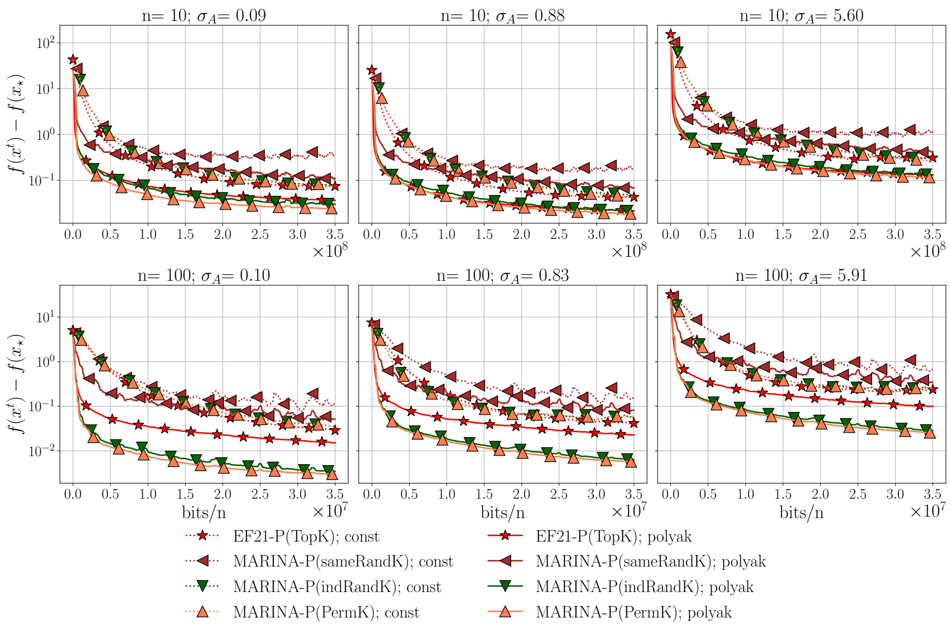

Figure 6: Performance comparison of EF21-P with Top$K$ and MARINA-P with sameRand$K$, indRand$K$, and Perm$K$ compressors ($K = d/n$). The left column of the legend corresponds to experiments with constant stepsizes, while the right column shows results with Polyak stepsizes. All stepsizes were set to the largest theoretically acceptable value multiplied by an individually tuned constant factor, selected from the set $\{2^{-9}, 2^{-8}, \ldots, 2^{7}\}$.

For MARINA-P, we initialize $w_i^0 = x^0$ for all $i \in [n]$, and $w^0 = x^0$ for EF21-P. This choice results in $V^0 = R^2 = \left\|x^0 - x^*\right\|_2^2 = \left\|x^0\right\|_2^2$, allowing explicit computation of V^0 constants in all cases.

In our experiments, we estimated the Lipschitz smoothness constants $L_{0,i}$ as $L_{0,i} \sim \|\mathbf{A}_i\|_2$. Although this approximation is not theoretically precise, we adopted it primarily for computational simplicity. Moreover, our constant multiplier tuning process compensates for any inaccuracies in the estimated $L_{0,i}$. It's worth noting that the $L_{0,i} \sim \|\mathbf{A}_i\|_2$ estimation is reasonably close to the worst-case bound, as demonstrated by:

$$\|\partial f_i(x)\|_2 = \left\|\mathbf{A}_i^\top \operatorname{sign}(\mathbf{A}_i x)\right\|_2 \leq \left\|\mathbf{A}_i^\top\right\|_2 \|\operatorname{sign}(\mathbf{A}_i x)\|_2 \leq \|\mathbf{A}_i\|_2 \sqrt{d}.$$

We also defined L_0 as $L_0 = \frac{1}{n}\sum_{i=1}^n L_{i,0}$.

Comparison of Convergence Behavior. We now present a more detailed version of the convergence comparison initially introduced in Section C of the main draft. Our experiments compare the performance of EF21-P with Top$K$ and MARINA-P with sameRand$K$, indRand$K$, and Perm$K$ compressors across the different datasets described in the previous section. Figure 6 illustrates the following key observations:

1. **Superiority of correlated compressors in the non-smooth convex setting.** For both constant and Polyak stepsizes, MARINA-P with Perm$K$ compressors slightly outperforms MARINA-P with indRand$K$ compressors, showing significant improvement over the conventional approach using the sameRand$K$ scheme. This observation suggests that correlated compressors indeed ensure better approximation of the compressed difference $\frac{1}{n}\sum_{i=1}^n \mathcal{Q}_i(x^{t+1} - x^t) \approx x^{t+1} - x^t$ (with equality in the case of Perm$K$), leading to superior convergence performance in practice. This behavior aligns well with experiments in the smooth non-convex setting from (Gruntkowska et al., 2024).
2. **Superior convergence behavior with adaptive stepsizes.** Each pair of experiments differing only in stepsize strategy (e.g., EF21-P with Top$K$, represented in Figure 6 with the same color and marker but different linestyles) demonstrates the practical efficiency of adaptive stepsize schemes. This marks the first time in the literature that such behavior has been

experimentally confirmed in the communication-efficient distributed non-smooth convex setting.

3. MARINA-P **with correlated compressors and Polyak stepsize outperforms for all datasets.** Figure 6 reveals that while all algorithms under constant stepsizes exhibit similar convergence behavior (slightly outperformed by EF21-P with Polyak stepsize), MARINA-P with correlated compressors and Polyak stepsize demonstrates superior performance. This advantage is particularly pronounced when $n = 100$.

These experimental results validate our theoretical findings and highlight the practical advantages of MARINA-P with correlated compressors and adaptive stepsizes in the non-smooth convex distributed optimization setting.

D Basic Facts and Inequalities

Useful inequalities: For all $x, y, x_1, \ldots, x_n \in \mathbb{R}^d$, $s > 0$ and $\alpha \in (0, 1]$, we have:

$$\left\|\frac{1}{n}\sum_{i=1}^n x_i\right\|_2 \leq \frac{1}{n}\sum_{i=1}^n \|x_i\|_2\,, \tag{20}$$

$$\left\|\frac{1}{n}\sum_{i=1}^n x_i\right\|_2^2 \leq \frac{1}{n}\sum_{i=1}^n \|x_i\|_2^2\,, \tag{21}$$

$$\|x+y\|_2^2 \leq (1+s)\|x\|_2^2 + (1+s^{-1})\|y\|_2^2\,, \tag{22}$$

$$\|x+y\|_2^2 \leq 2\|x\|_2^2 + 2\|y\|_2^2\,, \tag{23}$$

$$\langle x, y\rangle \leq \frac{\|x\|_2^2}{2s} + \frac{s\|y\|_2^2}{2}, \tag{24}$$

$$(1-\alpha)\left(1+\frac{\alpha}{2}\right) \leq 1 - \frac{\alpha}{2}, \tag{25}$$

$$(1-\alpha)\left(1+\frac{4}{\alpha}\right) \leq \frac{4}{\alpha}, \tag{26}$$

$$\langle a, b\rangle = \frac{1}{2}\left(\|a\|_2^2 + \|b\|_2^2 - \|a-b\|_2^2\right). \tag{27}$$

Tower property: For any random variables X and Y, we have

$$\mathbb{E}\left[\mathbb{E}\left[X \mid Y\right]\right] = \mathbb{E}\left[X\right]. \tag{28}$$

Cauchy-Bunyakovsky-Schwarz inequality: For any random variables X and Y, we have

$$|\mathbb{E}\left[XY\right]| \leq \sqrt{\mathbb{E}\left[X^2\right]\mathbb{E}\left[Y^2\right]}. \tag{29}$$

Variance decomposition: For any random vector $X \in \mathbb{R}^d$ and any non-random $c \in \mathbb{R}^d$, we have

$$\mathbb{E}\left[\|X - c\|_2^2\right] = \mathbb{E}\left[\|X - \mathbb{E}\left[X\right]\|_2^2\right] + \|\mathbb{E}\left[X\right] - c\|_2^2\,. \tag{30}$$

Jensen's inequality: For any random vector $X \in \mathbb{R}^d$ and any convex function $g : \mathbb{R}^d \mapsto \mathbb{R}$, we have

$$g(\mathbb{E}\left[X\right]) \leq \mathbb{E}\left[g(X)\right]. \tag{31}$$

Lemma 1 (Lemma 3 of Richtárik et al. (2021)). *Let $0 < p < 1$ and for $s > 0$ let $\theta(s)$ and $\beta(s)$ be defined as*

$$\theta(s) := 1 - (1-p)(1+s), \qquad \beta(s) := (1-p)(1+s^{-1}).$$

Then the solution of the optimization problem

$$\min_s \left\{\frac{\beta(s)}{\theta(s)} \;:\; 0 < s < \frac{p}{1-p}\right\} \tag{32}$$

is given by $s^* = \frac{1}{\sqrt{1-p}} - 1$. Furthermore, $\theta(s^*) = 1 - \sqrt{1-p}$, $\beta(s^*) = \frac{1-p}{1-\sqrt{1-p}}$ and

$$\sqrt{\frac{\beta(s^*)}{\theta(s^*)}} = \frac{1}{\sqrt{1-p}} - 1 = \frac{1}{p} + \frac{\sqrt{1-p}}{p} - 1 \leq \frac{2}{p} - 1. \tag{33}$$

In the trivial case $p = 1$, we have $\frac{\beta(s)}{\theta(s)} = 0$ for any $s > 0$, and (33) *is satisfied.*

E MISSING PROOFS FOR EF21-P

In this section, we present the detailed proofs for the theoretical results of EF21-P (Algorithm 1). Before delving into the proofs, we first discuss how our contribution extends the original results (Anonymous, 2024) on EF21-P for the non-smooth convex setting.

Recall that the standard single-node EF21-P algorithm (Gruntkowska et al., 2023) in the smooth case takes the form:

$$\begin{aligned} x^{t+1} &= x^t - \gamma_t \nabla f(w^t) \\ w^{t+1} &= w^t + \mathcal{C}^t\left(x^{t+1} - w^t\right). \end{aligned} \tag{34}$$

The key modification introduced by Anonymous (2024) was to replace the "smooth" update step (34) with a "non-smooth" one:

$$x^{t+1} = x^t - \gamma_t \partial f(w^t), \tag{35}$$

resulting in Algorithm 4.

Algorithm 4 EF21-P (single-node version)

Input: initial points $w^0, x^0 \in \mathbb{R}^d$, stepsize $\gamma_0 > 0$
for $t = 0, 1, 2, \ldots, T$ **do**
  Compute subgradient $g^t = \partial f(w^t)$
  Choose stepsize γ_t (can be set according to (9), (10), or (11))
  $x^{t+1} = x^t - \gamma_t g^t$
  Compute $\Delta^{t+1} = \mathcal{C}\left(x^{t+1} - w^t\right)$
  $w^{t+1} = w^t + \Delta^{t+1}$
end for
Output: x^T

Algorithm 5 EF21-P (distributed version)

Input: initial points $w^0 = x^0 \in \mathbb{R}^d$, stepsize $\gamma_0 > 0$
for $t = 0, 1, 2, \ldots, T$ **do**
  for $i = 1, \ldots, n$ **on Workers in parallel do**
    Receive compressed difference Δ^t from server
    Compute local subgradient $g_i^t = \partial f_i(w^t)$ and send it to server
  end for
  On Server:
  Receive g_i^t from workers
  Choose stepsize γ_t (can be set according to (9), (10), or (11))
  $x^{t+1} = x^t - \gamma_t \frac{1}{n} \sum_{i=1}^n g_i^t$
  Compute $\Delta^{t+1} = \mathcal{C}(x^{t+1} - w^t)$ and broadcast it to workers
  $w^{t+1} = w^t + \Delta^{t+1}$
  for $i = 1, \ldots, n$ **on Workers in parallel do**
    $w^{t+1} = w^t + \Delta^{t+1}$
  end for
end for
Output: x^T

As outlined in Section 3 of the main text, our primary contribution to the exploration of EF21-P is algorithmic. In Algorithm 5, we extend these results to the distributed setting, allowing for parallel computation of subgradients $\partial f(w^t)$. However, in both Algorithm 4 and 5, the gradient-like step (35) and state update step

$$w^{t+1} = w^t + \Delta^{t+1} \tag{36}$$

remain fundamentally the same.

Given that both single-node and distributed regimes result in the same update steps (35) and (36), the original proof by Anonymous (2024) for Algorithm 4 remains applicable to our Algorithm 5. Nevertheless, for completeness, we provide proofs for all necessary lemmas and theorems, following the approach in (Anonymous, 2024).

Our proof technique proceeds as follows: we first establish two key bounds in Lemma 2. We then combine these bounds to obtain a descent lemma (Lemma 3). Finally, we leverage this descent lemma to establish convergence results (Theorem 3 and Corollary 3) for different stepsize schedules.

Lemma 2 (Key bounds)**.** *Let Assumptions 1 and 2 hold. Define $W^t := \{w_1^t, \ldots, w_n^t\}$. Then, for a single iteration of EF21-P (Algorithm 1) with $\gamma_t > 0$, we have the following bounds:*

1.

$$\mathbb{E}\left[\left\|x^{t+1}-x^*\right\|_2^2 \mid x^t, W^t\right] \leq \left\|x^t-x^*\right\|_2^2-2\gamma_t\left(f(w^t)-f(x^*)\right)+\frac{1}{\lambda}\left\|w^t-x^t\right\|_2^2 + (1+\lambda)\gamma_t^2\left\|\partial f\left(w^t\right)\right\|_2^2, \quad (37)$$

where $\lambda>0$*;*

2.

$$\mathbb{E}\left[\left\|w^{t+1}-x^{t+1}\right\|_2^2 \mid x^t, W^t\right] \leq (1-\theta)\left\|w^t-x^t\right\|_2^2+\gamma_t^2\beta\left\|\partial f\left(w^t\right)\right\|_2^2, \quad (38)$$

where $\theta := 1-\sqrt{1-\alpha}$ *and* $\beta := \frac{1-\alpha}{1-\sqrt{1-\alpha}}$.

Proof. We prove each bound separately.

1. To establish the first bound, we begin by applying the definition of subgradient:

$$f(x^*) \geq f(w^t)+\left\langle \partial f\left(w^t\right), x^*-w^t\right\rangle, \quad (39)$$

which implies:

$$\left\langle \partial f\left(w^t\right), w^t-x^*\right\rangle \geq f(w^t)-f(x^*). \quad (40)$$

Next, we apply (24) with $s:=\lambda\gamma_t$:

$$2\gamma_t\left\langle \partial f\left(w^t\right), w^t-x^t\right\rangle \leq \lambda\gamma_t^2\left\|\partial f\left(w^t\right)\right\|_2^2+\frac{1}{\lambda}\left\|w^t-x^t\right\|_2^2. \quad (41)$$

where $\lambda>0$ is a constant to be specified later.

Using the linearity of inner product, we derive:

$$\begin{aligned}
-2\gamma_t\left\langle \partial f\left(w^t\right), x^*-x^t\right\rangle &= -2\gamma_t\left\langle \partial f\left(w^t\right), w^t-x^*\right\rangle+2\gamma_t\left\langle \partial f\left(w^t\right), w^t-x^t\right\rangle \\
&\overset{(41)}{\leq} -2\gamma_t\left(f(w^t)-f(x^*)\right)+\lambda\gamma_t^2\left\|\partial f\left(w^t\right)\right\|_2^2+\frac{1}{\lambda}\left\|w^t-x^t\right\|_2^2 \quad (42)
\end{aligned}$$

Finally, we establish the first bound (37):

$$\begin{aligned}
\mathbb{E}\left[\left\|x^{t+1}-x^*\right\|_2^2 \mid x^t, W^t\right] &= \mathbb{E}\left[\left\|x^t-\gamma_t\partial f\left(w^t\right)-x^*\right\|_2^2 \mid x^t, W^t\right] \\
&= \left\|x^t-x^*\right\|_2^2-2\gamma_t\left\langle \partial f\left(w^t\right), x^t-x^*\right\rangle+\gamma_t^2\left\|\partial f\left(w^t\right)\right\|_2^2 \\
&\overset{(42)}{\leq} \left\|x^t-x^*\right\|_2^2-2\gamma_t\left(f(w^t)-f(x^*)\right)+\frac{1}{\lambda}\left\|w^t-x^t\right\|_2^2 \\
&\quad +(1+\lambda)\gamma_t^2\left\|\partial f\left(w^t\right)\right\|_2^2. \quad (43)
\end{aligned}$$

2. For the second bound, we proceed as follows:

$$\begin{aligned}
\mathbb{E}\left[\left\|w^{t+1}-x^{t+1}\right\|_2^2 \mid x^t, W^t\right] &= \mathbb{E}\left[\left\|w^t-\mathcal{C}(x^{t+1}-w^t)-x^{t+1}\right\|_2^2 \mid x^t, W^t\right] \\
&\leq (1-\alpha)\left\|w^t-x^{t+1}\right\|_2^2 \\
&= (1-\alpha)\left\|w^t-x^t+\gamma_t\partial f\left(w^t\right)\right\|_2^2 \\
&\leq (1-\alpha)(1+s)\left\|w^t-x^t\right\|_2^2+\gamma_t^2(1-\alpha)(1+s^{-1})\left\|\partial f\left(w^t\right)\right\|_2^2 \\
&\leq (1-\theta(s))\left\|w^t-x^t\right\|_2^2+\gamma_t^2\beta(s)\left\|\partial f\left(w^t\right)\right\|_2^2,
\end{aligned} \quad (44)$$

where $\theta(s):=1-(1-\alpha)(1+s)$ and $\beta(s):=(1-\alpha)(1+s^{-1})$.

Following (Richtárik et al., 2021), the optimal s, minimizing $\frac{(1-\alpha)(1+1/s)}{1-(1-\alpha)(1+s)}$, is $s_* = \frac{1}{\sqrt{1-\alpha}}-1$, resulting in $\theta := 1-(1-\alpha)(1+s_*) = 1-\sqrt{1-\alpha}$ and $\beta := (1-\alpha)(1+1/s_*) = \frac{1-\alpha}{1-\sqrt{1-\alpha}}$.

Therefore, we can establish the second bound (38):

$$\mathbb{E}\left[\left\|w^{t+1}-x^{t+1}\right\|_2^2 \mid x^t, W^t\right] \leq (1-\theta)\left\|w^t-x^t\right\|_2^2+\gamma_t^2\beta\left\|\partial f\left(w^t\right)\right\|_2^2. \quad (45)$$

□

With these two key bounds established in Lemma 2, we can now proceed to the descent lemma. This lemma describes the one-step behavior of Algorithm 1 for any $\gamma_t > 0$ and will be crucial in establishing our convergence rates.

Lemma 3 (Descent lemma)**.** *Let the conditions of Lemma 2 hold. Define the Lyapunov function*

$$V_\lambda^t := \left\|x^t - x^*\right\|_2^2 + \frac{1}{\lambda\theta}\left\|w^t - x^t\right\|_2^2, \tag{46}$$

where $\lambda > 0$ and $\theta := 1 - \sqrt{1-\alpha}$. Then

$$\mathbb{E}\left[V_\lambda^{t+1} \mid x^t, W^t\right] \leq V_\lambda^t - 2\gamma_t\left(f(w^t) - f(x^*)\right) + \left(1 + \lambda + \frac{\beta}{\lambda\theta}\right)\gamma_t^2\left\|\partial f\left(w^t\right)\right\|_2^2, \tag{47}$$

where $\beta := \frac{1-\alpha}{1-\sqrt{1-\alpha}}$.

Proof. Recall that Lemma 2 provides us with two key bounds:

$$\begin{aligned}\mathbb{E}\left[\left\|x^{t+1} - x^*\right\|_2^2 \mid x^t, W^t\right] \quad &\leq \quad \left\|x^t - x^*\right\|_2^2 - 2\gamma_t\left(f(w^t) - f(x^*)\right) + \frac{1}{\lambda}\left\|w^t - x^t\right\|_2^2 \\ &\qquad + (1+\lambda)\gamma_t^2\left\|\partial f\left(w^t\right)\right\|_2^2,\end{aligned} \tag{48}$$

and

$$\mathbb{E}\left[\left\|w^{t+1} - x^{t+1}\right\|_2^2 \mid x^t, W^t\right] \leq (1-\theta)\left\|w^t - x^t\right\|_2^2 + \gamma_t^2\beta\left\|\partial f\left(w^t\right)\right\|_2^2. \tag{49}$$

To obtain our descent lemma, we combine (48) with $\frac{1}{\lambda\theta}$ times (49):

$$\begin{aligned}&\mathbb{E}\left[V_\lambda^{t+1} \mid x^t, W^t\right] \\ &\overset{(46)}{=} \mathbb{E}\left[\left\|x^{t+1} - x^*\right\|_2^2 \mid x^t, W^t\right] + \frac{1}{\lambda\theta}\mathbb{E}\left[\left\|w^{t+1} - x^{t+1}\right\|_2^2 \mid x^t, W^t\right] \\ &\leq \left\|x^t - x^*\right\|_2^2 - 2\gamma_t\left(f(w^t) - f(x^*)\right) + \frac{1}{\lambda}\left\|w^t - x^t\right\|_2^2 + (1+\lambda)\gamma_t^2\left\|\partial f\left(w^t\right)\right\|_2^2 \\ &\quad + \frac{1}{\lambda\theta}\left((1-\theta)\left\|w^t - x^t\right\|_2^2 + \gamma_t^2\beta\left\|\partial f\left(w^t\right)\right\|_2^2\right) \\ &= \left\|x^t - x^*\right\|_2^2 + \frac{1}{\lambda\theta}\left\|w^t - x^t\right\|_2^2 - 2\gamma_t\left(f(w^t) - f(x^*)\right) \\ &\quad + (1+\lambda)\gamma_t^2\left\|\partial f\left(w^t\right)\right\|_2^2 + \frac{\gamma_t^2\beta}{\lambda\theta}\left\|\partial f\left(w^t\right)\right\|_2^2 \\ &= V_\lambda^t - 2\gamma_t\left(f(w^t) - f(x^*)\right) + (1+\lambda)\gamma_t^2\left\|\partial f\left(w^t\right)\right\|_2^2 + \frac{\gamma_t^2\beta}{\lambda\theta}\left\|\partial f\left(w^t\right)\right\|_2^2.\end{aligned} \tag{50}$$

□

E.1 Proof of Theorem 1

Having established the descent lemma, we now proceed to the theorem, which characterizes the convergence behavior of EF21-P under various stepsize schedules.

Before we state and prove the theorem, it is important to make a notational remark to avoid confusion.

Remark 2. *In Lemmas 2 and 3, we used an auxiliary term $\lambda > 0$ arising from the application of Young's inequality. This term also appeared in the definition of the Lyapunov function V_λ^t. In the following theorem, we will show how to choose this λ optimally and denote it as λ_*. Consequently, we define a Lyapunov function V^t such that $V^t := V_{\lambda_*}^t$. For simplicity of notation, we will use V^t instead of $V_{\lambda_*}^t$ in the theorem statement and proof.*

Theorem 3 (Theorem 1)**.** *Let Assumptions 1, 2 and 3 hold. Define a Lyapunov function $V^t := \left\|x^t - x^*\right\|_2^2 + \frac{1}{\lambda_*\theta}\left\|w^t - x^t\right\|_2^2$, where $\lambda_* := \frac{\sqrt{1-\alpha}}{1-\sqrt{1-\alpha}}$ and $\theta := 1 - \sqrt{1-\alpha}$. Define also a constant $B_* := 1 + 2\frac{\sqrt{1-\alpha}}{1-\sqrt{1-\alpha}}$. Let $\{w^t\}$ be the sequence produced by EF21-P (Algorithm 1). Define $\overline{w}^T := \frac{1}{T}\sum_{t=0}^{T-1} w^t$ and $\widehat{w}^T := \frac{1}{\sum_{t=0}^{T-1}\gamma_t}\sum_{t=0}^{T-1}\gamma_t w^t$.*

1. (Constant stepsize). *If $\gamma_t := \gamma > 0$, then*

$$\mathbb{E}\left[f(\overline{w}^T) - f(x^*)\right] \leq \frac{V^0}{2\gamma T} + \frac{B_* L_0^2 \gamma}{2}. \tag{51}$$

If, moreover, optimal γ is chosen i.e.

$$\gamma := \frac{1}{\sqrt{T}}\sqrt{\frac{V^0}{B_* L_0^2}}, \tag{52}$$

then

$$\mathbb{E}\left[f(\overline{w}^T) - f(x^*)\right] \leq \frac{\sqrt{B_* L_0^2 V^0}}{\sqrt{T}}. \tag{53}$$

2. (Polyak stepsize). *If γ_t is chosen as*

$$\gamma_t \quad := \quad \frac{f(w^t) - f(x^*)}{B_* \left\|\partial f\left(w^t\right)\right\|_2^2}, \tag{54}$$

then

$$\mathbb{E}\left[f(\overline{w}^T) - f(x^*)\right] \leq \frac{\sqrt{B_* L_0^2 V^0}}{\sqrt{T}}. \tag{55}$$

3. (Decreasing stepsize). *If γ_t is chosen as*

$$\gamma_t := \frac{\gamma_0}{\sqrt{t+1}}, \tag{56}$$

then

$$\mathbb{E}\left[f(\widehat{w}^T) - f(x^*)\right] \leq \frac{V^0 + 2\gamma_0^2 B_* L_0^2 \log(T+1)}{\gamma_0\sqrt{T}}. \tag{57}$$

If, moreover, optimal γ_0 is chosen i.e.

$$\gamma_0 := \sqrt{\frac{V_0}{2B_* L_0^2 \log(T+1)}}, \tag{58}$$

then

$$\mathbb{E}\left[f(\widehat{w}^T) - f(x^*)\right] \leq 2\sqrt{2B_* L_0^2 V_0}\sqrt{\frac{\log(T+1)}{T}}. \tag{59}$$

Proof. We will prove each part of the theorem separately, starting with some general bounds that will be useful throughout the proof.

From Assumption 3, we can infer that f is L_0-Lipschitz with $L_0 \leq \frac{1}{n}\sum_{i=1}^n L_{0,i}$ and

$$\left\|\partial f(x)\right\|_2 \quad \leq L_0 \quad \forall x \in \mathbb{R}^d. \tag{60}$$

Now, we proceed to prove each part of the theorem.

1. (Constant stepsize). Using (60), Lemma 3, the tower property of expectation (28), and choosing constant stepsize $\gamma_t := \gamma > 0$, we obtain

$$\mathbb{E}\left[V^{t+1}\right] \leq \mathbb{E}\left[V^t\right] - 2\gamma\mathbb{E}\left[f(w^t) - f(x^*)\right] + \left(1 + \lambda + \frac{\beta}{\lambda\theta}\right)\gamma^2\mathbb{E}\left[\left\|\partial f\left(w^t\right)\right\|_2^2\right], \tag{61}$$

where $\lambda > 0$, $\theta := 1 - \sqrt{1-\alpha}$ and $\beta := \frac{1-\alpha}{1-\sqrt{1-\alpha}}$.

From the inequality (61), we have

$$\mathbb{E}\left[V^{t+1}\right] \leq \mathbb{E}\left[V^t\right] - 2\gamma\mathbb{E}\left[f(w^t) - f(x^*)\right] + B_\lambda L_0^2 \gamma^2, \tag{62}$$

where $B_\lambda := 1 + \lambda + \frac{\beta}{\lambda\theta}$.

Since f is convex, by Jensen's inequality (31), we have

$$\begin{aligned}\mathbb{E}\left[f(\widehat{w}^T) - f(x^*)\right] &\leq \mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T-1} f(w^t) - f(x^*)\right] \\ &\leq \frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[f(w^t) - f(x^*)\right] \\ &\overset{(62)}{\leq} \frac{\mathbb{E}\left[V^0\right] - \mathbb{E}\left[V^T\right]}{2\gamma T} + \frac{B_\lambda L_0^2 \gamma}{2} \\ &\overset{V^T \geq 0}{\leq} \frac{V^0}{2\gamma T} + \frac{B_\lambda L_0^2\gamma}{2}. \end{aligned} \tag{63}$$

To optimize this bound, we need to find the optimal λ. Note that $\phi(\lambda) := 1 + \lambda + \frac{\beta}{\lambda\theta}$ is a convex function on $(0, +\infty)$ for any fixed values $\beta > 0$ and $\theta \in (0,1]$.

Therefore, we define the optimal λ value (denoted λ_*) as

$$\lambda_* := \arg\min_{\lambda > 0}\left(1 + \lambda + \frac{\beta}{\lambda\theta}\right) = \sqrt{\frac{\beta}{\theta}} = \frac{\sqrt{1-\alpha}}{1 - \sqrt{1-\alpha}}. \tag{64}$$

Next, we define the optimal B_λ value (denoted B_*) as

$$B_* := B_{\lambda_*} = 1 + 2\sqrt{\frac{\beta}{\theta}} = 1 + 2\frac{\sqrt{1-\alpha}}{1-\sqrt{1-\alpha}}. \tag{65}$$

Plugging (65) into (63), we get

$$\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n} f_i(\overline{w}^T) - f(x^*)\right] \overset{(65),\,(63)}{\leq} \frac{V^0}{2\gamma T} + \frac{B_* L_0^2 \gamma}{2}. \tag{66}$$

Thus, we have established (51).

To derive the optimal rate (53), we need to find the optimal γ stepsize (which we denote γ_*):

$$\gamma_* := \arg\min_{\gamma}\left(\frac{V^0}{2\gamma T} + \frac{B_*\gamma}{2}\right) = \frac{1}{\sqrt{T}}\sqrt{\frac{V^0}{B_* L_0^2}}. \tag{67}$$

Therefore, choosing $\gamma := \gamma_*$, (66) reduces to

$$\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n} f_i(\overline{w}^T) - f(x^*)\right] \leq \frac{V^0}{2\gamma_* T} + \frac{B_*\gamma_*}{2} = \frac{\sqrt{V^0 B_* L_0^2}}{\sqrt{T}}, \tag{68}$$

which gives us (53).

2. (Polyak stepsize).

Using Lemma 3, we have

$$\mathbb{E}\left[V^{t+1} \mid x^t, W^t\right] \leq V^t - 2\gamma_t f(w^t) - f(x^*) + \left(1 + \lambda + \frac{\beta}{\lambda\theta}\right)\gamma_t^2 \left\|\partial f\left(w^t\right)\right\|_2^2, \tag{69}$$

where $\lambda > 0$, $\theta := 1 - \sqrt{1-\alpha}$ and $\beta := \frac{1-\alpha}{1-\sqrt{1-\alpha}}$.

We choose the Polyak stepsize γ_t as the one that minimizes the right-hand side of (69):

$$\begin{aligned}\gamma_t &:= \arg\min_{\gamma}\left\{V^t - 2\gamma\left(f(w^t) - f(x^*)\right) + \left(1 + \lambda + \frac{\beta}{\lambda\theta}\right)\gamma^2 \left\|\partial f\left(w^t\right)\right\|_2^2\right\} \\ &= \frac{f(w^t) - f(x^*)}{\left(1 + \lambda + \frac{\beta}{\lambda\theta}\right)\left\|\partial f\left(w^t\right)\right\|_2^2}. \end{aligned} \tag{70}$$

Note that the denominator in (70) is a convex function of λ. Therefore, similar to (64), we can choose the optimal λ as

$$\lambda_* := \arg\min_{\lambda>0}\left(1+\lambda+\frac{\beta}{\lambda\theta}\right) = \sqrt{\frac{\beta}{\theta}}, \tag{71}$$

and thus

$$B_* = 1+\lambda_*+\frac{\beta}{\lambda_*\theta} = 1+2\sqrt{\frac{\beta}{\theta}} = 1+2\frac{\sqrt{1-\alpha}}{1-\sqrt{1-\alpha}}. \tag{72}$$

Therefore, we derive the final expression for our Polyak stepsize:

$$\gamma_t \quad := \quad \frac{f(w^t)-f(x^*)}{B_*\left\|\partial f\left(w^t\right)\right\|_2^2}. \tag{73}$$

Next, plugging (73) into (69) and using the tower property of expectation (28), we obtain

$$\begin{aligned}\mathbb{E}\left[V^{t+1}\right] \quad &\overset{(70),(73)}{\leq} \quad \mathbb{E}\left[V^t\right]-\mathbb{E}\left[\frac{\left(f(w^t)-f(x^*)\right)^2}{\left\|\partial f\left(w^t\right)\right\|_2^2+2\left\|\partial f\left(w^t\right)\right\|_2\sqrt{\left\|\partial f\left(w^t\right)\right\|_2^2}\sqrt{\frac{(1-p)\omega}{p}}}\right]\\ &\overset{(60)}{\leq} \quad \mathbb{E}\left[V^t\right]-\frac{\mathbb{E}\left[\left(f(w^t)-f(x^*)\right)^2\right]}{L_0^2 B_*},\end{aligned} \tag{74}$$

Since f is convex, by Jensen's inequality (31) and the Cauchy-Bunyakovsky-Schwarz inequality (29) with $X := f(w^t)-f(x^*)$ and $Y := 1$, we have

$$\begin{aligned}\mathbb{E}\left[f_i(\overline{w}^T)-f(x^*)\right] \quad &\overset{(31)}{\leq} \quad \mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T-1} f(w^t)-f(x^*)\right]\\ &\leq \quad \frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[f(w^t)-f(x^*)\right]\\ &\overset{(29)}{\leq} \quad \frac{1}{T}\sum_{t=0}^{T-1}\sqrt{\mathbb{E}\left[\left(f(w^t)-f(x^*)\right)^2\right]}\\ &\leq \quad \sqrt{\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[\left(f(w^t)-f(x^*)\right)^2\right]}\\ &\overset{(74)}{\leq} \quad \frac{\sqrt{B_* L_0^2}}{\sqrt{T}}\sqrt{\mathbb{E}\left[V^0\right]-\mathbb{E}\left[V^T\right]}\\ &\leq \quad \frac{\sqrt{V^0 B_* L_0^2}}{\sqrt{T}}.\end{aligned} \tag{75}$$

Thus, we have established (55).

3. (Decreasing stepsize).

By the same arguments as in the analysis for the constant stepsize case, we can get a bound

$$\mathbb{E}\left[V^{t+1}\right] \leq \mathbb{E}\left[V^t\right]-2\gamma_t\mathbb{E}\left[f(w^t)-f(x^*)\right]+B_* L_0^2\gamma_t^2, \tag{76}$$

where $B_* \overset{(65)}{=} 1+2\frac{\sqrt{1-\alpha}}{1-\sqrt{1-\alpha}}$.

If $\gamma_t := \frac{\gamma_0}{\sqrt{t+1}}$ with $\gamma_0 > 0$, then we can get the bounds

$$\sum_{t=0}^{T-1}\gamma_t \geq \frac{\gamma_0\sqrt{T}}{2}, \quad \text{and} \quad \sum_{t=0}^{T-1}\gamma_t^2 \leq 2\gamma_0^2\log(T+1). \tag{77}$$

Since f is convex, by Jensen's inequality (31), we have

$$\begin{aligned}\mathbb{E}\left[f(\widehat{w}^T) - f(x^*)\right] &\overset{(31)}{\leq} \mathbb{E}\left[\frac{1}{\sum_{t=0}^{T-1}\gamma_t}\sum_{t=0}^{T-1}\gamma_t[f(w^t) - f(x^*)]\right] \\ &\overset{(76)}{\leq} \frac{(\mathbb{E}\left[V^0\right] - \mathbb{E}\left[V^T\right]) + B_* L_0^2 \sum_{t=0}^{T-1}\gamma_t^2}{2\sum_{t=0}^{T-1}\gamma_t} \\ &\overset{V^T \geq 0}{\leq} \frac{V^0 + B_* L_0^2 \sum_{t=0}^{T-1}\gamma_t^2}{2\sum_{t=0}^{T-1}\gamma_t} \\ &\overset{(77)}{\leq} \frac{V^0 + 2\gamma_0^2 B_* L_0^2 \log(T+1)}{\gamma_0\sqrt{T}}. \end{aligned} \tag{78}$$

The optimal γ_0 can be chosen by minimizing the right-hand side of (78), i.e.,

$$\gamma_* \quad = \quad \arg\min_{\gamma_0 > 0}\left(\frac{V_0}{\gamma_0\sqrt{T}} + \frac{2\gamma_0 B_* L_0^2 \log(T+1)}{\sqrt{T}}\right) = \sqrt{\frac{V_0}{2B_* L_0^2 \log(T+1)}}. \tag{79}$$

Therefore, choosing $\gamma_0 := \gamma_*$, (78) reduces to

$$\mathbb{E}\left[f(\widehat{w}^T) - f(x^*)\right] \leq \frac{V_0}{\gamma_*\sqrt{T}} + \frac{2\gamma_* \log(T+1)}{\sqrt{T}} \quad = \quad 2\sqrt{2V_0}\sqrt{B_* L_0^2}\sqrt{\frac{\log(T+1)}{T}}, \tag{80}$$

and we get (59). □

Having established our main theorem, we can now derive a corollary that provides more practical insights into the performance of EF21-P.

E.2 Proof of Corollary 1

Corollary 3 (Corollary 1)**.** *Let the conditions of Theorem 1 be met and $w^0 = x^0$. If γ_t is set according to* (9) *or* (10) *(constant or Polyak stepsizes) then* EF21-P *(Algorithm 5) requires*

$$T = \mathcal{O}\left(\frac{L_0^2 R_0^2}{\alpha\varepsilon^2}\right) \tag{81}$$

iterations/communication rounds in order to achieve $\mathbb{E}\left[f(\overline{w}^T) - f(x^*)\right] \leq \varepsilon$. Moreover, under the assumption that the communication cost is proportional to the number of non-zero components of vectors transmitted from the server to workers, we have that the expected total communication cost per worker equals

$$d + \zeta_{\mathcal{C}} T = \mathcal{O}\left(d + \frac{\zeta_{\mathcal{C}} L_0^2 R_0^2}{\alpha\varepsilon^2}\right). \tag{82}$$

Proof. From (53) and (55), we have the convergence rate

$$\mathbb{E}\left[f(\overline{w}^T) - f(x^*)\right] \leq \frac{\sqrt{B_* L_0^2 V^0}}{\sqrt{T}}, \tag{83}$$

where
$V^0 = \left\|x^0 - x^*\right\|_2^2 + \frac{1}{\lambda_*\theta}\left\|w^0 - x^0\right\|_2^2$, with $\lambda_* := \frac{\sqrt{1-\alpha}}{1-\sqrt{1-\alpha}}$ and $\theta := 1 - \sqrt{1-\alpha}$.

$B_* := 1 + 2\frac{\sqrt{1-\alpha}}{1-\sqrt{1-\alpha}}$, resulting in a complexity

$$T = \mathcal{O}\left(\frac{B_* L_0^2 V^0}{\varepsilon^2}\right) \tag{84}$$

required to achieve $\mathbb{E}\left[f(\overline{w}^T) - f(x^*)\right] \leq \varepsilon$. Assuming $w^0 = x^0$, we get

$$V^0 = R_0^2 = \left\|x^0 - x^*\right\|_2^2. \tag{85}$$

Further, note

$$\begin{aligned} B_* &= 1 + 2\frac{\sqrt{1-\alpha}}{1-\sqrt{1-\alpha}} \\ &= 1 + 2\frac{\sqrt{1-\alpha}(1+\sqrt{1-\alpha})}{\alpha} \\ &= 1 + 2\left(\frac{\sqrt{1-\alpha}+1-\alpha}{\alpha}\right) \\ &\leq \frac{4}{\alpha} - 1. \end{aligned} \tag{86}$$

Plugging (85) and (86) into (84), we get (81).

The expected total communication cost per worker is

$$d + \zeta_{\mathcal{C}} T = \mathcal{O}\left(d + \frac{\zeta_{\mathcal{C}} L_0^2 R_0^2}{\alpha \varepsilon^2}\right). \tag{87}$$

□

This concludes our analysis of the EF21-P algorithm. We have established its convergence rates for different stepsize schedules and derived complexity bounds. In the next section, we will proceed to analyze the MARINA-P algorithm.

F Missing Proofs For MARINA-P

In this section, we present the detailed proofs for the theoretical results for MARINA-P algorithm. Our proof technique proceeds as follows: we first establish two key bounds in Lemma 4. We then combine these bounds to obtain a descent lemma (Lemma 5). Finally, we leverage this descent lemma to establish convergence results (Theorem 4 and Corollary 4) for different stepsize schedules.

Lemma 4 (Key bounds)**.** *Let Assumptions 1 and 2 hold. Define $W^t := \{w_1^t, \ldots, w_n^t\}$. Then, for a single iteration of* MARINA-P *(Algorithm 2) with $\gamma_t > 0$, we have the following bounds:*

1.

$$\begin{aligned} \mathbb{E}\left[\left\|x^{t+1} - x^*\right\|_2^2 \mid x^t, W^t\right] \quad &\leq \quad \left\|x^t - x^*\right\|_2^2 - 2\gamma_t\left(\frac{1}{n}\sum_{i=1}^n f_i(w_i^t) - f(x^*)\right) + \lambda\gamma_t^2\frac{1}{n}\sum_{i=1}^n \left\|\partial f_i\left(w_i^t\right)\right\|_2^2 \\ &\quad + \frac{1}{\lambda}\frac{1}{n}\sum_{i=1}^n \left\|w_i^t - x^t\right\|_2^2 + \gamma_t^2\left\|\frac{1}{n}\sum_{i=1}^n \partial f_i\left(w_i^t\right)\right\|_2^2, \end{aligned} \tag{88}$$

where $\lambda > 0$;

2.

$$\frac{1}{n}\sum_{i=1}^n \mathbb{E}\left[\left\|w_i^{t+1} - x^{t+1}\right\|_2^2 \mid x^t, W^t\right] \leq (1-p)\frac{1}{n}\sum_{i=1}^n \left\|w_i^t - x^t\right\|_2^2 + (1-p)\omega\gamma_t^2\left\|\frac{1}{n}\sum_{i=1}^n \partial f_i\left(w_i^t\right)\right\|_2^2. \tag{89}$$

Proof. We prove each bound separately.

1.

To establish the first bound, we begin by applying the definition of subgradient:

$$f_i(x^*) \geq f_i(w_i^t) + \left\langle \partial f_i\left(w_i^t\right), x^* - w_i^t\right\rangle \quad \forall i \in [n]. \tag{90}$$

Summing over all $i \in [n]$, we obtain

$$\frac{1}{n}\sum_{i=1}^n f_i(x^*) \geq \frac{1}{n}\sum_{i=1}^n f_i(w_i^t) + \frac{1}{n}\sum_{i=1}^n \left\langle \partial f_i\left(w_i^t\right), x^* - w_i^t\right\rangle, \tag{91}$$

which implies

$$\frac{1}{n}\sum_{i=1}^{n}\left\langle \partial f_i\left(w_i^t\right), w_i^t - x^*\right\rangle \geq \frac{1}{n}\sum_{i=1}^{n} f_i(w_i^t) - f(x^*). \tag{92}$$

Next, we apply (24) with $s := \lambda\gamma_t$:

$$2\gamma_t\frac{1}{n}\sum_{i=1}^{n}\left\langle \partial f_i\left(w_i^t\right), w_i^t - x^t\right\rangle \leq \lambda\gamma_t^2\frac{1}{n}\sum_{i=1}^{n}\left\|\partial f_i\left(w_i^t\right)\right\|_2^2 + \frac{1}{\lambda}\frac{1}{n}\sum_{i=1}^{n}\left\|w_i^t - x^t\right\|_2^2, \tag{93}$$

where $\lambda > 0$ is a constant to be specified later.

Using the linearity of inner product, we derive

$$\begin{aligned}
&-2\gamma_t\left\langle \frac{1}{n}\sum_{i=1}^{n}\partial f_i\left(w_i^t\right), x^t - x^*\right\rangle \\
&= -2\gamma_t\frac{1}{n}\sum_{i=1}^{n}\left\langle \partial f_i\left(w_i^t\right), x^t - x^*\right\rangle \\
&= -2\gamma_t\frac{1}{n}\sum_{i=1}^{n}\left\langle \partial f_i\left(w_i^t\right), w_i^t - x^*\right\rangle + 2\gamma_t\frac{1}{n}\sum_{i=1}^{n}\left\langle \partial f_i\left(w_i^t\right), w_i^t - x^t\right\rangle \\
&\overset{(92),(93)}{\leq} -2\gamma_t\left(\frac{1}{n}\sum_{i=1}^{n} f_i(w_i^t) - f(x^*)\right) + \lambda\gamma_t^2\frac{1}{n}\sum_{i=1}^{n}\left\|\partial f_i\left(w_i^t\right)\right\|_2^2 + \frac{1}{\lambda}\frac{1}{n}\sum_{i=1}^{n}\left\|w_i^t - x^t\right\|_2^2.
\end{aligned} \tag{94}$$

Finally, we establish the first bound (88):

$$\begin{aligned}
&\mathbb{E}\left[\left\|x^{t+1} - x^*\right\|_2^2 \mid x^t, W^t\right] \\
&\overset{(9)}{=} \mathbb{E}\left[\left\|x^t - \gamma_t\frac{1}{n}\sum_{i=1}^{n}\partial f_i\left(w_i^t\right) - x^*\right\|_2^2 \mid x^t, W^t\right] \\
&= \left\|x^t - x^*\right\|_2^2 - 2\gamma_t\left\langle \frac{1}{n}\sum_{i=1}^{n}\partial f_i\left(w_i^t\right), x^t - x^*\right\rangle + \gamma_t^2\left\|\frac{1}{n}\sum_{i=1}^{n}\partial f_i\left(w_i^t\right)\right\|_2^2 \\
&\overset{(94)}{\leq} \left\|x^t - x^*\right\|_2^2 - 2\gamma_t\left(\frac{1}{n}\sum_{i=1}^{n} f_i(w_i^t) - f(x^*)\right) + \lambda\gamma_t^2\frac{1}{n}\sum_{i=1}^{n}\left\|\partial f_i\left(w_i^t\right)\right\|_2^2 + \frac{1}{\lambda}\frac{1}{n}\sum_{i=1}^{n}\left\|w_i^t - x^t\right\|_2^2 \\
&\quad + \gamma_t^2\left\|\frac{1}{n}\sum_{i=1}^{n}\partial f_i\left(w_i^t\right)\right\|_2^2.
\end{aligned} \tag{95}$$

2.

For the second bound, we consider the definition of w_i^{t+1} from step (17) of Algorithm 2:

$$w_i^{t+1} = \begin{cases} x^{t+1} & \text{with probability } p, \\ w_i^t + \mathcal{Q}_i^t(x^{t+1} - x^t) & \text{with probability } 1-p. \end{cases} \tag{96}$$

Applying the variance decomposition (30) and tower property (28), we establish the second bound (89):

$$
\begin{aligned}
&\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\left[\left\|w_i^{t+1}-x^{t+1}\right\|_2^2 \mid x^t, W^t\right] \\
&\overset{(28)}{=} (1-p)\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\left[\left\|w_i^t+\mathcal{Q}_i(x^{t+1}-x^t)-x^{t+1}\right\|_2^2 \mid x^t, W^t\right] \\
&= (1-p)\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\left[\left\|w_i^t-x^t-\mathcal{Q}_i(x^{t+1}-x^t)-(x^{t+1}-x^t)\right\|_2^2 \mid x^t, W^t\right] \\
&\overset{(30)}{=} (1-p)\frac{1}{n}\sum_{i=1}^{n}\left\|w_i^t-x^t\right\|_2^2+(1-p)\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\left[\left\|\mathcal{Q}_i(x^{t+1}-x^t)-(x^{t+1}-x^t)\right\|_2^2 \mid x^t, W^t\right] \\
&\leq (1-p)\frac{1}{n}\sum_{i=1}^{n}\left\|w_i^t-x^t\right\|_2^2+(1-p)\omega\left\|x^{t+1}-x^t\right\|_2^2 \\
&\overset{(9)}{=} (1-p)\frac{1}{n}\sum_{i=1}^{n}\left\|w_i^t-x^t\right\|_2^2+(1-p)\omega\gamma_t^2\left\|\frac{1}{n}\sum_{i=1}^{n}\partial f_i\left(w_i^t\right)\right\|_2^2. \qquad (97)
\end{aligned}
$$

□

With these two key bounds established in Lemma 4, we can now proceed to the descent lemma. This lemma describes the one-step behavior of Algorithm 2 for any $\gamma_t > 0$ and will be crucial in establishing our convergence rates.

Lemma 5 (Descent lemma)**.** *Let the conditions of Lemma 4 hold. Define the Lyapunov function*

$$
V_\lambda^t := \left\|x^t-x^*\right\|_2^2+\frac{1}{\lambda p}\frac{1}{n}\sum_{i=1}^{n}\left\|w_i^t-x^t\right\|_2^2, \qquad (98)
$$

where $\lambda > 0$ is a constant. Then

$$
\begin{aligned}
\mathbb{E}\left[V_\lambda^{t+1} \mid x^t, W^t\right] \quad\leq\quad & V_\lambda^t-2\gamma_t\left(\frac{1}{n}\sum_{i=1}^{n}f_i(w_i^t)-f(x^*)\right)+\lambda\gamma_t^2\frac{1}{n}\sum_{i=1}^{n}\left\|\partial f_i\left(w_i^t\right)\right\|_2^2 \\
&+\gamma_t^2\left(1+\frac{(1-p)\omega}{p\lambda}\right)\left\|\frac{1}{n}\sum_{i=1}^{n}\partial f_i\left(w_i^t\right)\right\|_2^2. \qquad (99)
\end{aligned}
$$

Proof. Recall that Lemma 4 provides us with two key bounds:

1.
$$
\begin{aligned}
\mathbb{E}\left[\left\|x^{t+1}-x^*\right\|_2^2 \mid x^t, W^t\right] \quad\leq\quad & \left\|x^t-x^*\right\|_2^2-2\gamma_t\left(\frac{1}{n}\sum_{i=1}^{n}f_i(w_i^t)-f(x^*)\right)+\frac{1}{\lambda}\frac{1}{n}\sum_{i=1}^{n}\left\|w_i^t-x^t\right\|_2^2 \\
&+\lambda\gamma_t^2\frac{1}{n}\sum_{i=1}^{n}\left\|\partial f_i\left(w_i^t\right)\right\|_2^2+\gamma_t^2\left\|\frac{1}{n}\sum_{i=1}^{n}\partial f_i\left(w_i^t\right)\right\|_2^2, \qquad (100)
\end{aligned}
$$
where $\lambda > 0$;

2.
$$
\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\left[\left\|w_i^{t+1}-x^{t+1}\right\|_2^2 \mid x^t, W^t\right] \quad\leq\quad (1-p)\frac{1}{n}\sum_{i=1}^{n}\left\|w_i^t-x^t\right\|_2^2+(1-p)\omega\gamma_t^2\left\|\frac{1}{n}\sum_{i=1}^{n}\partial f_i\left(w_i^t\right)\right\|_2^2. \qquad (101)
$$

To obtain our descent lemma, we combine (100) with $\frac{1}{\lambda p}$ times (101):

$$\begin{aligned}
&\mathbb{E}\left[V_\lambda^{t+1} \mid x^t, W^t\right] \\
&\overset{(98)}{=} \mathbb{E}\left[\left\|x^{t+1}-x^*\right\|_2^2 \mid x^t, W^t\right] + \frac{1}{\lambda p}\frac{1}{n}\sum_{i=1}^n \mathbb{E}\left[\left\|w_i^{t+1}-x^{t+1}\right\|_2^2 \mid x^t, W^t\right] \\
&\leq \left\|x^t-x^*\right\|_2^2 - 2\gamma_t\left(\frac{1}{n}\sum_{i=1}^n f_i(w_i^t) - f(x^*)\right) + \frac{1}{\lambda}\frac{1}{n}\sum_{i=1}^n \left\|w_i^t - x^t\right\|_2^2 \\
&\quad +\lambda\gamma_t^2\frac{1}{n}\sum_{i=1}^n \left\|\partial f_i\left(w_i^t\right)\right\|_2^2 + \gamma_t^2\left\|\frac{1}{n}\sum_{i=1}^n \partial f_i\left(w_i^t\right)\right\|_2^2 \\
&\quad +\frac{1}{\lambda p}\left((1-p)\frac{1}{n}\sum_{i=1}^n \left\|w_i^t-x^t\right\|_2^2 + (1-p)\omega\gamma_t^2\left\|\frac{1}{n}\sum_{i=1}^n \partial f_i\left(w_i^t\right)\right\|_2^2\right) \qquad (102) \\
&= \left\|x^t-x^*\right\|_2^2 + \frac{1}{\lambda p}\frac{1}{n}\sum_{i=1}^n \left\|w_i^t-x^t\right\|_2^2 - 2\gamma_t\left(\frac{1}{n}\sum_{i=1}^n f_i(w_i^t) - f(x^*)\right) \\
&\quad +\lambda\gamma_t^2\frac{1}{n}\sum_{i=1}^n \left\|\partial f_i\left(w_i^t\right)\right\|_2^2 + \gamma_t^2\left(1+\frac{(1-p)\omega}{p\lambda}\right)\left\|\frac{1}{n}\sum_{i=1}^n \partial f_i\left(w_i^t\right)\right\|_2^2 \\
&\overset{(98)}{=} V_\lambda^t - 2\gamma_t\left(\frac{1}{n}\sum_{i=1}^n f_i(w_i^t) - f(x^*)\right) \\
&\quad +\lambda\gamma_t^2\frac{1}{n}\sum_{i=1}^n \left\|\partial f_i\left(w_i^t\right)\right\|_2^2 + \gamma_t^2\left(1+\frac{(1-p)\omega}{p\lambda}\right)\left\|\frac{1}{n}\sum_{i=1}^n \partial f_i\left(w_i^t\right)\right\|_2^2 .
\end{aligned} \qquad (103)$$

This completes the proof of the descent lemma. □

F.1 Proof of the Theorem 2

With the descent lemma established, we can now proceed to the main theoretical result of this paper. Before we state and prove the theorem, it is important to make a notational remark to avoid confusion.

Remark 3. *In Lemmas 4 and 5, we used an auxiliary term $\lambda > 0$ arising from the application of Young's inequality. This term also appeared in the definition of the Lyapunov function V_λ^t. In the following theorem, we will show how to choose this λ optimally and denote it as λ_*. Consequently, we define a Lyapunov function V^t such that $V^t := V_{\lambda_*}^t$. For simplicity of notation, we will use V^t instead of $V_{\lambda_*}^t$ in the theorem statement and proof.*

Now, let us restate and prove the main theorem.

Theorem 4 (Theorem 2)**.** *Let Assumptions 1, 2 and 3 hold. Define a Lyapunov function $V^t := \left\|x^t-x^*\right\|_2^2 + \frac{1}{\lambda_* p}\frac{1}{n}\sum_{i=1}^n \left\|w_i^t - x^t\right\|_2^2$, where $\lambda_* := \frac{\overline{L}_0}{\widetilde{L}_0}\sqrt{\frac{(1-p)\omega}{p}}$. Define also a constant $\widetilde{B}_* := \overline{L}_0^2 + 2\overline{L}_0\widetilde{L}_0\sqrt{\frac{(1-p)\omega}{p}}$. Let $\{w_i^t\}$ be the sequence produced by* MARINA-P *(Algorithm 2). Define $\overline{w}_i^T := \frac{1}{T}\sum_{t=0}^{T-1} w_i^t$ and $\widehat{w}_i^T := \frac{1}{\sum_{t=0}^{T-1}\gamma_t}\sum_{t=0}^{T-1}\gamma_t w_i^t$ for all $i \in [n]$.*

1. (Constant stepsize). *If $\gamma_t := \gamma > 0$, then*

$$\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^n f_i(\overline{w}_i^T) - f(x^*)\right] \leq \frac{V^0}{2\gamma T} + \frac{\widetilde{B}_*\gamma}{2}. \qquad (104)$$

If, moreover, the optimal γ is chosen, i.e.,

$$\gamma := \frac{1}{\sqrt{T}}\sqrt{\frac{V^0}{\widetilde{B}_*}}, \qquad (105)$$

then

$$\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n} f_i(\overline{w}_i^T) - f(x^*)\right] \le \frac{\sqrt{\widetilde{B}_* V^0}}{\sqrt{T}}. \tag{106}$$

2. (Polyak stepsize). *If γ_t is chosen as*

$$\gamma_t \quad := \quad \frac{\frac{1}{n}\sum_{i=1}^{n} f_i(w_i^t) - f(x^*)}{\left\|\frac{1}{n}\sum_{i=1}^{n}\partial f_i\left(w_i^t\right)\right\|_2^2 + 2\left\|\frac{1}{n}\sum_{i=1}^{n}\partial f_i\left(w_i^t\right)\right\|_2 \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left\|\partial f_i\left(w_i^t\right)\right\|_2^2}\sqrt{\frac{(1-p)\omega}{p}}}, \tag{107}$$

then

$$\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n} f_i(\overline{w}_i^T) - f(x^*)\right] \le \frac{\sqrt{\widetilde{B}_* V^0}}{\sqrt{T}}. \tag{108}$$

3. (Decreasing stepsize). *If γ_t is chosen as*

$$\gamma_t := \frac{\gamma_0}{\sqrt{t+1}}, \tag{109}$$

then

$$\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n} f_i(\overline{w}_i^T) - f(x^*)\right] \le \frac{V^0 + 2\gamma_0^2 \widetilde{B}_* \log(T+1)}{\gamma_0\sqrt{T}}. \tag{110}$$

If, moreover, the optimal γ_0 is chosen, i.e.,

$$\gamma_0 := \sqrt{\frac{V_0}{2\widetilde{B}_* \log(T+1)}}, \tag{111}$$

then

$$\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n} f_i(\overline{w}_i^T) - f(x^*)\right] \le 2\sqrt{2\widetilde{B}_* V_0}\sqrt{\frac{\log(T+1)}{T}}. \tag{112}$$

Proof. We will prove each part of the theorem separately, starting with some general bounds that will be useful throughout the proof.

From Assumption 3, we can infer that

$$\|\partial f_i(x)\|_2 \quad \le L_{0,i} \quad \forall x \in \mathbb{R}^d \text{ and } \forall i \in [n]. \tag{113}$$

This implies

$$\frac{1}{n}\sum_{i=1}^{n}\left\|\partial f_i\left(w_i^t\right)\right\|_2^2 \quad \le \quad \widetilde{L}_0^2, \quad \forall w_i^t \in \mathbb{R}^d \text{ and } i \in [n], \tag{114}$$

where $\widetilde{L}_0 := \sqrt{\frac{1}{n}\sum_{i=1}^{n} L_{0,i}^2}$, and

$$\left\|\frac{1}{n}\sum_{i=1}^{n}\partial f_i\left(w_i^t\right)\right\|_2 \quad \overset{(20)}{\le} \quad \frac{1}{n}\sum_{i=1}^{n}\left\|\partial f_i\left(w_i^t\right)\right\|_2 \le \overline{L}_0, \quad \forall w_i^t \in \mathbb{R}^d \text{ and } i \in [n], \tag{115}$$

where $\overline{L}_0 := \frac{1}{n}\sum_{i=1}^{n} L_{0,i}$.

Now, we proceed to prove each part of the theorem.

1. (Constant stepsize).

Using (114), (115), Lemma 5, the tower property of expectation (28), and choosing constant stepsize $\gamma_t := \gamma > 0$, we obtain

$$\begin{aligned}
&\mathbb{E}\left[V^{t+1}\right] \\
&\leq \mathbb{E}\left[V^t\right] - 2\gamma\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^n f_i(w_i^t) - f(x^*)\right] + \lambda\gamma^2\frac{1}{n}\sum_{i=1}^n \mathbb{E}\left[\left\|\partial f_i\left(w_i^t\right)\right\|_2^2\right] \\
&\quad + \left(1 + \frac{(1-p)\omega}{p\lambda}\right)\gamma^2\mathbb{E}\left[\left\|\frac{1}{n}\sum_{i=1}^n \partial f_i\left(w_i^t\right)\right\|_2^2\right] \\
&\overset{(114),(115)}{\leq} \mathbb{E}\left[V^t\right] - 2\gamma\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^n f_i(w_i^t) - f(x^*)\right] + \widetilde{B}_\lambda\gamma^2,
\end{aligned} \tag{116}$$

where $\widetilde{B}_\lambda := \lambda\widetilde{L}_0^2 + \overline{L}_0^2\left(1 + \frac{(1-p)\omega}{\lambda p}\right)$.

Since each f_i for all $i \in [n]$ is convex, by Jensen's inequality (31), we have

$$\begin{aligned}
\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^n f_i(\overline{w}_i^T) - f(x^*)\right] &\overset{(31)}{\leq} \mathbb{E}\left[\frac{1}{n}\sum_{i=1}^n \frac{1}{T}\sum_{t=0}^{T-1} f_i(w_i^t) - f(x^*)\right] \\
&\leq \frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^n f(w_i^t) - f(x^*)\right] \\
&\overset{(116)}{\leq} \frac{\mathbb{E}\left[V^0\right] - \mathbb{E}\left[V^T\right]}{2\gamma T} + \frac{\widetilde{B}_\lambda\gamma}{2} \\
&\overset{V^T\geq 0}{\leq} \frac{V^0}{2\gamma T} + \frac{\widetilde{B}_\lambda\gamma}{2}.
\end{aligned} \tag{117}$$

To optimize this bound, we need to find the optimal λ. Note that $\phi(\lambda) := \lambda\widetilde{L}_0^2 + \overline{L}_0^2\left(1 + \frac{(1-p)\omega}{\lambda p}\right)$ is a convex function on $(0, +\infty)$ for any fixed values $\widetilde{L}_0 > 0$, $\overline{L}_0 > 0$, $p \in (0,1]$, $\omega > 0$.

Therefore, we define the optimal λ value (denoted λ_*) as

$$\lambda_* := \arg\min_{\lambda > 0}\left(\lambda\widetilde{L}_0^2 + \overline{L}_0^2\left(1 + \frac{(1-p)\omega}{\lambda p}\right)\right) = \frac{\overline{L}_0}{\widetilde{L}_0}\sqrt{\frac{(1-p)\omega}{p}}. \tag{118}$$

Next, we define the optimal $\widetilde{B}$ value (denoted $\widetilde{B}_*$) as

$$\widetilde{B}_* := \widetilde{B}_{\lambda_*} = \lambda_*\widetilde{L}_0^2 + \overline{L}_0^2\left(1 + \frac{(1-p)\omega}{\lambda_* p}\right) = \overline{L}_0^2 + 2\overline{L}_0\widetilde{L}_0\sqrt{\frac{(1-p)\omega}{p}}. \tag{119}$$

Plugging (119) into (117), we get

$$\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^n f_i(\overline{w}_i^T) - f(x^*)\right] \overset{(119),(117)}{\leq} \frac{V^0}{2\gamma T} + \frac{\widetilde{B}_*\gamma}{2}. \tag{120}$$

Thus, we have established (104).

To derive the optimal rate (106), we need to find the optimal γ stepsize (which we denote γ_*):

$$\gamma_* := \arg\min_{\gamma}\left(\frac{V^0}{2\gamma T} + \frac{\widetilde{B}_*\gamma}{2}\right) = \frac{1}{\sqrt{T}}\sqrt{\frac{V^0}{\widetilde{B}_*}}. \tag{121}$$

Therefore, choosing $\gamma := \gamma_*$, (120) reduces to

$$\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^n f_i(\overline{w}_i^T) - f(x^*)\right] \leq \frac{V^0}{2\gamma_* T} + \frac{\widetilde{B}_*\gamma_*}{2} = \frac{\sqrt{V^0\widetilde{B}_*}}{\sqrt{T}}, \tag{122}$$

which gives us (106).

2. (Polyak stepsize). By Lemma 5, we have

$$\mathbb{E}\left[V^{t+1} \mid x^t, W^t\right] \leq V^t - 2\gamma_t\left(\frac{1}{n}\sum_{i=1}^n f_i(w_i^t) - f(x^*)\right) + \lambda\gamma_t^2\frac{1}{n}\sum_{i=1}^n \left\|\partial f_i\left(w_i^t\right)\right\|_2^2$$
$$+\gamma_t^2\left(1+\frac{(1-p)\omega}{p\lambda}\right)\left\|\frac{1}{n}\sum_{i=1}^n \partial f_i\left(w_i^t\right)\right\|_2^2. \tag{123}$$

We choose the Polyak stepsize γ_t as the one that minimizes the right-hand side of (123):

$$\begin{aligned}\gamma_t &:= \arg\min_\gamma\left\{V^t - 2\gamma\left(\frac{1}{n}\sum_{i=1}^n f_i(w_i^t) - f(x^*)\right) + \lambda\gamma^2\frac{1}{n}\sum_{i=1}^n \left\|\partial f_i\left(w_i^t\right)\right\|_2^2\right.\\
&\qquad\left.+\left(1+\frac{(1-p)\omega}{p\lambda}\right)\gamma^2\left\|\frac{1}{n}\sum_{i=1}^n \partial f_i\left(w_i^t\right)\right\|_2^2\right\}\\
&= \frac{\frac{1}{n}\sum_{i=1}^n f_i(w_i^t) - f(x^*)}{\lambda\frac{1}{n}\sum_{i=1}^n\left\|\partial f_i\left(w_i^t\right)\right\|_2^2 + \left(1+\frac{(1-p)\omega}{p\lambda}\right)\left\|\frac{1}{n}\sum_{i=1}^n \partial f_i\left(w_i^t\right)\right\|_2^2}.\end{aligned} \tag{124}$$

Note that the denominator in (124) is a convex function of λ. Therefore, similar to (118), we can choose the optimal λ as

$$\begin{aligned}\lambda_* &:= \arg\min_{\lambda>0}\left(\lambda\frac{1}{n}\sum_{i=1}^n\left\|\partial f_i\left(w_i^t\right)\right\|_2^2 + \left(1+\frac{(1-p)\omega}{p\lambda}\right)\left\|\frac{1}{n}\sum_{i=1}^n \partial f_i\left(w_i^t\right)\right\|_2^2\right)\\
&= \frac{\left\|\frac{1}{n}\sum_{i=1}^n \partial f_i\left(w_i^t\right)\right\|_2}{\sqrt{\frac{1}{n}\sum_{i=1}^n\left\|\partial f_i\left(w_i^t\right)\right\|_2^2}}\sqrt{\frac{(1-p)\omega}{p}},\end{aligned} \tag{125}$$

and thus

$$\begin{aligned}&\lambda_*\frac{1}{n}\sum_{i=1}^n\left\|\partial f_i\left(w_i^t\right)\right\|_2^2 + \left(1+\frac{(1-p)\omega}{p\lambda_*}\right)\left\|\frac{1}{n}\sum_{i=1}^n \partial f_i\left(w_i^t\right)\right\|_2^2\\
&= \left(\frac{\left\|\frac{1}{n}\sum_{i=1}^n \partial f_i\left(w_i^t\right)\right\|_2}{\sqrt{\frac{1}{n}\sum_{i=1}^n\left\|\partial f_i\left(w_i^t\right)\right\|_2^2}}\sqrt{\frac{(1-p)\omega}{p}}\right)\frac{1}{n}\sum_{i=1}^n\left\|\partial f_i\left(w_i^t\right)\right\|_2^2\\
&\quad+\left(1+\frac{(1-p)\omega}{p\left(\frac{\left\|\frac{1}{n}\sum_{i=1}^n \partial f_i\left(w_i^t\right)\right\|_2}{\sqrt{\frac{1}{n}\sum_{i=1}^n\left\|\partial f_i\left(w_i^t\right)\right\|_2^2}}\sqrt{\frac{(1-p)\omega}{p}}\right)}\right)\left\|\frac{1}{n}\sum_{i=1}^n \partial f_i\left(w_i^t\right)\right\|_2^2\\
&= \left\|\frac{1}{n}\sum_{i=1}^n \partial f_i\left(w_i^t\right)\right\|_2^2 + 2\left\|\frac{1}{n}\sum_{i=1}^n \partial f_i\left(w_i^t\right)\right\|_2\sqrt{\frac{1}{n}\sum_{i=1}^n\left\|\partial f_i\left(w_i^t\right)\right\|_2^2}\sqrt{\frac{(1-p)\omega}{p}}.\end{aligned} \tag{126}$$

Therefore, we derive the final expression for our Polyak stepsize:

$$\gamma_t := \frac{\frac{1}{n}\sum_{i=1}^n f_i(w_i^t) - f(x^*)}{\left\|\frac{1}{n}\sum_{i=1}^n \partial f_i\left(w_i^t\right)\right\|_2^2 + 2\left\|\frac{1}{n}\sum_{i=1}^n \partial f_i\left(w_i^t\right)\right\|_2\sqrt{\frac{1}{n}\sum_{i=1}^n\left\|\partial f_i\left(w_i^t\right)\right\|_2^2}\sqrt{\frac{(1-p)\omega}{p}}}. \tag{127}$$

Next, plugging (127) into (123) and using the tower property of expectation (28), we obtain

$$
\begin{aligned}
&\mathbb{E}\left[V^{t+1}\right] \\
&\overset{(123),(127)}{\leq} \mathbb{E}\left[V^t\right] - \mathbb{E}\left[\frac{\left(\frac{1}{n}\sum_{i=1}^n f_i(w_i^t) - f(x^*)\right)^2}{\left\|\frac{1}{n}\sum_{i=1}^n \partial f_i\left(w_i^t\right)\right\|_2^2 + 2\left\|\frac{1}{n}\sum_{i=1}^n \partial f_i\left(w_i^t\right)\right\|_2 \sqrt{\frac{1}{n}\sum_{i=1}^n \left\|\partial f_i\left(w_i^t\right)\right\|_2^2}\sqrt{\frac{(1-p)\omega}{p}}}\right] \\
&\overset{(114),(115),(119)}{\leq} \mathbb{E}\left[V^t\right] - \frac{\mathbb{E}\left[\left(\frac{1}{n}\sum_{i=1}^n f_i(w_i^t) - f(x^*)\right)^2\right]}{\widetilde{B}_*},
\end{aligned} \tag{128}
$$

where $\widetilde{B}_* \overset{(119)}{=} \overline{L}_0^2 + 2\overline{L}_0\widetilde{L}_0\sqrt{\frac{(1-p)\omega}{p}}$.

Since each f_i for all $i \in [n]$ is convex, by Jensen's inequality (31) and the Cauchy-Bunyakovsky-Schwarz inequality (29) with $X := \frac{1}{n}\sum_{i=1}^n f_i(w_i^t) - f(x^*)$ and $Y := 1$, we have

$$
\begin{aligned}
\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^n f_i(\overline{w}_i^T) - f(x^*)\right] &\overset{(31)}{\leq} \mathbb{E}\left[\frac{1}{n}\sum_{i=1}^n \frac{1}{T}\sum_{t=0}^{T-1} f_i(w_i^t) - f(x^*)\right] \\
&\leq \frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^n f_i(w_i^t) - f(x^*)\right] \\
&\overset{(29)}{\leq} \frac{1}{T}\sum_{t=0}^{T-1}\sqrt{\mathbb{E}\left[\left(\frac{1}{n}\sum_{i=1}^n f_i(w_i^t) - f(x^*)\right)^2\right]} \\
&\leq \sqrt{\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[\left(\frac{1}{n}\sum_{i=1}^n f_i(w_i^t) - f(x^*)\right)^2\right]} \\
&\overset{(128)}{\leq} \frac{\sqrt{\widetilde{B}}}{\sqrt{T}}\sqrt{\mathbb{E}\left[V^0\right] - \mathbb{E}\left[V^T\right]} \\
&\leq \frac{\sqrt{\widetilde{B}}\sqrt{V^0}}{\sqrt{T}}.
\end{aligned} \tag{129}
$$

Thus, we have established (108).

3. (Decreasing stepsize). By the same arguments as in the analysis for the constant stepsize case, we can get a bound

$$
\mathbb{E}\left[V^{t+1}\right] \overset{(119)}{\leq} \mathbb{E}\left[V^t\right] - 2\gamma_t\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^n f_i(w_i^t) - f(x^*)\right] + \widetilde{B}_*\gamma_t^2, \tag{130}
$$

where $\widetilde{B}_* \overset{(119)}{=} \overline{L}_0^2 + 2\overline{L}_0\widetilde{L}_0\sqrt{\frac{(1-p)\omega}{p}}$.

If $\gamma_t := \frac{\gamma_0}{\sqrt{t+1}}$ with $\gamma_0 > 0$, then we can get the bounds

$$
\sum_{t=0}^{T-1}\gamma_t \geq \frac{\gamma_0\sqrt{T}}{2}, \quad \text{and} \quad \sum_{t=0}^{T-1}\gamma_t^2 \leq 2\gamma_0^2\log(T+1). \tag{131}
$$

Since each f_i for all $i \in [n]$ is convex, by Jensen's inequality (31), we have

$$\begin{aligned}
\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n} f_i(\widehat{w}_i^T) - f(x^*)\right] &\overset{(31)}{\leq} \mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\frac{1}{\sum_{t=0}^{T-1}\gamma_t}\sum_{t=0}^{T-1}\gamma_t[f_i(w_i^t) - f(x^*)]\right] \\
&\overset{(130)}{\leq} \frac{\left(\mathbb{E}\left[V^0\right] - \mathbb{E}\left[V^T\right]\right) + \widetilde{B}_* \sum_{t=0}^{T-1}\gamma_t^2}{2\sum_{t=0}^{T-1}\gamma_t} \\
&\overset{V^T \geq 0}{\leq} \frac{V^0 + \widetilde{B}_* \sum_{t=0}^{T-1}\gamma_t^2}{2\sum_{t=0}^{T-1}\gamma_t} \\
&\overset{(131)}{\leq} \frac{V^0 + 2\gamma_0^2 \widetilde{B}_* \log(T+1)}{\gamma_0\sqrt{T}}.
\end{aligned} \tag{132}$$

The optimal γ_0 can be chosen by minimizing the right-hand side of (132), i.e.,

$$\gamma_* = \arg\min_{\gamma_0 > 0}\left(\frac{V_0}{\gamma_0\sqrt{T}} + \frac{2\gamma_0\widetilde{B}_*\log(T+1)}{\sqrt{T}}\right) = \sqrt{\frac{V_0}{2\widetilde{B}_*\log(T+1)}}, \tag{133}$$

Therefore, choosing $\gamma_0 := \gamma_*$, (132) reduces to

$$\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n} f_i(\widehat{w}_i^T) - f(x^*)\right] \leq \frac{V_0}{\gamma_*\sqrt{T}} + \frac{2\gamma_*\log(T+1)}{\sqrt{T}} = 2\sqrt{2V_0}\sqrt{\widetilde{B}_*}\sqrt{\frac{\log(T+1)}{T}}, \tag{134}$$

and we get (112). $\square$

Having established our main theorem, we can now derive a corollary that provides more practical insights into the performance of MARINA-P.

F.2 Proof of the Corollary 2

Corollary 4 (Corollary 2)**.** *Let the conditions of Theorem 2 be met, $p = \frac{\zeta_{\mathcal{Q}}}{d}$ and $w_i^0 = x^0$ for all $i \in [n]$. If γ_t is set according to* (13) *or* (14) *(constant or Polyak stepsizes) then* MARINA-P *(Algorithm 2) requires*

$$T = \mathcal{O}\left(\frac{R_0^2}{\varepsilon^2}\left(\overline{L}_0^2 + \overline{L}_0\widetilde{L}_0\sqrt{\omega\left(\frac{d}{\zeta_{\mathcal{Q}}} - 1\right)}\right)\right) \tag{135}$$

iterations/communication rounds in order to achieve $\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n} f_i(\overline{w}_i^T) - f(x^*)\right] \leq \varepsilon$. Moreover, under the assumption that the communication cost is proportional to the number of non-zero components of vectors transmitted from the server to workers, we have that the expected total communication cost per worker equals

$$\mathcal{O}\left(d + \frac{\widetilde{L}_0^2 R_0^2 \zeta_{\mathcal{Q}}}{\varepsilon^2}\left(1 + \sqrt{\omega\left(\frac{d}{\zeta_{\mathcal{Q}}} - 1\right)}\right)\right). \tag{136}$$

Proof. From (106) and (108), we have the convergence rate

$$\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n} f_i(\overline{w}_i^T) - f(x^*)\right] \leq \frac{\sqrt{\widetilde{B}_* V^0}}{\sqrt{T}}, \tag{137}$$

where $V^0 = \left\|x^0 - x^*\right\|_2^2 + \frac{1}{\lambda_* p}\frac{1}{n}\sum_{i=1}^{n}\left\|w_i^0 - x^0\right\|_2^2$ and $\widetilde{B}_* = \overline{L}_0^2 + 2\overline{L}_0\widetilde{L}_0\sqrt{\frac{(1-p)\omega}{p}}$, resulting in a complexity

$$T = \mathcal{O}\left(\frac{\widetilde{B}_* V^0}{\varepsilon^2}\right) \tag{138}$$

required to achieve $\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n} f_i(\overline{w}_i^T) - f(x^*)\right] \leq \varepsilon$.

Assuming $w_i^0 = x^0$ for all $i \in [n]$ and $p = \frac{\zeta_{\mathcal{Q}}}{d}$, we get

$$V^0 = R_0^2 = \left\|x^0 - x^*\right\|_2^2 \quad \text{and} \quad \widetilde{B}_* = \overline{L}_0^2 + \overline{L}_0 \widetilde{L}_0 \sqrt{\omega\left(\frac{d}{\zeta_{\mathcal{Q}}} - 1\right)}. \tag{139}$$

Plugging (139) into (138), we get (135).

The expected total communication cost per worker is

$$\begin{aligned} d + (dp + \zeta_{\mathcal{Q}}(1-p))T &\leq d + \frac{\widetilde{L}_0^2 R_0^2}{\varepsilon^2}(dp + \zeta_{\mathcal{Q}}(1-p))\left(1 + 2\sqrt{\frac{(1-p)\omega}{p}}\right) \\ &= d + \frac{\widetilde{L}_0^2 R_0^2}{\varepsilon^2}(dp + \zeta_{\mathcal{Q}}(1-p))\left(1 + 2\sqrt{\omega\left(\frac{d}{\zeta_{\mathcal{Q}}} - 1\right)}\right) \\ &\leq d + \frac{2\widetilde{L}_0^2 R_0^2}{\varepsilon^2}\zeta_{\mathcal{Q}}\left(1 + 2\sqrt{\omega\left(\frac{d}{\zeta_{\mathcal{Q}}} - 1\right)}\right) \\ &= \mathcal{O}\left(d + \frac{\widetilde{L}_0^2 R_0^2 \zeta_{\mathcal{Q}}}{\varepsilon^2}\left(1 + \sqrt{\omega\left(\frac{d}{\zeta_{\mathcal{Q}}} - 1\right)}\right)\right), \end{aligned} \tag{140}$$

where we used the bound $p + \zeta_{\mathcal{Q}}(1-p) \leq 2\zeta_{\mathcal{Q}}$. □

G CONCLUSION

In this paper, we have presented a comprehensive analysis of distributed non-smooth optimization with server-to-worker compression. We extended EF21-P to the distributed setting and introduced a non-smooth version of MARINA-P, providing theoretical guarantees for both algorithms under constant, decreasing, and Polyak stepsizes. To the best of our knowledge, this work presents the first theoretical results for distributed non-smooth optimization that incorporate server-to-worker compression and adaptive stepsizes. Our empirical studies demonstrate the superior performance of MARINA-P with correlated compressors in non-smooth settings.

USE OF LARGE LANGUAGE MODELS

In the preparation of this manuscript, a Large Language Model (LLM) was utilized as a general-purpose assistive tool. The specific applications of the LLM were limited to the following:

- **Main Text Correction:** The LLM was employed to enhance the clarity, grammar, and style of the main text. This involved proofreading for typographical errors, improving sentence structure, and ensuring overall readability.
- **Programming Assistance:** The LLM served as a supportive tool for various programming tasks. This included assistance with code generation for standard functions, debugging, and the exploration of different implementation strategies.