Byzantine Machine Learning Made Easy By Resilient Averaging of Momentums

Sadegh Farhadkhani[*1] **Rachid Guerraoui**[*1] **Nirupam Gupta**[*1] **Rafael Pinot**[*1] **John Stephan**[*1]

Abstract

Byzantine resilience emerged as a prominent topic within the distributed machine learning community. Essentially, the goal is to enhance distributed optimization algorithms, such as distributed SGD, in a way that guarantees convergence despite the presence of some misbehaving (a.k.a., *Byzantine*) workers. Although a myriad of techniques addressing the problem have been proposed, the field arguably rests on fragile foundations. These techniques are hard to prove correct and rely on assumptions that are (a) quite unrealistic, i.e., often violated in practice, and (b) heterogeneous, i.e., making it difficult to compare approaches.

We present *RESAM (RESilient Averaging of Momentums)*, a unified framework that makes it simple to establish optimal Byzantine resilience, relying only on standard machine learning assumptions. Our framework is mainly composed of two operators: *resilient averaging* at the server and *distributed momentum* at the workers. We prove a general theorem stating the convergence of distributed SGD under RESAM. Interestingly, demonstrating and comparing the convergence of many existing techniques become direct corollaries of our theorem, without resorting to stringent assumptions. We also present an empirical evaluation of the practical relevance of RESAM.

1. Introduction

The vast amount of data collected every day, combined with the increasing complexity of machine learning models, has led to the emergence of distributed learning schemes (Abadi et al., 2015; Kairouz et al., 2021). In the now classical parameter server distributed architecture, the learning procedure consists of multiple data owners (or *workers*) collaborating to build a global model with the help of a central entity (the *parameter server*), typically using the celebrated distributed stochastic gradient descent (SGD) algorithm (Tsitsiklis et al., 1986; Bertsekas & Tsitsiklis, 2015). The server essentially maintains an estimate of the model parameter which is updated iteratively using the *average* of the *stochastic gradients* computed by the workers.

Nevertheless, this algorithm is vulnerable to "misbehaving" workers that could (either intentionally or inadvertently) sabotage the learning by sending arbitrarily bad gradients to the server (Feng et al., 2015; Su & Vaidya, 2016). These workers are commonly referred to as *Byzantine* (Lamport et al., 1982). To address this critical issue, a large body of research has been devoted to adapting distributed SGD to make it converge despite the presence of (a fraction of) Byzantine workers, e.g., (Blanchard et al., 2017; Chen et al., 2017; El Mhamdi et al., 2018; Yin et al., 2018; Xie et al., 2018; Alistarh et al., 2018; Diakonikolas et al., 2019b; Allen-Zhu et al., 2020; Prasad et al., 2020; Karimireddy et al., 2021). The general idea consists in replacing the averaging step of the algorithm with a *robust aggregation rule*, basically seeking to filter out the bad gradients.

Demonstrating the correctness of the resulting algorithms reveals however very challenging, and previous works rely on unusual assumptions. For instance, a large body of work assumes stochastic gradients that follow a specific distribution, e.g., sub-Gaussian/exponential (Chen et al., 2017; Feng et al., 2017; Yin et al., 2018; Prasad et al., 2020). Some approaches rely on stronger assumptions that are not even satisfied by a Gaussian distribution, such as *almost surely absolutely boundedness* (Alistarh et al., 2018; Diakonikolas et al., 2019b; Allen-Zhu et al., 2020), or *vanishing variance* (Blanchard et al., 2017; Xie et al., 2018; El Mhamdi et al., 2018; 2021a). Indeed, these assumptions are often violated in practice, resulting in the failure of these approaches when some workers behave maliciously (Baruch et al., 2019; Xie et al., 2019a). Ultimately, the considerable difference in these assumptions from one approach to another makes it quite difficult to compare the underlying techniques. Byz

[1]Distributed Computing Laboratory (DCL), School of Computer and Communication Sciences, École Polytechnique Fédérale de Lausanne (EPFL), Switzerland. Correspondence to: Nirupam Gupta <nirupam.gupta@epfl.ch>, Rafael Pinot <rafael.pinot@epfl.ch>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

In short, whilst Byzantine resilience is considered crucial to establish robustness in distributed machine learning, the field arguably rests on fragile foundations.

1.1. Our Contributions

We present **RESAM** *(RESilient Averaging of Momentums)*, a general framework for studying Byzantine resilience in distributed machine learning under minimal assumptions: (1) *unbiased* stochastic gradients with *bounded variance* and (2) first-order *Lipschitz smoothness*.[1] RESAM integrates two main components within distributed SGD, namely *resilient averaging* and *distributed momentum*.

(a) We introduce resilient averaging as a new elementary criterion of robustness for aggregation rules. It can be verified in an off-line manner and is readily satisfied by many existing schemes, under classical assumptions. It also standardizes the way to measure the robustness of aggregation rules through a parameter λ, that we call the *resilience coefficient*.

(b) We make use of distributed momentum which adapts the notion of gradient momentum (Polyak, 1964) to distributed architectures. Specifically, at each step of the algorithm, honest (i.e., non-Byzantine) workers send the momentums of their stochastic gradients to the server, instead of simply sending their gradients.

Byzantine resilience. We prove a general theorem establishing finite-time convergence of distributed SGD enhanced through RESAM. As an immediate corollary, we make the following contributions.

(a) We show (for the first time) the Byzantine resilience of several existing schemes, without resorting to non-standard assumptions. Our result holds as long as the Byzantine workers represent less than $1/2$ of the system, which is optimal (Alistarh et al., 2018).

(b) We precisely characterize the convergence rates of these schemes through our framework, enabling comparison of their performances on a common theoretical ground. Essentially, our analysis indicates that using aggregation rules with smaller resilience coefficient λ results in faster convergence.

Technical significance. A key observation that enables us to prove our theorem is that the momentums of honest workers' gradients *converge toward one another* as the learning algorithm proceeds. This significantly mitigates the impact of Byzantine workers when using a resilient averaging rule. The caveat is that the conventional techniques used for analyzing the convergence of SGD do not readily apply, since the honest workers' momentums *deviate* from the true gradient. To overcome this challenge, we devise a proof technique based on a *novel Lyapunov function* which we also believe to be of independent interest to the distributed optimization community.

Practical relevance. We report on a comprehensive set of experiments evaluating RESAM on benchmark image classification tasks: MNIST, Fashion-MNIST, and CIFAR-10. We simulate Byzantine behavior using 4 state-of-the-art attacks. We observe that the algorithm works best when combining resilient averaging and distributed momentum, but performs poorly against some attacks when using only one of these notions. This advocates that the combination proposed by RESAM is critical to Byzantine resilience.

1.2. Closely Related Work

We present below comparisons to closely related work.

Resilient averaging. Whilst the robustness criterion of *C-averaging agreement* introduced in (El Mhamdi et al., 2021a) shares similarities with our notion of resilient averaging, it is studied under a non-standard setting where the batch size is monotonically increased over the iterations to ensure *vanishing variance* of the stochastic gradients (and without exploiting the power of distributed momentum). Our notion of resilient averaging should also not be confused with the notion of resilience introduced by (Steinhardt et al., 2018), for the latter is an assumption on the distribution of honest workers' gradients. Our notion, on the other hand, is a criterion that can be satisfied by an aggregation rule regardless of the distribution of the workers' gradients.

Distributed momentum. The first paper to discuss the usefulness of distributed momentum for boosting Byzantine resilience in distributed machine learning is (El Mhamdi et al., 2021b). Essentially, the paper observes through an extensive set of experiments that distributed momentum helps *some* robustness techniques counter two state-of-the art attacks, namely *little* (Baruch et al., 2019) and *empire* (Xie et al., 2019a). However, the work lacks concrete theoretical explanations. Moreover, our experimental findings go beyond (El Mhamdi et al., 2021b) by considering a wider range of attacks and robustness techniques. Another related work (Karimireddy et al., 2021) attempts to formally demonstrate that distributed momentum grants provable Byzantine resilience to the robustness technique they devise, called *centered clipping* (CC). While the proof relies on standard assumptions, the algorithm requires prior knowledge on the variance of the gradients, which is quite impractical. Furthermore, their result only holds for small fractions of Byzantine workers less than $1/9.7$, which is clearly sub-optimal.

[1] These assumptions are elementary for analyzing SGD, even in the non-Byzantine setting (Bottou et al., 2018), and are used in all prior works on Byzantine resilience.

1.3. Paper Outline

Section 2 formally presents the problem of Byzantine resilience in distributed learning. Section 3 introduces RESAM. Section 4 presents our main theorem and its corollary showing resilience of some prominent existing approaches. Section 5 presents our experimental results. Section 6 provides additional related work and discussions. Due to space constraints, we defer proofs to appendices A, B, and C.

2. Problem Statement

We consider the parameter server architecture with n workers $w_1, \ldots, w_n$, and a trusted central server. The workers only communicate with the server and there is no inter-worker communication. We let $\mathcal{D}$ be an unknown data distribution. For a given parameter $\theta \in \mathbb{R}^d$, a data point $x \sim \mathcal{D}$ has a real-valued loss function $q(\theta, x)$. The server aims to compute, by collaborating with the workers, a parameter θ^* minimizing the expected loss function $Q(\theta)$ defined to be

$$Q(\theta) = \mathbb{E}_{x \sim \mathcal{D}} \left[q(\theta, x) \right] \quad \forall \theta \in \mathbb{R}^d. \tag{1}$$

We assume Q to be differentiable and to have a minimum, i.e., $\min_{\theta \in \mathbb{R}^d} Q(\theta)$ exists and has a finite value. However, as the loss function Q could be non-convex, e.g., when considering deep neural networks, solving the above optimization problem may be NP-hard (Boyd et al., 2004). Thus, a more reasonable goal is to compute a critical point of Q, i.e., θ^* such that $\|\nabla Q(\theta^*)\| = 0$ where ∇Q denotes the *gradient* of Q and $\|\cdot\|$ the Euclidean norm on $\mathbb{R}^d$.

2.1. Vanilla Distributed SGD

The traditional way to solve this learning problem is through a distributed implementation of the classical stochastic gradient descent (SGD) method (Bertsekas & Tsitsiklis, 2015). This is an iterative algorithm where, in each step t, the server maintains a parameter vector θ_t which is broadcast to all the workers. Each worker w_i then returns an *unbiased* stochastic estimate $g_t^{(i)}$ of the gradient $\nabla Q(\theta_t)$. Specifically,

$$g_t^{(i)} = \nabla Q(\theta_t) + u_t^{(i)}, \tag{2}$$

where $u_t^{(i)}$ is the realization of a random vector $U(\theta_t)$, defined over $\mathbb{R}^d$, that characterizes the *noise* in the gradient computation at θ_t.[2] Ultimately, the server updates θ_t by using the average of the received gradients as follows,

$$\theta_{t+1} = \theta_t - \gamma_t \frac{1}{n} \sum_{i=1}^{n} g_t^{(i)}, \tag{3}$$

where $\gamma_t \geq 0$ is referred to as the *learning rate* at step t.

[2]The noise $U(\theta_t)$ is usually assumed to be a result of sampling data points from $\mathcal{D}$. However, to keep our discussion more general, we let $U(\theta_t)$ follow any distribution subject to Assumption 2.

2.2. Classical Assumptions

When all the workers are honest, i.e., they follow the prescribed instructions correctly, the above iterative algorithm provably converges to a critical point of function Q, under the following assumptions.

Assumption 1 (Lipschitz smooth loss function)**.** *There exists $L < \infty$ such that for all θ, $\theta' \in \mathbb{R}^d$,*

$$\left\| \nabla Q(\theta) - \nabla Q(\theta') \right\| \leq L \left\| \theta - \theta' \right\|.$$

Assumption 2 (Unbiased gradients with bounded variance)**.** *For all $\theta \in \mathbb{R}^d$, the random vector $U(\theta)$ characterizing the gradient noise at θ is such that $\mathbb{E}\left[U(\theta)\right] = 0$, and there exists $\sigma < \infty$ such that $\mathbb{E}\left[\|U(\theta)\|^2\right] \leq \sigma^2$.*

These assumptions are indeed satisfied in many learning problems (Ghadimi & Lan, 2013; Bottou et al., 2018).

2.3. Byzantine Resilience

We study a scenario where up to f workers of *unknown identities* may be *Byzantine* (Lamport et al., 1982). Such workers may send arbitrarily incorrect information to the server, preventing it from solving the learning problem (Su & Vaidya, 2016). The goal is then to design a learning algorithm that computes a critical point despite the fact that a fraction of the workers may be Byzantine. Formally, given f and a real value $\epsilon > 0$, we aim to design an (f, ϵ)-*resilient* algorithm, as defined below.

Definition 1 ((f, ϵ)**-Resilience).** *A distributed learning algorithm is said to be* (f, ϵ)-resilient *if, despite the presence of up to f Byzantine workers, it enables the server to output a learning parameter $\widehat{\theta}$ such that*

$$\mathbb{E}\left[\left\| \nabla Q\left(\widehat{\theta}\right) \right\|^2 \right] \leq \epsilon,$$

where $\mathbb{E}\left[\cdot\right]$ is defined over the randomness of the algorithm. Moreover, an algorithm is said to be optimally resilient *if it is (f, ϵ)-resilient for any $f < n/2$ and $\epsilon > 0$.*

A standard approach to confer Byzantine resilience to distributed SGD is to replace the simple averaging of the workers' gradients at the server by a more sophisticated aggregation rule that seeks to mitigate the adversarial impact of any incorrect information sent by the Byzantine workers. In particular, consider an aggregation rule $F : \mathbb{R}^{d \times n} \to \mathbb{R}^d$. Then, at every step t the server updates θ_t as follows:

$$\theta_{t+1} = \theta_t - \gamma_t F\left(g_t^{(1)}, \ldots, g_t^{(n)}\right). \tag{4}$$

Note that the gradient $g_t^{(i)}$ of any Byzantine worker w_i need not follow (2) and may take arbitrary values.

Aggregation rule	MDA	CWTM	MeaMed	Krum*	GM	CWMed	**Lower bound**
λ	$\frac{2f}{n-f}$	$\frac{f}{n-f}\Delta$	$\frac{2f}{n-f}\Delta$	$1+\sqrt{\frac{n-f}{n-2f}}$	$1+\frac{n-f}{\sqrt{(n-2f)n}}$	$\frac{n}{2(n-f)}\Delta$	$\frac{f}{n-f}$

$$\Delta \coloneqq \min\{2\sqrt{n-f}, \sqrt{d}\}$$

Table 1. Resilience coefficients λ for various aggregation rules satisfying Definition 2, when $f < n/2$. Note that the lower bound for λ is $f/n-f$. Thus, MDA has an order-optimal coefficient (differs from the lower bound only by a constant factor).

Some notable aggregation rules. In this paper, we consider a wide range of aggregation rules: Krum* 2017,[3] *geometric median* (GM) 2017, *minimum diameter averaging* (MDA) 2018, *coordinate-wise trimmed mean* (CWTM) 2018, *coordinate-wise median* (CWMed) 2018, *mean-around-median* (MeaMed) 2018, *centered clipping* (CC) 2021, and *comparative gradient elimination* (CGE) 2021. We refer the interested reader to Appendix C for a detailed description of these aggregation rules.

3. RESilient Averaging of Momentums

Our framework incorporates the notions of *resilient averaging* and *distributed momentum* in distributed SGD. We first recall distributed momentum, followed by the introduction of resilient averaging. Finally, we present the skeleton of a learning algorithm within RESAM.

3.1. Distributed Momentum

At each step t of this scheme, upon receiving the current learning parameter θ_t from the server, each honest worker w_i returns the *Polyak's momentum* of its stochastic gradient (Polyak, 1964). This momentum is defined as

$$m_t^{(i)} = \beta m_{t-1}^{(i)} + (1-\beta) g_t^{(i)}, \tag{5}$$

where $m_0^{(i)} = 0$ by convention, $\beta \in [0, 1)$, and $g_t^{(i)}$ is as defined in (2). We refer to β as the *momentum coefficient*. Recall that for a Byzantine worker w_i, the momentum $m_t^{(i)}$ need not follow (5). Upon receiving workers' momentums, the server applies the aggregation rule F to update the parameter θ_t. Specifically, the server computes

$$\theta_{t+1} = \theta_t - \gamma_t F\left(m_t^{(1)}, \ldots, m_t^{(n)}\right). \tag{6}$$

Remark 1. *Distributed momentum differs from its centralized counterpart in that the momentum operation in the former is performed by the workers, unlike in the latter where it is applied by the server after aggregating the gradients.*

3.2. Resilient Averaging

The idea behind the notion of resilient averaging is to ensure that the distance between the result of the aggregation rule and the average of honest workers' momentums is bounded by their *diameter* times a factor λ. We refer to λ as the *resilience coefficient*. Essentially, smaller the λ better the resilience. We formally define this notion below.

Definition 2 (**(f, λ)-Resilient averaging).** *For $f < n$ and real value $\lambda \geq 0$, an aggregation rule F is called* (f, λ)-resilient averaging *if for any collection of n vectors $x_1, \ldots, x_n$, and any set $S \subseteq \{1, \ldots, n\}$ of size $n-f$,*

$$\|F(x_1, \ldots, x_n) - \overline{x}_S\| \leq \lambda \max_{i,j \in S} \|x_i - x_j\|$$

where $\overline{x}_S \coloneqq \frac{1}{|S|}\sum_{i\in S} x_i$, and $|S|$ is the cardinality of S.

Salient features. Resilient averaging is a simple robustness criterion that is verifiable in an off-line manner, i.e., independently of the dynamics of the learning algorithm. Moreover, this criterion is so elementary that it can be satisfied by a wide class of state-of-the-art aggregation rules under only standard assumptions. This makes it possible to study and compare their resilience properties on a common theoretical ground. Indeed, we show (in Proposition 1 below) that all the aggregation rules mentioned in Section 2.3 satisfy this criterion, except CC and CGE that we discuss separately.

Proposition 1. *Consider an aggregation rule $F \in \{\text{MDA}, \text{CWTM}, \text{MeaMed}, \text{Krum}^*, \text{GM}, \text{CWMed}\}$. For any $f < n/2$, there exists a resilience coefficient λ for which F is (f, λ)-resilient averaging.*

We list in Table 1 the respective values of λ for several aggregation rules satisfying Definition 2. Formal derivations of these coefficients can be found in Appendix C. It is worth noting that an (f, λ)-resilient averaging rule cannot have a resilience coefficient smaller than $f/n-f$ (Lower bound in Table 1). Accordingly, the resilience coefficient we compute for MDA is *order-optimal*, i.e., it differs from the lower bound by a constant factor.

Sanity check. When the inputs of the honest workers are identical, the output of an (f, λ)-resilient averaging rule is equal to their inputs from Definition 2 (as the diameter of at least $n-f$ inputs is null). This simple yet important sanity check guarantees that when the gradients of honest workers are computed without uncertainty (i.e., $U(\theta)$ is null for all $\theta \in \mathbb{R}^d$) the aggregation rule mimics the majority voting scheme, which is known to be optimal when there is

[3] Krum* is a variant of Krum, described in Appendix C.5.

no uncertainty in the correct responses (Lynch, 1996). Note that satisfying this sanity check is a necessary condition for being (f, λ)-resilient averaging.

The cases of CGE and CC. When studying existing rules we encountered two special cases, namely CGE and CC. While CGE clearly does not satisfy the condition of resilient averaging, CC may only satisfy it approximately. Besides, CC uses a *clipping parameter* that requires a priori knowledge on σ, and an initial guess on the average of the honest vectors with *known* bounded error. These are impractical requirements that are not needed by other rules we consider. As it is unclear whether CC can satisfy our definition under the classical assumptions, in the remaining we adopt an agnostic point of view assuming that it does not.

3.3. Skeleton of an Algorithm within RESAM

The overall learning procedure combining distributed momentum and a resilient averaging rule is captured in Algorithm 1, presented below.

Algorithm 1: Distributed SGD using distributed momentum and an (f, λ)-resilient averaging rule F

Initialization: **Server** chooses an arbitrary initial parameter vector $\theta_1 \in \mathbb{R}^d$, a set of T learning rates $\{\gamma_1, \ldots, \gamma_T\}$, a deterministic aggregation rule $F : \mathbb{R}^{d \times n} \to \mathbb{R}^d$, and sends the momentum coefficient $\beta \in [0, 1)$ to all the workers. Each **honest worker** w_i sets its initial momentum $m_0^{(i)} = 0$.

Algorithm's body: In each **step** $t = 1, \ldots, T$.

1. **Server** broadcasts θ_t to all workers.
2. Each **honest worker** w_i sends to the server the momentum $m_t^{(i)}$ defined by (5), i.e., $m_t^{(i)} = \beta m_{t-1}^{(i)} + (1-\beta) g_t^{(i)}$ where $g_t^{(i)}$ is a stochastic gradient as defined in (2). *(A Byzantine worker w_i may send an arbitrary value for its "momentum" $m_t^{(i)}$.)*
3. **Server** updates the parameter vector as per (6), i.e., $\theta_{t+1} = \theta_t - \gamma_t F\left(m_t^{(1)}, \ldots, m_t^{(n)}\right)$.

Output: **Server** outputs a learning parameter $\widehat{\theta}$ chosen randomly from the set $\{\theta_1, \ldots, \theta_T\}$.

4. General Convergence Theorem

We present below our main technical result demonstrating the convergence of Algorithm 1 when up to f workers may be Byzantine. Then, as an immediate corollary, we derive the (f, ϵ)-resilience property of the algorithm. Formal proofs of the results are deferred to appendices A and B.

4.1. Formal Statements

We first present our main result in Theorem 1 below. Essentially, we analyze Algorithm 1 upon assuming a sufficient small constant learning rate γ_t for all steps t, provided that assumptions 1 and 2 hold true. For simplified presentation of our formal results, we introduce the following notation.

$$Q^* \coloneqq \min_{\theta \in \mathbb{R}^d} Q(\theta),$$

$$a_o \coloneqq 4\left(2\left(Q(\theta_1) - Q^*\right) + \frac{1}{8L}\left\|\nabla Q\left(\theta_1\right)\right\|^2\right),$$

$$a_1 \coloneqq 6912L, \quad \text{and} \quad a_2 \coloneqq 288L.$$

Theorem 1. *Consider Algorithm 1 with an (f, λ)-resilient averaging rule and a constant learning rate γ, i.e., $\gamma_t = \gamma$, $\forall t$ where*

$$\gamma = \left(\sqrt{\frac{a_o(n-f)}{a_1\lambda^2(n-f)^2 + a_2}}\right)\frac{1}{\sigma\sqrt{T}}.$$

If $T \geq \frac{a_o L}{12\sigma^2\lambda^2(n-f)}$ and $\beta = \sqrt{1 - 24\gamma L}$, then

$$\mathbb{E}\left[\left\|\nabla Q\left(\widehat{\theta}\right)\right\|^2\right] \leq 2\sqrt{\left(a_1\lambda^2(n-f) + \frac{a_2}{n-f}\right)\frac{a_o\sigma^2}{T}} + \left(\frac{a_2\sigma}{n-f}\right)\left(\sqrt{\frac{a_o(n-f)}{a_1\lambda^2(n-f) + a_2}}\right)\frac{1}{T^{3/2}}.$$

Idea of the Proof. Recall that $m_t^{(i)}$ denotes the momentum of worker w_i at step t. Below, we denote by $\overline{m}_t$ the average momentum of all the honest workers at step t. Our proof of Theorem 1 rests on two key observations, detailed below.

(a) At every step t of Algorithm 1, the growth of the loss function (i.e., $Q(\theta_{t+1}) - Q(\theta_t)$) depends positively on both the *drift* of each honest worker w_i (i.e., $m_t^{(i)} - \overline{m}_t$) and the *deviation* of the honest workers from the true gradient (i.e., $\overline{m}_t - \nabla Q(\theta_t)$). Essentially, to prove convergence, we need the accumulation of both the drift and the deviation to be inversely proportional to T, when scaled by the learning rate γ.

(b) Upon analyzing these two quantities separately, we observe that whilst increasing the momentum coefficient β decreases the accumulation of drift, it increases the accumulation of deviation. Hence, we need to carefully determine an appropriate value for β to establish

Aggregation rule	MDA	CWTM	MeaMed	Krum*	GM	CWMed
$\epsilon \in \mathcal{O}\left(\sqrt{\frac{\sigma^2}{T}\left(\frac{1}{n-f}+\boldsymbol{\kappa}\right)}\right)$	$\frac{f^2}{(n-f)}$	$\frac{f^2}{(n-f)}\Delta^2$	$\frac{f^2}{(n-f)}\Delta^2$	$\frac{(n-f)^2}{n-2f}$	$\frac{(n-f)^3}{(n-2f)n}$	$\frac{n^2}{(n-f)}\Delta^2$

$$\vartheta \in (0,\, 1) \;\text{ and }\; \Delta \coloneqq \min\{2\sqrt{n-f}, \sqrt{d}\}$$

Table 2. Rates of convergence, as defined in Corollary 1, for several (f, λ)-resilient averaging rules when $f < n/2$. Note that the rates only differ in the $\boldsymbol{\kappa}$ term. For simplicity, we only present the values of $\boldsymbol{\kappa}$ for the different rules.

the convergence of Algorithm 1. However, the traditional Lyapunov function of $\mathbb{E}\left[Q(\theta_t)\right]$ turns out to be inadequate for solving this problem.

To address this issue, we devise a novel Lyapunov function

$$V_t \coloneqq \mathbb{E}\left[2Q(\theta_t) + \frac{1}{8L^2}\left\|\overline{m}_t - \nabla Q(\theta_t)\right\|^2\right].$$

By analyzing the growth of V_t along the steps of Algorithm 1, we show that setting the momentum coefficient $\beta = \sqrt{1-24\gamma L}$ yields the stated finite-time convergence. Note that this momentum coefficient is well defined (i.e., it belongs to $[0,\, 1)$) as soon as $T \geq \frac{a_o L}{12\sigma^2\lambda^2(n-f)}$, which explains the condition on T in Theorem 1. □

Using Theorem 1, we can show that Algorithm 1 is $(f,\, \epsilon)$-resilient. Specifically, by ignoring the higher-order term in T, and the constants, we obtain the following corollary.

Corollary 1. *Suppose that assumptions 1 and 2 hold true. Then, Algorithm 1 with an (f, λ)-resilient averaging rule, and parameters γ_t, T and β as defined in Theorem 1, is is (f, ϵ)-resilient with*

$$\epsilon \in \mathcal{O}\left(\sqrt{\frac{\sigma^2}{T}\left(\frac{1}{n-f}+\lambda^2(n-f)\right)}\right).$$

Basically, we can obtain an arbitrarily small ϵ if the algorithm is run for a sufficiently large number of steps. In particular, we can use Corollary 1 to determine, for any $f < n/2$ and $\epsilon > 0$, the number of steps T and the momentum coefficient β for which Algorithm 1 is (f, ϵ)-resilient, for any of the six aggregation rules listed in Table 1. This shows that, by Definition 1, Algorithm 1 is *optimally resilient* for any of these rules. In Table 2, we summarize the rates of convergence (i.e., order of ϵ) for the aggregation rules we consider. These rates are simply computed by substituting in Corollary 1 the values of λ from Table 1.

4.2. Analysis & Discussion

Impact of the fraction of Byzantine workers. From Table 2 we note that the order of ϵ grows proportionally to f/n for all the aggregation rules listed, except for CWMed. Basically, a smaller fraction of Byzantine workers enables faster convergence to Algorithm 1 when using an appropriate resilience averaging rule.

Comparison of convergence rates. The rate of convergence of Algorithm 1, shown in Corollary 1, matches that of vanilla distributed SGD (Lei et al., 2019) in terms of the total number of steps T.[4] Moreover, when the Byzantine workers are very few, i.e., $f \ll n$, the rate for MDA, CWTM, and MeaMed is $\mathcal{O}\left(\sigma/\sqrt{nT}\right)$. Thus, their rate improves with larger n in a similar manner as vanilla distributed SGD (Lian et al., 2015). However, in the same scenario, the rate for Krum*, GM and CWMed is $\mathcal{O}\left(\sigma\sqrt{n}/\sqrt{T}\right)$, i.e., it is directly proportional to n.

This phenomenon could be explained by the fact that Krum*, CWMed, and GM are simply *median-based* aggregation rules, without any averaging operation. Thus, the variance of their outputs grows with n, as suggested by the standard bounds from *order statistics* (Arnold & Groeneveld, 1979; Bertsimas et al., 2006). On the contrary, MDA, CWTM, and MeaMed perform an averaging operation after filtering out dubious vectors, thus mimicking the variance reduction property of the averaging scheme traditionally used in the vanilla distributed SGD.

5. Empirical Evaluation

To investigate the practical relevance of RESAM, we report on a comprehensive set of experiments evaluating it on benchmark image classification tasks under four different Byzantine threats. We implement Algorithm 1 with six different resilient averaging rules and six momentum coefficients. To verify the benefits of our framework, we also run the same set of experiments using two *non* resilient averaging rules. Essentially, our experiments suggest that combining resilient averaging and distributed momentum is critical to Byzantine resilience even in practice.

[4] Vanilla distributed SGD refers to the case when the server uses the simple averaging rule and there are no Byzantine workers.

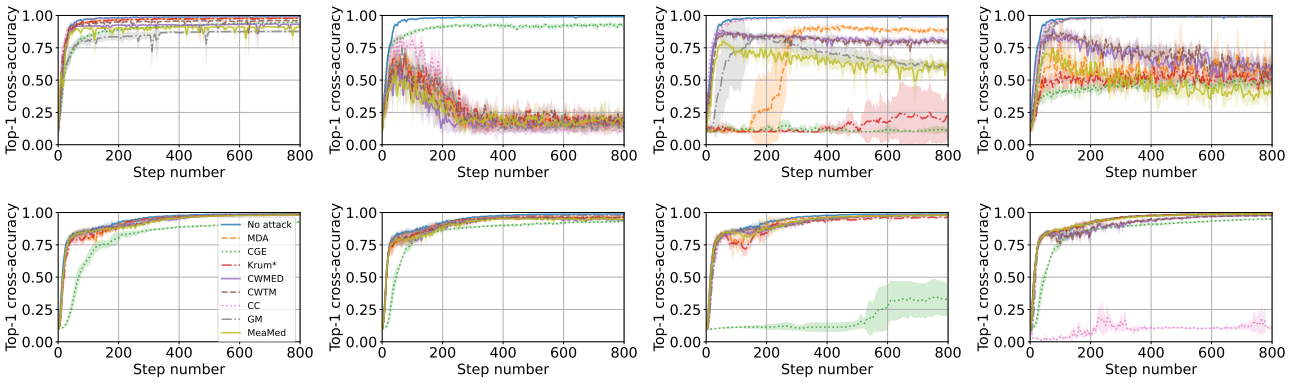

Figure 1. We report on experiments performed on MNIST with $f = 5$ Byzantine among $n = 15$ workers. The 1st and 2nd rows depict the results for $\beta = 0$ and $\beta = 0.99$, respectively. The columns depict the performance of the learning under the *empire*, *little*, *sign-flipping*, and *label-flipping* attacks, respectively.

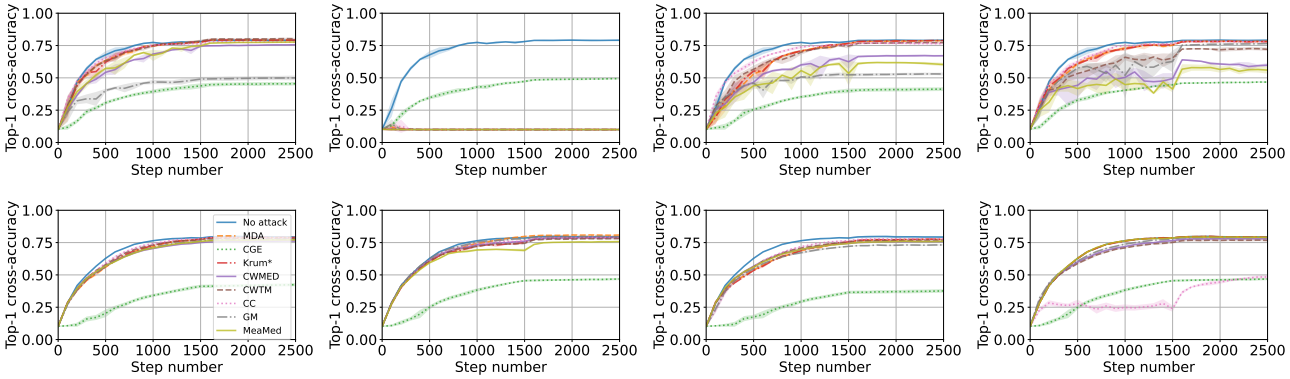

Figure 2. We report on experiments performed on CIFAR-10 with $f = 5$ Byzantine among $n = 25$ workers. The 1st and 2nd rows depict the results for $\beta = 0$ and $\beta = 0.99$, respectively. The columns depict the performance of the learning under the *empire*, *little*, *sign-flipping*, and *label-flipping* attacks, respectively.

5.1. Experimental Setup

Datasets. We use MNIST (LeCun & Cortes, 2010), Fashion-MNIST (Xiao et al., 2017), and CIFAR-10 (Krizhevsky et al., 2009). The datasets are pre-processed as in (Baruch et al., 2019) and (El Mhamdi et al., 2021b).

Architectures and fixed hyperparameters. For MNIST and Fashion-MNIST, we consider a convolutional neural network (CNN) with two convolutional layers followed by two fully-connected layers. To train the model, we use a Cross Entropy loss, a total number of workers $n = 15$, a constant learning rate $\gamma = 0.75$, and a clipping parameter $C = 2$. We also add an ℓ_2-regularization factor of 10^{-4}. Finally, we use a mini-batch size of $b = 25$. For CIFAR-10, we use a CNN with 4 convolutional layers and 2 fully-connected layers, a Cross Entropy loss, and an ℓ_2-regularization factor of 10^{-2}. We set $n = 25$, $\gamma = 0.25$, $C = 5$, and $b = 50$. Refer to Appendix D.3 for more details on our models.

Varying hyperparameters. We vary the number of Byzantine workers f in $\{1, 3, 5, 6, 7\}$ for MNIST and Fashion-MNIST, and $\{5, 11\}$ for CIFAR-10. We also vary the attack implemented by the Byzantine workers. Specifically, we consider *little* (Baruch et al., 2019), *empire* (Xie et al., 2019a), *sign-flipping* (Allen-Zhu et al., 2020), and *label-flipping* (Allen-Zhu et al., 2020). We consider six resilient aggregation rules (MDA, CWTM, CWMed, Krum*, MeaMed, and GM), and two that are not resilient averaging (CGE and CC). As benchmark, we also use the *averaging* aggregation rule without Byzantine workers (denoted by "No attack"). Finally, we vary the momentum coefficient β in $\{0, 0.6, 0.8, 0.9, 0.99, 0.999\}$.

Intractability of MDA and GM. Although MDA presents an *order-optimal* resilience coefficient, it is computationally demanding. As pointed out in (El Mhamdi et al., 2020), its time complexity is in $\mathcal{O}\left(\binom{n}{f} + dn^2\right)$. Additionally, GM does not have a closed-form solution. Existing methods implementing GM, such as (Cohen et al., 2016; Pillutla

et al., 2019) and references therein, are iterative and only approximate GM. Moreover, these methods require expensive computations, e.g., determining eigenvalues and eigenvectors of $d \times d$ matrices (Cohen et al., 2016) in each iteration. Here, we use the approximation algorithm from (Pillutla et al., 2019) to compute GM and only implement MDA whenever its computational complexity is not prohibitive, i.e., when neither $\binom{n}{f}$ nor dn^2 are too large.

Reproducibility and reusability. Each experiment is repeated 5 times using seeds from 1 to 5 for reproducibility purposes. Overall, we performed over $1,512$ experiments ($7,560$ runs), of which we provide a brief overview below. Additional plots and code base to reproduce our experiments are available in the supplementary material. Our implementation will also be made accessible online.

5.2. Experimental Results

We present in Figures 1 and 2 the top-1 cross-accuracy achieved on MNIST and CIFAR-10 when running distributed SGD for 800 and 2500 steps respectively for different aggregation rules and Byzantine attacks. We consider $f = 5$ Byzantine workers in both cases. Due to space limitations, we only show here the results for MNIST and CIFAR-10. Similar results for Fashion-MNIST are deferred to Appendix E.1.

The **main takeaway** of our experiments is that RESAM is crucial to Byzantine resilience in practice. For all datasets considered, we observe from Figures 1 and 2 that combining resilient averaging rules (identified by blue points) and distributed momentum (with $\beta = 0.99$) consistently provides similar cross-accuracies as the benchmark ("No attack") in all attack scenarios. However, when using a resilient averaging rule without momentum ($\beta = 0$), the Byzantine workers can deteriorate the learning (e.g., see second column, *little* attack). Furthermore, using momentum by itself might not suffice either. For instance, on CIFAR-10, using CGE (which is not resilient averaging) results in equally-bad cross-accuracies both when $\beta = 0$ and when $\beta = 0.99$.

The case of CC. In Figure 2, we observe that CC does not present a consistent behavior regarding momentum. In fact, setting $\beta = 0.99$ clearly mitigates the impact of the *little* attack, but drastically deteriorates the performance of the algorithm against *label-flipping*. Similar inconsistencies are observed for MNIST. Note however that although CC does not behave as a resilient averaging rule, it can present good performances when combined with other levels of momentum (e.g., see $\beta = 0.9$ in Appendix E.2).

6. Additional Related Work & Discussion

We discuss hereafter other work that we believe to be related to ours, as well as some possible extensions of our approach.

Applicability to robust estimation. The problem of robust estimation with corrupted data (Lai et al., 2016; Charikar et al., 2017; Diakonikolas et al., 2017; 2019a;b; Steinhardt et al., 2018) can be treated as a special case of Byzantine resilience in distributed machine learning where a Byzantine worker behaves just like an honest worker, except that its stochastic gradients may correspond to an incorrect data distribution (instead of $\mathcal{D}$). RESAM can thus be readily used for robust estimation over an arbitrary distribution $\mathcal{D}$.

Momentum variants. Besides Polyak's momentum, which we considered, it would be interesting to study the impact of the recently proposed *momentum-based variance reduction* (MVR) technique, which has been shown to have optimal convergence rate in non-convex learning (Cutkosky & Orabona, 2019). However, to apply this technique, the gradients (of honest workers) must be defined in a different way than in (2). Basically, $U(\theta)$ cannot have an arbitrary distribution subject to Assumption 2 anymore.

Second-order stationarity. Although a critical point, i.e., a first-order stationary point, represents a global minimum when the loss function Q is convex, this need not be true in general. Indeed, a critical point may not even represent a local minimum when Q is non-convex, and theoretically speaking, our algorithm may get entrapped at *saddle points*. Thus, a stronger learning goal would be to output a second-order stationary point, assuming Q to be second-order Lipschitz smooth. Previous works achieving this goal in the presence of Byzantine workers include (Allen-Zhu et al., 2020; Yin et al., 2019). However, they again resort to non-standard assumptions for stochastic gradients. Showing second-order convergence via RESAM under only standard assumptions represents an interesting future work.

Non-identical workers. When honest workers do not have identical data distributions, Byzantine resilience becomes much more challenging (Su & Shahrampour, 2019; Gupta & Vaidya, 2020; Data & Diggavi, 2021). In this case, the goal changes to minimizing the average of the honest workers' loss functions (Su & Vaidya, 2016). More importantly, we cannot achieve a desirable level of resilience anymore unless there is some redundancy in the data (Liu et al., 2021). Apart from using a robust aggregation rule, there has been some work on the use of ℓ_1-norm regularization (Li et al., 2019). Recently, (Karimireddy et al., 2020) also proposed a meta scheme called *bucketing* that helps in this setting. Extending RESAM to incorporate non-identical honest workers is an interesting future direction.

Knowledgeable server. There is some work studying Byzantine resilience in "non-standard" distributed learning

settings where the server either has prior knowledge on specific verified datapoints (Cao & Lai, 2019; Yao et al., 2019; Xie et al., 2019b; 2020; Regatti et al., 2020), or has control over the sampling of datapoints (Chen et al., 2018; Rajput et al., 2019; Gupta & Vaidya, 2019; Data et al., 2020). In the latter case, we can simply use *error-correction coding*. In the former case, we can also tolerate a majority of Byzantine workers. While these solutions might reveal impractical, deriving an optimal condition to overcome the limit of $1/2$ Byzantine workers remains an interesting future direction.

Acknowledgments

Sadegh and Nirupam are partly supported by Swiss National Science Foundation (SNSF) project 200021_200477, controlling the spread of Epidemics. John is partly supported by SNSF project 200021_182542, machine learning. Rafaël is partly supported by an Ecocloud postdoctoral fellowship. The authors are thankful to Pierre-Louis Roman for fruitful discussion on the introduction, to Youssef Alouah for proofreading the technical part, and to the anonymous reviewers of ICML 2022 for their constructive comments.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X. Tensorflow: Large-scale machine learning on heterogeneous distributed systems, 2015.

Alistarh, D., Allen-Zhu, Z., and Li, J. Byzantine stochastic gradient descent. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pp. 4618–4628, 2018.

Allen-Zhu, Z., Ebrahimianghazani, F., Li, J., and Alistarh, D. Byzantine-resilient non-convex stochastic gradient descent. In *International Conference on Learning Representations*, 2020.

Arnold, B. C. and Groeneveld, R. A. Bounds on expectations of linear systematic statistics based on dependent samples. *The Annals of Statistics*, pp. 220–223, 1979.

Baruch, M., Baruch, G., and Goldberg, Y. A little is enough: Circumventing defenses for distributed learning. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, 8-14 December 2019, Long Beach, CA, USA*, 2019.

Bertsekas, D. and Tsitsiklis, J. *Parallel and distributed computation: numerical methods*. Athena Scientific, 2015.

Bertsimas, D., Natarajan, K., and Teo, C.-P. Tight bounds on expected order statistics. *Probability in the Engineering and Informational Sciences*, 20(4):667–686, 2006.

Blanchard, P., El Mhamdi, E. M., Guerraoui, R., and Stainer, J. Machine learning with adversaries: Byzantine tolerant gradient descent. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 30*, pp. 119–129. Curran Associates, Inc., 2017.

Bottou, L., Curtis, F. E., and Nocedal, J. Optimization methods for large-scale machine learning. *Siam Review*, 60(2):223–311, 2018.

Boyd, S., Boyd, S. P., and Vandenberghe, L. *Convex optimization*. Cambridge university press, 2004.

Cao, X. and Lai, L. Distributed gradient descent algorithm robust to an arbitrary number of byzantine attackers. *IEEE Transactions on Signal Processing*, 67(22):5850–5864, 2019.

Charikar, M., Steinhardt, J., and Valiant, G. Learning from untrusted data. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, pp. 47–60, 2017.

Chen, L., Wang, H., Charles, Z. B., and Papailiopoulos, D. S. DRACO: byzantine-resilient distributed training via redundant gradients. In Dy, J. G. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pp. 902–911. PMLR, 2018.

Chen, Y., Su, L., and Xu, J. Distributed statistical machine learning in adversarial settings: Byzantine gradient descent. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 1(2):1–25, 2017.

Cohen, M. B., Lee, Y. T., Miller, G., Pachocki, J., and Sidford, A. Geometric median in nearly linear time. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pp. 9–21, 2016.

Cutkosky, A. and Orabona, F. Momentum-based variance reduction in non-convex sgd. *Advances in Neural Information Processing Systems*, 32:15236–15245, 2019.

Data, D. and Diggavi, S. Byzantine-resilient high-dimensional sgd with local iterations on heterogeneous data. In *International Conference on Machine Learning*, pp. 2478–2488. PMLR, 2021.

Data, D., Song, L., and Diggavi, S. N. Data encoding for byzantine-resilient distributed optimization. *IEEE Transactions on Information Theory*, 67(2):1117–1140, 2020.

Diakonikolas, I., Kane, D. M., and Stewart, A. Statistical query lower bounds for robust estimation of high-dimensional gaussians and gaussian mixtures. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 73–84. IEEE, 2017.

Diakonikolas, I., Kamath, G., Kane, D., Li, J., Moitra, A., and Stewart, A. Robust estimators in high-dimensions without the computational intractability. *SIAM Journal on Computing*, 48(2):742–864, 2019a.

Diakonikolas, I., Kamath, G., Kane, D., Li, J., Steinhardt, J., and Stewart, A. Sever: A robust meta-algorithm for stochastic optimization. In *International Conference on Machine Learning*, pp. 1596–1606. PMLR, 2019b.

El Mhamdi, E. M., Guerraoui, R., and Rouault, S. The hidden vulnerability of distributed learning in byzantium, 2018.

El Mhamdi, E. M., Guerraoui, R., Guirguis, A., Hoang, L. N., and Rouault, S. Genuinely distributed byzantine machine learning. In *Proceedings of the 39th Symposium on Principles of Distributed Computing*, pp. 355–364, 2020.

El Mhamdi, E. M., Farhadkhani, S., Guerraoui, R., Guirguis, A., Hoang, L. N., and Rouault, S. Collaborative learning in the jungle (decentralized, byzantine, heterogeneous, asynchronous and nonconvex learning). In *Thirty-Fifth Conference on Neural Information Processing Systems*, 2021a.

El Mhamdi, E. M., Guerraoui, R., and Rouault, S. Distributed momentum for byzantine-resilient stochastic gradient descent. In *9th International Conference on Learning Representations, ICLR 2021, Vienna, Austria, May 4–8, 2021*. OpenReview.net, 2021b.

Feng, J., Xu, H., and Mannor, S. Distributed robust learning, 2015.

Feng, J., Xu, H., and Mannor, S. Outlier robust online learning. *CoRR*, abs/1701.00251, 2017.

Ghadimi, S. and Lan, G. Stochastic first-and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization*, 23(4):2341–2368, 2013.

Gupta, N. and Vaidya, N. H. Randomized reactive redundancy for byzantine fault-tolerance in parallelized learning. *arXiv preprint arXiv:1912.09528*, 2019.

Gupta, N. and Vaidya, N. H. Fault-tolerance in distributed optimization: The case of redundancy. In *Proceedings of the 39th Symposium on Principles of Distributed Computing*, pp. 365–374, 2020.

Gupta, N., Liu, S., and Vaidya, N. Byzantine fault-tolerant distributed machine learning with norm-based comparative gradient elimination. In *2021 51st Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W)*, pp. 175–181. IEEE, 2021.

Kairouz, P., McMahan, H. B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A. N., Bonawitz, K., Charles, Z., Cormode, G., Cummings, R., D'Oliveira, R. G. L., Eichner, H., Rouayheb, S. E., Evans, D., Gardner, J., Garrett, Z., Gascón, A., Ghazi, B., Gibbons, P. B., Gruteser, M., Harchaoui, Z., He, C., He, L., Huo, Z., Hutchinson, B., Hsu, J., Jaggi, M., Javidi, T., Joshi, G., Khodak, M., Konecný, J., Korolova, A., Koushanfar, F., Koyejo, S., Lepoint, T., Liu, Y., Mittal, P., Mohri, M., Nock, R., Özgür, A., Pagh, R., Qi, H., Ramage, D., Raskar, R., Raykova, M., Song, D., Song, W., Stich, S. U., Sun, Z., Suresh, A. T., Tramèr, F., Vepakomma, P., Wang, J., Xiong, L., Xu, Z., Yang, Q., Yu, F. X., Yu, H., and Zhao, S. Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning*, 14(1–2):1–210, 2021. ISSN 1935-8237. doi: 10.1561/2200000083.

Karimireddy, S. P., He, L., and Jaggi, M. Byzantine-robust learning on heterogeneous datasets via bucketing. *arXiv preprint arXiv:2006.09365*, 2020.

Karimireddy, S. P., He, L., and Jaggi, M. Learning from history for byzantine robust optimization. *International Conference On Machine Learning, Vol 139*, 139, 2021.

Krizhevsky, A., Nair, V., and Hinton, G. Cifar-100 (canadian institute for advanced research). 2009.

Lai, K. A., Rao, A. B., and Vempala, S. Agnostic estimation of mean and covariance. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 665–674. IEEE, 2016.

Lamport, L., Shostak, R., and Pease, M. The byzantine generals problem. *ACM Trans. Program. Lang. Syst.*, 4 (3):382–401, July 1982. ISSN 0164-0925. doi: 10.1145/357172.357176.

LeCun, Y. and Cortes, C. MNIST handwritten digit database. 2010.

Lei, Y., Hu, T., Li, G., and Tang, K. Stochastic gradient descent for nonconvex learning without bounded gradient assumptions. *IEEE transactions on neural networks and learning systems*, 31(10):4394–4400, 2019.

Li, L., Xu, W., Chen, T., Giannakis, G. B., and Ling, Q. Rsa: Byzantine-robust stochastic aggregation methods for distributed learning from heterogeneous datasets. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 1544–1551, 2019.

Lian, X., Huang, Y., Li, Y., and Liu, J. Asynchronous parallel stochastic gradient for nonconvex optimization. *Advances in Neural Information Processing Systems*, 28: 2737–2745, 2015.

Liu, S., Gupta, N., and Vaidya, N. H. Approximate byzantine fault-tolerance in distributed optimization. In *Proceedings of the 2021 ACM Symposium on Principles of Distributed Computing*, PODC'21, pp. 379–389, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450385480. doi: 10.1145/3465084.3467902.

Lynch, N. A. *Distributed algorithms*. Elsevier, 1996.

Mhamdi, E. M. E., Farhadkhani, S., Guerraoui, R., and Hoang, L. N. On the strategyproofness of the geometric median. *CoRR*, abs/2106.02394, 2021.

Pillutla, K., Kakade, S. M., and Harchaoui, Z. Robust aggregation for federated learning, 2019.

Polyak, B. Some methods of speeding up the convergence of iteration methods. *USSR Computational Mathematics and Mathematical Physics*, 4:1–17, 12 1964. doi: 10.1016/0041-5553(64)90137-5.

Prasad, A., Suggala, A. S., Balakrishnan, S., and Ravikumar, P. Robust estimation via robust gradient estimation. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 82(3):601–627, 2020.

Rajput, S., Wang, H., Charles, Z., and Papailiopoulos, D. Detox: A redundancy-based framework for faster and more robust gradient aggregation. In *International Conference on Machine Learning*, 2019.

Regatti, J., Chen, H., and Gupta, A. Bygars: Byzantine sgd with arbitrary number of attackers. *arXiv preprint arXiv:2006.13421*, 2020.

Rousseeuw, P. J. Multivariate estimation with high breakdown point. *Mathematical statistics and applications*, 8 (37):283–297, 1985.

Steinhardt, J., Charikar, M., and Valiant, G. Resilience: A criterion for learning in the presence of arbitrary outliers. In *9th Innovations in Theoretical Computer Science Conference (ITCS 2018)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018.

Su, L. and Shahrampour, S. Finite-time guarantees for byzantine-resilient distributed state estimation with noisy measurements. *IEEE Transactions on Automatic Control*, 65(9):3758–3771, 2019.

Su, L. and Vaidya, N. H. Fault-tolerant multi-agent optimization: optimal iterative distributed algorithms. In *Proceedings of the 2016 ACM symposium on principles of distributed computing*, pp. 425–434, 2016.

Tsitsiklis, J., Bertsekas, D., and Athans, M. Distributed asynchronous deterministic and stochastic gradient optimization algorithms. *IEEE transactions on automatic control*, 31(9):803–812, 1986.

Xiao, H., Rasul, K., and Vollgraf, R. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

Xie, C., Koyejo, O., and Gupta, I. Generalized byzantine-tolerant sgd, 2018.

Xie, C., Koyejo, O., and Gupta, I. Fall of empires: Breaking byzantine-tolerant SGD by inner product manipulation. In *Proceedings of the Thirty-Fifth Conference on Uncertainty in Artificial Intelligence, UAI 2019, Tel Aviv, Israel, July 22-25, 2019*, pp. 83, 2019a.

Xie, C., Koyejo, S., and Gupta, I. Zeno: Distributed stochastic gradient descent with suspicion-based fault-tolerance. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pp. 6893–6901. PMLR, 2019b.

Xie, C., Koyejo, S., and Gupta, I. Zeno++: Robust fully asynchronous sgd. In *International Conference on Machine Learning*, pp. 10495–10503. PMLR, 2020.

Yao, X., Huang, T., Zhang, R.-X., Li, R., and Sun, L. Federated learning with unbiased gradient aggregation and controllable meta updating. *arXiv preprint arXiv:1910.08234*, 2019.

Yin, D., Chen, Y., Kannan, R., and Bartlett, P. Byzantine-robust distributed learning: Towards optimal statistical rates. In *International Conference on Machine Learning*, pp. 5650–5659. PMLR, 2018.

Yin, D., Chen, Y., Kannan, R., and Bartlett, P. Defending against saddle point attack in byzantine-robust distributed learning. In *International Conference on Machine Learning*, pp. 7074–7084. PMLR, 2019.

Appendix

A. Skeleton of the Proof for Theorem 1

Our formal analysis of Algorithm 1 constitutes of three critical elements

1. The *momentum drift* (see Section A.2)
2. The *momentum deviation* (see Section A.3)
3. The *growth of loss function Q* (see Section A.4)

Ultimately, we combine these elements to obtain the final convergence result stated in Theorem 1. Essentially, the proof of Theorem 1, deferred to Appendix B.5, is obtained by combining the three sub-results presented by lemmas 2, 3 and 4 below.

A.1. Preliminary Notations

For a positive integer T, we let $[T]$ denote the set $\{1, \ldots, T\}$. For a finite set S, we let $|S|$ denote its cardinality. For each step t, we denote by R_t the output of aggregation rule F, i.e.,

$$R_t \coloneqq F\left(m_t^{(1)}, \ldots, m_t^{(n)}\right). \tag{7}$$

We denote by $\mathcal{P}_t$ the history from steps 1 to t. Specifically,

$$\mathcal{P}_t \coloneqq \left\{\theta_1, \ldots, \theta_t;\ m_1^{(i)}, \ldots, m_{t-1}^{(i)}; i = 1, \ldots, n\right\}.$$

By convention, $\mathcal{P}_1 = \{\theta_1\}$. We denote by $\mathbb{E}_t[\cdot]$ and $\mathbb{E}[\cdot]$ the conditional expectation $\mathbb{E}[\cdot \,|\, \mathcal{P}_t]$ and the total expectation, respectively. Thus, $\mathbb{E}[\cdot] = \mathbb{E}_1[\cdots \mathbb{E}_T[\cdot]]$.

A.2. Momentum Drift

We first note that at any step t, given the history $\mathcal{P}_t$, the momentums $m_t^{(i)}$ of the honest workers need not be identically distributed, even when the said property is true for their stochastic gradients $g_t^{(i)}$. Nevertheless, we show in Lemma 1 below that the *drift* between the honest workers' momentums can be controlled up to a certain extent by tuning the momentum coefficient β. We consider an arbitrary subset $\mathcal{H} \subseteq [n]$ of $n - f$ honest workers, i.e., $|\mathcal{H}| = n - f$ and $i \in \mathcal{H}$ only if w_i is an honest worker. Such a set always exists as there are at least $n - f$ honest workers in the system. Then, defining

$$\overline{m}_t \coloneqq {}^1\!/_{(n-f)} \sum_{i \in \mathcal{H}} m_t^{(i)}, \tag{8}$$

we can demonstrate the following. (Proof of Lemma 1 can be found in Appendix B.1.)

Lemma 1. *Suppose that Assumption 2 holds true. Consider Algorithm 1. For each $i \in \mathcal{H}$ and $t \in [T]$, we obtain that*

$$\mathbb{E}\left[\left\|m_t^{(i)} - \overline{m}_t\right\|^2\right] \leq 2\sigma^2 (1-\beta)^2 \beta^{2(t-1)} + 2\left(\frac{1-\beta}{1+\beta}\right)\left(1 + \frac{1}{n-f}\right)\sigma^2.$$

Not that the above result holds even when F is not a resilient averaging rule, as it only analyzes the behavior of the worker's momentum. By building upon this first lemma, we can obtain a bound on the distance between the actual output of F and the average momentum of honest workers for the case when F is (f, λ)-resilient averaging. Specifically, when defining

$$\xi_t \coloneqq R_t - \overline{m}_t, \tag{9}$$

we get the following. (Proof of Lemma 2 can be found in Appendix B.2.)

Lemma 2. *Suppose that Assumption 2 holds true. Consider Algorithm 1 when F is (f, λ)-resilient averaging. For each step $t \in [T]$, we obtain that*

$$\mathbb{E}\left[\|\xi_t\|^2\right] \leq 8\sigma^2\lambda^2(n-f)(1-\beta)^2\beta^{2(t-1)} + 8\left(\frac{1-\beta}{1+\beta}\right)(n-f+1)\,\lambda^2\,\sigma^2.$$

A.3. Momentum Deviation

Next, we study the distance between the average honest momentum $\overline{m}_t$ and the true gradient $\nabla Q(\theta_t)$. Specifically, we define *deviation* to be

$$\delta_t \coloneqq \overline{m}_t - \nabla Q\left(\theta_t\right), \tag{10}$$

and obtain in Lemma 3 below an upper bound on the growth of the deviation over the learning steps $t \in [T]$. (Proof of Lemma 3 can be found in Appendix B.3.)

Lemma 3. *Suppose that assumptions 1 and 2 hold true. Consider Algorithm 1 with $T > 1$. For all $t > 1$ we obtain that*

$$\begin{aligned}\mathbb{E}\left[\left\|\delta_t\right\|^2\right] \leq & \beta^2\zeta_{t-1}\,\mathbb{E}\left[\left\|\delta_{t-1}\right\|^2\right] + 4\gamma_{t-1}L(1+\gamma_{t-1}L)\beta^2\,\mathbb{E}\left[\left\|\nabla Q(\theta_{t-1})\right\|^2\right] + (1-\beta)^2\frac{\sigma^2}{(n-f)} \\ & + 2\gamma_{t-1}L(1+\gamma_{t-1}L)\beta^2\,\mathbb{E}\left[\left\|\xi_{t-1}\right\|^2\right].\end{aligned}$$

where $\zeta_t \coloneqq (1+\gamma_t L)\,(1+4\gamma_t L)$.

A.4. Growth of Loss Function

Finally, we analyze the third element, i.e., the growth of cost function $Q(\theta)$ along the trajectory of Algorithm 1. From (6) and (7), we obtain that for each step t

$$\theta_{t+1} = \theta_t - \gamma_t R_t = \theta_t - \gamma_t\,\overline{m}_t - \gamma_t\left(R_t - \overline{m}_t\right).$$

Furthermore, by (9), $R_t - \overline{m}_t = \xi_t$. Thus, for all t,

$$\theta_{t+1} = \theta_t - \gamma_t\,\overline{m}_t - \gamma_t\xi_t. \tag{11}$$

This means that Algorithm 1 can actually be treated as distributed SGD with a momentum term that is subject to perturbation proportional to ξ_t at each step t. This perspective leads us to the following result. (Proof of Lemma 4 can be found in Appendix B.4.)

Lemma 4. *Suppose that Assumption 1 holds true. Consider Algorithm 1. For all $t \in [T]$, we obtain that*

$$\begin{aligned}\mathbb{E}\left[2Q(\theta_{t+1}) - 2Q(\theta_t)\right] \leq & -\gamma_t\left(1-4\gamma_t L\right)\mathbb{E}\left[\left\|\nabla Q(\theta_t)\right\|^2\right] + 2\gamma_t\left(1+2\gamma_t L\right)\mathbb{E}\left[\left\|\delta_t\right\|^2\right] \\ & + 2\gamma_t\left(1+\gamma_t L\right)\mathbb{E}\left[\left\|\xi_t\right\|^2\right].\end{aligned}$$

B. Proof of Formal Statements

We now present technical proof for both the aforementioned Lemmas as well as Theorem 1 and Corollary 1.

B.1. Proof of Lemma 1

Lemma 1. *Suppose that Assumption 2 holds true. Consider Algorithm 1. For each $i \in \mathcal{H}$ and $t \in [T]$, we obtain that*

$$\mathbb{E}\left[\left\|m_t^{(i)} - \overline{m}_t\right\|^2\right] \leq 2\sigma^2 (1-\beta)^2 \beta^{2(t-1)} + 2\left(\frac{1-\beta}{1+\beta}\right)\left(1 + \frac{1}{n-f}\right)\sigma^2.$$

Proof. Recall that $\mathcal{H} \subseteq \{1, \ldots, n\}$ is a set of $n-f$ honest workers, i.e., $|\mathcal{H}| = n-f$ and $i \in \mathcal{H}$ only if w_i is an honest worker. Also, recall from (8) that

$$\overline{m}_t \coloneqq {}^1\!/_{(n-f)} \sum_{i \in \mathcal{H}} m_t^{(i)}.$$

We consider an arbitrary $i \in \mathcal{H}$. For simplicity we define

$$\widetilde{m}_t^{(i)} \coloneqq m_t^{(i)} - \overline{m}_t,$$

and

$$\overline{g}_t \coloneqq {}^1\!/_{(n-f)} \sum_{j \in \mathcal{H}} g_t^{(j)}. \tag{12}$$

Now, we consider an arbitrary step $t \in [T]$. Substituting from (5), i.e., $m_t^{(i)} = \beta\, m_{t-1}^{(i)} + (1-\beta)\, g_t^{(i)}$ for all $i \in \mathcal{H}$, in (8), i.e., $\overline{m}_t = {}^1\!/_{(n-f)} \sum_{i \in \mathcal{H}} m_t^{(i)}$, we obtain that

$$\overline{m}_t = \beta\, \overline{m}_{t-1} + (1-\beta)\, \overline{g}_t$$

where $\overline{m}_0 = 0$, as $m_0^{(i)} = 0$ for all honest w_i by convention. Thus,

$$\widetilde{m}_t^{(i)} = \beta\, \widetilde{m}_{t-1}^{(i)} + (1-\beta)\left(g_t^{(i)} - \overline{g}_t\right). \tag{13}$$

Recall that for any vector v, $\|v\|^2 = \langle v, v \rangle$. From above we obtain that

$$\left\|\widetilde{m}_t^{(i)}\right\|^2 = \beta^2 \left\|\widetilde{m}_{t-1}^{(i)}\right\|^2 + (1-\beta)^2 \left\|g_t^{(i)} - \overline{g}_t\right\|^2 + 2\beta(1-\beta)\left\langle \widetilde{m}_{t-1}^{(i)},\, g_t^{(i)} - \overline{g}_t \right\rangle.$$

Upon taking conditional expectation $\mathbb{E}_t[\cdot]$ on both sides, and using the fact that $\widetilde{m}_{t-1}^{(i)}$ is a deterministic function of the history $\mathcal{P}_t$, we obtain that

$$\begin{aligned}
\mathbb{E}_t\left[\left\|\widetilde{m}_t^{(i)}\right\|^2\right] &= \beta^2 \mathbb{E}_t\left[\left\|\widetilde{m}_{t-1}^{(i)}\right\|^2\right] + (1-\beta)^2 \mathbb{E}_t\left[\left\|g_t^{(i)} - \overline{g}_t\right\|^2\right] + 2\beta(1-\beta)\,\mathbb{E}_t\left[\left\langle \widetilde{m}_{t-1}^{(i)},\, g_t^{(i)} - \overline{g}_t \right\rangle\right] \\
&= \beta^2 \left\|\widetilde{m}_{t-1}^{(i)}\right\|^2 + (1-\beta)^2 \mathbb{E}_t\left[\left\|g_t^{(i)} - \overline{g}_t\right\|^2\right] + 2\beta(1-\beta)\left\langle \widetilde{m}_{t-1}^{(i)},\, \mathbb{E}_t\left[g_t^{(i)} - \overline{g}_t\right] \right\rangle.
\end{aligned}$$

Due to Assumption 2 and the definition of $\overline{g}_t$ in (12), $\mathbb{E}_t\left[g_t^{(i)} - \overline{g}_t\right] = \mathbb{E}_t\left[g_t^{(i)}\right] - \mathbb{E}_t\left[\overline{g}_t\right] = \nabla Q(\theta_t) - \nabla Q(\theta_t) = 0$. Thus, from above we obtain that

$$\mathbb{E}_t\left[\left\|\widetilde{m}_t^{(i)}\right\|^2\right] = \beta^2 \left\|\widetilde{m}_{t-1}^{(i)}\right\|^2 + (1-\beta)^2 \mathbb{E}_t\left[\left\|g_t^{(i)} - \overline{g}_t\right\|^2\right].$$

Assumption 2 also implies that $\mathbb{E}_t\left[\left\|g_t^{(i)} - \nabla Q(\theta_t)\right\|^2\right] \leq \sigma^2$ for all $i \in \mathcal{H}$. As $g_t^{(j)}$'s for $j \in \mathcal{H}$ are independent of each other, we have $\mathbb{E}_t\left[\left\|\overline{g}_t - \nabla Q(\theta_t)\right\|^2\right] \leq {}^{\sigma^2}/_{(n-f)}$. Therefore, $\mathbb{E}_t\left[\left\|g_t^{(i)} - \overline{g}_t\right\|^2\right] \leq 2\left(1 + {}^1/_{(n-f)}\right)\sigma^2$. Substituting this above we obtain that

$$\mathbb{E}_t\left[\left\|\widetilde{m}_t^{(i)}\right\|^2\right] \leq \beta^2\left\|\widetilde{m}_{t-1}^{(i)}\right\|^2 + 2(1-\beta)^2\left(1 + \frac{1}{n-f}\right)\sigma^2.$$

Taking total expectation on both sides we obtain that

$$\mathbb{E}\left[\left\|\widetilde{m}_t^{(i)}\right\|^2\right] \leq \beta^2\,\mathbb{E}\left[\left\|\widetilde{m}_{t-1}^{(i)}\right\|^2\right] + 2(1-\beta)^2\left(1 + \frac{1}{n-f}\right)\sigma^2.$$

As the above holds true for an arbitrary $t \in [T]$, by telescopic expansion we obtain for all $t \in [T]$ that

$$\begin{aligned}\mathbb{E}\left[\left\|\widetilde{m}_t^{(i)}\right\|^2\right] &\leq \beta^{2(t-1)}\,\mathbb{E}\left[\left\|\widetilde{m}_1^{(i)}\right\|^2\right] + 2(1-\beta)^2\left(1 + \frac{1}{n-f}\right)\sigma^2\sum_{\tau=0}^{t-2}\beta^{2\tau} \\ &= \beta^{2(t-1)}\,\mathbb{E}\left[\left\|\widetilde{m}_1^{(i)}\right\|^2\right] + 2(1-\beta)^2\left(1 + \frac{1}{n-f}\right)\sigma^2\left(\frac{1-\beta^{2(t-1)}}{1-\beta^2}\right).\end{aligned}$$

As $0 \leq \beta < 1$, we have $1 - \beta^{2(t-1)} \leq 1$. Thus, from above we obtain for all $t \in [T]$ that

$$\mathbb{E}\left[\left\|\widetilde{m}_t^{(i)}\right\|^2\right] \leq \beta^{2(t-1)}\,\mathbb{E}\left[\left\|\widetilde{m}_1^{(i)}\right\|^2\right] + 2\left(\frac{1-\beta}{1+\beta}\right)\left(1 + \frac{1}{n-f}\right)\sigma^2. \tag{14}$$

From (13), for each $i \in \mathcal{H}$ we have (upon recalling that $m_0^{(i)} = 0$ for all $i \in \mathcal{H}$),

$$\widetilde{m}_1^{(i)} = (1-\beta)\left(g_1^{(i)} - \overline{g}_1\right).$$

By definition of $\overline{g}_t$ in (12),

$$\mathbb{E}\left[\left\|\widetilde{m}_1^{(i)}\right\|^2\right] = (1-\beta)^2\,\mathbb{E}\left[\left\|g_1^{(i)} - \overline{g}_1\right\|^2\right] = (1-\beta)^2\,\mathbb{E}\left[\left\|\frac{1}{(n-f)}\sum_{j\in\mathcal{H}}\left(g_1^{(i)} - g_1^{(j)}\right)\right\|^2\right].$$

Thus, by applying Jensen's inequality,

$$\mathbb{E}\left[\left\|\widetilde{m}_1^{(i)}\right\|^2\right] \leq \frac{(1-\beta)^2}{(n-f)}\sum_{j\in\mathcal{H}}\mathbb{E}\left[\left\|g_1^{(i)} - g_1^{(j)}\right\|^2\right].$$

By Assumption 2, as gradients of honest workers are pair-wise independent, $\mathbb{E}\left[\left\|g_1^{(i)} - g_1^{(j)}\right\|^2\right] \leq 2\sigma^2$. Substituting this above we obtain that for each $i \in \mathcal{H}$,

$$\mathbb{E}\left[\left\|\widetilde{m}_1^{(i)}\right\|^2\right] \leq 2\sigma^2(1-\beta)^2.$$

Substituting from above in (14) proves the lemma, i.e., for all $t \in [T]$,

$$\mathbb{E}\left[\left\|\widetilde{m}_t^{(i)}\right\|^2\right] \leq 2\sigma^2\,(1-\beta)^2\beta^{2(t-1)} + 2\left(\frac{1-\beta}{1+\beta}\right)\left(1 + \frac{1}{n-f}\right)\sigma^2.$$

□

B.2. Proof of Lemma 2

Lemma 2. *Suppose that Assumption 2 holds true. Consider Algorithm 1 when F is $(f,\, \lambda)$-resilient averaging. For each step $t \in [T]$, we obtain that*

$$\mathbb{E}\left[\left\|\xi_t\right\|^2\right] \leq 8\sigma^2\lambda^2(n-f)(1-\beta)^2\beta^{2(t-1)} + 8\left(\frac{1-\beta}{1+\beta}\right)(n-f+1)\,\lambda^2\,\sigma^2.$$

Proof. Recall from (7) and (9), respectively, that

$$R_t \coloneqq F\left(m_t^{(1)}, \ldots,\, m_t^{(n)}\right) \;\text{ and }\; \xi_t \coloneqq R_t - \overline{m}_t.$$

We consider an arbitrary step t. As F is assumed (f, λ)-resilient averaging, by Definition 2 we obtain that

$$\left\|\xi_t\right\|^2 = \left\|R_t - \overline{m}_t\right\|^2 \leq \lambda^2 \max_{i,\, j \in \mathcal{H}} \left\|m_t^{(i)} - m_t^{(j)}\right\|^2. \tag{15}$$

Note that for any pair $i,\, j \in \mathcal{H}$, from triangle inequality we have $\left\|m_t^{(i)} - m_t^{(j)}\right\| \leq \left\|m_t^{(i)} - \overline{m}_t\right\| + \left\|m_t^{(j)} - \overline{m}_t\right\|$. As $2ab \leq a^2 + b^2$, we also have $\left\|m_t^{(i)} - m_t^{(j)}\right\|^2 \leq 2\left\|m_t^{(i)} - \overline{m}_t\right\|^2 + 2\left\|m_t^{(j)} - \overline{m}_t\right\|^2 \leq 4\max_{i\in\mathcal{H}}\left\|m_t^{(i)} - \overline{m}_t\right\|^2$. Therefore,

$$\max_{i,\, j \in \mathcal{H}} \left\|m_t^{(i)} - m_t^{(j)}\right\|^2 \leq 4 \max_{i \in \mathcal{H}} \left\|m_t^{(i)} - \overline{m}_t\right\|^2.$$

As $\max_{i\in\mathcal{H}}\left\|m_t^{(i)} - \overline{m}_t\right\|^2 \leq \sum_{i\in\mathcal{H}}\left\|m_t^{(i)} - \overline{m}_t\right\|^2$, from above we obtain that

$$\max_{i,\, j \in \mathcal{H}} \left\|m_t^{(i)} - m_t^{(j)}\right\|^2 \leq 4 \sum_{i \in \mathcal{H}} \left\|m_t^{(i)} - \overline{m}_t\right\|^2.$$

Substituting from above in (15) we obtain that $\left\|\xi_t\right\|^2 \leq 4\lambda^2 \sum_{i\in\mathcal{H}}\left\|m_t^{(i)} - \overline{m}_t\right\|^2$. Upon taking total expectations on both sides we obtain that

$$\mathbb{E}\left[\left\|\xi_t\right\|^2\right] \leq 4\lambda^2 \sum_{i \in \mathcal{H}} \mathbb{E}\left[\left\|m_t^{(i)} - \overline{m}_t\right\|^2\right]. \tag{16}$$

From Lemma 1, under Assumption 2, we have for all $i \in \mathcal{H}$ that

$$\mathbb{E}\left[\left\|m_t^{(i)} - \overline{m}_t\right\|^2\right] \leq 2\sigma^2\,(1-\beta)^2\beta^{2(t-1)} + 2\left(\frac{1-\beta}{1+\beta}\right)\left(1 + \frac{1}{n-f}\right)\sigma^2.$$

As $|\mathcal{H}| = n - f$, Substituting from above in (16) proves the lemma, i.e., we obtain that

$$\mathbb{E}\left[\left\|\xi_t\right\|^2\right] \leq 8\lambda^2\sigma^2(n-f)(1-\beta)^2\beta^{2(t-1)} + 8\lambda^2\left(\frac{1-\beta}{1+\beta}\right)(n-f+1)\,\sigma^2.$$

□

B.3. Proof of Lemma 3

Lemma 3. *Suppose that assumptions 1 and 2 hold true. Consider Algorithm 1 with $T > 1$. For all $t > 1$ we obtain that*

$$\begin{aligned}\mathbb{E}\left[\left\|\delta_t\right\|^2\right] \leq{}& \beta^2\zeta_{t-1}\,\mathbb{E}\left[\left\|\delta_{t-1}\right\|^2\right] + 4\gamma_{t-1}L(1+\gamma_{t-1}L)\beta^2\,\mathbb{E}\left[\left\|\nabla Q(\theta_{t-1})\right\|^2\right] + (1-\beta)^2\frac{\sigma^2}{(n-f)} \\ &+ 2\gamma_{t-1}L(1+\gamma_{t-1}L)\beta^2\,\mathbb{E}\left[\left\|\xi_{t-1}\right\|^2\right].\end{aligned}$$

where $\zeta_t \coloneqq (1+\gamma_t L)\,(1+4\gamma_t L)$.

Proof. Recall from Definition (10) that

$$\delta_t \coloneqq \overline{m}_t - \nabla Q\left(\theta_t\right).$$

Consider an arbitrary step $t > 1$. By Definitions (5) and (8), we obtain that

$$\delta_t = \beta\, \overline{m}_{t-1} + (1-\beta)\, \overline{g}_t - \nabla Q\left(\theta_t\right).$$

Upon adding and subtracting $\beta \nabla Q(\theta_{t-1})$ and $\beta \nabla Q(\theta_t)$ on the R.H.S. above we obtain that

$$\begin{aligned}\delta_t &= \beta\, \overline{m}_{t-1} - \beta \nabla Q(\theta_{t-1}) + (1-\beta)\, \overline{g}_t - \nabla Q\left(\theta_t\right) + \beta \nabla Q(\theta_t) + \beta \nabla Q(\theta_{t-1}) - \beta \nabla Q(\theta_t) \\ &= \beta\left(\overline{m}_{t-1} - \nabla Q(\theta_{t-1})\right) + (1-\beta)\, \overline{g}_t - (1-\beta)\nabla Q\left(\theta_t\right) + \beta\left(\nabla Q(\theta_{t-1}) - \nabla Q(\theta_t)\right).\end{aligned}$$

As $\overline{m}_{t-1} - \nabla Q(\theta_{t-1}) = \delta_{t-1}$ (by Definition (10)), from above we obtain that

$$\delta_t = \beta \delta_{t-1} + (1-\beta)\, \left(\overline{g}_t - \nabla Q\left(\theta_t\right)\right) + \beta\left(\nabla Q(\theta_{t-1}) - \nabla Q(\theta_t)\right).$$

Therefore,

$$\begin{aligned}\left\|\delta_t\right\|^2 =& \beta^2 \left\|\delta_{t-1}\right\|^2 + (1-\beta)^2 \left\|\overline{g}_t - \nabla Q\left(\theta_t\right)\right\|^2 + \beta^2 \left\|\nabla Q(\theta_{t-1}) - \nabla Q(\theta_t)\right\|^2 + 2\beta(1-\beta)\left\langle \delta_{t-1},\, \overline{g}_t - \nabla Q\left(\theta_t\right)\right\rangle \\ &+ 2\beta^2 \left\langle \delta_{t-1},\, \nabla Q(\theta_{t-1}) - \nabla Q(\theta_t)\right\rangle + 2\beta(1-\beta)\left\langle \overline{g}_t - \nabla Q\left(\theta_t\right),\, \nabla Q(\theta_{t-1}) - \nabla Q(\theta_t)\right\rangle.\end{aligned}$$

By taking conditional expectation $\mathbb{E}_t\left[\cdot\right]$ on both sides, and recalling that δ_{t-1}, θ_t and θ_{t-1} are deterministic values when the history $\mathcal{P}_t$ is given, we obtain that

$$\begin{aligned}\mathbb{E}_t\left[\left\|\delta_t\right\|^2\right] =& \beta^2 \left\|\delta_{t-1}\right\|^2 + (1-\beta)^2 \mathbb{E}_t\left[\left\|\overline{g}_t - \nabla Q\left(\theta_t\right)\right\|^2\right] + \beta^2 \left\|\nabla Q(\theta_{t-1}) - \nabla Q(\theta_t)\right\|^2 + \\ &2\beta(1-\beta)\left\langle \delta_{t-1},\, \mathbb{E}_t\left[\overline{g}_t\right] - \nabla Q\left(\theta_t\right)\right\rangle + 2\beta^2 \left\langle \delta_{t-1},\, \nabla Q(\theta_{t-1}) - \nabla Q(\theta_t)\right\rangle \\ &+ 2\beta(1-\beta)\left\langle \mathbb{E}_t\left[\overline{g}_t\right] - \nabla Q\left(\theta_t\right),\, \nabla Q(\theta_{t-1}) - \nabla Q(\theta_t)\right\rangle.\end{aligned}$$

Recall that $\overline{g}_t \coloneqq {}^1\!/_{(n-f)} \sum_{j \in \mathcal{H}} g_t^{(j)}$. Thus, owing to Assumption 2, $\mathbb{E}_t\left[\overline{g}_t\right] = \nabla Q(\theta_t)$. Using this above we obtain that

$$\begin{aligned}\mathbb{E}_t\left[\left\|\delta_t\right\|^2\right] =& \beta^2 \left\|\delta_{t-1}\right\|^2 + (1-\beta)^2 \mathbb{E}_t\left[\left\|\overline{g}_t - \nabla Q\left(\theta_t\right)\right\|^2\right] + \beta^2 \left\|\nabla Q(\theta_{t-1}) - \nabla Q(\theta_t)\right\|^2 \\ &+ 2\beta^2 \left\langle \delta_{t-1},\, \nabla Q(\theta_{t-1}) - \nabla Q(\theta_t)\right\rangle.\end{aligned}$$

Also, by Assumption 2 and the fact that $g_t^{(j)}$'s for $j \in \mathcal{H}$ are independent of each other, we have $\mathbb{E}_t\left[\left\|\overline{g}_t - \nabla Q\left(\theta_t\right)\right\|^2\right] \leq \frac{\sigma^2}{(n-f)}$. Thus,

$$\mathbb{E}_t\left[\left\|\delta_t\right\|^2\right] \leq \beta^2 \left\|\delta_{t-1}\right\|^2 + (1-\beta)^2 \frac{\sigma^2}{(n-f)} + \beta^2 \left\|\nabla Q(\theta_{t-1}) - \nabla Q(\theta_t)\right\|^2 + 2\beta^2 \left\langle \delta_{t-1},\, \nabla Q(\theta_{t-1}) - \nabla Q(\theta_t)\right\rangle.$$

By Cauchy-Schwartz inequality, $\left\langle \delta_{t-1},\, \nabla Q(\theta_{t-1}) - \nabla Q(\theta_t)\right\rangle \leq \left\|\delta_{t-1}\right\| \left\|\nabla Q(\theta_{t-1}) - \nabla Q(\theta_t)\right\|$. By Assumption 1, $\left\|\nabla Q(\theta_{t-1}) - \nabla Q(\theta_t)\right\| \leq L \left\|\theta_t - \theta_{t-1}\right\|$. Recall from (6) that $\theta_t = \theta_{t-1} - \gamma_{t-1} R_{t-1}$. Thus, $\left\|\nabla Q(\theta_{t-1}) - \nabla Q(\theta_t)\right\| \leq \gamma_{t-1} L \left\|R_{t-1}\right\|$. Using this above we obtain that

$$\mathbb{E}_t\left[\left\|\delta_t\right\|^2\right] \leq \beta^2 \left\|\delta_{t-1}\right\|^2 + (1-\beta)^2 \frac{\sigma^2}{(n-f)} + \gamma_{t-1}^2 \beta^2 L^2 \left\|R_{t-1}\right\|^2 + 2\gamma_{t-1}\beta^2 L \left\|\delta_{t-1}\right\| \left\|R_{t-1}\right\|.$$

As $2ab \leq a^2 + b^2$, from above we obtain that

$$\begin{aligned}\mathbb{E}_t\left[\left\|\delta_t\right\|^2\right] &\leq \beta^2 \left\|\delta_{t-1}\right\|^2 + (1-\beta)^2 \frac{\sigma^2}{(n-f)} + \gamma_{t-1}^2 \beta^2 L^2 \left\|R_{t-1}\right\|^2 + \gamma_{t-1} L \beta^2 \left(\left\|\delta_{t-1}\right\|^2 + \left\|R_{t-1}\right\|^2\right) \\ &= (1+\gamma_{t-1}L)\beta^2 \left\|\delta_{t-1}\right\|^2 + (1-\beta)^2 \frac{\sigma^2}{(n-f)} + \gamma_{t-1} L (1+\gamma_{t-1}L)\beta^2 \left\|R_{t-1}\right\|^2. \qquad (17)\end{aligned}$$

By definition of R_t in (9), $R_{t-1} = \xi_{t-1} + \overline{m}_{t-1}$. Thus, owing to the triangle inequality and the fact that $2ab \leq a^2 + b^2$, we have $\|R_{t-1}\|^2 \leq 2\|\xi_{t-1}\|^2 + 2\|\overline{m}_{t-1}\|^2$. Similarly, by definition of δ_t in (10), we have $\|\overline{m}_{t-1}\|^2 \leq 2\|\delta_{t-1}\|^2 + 2\left\|\nabla Q(\theta_{t-1})\right\|^2$. Thus, $\|R_{t-1}\|^2 \leq 2\|\xi_{t-1}\|^2 + 4\|\delta_{t-1}\|^2 + 4\left\|\nabla Q(\theta_{t-1})\right\|^2$. Using this in (17) we obtain that

$$\begin{aligned}\mathbb{E}_t\left[\|\delta_t\|^2\right] \leq & (1+\gamma_{t-1}L)\beta^2\|\delta_{t-1}\|^2 + (1-\beta)^2\frac{\sigma^2}{(n-f)} \\ & + 2\gamma_{t-1}L(1+\gamma_{t-1}L)\beta^2\left(\|\xi_{t-1}\|^2 + 2\|\delta_{t-1}\|^2 + 2\left\|\nabla Q(\theta_{t-1})\right\|^2\right).\end{aligned}$$

By re-arranging the terms on the R.H.S. we get

$$\begin{aligned}\mathbb{E}_t\left[\|\delta_t\|^2\right] \leq & \beta^2(1+\gamma_{t-1}L)\left(1+4\gamma_{t-1}L\right)\|\delta_{t-1}\|^2 + 4\gamma_{t-1}L(1+\gamma_{t-1}L)\beta^2\left\|\nabla Q(\theta_{t-1})\right\|^2 + (1-\beta)^2\frac{\sigma^2}{(n-f)} \\ & + 2\gamma_{t-1}L(1+\gamma_{t-1}L)\beta^2\|\xi_{t-1}\|^2.\end{aligned}$$

Substituting $\zeta_{t-1} = (1+\gamma_{t-1}L)\left(1+4\gamma_{t-1}L\right)$ above we obtain that

$$\mathbb{E}_t\left[\|\delta_t\|^2\right] \leq \beta^2\zeta_{t-1}\|\delta_{t-1}\|^2 + 4\gamma_{t-1}L(1+\gamma_{t-1}L)\beta^2\left\|\nabla Q(\theta_{t-1})\right\|^2 + (1-\beta)^2\frac{\sigma^2}{(n-f)} + 2\gamma_{t-1}L(1+\gamma_{t-1}L)\beta^2\|\xi_{t-1}\|^2.$$

Recall that t in the above is an arbitrary value in $[T]$ greater than 1. Hence, upon taking total expectation on both sides above proves the lemma.

□

B.4. Proof of Lemma 4

Lemma 4. *Suppose that Assumption 1 holds true. Consider Algorithm 1. For all $t \in [T]$, we obtain that*

$$\begin{aligned}\mathbb{E}\left[2Q(\theta_{t+1}) - 2Q(\theta_t)\right] \leq & -\gamma_t\left(1-4\gamma_t L\right)\mathbb{E}\left[\|\nabla Q(\theta_t)\|^2\right] + 2\gamma_t\left(1+2\gamma_t L\right)\mathbb{E}\left[\|\delta_t\|^2\right] \\ & + 2\gamma_t\left(1+\gamma_t L\right)\mathbb{E}\left[\|\xi_t\|^2\right].\end{aligned}$$

Proof. Consider an arbitrary step t. Due to Assumption 1 (i.e., Lipschitz continuity of $\nabla Q(\theta)$), we have (see (Bottou et al., 2018))

$$Q(\theta_{t+1}) - Q(\theta_t) \leq \left\langle \theta_{t+1} - \theta_t,\, \nabla Q(\theta_t)\right\rangle + \frac{L}{2}\left\|\theta_{t+1} - \theta_t\right\|^2.$$

Substituting from (11), i.e., $\theta_{t+1} = \theta_t - \gamma_t\,\overline{m}_t - \gamma_t\xi_t$, we obtain that

$$\begin{aligned}Q(\theta_{t+1}) - Q(\theta_t) &\leq -\gamma_t\left\langle \overline{m}_t,\, \nabla Q(\theta_t)\right\rangle - \gamma_t\left\langle \xi_t,\, \nabla Q(\theta_t)\right\rangle + \gamma_t^2\frac{L}{2}\left\|\,\overline{m}_t + \xi_t\right\|^2. \\ &= -\gamma_t\left\langle \overline{m}_t - \nabla Q(\theta_t) + \nabla Q(\theta_t),\, \nabla Q(\theta_t)\right\rangle - \gamma_t\left\langle \xi_t,\, \nabla Q(\theta_t)\right\rangle + \gamma_t^2\frac{L}{2}\left\|\,\overline{m}_t + \xi_t\right\|^2.\end{aligned}$$

By Definition (10), $\overline{m}_t - \nabla Q(\theta_t) = \delta_t$. Thus, from above we obtain that (scaling by factor of 2)

$$2Q(\theta_{t+1}) - 2Q(\theta_t) \leq -2\gamma_t\|\nabla Q(\theta_t)\|^2 - 2\gamma_t\left\langle \delta_t,\, \nabla Q(\theta_t)\right\rangle - 2\gamma_t\left\langle \xi_t,\, \nabla Q(\theta_t)\right\rangle + \gamma_t^2 L\left\|\,\overline{m}_t + \xi_t\right\|^2. \tag{18}$$

Now, we consider the last three terms on the R.H.S. separately. Using Cauchy-Schwartz inequality, and the fact that $2ab \leq \frac{1}{c}a^2 + cb^2$ for any $c > 0$, we obtain that (by substituting $c = 2$)

$$2\left|\left\langle \delta_t,\, \nabla Q(\theta_t)\right\rangle\right| \leq 2\|\delta_t\|\,\|\nabla Q(\theta_t)\| \leq \frac{2}{1}\|\delta_t\|^2 + \frac{1}{2}\|\nabla Q(\theta_t)\|^2. \tag{19}$$

Similarly,

$$2\left|\langle \xi_t,\, \nabla Q(\theta_t)\rangle\right| \leq 2\left\|\xi_t\right\| \left\|\nabla Q(\theta_t)\right\| \leq \frac{2}{1}\left\|\xi_t\right\|^2 + \frac{1}{2}\left\|\nabla Q(\theta_t)\right\|^2. \tag{20}$$

Finally, using triangle inequality and the fact that $2ab \leq a^2 + b^2$ we have

$$\begin{aligned}\left\|\,\overline{m}_t + \xi_t\right\|^2 &\leq 2\,\left\|\overline{m}_t\right\|^2 + 2\left\|\xi_t\right\|^2 = 2\,\left\|\overline{m}_t - \nabla Q(\theta_t) + \nabla Q(\theta_t)\right\|^2 + 2\left\|\xi_t\right\|^2 \\ &\leq 4\,\left\|\delta_t\right\|^2 + 4\,\left\|\nabla Q(\theta_t)\right\|^2 + 2\left\|\xi_t\right\|^2. \qquad [since\ \ \overline{m}_t - \nabla Q(\theta_t) = \delta_t]\end{aligned} \tag{21}$$

Substituting from (19), (20) and (21) in (18) we obtain that

$$\begin{aligned}2Q(\theta_{t+1}) - 2Q(\theta_t) \leq &-2\gamma_t\left\|\nabla Q(\theta_t)\right\|^2 + \gamma_t\left(2\left\|\delta_t\right\|^2 + \frac{1}{2}\left\|\nabla Q(\theta_t)\right\|^2\right) + \gamma_t\left(2\left\|\xi_t\right\|^2 + \frac{1}{2}\left\|\nabla Q(\theta_t)\right\|^2\right) \\ &+ \gamma_t^2 L\left(4\,\left\|\delta_t\right\|^2 + 4\,\left\|\nabla Q(\theta_t)\right\|^2 + 2\left\|\xi_t\right\|^2\right).\end{aligned}$$

Upon re-arranging the terms in the R.H.S. we obtain that

$$2Q(\theta_{t+1}) - 2Q(\theta_t) \leq -\gamma_t\left(1 - 4\gamma_t L\right)\left\|\nabla Q(\theta_t)\right\|^2 + 2\gamma_t\left(1 + 2\gamma_t L\right)\left\|\delta_t\right\|^2 + 2\gamma_t\left(1 + \gamma_t L\right)\left\|\xi_t\right\|^2.$$

As t is arbitrarily chosen from $[T]$, taking expectation on both sides above proves the lemma. □

B.5. Proof of Theorem 1

We recall the theorem statement below for convenience.

Theorem 1. *Suppose that assumptions 1 and 2 hold true. Let us denote*

$$Q^* = \min_{\theta\in\mathbb{R}^d} Q(\theta),\ a_o = 4\left(2\left(Q(\theta_1) - Q^*\right) + \frac{1}{8L}\left\|\nabla Q\left(\theta_1\right)\right\|^2\right),\ a_1 = 6912L,\ \ and\ a_2 = 288L. \tag{22}$$

Consider Algorithm 1 with an (f,λ)-resilient averaging rule and a constant learning rate of γ. Specifically, for all t, $\gamma_t = \gamma$ where

$$\gamma = \left(\sqrt{\frac{a_o(n-f)}{a_1\lambda^2(n-f)^2 + a_2}}\right)\frac{1}{\sigma\sqrt{T}}. \tag{23}$$

If $T \geq \frac{a_o L}{12\sigma^2\lambda^2(n-f)}$ and $\beta = \sqrt{1-24\gamma L}$, then

$$\mathbb{E}\left[\left\|\nabla Q\left(\widehat{\theta}\right)\right\|^2\right] \leq 2\sqrt{\left(a_1\lambda^2(n-f) + \frac{a_2}{n-f}\right)\frac{a_o\sigma^2}{T}} + \left(\frac{a_2\sigma}{n-f}\right)\left(\sqrt{\frac{a_o(n-f)}{a_1\lambda^2(n-f)+a_2}}\right)\frac{1}{T^{3/2}}.$$

Proof. Define

$$\gamma_o := \frac{1}{18L}. \tag{24}$$

Note that as specified in the theorem statement,

$$T \geq \frac{a_o L}{12\sigma^2\lambda^2(n-f)} \geq \frac{576 a_o L^2}{12\times 576\sigma^2\lambda^2(n-f)L} > \frac{576 a_o L^2(n-f)}{6912\sigma^2\lambda^2(n-f)^2L + 288L\sigma^2} = \frac{576L^2 a_o(n-f)}{\left(a_1\lambda^2(n-f)^2 + a_2\right)\sigma^2}$$

This implies that for the learning rate γ defined in (23),

$$\gamma = \left(\sqrt{\frac{a_o(n-f)}{a_1\lambda^2(n-f)^2 + a_2}}\right)\frac{1}{\sigma\sqrt{T}} < \frac{1}{24L} < \frac{1}{18L} = \gamma_o. \tag{25}$$

This also implies

$$24\gamma L < 1$$

Therefore $\beta = \sqrt{1 - 24\gamma L}$ (as defined in the theorem) is a well-defined real value in $(0, 1)$.

To obtain the convergence result we define the Lyapunov function to be

$$V_t \coloneqq \mathbb{E}\left[2Q(\theta_t) + z\left\|\delta_t\right\|^2\right] \quad \text{and } z = \frac{1}{8L}. \tag{26}$$

We consider an arbitrary $t \in [T]$.

Invoking Lemma 3. Upon substituting $\gamma_t = \gamma$ in Lemma 3, we obtain that

$$\begin{aligned}\mathbb{E}\left[z\left\|\delta_{t+1}\right\|^2 - z\left\|\delta_t\right\|^2\right] \leq & z\beta^2\zeta\,\mathbb{E}\left[\left\|\delta_t\right\|^2\right] + 4z\gamma L(1+\gamma L)\beta^2\,\mathbb{E}\left[\left\|\nabla Q(\theta_t)\right\|^2\right] + z(1-\beta)^2\frac{\sigma^2}{n-f} \\ & + 2z\gamma L(1+\gamma L)\beta^2\,\mathbb{E}\left[\left\|\xi_t\right\|^2\right] - z\,\mathbb{E}\left[\left\|\delta_t\right\|^2\right].\end{aligned} \tag{27}$$

Recall that

$$\zeta = (1+\gamma L)\,(1+4\gamma L) = 1 + 5\gamma L + 4\gamma^2 L^2. \tag{28}$$

Invoking Lemma 4. By the same substitution in Lemma 4 we obtain that

$$\begin{aligned}\mathbb{E}\left[2Q(\theta_{t+1}) - 2Q(\theta_t)\right] \leq & -\gamma\,(1-4\gamma L)\,\mathbb{E}\left[\left\|\nabla Q(\theta_t)\right\|^2\right] + 2\gamma\,(1+2\gamma L)\,\mathbb{E}\left[\left\|\delta_t\right\|^2\right] \\ & + 2\gamma\,(1+\gamma L)\,\mathbb{E}\left[\left\|\xi_t\right\|^2\right]\end{aligned} \tag{29}$$

Substituting from (27) and (29) in (26) we obtain that

$$\begin{aligned}V_{t+1} - V_t =& \mathbb{E}\left[2Q(\theta_{t+1}) - 2Q(\theta_t)\right] + \mathbb{E}\left[z\left\|\delta_{t+1}\right\|^2 - z\left\|\delta_t\right\|^2\right] \\ \leq & -\gamma\,(1-4\gamma L)\,\mathbb{E}\left[\left\|\nabla Q(\theta_t)\right\|^2\right] + 2\gamma\,(1+2\gamma L)\,\mathbb{E}\left[\left\|\delta_t\right\|^2\right] + 2\gamma\,(1+\gamma L)\,\mathbb{E}\left[\left\|\xi_t\right\|^2\right] \\ & + z\beta^2\zeta\,\mathbb{E}\left[\left\|\delta_t\right\|^2\right] + 4z\gamma L(1+\gamma L)\beta^2\,\mathbb{E}\left[\left\|\nabla Q(\theta_t)\right\|^2\right] + z(1-\beta)^2\frac{\sigma^2}{n-f} \\ & + 2z\gamma L(1+\gamma L)\beta^2\,\mathbb{E}\left[\left\|\xi_t\right\|^2\right] - z\,\mathbb{E}\left[\left\|\delta_t\right\|^2\right].\end{aligned} \tag{30}$$

Upon re-arranging the R.H.S. in (30) we obtain that

$$\begin{aligned}V_{t+1} - V_t \leq & -\gamma\left((1-4\gamma L) - 4zL(1+\gamma L)\beta^2\right)\mathbb{E}\left[\left\|\nabla Q(\theta_t)\right\|^2\right] + z(1-\beta)^2\frac{\sigma^2}{n-f} \\ & + \left(2\gamma\,(1+2\gamma L) + z\beta^2\zeta - z\right)\mathbb{E}\left[\left\|\delta_t\right\|^2\right] + 2\gamma\left(1+\gamma L + zL(1+\gamma L)\beta^2\right)\mathbb{E}\left[\left\|\xi_t\right\|^2\right].\end{aligned}$$

For simplicity, we define

$$A \coloneqq (1-4\gamma L) - 4zL(1+\gamma L)\beta^2, \tag{31}$$

$$B \coloneqq 2\gamma\,(1+2\gamma L) + z\beta^2\zeta - z, \tag{32}$$

and

$$C \coloneqq 2\gamma\left(1+\gamma L + zL(1+\gamma L)\beta^2\right). \tag{33}$$

Thus,

$$V_{t+1} - V_t \leq -A\gamma\,\mathbb{E}\left[\left\|\nabla Q(\theta_t)\right\|^2\right] + B\,\mathbb{E}\left[\left\|\delta_t\right\|^2\right] + C\,\mathbb{E}\left[\left\|\xi_t\right\|^2\right] + z(1-\beta)^2\frac{\sigma^2}{n-f}. \tag{34}$$

We now analyse below the terms A, B and C.

Term A. Recall from (25) that $\gamma \leq \gamma_o = \frac{1}{18L}$. Upon using this in (31), and the facts that $z = \frac{1}{8L}$ and $\beta^2 < 1$, we obtain that

$$A \geq 1 - 4\gamma_o L - \frac{4L}{8L}(1+\gamma_o L) \geq \frac{1}{2} - \frac{9\gamma_o L}{2} \geq \frac{1}{4}. \tag{35}$$

Term B. Substituting ζ from (28) in (32) we obtain that

$$\begin{aligned} B &= 2\gamma\left(1+2\gamma L\right) + z\beta^2\left(1+5\gamma L+4\gamma^2L^2\right) - z \\ &= -\left(1-\beta^2\right)z + \gamma\left(2+4\gamma L+5z\beta^2 L+4z\beta^2 L\gamma L\right). \end{aligned}$$

Using the facts that $\beta^2 \leq 1$ and $\gamma \leq \gamma_o \leq \frac{1}{18L}$, and then substituting $z = \frac{1}{8L}$ we obtain that

$$\begin{aligned} B &\leq \frac{-(1-\beta^2)}{8L} + \gamma\left(2 + \frac{4}{18} + \frac{5}{8} + \frac{4}{18\times 8}\right) \leq \frac{-(1-\beta^2)}{8L} + 3\gamma \\ &\leq \frac{-(1-\beta^2)+24\gamma L}{8L} = 0, \end{aligned} \tag{36}$$

where in the last equality we used the fact that $1-\beta^2 = 24\gamma L$.

Term C. Substituting $z = \frac{1}{8L}$ in (33), and then using the fact that $\beta^2 < 1$, we obtain that

$$C = 2\gamma\left(1+\gamma L + \frac{1}{8}(1+\gamma L)\right) \leq \frac{9\gamma}{4}\left(1+\gamma L\right).$$

As $\gamma \leq \gamma_o \leq \frac{1}{18L}$, from above we obtain that

$$C \leq \frac{9\gamma}{4}\left(1+\frac{1}{18}\right) \leq 3\gamma. \tag{37}$$

Combining terms A, B and C. Finally, substituting from (35), (36) and (37) in (34) (and recalling that $z = \frac{1}{8L}$) we obtain that

$$V_{t+1} - V_t \leq -\frac{\gamma}{4}\,\mathbb{E}\left[\left\|\nabla Q(\theta_t)\right\|^2\right] + 3\gamma\,\mathbb{E}\left[\left\|\xi_t\right\|^2\right] + (1-\beta)^2\frac{\sigma^2}{8L(n-f)}.$$

As the above is true for an arbitrary $t \in [T]$, by taking summation on both sides from $t=1$ to $t=T$ we obtain that

$$V_{T+1} - V_1 \leq -\frac{\gamma}{4}\sum_{t=1}^{T}\mathbb{E}\left[\left\|\nabla Q(\theta_t)\right\|^2\right] + 3\gamma\sum_{t=1}^{T}\mathbb{E}\left[\left\|\xi_t\right\|^2\right] + (1-\beta)^2\frac{\sigma^2}{8L(n-f)}T.$$

Thus,

$$\frac{\gamma}{4}\sum_{t=1}^{T}\mathbb{E}\left[\left\|\nabla Q(\theta_t)\right\|^2\right] \leq V_1 - V_{T+1} + 3\gamma\sum_{t=1}^{T}\mathbb{E}\left[\left\|\xi_t\right\|^2\right] + (1-\beta)^2\frac{\sigma^2}{8L(n-f)}T. \tag{38}$$

Note that, as $\beta > 0$, and $1-\beta^2 = 24\gamma L$, we have

$$(1-\beta)^2 = \frac{\left(1-\beta^2\right)^2}{\left(1+\beta\right)^2} \leq \left(1-\beta^2\right)^2 = 576\gamma^2L^2.$$

Substituting from above in (38) we obtain that

$$\frac{\gamma}{4}\sum_{t=1}^{T}\mathbb{E}\left[\left\|\nabla Q(\theta_t)\right\|^2\right] \leq V_1 - V_{T+1} + 3\gamma\sum_{t=1}^{T}\mathbb{E}\left[\left\|\xi_t\right\|^2\right] + \frac{576\gamma^2L^2\sigma^2}{8L(n-f)}\,T.$$

Multiplying both sides by $4/\gamma$ we obtain that

$$\sum_{t=1}^{T}\mathbb{E}\left[\left\|\nabla Q(\theta_t)\right\|^2\right] \leq \frac{4\left(V_1 - V_{T+1}\right)}{\gamma} + 12\sum_{t=1}^{T}\mathbb{E}\left[\left\|\xi_t\right\|^2\right] + \frac{288\gamma L\sigma^2}{(n-f)}\,T. \tag{39}$$

Next, we use Lemma 2 to derive an upper bound on $\sum_{t=1}^{T}\mathbb{E}\left[\left\|\xi_t\right\|^2\right]$.

Invoking Lemma 2. Recall from Lemma 2 that as F is assumed f-resilient averaging we have for all $t \in [T]$,

$$\mathbb{E}\left[\left\|\xi_t\right\|^2\right] \leq 8\sigma^2\lambda^2(n-f)(1-\beta)^2\beta^{2(t-1)} + 8\left(\frac{1-\beta}{1+\beta}\right)(n-f+1)\lambda^2\sigma^2.$$

By taking summation over t from 1 to T, we obtain that

$$\begin{aligned}\sum_{t=1}^{T}\mathbb{E}\left[\left\|\xi_t\right\|^2\right] &\leq 8\sigma^2\lambda^2(n-f)(1-\beta)^2\sum_{t=1}^{T}\beta^{2(t-1)} + 8\left(\frac{1-\beta}{1+\beta}\right)(n-f+1)\lambda^2\,\sigma^2T\\ &= 8\sigma^2\lambda^2(n-f)(1-\beta)^2\left(\frac{1-\beta^{2T}}{1-\beta^2}\right) + 8\left(\frac{1-\beta}{1+\beta}\right)(n-f+1)\lambda^2\,\sigma^2T\\ &= 8\sigma^2\lambda^2(n-f)\left(\frac{1-\beta}{1+\beta}\right)\left(1-\beta^{2T}\right) + 8\left(\frac{1-\beta}{1+\beta}\right)(n-f+1)\lambda^2\,\sigma^2T\end{aligned}$$

As $0 < \beta < 1$, we have $\left(1-\beta^{2T}\right) \leq 1$. Thus, as $1 \leq n-f \leq (n-f)T$, from above we obtain that

$$\sum_{t=1}^{T}\mathbb{E}\left[\left\|\xi_t\right\|^2\right] \leq 8\sigma^2\lambda^2(n-f)\left(\frac{1-\beta}{1+\beta}\right) + 16\left(\frac{1-\beta}{1+\beta}\right)(n-f)\lambda^2\sigma^2T = 24\sigma^2\lambda^2(n-f)T\left(\frac{1-\beta}{1+\beta}\right). \tag{40}$$

As $\beta > 0$, and the fact that $1-\beta^2 = 24\gamma L$, we have

$$\frac{1-\beta}{1+\beta} = \frac{1-\beta^2}{(1+\beta)^2} \leq 1-\beta^2 = 24\gamma L.$$

Substituting the above in (40), we obtain that

$$\sum_{t=1}^{T}\mathbb{E}\left[\left\|\xi_t\right\|^2\right] \leq (24\times 24)\sigma^2\lambda^2\gamma L(n-f)T = 576\sigma^2\lambda^2\gamma L(n-f)T.$$

Substituting from above in (39) we obtain that

$$\sum_{t=1}^{T}\mathbb{E}\left[\left\|\nabla Q(\theta_t)\right\|^2\right] \leq \frac{4\left(V_1 - V_{T+1}\right)}{\gamma} + (12\times 576)\sigma^2\lambda^2\gamma L(n-f)T + \frac{288\gamma L\sigma^2}{(n-f)}\,T$$

Recall that

$$a_1 = (12\times 576)L = 6912L, \text{ and } a_2 = 288L.$$

Thus, from above we obtain that

$$\sum_{t=1}^{T}\mathbb{E}\left[\left\|\nabla Q(\theta_t)\right\|^2\right] \leq \frac{4\left(V_1 - V_{T+1}\right)}{\gamma} + a_1\lambda^2(n-f)\sigma^2\gamma T + \frac{a_2\sigma^2}{(n-f)}\,\gamma T.$$

Diving both sides by T we obtain that

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\left\|\nabla Q(\theta_t)\right\|^2\right] \leq \frac{4\left(V_1 - V_{T+1}\right)}{\gamma T} + a_1\lambda^2(n-f)\sigma^2\gamma + \frac{a_2\sigma^2}{(n-f)}\gamma. \tag{41}$$

Analysing V_t. Recall that $Q^* = \min_{\theta\in\mathbb{R}^d} Q(\theta)$. Note that for an arbitrary t, by definition of V_t in (26),

$$V_t - 2Q^* = 2\,\mathbb{E}\left[Q(\theta_t) - Q^*\right] + z\,\mathbb{E}\left[\left\|\delta_t\right\|^2\right] \geq 0 + z\,\mathbb{E}\left[\left\|\delta_t\right\|^2\right] \geq 0.$$

Thus,

$$V_1 - V_{T+1} = V_1 - 2Q^* - (V_{T+1} - 2Q^*) \leq V_1 - 2Q^*. \tag{42}$$

Moreover,

$$V_1 = 2Q(\theta_1) + z\,\mathbb{E}\left[\left\|\delta_1\right\|^2\right]. \tag{43}$$

By definition of δ_t in (10), and the definition of $\overline{m}_t$ in (8),we obtain that

$$\mathbb{E}\left[\left\|\delta_1\right\|^2\right] = \mathbb{E}\left[\left\|\overline{m}_1 - \nabla Q(\theta_1)\right\|^2\right] = \mathbb{E}\left[\left\|(1-\beta)\overline{g}_1 - \nabla Q(\theta_1)\right\|^2\right]$$

where $\overline{g}_t$, defined in (12), is the average of $n-f$ honest workers' stochastic gradients in step 1. Expanding the R.H.S. above we obtain that

$$\mathbb{E}\left[\left\|\delta_1\right\|^2\right] = (1-\beta)^2\,\mathbb{E}\left[\left\|\overline{g}_1 - \nabla Q(\theta_1)\right\|^2\right] + \beta^2\left\|\nabla Q(\theta_1)\right\|^2 - 2\beta(1-\beta)\left\langle\mathbb{E}\left[\overline{g}_1\right] - \nabla Q(\theta_1),\, \nabla Q(\theta_1)\right\rangle.$$

Under Assumption 2, we have $\mathbb{E}\left[\overline{g}_1\right] = \nabla Q(\theta_1)$ and $\mathbb{E}\left[\left\|\overline{g}_1 - \nabla Q(\theta_1)\right\|^2\right] \leq \sigma^2/(n-f)$. Therefore,

$$\mathbb{E}\left[\left\|\delta_1\right\|^2\right] \leq \frac{(1-\beta)^2\sigma^2}{(n-f)} + \beta^2\left\|\nabla Q(\theta_1)\right\|^2.$$

Substituting the above in (43) we obtain that

$$V_1 \leq 2Q(\theta_1) + z\left(\frac{(1-\beta)^2\sigma^2}{(n-f)} + \beta^2\left\|\nabla Q(\theta_1)\right\|^2\right).$$

Recall that $(1-\beta)^2 \leq \left(1-\beta^2\right)^2 = 576\gamma^2L^2$. Using this, and the facts that $\beta^2 < 1$ and $z = \frac{1}{8L}$, we obtain that

$$\begin{aligned} V_1 &\leq 2Q(\theta_1) + \frac{1}{8L}\left\|\nabla Q\left(\theta_1\right)\right\|^2 + \frac{576\gamma^2L^2\sigma^2}{8L(n-f)} \\ &= 2Q(\theta_1) + \frac{1}{8L}\left\|\nabla Q\left(\theta_1\right)\right\|^2 + \frac{72\gamma^2L\sigma^2}{(n-f)}. \end{aligned}$$

Recall that $a_2 = 288L$. Therefore,

$$V_1 \leq 2Q(\theta_1) + \frac{1}{8L}\left\|\nabla Q\left(\theta_1\right)\right\|^2 + \frac{a_2\sigma^2}{4(n-f)}\gamma^2.$$

Substituting the above in (42) we obtain that

$$V_1 - V_{T+1} \leq 2Q(\theta_1) - 2Q^* + \frac{1}{8L}\left\|\nabla Q\left(\theta_1\right)\right\|^2 + \frac{a_2\sigma^2}{4(n-f)}\gamma^2.$$

Substituting from above in (41) we obtain that

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\left\|\nabla Q(\theta_t)\right\|^2\right] \leq \frac{4\left(2\left(Q(\theta_1) - Q^*\right) + \frac{\left\|\nabla Q(\theta_1)\right\|^2}{8L}\right)}{\gamma T} + \left(\frac{a_2\sigma^2}{n-f}\right)\frac{\gamma}{T} + a_1\lambda^2(n-f)\sigma^2\gamma + \frac{a_2\sigma^2}{(n-f)}\gamma.$$

Upon re-arranging the terms on R.H.S. above we obtain that

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\|\nabla Q(\theta_t)\|^2\right] \leq \frac{4\left(2\left(Q(\theta_1)-Q^*\right)+\frac{\|\nabla Q(\theta_1)\|^2}{8L}\right)}{\gamma T} + \left(a_1\lambda^2(n-f)+\frac{a_2}{n-f}\right)\sigma^2\gamma + \left(\frac{a_2\sigma^2}{n-f}\right)\frac{\gamma}{T}.$$

Recall that $a_o = 4\left(2\left(Q(\theta_1)-Q^*\right)+\frac{\|\nabla Q(\theta_1)\|^2}{8L}\right)$, we obtain that

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\|\nabla Q(\theta_t)\|^2\right] \leq \frac{a_o}{\gamma T} + \left(\frac{a_1\lambda^2(n-f)^2+a_2}{n-f}\right)\sigma^2\gamma + \left(\frac{a_2\sigma^2}{n-f}\right)\frac{\gamma}{T}. \tag{44}$$

Final step. Recall that

$$\gamma = \left(\sqrt{\frac{a_o(n-f)}{a_1\lambda^2(n-f)^2+a_2}}\right)\frac{1}{\sigma\sqrt{T}}.$$

Substituting this value of γ in (44) we obtain that

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\|\nabla Q(\theta_t)\|^2\right] \leq 2\sqrt{\left(a_1\lambda^2(n-f)+\frac{a_2}{n-f}\right)\frac{a_o\sigma^2}{T}} + \left(\frac{a_2\sigma}{n-f}\right)\left(\sqrt{\frac{a_o(n-f)}{a_1\lambda^2(n-f)^2+a_2}}\right)\frac{1}{T^{3/2}}.$$

Finally, recall from Algorithm 1 that $\widehat{\theta}$ is chosen randomly from the set of computed parameter vectors $\{\theta_1, \ldots, \theta_T\}$. Thus, $\mathbb{E}\left[\left\|\nabla Q\left(\widehat{\theta}\right)\right\|^2\right] = \frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\|\nabla Q(\theta_t)\|^2\right]$. Substituting this above proves the theorem.

□

B.6. Proof of Corollary 1

Corollary 1. *Suppose that assumptions 1 and 2 hold true. Then, Algorithm 1 with an (f,λ)-resilient averaging rule, and parameters γ_t, T and β as defined in Theorem 1, is is (f,ϵ)-resilient with*

$$\epsilon \in \mathcal{O}\left(\sqrt{\frac{\sigma^2}{T}\left(\frac{1}{n-f}+\lambda^2(n-f)\right)}\right).$$

Proof. Owing to Theorem 1, we have

$$\mathbb{E}\left[\left\|\nabla Q\left(\widehat{\theta}\right)\right\|^2\right] \leq 2\sqrt{\left(a_1\lambda^2(n-f)+\frac{a_2}{n-f}\right)\frac{a_o\sigma^2}{T}} + \left(\frac{a_2\sigma}{n-f}\right)\left(\sqrt{\frac{a_o(n-f)}{a_1\lambda^2(n-f)^2+a_2}}\right)\frac{1}{T^{3/2}},$$

where

$$a_o = 4\left(2\left(Q(\theta_1)-Q^*\right)+\frac{1}{8L}\left\|\nabla Q\left(\theta_1\right)\right\|^2\right),\ a_1 = 6912L,\ \text{ and } a_2 = 288L.$$

Thus, by Definition 1, Algorithm 1 is $(f,\,\epsilon)$-resilient where

$$\epsilon = \mathbb{E}\left[\left\|\nabla Q\left(\widehat{\theta}\right)\right\|^2\right] \leq 2\sqrt{\left(a_1\lambda^2(n-f)+\frac{a_2}{n-f}\right)\frac{a_o\sigma^2}{T}} + \left(\frac{a_2\sigma}{n-f}\right)\left(\sqrt{\frac{a_o(n-f)}{a_1\lambda^2(n-f)^2+a_2}}\right)\frac{1}{T^{3/2}}.$$

Upon ignoring constants, including a_o, a_1 and a_2, and the higher-order term of $T^{3/2}$, we obtain that

$$\epsilon \in \mathcal{O}\left(\sqrt{\left(\lambda^2(n-f)+\frac{1}{n-f}\right)\frac{\sigma^2}{T}}\right).$$

Hence, the proof. □

C. Resilience coefficient λ for several aggregation rules (Proof of Proposition 1)

In this section, we first present a lower bound in Section C.1 on the resilience coefficient for any deterministic (f, λ)-resilient averaging rule. Then, we present the aggregation rules listed in Table 1 and derive their resilience coefficients. More precisely, we compute the resilience coefficients of the following rules.

- *Minimum diameter averaging* (MDA) in Section C.2
- *Coordinate-wise trimmed mean* (CWTM) in Section C.3
- *Mean-around-median* (MeaMed) in Section C.4
- *(Multi-)Krum** in Section C.5
- *Geometric median* (GM) in Section C.6
- *Coordinate-wise median* (CWMed) in Section C.7.

As an immediate corollary of the result we get for theses aggregation rules, we obtain Proposition 1, that we recall below.

Proposition 1. *Consider an aggregation rule $F \in \{\text{MDA}, \text{CWTM}, \text{MeaMed}, \text{Krum}^*, \text{GM}, \text{CWMed}\}$. For any $f < n/2$, there exists a resilience coefficient λ for which F is (f, λ)-resilient averaging.*

Besides computing the aforementioned resilience coefficients, we also discuss the case of *centred clipping* (CC) and *comparative gradient elimination* (CGE) in Section C.8 and Section C.9 respectively.

C.1. Lower Bound

Proposition 2. *For $0 \leq f < n$, there cannot exist an (f, λ)-resilient averaging rule for $\lambda < \frac{f}{n-f}$.*

Proof. Consider an arbitrary value of $f \in \{0, \ldots, n-1\}$. Let F be an (f, λ)-resilient averaging aggregation rule. Consider a set of n one dimensional vectors $x_1, \ldots, x_n$ such that $x_1 = \ldots = x_{n-f} = 0$, and $x_{n-f+1} = \ldots = x_n = 1$. Let us first consider a set $S_0 = \{1, \ldots, n-f\}$. Since $|S_0| = n - f$, by Definition 2, we have

$$\|F(x_1, \ldots, x_n) - \overline{x}_{S_0}\| \leq \lambda \max_{i,j \in S_0} \|x_i - x_j\| = 0.$$

Thus, $F(x_1, \ldots, x_n) = \overline{x}_{S_0} = 0$. Now, consider another set $S_1 = \{f+1, \ldots, n\}$. Note that $\overline{x}_{S_1} = \frac{f}{n-f}$. Thus,

$$\|F(x_1, \ldots, x_n) - \overline{x}_{S_1}\| = \frac{f}{n-f}. \tag{45}$$

As F is assumed to be an (f, λ)-resilient averaging rule, by Definition 2 we have

$$\|F(x_1, \ldots, x_n) - \overline{x}_{S_1}\| \leq \lambda \max_{i,j \in S_1} \|x_i - x_j\| = \lambda.$$

If $\lambda < \frac{f}{n-f}$ then the above contradicts (45). This concludes the proof. □

C.2. Minimum Diameter Averaging (MDA)

Given a set of n vectors $x_1, \ldots, x_n$, the MDA algorithm, originally proposed in (Rousseeuw, 1985) and reused in (El Mhamdi et al., 2018), first chooses a set S^* of cardinality $n - f$ with the smallest *diameter*, i.e.,

$$S^* \in \underset{\substack{S \subset \{1, \ldots, n\} \\ |S| = n-f}}{\operatorname{argmin}} \left\{ \max_{i,j \in S} \|x_i - x_j\| \right\}. \tag{46}$$

Then the algorithm outputs, the average of the inputs in set S^*. Specifically it outputs

$$\text{MDA}(x_1, \ldots, x_n) \coloneqq \frac{1}{n-f} \sum_{i \in S^*} x_i. \tag{47}$$

Proposition 3. *If $f < n/2$ then* MDA *is an (f, λ)-resilient averaging rule for $\lambda = \frac{2f}{n-f}$.*

Proof. Let S be an arbitrary subset of $\{1, \ldots, n\}$ such that $|S| = n - f$. To prove the proposition we first show that

$$\|\text{MDA}(x_1, \ldots, x_n) - \overline{x}_S\| \leq \frac{f}{n-f} \max_{i \in S, j \in S^*} \|x_i - x_j\|$$

where $\overline{x}_S \coloneqq \frac{1}{|S|} \sum_{i \in S} x_i$.

In doing so , we note that $|S^* \setminus S| = |S^* \cup S| - |S| \leq n - (n-f) = f$. The same observation holds for $|S \setminus S^*|$. Hence we obtain that

$$\begin{aligned} \|\text{MDA}(x_1, \ldots, x_n) - \overline{x}_S\| &= \left\| \frac{1}{n-f} \sum_{i \in S^*} x_i - \frac{1}{n-f} \sum_{i \in S} x_i \right\| = \frac{1}{n-f} \left\| \sum_{i \in S^* \setminus S} x_i - \sum_{i \in S \setminus S^*} x_i \right\| \\ &\leq \frac{\max(|S^* \setminus S| \, ; |S \setminus S^*|)}{n-f} \max_{i \in S^*, j \in S} \|x_i - x_j\| \leq \frac{f}{n-f} \max_{i \in S, j \in S^*} \|x_i - x_j\| \, . \end{aligned} \tag{48}$$

As we assume that $f < n/2$, we also have

$$|S^* \cap S| = |S| + |S^*| - |S^* \cup S| \geq (n-f) + (n-f) - n \geq n - 2f > 0.$$

Therefore, $S^* \cap S \neq \emptyset$. Let i^* be an arbitrary index that belongs to both S and S^*. From triangle inequality, we obtain that, for any $i^\dagger \in S^*$ and $j^\dagger \in S$,

$$\left\|x_{i^\dagger} - x_{j^\dagger}\right\| \leq \|x_{i^\dagger} - x_{i^*}\| + \left\|x_{i^*} - x_{j^\dagger}\right\| \leq \max_{i,j \in S^*} \|x_i - x_j\| + \max_{i,j \in S} \|x_i - x_j\| \, .$$

By definition of S^* in (46), $\max_{i,j \in S^*} \|x_i - x_j\| \leq \max_{i,j \in S} \|x_i - x_j\|$. Thus, from above we obtain that

$$\left\|x_{i^\dagger} - x_{j^\dagger}\right\| \leq 2 \max_{i,j \in S} \|x_i - x_j\| \, . \tag{49}$$

As $i^\dagger$ and $j^\dagger$ above are arbitrary elements in S^* and S, respectively, from above we obtain that

$$\max_{i \in S, j \in S^*} \|x_i - x_j\| \leq 2 \max_{i,j \in S} \|x_i - x_j\| \, .$$

Combining the above with (49) we obtain that

$$\|\text{MDA}(x_1, \ldots, x_n) - \overline{x}_S\| \leq \frac{2f}{n-f} \max_{i,j \in S} \|x_i - x_j\| \, .$$

As S is an arbitrary subset of $[n]$ of size $n - f$, the above proves the proposition. □

C.3. Coordinate-Wise Trimmed Mean (CWTM)

Let $x \in \mathbb{R}^d$, we denote by $[x]_k$, the k-th coordinate of x. Given the input vectors $x_1, \ldots, x_n$ (in $\mathbb{R}^d$), we let τ_k denote a permutation on $[n]$ that sorts the k-coordinate of the input vectors in non-decreasing order, i.e., $[x_{\tau_k(1)}]_k \leq [x_{\tau_k(2)}]_k \leq \ldots \leq [x_{\tau_k(n)}]_k$. Then, the CWTM of $x_1, \ldots, x_n$, denoted by $\text{CWTM}(x_1, \ldots, x_n)$, is a vector in $\mathbb{R}^d$ whose k-th coordinate is defined as follows,

$$[\text{CWTM}(x_1, \ldots, x_n)]_k \coloneqq \frac{1}{n-2f} \sum_{j \in [f+1, n-f]} [x_{\tau_k(j)}]_k.$$

To obtain the resilience coefficient of CWTM, we recall show in Lemma 5 below how the *diameter* of a set of vectors is related to their *coordinate-wise diameter*. This lemma also proves useful to other coordinate-wise aggregation rules, e.g., CWMed.

Lemma 5. *For a non-empty set of d-dimensional vectors S, we have*

$$\sqrt{\sum_{k=1}^{d}\left(\max_{i,j\in S}|[x_i]_k-[x_j]_k|\right)^2}\leq \min\left\{2\sqrt{|S|},\sqrt{d}\right\}\max_{i,j\in S}\|x_i-x_j\|\,.$$

Proof. Special case of Lemma 18 in (El Mhamdi et al., 2021a) for $r=2$. □

We can now formally state the proposition proving that CWTM is an (f,λ)-resilient averaging rule.

Proposition 4. *If $f<n/2$ then* CWTM *is an (f,λ)-resilient averaging rule for $\lambda=\frac{f}{n-f}\min\left\{2\sqrt{n-f},\sqrt{d}\right\}$.*

Proof. The idea of the proof is similar to that of Theorem 5 in (El Mhamdi et al., 2021a). Consider an arbitrary set $S\subseteq[n]$ such that $|S|=n-f>f$. For each coordinate $k\in[d]$, let π_k^S denote a permutation on S such that $[x_{\pi_k^S(1)}]_k\leq[x_{\pi_k^S(2)}]_k\leq\ldots\leq[x_{\pi_k^S(|S|)}]_k$. Let $c=\text{CWTM}(x_1,\ldots,x_n)$. Then, by Definition of the permuations τ_k, for each k we have that

$$\frac{1}{n-2f}\sum_{i=1}^{n-2f}[x_{\pi_k^S(i)}]_k\leq\frac{1}{n-2f}\sum_{j=f+1}^{n-f}[x_{\tau_k(j)}]_k=[c]_k. \tag{50}$$

Note that for all $j\in S$ and $k\in[d]$, we have

$$\begin{aligned}[x_j]_k&=[x_j]_k+\frac{1}{n-2f}\sum_{i=1}^{n-2f}[x_{\pi_k^S(i)}]_k-\frac{1}{n-2f}\sum_{i=1}^{n-2f}[x_{\pi_k^S(i)}]_k\\&=\frac{1}{n-2f}\sum_{i=1}^{n-2f}[x_{\pi_k^S(i)}]_k+\frac{1}{n-2f}\sum_{i=1}^{n-2f}\left([x_j]_k-[x_{\pi_k^S(i)}]_k\right).\end{aligned}$$

Substituting from (50) above we obtain for all $j\in S$ and $k\in[d]$ that

$$[x_j]_k\leq[c]_k+\frac{1}{n-2f}\sum_{i=1}^{n-2f}\left([x_j]_k-[x_{\pi_k^S(i)}]_k\right)\leq[c]_k+\max_{l,m\in S}|[x_l]_k-[x_m]_k|\,. \tag{51}$$

Recall that $\overline{x}_S:={}^1\!/_{|S|}\sum_{i\in S}x_i$. From (50) and (51) we obtain that

$$\begin{aligned}[\overline{x}_S]_k&=\frac{1}{n-f}\sum_{i\in[n-f]}[x_{\pi_k^S(i)}]_k=\frac{1}{n-f}\sum_{i=1}^{n-2f}[x_{\pi_k^S(i)}]_k+\frac{1}{n-f}\sum_{i=n-2f+1}^{n-f}[x_{\pi_k^S(i)}]_k\\&\leq\frac{n-2f}{n-f}[c]_k+\frac{f}{n-f}([c]_k+\max_{i,j\in S}|[x_i]_k-[x_j]_k|)=[c]_k+\frac{f}{n-f}\max_{i,j\in S}|[x_i]_k-[x_j]_k|\,.\end{aligned} \tag{52}$$

Now, similar to (50), we obtain for all $k\in[d]$ that

$$\frac{1}{n-2f}\sum_{i=f+1}^{n-f}[x_{\pi_k^S(i)}]_k\geq\frac{1}{n-2f}\sum_{j=f+1}^{n-f}[x_{\tau_k(j)}]_k=[c]_k. \tag{53}$$

In a similar manner to (51), we obtain for all $j\in S$ and $k\in[d]$ that

$$[x_j]_k\geq[c]_k-\max_{l,m\in S}|[x_l]_k-[x_m]_k|\,. \tag{54}$$

From (53) and (54), in a similar manner to (52), we obtain that

$$[\overline{x}_S]_k \geq [c]_k - \frac{f}{n-f} \max_{i,j \in S} |[x_i]_k - [x_j]_k| \,. \tag{55}$$

Owing to (52) and (55) we obtain that, for all $k \in [d]$,

$$|[\overline{x}_S]_k - [c]_k| \leq \frac{f}{n-f} \max_{i,j \in S} |[x_i]_k - [x_j]_k| \,.$$

Thus,

$$\|\overline{x}_S - c\| = \sqrt{\sum_{k \in [d]} |[\overline{x}_S]_k - [c]_k|^2} \leq \frac{f}{n-f} \sqrt{\sum_{k \in [d]} \left(\max_{i,j \in S} |[x_i]_k - [x_j]_k| \right)^2} .$$

Recall that $c = \text{CWTM}(x_1, \ldots, x_n)$. Thus, using Lemma 5 we obtain that

$$\|\overline{x}_S - \text{CWTM}(x_1, \ldots, x_n)\| \leq \frac{f}{n-f} \min \left\{ 2\sqrt{n-f}, \sqrt{d} \right\} \max_{i,j \in S} \|x_i - x_j\| \,.$$

As S is an arbitrary subset of $[n]$ of size $n - f$, concludes the proof. □

C.4. Mean around Median (MeaMed)

Let $x \in \mathbb{R}^d$, we denote by $[x]_k$, the k-th coordinate of x. Given the input vectors $x_1, \ldots, x_n$ (in $\mathbb{R}^d$), MeaMed computes the average of the $n - f$ closest elements to the median in each dimension. Specifically, for each $k \in [d]$, $m \in [n]$, let $i_{m;k}$ be the index of the input vector with k-th coordinate that is m-th closest to $\text{Median}([x_1]_k, \ldots, [x_n]_k)$. Let C_k be the set of $n - f$ indices defined as

$$C_k = \{i_{1;k}, \ldots, i_{n-f;k}\}.$$

Then we have

$$[\text{MeaMed}(x_1, \ldots, x_n)]_k = \frac{1}{n-f} \sum_{i \in C_k} [x_i]_k,$$

where $\text{MeaMed}(x_1, \ldots, x_n)$ denotes the output of the aggregation rule.

Proposition 5. *If $f < n/2$, then* MeaMed *is an (f, λ)-resilient averaging for $\lambda = \frac{2f}{n-f} \min \left\{ 2\sqrt{n-f}, \sqrt{d} \right\}$.*

Proof. Consider an arbitrary set S such that $|S| = n - f$. Since $|S| > n/2$, by the definition of the median, for each $k \in [d]$, we have

$$\min_{i \in S} [x_i]_k \leq \text{Median}([x_1]_k, \ldots, [x_n]_k) \leq \max_{i \in S} [x_i]_k.$$

Accordingly, for any $j \in S$ and $k \in [d]$, we have

$$|\text{Median}([x_1]_k, \ldots, [x_n]_k) - [x_j]_k| \leq \max_{i \in S} [x_i]_k - \min_{i \in S} [x_i]_k. \tag{56}$$

In particular, this means that there exist at least $n - f$ vectors within $[x_1]_k, \ldots, [x_n]_k$, whose absolute deviation from $\text{Median}([x_1]_k, \ldots, [x_n]_k)$ is upper-bound by $\max_{i \in S} [x_i]_k - \min_{i \in S} [x_i]_k$. Therefore, by the definition of C_k, for any $j \in C_k$, we have

$$|Median([x_1]_k, \ldots, [x_n]_k) - [x_j]_k| \leq \max_{i \in S} [x_i]_k - \min_{i \in S} [x_i]_k. \tag{57}$$

Combining (56) and (57), then implies that for any $l \in S$ and $m \in C_k$, we have

$$|[x_l]_k - [x_m]_k| \leq 2 \max_{i,j \in S} ([x_i]_k - [x_j]_k).$$

Also note that $|S \setminus C_k| = |C_k \setminus S| = |C_k \cup S| - |S| \leq n - (n-f) = f$. Hence we get

$$\begin{aligned}
|[\text{MeaMed}(x_1, \ldots, x_n)]_k - [\overline{x}_S]_k| &= \frac{1}{n-f} \left| \sum_{i \in C_k} [x_i]_k - \sum_{i \in S} [x_i]_k \right| \\
&= \frac{1}{n-f} \left| \sum_{i \in C_k \setminus S} [x_i]_k - \sum_{i \in S \setminus C_k} [x_i]_k \right| \\
&\leq \frac{2f}{n-f} \max_{i,j \in S} ([x_i]_k - [x_j]_k).
\end{aligned}$$

Finally, by using Lemma 5, we get

$$\|\overline{x}_S - \text{MeaMed}(x_1, \ldots, x_n)\| \leq \frac{2f}{(n-f)} \sqrt{\sum_{k \in [d]} \left(\max_{i,j \in S} |[x_i]_k - [x_j]_k| \right)^2} \tag{58}$$

$$\leq \frac{2f}{(n-f)} \min\{2\sqrt{n-f}, \sqrt{d}\} \max_{i,j \in S} \|x_i - x_j\| . \tag{59}$$

The above concludes the proof. □

C.5. (Multi-)Krum*

In this section, we study a slight adaptation of the Multi-Krum algorithm first introduced in (Blanchard et al., 2017). This adaptation, called Multi-Krum*, is mainly changing one step of the procedure to enhance the tolerance of the method from $f < (n-2)/2$ (needed for the original method) to $f < n/2$ (i.e., the optimal tolerance threshold).

Essentially, given the input vectors $x_1, \ldots, x_n$, Multi-Krum* outputs an average of the vectors that are the closest to their neighbors upon discarding f[5] farthest vectors. Specifically, for each $i \in [n]$ and $k \in [n-1]$, let $i_k \in [n] \setminus \{i\}$ be the index of the k-th closest input vector from x_i, i.e., we have $\|x_i - x_{i_1}\| \leq \ldots \leq \left\|x_i - x_{i_{n-1}}\right\|$ with ties broken arbitrarily. Let C_i be the set of $n - f - 1$ closest vectors to x_i, i.e.,

$$C_i = \{i_1, \ldots, i_{n-f-1}\}.$$

Then, for each $i \in [n]$, we define $score(i) \coloneqq \sum_{j \in C_i} \|x_i - x_j\|^2$. Finally, Multi-Krum$_q^*$ outputs the average of q input vectors with the smallest scores, i.e.,

$$\text{Multi-Krum}_q^*(x_1, \ldots, x_n) = \frac{1}{q} \sum_{i \in M(q)} x_i,$$

where $M(q)$ is the set of q vectors with the smallest scores. We call by Krum* the special case of Multi-Krum$_q^*$ for $q = 1$.

Before analyzing Multi-Krum$_q^*$, we prove the following lemma.

Lemma 6. *Consider a set $S \subset [n]$ such that $|S| = n - f$. Suppose $q \leq n - f$. For any $k \in M(q)$ and $l \in S$, we have*

$$\|x_k - x_l\| \leq \left(1 + \sqrt{\frac{n-f}{n-2f}}\right) \max_{i,j \in S} \|x_i - x_j\| .$$

Proof. To demonstrate this result, we study two cases separately; **case i)** $k \in S$, and **case ii)** $k \notin S$.

Case i) Let $l \in S$, if $k \in S$, by definition we have

$$\|x_k - x_l\| \leq \max_{i,j \in S} \|x_i - x_j\| . \tag{60}$$

Thus, (6) trivially holds in case i).

[5] As opposed to $f + 1$ in the original version.

Case ii) Let us now consider that $k \notin S$. Since $|M(q)| = q \leq n - f$, there exists at least an index $m \in S$ such that $m \notin M(q)$. Then by the definitions of the score function $score(\cdot)$ and of the set C_m, we get that

$$score(m) = \sum_{j \in C_m} \|x_m - x_j\|^2 \leq \sum_{j \in S} \|x_m - x_j\|^2 \leq (n-f) \max_{i,j \in S} \|x_i - x_j\|^2 . \tag{61}$$

Since $m \notin M(q)$, we have $score(k) \leq score(m)$. Accordingly, we have that

$$score(k) = \sum_{j \in C_m} \|x_k - x_j\|^2 \leq score(m). \tag{62}$$

Note that $|C_k \cap S| = |C_k| + |S| - |C_k \cup S| \geq (n-f) + (n-f) - n = n - 2f$. As $f < n/2$, we get $C_k \cap S \neq \emptyset$. Now, as $C_k \cap S \subseteq C_k$, we have $\sum_{j \in C_k \cap S} \|x_k - x_j\|^2 \leq \sum_{j \in C_k} \|x_k - x_j\|^2$. Thus, from (62) we obtain that

$$\sum_{j \in C_k \cap S} \|x_k - x_j\|^2 \leq score(m).$$

Substituting from (61) above, we obtain that

$$\sum_{j \in C_k \cap S} \|x_k - x_j\|^2 \leq (n-f) \max_{i,j \in S} \|x_i - x_j\|^2 .$$

As $|C_k \cap S| \geq n - 2f$, $\sum_{j \in C_k \cap S} \|x_k - x_j\|^2 \geq (n-2f) \min_{j \in S} \|x_k - x_j\|^2$. Thus, from above we obtain that

$$(n-2f) \min_{j \in S} \|x_k - x_j\|^2 \leq (n-f) \max_{i,j \in S} \|x_i - x_j\|^2 .$$

This implies that

$$\min_{j \in S} \|x_k - x_j\| \leq \sqrt{\frac{n-f}{n-2f}} \max_{i,j \in S} \|x_i - x_j\| . \tag{63}$$

Let $l \in S$ and $j^* \in \arg\min_{j \in S} \|x_k - x_j\|$. By the triangle inequality, we then obtain that

$$\|x_k - x_l\| = \|x_k - x_{j^*} + x_{j^*} - x_l\| \leq \|x_k - x_{j^*}\| + \|x_{j^*} - x_l\| \tag{64}$$

Substituting from above in (63) and using the fact that $\|x_{j^*} - x_l\| \leq \max_{i,j \in S} \|x_i - x_j\|$, we then obtain that

$$\|x_k - x_l\| \leq \left(1 + \sqrt{\frac{n-f}{n-2f}}\right) \max_{i,j \in S} \|x_i - x_j\| . \tag{65}$$

The above proves (6) in case ii).

As (6) holds true in either case (see (60) and (65)), the lemma holds true. □

We present below a proposition characterizing the resilient averaging property of Multi-Krum$^*_q$. Note that the resilience coefficient of Krum* can be immediately derived from this proposition by substituting $q = 1$.

Proposition 6. *If $f < n/2$, and $q \leq n - f$ then Multi-Krum$^*_q$ is an (f, λ)-resilient averaging rule for*

$$\lambda = \left(1 + \sqrt{\frac{n-f}{n-2f}}\right) \cdot \min\left\{1, \frac{n-q}{n-f}\right\} .$$

Proof. Consider a set of vectors S such that $|S| = n - f$. As in the proof of Lemma 6, we consider two different cases separately; **case i)** $q \leq f$, and **case ii)** $q > f$.

Case i) Let $q \leq f$. By triangle inequality and Lemma 6, we obtain that

$$\begin{aligned}\left\|\text{Multi-Krum}_q^*\left(x_1, \ldots, x_n\right) - \overline{x}_S\right\| &= \left\|\frac{1}{q}\sum_{i\in M(q)} x_i - \overline{x}_S\right\| \leq \frac{1}{q}\sum_{i\in M(q)}\left\|x_i - \overline{x}_S\right\| \leq \frac{1}{q}\sum_{i\in M(q)}\left\|x_i - \frac{1}{n-f}\sum_{j\in S} x_j\right\| \\ &\leq \frac{1}{q(n-f)}\sum_{i\in M(q)}\sum_{j\in S}\left\|x_i - x_j\right\| \\ &\leq \frac{1}{q(n-f)}\sum_{i\in M(q)}\sum_{j\in S}\left(1+\sqrt{\frac{n-f}{n-2f}}\right)\max_{i,j\in S}\left\|x_i - x_j\right\| \\ &= \left(1+\sqrt{\frac{n-f}{n-2f}}\right)\max_{i,j\in S}\left\|x_i - x_j\right\|.\end{aligned}$$

Thus, the proposition holds true in case i).

Case ii) Let us now consider $q > f$. We have $|S \cap M(q)| = |S| + |M(q)| - |S \cup M(q)| \geq (n-f) + q - n = q - f > 0$. Therefore, there exists a set P with cardinality $q - f$ such that $P \subset S \cap M(q)$. Hence we get

$$\left\|\text{Multi-Krum}_q^*\left(x_1, \ldots, x_n\right) - \overline{x}_S\right\| = \left\|\frac{1}{n-f}\sum_{i\in S} x_i - \frac{1}{q}\sum_{i\in M(q)} x_i\right\| \tag{66}$$

$$= \left\|\frac{1}{n-f}\sum_{i\in S\setminus P} x_i - \frac{1}{q}\sum_{i\in M(q)\setminus P} x_i - \left(\frac{1}{q} - \frac{1}{n-f}\right)\sum_{i\in P} x_i\right\| \tag{67}$$

$$= \frac{1}{q(n-f)}\left\|q\sum_{i\in S\setminus P} x_i - \left((n-f)\sum_{i\in M(q)\setminus P} x_i + (n-f-q)\sum_{i\in P} x_i\right)\right\| = \frac{1}{q(n-f)}\left\|A - B\right\|, \tag{68}$$

where

$$A \coloneqq q\sum_{i\in S\setminus P} x_i \quad \text{and} \quad B \coloneqq \left((n-f)\sum_{i\in M(q)\setminus P} x_i + (n-f-q)\sum_{i\in P} x_i\right). \tag{69}$$

Since $|S \setminus P| = (n-f) - (q-f) = n - q$, A is a sum of $q(n-q)$ (potentially repetitive) vectors all of which belong to S. Also, $f(n-f) + (n-f-q)(q-f) = q(n-q)$. Thus, B is also a sum of $q(n-q)$ (potentially repetitive) vectors all of which belong to $M(q)$. We now match each vector in A to a vector in B. Using the triangle inequality and Lemma 6, we the obtain

$$\left\|A - B\right\| \leq q(n-q)\left(1+\sqrt{\frac{n-f}{n-2f}}\right)\max_{i,j\in S}\left\|x_i - x_j\right\|. \tag{70}$$

Combining above with (68), we then obtain

$$\left\|\text{Multi-Krum}_q^*\left(x_1, \ldots, x_n\right) - \overline{x}_S\right\| \leq \frac{n-q}{n-f}\left(1+\sqrt{\frac{n-f}{n-2f}}\right)\max_{i,j\in S}\left\|x_i - x_j\right\|. \tag{71}$$

This shows that the proposition holds true in case ii).

Combing the conclusions for cases i) and ii) concludes the proof □

C.6. Geometric Median (GM)

For input vectors $x_1, \ldots, x_n$, their geometric median, denoted by $\text{GM}(x_1, \ldots, x_n)$, is defined to be a vector that minimizes the sum of the distances to these vectors. Specifically, we have

$$\text{GM}(x_1, \ldots, x_n) \in \operatorname*{argmin}_{z\in\mathbb{R}^d}\sum_{i=1}^{n}\left\|z - x_i\right\|.$$

For obtaining the resilience coefficient of GM, we make use of the following three lemmas. Below, we denote by $\text{Conv}(x_1, \ldots, x_n)$ the *convex hull* of $x_1, \ldots, x_n$, i.e.,

$$\text{Conv}(x_1, \ldots, x_n) = \left\{ \sum_{i=1}^{n} a_i x_i \;\middle|\; \sum_{i=1}^{n} a_i = 1,\ a_i \geq 0,\ \forall i \in [n] \right\}$$

Lemma 7. *Let y and z be any two points in* $\text{Conv}(x_1, \ldots, x_n)$*. Then,* $\|y - z\| \leq \max_{i,\, j \in [n]} \|x_i - x_j\|$.

Proof. By definition, suppose that $y = \sum_{i=1}^{n} a_i x_i$ and $z = \sum_{i=1}^{n} b_i x_i$ such that $\sum_{i=1}^{n} a_i = 1$, $\sum_{i=1}^{n} b_i = 1$, and $a_i \geq 0$, $b_i \geq 0$ for $i \in [n]$. We then obtain

$$\|y - z\| = \left\| \sum_{i=1}^{n} a_i x_i - z \right\| = \left\| \sum_{i=1}^{n} a_i (x_i - z) \right\| = \left\| \sum_{i=1}^{n} a_i \left(x_i - \sum_{j=1}^{n} b_j x_j \right) \right\| = \left\| \sum_{i=1}^{n} a_i \left(\sum_{j=1}^{n} b_j (x_i - x_j) \right) \right\|.$$

Using triangle inequality we obtain that

$$\begin{aligned} \|y - z\| &\leq \sum_{i=1}^{n} a_i \left(\sum_{j=1}^{n} b_j \|x_i - x_j\| \right) \leq \sum_{i=1}^{n} a_i \left(\sum_{j=1}^{n} b_j \max_{k,\, l \in [n]} \|x_k - x_l\| \right) = \max_{k,\, l \in [n]} \|x_k - x_l\| \sum_{i=1}^{n} a_i \left(\sum_{j=1}^{n} b_j \right) \\ &= \max_{k,\, l \in [n]} \|x_k - x_l\|. \end{aligned}$$

Hence, the proof. □

Lemma 8 (Proposition 6 in (Mhamdi et al., 2021))**.** *For any input vectors $x_1, \ldots, x_n \in \mathbb{R}^d$, the following holds true:*

$$\text{GM}(x_1, \ldots, x_n) \in \text{Conv}(x_1, \ldots, x_n).$$

For a non-empty set $S \subseteq [n]$. In the remaining, we denote by $\{x_i\}_{i \in S}$ the set of vectors which index is in S, i.e., $\{x_i,\ i \in S\}$.

Lemma 9 (Theorem 1 (Part 1) in (Mhamdi et al., 2021))**.** *For any set $S \subseteq [n]$ such that $|S| > n/2$,*

$$\|\text{GM}(x_1, \ldots, x_n) - \text{GM}\left(\{x_i\}_{i \in S}\right)\| \leq \frac{1}{\sqrt{1 - \frac{(n - |S|)^2}{|S|^2}}} \max_{j \in S} \|x_j - \text{GM}\left(\{x_i\}_{i \in S}\right)\|.$$

By combing the above lemmas, we can devise the following result.

Proposition 7. *If $f < n/2$ then the* GM *is an $(f,\, \lambda)$-resilient averaging rule for $\lambda = 1 + \frac{n - f}{\sqrt{(n - 2f)n}}$.*

Proof. Consider any set $S \subseteq [n]$ such that $|S| = n - f > n/2$. By triangle inequality we obtain that

$$\|\text{GM}(x_1, \ldots, x_n) - \overline{x}_S\| \leq \|\text{GM}(x_1, \ldots, x_n) - \text{GM}\left(\{x_i\}_{i \in S}\right)\| + \|\text{GM}\left(\{x_i\}_{i \in S}\right) - \overline{x}_S\|.$$

Substituting from Lemma 9 above we obtain that

$$\|\text{GM}(x_1, \ldots, x_n) - \overline{x}_S\| \leq \frac{n - f}{\sqrt{(n - 2f)n}} \max_{j \in S} \|x_j - \text{GM}\left(\{x_i\}_{i \in S}\right)\| + \|\text{GM}\left(\{x_i\}_{i \in S}\right) - \overline{x}_S\|. \tag{72}$$

From Lemma 8, we know that $\text{GM}\left(\{x_i\}_{i \in S}\right) \in \text{Conv}\left(\{x_i\}_{i \in S}\right)$. Thus, owing to Lemma 7, we have

$$\|x_j - \text{GM}\left(\{x_i\}_{i \in S}\right)\| \leq \max_{k,l \in S} \|x_k - x_l\|,\ \forall j \in S.$$

Similarly, as $\overline{x}_S \in \text{Conv}\left(\{x_i\}_{i \in S}\right)$, we get

$$\|\text{GM}\left(\{x_i\}_{i \in S}\right) - \overline{x}_S\| \leq \max_{k,l \in S} \|x_k - x_l\|.$$

Using these in (72) we obtain that

$$\|\text{GM}(x_1,\ldots,x_n)-\overline{x}_S\| \leq \frac{n-f}{\sqrt{(n-2f)n}}\max_{i,j\in S}\|x_i-x_j\| + \max_{i,j\in S}\|x_i-x_j\| = \left(1+\frac{n-f}{\sqrt{(n-2f)n}}\right)\max_{i,j\in S}\|x_i-x_j\|\,.$$

As S is an arbitrary subset of $[n]$ of size $n-f$, by Definition 2, the above proves the proposition. □

C.7. Coordinate-Wise Median (CWMed)

For input vectors $x_1,\ldots,x_n$, their coordinate-wise median, denoted by $\text{CWMed}(x_1,\ldots,x_n)$, is defined to be a vector whose k-th coordinate, for all $k\in[d]$, is defined to be

$$[\text{CWMed}\,(x_1,\ldots,x_n)]_k \coloneqq \text{Median}\,([x_1]_k,\ldots[x_n]_k)\,. \tag{73}$$

Before analyzing CWMed, we prove a useful lemma for the median operator.

Lemma 10. *Consider a set of n real numbers $\{y_1,\ldots,y_n\}$. If $f<n/2$ then for any subset $S\subseteq[n]$ with $|S|=n-f$ we obtain that*

$$|\{i\in S\,|\,y_i\leq \text{Median}(y_1,\ldots,y_n)\}|\geq\frac{n}{2}-f \quad \textit{and} \quad |\{i\in S\,|\,y_i\geq \text{Median}(y_1,\ldots,y_n)\}|\geq\frac{n}{2}-f. \tag{74}$$

Proof. Consider an arbitrary set $S\subseteq[n]$ with $|S|=n-f$. By the definition of the median operator, we have

$$|\{i\in[n]\,|\,y_i\leq \text{Median}(y_1,\ldots,y_n)\}|\geq\frac{n}{2} \quad \text{and} \quad |\{i\in S\,|\,y_i\geq \text{Median}(y_1,\ldots,y_n)\}|\geq\frac{n}{2}.$$

x As $|S|=n-f>f$, the proof follows immediately from above. □

Proposition 8. *If $f<n/2$ then* CWMed *is an $(f,\,\lambda)$-resilient averaging rule for $\lambda=\frac{n}{2(n-f)}\min\left\{2\sqrt{n-f},\sqrt{d}\right\}$.*

Proof. Consider a $S\subset[n]$ such that $|S|=n-f$. As $f<n/2$, from Lemma 10 we obtain that

$$\min_{i\in S}[x_i]_k \leq \text{Median}\,([x_1]_k,\ldots[x_n]_k)\leq\max_{i\in S}[x_i]_k.$$

This implies that

$$\text{Median}\,([x_1]_k,\ldots[x_n]_k)-(\max_{i\in S}[x_i]_k-\min_{i\in S}[x_i]_k)\leq\min_{i\in S}[x_i]_k. \tag{75}$$

Note that, by Lemma 10, at least $n/2-f$ values in $\{y_i,\ i\in S\}$ are greater than or equal to $\text{Median}\,([x_1]_k,\ldots[x_n]_k)$. Thus, as the remaining $n/2$ values in $\{y_i,\ i\in S\}$ are greater than or equal to $\min_{i\in S}[x_i]_k$, we obtain that

$$[\overline{x}_S]_k=\frac{1}{n-f}\sum_{i\in S}[x_i]_k\geq\frac{1}{n-f}\left(\left(\frac{n}{2}-f\right)\text{Median}\,([x_1]_k,\ldots[x_n]_k)+\frac{n}{2}\min_{i\in S}[x_i]_k\right)$$

Substituting from (75) above we obtain that

$$\begin{aligned}[\overline{x}_S]_k &\geq \frac{1}{n-f}\left(\left(\frac{n}{2}-f\right)\text{Median}\,([x_1]_k,\ldots[x_n]_k)+\frac{n}{2}\left(\text{Median}\,([x_1]_k,\ldots[x_n]_k)-(\max_{i\in S}[x_i]_k-\min_{i\in S}[x_i]_k)\right)\right)\\ &= \text{Median}\,([x_1]_k,\ldots[x_n]_k)-\frac{n}{2(n-f)}(\max_{i\in S}[x_i]_k-\min_{i\in S}[x_i]_k).\end{aligned} \tag{76}$$

Similarly, we can show that

$$[\overline{x}_S]_k\leq\text{Median}\,([x_1]_k,\ldots[x_n]_k)+\frac{n}{2(n-f)}(\max_{i\in S}[x_i]_k-\min_{i\in S}[x_i]_k). \tag{77}$$

From (76) and (77) we obtain that

$$|[\overline{x}_S]_k - \text{Median}\left([x_1]_k, \dots [x_n]_k\right)| \leq \frac{n}{2(n-f)} \max_{i,j \in S} ([x_i]_k - [x_j]_k).$$

Finally, substituting from Lemma 5 we obtain that

$$\begin{aligned}
\|\overline{x}_S - \text{CWMed}(x_1, \dots, x_n)\| &= \sqrt{\sum_{k \in [d]} |[\overline{x}_S]_k - \text{Median}\left([x_1]_k, \dots [x_n]_k\right)|^2} \\
&\leq \frac{n}{2(n-f)} \sqrt{\sum_{k \in [d]} \left(\max_{i,j \in S} |[x_i]_k - [x_j]_k| \right)^2} \\
&\leq \frac{n}{2(n-f)} \min\left\{2\sqrt{n-f}, \sqrt{d}\right\} \max_{i,j \in S} \|x_i - x_j\|.
\end{aligned}$$

The above concludes the proof. □

C.8. Centered Clipping (CC)

This aggregation rule was proposed by (Karimireddy et al., 2021). Specifically, given the input vectors $x_1, \dots, x_n \in \mathbb{R}^d$, upon choosing a *clipping parameter* $c_\tau \geq 0$, we compute a sequence of vectors $v_0, \dots, v_L$ in $\mathbb{R}^d$ such that for all $l \in [L]$,

$$v_l \leftarrow v_{l-1} + \frac{1}{n} \sum_{i \in [n]} (x_i - v_{l-1}) \min\left\{1, \frac{c_\tau}{\|x_i - v_{l-1}\|}\right\}$$

where v_0 may be chosen arbitrary. Then, $\text{CC}(x_1, \dots, x_n) = v_L$.

According to Karimireddy et al. (2021), by setting specific values for parameters c_τ and L, CC can satisfy the condition of (f, λ)-resilient averaging for $\lambda = 20\sqrt{10}f/n$ when $f < n/9.7$. However, they rely on extra information that is often not possible in practice. Specifically, the values for parameters c_τ and L depend on the maximal variance of the honest gradients σ, and we must also know a bound on the initial estimate error $\mathbb{E}\left[\|\overline{x_\mathcal{H}} - v_0\|^2\right]$ where $\overline{x_\mathcal{H}}$ is the average of the honest vectors. Analyzing CC under standard assumptions and without any extra information remains an open question.

C.9. Comparative Gradient Elimination (CGE)

For input vectors $x_1, \dots, x_n$, let τ denote a permutation on $[n]$ that sorts the input vectors based on their norm and in non-decreasing order, i.e., $\left\|x_{\tau(1)}\right\| \leq \left\|x_{\tau(2)}\right\| \leq \dots \leq \left\|x_{\tau(n)}\right\|$. CGE outputs the average of the $n - f$ vectors with smallest norm (Gupta et al., 2021), i.e.,

$$\text{CGE}(x_1, \dots, x_n) = \frac{1}{n-f} \sum_{i=1}^{n-f} x_{\tau(i)}.$$

In general, CGE is *not* resilient averaging as shown below using a counter-example.

Counter-example. Consider input vectors $x_1, \dots, x_n$ and a subset $S \subset [n]$ with $|S| = n - f$ such that $x_i = x$ for all $i \in S$ where $\|x\| > 0$. If $\|x_j\| < \|x\|$ for all $j \in S \setminus [n]$, and $\sum_{j \in [n] \setminus S} x_j \neq f \times x$ then

$$\text{CGE}(x_1, \dots, x_n) = \frac{1}{n-f} \left(\sum_{j \in [n] \setminus S} x_j + (n - 2f)\, x \right) \neq x.$$

As $\overline{x_S} = x$ and $\max_{i,\, j \in S} \|x_i - x_j\| = 0$, from above we obtain that, for all $\lambda \geq 0$,

$$\|\text{CGE}(x_1, \dots, x_n) - \overline{x_S}\| = \|\text{CGE}(x_1, \dots, x_n) - x\| > 0 = \lambda \max_{i,\, j \in S} \|x_i - x_j\|.$$

Thus, by Definition 2, CGE is *not* resilient averaging. □

D. Additional Information on the Experimental Setup

D.1. Attacks Simulating Byzantine Behavior

In the experiments of this paper, we use four state-of-the-art attacks that we refer to as *empire* (Xie et al., 2019a), *little* (Baruch et al., 2019), *sign-flipping* (Allen-Zhu et al., 2020), and *label-flipping* (Allen-Zhu et al., 2020). The first two attacks rely on the same core idea. Let ζ be fixed a non-negative real number and let a_t be the attack vector at time step t. At every time step t, all Byzantine workers send $\overline{g_t} + \zeta a_t$ to the server, where $\overline{g_t}$ is an estimate of the true gradient at step t. The specific details of these attacks are mentioned below.

- **Fall of Empires.** In this attack, $a_t = -\overline{g_t}$. All Byzantine workers thus send $(1-\zeta)\overline{g_t}$ at step t. In our experiments, we set $\zeta = 1.1$ for *empire*, corresponding to $\epsilon = 0.1$ in the notation of the original paper.
- **Little is Enough.** In this attack, $a_t = -\sigma_t$, where σ_t is the opposite vector of the coordinate-wise standard deviation of $\overline{g_t}$. In our experiments, we set $\zeta = 1$ for *little*.

The remaining attacks rely on different primitives. Specifically, they are defined as follows.

- **Sign-flipping.** In this attack, every Byzantine worker sends the negative of its gradient to the server.
- **Label-flipping.** In this attack, every Byzantine worker computes its gradient on flipped labels before sending it to the server. Since the labels for MNIST, Fashion-MNIST, and CIFAR-10 are in $\{0, 1, ..., 9\}$, the Byzantine workers flip the labels by computing $l' = 9 - l$ for every training datapoint of the batch, where l is the original label and l' is the flipped/modified label.

D.2. Dataset Pre-processing

MNIST receives an input image normalization of mean 0.1307 and standard deviation 0.3081. Fashion-MNIST is horizontally flipped. CIFAR-10 is horizontally flipped and we apply a per-channel normalization with means $0.4914, 0.4822, 0.4465$ and standard deviations $0.2023, 0.1994, 0.2010$.

D.3. Detailed Model Architecture

In this section, we discuss the different models tested in our experimental study. In particular, we experimented with one *convolutional* model and one simple *feed-forward neural network* for both MNIST and Fashion-MNIST, as well as one *convolutional* model for CIFAR-10. In order to present the architecture of the different models, we use the compact notation introduced in (El Mhamdi et al., 2021b).

L(#outputs) represents a **fully-connected linear layer**, R stands for **ReLU activation**, S stands for **log-softmax**, C(#channels) represents a **fully-connected 2D-convolutional layer** (kernel size 3, padding 1, stride 1), M stands for **2D-maxpool** (kernel size 2), B stands for **batch-normalization**, and D represents **dropout** (with fixed probability 0.25).

Convolutional Model for CIFAR-10. The convolutional model used for CIFAR-10, introduced in (Baruch et al., 2019), can thus be written in the following way:

(3,32×32)-C(64)-R-B-C(64)-R-B-M-D-C(128)-R-B-C(128)-R-B-M-D-L(128)-R-D-L(10)-S.

Convolutional Model for (Fashion-)MNIST. We adopt the same notation introduced earlier, with the only difference that C(#channels) now represents a fully-connected 2D-convolutional layer of kernel size 5, padding 0, and stride 1. The convolutional model we used for MNIST and Fashion-MNIST can thus be written in the following way:

C(20)-R-M-C(20)-R-M-L(500)-R-L(10)-S.

Simple Feed-forward Network for (Fashion-)MNIST. We consider a feed-forward neural network composed of two fully-connected linear layers of respectively 784 and 100 inputs (for a total of $d = 79\,510$ parameters) and terminated by a *softmax* layer of 10 dimensions. ReLU is used between the two linear layers. For this particular model, we used the Cross

Entropy loss, a total number of workers $n = 15$, a constant learning rate $\gamma = 0.5$, and a clipping parameter $C = 2$. We also add an ℓ_2-regularization factor of 10^{-4}. Note that some of these constants are reused from the literature on BR, especially from (Baruch et al., 2019; Xie et al., 2019a; El Mhamdi et al., 2021b).

E. Additional Experimental Results

Reproducibility. All our experiments (training + graphs) are reproducible in one command. Please see `code/README.md` in the supplementary material. Additional graphs are available in `plots/`.

E.1. Results on Fashion-MNIST

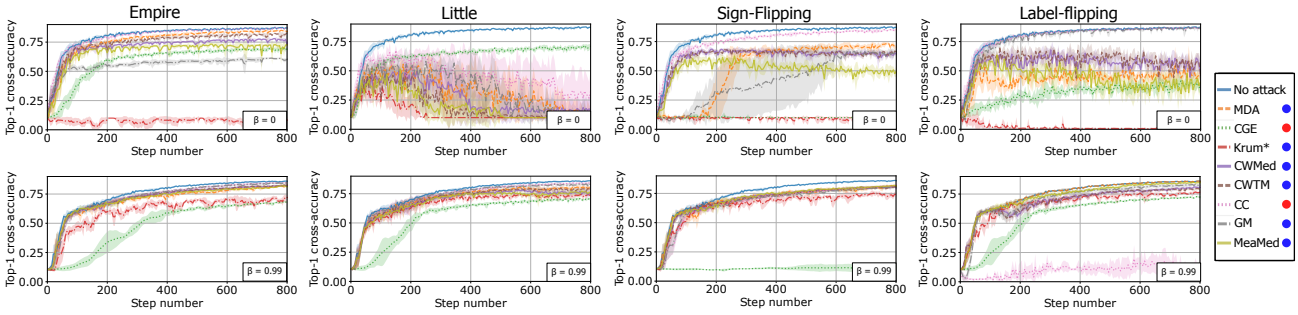

Figure 3. The 1st and 2nd rows correspond to experiments performed on Fashion-MNIST with $\beta = 0$ and $\beta = 0.99$, respectively. The different columns show the performance of the learning under the *empire*, *little*, *sign-flipping*, and *label-flipping* attacks with $f = 5$ Byzantine workers.

We also perform experiments (similar to those described in Section 5) on the Fashion-MNIST dataset. In Figure 3, we display the top-1 cross accuracies achieved by different aggregation rules on the Fashion-MNIST dataset in a distributed system comprising $n = 15$ workers, out of which $f = 5$ are Byzantine executing four different state-of-the-art attacks. We compare the performances under two momentum settings: $\beta = 0$ (i.e., momentum is not used) and $\beta = 0.99$.
We can clearly see from Figure 3 the improvement that momentum brings to the learning in every single Byzantine setting (i.e., in each of the four attack scenarios), especially for the six resilient averaging aggregation rules (MDA, CWTM, CWMed, MeaMed, Krum*, and GM). However, the performance of CGE seems unaffected by the increase in momentum especially under the *empire*, *little*, and *sign-flipping* attacks. Furthermore, CC displays poor performance under *little* for $\beta = 0$ and under *label-flipping* for $\beta = 0.99$, indicating that there always seems to exist at least one setting where CC (and CGE) display poor performance. All these observations clearly echo the main takeaway of our experiments in Section 5, where using **both** a (f, λ)-resilient averaging aggregation rule together with momentum seems to be crucial to mitigate the effect of Byzantine workers and dramatically improve the learning in an arbitrary adversarial setting (i.e., when the executed attack is not known beforehand).

E.2. The case of CC - $\beta = 0.9$

In Figure 4, we show the performance of CC (which is not an (f, λ)-resilient averaging rule) on the MNIST and Fashion-MNIST datasets, with $\beta = 0.9$ and $f = 5$ Byzantine workers. CC displays good performance against all four attacks for that particular value of β. Essentially, CC can consistently work for some values of momentum ($\beta = 0.9$), while others significantly deteriorate its performance in some cases (see $\beta = 0.99$ in Figures 1 and 2 of the main paper). Precisely characterizing the impact of momentum on CC's performance remains arguably an open question.

E.3. Results on MNIST With 7 Byzantine Workers

In this paragraph, we present some learning performances on the MNIST dataset in four adversarial settings where $f = 7$ out of 15 workers are Byzantine. It turns out that in such an extreme adversarial scenario where f reaches the maximum tolerable value of $\lfloor \frac{n}{2} \rfloor$, an even larger value of β, and thus more learning steps, are needed to guarantee a good performance in the presence of Byzantine workers. In Figure 5, we consider two values for β (0 and 0.999), and showcase the advantages of using momentum in such a setting.

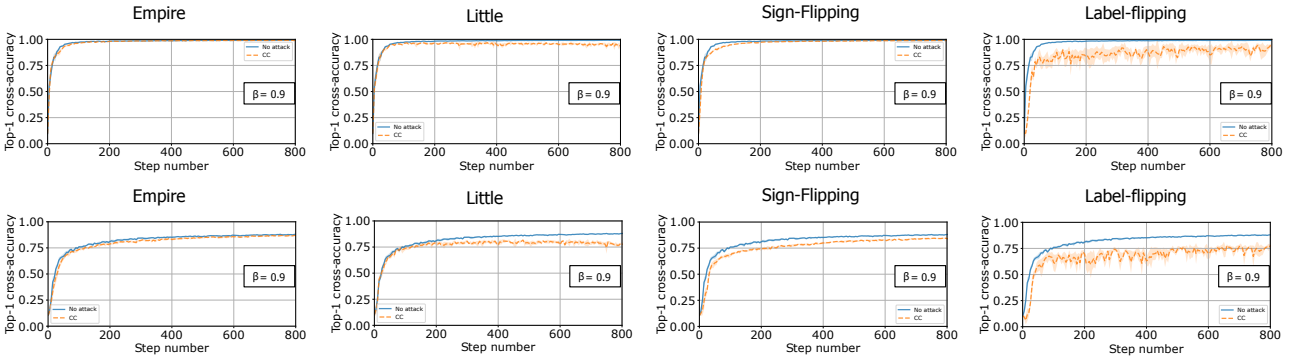

Figure 4. The 1st and 2nd rows correspond to experiments performed on MNIST and Fashion-MNIST, respectively, using $\beta = 0.9$ and the CC aggregation rule. The different columns show the performance of the learning under the *empire*, *little*, *sign-flipping*, and *label-flipping* attacks with $f = 5$ Byzantine workers. The "No attack" curve is also provided as a baseline for comparison.

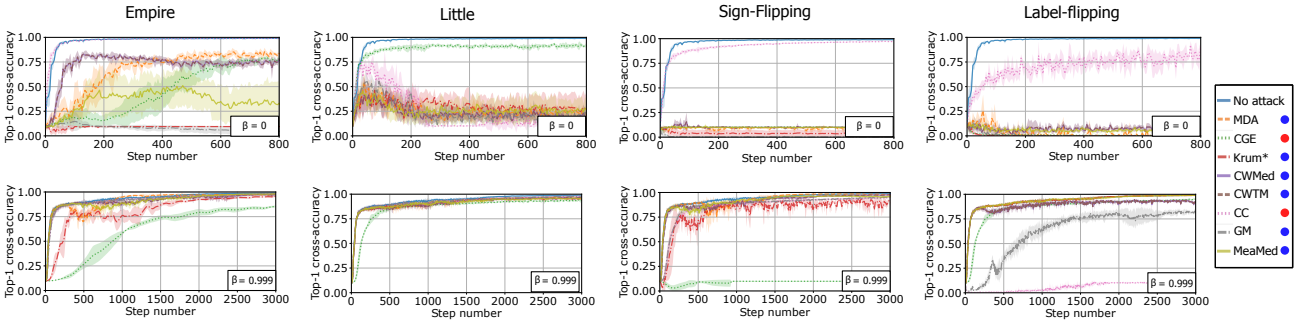

Figure 5. The 1st and 2nd rows correspond to experiments performed on MNIST with $\beta = 0$ and $\beta = 0.999$, respectively. The different columns show the performance of the learning under the *empire*, *little*, *sign-flipping*, and *label-flipping* attacks with $f = 7$ Byzantine workers. Note that in this experiment, we use Multi-Krum* with $q = n - f$ instead of Krum*.

The observations to be made here are very similar to the ones already stated in Sections 5.2 and E.1. In a few words, we can clearly see that setting β to 0.999 improves the top-1 cross-accuracies of all six (f, λ)-resilient averaging aggregation rules. However, for the two non-resilient averaging rules (CC and CGE), momentum need not improve the learning.