Byzantine Machine Learning Made Easy By Resilient Averaging of Momentums

Sadegh Farhadkhani Rachid Guerraoui Nirupam Gupta

Rafaël Pinot ***John Stephan***

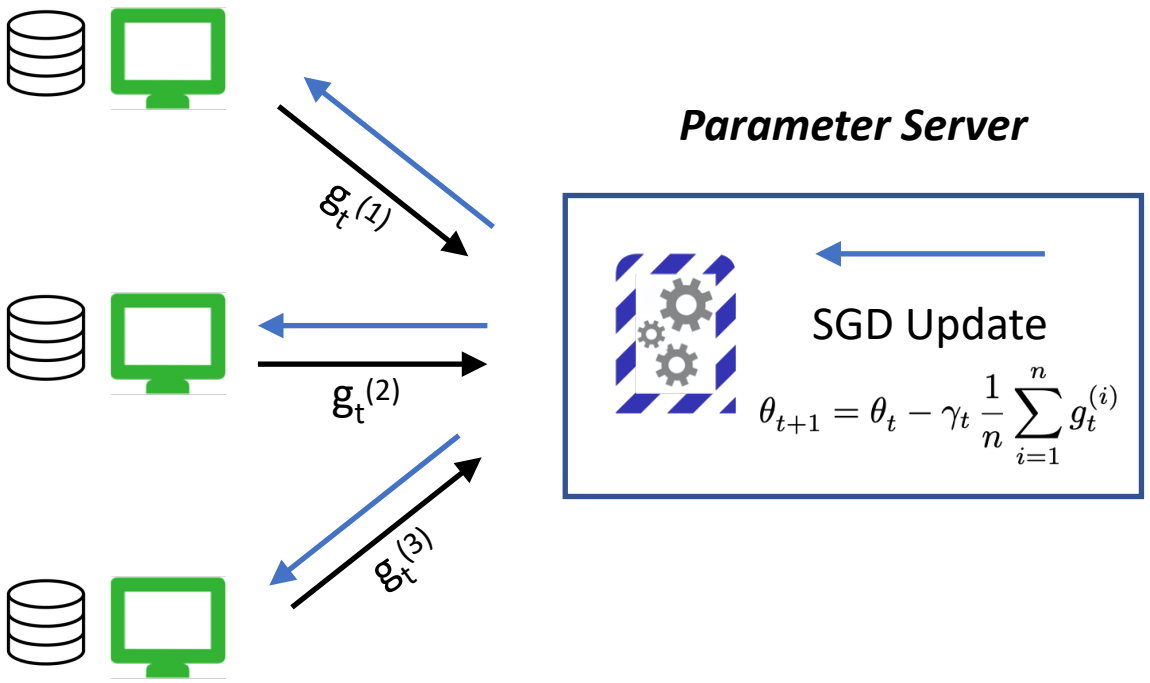
Distributed Computing Laboratory (DCL)

EPFL

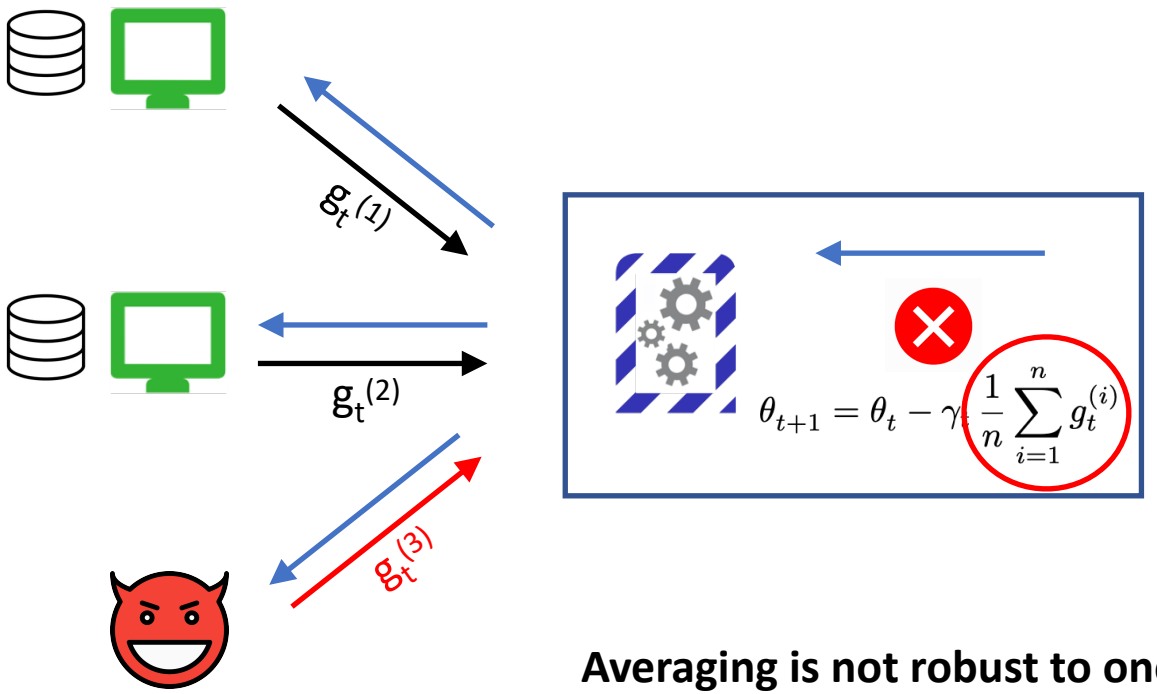
ecocloud

an EPFL research center

Distributed Stochastic Gradient Descent (SGD)

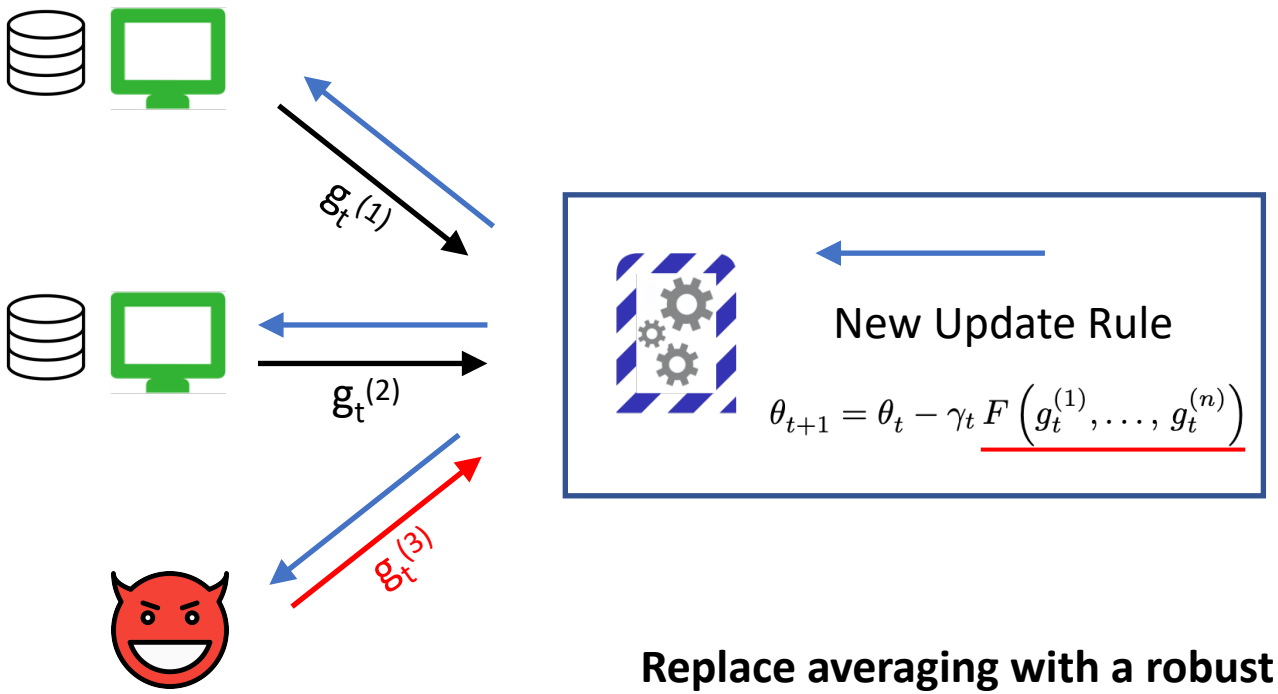


Byzantine Threat Model



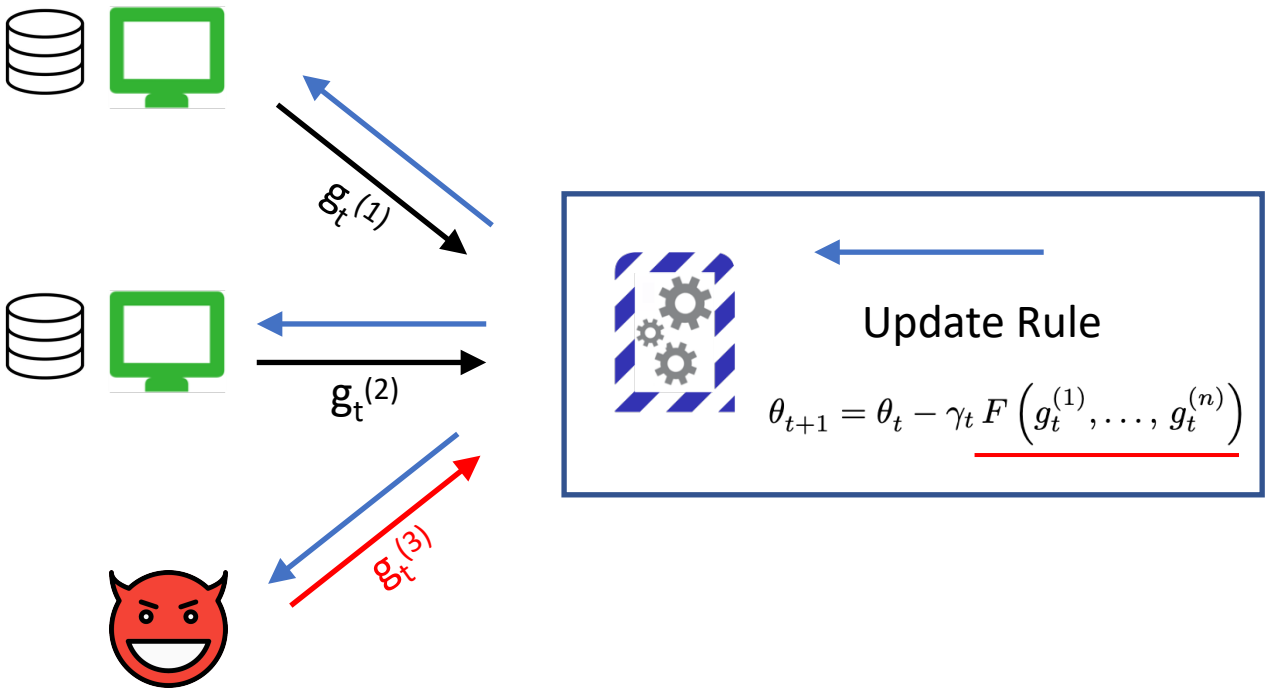
Averaging is not robust to one Byzantine machine!

Byzantine-Resilient SGD



Replace averaging with a robust aggregation rule F

Brittleness of Previous Works



Previous works make **non-standard** assumptions on

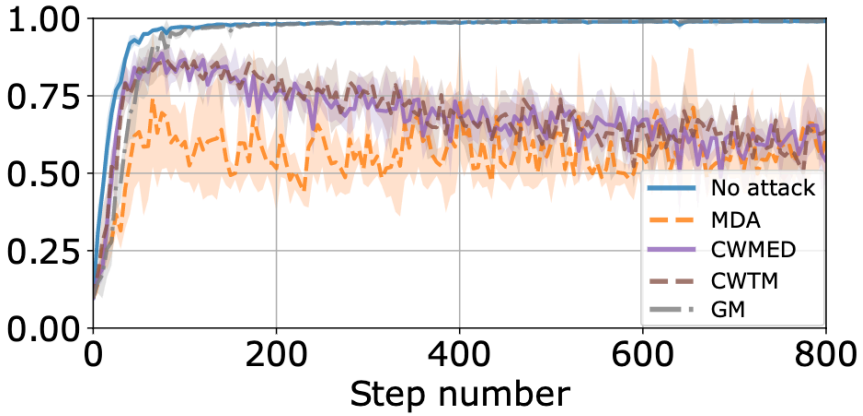
- The number of Byzantine machines
- The stochastic gradients

  E.g., sub-exponential, vanishing uncertainty

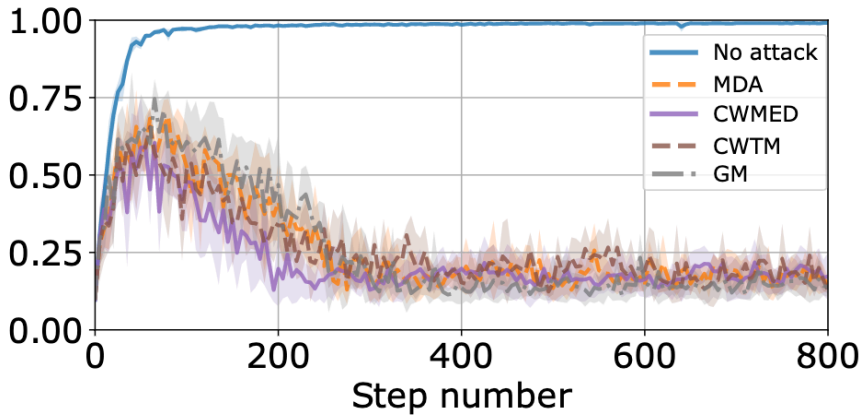
Theoretically: Impossible to compare them
Empirically: Vulnerable to attacks

Brittleness of Previous Works

Fall of Empires (1)



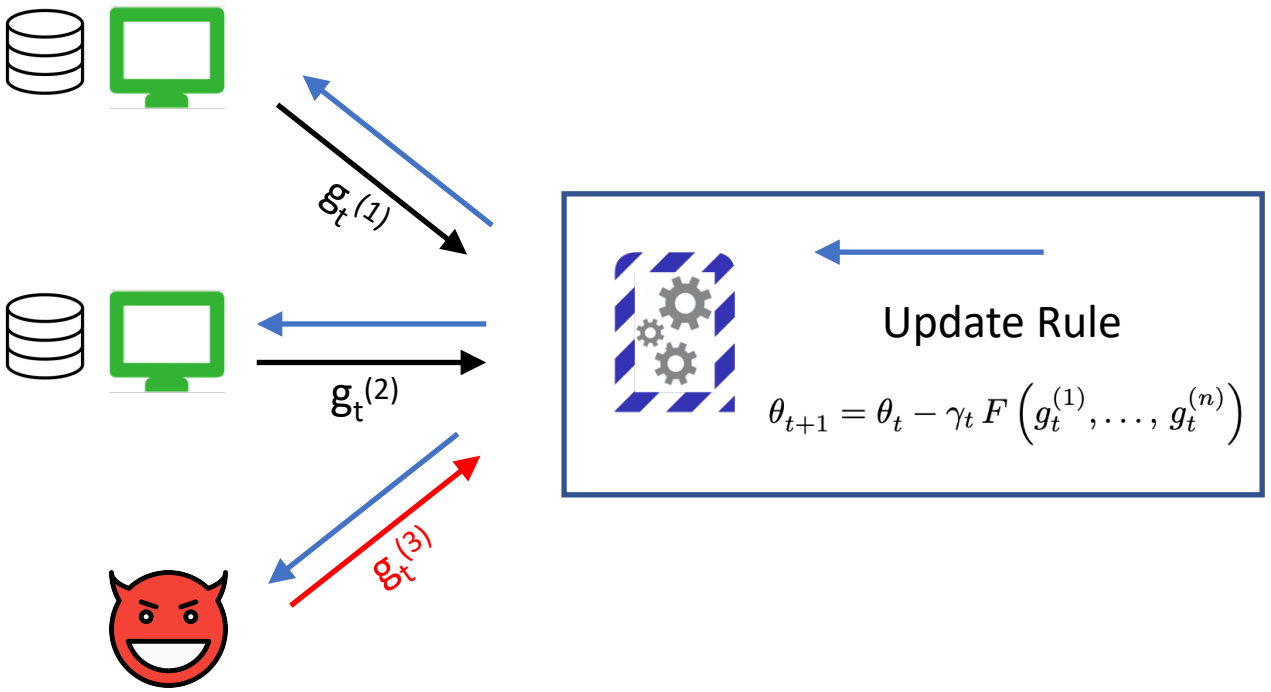
A Little is Enough (2)



(1) 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.

(2) 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.

Our Contribution: RESAM



Prior works make **non-standard** assumptions on

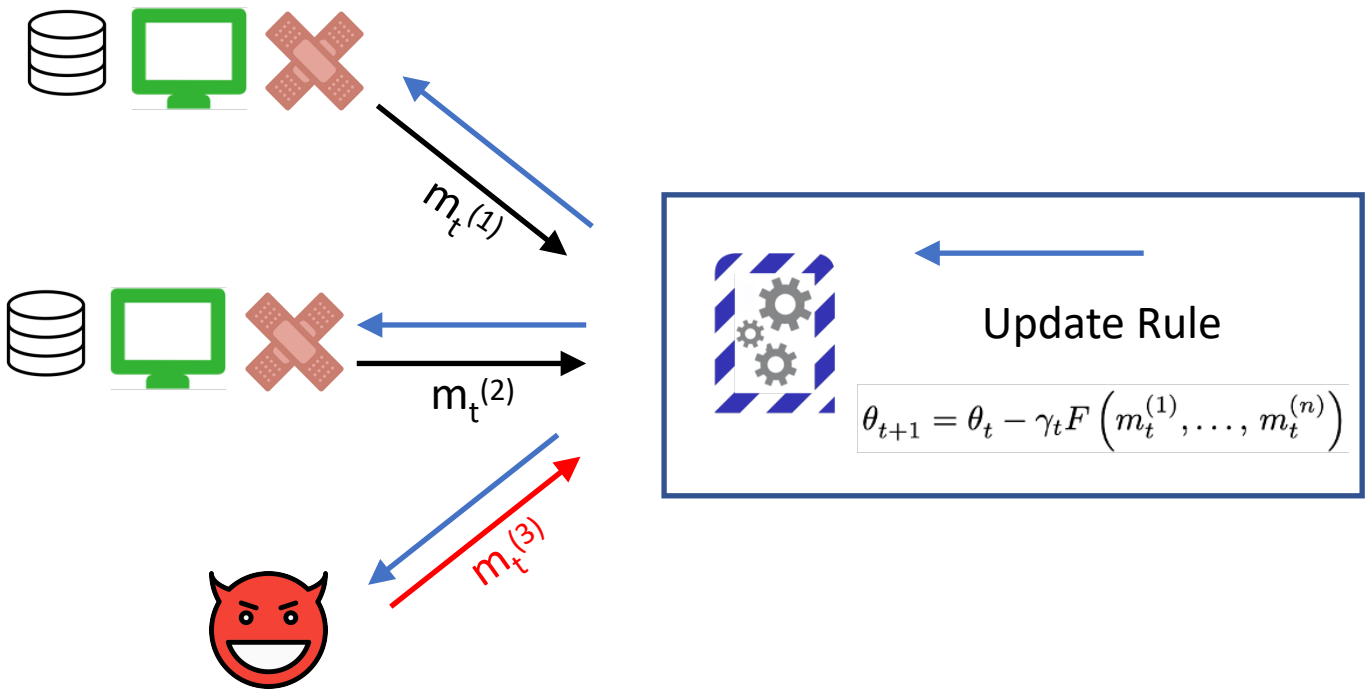
- The number of Byzantine machines
- The stochastic gradients

  E.g., sub-exponential, vanishing uncertainty

Eliminate all **non-standard** assumptions

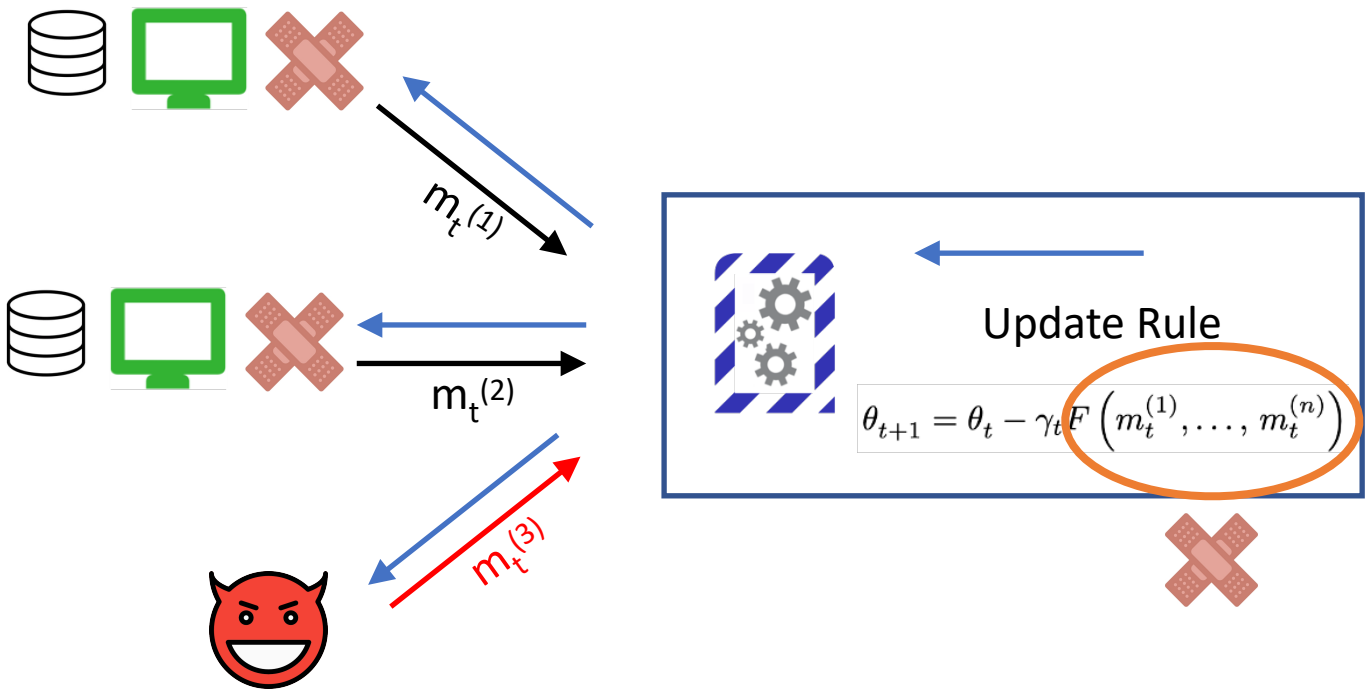
- Provide unified theoretical framework to compare aggregation rules
- Optimal in number of Byzantine machines
- Works in practice

Patch 1: Polyak's Momentum



Apply **Polyak's momentum** on honest machines before sending the gradients to the server

Patch 2: Resilient Averaging Criterion



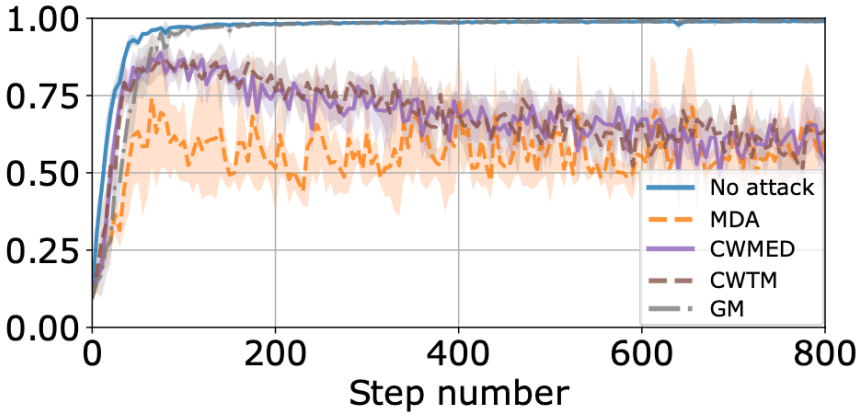
New resilience criterion that F must satisfy

- Encompasses most existing rules
- Enables us to unify the field and compare existing rules

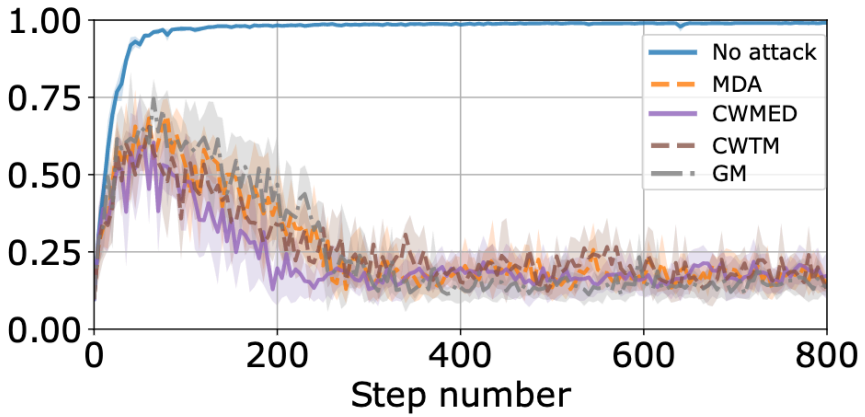
Experiments: RESAM vs. Previous Works

Previous works

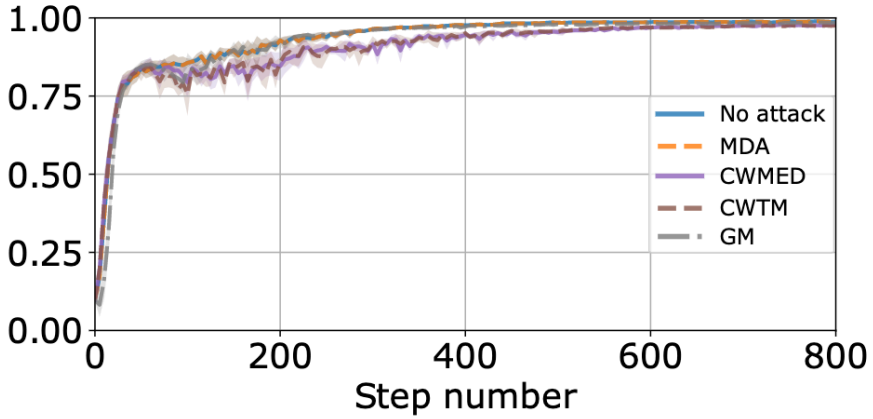
Fall of Empires

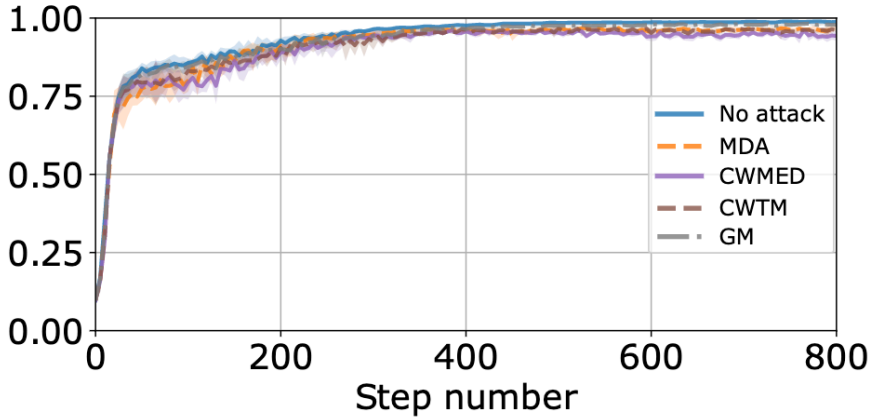


A Little is Enough



RESAM





Thanks for Listening!

arXiv:2205.12173v1 [cs.LG] 24 May 2022

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.

*Authors listed in alphabetical order. [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>.

[Accepted for] *Proceedings of the 39th International Conference on Machine Learning*, Baltimore, Maryland, USA, PMLR 162, 2022.

→ Check the full paper

`ICML2022,paper1455`

→ Drop us an email directly

`{firstname}.{lastname}@epfl.ch`