Robust and Scalable Collaborative Learning via Pull-Based Epidemic Communication

Abdellah El Mrini* **Sadegh Farhadkhani**

Rachid Guerraoui
Ecole Polytechnique Fédérale de Lausanne (EPFL), Switzerland

Abstract

Collaborative machine learning is vulnerable to adversarial behaviors during training. Existing defenses typically rely on central coordination or induce high communication costs. We introduce *Robust Pull-based Epidemic Learning (RPEL)*, a scalable and fully decentralized framework that achieves robustness without a central server. Unlike traditional methods, whose communication cost grows as $\mathcal{O}(n^2)$ with the number of nodes n, RPEL uses a pull-based epidemic communication scheme that scales as $\mathcal{O}(n \log n)$. By pulling model parameters from small, randomly selected subsets of peers, RPEL significantly lowers the number of required messages while preserving convergence guarantees with high probability. Experimental results demonstrate that RPEL indeed tolerates attacks, attains accuracy comparable with all-to-all communication, and scales efficiently to large networks.

1 Introduction

Collaborative machine learning has recently gained traction as a solution for data-intensive, privacy-sensitive tasks where data resides across multiple nodes (Kairouz et al., 2021; McMahan et al., 2017). This approach, however, faces significant challenges in the presence of malfunctioning or malicious nodes that can disrupt the learning process by sending corrupted or misleading updates (Lamport et al., 2019; Blanchard et al., 2017). Robustness against so-called Byzantine attacks is critical, especially as machine learning applications expand into sensitive domains such as healthcare, autonomous systems, and finance, where incorrect model updates could lead to catastrophic outcomes. Prior approaches that cope with adversaries have relied on robust aggregation to mitigate their impact. However, those approaches generally required a trusted centralized server (Blanchard et al., 2017; Yin et al., 2018; Chen et al., 2017; Allen-Zhu et al., 2020; Karimireddy et al., 2020; Allouah et al., 2023).

The decentralized, peer-to-peer communication setting offers a promising alternative for collaborative machine learning, eliminating the need for a central coordinator, thereby enhancing scalability (Lian et al., 2017; Koloskova et al., 2019; Vogels et al., 2023; Lian et al., 2018; Lu and Wu, 2020). Yet, the absence of a trusted server exacerbates the challenge of ensuring robustness. Existing robust peer-to-peer learning algorithms generally depend on fully-connected (or at least dense) networks, with communication complexity that scales quadratically with the number of nodes (Farhadkhani et al., 2023; Gorbunov et al., 2022; El Mhamdi et al., 2021a).The question of whether it is possible to devise a scalable, communication-efficient scheme that can uphold robustness across heterogeneous nodes in decentralized peer-to-peer networks remained open.

We contribute to answering this question in the affirmative by presenting *Robust Pull-based Epidemic Learning (RPEL)*, a novel collaborative learning approach that uses a randomized, asynchronous

*Correspondence to : Abdellah El Mrini <abdellah.elmrini@epfl.ch>.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

epidemic-based communication scheme in which each node periodically pulls model updates from a small, random subset of peers. This approach significantly decreases the communication burden (compared to traditional peer-to-peer solutions) while preserving robustness against adversarial nodes. RPEL leverages the convergence properties of epidemic-based protocols, ensuring that information about the model parameters spreads rapidly across nodes with minimal communication overhead.

A key technical difficulty in robust peer-to-peer settings lies in determining the number of attackers each honest node could be exposed to. This information is crucial for designing effective robust aggregation rules, e.g. (Blanchard et al., 2017; Yin et al., 2018). The possibility of the attackers' positions on the communication graph to be chosen in an adversarial manner adds an extra layer of complexity. Consequently, most current works typically make strong assumptions about the graph connectivity and induce a high communication cost (Gaucher et al., 2025; He et al., 2022; Gaucher et al., 2024). We show that such a high connectivity and communication cost is not needed. Essentially, RPEL converts the worst-case graph-neighborhood adversary bound used in robust decentralized learning into a high-probability sampling bound, reducing per-round degree from $\Theta(n)$ to $\mathcal{O}(\log(n))$.

Summary of contributions.

We introduce RPEL, a pull-based epidemic learning protocol for Byzantine-robust decentralized optimization. We prove finite-time convergence for asynchronous non-convex learning under heterogeneity, and establish robustness under strong omniscient attacks by introducing the Effective adversarial fraction, a high-probability bound on sampled attackers. Experiments on MNIST and CIFAR-10 show accuracy comparable to all-to-all robust methods and stronger robustness than fixed-graph baselines at the same communication budget.

2 Related Work

Byzantine distributed learning. Learning in a distributed fashion is appealing but challenging in the presence of Byzantine adversaries, namely nodes that may deliberately send malicious or corrupted updates. In the context of Federated Learning, i.e., when a trusted server is available to orchestrate the learning procedure, the use of robust aggregation rules can effectively mitigate the influence of an adversary (Allouah et al., 2023; Yin et al., 2018; Blanchard et al., 2017).

Robust Decentralized Learning. Decentralized Learning (DL) removes the trusted server and allows the clients to communicate directly with each other. The most popular DL approach is gossip averaging, in which each client iteratively updates their current model using a weighted combination of their neighbors' models (Boyd et al., 2005; Berthier et al., 2020). This approach fails however in the presence of attacks (Blanchard et al., 2017). More recent DL approaches were proposed to ensure robustness, including NNA (Farhadkhani et al., 2023), ClippedGossip (He et al., 2022), CS+ (Gaucher et al., 2025) and RTC (Yang and Ghaderi, 2024). These methods, however, obtain guarantees either in the all-to-all or sufficiently connected fixed graphs. In sparse graphs, the guarantees often depend on the honest subgraph and adversary placement. In particular, ensuring that every honest node has enough honest neighbors typically requires connectivity growing with the number of Byzantine nodes, which is prohibitive at scale. RPEL avoids fixed worst-case neighborhoods by resampling peers each round and controlling the number of sampled adversaries with high probability.

Epidemic Learning. The advantages of randomization in peer-to-peer learning were highlighted by De Vos et al. (2024), which introduced Epidemic Learning as a powerful tool to address the high communication costs that often hinder the practical deployments of Decentralized Learning (Koloskova et al., 2019). In this paper, we introduce a Pull variant of Epidemic Learning that differs from the one originally studied by De Vos et al. (2024). We demonstrate that pulling preserves the advantages of Epidemic Learning while ensuring robustness in adversarial settings.

3 Problem Statement

3.1 Setting

We consider a decentralized, serverless system comprising n nodes $\mathcal{N} = \{1, 2, \ldots, n\}$. Each node i has access to a local dataset sampled from a distribution $\mathcal{D}_i$, which may differ across nodes, reflecting

non-IID data conditions. The nodes collectively aim to minimize a global objective function:

$$\min_{\mathbf{x}\in\mathbb{R}^d} F(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{x}),$$

where $f_i(\mathbf{x}) = \mathbb{E}_{\xi\sim\mathcal{D}_i}[\ell(\mathbf{x},\xi)]$ is the local loss function for node i, and $\mathbf{x}\in\mathbb{R}^d$ represents the model parameters.

3.2 Threat model

In this setting, up to $b < \lfloor n/3 \rfloor$ nodes may exhibit Byzantine behavior, arbitrarily deviating from the prescribed protocol. Controlled by an omniscient adversary, these nodes can send different, potentially malicious updates. The omniscient nature of these attacks implies the adversaries have full knowledge of the honest nodes' updates, the selected nodes, and how the aggregation is executed.

We denote by $\mathcal{H} \subset \mathcal{N}$ the set of honest nodes in the system. As truthful information about the adversaries' datasets cannot be guaranteed, the objective for the honest clients is to minimize:

$$\min_{\mathbf{x}\in\mathbb{R}^d} F_{\mathcal{H}}(\mathbf{x}) = \frac{1}{|\mathcal{H}|}\sum_{i\in\mathcal{H}} f_i(\mathbf{x}). \tag{1}$$

To formalize the robustness of our algorithm, we use the following notion of Byzantine resilience

Definition 3.1 (Byzantine Resilience)**.** A decentralized algorithm is said to be (b,ε)-resilient if, despite the presence of b Byzantine clients, each honest client $i\in\mathcal{H}$ computes $\hat{\mathbf{x}}_i$ such that:

$$\mathbb{E}\left[\left\|\nabla F_{\mathcal{H}}(\hat{\mathbf{x}}_i)\right\|^2\right] \leq \varepsilon,$$

where the expectation is taken over the randomness of the algorithm.

3.3 Communication Model

The nodes communicate via a peer-to-peer network, using an epidemic-style communication strategy:

- **Randomized Communication**: In each communication round, all honest nodes pull model updates from a randomly chosen subset of s peers. This pull-based approach to Epidemic Learning was not explored in the work of De Vos et al. (2024), which focused on the non-robust push-based Epidemic Learning where nodes select recipients for their model updates. The pull-based approach is especially important in Byzantine settings because it prevents attackers from injecting malicious updates to all the honest nodes at each iteration. Similar to De Vos et al. (2024), RPEL assumes nodes can address any peer, but at each round, only a fraction of the nodes is contacted.
- **Asynchronous setting**: Honest nodes request model updates from their peers without any assumptions on the response timing. This asynchrony makes robustness more challenging, as Byzantine nodes may delay or withhold responses, potentially stalling honest nodes indefinitely. At the same time, it makes the approach more practical by avoiding unrealistic synchronization or timing assumptions.

4 Robust Epidemic Learning

In this section, we first present the RPEL algorithm, designed to guarantee Byzantine robustness in the randomized peer-to-peer communication setting presented in the previous section. We then introduce the concept of Effective adversarial fraction, a key quantity in RPEL that allows a principled approach to the choice of the algorithm's hyperparameters.

4.1 Algorithm

Algorithm 1 outlines the general procedure executed by each node. This follows a pull-based Epidemic Learning approach, where each honest node randomly samples s peers, expects at most $\hat{b}$ among them to be Byzantine, waits for the first $s - \hat{b}$ responses, and robustly aggregates them with its own update.

Waiting for only $s - \hat{b}$ updates prevents Byzantine nodes from stalling progress by withholding messages, while the aggregation rule is chosen to tolerate up to $\hat{b}$ corrupted received updates.

The true number of selected adversaries is upper bounded by $\hat{b}$ with a high probability. We discuss this upper bound in more detail in Section 4.2 and Section 6.1.

A synchronous version of RPEL, along with its analysis, are presented in the Section A.4.

Algorithm 1 RPEL algorithm

Require: number of iterations T, number of neighbors s, momentum coefficient β, step size η, effective number of adversaries $\hat{b}$, Aggregation Rule $\mathcal{R}$

for $t = 1 \dots T$ **do**
  for $i \in \mathcal{H}$ in parallel **do**
    Randomly sample a data point ξ_i^t from $\mathcal{D}_i$
    Compute a local stochastic gradient $g_i^t = \nabla \ell(\mathbf{x}_i^t, \xi_i^t)$
    Update local momentum $m_i^t = \beta m_i^{t-1} + (1-\beta) g_i^t$
    Optimization step: $\mathbf{x}_i^{t+1/2} = \mathbf{x}_i^t - \eta m_i^t$
    Randomly sample a set S_i^t of s nodes and request their current models
    Initialize $\hat{S}_i^t = \varnothing$
    while $\left|\hat{S}_i^t\right| < s - \hat{b}$ **do**
      Upon receiving an update from node j, update $\hat{S}_i^t = \hat{S}_i^t \cup \{j\}$
    end while
    Aggregation step
    $\mathbf{x}_i^{t+1} = \mathcal{R}\left(\mathbf{x}_i^{t+1/2}; \left\{\mathbf{x}_j^{t+1/2}, j \in \hat{S}_i^t\right\}\right)$
  end for
end for
for $i \in \mathcal{H}$ **do**
  Output $\hat{\mathbf{x}}_i \sim \mathcal{U}\{\mathbf{x}_i^1, \dots, \mathbf{x}_i^T\}$, $\mathcal{U}$ being the uniform distribution
end for

4.2 Effective adversarial fraction

Randomized communication enables reasoning about the adversarial fraction in a manner analogous to federated learning. Although adversaries in the peer-to-peer context can have a significantly stronger impact, since they can send different updates to different clients, having guarantees in terms of the total adversarial fraction, rather than the total Byzantine weight (e.g. He et al. (2022); Gaucher et al. (2025)), is necessary from a practical standpoint.

Through randomization, at each iteration t, each honest node i encounters a random number b_i^t of attackers. When the neighbors are selected uniformly at random among all the participants, this number is drawn from a hypergeometric distribution $b_i^t \sim \mathrm{HG}(n-1, b, s)$.

For $\hat{b}$ such that $b/n < \hat{b}/s+1 < 1/3$, define the following event

$$\Gamma = \left\{\forall t \leq T, \forall i \in \mathcal{H}, b_i^t \leq \hat{b}\right\}. \tag{2}$$

We call the ratio $\frac{\hat{b}}{s+1}$ the *Effective adversarial fraction*.

The next lemma provides a sufficient condition on the number of sampled nodes s for the event Γ to hold with a high probability.

Lemma 4.1. *Let $p < 1$ and assume s and $\hat{b}$ are such that $b/n \leq \hat{b}/s+1 < 1/2$ and*

$$s \geq \min\left\{ n, D\left(\frac{\hat{b}}{s+1}, \frac{b}{n}\right)^{-1} \ln\left(\frac{T|\mathcal{H}|}{1-p}\right)\right\} - 1, \tag{3}$$

with $D(\alpha, \beta) = \alpha \ln(\alpha/\beta) + (1-\alpha)\ln(1-\alpha/1-\beta)$.

Then the event Γ holds with probability at least p.

The following lemma shows that logarithmic sampling is enough for our algorithm at scale.

Lemma 4.2. *Suppose $0 < b/n < 1/3$, and define the following sampling threshold*

$$s_* \coloneqq \left\lceil \max\left\{ \frac{4/9}{(1/3 - b/n)^2}, \frac{3}{b/n} \right\} \ln\left(\frac{27T|\mathcal{H}|}{8(1-p)}\right) \right\rceil + 2.$$

if $s \geq s_*$, then there exists $\hat{b}_*$ such that s and $\hat{b}_*$ satisfy Equation (3), $\hat{b}_*/s+1 < 1/3$ and additionally $\frac{\hat{b}_*}{s+1} \in \mathcal{O}(b/n)$.

Lemma 4.2 emphasizes that logarithmic scaling of the number of selected nodes s is sufficient to ensure that the event Γ occurs with high probability while the Effective adversarial fraction is of the same order as the total Byzantine fraction. Indeed, assuming the number of Byzantine clients grows linearly in n (which is equivalent to fixing the total Byzantine fraction b/n), one only needs to increase s logarithmically in n. This illustrates the efficiency of the RPEL framework, demonstrating that robustness can be achieved with a communication complexity of $\mathcal{O}(n \log n)$. Notice, however, that, due to the constants, this Lemma can be trivial for small values of n. Lemma A.10 in the Appendix provides a much less constraining condition on s which ensures the existence of $\hat{b}$ such that $\hat{b}/s+1 < 1/3$.

In the rest of the paper, $\hat{b}$ is set such that with probability at least p, Γ occurs. We note $\hat{h} = s + 1 - \hat{b}$, which is a lower bound on the number of honest nodes guaranteed to be selected by each node at each iteration (with probability at least p).

Remark 4.3. RPEL sends $O(Tns)$ messages over T rounds, whereas all-to-all decentralized communication sends $O(Tn^2)$ messages. Lemma 4.2 gives $s = O\left(\log \frac{Tn}{1-p}\right)$ when the Byzantine fraction is fixed away from the asynchronous breakdown threshold. Consequently, the subquadratic gain holds whenever $\log\left(\frac{Tn}{1-p}\right) = o(n)$. A complexity analysis of the local robust aggregation is included in Appendix C.2.

5 Theoretical Analysis

We now present our main theoretical results demonstrating the finite-time convergence of Algorithm 1.

5.1 Single iteration reduction

To formalize and quantify the robustness of a given aggregation rule, we use the following notion of $(m, \hat{b}, \kappa)$-robustness. This provides an upper bound on the error between the aggregation output and the honest average with the variance of the input of honest nodes.

Definition 5.1 ($(m, \hat{b}, \kappa)$-robustness (Allouah et al., 2023))**.** Let $\hat{b} < m/2$ and $\kappa \geq 0$. An aggregation rule $\mathcal{R}$ is said to be $(m, \hat{b}, \kappa)$-robust if for any vectors $v_1, \ldots, v_m \in \mathbb{R}^d$, and any set $\mathcal{U} \subseteq \{1, \ldots, m\}$ of size $m - \hat{b}$,

$$\|\mathcal{R}(v_1, \ldots, v_m) - \overline{v}_{\mathcal{U}}\|^2 \leq \frac{\kappa}{|\mathcal{U}|} \sum_{i \in \mathcal{U}} \|v_i - \overline{v}_{\mathcal{U}}\|^2,$$

with $\overline{v}_{\mathcal{U}} \coloneqq \frac{1}{|\mathcal{U}|} \sum_{i \in \mathcal{U}} v_i$.

The aggregation rule $\mathcal{R}$ used in Algorithm 1 needs to satisfy Definition 5.1 with $m = s + 1 - \hat{b}$. This enforces the condition $\hat{b}/s+1 < 1/3$. In the synchronous case (Algorithm 2), $m = s + 1$ and only $\hat{b}/s+1 < 1/2$ is required.

The next lemma demonstrates that each step of Algorithm 1, when using a robust aggregation rule, effectively reduces the variance among the honest nodes and quantifies the amount of bias introduced.

Lemma 5.2. *Assuming $\mathcal{R}$ satisfies Definition 5.1, and assuming Equation* (3) *is satisfied, the iterates of Algorithm 1 with $\mathcal{R}$ satisfy the following reduction at any iteration t:*

$$\mathbb{E}_t \left[\left\| \overline{\mathbf{x}_{\mathcal{H}}^{t+1}} - \overline{\mathbf{x}_{\mathcal{H}}^{t+1/2}} \right\|^2 \right] \leq \left(9\kappa + 3 \frac{|\mathcal{H}| - \hat{h}}{\hat{h}(|\mathcal{H}| - 1)|\mathcal{H}|} \right) V^t, \tag{4}$$

$$\mathbb{E}_t \left[\frac{1}{|\mathcal{H}|} \sum_{i \in \mathcal{H}} \left\| \mathbf{x}_i^{t+1} - \overline{\mathbf{x}_{\mathcal{H}}^{t+1}} \right\|^2 \right] \leq \left(9\kappa + 3 \frac{|\mathcal{H}| - \hat{h}}{\hat{h}(|\mathcal{H}| - 1)} \right) V^t, \tag{5}$$

with $V^t = \frac{1}{|\mathcal{H}|} \sum_{i \in \mathcal{H}} \left\| x_i^{t+1/2} - \overline{x_{\mathcal{H}}^{t+1/2}} \right\|^2$ is the variance of honest updates models at iteration t.

As a consequence, and using a classic result described in Section A.3, a way to ensure convergence of RPEL is to provide a robust aggregation rule satisfying Definition 5.1 with $3\kappa + 1/\hat{h} < \frac{1}{3}$.

5.2 Convergence rates

To prove the convergence, we use the following assumptions on the objective functions of the honest clients. The first two are standard in first-order stochastic optimization (Bottou et al., 2018), and the third one is standard in robust optimization (Allouah et al., 2024b; Data and Diggavi, 2021).

Assumption 5.3. (L-Smoothness). $\forall i \in \mathcal{H}, \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^d$, we have

$$\|\nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y})\| \leq L \|\mathbf{x} - \mathbf{y}\|.$$

Assumption 5.4. (Bounded stochastic noise). $\forall i \in \mathcal{H}, \forall \mathbf{x} \in \mathbb{R}^d$, we have

$$\mathbb{E}_{\xi \sim \mathcal{D}_i} \left[\ell(\mathbf{x}, \xi) \right] = \nabla f_i(\mathbf{x}) \quad \text{and} \quad \mathbb{E}_{\xi \sim \mathcal{D}_i} \left[\|\nabla \ell(\mathbf{x}, \xi) - \nabla f_i(\mathbf{x})\|^2 \right] \leq \sigma^2.$$

Assumption 5.5. (Bounded heterogeneity). $\forall i \in \mathcal{H}, \forall \mathbf{x} \in \mathbb{R}^d$, we have

$$\frac{1}{|\mathcal{H}|} \sum_{i \in \mathcal{H}} \|\nabla f_i(\mathbf{x}) - \nabla F_{\mathcal{H}}(\mathbf{x})\|^2 \leq G^2. \tag{6}$$

We now present the main theoretical result of our paper, showcasing the convergence of both previous aggregations in the non-convex setting.

Theorem 5.6 (Convergence of Algorithm 1)**.** *Under the assumptions 5.3, 5.4 and 5.5, assuming that s and $\hat{b}$ satisfy Equation* (3) *and $\mathcal{R}$ satisfies Definition 5.1 with $m = s + 1 - \hat{b}$, for a choice of η satisfying Equation* (23)*, for all $i \in \mathcal{H}$, after T iterations, with probability at least p on the sampling procedure*

$$\frac{1}{T} \sum_{t=1}^{T} \mathbb{E} \left[\|\nabla F_{\mathcal{H}} \left(\mathbf{x}_i^t \right)\|^2 \right] \leq 2 \sqrt{\frac{\left(c_2 L \sigma^2 + \frac{432L}{T} \left(\frac{\sigma^2}{n-b} \right) \right) c_0}{T}} + 2 \sqrt[3]{\frac{9 c_0^2 c_1 n L^2 (\sigma^2 + G^2)}{T^2}} + 12 \frac{L c_0}{T} + c_3 G^2,$$

with $c_0 = 12 \left(F_{\mathcal{H}} \left(\overline{\mathbf{x}}_{\mathcal{H}}^0 \right) - F_{\mathcal{H}}^* \right), c_1 = \frac{18\alpha(1+\alpha)}{(1-\alpha)^2}, c_2 = 72L \left(\frac{3}{n-b} + 2c_1 + \frac{9\lambda}{2}(2c_1 + 3) \right),$
and $c_3 = 6 \left(6c_1 + \frac{9\lambda}{2}(4c_1 + 9) \right)$, *where* $\alpha = 9\kappa + 3 \frac{|\mathcal{H}| - \hat{h}}{(|\mathcal{H}| - 1)\hat{h}}$ *and* $\lambda = 9\kappa + 3 \frac{|\mathcal{H}| - \hat{h}}{(|\mathcal{H}| - 1)|\mathcal{H}|\hat{h}}$.

Ignoring the constants and the higher-order terms, we deduce the following corollary.

Corollary 5.7 (Leading-order stochastic convergence rate)**.** *Under the conditions of Theorem 5.6, when $\mathcal{R}$ satisfies Definition 5.1 with $\kappa = \mathcal{O}\left(\frac{\hat{b}}{s+1}\right)$, with probability at least p, Algorithm 1 satisfies (b, ε)-resilience (Definition 3.1) with:*

$$\varepsilon \in \mathcal{O}\left(\sqrt{\frac{\hat{b}+1}{(s+1)T}\sigma^2} + \frac{\hat{b}+1}{s+1}G^2\right).$$

Remark 5.8. In the rest of the paper, we use as $\mathcal{R}$ the composition of coordinate-wise trimmed mean (CWTM) with NNM pre-aggregation, which was shown in Allouah et al. (2023) to satisfy Definition 5.1 with $\kappa = \mathcal{O}\left(\hat{b}/s+1\right)$. Section A.10 provides the exact expression of κ in that case.

Remark 5.9. Corollary 5.7 retains only the leading terms used for comparison with stochastic results. The full finite-time bound in Theorem 5.6 additionally contains the initialization-dependent $O\left(\frac{Lc_0}{T}\right)$ term and a higher-order $T^{-2/3}$ term. These terms remain when $\sigma = 0$, i.e., for full-batch gradient descent.

5.3 Discussion

The first term in Corollary 5.7 shows an expected convergence rate. In the absence of attackers, taking $s = n-1$ (all-to-all communication) recovers the classical SGD rate $\mathcal{O}\left(\sqrt{\sigma^2/nT}\right)$ (Ghadimi and Lan, 2013). In the scenario when $s = n-1$ and $\hat{b} = b$, the vanishing term becomes $\mathcal{O}\left(\sqrt{(b+1)\sigma^2/nT}\right)$, matching the best-known rate of Farhadkhani et al. (2023), which holds even with a trusted server Karimireddy et al. (2020).

The leading non-vanishing term in our convergence bound, $\mathcal{O}\left(\frac{\hat{b}+1}{s+1}G^2\right)$, depends on data heterogeneity G^2, but is independent of the stochastic-gradient noise variance σ^2. Since by Lemma 4.2 $\frac{\hat{b}}{s+1} \in \mathcal{O}\left(\frac{b}{n}\right)$, this dependence is consistent with the optimal heterogeneity scaling established by Allouah et al. (2023) in federated learning. In contrast, Allouah et al. (2024a) obtain a residual term that also depends on stochastic noise when studying Byzantine robustness with client subsampling in the federated setting. The absence of a noise-dependent leading non-vanishing term in our bound is enabled by our use of variance reduction through local momentum (Polyak, 1964) in Algorithm 1, a mechanism not employed in the federated subsampling algorithm analyzed by Allouah et al. (2024a).

6 Experiments

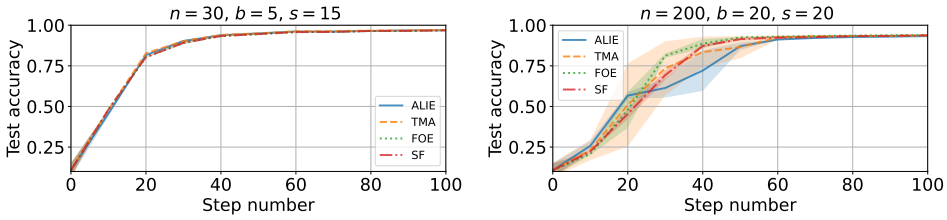


Figure 1: Test accuracies on MNIST ($\alpha = 1$)

We empirically evaluate RPEL across various settings and we benchmark it against state-of-the-art methods. We first describe the experimental setup, and then show some results on both MNIST LeCun et al. (2010) and CIFAR-10 Krizhevsky et al. (2014) datasets. Implementation details and other experiments, including the FEMNIST dataset (Caldas et al., 2019) and AG's News dataset (Zhang et al., 2015), as well as other comparisons to baselines are included in Appendix C. For reproducibility, we include the full code in the Supplementary Material. While the main experiments use the synchronous version to enable direct comparison with fixed-graph baselines, additional evaluations of the asynchronous Algorithm 1 on MNIST and CIFAR-10 are provided in Appendix C.

For reproducibility, our code and experiments are publicly available in this repository.

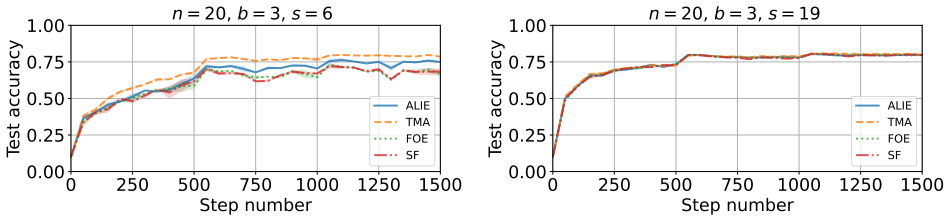


Figure 2: Test accuracies on CIFAR-10 ($\alpha = 1$)

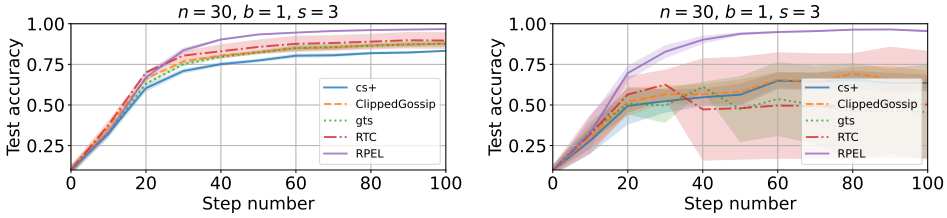


Figure 3: Comparison with fixed-graph methods on MNIST - Random attackers placement (5 seeds). (left) Average accuracy, (right) Worst accuracy

6.1 Experimental setup

Setting up s and $\hat{b}$

As stated before, Equation (3), gives a sufficient condition for $\hat{b}$ to be the maximum number of adversaries selected by each node throughout the whole training. However, this condition can be quite constraining in practice, requiring s to be very large, especially for small values of n.

In the experiments, we choose s and $\hat{b}$ by computing the hypergeometric tail probability of the maximum number of sampled Byzantine peers over all honest clients and rounds. We select the smallest s such that the resulting Effective adversarial fraction is below the threshold supported by the aggregation rule; details are in Algorithm 3.

Attack model and defense

To make the attacks as strong as possible, we allow the attackers to send distinct malicious updates to the honest clients. We also assume the attackers to be omniscient; every time a number b_i^t of adversarial nodes are selected by client i, they first observe the honest updates received by client i, and then craft their parameters according to the aggregation rule in use, employing one of the state-of-the-art attacks, namely sign flipping (SF) (Li et al., 2020), Fall of Empires (FOE) (Xie et al., 2020), A Little is Enough (ALIE) (Baruch et al., 2019), Trimmed Mean Attack (TMA) (Fang et al., 2020) and Dissensus (He et al., 2022) for fixed graphs. To defend against these attacks, we use an NNM pre-aggregation rule followed by a coordinate-wise trimmed mean (CWTM) (Yin et al., 2018).

Heterogeneity

We use Dirichlet Heterogeneity Hsu et al. (2019) to model the discrepancy between the clients' datasets. A parameter α controls the level of heterogeneity: larger values of α result in more homogeneous datasets among the clients. This model is well suited for capturing the non-IID effects in robust machine learning (Karimireddy et al., 2020; Allouah et al., 2023).

6.2 Empirical results

Robustness to high Effective adversarial fraction

Figures 1 and 2 show that RPEL remains accurate under ALIE, TMA, FOE, and sign-flipping attacks on MNIST and CIFAR-10, including regimes with effective adversarial fraction up to 0.43. Comparing $s = 6$ and all-to-all $s = 19$ on CIFAR-10 shows the expected accuracy gap is modest

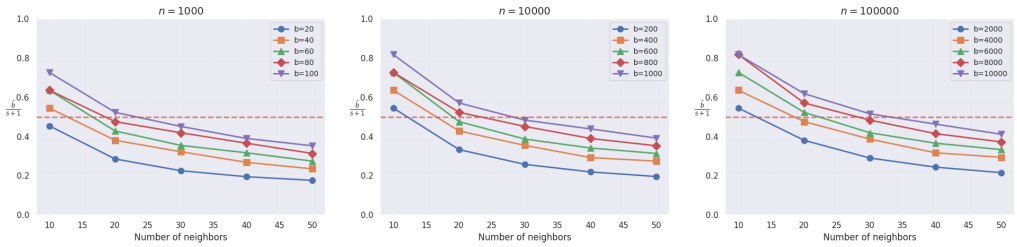


Figure 4: Effective adversarial fraction simulation for $p = 0.95$. For equal adversarial fraction, increasing n does not require increasing the number of selected neighbors by much.

relative to the communication reduction. On MNIST with $n = 200, b = 20$, sampling only $s = 20$ peers already reaches near-saturated accuracy, illustrating the practical benefit of randomized pull communication.

Competitive Performance with all-to-all robust algorithms

To compare RPEL with all-to-all robust Decentralized Learning methods, we tested $s = 19$ (Figure 2, Right) to allow each honest node to communicate with all the participating nodes at each iteration. As the Effective adversarial fraction in this case is much smaller (0.15 compared to the previous 0.43), it is natural for the performance in this case to be better. However, interestingly, the results show that reducing the number of neighbors to only $s = 6$ allows us to reach competitive accuracy for a much reduced communication cost. A similar conclusion can be drawn from the MNIST dataset Figure 1, since selecting $s = 20$ out of the 200 participating nodes is sufficient to maximize the test accuracy. Running all-to-all algorithms in this case would be extremely costly and impractical. These experimental findings show the practical benefits of randomization, which alleviates the requirement for all-to-all communication to achieve good accuracy. In Appendix C.3, we include experiments with higher heterogeneity, which show that RPEL's performance remains comparable to all-to-all methods even under these more challenging conditions.

Competitive performance with fixed-graph algorithms

For a fair comparison with other robust peer-to-peer methods with fixed graphs, we generate random connected graphs with the same number of communication edges as in our method. We give more details in Appendix C.2. We evaluate both the average and worst-case performance. Figure 3 compares the performance of RPEL to four fixed graph methods on MNIST dataset. Other results are presented in Figure 5 and Figure 6 in Appendix C, showing that for the same communication budget, RPEL is more robust, especially when the communication graph is sparse (low values of s). Thanks to the uniform sampling procedure of RPEL, the performance for the worst client is consistently better than the baselines. This suggests that our method also grants better fairness as a byproduct. In these experiments, the position of the attackers on the communication graph were either chosen at random or selected adversarially to augment the attackers impact or target vulnerable nodes.

It is essential to note, however, that fixed-graph methods are particularly suited for inherently sparse settings, where direct connections between all nodes are not feasible. In such cases, the graph topology is given as an input. In contrast, our setting assumes all-to-all communications are available.

Asynchronous evaluation

While the preceding experiments primarily evaluate the synchronous version of RPEL, we additionally evaluate Algorithm 1 on CIFAR-10 under IID random delays and persistent client-specific speed heterogeneity. In both settings, sampled Byzantine responses are retained first, representing a pessimistic response-ordering scenario. Across four seeds, RPEL achieves average accuracy of 73.15% and 71.96%, respectively, over the final three evaluations, compared with 74.77% under synchrony. These results provide additional empirical support for the effectiveness of the asynchronous protocol under heterogeneous response availability. Full experimental details and worst-client results are provided in Appendix C.3.

6.3 Scalability

Figure 4 and Figure 18 in Appendix C present an empirical simulation of the Effective adversarial fraction for different scenarios of n, b, and s using Algorithm 3.

The simulations in Figure 4 highlight the critical role of randomization in large-scale scenarios involving adversaries. As the number of nodes n grows, it is remarkable that the number of sampled neighbors s does not need to increase proportionally to maintain robustness against a fixed proportion of attackers. This scalability is illustrated in the rightmost plot in Figure 4, where even in a network of $n = 100\,000$ nodes with an adversarial fraction of 10% ($10\,000$ corrupted nodes), sampling only 40 random neighbors per node is sufficient to maintain an honest majority throughout the entire training process ($T = 200$) for all the $90\,000$ honest clients. This is far more efficient than having to ensure the neighborhood of each honest node contains at least $20\,001$ nodes.

To complement our theoretical communication analysis, we evaluate RPEL on a simple synthetic classification task with up to $10\,000$ clients, including 10% Byzantine participants. The detailed setting and results are provided in Appendix C. Table 7 shows that RPEL with NNM+CWTM maintains competitive accuracy as the population grows, achieving 73.2%, 74.5%, 71.9%, and 71.5% for $1\,000$, $2\,000$, $5\,000$, and $10\,000$ clients, respectively. At $10\,000$ clients, RPEL achieves 71.5% accuracy, compared with 71.3% for robust federated learning with CWTM and 73.5% for an attack-free oracle. These results provide additional empirical evidence that sparse randomized communication can support robust learning in large client populations.

This underscores the power of randomization in mitigating adversarial threats while preserving communication efficiency in decentralized learning. However, conducting realistic machine learning experiments at this scale with tens of thousands of nodes remains computationally prohibitive and is therefore beyond the scope of this paper.

7 Conclusion

This paper presents Robust Pull-based Epidemic Learning (RPEL), an efficient solution to the long-standing problem of learning despite an adversary controlling several nodes in a serverless setting. We establish rigorous convergence guarantees of RPEL in asynchrony, considering a general non-convex setting, in the presence of data heterogeneity and under standard assumptions. Our analysis proves robustness against powerful, omniscient attacks. A key concept underlying our approach is the notion of *Effective adversarial fraction*, a probabilistic upper bound on the selected attackers, providing a nuanced understanding of the system's ability to tolerate adversaries. We empirically evaluate RPEL on MNIST and CIFAR-10 datasets, and we show that RPEL achieves comparable accuracy with all-to-all robust methods for a much cheaper communication budget. Possible future directions include devising more sophisticated sampling techniques and deploying and testing RPEL on large-scale concrete applications.

Acknowledgments and Disclosure of Funding

This work was supported in part by the Swiss National Science Foundation under Grant N°20CH21_218778, TruBrain and by SNSF grant 200021_200477. The authors are thankful to the anonymous reviewers for their constructive comments.

References

Allen-Zhu, Z., Ebrahimian, F., Li, J. Z., and Alistarh, D. (2020). Byzantine-resilient non-convex stochastic gradient descent. *ArXiv*, abs/2012.14368.

Allouah, Y., Farhadkhani, S., Guerraoui, R., Gupta, N., Pinot, R., Rizk, G., and Voitovych, S. (2024a). Byzantine-robust federated learning: Impact of client subsampling and local updates. In *Forty-first International Conference on Machine Learning*.

Allouah, Y., Farhadkhani, S., Guerraoui, R., Gupta, N., Pinot, R., and Stephan, J. (2023). Fixing by mixing: A recipe for optimal byzantine ml under heterogeneity. In *International Conference on Artificial Intelligence and Statistics*, pages 1232–1300. PMLR.

Allouah, Y., Mrini, A. E., Guerraoui, R., Gupta, N., and Pinot, R. (2024b). Fine-tuning personalization in federated learning to mitigate adversarial clients.

Baruch, G., Baruch, M., and Goldberg, Y. (2019). A little is enough: Circumventing defenses for distributed learning. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Berthier, R., Bach, F., and Gaillard, P. (2020). Accelerated gossip in networks of given dimension using jacobi polynomial iterations. *SIAM Journal on Mathematics of Data Science*, 2(1):24–47.

Blanchard, P., El Mhamdi, E. M., Guerraoui, R., and Stainer, J. (2017). Machine learning with adversaries: Byzantine tolerant gradient descent. *Advances in neural information processing systems*, 30.

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

Boyd, S. P., Ghosh, A., Prabhakar, B., and Shah, D. (2005). Mixing times for random walks on geometric random graphs. In *ALENEX/ANALCO*, pages 240–249.

Brandes, U. (2001). A faster algorithm for betweenness centrality*. *The Journal of Mathematical Sociology*, 25(2):163–177.

Caldas, S., Duddu, S. M. K., Wu, P., Li, T., Konečný, J., McMahan, H. B., Smith, V., and Talwalkar, A. (2019). Leaf: A benchmark for federated settings.

Chen, Y., Su, L., and Xu, J. (2017). Distributed statistical machine learning in adversarial settings: Byzantine gradient descent. *Proc. ACM Meas. Anal. Comput. Syst.*, 1(2).

Data, D. and Diggavi, S. (2021). Byzantine-resilient sgd in high dimensions on heterogeneous data. In *2021 IEEE International Symposium on Information Theory (ISIT)*, pages 2310–2315.

De Vos, M., Farhadkhani, S., Guerraoui, R., Kermarrec, A.-M., Pires, R., and Sharma, R. (2024). Epidemic learning: Boosting decentralized learning with randomized communication. *Advances in Neural Information Processing Systems*, 36.

El Mhamdi, E. M., Farhadkhani, S., Guerraoui, R., Guirguis, A., Hoang, L.-N., and Rouault, S. (2021a). Collaborative learning in the jungle (decentralized, byzantine, heterogeneous, asynchronous and nonconvex learning). *Advances in Neural Information Processing Systems*, 34:25044–25057.

El Mhamdi, E. M., Guerraoui, R., and Rouault, S. L. A. (2021b). Distributed momentum for byzantine-resilient stochastic gradient descent. In *9th International Conference on Learning Representations (ICLR)*.

Fang, M., Cao, X., Jia, J., and Gong, N. (2020). Local model poisoning attacks to Byzantine-Robust federated learning. In *29th USENIX Security Symposium (USENIX Security 20)*, pages 1605–1622. USENIX Association.

Farhadkhani, S., Guerraoui, R., Gupta, N., Hoang, L.-N., Pinot, R., and Stephan, J. (2023). Robust collaborative learning with linear gradient overhead. In *International Conference on Machine Learning*, pages 9761–9813. PMLR.

Gaucher, R., Dieuleveut, A., and Hendrikx, H. (2025). Unified breakdown analysis for byzantine robust gossip. In *Forty-second International Conference on Machine Learning*.

Gaucher, R., Hendrikx, H., and Dieuleveut, A. (2024). Byzantine-robust gossip: Insights from a dual approach. *arXiv preprint arXiv:2405.03449*.

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

Gorbunov, E., Borzunov, A., Diskin, M., and Ryabinin, M. (2022). Secure distributed training at scale. In *International Conference on Machine Learning*, pages 7679–7739. PMLR.

He, L., Karimireddy, S. P., and Jaggi, M. (2022). Byzantine-robust decentralized learning via clippedgossip. *arXiv preprint arXiv:2202.01545*.

Hsu, H., Qi, H., and Brown, M. (2019). Measuring the effects of non-identical data distribution for federated visual classification.

Kairouz, P., McMahan, H. B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A. N., Bonawitz, K., Charles, Z., Cormode, G., Cummings, R., et al. (2021). Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning*, 14(1–2):1–210.

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

Koloskova, A., Stich, S., and Jaggi, M. (2019). Decentralized stochastic optimization and gossip algorithms with compressed communication. In *International Conference on Machine Learning*, pages 3478–3487. PMLR.

Krizhevsky, A., Nair, V., and Hinton, G. (2014). The cifar-10 dataset.

Lamport, L., Shostak, R., and Pease, M. (2019). The byzantine generals problem. In *Concurrency: the works of leslie lamport*, pages 203–226.

LeCun, Y., Cortes, C., and Burges, C. (2010). Mnist handwritten digit database. *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, 2.

Li, S., Cheng, Y., Wang, W., Liu, Y., and Chen, T. (2020). Learning to detect malicious clients for robust federated learning.

Lian, X., Zhang, C., Zhang, H., Hsieh, C.-J., Zhang, W., and Liu, J. (2017). Can decentralized algorithms outperform centralized algorithms? a case study for decentralized parallel stochastic gradient descent. *Advances in neural information processing systems*, 30.

Lian, X., Zhang, W., Zhang, C., and Liu, J. (2018). Asynchronous decentralized parallel stochastic gradient descent. In *International Conference on Machine Learning*, pages 3043–3052. PMLR.

Lu, S. and Wu, C. W. (2020). Decentralized stochastic non-convex optimization over weakly connected time-varying digraphs. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5770–5774. IEEE.

McMahan, B., Moore, E., Ramage, D., Hampson, S., and y Arcas, B. A. (2017). Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR.

Polyak, B. (1964). Some methods of speeding up the convergence of iteration methods. *Ussr Computational Mathematics and Mathematical Physics*, 4:1–17.

Vogels, T., Hendrikx, H., and Jaggi, M. (2023). Beyond spectral gap (extended): The role of the topology in decentralized learning. *arXiv preprint arXiv:2301.02151*.

Xie, C., Koyejo, O., and Gupta, I. (2020). Fall of empires: Breaking byzantine-tolerant sgd by inner product manipulation. In Adams, R. P. and Gogate, V., editors, *Proceedings of The 35th Uncertainty in Artificial Intelligence Conference*, volume 115 of *Proceedings of Machine Learning Research*, pages 261–270. PMLR.

Yang, C. and Ghaderi, J. (2024). Byzantine-robust decentralized learning via remove-then-clip aggregation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 21735–21743.

Yin, D., Chen, Y., Kannan, R., and Bartlett, P. (2018). Byzantine-robust distributed learning: Towards optimal statistical rates. In *International conference on machine learning*, pages 5650–5659. Pmlr.

Zhang, X., Zhao, J., and LeCun, Y. (2015). Character-level convolutional networks for text classification. In *Proceedings of the 29th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'15, page 649–657, Cambridge, MA, USA. MIT Press.

Organization of the appendix. Appendix A contains the theoretical material omitted from the main text, including notation, auxiliary lemmas, and the proofs of the main convergence results. In particular, Appendix A.5 provides the detailed proof and discussion of the convergence guarantee for the asynchronous version of RPEL; we also include a dedicated limitations paragraph in Section C.4. Appendix B describes the practical procedure used to choose the sampling parameter s and the effective number of adversaries $\hat{b}$. Appendix C provides additional experimental details and results.

A Proofs

A.1 Notations

This subsection presents the different notations we will use throughout the proofs, some of which have already been introduced in the main text.

Recall that in our algorithm, at each iteration t, each honest node i samples a set S_i^t among the n nodes present in the graph. For the sake of simplicity of the proof, we consider $|S_i^t| = s + 1$ is sampled from the entire population, without enforcing the inclusion of the local node.

Among all the participating nodes, at most b can be adversarial. As stated in the paper, we assume with high probability that at most $\hat{b}$ among the selected nodes are adversarial. We note $\hat{h} = s + 1 - \hat{b}$. $\hat{h}$ is a lower bound on the number of honest clients guaranteed to be sampled at each iteration.

The set of honest clients is denoted $\mathcal{H}$. At each iteration t, for each $i \in \mathcal{H}$, $\hat{\mathcal{H}}_i^t$ is used to denote a random sample of $\hat{h}$ honest clients among the sampled clients S_i^t.

It is important to notice that taking a random sample S_i^t uniformly out of all the participating nodes, and then taking a random number of $\hat{h}$ honest nodes uniformly among the selected nodes S_i^t is equivalent to choosing a random subset of size $\hat{h}$ uniformly out of all the honest participating nodes.

We note $f_i(x) = \mathbb{E}_{\xi \sim \mathcal{D}_i}[\ell(x, \xi)]$ and $F_{\mathcal{H}}(x) = \frac{1}{|\mathcal{H}|} \sum_{i \in \mathcal{H}} f_i(x)$.

At each iteration t, each node i computes one stochastic gradient $g_i^t = \nabla \ell(\mathbf{x}_i^t, \xi_i^t)$, the local momentum $m_i^t = \beta m_i^{t-1} + (1 - \beta) g_i^t$ and computes the update

$$\mathbf{x}_i^{t+1/2} = \mathbf{x}_i^t - \eta m_i^t \tag{7}$$

Additionally, for $i, j \in \mathcal{H}$, we use the notation $\mathcal{I}(j \in \hat{\mathcal{H}}_i^t)$ for the indicator variable of the event $\{j \in \hat{\mathcal{H}}_i^t\}$.

Finally, we use the notation $\mathbb{E}_t$ to denote the conditional expectation on all the past events up until iteration t, including the random stochastic gradients computed at iteration t. In other words, $\mathbb{E}_t$ is the expectation with respect to the random sampling at iteration t.

A.2 Useful proofs

In this section, we present a few results which will be useful for the rest of the proofs. For the following results, we are given m points $y_1, \ldots, y_m$ in $\mathbb{R}^d$, if S is a subset of $\{1, \ldots, m\}$, we use the following notation:

$$\overline{y}_S = \frac{1}{|S|}\sum_{i\in S} y_i$$
$$\overline{y} = \frac{1}{m}\sum_{i=1}^{m} y_i$$
$$\mathrm{Var}_S(y) = \frac{1}{|S|}\sum_{i\in S} \|y_i - \overline{y}_S\|^2$$
$$\mathrm{Var}(y) = \frac{1}{m}\sum_{i=1}^{m} \|y_i - \overline{y}\|^2.$$

Lemma A.1. *Let S be a subset of $\{1,\dots,m\}$ of size k taken uniformly at random (without replacement), then*

$$\mathbb{E}\left[\overline{y}_S\right] = \overline{y}.$$

Proof. We have

$$\begin{aligned}\overline{y}_S &= \frac{1}{k}\sum_{i\in S} y_i \\ &= \frac{1}{k}\sum_{i=1}^{m} \mathcal{I}(i\in S) y_i.\end{aligned}$$

We conclude using

$$\mathbb{E}\left[\mathcal{I}(i\in S)\right] = \frac{k}{m}.$$

□

Lemma A.2. *Let S be a subset of $\{1,\dots,m\}$ of size k taken uniformly at random (without replacement), for $j,l\in\{1,\dots,m\}$, such that $j\neq l$ we can compute*

$$\mathrm{Var}(\mathcal{I}(j\in S)) = \frac{k}{m}\left(1-\frac{k}{m}\right),$$

and

$$\mathrm{Cov}\left(\mathcal{I}(j\in S),\mathcal{I}(l\in S)\right) = \frac{k(k-1)}{m(m-1)} - \frac{k^2}{m^2}.$$

Proof. The first identity is the variance of $\mathcal{I}(j\in S)$ which is a Bernoulli variable with the parameter $\frac{k}{m}$ for any $j\in S$.

For the second identity, if $j\neq l$,

$$\begin{aligned}\mathrm{Cov}\left(\mathcal{I}(j\in S),\mathcal{I}(l\in S)\right) &= \mathbb{E}\left[\mathcal{I}(j\in S)\mathcal{I}(l\in S)\right] - \mathbb{E}\left[\mathcal{I}(j\in S)\right]\mathbb{E}\left[\mathcal{I}(l\in S)\right] \\ &= \mathbb{P}((j,l)\in S) - \frac{k^2}{m^2} \\ &= \frac{k(k-1)}{m(m-1)} - \frac{k^2}{m^2},\end{aligned}$$

which concludes the lemma. □

Lemma A.3. *Let S be any (arbitrary) subset of $\{1,\dots,m\}$ of size k, then*

$$\|\overline{y}_S - \overline{y}\|^2 \leq \frac{m-k}{k}\mathrm{Var}(y). \tag{8}$$

Proof. We start by the case when $\overline{y} = 0$. Let S^c be the set complementary to S in $\{1, \ldots, m\}$. First, we have:

$$k\overline{y}_S + (m-k)\overline{y}_{S^c} = 0.$$

So:

$$\overline{y}_{S^c} = -\frac{k}{m-k}\overline{y}_S.$$

Now we can write:

$$\begin{aligned}
m\mathrm{Var}(y) &= \sum_{i=1}^{m} \|y_i\|^2 \\
&= \sum_{i\in S} \|y_i\|^2 + \sum_{i\in S^c} \|y_i\|^2 \\
&= \sum_{i\in S} \|y_i - \overline{y}_S\|^2 + k\|\overline{y}_S\|^2 + \sum_{i\in S^c} \|y_i - \overline{y}_{S^c}\|^2 + (m-k)\|\overline{y}_{S^c}\|^2 \\
&= \sum_{i\in S} \|y_i - \overline{y}_S\|^2 + \sum_{i\in S^c} \|y_i - \overline{y}_{S^c}\|^2 + \left(k + (m-k)\frac{k^2}{(m-k)^2}\right)\|\overline{y}_S\|^2 \\
&= \sum_{i\in S} \|y_i - \overline{y}_S\|^2 + \sum_{i\in S^c} \|y_i - \overline{y}_{S^c}\|^2 + \frac{mk}{(m-k)}\|\overline{y}_S\|^2.
\end{aligned}$$

It follows that:

$$\|\overline{y}_S\|^2 \leq \frac{m-k}{k}\mathrm{Var}(y).$$

In the general case, for each i, let $z_i = y_i - \overline{y}$, then

$$\|\overline{y}_S - \overline{y}\|^2 = \|\overline{z}_S\|^2 \leq \frac{m-k}{k}\mathrm{Var}(z) = \frac{m-k}{k}\mathrm{Var}(y).$$

□

Lemma A.4. *Let S be a subset of $\{1, \ldots, m\}$ of size k taken uniformly at random (without replacement), then*

$$\mathbb{E}\left[\|\overline{y}_S - \overline{y}\|^2\right] = \frac{m-k}{k(m-1)}\mathrm{Var}(y). \tag{9}$$

Proof. Here again, we start by assuming that $\overline{y} = 0$. The set S being uniformly sampled out of $\{1, \dots, m\}$, we can write :

$$\begin{aligned}
\mathbb{E}\left[\|\overline{y}_S\|^2\right] &= \mathbb{E}\left[\left\|\frac{1}{k}\sum_{i=1}^{m}\mathcal{I}(i \in S)y_i\right\|^2\right] \\
&= \frac{1}{k^2}\left(\sum_{i=1}^{m}\mathbb{E}\left[\mathcal{I}(i \in S)\right]\|y_i\|^2 + \sum_{i=1}^{m}\sum_{j\neq i}\mathbb{E}\left[\mathcal{I}(i \in S)\mathcal{I}(j \in S)\right]\langle y_i, y_j\rangle\right) \\
&= \frac{1}{k^2}\left(\sum_{i=1}^{m}\frac{k}{m}\|y_i\|^2 + \sum_{i=1}^{m}\sum_{j\neq i}\mathbb{E}\left[\mathcal{I}(i, j \in S)\right]\langle y_i, y_j\rangle\right) \\
&= \frac{1}{k^2}\left(\sum_{i=1}^{m}\frac{k}{m}\|y_i\|^2 + \sum_{i=1}^{m}\sum_{j\neq i}\frac{k(k-1)}{m(m-1)}\langle y_i, y_j\rangle\right).
\end{aligned}$$

Additionally, given that $\sum_{i=1}^{m} y_i = 0$, we have for each i:

$$\sum_{j\neq i}\langle y_i, y_j\rangle = -\|y_i\|^2.$$

Consequently,

$$\begin{aligned}
\mathbb{E}\left[\|\overline{y}_S\|^2\right] &= \frac{1}{k^2}\left(\sum_{i=1}^{m}\frac{k}{m}\|y_i\|^2 - \sum_{i=1}^{m}\frac{k(k-1)}{m(m-1)}\|y_i\|^2\right) \\
&= \frac{m-k}{k(m-1)}\frac{1}{m}\sum_{i=1}^{m}\|y_i\|^2 \\
&= \frac{m-k}{k(m-1)}\mathrm{Var}(y).
\end{aligned}$$

We conclude about the general case using the same trick used for Lemma A.3. □

Lemma A.5. *Let S be any (arbitrary) subset of $\{1, \dots, m\}$ of size k, then*

$$\mathrm{Var}_S(y) \leq \frac{m}{k}\mathrm{Var}(y). \tag{10}$$

Proof. We have:

$$\begin{aligned}
\mathrm{Var}_S(y) &= \frac{1}{k}\sum_{i\in S}\|y_i - \overline{y}_S\|^2 \\
&\leq \frac{1}{k}\sum_{i\in S}\|y_i - \overline{y}\|^2 \\
&\leq \frac{1}{k}\sum_{i=1}^{m}\|y_i - \overline{y}\|^2 \\
&= \frac{m}{k}\mathrm{Var}(y).
\end{aligned}$$

□

Lemma A.6. *Let S be a subset of $\{1, \dots, m\}$ of size k taken uniformly at random (without replacement), then*

$$\mathbb{E}\left[\mathrm{Var}_S(y)\right] = \frac{k-1}{k}\frac{m}{m-1}\mathrm{Var}(y). \tag{11}$$

Proof. We start from the following identity:

$$\frac{1}{k}\sum_{i\in S}\|y_i - \overline{y}\|^2 = \frac{1}{k}\sum_{i\in S}\|y_i - \overline{y}_S\|^2 + \|\overline{y}_S - \overline{y}\|^2.$$

It follows that:

$$\mathrm{Var}_S(y) = \frac{1}{k}\sum_{i\in S}\|y_i - \overline{y}\|^2 - \|\overline{y}_S - \overline{y}\|^2.$$

Taking the expectation:

$$\mathbb{E}[\mathrm{Var}_S(y)] = \mathrm{Var}(y) - \mathbb{E}[\|\overline{y}_S - \overline{y}\|^2].$$

Using Lemma A.4, we can write:

$$\begin{aligned}\mathbb{E}[\mathrm{Var}_S(y)] &= \mathrm{Var}(y) - \frac{m-k}{k(m-1)}\mathrm{Var}(y)\\ &= \frac{k-1}{k}\frac{m}{m-1}\mathrm{Var}(y)\end{aligned}$$

□

A.3 (α, λ)-reduction

We now present the so-called (α, λ)-reduction, introduced in Farhadkhani et al. (2023), which is a sufficient condition for convergence in the Byzantine peer-to-peer setting. For clarity, we restate it here.

Definition A.7 ((α, λ)-reduction)**.** An algorithm $\mathcal{F}$ is said to satisfy (α, λ)-reduction if for the inputs $(x_1, \dots, x_{|\mathcal{H}|})$, the outputs $(y_1, \dots, y_{|\mathcal{H}|}) = \mathcal{F}(x_1, \dots, x_{|\mathcal{H}|})$ satisfy:

$$\frac{1}{|\mathcal{H}|}\sum_{i=1}^{|\mathcal{H}|}\|y_i - \overline{y_{\mathcal{H}}}\|^2 \leq \alpha\frac{1}{|\mathcal{H}|}\sum_{i=1}^{|\mathcal{H}|}\|x_i - \overline{x_{\mathcal{H}}}\|^2,$$

$$\|\overline{y_{\mathcal{H}}} - \overline{x_{\mathcal{H}}}\|^2 \leq \lambda\frac{1}{|\mathcal{H}|}\sum_{i=1}^{|\mathcal{H}|}\|x_i - \overline{x_{\mathcal{H}}}\|^2,$$

where $\overline{x_{\mathcal{H}}} = \frac{1}{|\mathcal{H}|}\sum_{i\in\mathcal{H}} x_i$ and $\overline{y_{\mathcal{H}}} = \frac{1}{|\mathcal{H}|}\sum_{i\in\mathcal{H}} y_i$.

As proven in Farhadkhani et al. (2023), when this condition is satisfied by the iterates of any decentralized algorithm, it is sufficient to have $\alpha < 1$ and $\lambda < +\infty$ for the algorithm to satisfy (b, ε)-Byzantine resilience, for some value of ε.

Algorithm 2 RPEL algorithm (synchronous version)

Require: number of iterations T, number of neighbors s, momentum coefficient β, communication step size ρ, effective number of adversaries $\hat{b}$, Aggregation Rule $\mathcal{R}$

for $t = 1 \dots T$ **do**
  for $i \in \mathcal{H}$ in parallel **do**
    Randomly sample a data point ξ_i^t from $\mathcal{D}_i$
    Compute a local stochastic gradient $g_i^t = \nabla f_i(\mathbf{x}_i^t, \xi_i^t)$
    Update local momentum $m_i^t = \beta m_i^{t-1} + (1-\beta) g_i^t$
    Optimization step: $\mathbf{x}_i^{t+1/2} = \mathbf{x}_i^t - \eta m_i^t$
    Randomly sample a set S_i^t of s nodes
    Receive $\mathbf{x}_j^{t+1/2}$ from each $j \in S_i^t$ ▷ Selected attackers send corrupted updates
    Aggregation step: $\mathbf{x}_i^{t+1} = \mathcal{R}\left(\mathbf{x}_i^{t+1/2}; \left\{\mathbf{x}_j^{t+1/2}, j \in S_i^t\right\}\right)$
  end for
end for
for $i \in \mathcal{H}$ **do**
  Output $\hat{\mathbf{x}}_i \sim \mathcal{U}\{\mathbf{x}_i^1, \dots, \mathbf{x}_i^T\}$
end for

A.4 The Synchronous case

We start by providing the convergence analysis in the synchronous case, where we assume all the selected nodes to respond at iteration t. Algorithm 2 describes the simplified procedure in this case. At each iteration t, each honest node applies a robust aggregation $\mathcal{R}$ to the $s+1$ model updates it receives (including its own).

The following Lemma shows the α, λ reduction of Algorithm 2.

Lemma A.8. *Assuming $\mathcal{R}$ satisfies Definition 5.1, and assuming Equation* (3) *is satisfied, then the iterates of Algorithm 2 with $\mathcal{R}$ satisfy the following reduction at any iteration t:*

$$\mathbb{E}_t\left[\left\|\overline{\mathbf{x}_{\mathcal{H}}^{t+1}} - \overline{\mathbf{x}_{\mathcal{H}}^{t+1/2}}\right\|^2\right] \leq \left(2\kappa + 2\frac{|\mathcal{H}| - \hat{h}}{(|\mathcal{H}|-1)|\mathcal{H}|\hat{h}}\right) V^t,$$

$$\mathbb{E}_t\left[\frac{1}{|\mathcal{H}|}\sum_{i\in\mathcal{H}}\left\|\mathbf{x}_i^{t+1} - \overline{\mathbf{x}_{\mathcal{H}}^{t+1}}\right\|^2\right] \leq \left(2\kappa + 2\frac{|\mathcal{H}| - \hat{h}}{(|\mathcal{H}|-1)\hat{h}}\right) V^t.$$

where the expectation is taken on the random sampling at iteration t, and

$$V^t = \frac{1}{|\mathcal{H}|}\sum_{i\in\mathcal{H}}\left\|\mathbf{x}_i^{t+1/2} - \overline{\mathbf{x}_{\mathcal{H}}^{t+1/2}}\right\|^2. \tag{12}$$

Proof. • **First inequality**

For ease of notation, we omit the exponent $t + 1/2$, denoting $\mathbf{x}_i^{t+1/2}$ by simply $\mathbf{x}_i$, and we replace $\mathbf{x}_i^{t+1}$ by $\mathbf{x}_i^+$.

Using the definition of the variance, we write the first inequality as follows.

$$\mathbb{E}_t\left[\left\|\overline{\mathbf{x}_{\mathcal{H}}^+} - \overline{\mathbf{x}_{\mathcal{H}}}\right\|^2\right] \leq \left(\kappa + \frac{|\mathcal{H}| - \hat{h}}{(|\mathcal{H}|-1)|\mathcal{H}|\hat{h}}\right)\frac{1}{2|\mathcal{H}|^2}\sum_{i,j\in\mathcal{H}}\|\mathbf{x}_i - \mathbf{x}_j\|^2. \tag{13}$$

Using the aggregation formula of Algorithm 2, we have

$$\overline{\mathbf{x}_{\mathcal{H}}^{+}} - \overline{\mathbf{x}_{\mathcal{H}}} = \underbrace{\frac{1}{|\mathcal{H}|}\sum_{i\in\mathcal{H}}\left(\mathcal{R}\left(\mathbf{x}_j, j\in S_i^t\right) - \frac{1}{\hat{h}}\sum_{j\in\hat{\mathcal{H}}_i^t}\mathbf{x}_j\right)}_{T_1} + \underbrace{\frac{1}{|\mathcal{H}|\hat{h}}\sum_{i\in\mathcal{H}}\sum_{j\in\hat{\mathcal{H}}_i^t}\mathbf{x}_j - \frac{1}{|\mathcal{H}|}\sum_{i\in\mathcal{H}}\mathbf{x}_j}_{T_2}.$$

Taking the norm and the expectation, and using the inequality $\|a+b\|^2 \leq 2(\|a\|^2 + \|b\|^2)$, we have:

$$\mathbb{E}_t\left[\|\overline{\mathbf{x}_{\mathcal{H}}^{+}} - \overline{\mathbf{x}_{\mathcal{H}}}\|^2\right] \leq 2\left(\mathbb{E}_t[\|T_1\|^2] + \mathbb{E}_t[\|T_2\|^2]\right). \tag{14}$$

For T_1, by the AM-QM inequality:

$$\mathbb{E}_t[\|T_1\|^2] \leq \frac{1}{|\mathcal{H}|}\sum_{i\in\mathcal{H}}\mathbb{E}_t\left\|\mathcal{R}\left(\mathbf{x}_j, j\in S_i^t\right) - \frac{1}{\hat{h}}\sum_{j\in\hat{\mathcal{H}}_i^t}\mathbf{x}_j\right\|^2. \tag{15}$$

Since $\mathcal{R}$ satisfies $(s,\hat{b},\kappa)$-robustness, we have for each $i\in\mathcal{H}$

$$\begin{aligned}\left\|\mathcal{R}\left(\mathbf{x}_j, j\in S_i^t\right) - \frac{1}{\hat{h}}\sum_{j\in\hat{\mathcal{H}}_i^t}\mathbf{x}_j\right\|^2 &\leq \frac{\kappa}{\hat{h}}\sum_{j\in\hat{\mathcal{H}}_i^t}\left\|\mathbf{x}_j - \overline{\mathbf{x}_{\hat{\mathcal{H}}_i^t}}\right\|^2 \\ &= \frac{\kappa}{2\hat{h}^2}\sum_{j,k\in\hat{\mathcal{H}}_i^t}\|\mathbf{x}_j - \mathbf{x}_k\|^2.\end{aligned}$$

Taking the expectation on the random sampling of each honest node, and using Lemma A.1, we have:

$$\mathbb{E}_t\left[\left\|\mathcal{R}\left(\mathbf{x}_j, j\in S_i^t\right) - \frac{1}{\hat{h}}\sum_{j\in\hat{\mathcal{H}}_i^t}\mathbf{x}_j\right\|^2\right] \leq \frac{\kappa}{2|\mathcal{H}|^2}\sum_{j,k\in\mathcal{H}}\|\mathbf{x}_j - \mathbf{x}_k\|^2. \tag{16}$$

Plugging Equation (16) in Equation (15), we get

$$\mathbb{E}_t[\|T_1\|^2] \leq \frac{\kappa}{2|\mathcal{H}|^2}\sum_{i,j\in\mathcal{H}}\|\mathbf{x}_i - \mathbf{x}_j\|^2. \tag{17}$$

For T_2, noticing that $\mathbb{E}_t[T_2] = 0$, thanks to Lemma A.1, and using the independence of $\hat{\mathcal{H}}_i^t$'s, which is a key feature of pull-based Epidemic Learning, we can write :

$$\begin{aligned}
\mathbb{E}_t[\|T_2\|^2] &= \mathrm{Var}_t\left(\frac{1}{|\mathcal{H}|\hat{h}}\sum_{i\in\mathcal{H}}\sum_{j\in\hat{\mathcal{H}}_i^t}\mathbf{x}_j\right)\\
&= \frac{1}{|\mathcal{H}|}\mathrm{Var}_t\left(\frac{1}{\hat{h}}\sum_{j\in\hat{\mathcal{H}}_i^t}\mathbf{x}_j\right)\\
&= \frac{1}{|\mathcal{H}|\hat{h}^2}\mathrm{Var}_t\left(\sum_{j\in\mathcal{H}}\mathcal{I}(j\in\hat{\mathcal{H}}_i^t)\mathbf{x}_j\right)\\
&= \frac{1}{|\mathcal{H}|\hat{h}^2}\left(\sum_{j\in\mathcal{H}}\mathrm{Var}_t(\mathcal{I}(j\in\hat{\mathcal{H}}_i^t))\|\mathbf{x}_j\|^2 + \sum_{j\neq k}\mathrm{Cov}_t\left(\mathcal{I}(j\in\hat{\mathcal{H}}_i^t),\mathcal{I}(k\in\hat{\mathcal{H}}_i^t)\right)\mathbf{x}_j^T\mathbf{x}_k\right).
\end{aligned}$$

Now using Lemma A.2, it follows that:

$$\begin{aligned}
\mathbb{E}_t[\|T_2\|^2] &= \frac{1}{|\mathcal{H}|\hat{h}^2}\left(\frac{\hat{h}}{|\mathcal{H}|}\left(1-\frac{\hat{h}}{|\mathcal{H}|}\right)\sum_{j\in\mathcal{H}}\|\mathbf{x}_j\|^2 + \left(\frac{\hat{h}(\hat{h}-1)}{|\mathcal{H}|(|\mathcal{H}|-1)} - \frac{\hat{h}^2}{|\mathcal{H}|^2}\right)\sum_{j\neq k}\mathbf{x}_j^T\mathbf{x}_k\right)\\
&= \frac{1}{|\mathcal{H}|^2\hat{h}}\left(\frac{|\mathcal{H}|-\hat{h}}{|\mathcal{H}|}\sum_j\|\mathbf{x}_j\|^2 + \left(\frac{\hat{h}-1}{|\mathcal{H}|-1} - \frac{\hat{h}}{|\mathcal{H}|}\right)\sum_{j\neq k}\mathbf{x}_j^T\mathbf{x}_k\right)\\
&= \frac{|\mathcal{H}|-\hat{h}}{|\mathcal{H}|^2\hat{h}}\left(\frac{1}{|\mathcal{H}|}\sum_j\|\mathbf{x}_j\|^2 - \frac{1}{|\mathcal{H}|(|\mathcal{H}|-1)}\sum_{j\neq k}\mathbf{x}_j^T\mathbf{x}_k\right)\\
&= \frac{|\mathcal{H}|-\hat{h}}{(|\mathcal{H}|-1)|\mathcal{H}|\hat{h}}\left(\frac{1}{|\mathcal{H}|}\sum_j\|\mathbf{x}_j\|^2 - \left\|\frac{1}{|\mathcal{H}|}\sum_j\mathbf{x}_j\right\|^2\right)\\
&= \frac{|\mathcal{H}|-\hat{h}}{(|\mathcal{H}|-1)|\mathcal{H}|\hat{h}}\left(\frac{1}{2|\mathcal{H}|^2}\sum_{i,j\in\mathcal{H}}\|\mathbf{x}_i-\mathbf{x}_j\|^2\right).
\end{aligned} \tag{18}$$

Plugging Equation (17) and Equation (18) back into Equation (14), this completes the first part of the proof.

- **Second inequality.**

  First, given that $\overline{\mathbf{x}_{\mathcal{H}}^+} = \arg\min_\mu \frac{1}{|\mathcal{H}|}\sum_{i\in\mathcal{H}}\|\mathbf{x}_i^+ - \mu\|^2$, we have:

$$\mathbb{E}_t\left[\frac{1}{|\mathcal{H}|}\sum_{i\in\mathcal{H}}\left\|\mathbf{x}_i^+ - \overline{\mathbf{x}_{\mathcal{H}}^+}\right\|^2\right] \leq \mathbb{E}_t\left[\frac{1}{|\mathcal{H}|}\sum_{i\in\mathcal{H}}\left\|\mathbf{x}_i^+ - \overline{\mathbf{x}_{\mathcal{H}}}\right\|^2\right]. \tag{19}$$

  Now for each i, we have :

$$\begin{aligned}
\left\|\mathbf{x}_i^+ - \overline{\mathbf{x}_{\mathcal{H}}}\right\|^2 &= \left\|\mathcal{R}\left(\mathbf{x}_j, j\in S_i^t\right) - \overline{\mathbf{x}_{\mathcal{H}}}\right\|^2\\
&\leq 2\left\|\mathcal{R}\left(\mathbf{x}_j, j\in S_i^t\right) - \overline{\mathbf{x}_{\hat{\mathcal{H}}_i^t}}\right\|^2 + 2\left\|\overline{\mathbf{x}_{\hat{\mathcal{H}}_i^t}} - \overline{\mathbf{x}_{\mathcal{H}}}\right\|^2.
\end{aligned}$$

  For the first term, taking the expectation, and using Equation (16), we have:

$$\mathbb{E}_t\left[\left\|\mathcal{R}\left(\mathbf{x}_j, j \in S_i^t\right) - \overline{\mathbf{x}_{\hat{\mathcal{H}}_i^t}}\right\|^2\right] \leq \kappa \mathrm{Var}_{\mathcal{H}}(\mathbf{x}).$$

For the second term, taking the expectation and using Lemma A.4, we have:

$$\mathbb{E}_t\left[\left\|\overline{\mathbf{x}_{\hat{\mathcal{H}}_i^t}} - \overline{\mathbf{x}_{\mathcal{H}}}\right\|^2\right] \leq \frac{|\mathcal{H}| - \hat{h}}{(|\mathcal{H}| - 1)\hat{h}} \mathrm{Var}_{\mathcal{H}}(\mathbf{x}).$$

By taking the average over all the honest nodes, we get:

$$\mathbb{E}_t\left[\frac{1}{|\mathcal{H}|}\sum_{i\in\mathcal{H}}\left\|\mathbf{x}_i^+ - \overline{\mathbf{x}_{\mathcal{H}}}\right\|^2\right] \leq \left(2\kappa + 2\frac{|\mathcal{H}| - \hat{h}}{(|\mathcal{H}| - 1)\hat{h}}\right)\mathrm{Var}_{\mathcal{H}}(\mathbf{x})$$

Now using Equation (19), we conclude the proof.

□

A.5 The Asynchronous case

A.5.1 Motivation

We previously stated that the Push variant of Epidemic learning fails to Byzantine flooding attacks, since the honest nodes cannot control who sends them the updates. The Pull variant reduces the attack surface and gives back control to the honest nodes, which explains the scalability benefits of RPEL.

However, the Pull variant can be vulnerable to Denial of Service attacks, where the Byzantine nodes may intentionally delay sending their models to slow down the learning, and may even decline to send their updates to stall the honest nodes.

In the synchronous model, bypassing this issue is straightforward. Indeed, synchrony implicitly assumes that honest nodes will send their updates in a timely manner, and only Byzantine clients are allowed to refuse sending their updates. But if they do refuse, this simplifies the task for the honest nodes since the identity of the adversaries is revealed.

In this section, we will consider the asynchronous case where it is impossible for an honest node to distinguish between a Byzantine node that refuses to send their update, and a slow honest node. The pseudo-code of the procedure in the asynchronous case is provided in Algorithm 1.

A.5.2 Notations and Proofs foundation

As described in Algorithm 1, $\hat{S}_i^t$ is the set of nodes that will be used for aggregation by node i at iteration t. We denote by G_i^t the set of honest nodes among them : $G_i^t := \hat{S}_i^t \cap \mathcal{H}$, and we denote by $\hat{G}_i^t$ a random set of size $s + 1 - 2\hat{b}$ from G_i^t. Similar to the synchronous case, we note $\hat{\mathcal{H}}_i^t$ a random set of $\hat{h} = s + 1 - \hat{b}$ honest nodes among all the initially selected nodes S_i^t. For technical purposes, we require $\hat{G}_i^t \subset \hat{\mathcal{H}}_i^t$, meaning that $\hat{G}_i^t$ and $\hat{\mathcal{H}}_i^t$ share the same randomness conditional on knowledge of S_i^t. This technical complexity appears only when the number of honest nodes in S_i^t is strictly larger that its lower bound $s + 1 - \hat{b}$, which happens in the favorable case when the honest node i receives less adversaries than expected $|S_i^t \cap \mathcal{B}| < \hat{b}$.

To see why it is possible to ensure $\hat{G}_i^t \subset \hat{\mathcal{H}}_i^t$, we can use the following construction: S_i^t contains at least $s + 1 - \hat{b}$ honest nodes, let $\hat{H}_i^t$ be a random set of size $s + 1 - \hat{b}$ among these honest nodes, $\hat{G}_i^t$ is a random set of size $s + 1 - 2\hat{b}$ taken uniformly at random from $G_i^t \cap \hat{H}_i^t$. For this construction to be valid, we only need to show that $|G_i^t \cap \hat{H}_i^t| \geq s + 1 - 2\hat{b}$.

To do so, let $n_{\mathcal{H}}^t = |S_i^t \cap \mathcal{H}|$ be the number of sampled honest nodes. We have $|G_i^t| \leq n_{\mathcal{H}}^t - \hat{b}$, and

$$
\begin{aligned}
|G_i^t \cap \hat{H}_i^t| &= |G_i^t| + |\hat{H}_i^t| - |G_i^t \cup \hat{H}_i^t| \\
&\geq n_{\mathcal{H}}^t - \hat{b} + s + 1 - \hat{b} - n_{\mathcal{H}}^t \\
&\geq s + 1 - 2\hat{b}.
\end{aligned}
$$

Notice that contrary to $\hat{\mathcal{H}}_i^t$, $\hat{G}_i^t$ need not be sampled uniformly at random from $\mathcal{H}$. This is a key technical issue that will result in additional terms to the error. This can be interpreted as follows : even though the initial selection is uniformly random, the updates effectively used can be biased (e.g. against the slowest nodes).

We note $\hat{g} = s + 1 - 2\hat{b}$ the cardinal of the set $\hat{G}_i^t$.

Contrary to the synchronous setting, we require here to have $\hat{b}/s+1 < 1/3$, since we need to account for both the case when the Byzantine nodes do no send anything and the case when all their $\hat{b}$ updates are in $\hat{S}_i^t$.

A.5.3 Asynchronous case - analysis

In order to prove Lemma 5.2, we prove the following Lemma:

Lemma A.9. *In the asynchronous case, assuming $\mathcal{R}$ satisfies Definition 5.1, and assuming Equation* (3) *is satisfied, then the iterates of Algorithm 1 with $\mathcal{R}$ satisfy the following reduction at any iteration t:*

$$
\mathbb{E}_t \left[\left\| \overline{\mathbf{x}_{\mathcal{H}}^{t+1}} - \overline{\mathbf{x}_{\mathcal{H}}} \right\|^2 \right] \leq \left(3\gamma_1\kappa + 3\gamma_2 + 3\frac{|\mathcal{H}| - \hat{h}}{\hat{h}(|\mathcal{H}| - 1)|\mathcal{H}|} \right) \mathrm{Var}_{\mathcal{H}}(\mathbf{x}), \tag{20}
$$

$$
\mathbb{E}_t \left[\frac{1}{|\mathcal{H}|} \sum_{i \in \mathcal{H}} \left\| \mathbf{x}_i^{t+1} - \overline{\mathbf{x}_{\mathcal{H}}^{t+1}} \right\|^2 \right] \leq \left(3\gamma_1\kappa + 3\gamma_2 + 3\frac{|\mathcal{H}| - \hat{h}}{\hat{h}(|\mathcal{H}| - 1)} \right) \mathrm{Var}_{\mathcal{H}}(\mathbf{x}), \tag{21}
$$

with $\gamma_1 = \frac{s+1-\hat{b}}{s+1-2\hat{b}}$ *and* $\gamma_2 = \frac{\hat{b}}{s+1-2\hat{b}}$.

Notice that $1 \leq \gamma_1 < 2$, and it increases with the Effective Byzantine fraction. γ_2 is of the same order as κ; indeed Allouah et al. (2023) show that $\frac{\hat{b}}{s+1-2\hat{b}}$ is the lower bound on κ such that an aggregation rule can satisfy Definition 5.1.

Proof. • **First inequality.**

Here again, we omit the exponent $t + 1/2$, denoting $\mathbf{x}_i^{t+1/2}$ by simply $\mathbf{x}_i$, and we replace $\mathbf{x}_i^{t+1}$ by $\mathbf{x}_i^+$.

We can write the difference $\overline{\mathbf{x}_{\mathcal{H}}^+} - \overline{\mathbf{x}_{\mathcal{H}}}$ as follows:

$$
\begin{aligned}
\overline{\mathbf{x}_{\mathcal{H}}^+} - \overline{\mathbf{x}_{\mathcal{H}}} = &\underbrace{\frac{1}{|\mathcal{H}|} \sum_{i \in \mathcal{H}} \left(\mathcal{R}\left(\mathbf{x}_j, j \in \hat{S}_i^t \right) - \frac{1}{\hat{g}} \sum_{j \in \hat{G}_i^t} \mathbf{x}_j \right)}_{T_1} \\
&+ \underbrace{\frac{1}{|\mathcal{H}|\hat{g}} \sum_{i \in \mathcal{H}} \sum_{j \in \hat{G}_i^t} \mathbf{x}_j - \frac{1}{|\mathcal{H}|\hat{h}} \sum_{i \in \mathcal{H}} \sum_{j \in \hat{\mathcal{H}}_i^t} \mathbf{x}_j}_{T_2} \\
&+ \underbrace{\frac{1}{|\mathcal{H}|\hat{h}} \sum_{i \in \mathcal{H}} \sum_{j \in \hat{\mathcal{H}}_i^t} \mathbf{x}_j - \frac{1}{|\mathcal{H}|} \sum_{i \in \mathcal{H}} \mathbf{x}_j}_{T_3}.
\end{aligned}
$$

For T_1, using AM-QM inequality, since $\mathcal{R}$ satisfies $(s, \hat{b}, \kappa)$-robustness, we have:

$$\begin{aligned}\|T_1\|^2 &\leq \frac{1}{|\mathcal{H}|}\sum_{i\in\mathcal{H}}\left\|\mathcal{R}\left(\mathbf{x}_j, j\in\hat{S}_i^t\right) - \frac{1}{\hat{g}}\sum_{j\in\hat{G}_i^t}\mathbf{x}_j\right\|^2 \\ &\leq \kappa\frac{1}{|\mathcal{H}|}\sum_{i\in\mathcal{H}}\mathrm{Var}_{\hat{G}_i^t}(\mathbf{x}).\end{aligned}$$

Now, for each i, using Lemma A.5, since $\hat{G}_i^t \subset \hat{\mathcal{H}}_i^t$ we have

$$\mathrm{Var}_{\hat{G}_i^t}(\mathbf{x}) \leq \frac{s+1-\hat{b}}{s+1-2\hat{b}}\mathrm{Var}_{\hat{\mathcal{H}}_i^t}(\mathbf{x}).$$

Taking the expectation and using Lemma A.6, we get:

$$\begin{aligned}\mathbb{E}_t\left[\mathrm{Var}_{\hat{G}_i^t}(\mathbf{x})\right] &\leq \frac{s+1-\hat{b}}{s+1-2\hat{b}}\frac{\hat{h}-1}{\hat{h}}\frac{|\mathcal{H}|}{|\mathcal{H}|-1}\mathrm{Var}_{\mathcal{H}}(\mathbf{x}) \\ &\leq \frac{s+1-\hat{b}}{s+1-2\hat{b}}\mathrm{Var}_{\mathcal{H}}(\mathbf{x}).\end{aligned}$$

Noting $\gamma_1 = \frac{s+1-\hat{b}}{s+1-2\hat{b}}$, we can write:

$$\mathbb{E}_t[\|T_1\|^2] \leq \gamma_1\kappa\mathrm{Var}_{\mathcal{H}}(\mathbf{x}).$$

For the term T_2, we have:

$$T_2 = \frac{1}{|\mathcal{H}|}\sum_{i\in\mathcal{H}}\left(\frac{1}{\hat{g}}\sum_{j\in\hat{G}_i^t}\mathbf{x}_j - \frac{1}{\hat{h}}\sum_{j\in\hat{\mathcal{H}}_i^t}\mathbf{x}_j\right).$$

For each i, using Lemma A.3, we have:

$$\left\|\frac{1}{\hat{g}}\sum_{j\in\hat{G}_i^t}\mathbf{x}_j - \frac{1}{\hat{h}}\sum_{i\in\hat{\mathcal{H}}_i^t}\mathbf{x}_j\right\|^2 \leq \frac{\hat{b}}{s+1-2\hat{b}}\mathrm{Var}_{\hat{\mathcal{H}}_i^t}(\mathbf{x}).$$

Taking the expectation, and using Lemma A.6, we get:

$$\begin{aligned}\mathbb{E}\left[\left\|\frac{1}{\hat{g}}\sum_{j\in\hat{G}_i^t}\mathbf{x}_j - \frac{1}{\hat{h}}\sum_{i\in\hat{\mathcal{H}}_i^t}\mathbf{x}_j\right\|^2\right] &\leq \frac{\hat{b}}{s+1-2\hat{b}}\frac{\hat{h}-1}{\hat{h}}\frac{|\mathcal{H}|}{|\mathcal{H}|-1}\mathrm{Var}_{\mathcal{H}}(\mathbf{x}) \\ &\leq \gamma_2\mathrm{Var}_{\mathcal{H}}(\mathbf{x}),\end{aligned}$$

with $\gamma_2 = \frac{\hat{b}}{s+1-2\hat{b}}$.

The term T_3 is the same as the T_2 in the previous proof of the synchronous case. Using Equation (18), we have:

$$\mathbb{E}_t\left[\|T_3\|^2\right] \leq \frac{|\mathcal{H}|-\hat{h}}{\hat{h}(|\mathcal{H}|-1)|\mathcal{H}|}\mathrm{Var}_{\mathcal{H}}(\mathbf{x}).$$

In conclusion, using AM-QM inequality, we have:

$$\mathbb{E}_t[\|\overline{\mathbf{x}_{\mathcal{H}}^+} - \overline{\mathbf{x}_{\mathcal{H}}}\|] \leq \left(3\gamma_1\kappa + 3\gamma_2 + 3\frac{|\mathcal{H}|-\hat{h}}{\hat{h}(|\mathcal{H}|-1)|\mathcal{H}|}\right)\mathrm{Var}_{\mathcal{H}}(\mathbf{x}).$$

- **Second inequality.**

  Here again we use Equation (19) which we restate here

$$\mathbb{E}_t\left[\frac{1}{|\mathcal{H}|}\sum_{i\in\mathcal{H}}\left\|\mathbf{x}_i^+ - \overline{\mathbf{x}_{\mathcal{H}}^+}\right\|^2\right] \leq \mathbb{E}_t\left[\frac{1}{|\mathcal{H}|}\sum_{i\in\mathcal{H}}\left\|\mathbf{x}_i^+ - \overline{\mathbf{x}_{\mathcal{H}}}\right\|^2\right].$$

  This time, for each i, using AM-QM, we write :

$$\begin{aligned}\left\|\mathbf{x}_i^+ - \overline{\mathbf{x}_{\mathcal{H}}}\right\|^2 &= \left\|\mathcal{R}\left(\mathbf{x}_j, j\in\hat{S}_i^t\right) - \overline{\mathbf{x}_{\mathcal{H}}}\right\|^2 \\ &\leq 3\left\|\mathcal{R}\left(\mathbf{x}_j, j\in\hat{S}_i^t\right) - \overline{\mathbf{x}_{\hat{G}_i^t}}\right\|^2 + 3\left\|\overline{\mathbf{x}_{\hat{G}_i^t}} - \overline{\mathbf{x}_{\hat{\mathcal{H}}_i^t}}\right\|^2 + 3\left\|\overline{\mathbf{x}_{\hat{\mathcal{H}}_i^t}} - \overline{\mathbf{x}_{\mathcal{H}}}\right\|^2.\end{aligned}$$

  From the previous analysis of the term T_1 in the first inequality, we have:

$$\mathbb{E}_t\left[\left\|\mathcal{R}\left(\mathbf{x}_j, j\in\hat{S}_i^t\right) - \overline{\mathbf{x}_{\hat{G}_i^t}}\right\|^2\right] \leq \gamma_1\kappa\mathrm{Var}_{\mathcal{H}}(\mathbf{x}), \tag{22}$$

  with $\gamma_1 = \frac{s+1-\hat{b}}{s+1-2\hat{b}}$.

  Using Lemma A.3, we have:

$$\left\|\overline{\mathbf{x}_{\hat{G}_i^t}} - \overline{\mathbf{x}_{\hat{\mathcal{H}}_i^t}}\right\|^2 \leq \frac{\hat{b}}{s+1-2\hat{b}}\mathrm{Var}_{\hat{\mathcal{H}}_i^t}(\mathbf{x}).$$

  Taking the expectation:

$$\leq \gamma_2\mathrm{Var}_{\mathcal{H}}(\mathbf{x}).$$

  For the last term, taking the expectation and using Lemma A.4, for all $\in \mathcal{H}$:

$$\mathbb{E}_t\left[\left\|\overline{\mathbf{x}_{\hat{\mathcal{H}}_i^t}} - \overline{\mathbf{x}_{\mathcal{H}}}\right\|^2\right] \leq \frac{|\mathcal{H}|-\hat{h}}{(|\mathcal{H}|-1)\hat{h}}\mathrm{Var}_{\mathcal{H}}(\mathbf{x}).$$

  Taking the average over all the honest nodes, and combining the three terms, we get :

$$\mathbb{E}_t\left[\frac{1}{|\mathcal{H}|}\sum_{i\in\mathcal{H}}\left\|\mathbf{x}_i^+ - \overline{\mathbf{x}_{\mathcal{H}}^+}\right\|^2\right] \leq \left(3\gamma_1\kappa + 3\gamma_2 + 3\frac{|\mathcal{H}|-\hat{h}}{\hat{h}(|\mathcal{H}|-1)}\right)\mathrm{Var}_{\mathcal{H}}(\mathbf{x}),$$

  which concludes the proof.

□

A.5.4 Discussion

Similar to the synchronous case, Lemma A.9 shows that the iterates of Algorithm 1 satisfy (α, λ)-reduction. More importantly, α and λ are of the same order of magnitude as for the synchronous case, and only the constants are worse.

Indeed, knowing that $\gamma_2 \leq \kappa$ and $\gamma_1 < 2$, we can deduce Lemma 5.2, which in particular states that:

$$\mathbb{E}_t\left[\frac{1}{|\mathcal{H}|}\sum_{i\in\mathcal{H}}\left\|\mathbf{x}_i^+ - \overline{\mathbf{x}_{\mathcal{H}}^+}\right\|^2\right] \leq \left(9\kappa + 3\frac{|\mathcal{H}| - \hat{h}}{\hat{h}(|\mathcal{H}|-1)}\right)\mathrm{Var}_{\mathcal{H}}(\mathbf{x}).$$

In this case, the theoretic convergence of the algorithm requires that $9\kappa + 3\frac{|\mathcal{H}|-\hat{h}}{\hat{h}(|\mathcal{H}|-1)} < 1$, which reduces the Byzantine fraction that the algorithm can support. However, up to constants, this is the same dependence on κ, $\hat{h}$ and $|\mathcal{H}|$ as in the synchronous case.

A.6 Proof of Theorem 5.6

Theorem 5.6 (Convergence of Algorithm 1)**.** *Under the assumptions 5.3, 5.4 and 5.5, assuming that s and $\hat{b}$ satisfy Equation* (3) *and $\mathcal{R}$ satisfies Definition 5.1 with $m = s + 1 - \hat{b}$, for a choice of η satisfying Equation* (23)*, for all $i \in \mathcal{H}$, after T iterations, with probability at least p on the sampling procedure*

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\|\nabla F_{\mathcal{H}}\left(\mathbf{x}_i^t\right)\|^2\right] \leq 2\sqrt{\frac{\left(c_2 L\sigma^2 + \frac{432L}{T}\left(\frac{\sigma^2}{n-b}\right)\right)c_0}{T}} + 2\sqrt[3]{\frac{9c_0^2 c_1 nL^2(\sigma^2+G^2)}{T^2}} + 12\frac{Lc_0}{T} + c_3 G^2,$$

with $c_0 = 12\left(F_{\mathcal{H}}\left(\overline{\mathbf{x}}_{\mathcal{H}}^0\right) - F_{\mathcal{H}}^*\right), c_1 = \frac{18\alpha(1+\alpha)}{(1-\alpha)^2}, c_2 = 72L\left(\frac{3}{n-b} + 2c_1 + \frac{9\lambda}{2}(2c_1+3)\right),$
and $c_3 = 6\left(6c_1 + \frac{9\lambda}{2}(4c_1+9)\right)$*, where* $\alpha = 9\kappa + 3\frac{|\mathcal{H}|-\hat{h}}{(|\mathcal{H}|-1)\hat{h}}$ *and* $\lambda = 9\kappa + 3\frac{|\mathcal{H}|-\hat{h}}{(|\mathcal{H}|-1)|\mathcal{H}|\hat{h}}$.

Proof. Based on the proof of Theorem 1 from Farhadkhani et al. (2023), we have after T iterations satisfying α, λ reduction, if $\eta \leq \frac{1}{12L}$:

$$\frac{1}{T}\sum_{t=0}^{T}\mathbb{E}\left[\|\nabla F_{\mathcal{H}}\left(\mathbf{x}_i^t\right)\|^2\right] \leq \frac{c_0}{\eta T} + c_2\eta L\sigma^2 + \frac{432\eta L}{T}\left(\frac{\sigma^2}{n-b}\right) + c_3 G^2 + 9c_1 n\eta^2 L^2(\sigma^2+G^2),$$

with $c_0 = 12(F_{\mathcal{H}}(\overline{\theta^0}) - F_{\mathcal{H}}^*)$, $c_1 = \frac{18\alpha(1+\alpha)}{(1-\alpha)^2}$, $c_2 = 72L\left(\frac{3}{n-b} + 2c_1 + \frac{9\lambda}{2}(2c_1+3)\right)$, $c_3 = 6\left(6c_1 + \frac{9\lambda}{2}(4c_1+9)\right)$.

Taking

$$\eta \leq \min\left\{\frac{1}{12L}, \sqrt[3]{\frac{c_0}{9Tc_1 nL^2(\sigma^2+G^2)}}, \sqrt{\frac{c_0}{T\left(c_2 L\sigma^2 + \frac{432L}{T}\left(\frac{\sigma^2}{n-b}\right)\right)}}\right\}, \tag{23}$$

we get

$$\frac{1}{T}\sum_{t=0}^{T}\mathbb{E}\left[\|\nabla F_{\mathcal{H}}\left(\mathbf{x}_i^t\right)\|^2\right] \leq 2\sqrt[3]{\frac{9c_0^2 c_1 nL^2(\sigma^2+G^2)}{T^2}} + 2\sqrt{\frac{\left(c_2 L\sigma^2 + \frac{432L}{T}\left(\frac{\sigma^2}{n-b}\right)\right)c_0}{T}} + 12\frac{Lc_0}{T} + c_3 G^2.$$

This choice of η is slightly different than the original proof, and provides better guarantees, especially with respect to the dependence on the term $\sigma^2 + G^2$. □

A.7 Proof of Corollary 5.7

Corollary 5.7 (Leading-order stochastic convergence rate)**.** *Under the conditions of Theorem 5.6, when $\mathcal{R}$ satisfies Definition 5.1 with $\kappa = \mathcal{O}\left(\frac{\hat{b}}{s+1}\right)$, with probability at least p, Algorithm 1 satisfies (b, ε)-resilience (Definition 3.1) with:*

$$\varepsilon \in \mathcal{O}\left(\sqrt{\frac{\hat{b}+1}{(s+1)T}\sigma^2} + \frac{\hat{b}+1}{s+1}G^2\right).$$

Proof. In a similar fashion to the proof of Corollary 1 in Farhadkhani et al. (2023), we have:

$$c_0 = 12(F_{\mathcal{H}}(\overline{\theta^0}) - F^*_{\mathcal{H}}) = \mathcal{O}(1),$$
$$c_1 = \frac{18\alpha(1+\alpha)}{(1-\alpha)^2} \in \mathcal{O}(\alpha),$$
$$c_2 = 72L\left(\frac{3}{n-b} + 2c_1 + \frac{9\lambda}{2}(2c_1+3)\right) = \mathcal{O}(\frac{1}{n} + \alpha + \lambda),$$
$$c_3 = 6\left(6c_1 + \frac{9\lambda}{2}(4c_1+9)\right) = \mathcal{O}(\alpha+\lambda).$$

Since $\alpha = 9\kappa + 3\frac{|\mathcal{H}|-\hat{h}}{(|\mathcal{H}|-1)\hat{h}}$ and $\lambda = 9\kappa + 3\frac{|\mathcal{H}|-\hat{h}}{(|\mathcal{H}|-1)|\mathcal{H}|\hat{h}}$. Following Allouah et al. (2023), using NNM pre-aggregation followed by trimmed mean allows us to have $\kappa \in \mathcal{O}(\frac{\hat{b}}{s+1})$.

It follows that in both cases, we have:

$$\alpha \in \mathcal{O}\left(\frac{\hat{b}+1}{s+1}\right), \lambda \in \mathcal{O}\left(\frac{\hat{b}+1}{s+1}\right).$$

From this, ignoring the higher order term (in T), we can conclude that Algorithm 1 satisfies (b, ε)-resilience (Definition 3.1) with:

$$\varepsilon \in \mathcal{O}\left(\sqrt{\frac{\sigma^2}{nT} + \frac{(\hat{b}+1)\sigma^2}{(s+1)T}} + \frac{\hat{b}+1}{s+1}G^2\right),$$

which can be simplified to get the corollary.

□

A.8 Proof of Lemma 4.1

Lemma 4.1. *Let $p < 1$ and assume s and $\hat{b}$ are such that $b/n \leq \hat{b}/s+1 < 1/2$ and*

$$s \geq \min\left\{n, D\left(\frac{\hat{b}}{s+1}, \frac{b}{n}\right)^{-1} \ln\left(\frac{T|\mathcal{H}|}{1-p}\right)\right\} - 1, \tag{3}$$

with $D(\alpha, \beta) = \alpha \ln(\alpha/\beta) + (1-\alpha)\ln((1-\alpha)/(1-\beta))$.

Then the event Γ holds with probability at least p.

Proof. Following the description in Section A.1, $b_i^t \sim \text{HG}(n, b, s+1)$. Hence, by Lemma 13 in Allouah et al. (2024a), we have:

$$\mathbb{P}(b_i^t \geq \hat{b}) \leq \exp\left(-(s+1)D\left(\frac{\hat{b}}{s+1}, \frac{b}{n}\right)\right). \tag{24}$$

Assuming s satisfies Equation (3), we have:

$$\mathbb{P}(b_i^t \geq \hat{b}) \leq \frac{1-p}{|\mathcal{H}|T}. \tag{25}$$

We can conclude by taking the union over all the events $\{b_i^t \geq \hat{b}\}$ for $i \in \mathcal{H}$ and $t \leq T$.

□

A.9 Proof of Lemma 4.2

The following Lemma extends Lemma 2 from Allouah et al. (2024a) to the case when $\hat{b}/s+1$ needs to be less than $1/3$.

Lemma A.10. *Consider $\hat{s}$ defined as follows:*

$$\hat{s} \coloneqq \left\lceil D\left(\frac{1}{3}, \frac{b}{n}\right)^{-1} \ln\left(\frac{27T|\mathcal{H}|}{8(1-p)}\right)\right\rceil + 2, \tag{26}$$

where D is defined in Lemma 4.1. If $s \geq \hat{s}$, then there exists $\hat{b}$ such that Equation (3) *holds and $b/n < \hat{b}/s+1 < 1/3$.*

Proof. Following the proof of Lemma 2 from Allouah et al. (2024a), let $\beta = b/n$ and $\alpha \in [1/3 - 1/s+1, 1/3)$. $D_\alpha(\alpha, \beta)$ denotes the derivative of $D(\alpha, \beta)$ with respect to the first parameter, and has the following expression:

$$D_\alpha(\alpha, \beta) = \ln\left(\frac{\alpha}{\beta}\right) - \ln\left(\frac{1-\alpha}{1-\beta}\right).$$

We slightly change the original proof here by developing the Taylor expansion around $\alpha = 1/3$, we get:

$$\begin{aligned}
D(\alpha, \beta) &\geq D\left(\frac{1}{3}, \beta\right) + \left(\alpha - \frac{1}{3}\right) D_\alpha\left(\frac{1}{3}, \beta\right) \\
&= \frac{1}{3}\ln\left(\frac{1}{3\beta}\right) + \frac{2}{3}\ln\left(\frac{2}{3(1-\beta)}\right) - \left(\frac{1}{3} - \alpha\right)\left(\ln\left(\frac{1}{3\beta}\right) - \ln\left(\frac{2}{3(1-\beta)}\right)\right) \\
&= \alpha\ln\left(\frac{1}{3\beta}\right) + (1-\alpha)\ln\left(\frac{2}{3(1-\beta)}\right) \\
&= 3\alpha\left(\frac{1}{3}\ln\left(\frac{1}{3\beta}\right) + \frac{2}{3}\ln\left(\frac{2}{3(1-\beta)}\right)\right) + (1-3\alpha)\ln\left(\frac{2}{3(1-\beta)}\right) \\
&= 3\alpha D\left(\frac{1}{3}, \frac{b}{n}\right) + (1-3\alpha)\left(\ln\left(\frac{1}{1-\beta}\right) - \ln\left(\frac{3}{2}\right)\right) \\
&> 3\alpha D\left(\frac{1}{3}, \frac{b}{n}\right) - (1-3\alpha)\ln\left(\frac{3}{2}\right).
\end{aligned}$$

Since $\alpha \geq 1/3 - 1/s+1$, we have $3\alpha \geq 1 - 3/s+1$.

Consequently,

$$D(\alpha,\beta) > \left(1-\frac{3}{s+1}\right) D\left(\frac{1}{3},\frac{b}{n}\right) - \frac{3}{s+1}\ln\left(\frac{3}{2}\right).$$
$$= \frac{(s-2)D\left(\frac{1}{3},\frac{b}{n}\right) - 3\ln\left(\frac{3}{2}\right)}{s+1}.$$

Taking $s \geq \hat{s} = \left\lceil D\left(\frac{1}{3},\frac{b}{n}\right)^{-1} \ln\left(\frac{27T|\mathcal{H}|}{8(1-p)}\right)\right\rceil + 2$, it follows:

$$D(\alpha,\beta) > \frac{\ln\left(\frac{27T|\mathcal{H}|}{8(1-p)}\right) - 3\ln\left(\frac{3}{2}\right)}{s+1}$$
$$= \frac{\ln\left(\frac{T|\mathcal{H}|}{1-p}\right)}{s+1}.$$

At this stage, taking $\hat{b} = \left\lceil \frac{s+1}{3} - 1 \right\rceil$, and setting $\alpha = \hat{b}/s+1$ which satisfies $\alpha \in [1/3 - 1/s+1, 1/3)$ (by definition of the ceiling function $t \mapsto \lceil t \rceil$), we get:

$$D\left(\frac{\hat{b}}{s+1},\frac{b}{n}\right) > \frac{\ln\left(\frac{T|\mathcal{H}|}{1-p}\right)}{s+1}.$$

In conclusion, s and $\hat{b}$ satisfy Equation (3), and $b/n < \hat{b}/s+1 < 1/3$. □

Lemma 4.2. *Suppose $0 < b/n < 1/3$, and define the following sampling threshold*

$$s_* \coloneqq \left\lceil \max\left\{\frac{4/9}{(1/3 - b/n)^2}, \frac{3}{b/n}\right\} \ln\left(\frac{27T|\mathcal{H}|}{8(1-p)}\right)\right\rceil + 2.$$

if $s \geq s_*$, then there exists $\hat{b}_*$ such that s and $\hat{b}_*$ satisfy Equation (3)*, $\hat{b}_*/s+1 < 1/3$ and additionally $\frac{\hat{b}_*}{s+1} \in \mathcal{O}(b/n)$.*

Proof. Let $\beta = b/n$ and $\eta \coloneqq 1/3 - \beta$. Let $0 < \alpha \leq 1/3$, and define g_α as follows:

$$g_\alpha(u) \coloneqq D(\alpha, u).$$

The first, second and third order derivatives of g_α write as follows :

$$g'_\alpha(u) = -\frac{\alpha}{u} + \frac{1-\alpha}{1-u}$$
$$g''_\alpha(u) = \frac{\alpha}{u^2} + \frac{1-\alpha}{(1-u)^2}.$$
$$g'''_\alpha(u) = -\frac{2\alpha}{u^3} + \frac{2(1-\alpha)}{(1-u)^3}.$$

For $0 < u \leq \alpha$, we will verify that $g'''_\alpha(u) \leq 0$. Indeed, the function $t \mapsto t/1-t$ is increasing on $(0,1)$, which implies that :

$$\frac{u}{1-u} \leq \frac{\alpha}{1-\alpha},$$

Raising both sides to the third power, and using the fact that $\alpha \leq 1/3 < 1/2$ which implies that $\alpha/1-\alpha < 1$, we get :

$$\frac{u^3}{(1-u)^3} \leq \frac{\alpha^3}{(1-\alpha)^3} \leq \frac{\alpha}{1-\alpha}.$$

This shows that $g'''_\alpha(u) \leq 0$ for $0 < u \leq \alpha$. Hence $u \mapsto g''_\alpha(u)$ is decreasing in $(0, \alpha)$, and consequently:

$$g''_\alpha(u) \geq g''_\alpha(\alpha) = \frac{1}{\alpha} + \frac{1}{1-\alpha}. \tag{27}$$

We also have $g_\alpha(\alpha) = 0$ and $g'_\alpha(\alpha) = 0$.

Applying the mean-value form of Taylor theorem between α and β, we have $\exists u \in [\beta, \alpha]$, such that :

$$g_\alpha(\beta) = g''_\alpha(u)\frac{(\alpha-\beta)^2}{2}.$$

Using Equation (27), we can deduce that:

$$g_\alpha(\beta) \geq \left(\frac{1}{\alpha} + \frac{1}{1-\alpha}\right)\frac{(\alpha-\beta)^2}{2}.$$

For the choice of $\alpha = 1/3$, we get

$$D\left(\frac{1}{3}, \beta\right) \geq \frac{9}{4}\eta^2. \tag{28}$$

Using the definition of s_* and Equation (28), we have:

$$\begin{aligned} s_* &\geq \frac{4}{9\eta^2}\ln\left(\frac{27T|\mathcal{H}|}{8(1-p)}\right) + 2 \\ &\geq D\left(\frac{1}{3}, \beta\right)^{-1}\ln\left(\frac{27T|\mathcal{H}|}{8(1-p)}\right) + 2. \end{aligned}$$

$s_* \geq \hat{s}$ where $\hat{s}$ was introduced in Equation (26), therefore by Lemma A.10, we have the existence of $\hat{b}_*$ that satisfies Equation (3), and such that $b/n < \hat{b}_*/s+1 < 1/3$. We take b_* to be the minimum value possible.

It remains to show that $\hat{b}_*/s+1 \in \mathcal{O}(b/n)$. We consider two cases.

- **Case 1:** $\beta \geq 1/8$

  This case is trivial since $\hat{b}_*/s+1 < 1/3 \in \mathcal{O}(\beta)$

- **Case 2:** $\beta \leq 1/8$

  For this case, we follow the same steps as in the proof of Lemma 4 in Allouah et al. (2024a). The key idea is to choose $\hat{b} = \lceil 2\beta(s+1) \rceil$, for which $\hat{b}/s+1 \in \mathcal{O}(b/n)$ is obvious. It suffices to show that this choice satisfies Equation (3). For completion, we reiterate the proof here.

  Define $G(\beta) = D(2\beta, \beta)$. We have:

$$G(\beta) = 2\beta\ln(2) + (1-2\beta)\ln\left(\frac{1-2\beta}{1-\beta}\right).$$

The first and second order derivatives are as follows

$$G'(\beta) = 2\ln(2) - 2\ln\left(\frac{1-2\beta}{1-\beta}\right) - \frac{1}{1-\beta}$$
$$G''(\beta) = \frac{1}{(1-\beta)^2(1-2\beta)}.$$

Since $G''(\beta)$ is positive, $G'(\beta)$ is increasing, and we have:

$$G'(\beta) \geq G'(0) = 2\ln(2) - 1.$$

Given that $G(0,0) = 0$, we have:

$$G(\beta) \geq (2\ln(2) - 1)\beta > \frac{1}{3}\beta.$$

Since s_* satisfies :

$$s_* \geq \frac{3}{\beta}\ln\left(\frac{27T|\mathcal{H}|}{8(1-p)}\right),$$

we have:

$$\begin{aligned} D(2\beta,\beta) &\geq \frac{\ln\left(\frac{27T|\mathcal{H}|}{8(1-p)}\right)}{s_*} \\ &\geq \frac{\ln\left(\frac{27T|\mathcal{H}|}{8(1-p)}\right)}{s+1}. \end{aligned}$$

For the choice of $\hat{b} = \lceil 2\beta(s+1)\rceil$, we have : $\hat{b}/s+1 \geq 2\beta$. Using that $D(\alpha,\beta)$ is increasing in the first parameter, we have:

$$\begin{aligned} D\left(\hat{b}/s+1, \beta\right) &\geq D(2\beta,\beta) \\ &\geq \frac{\ln\left(\frac{27T|\mathcal{H}|}{8(1-p)}\right)}{s+1}. \end{aligned}$$

Hence, $\hat{b}$ satisfies Equation (3). Additionally, $\hat{b}_*$ being minimal, we have

$$\begin{aligned} \frac{\hat{b}_*}{s+1} &\leq \frac{\hat{b}}{s+1} \\ &\leq 2\beta + \frac{1}{s+1}. \end{aligned}$$

By using the fact that $T \geq 1$ and $p \geq 0$, we have

$$s+1 \geq s_*+1 \geq \frac{3}{\beta}\ln\left(\frac{27T|\mathcal{H}|}{8(1-p)}\right) \geq \frac{3}{\beta}\ln\left(\frac{27}{8}\right) \geq \frac{3}{\beta}.$$

Consequently:

$$\frac{\hat{b}_*}{s+1} \leq \frac{7}{3}\beta \in \mathcal{O}\left(\frac{b}{n}\right).$$

Additionally, since $\beta \leq \frac{1}{8}$, we have $\frac{7}{3}\beta < \frac{1}{3}$, which concludes the proof.

□

A.10 On $(m, \kappa, \hat{b})$-robustness

Reformulated in our notation, Lemma 1 in Allouah et al. (2023) shows that given an aggregation rule F that satisfies $(m, \kappa', \hat{b})$-robustness, its composition with NNM preaggregation $\mathcal{R} = F \circ NNM$, satisfies $(m, \kappa, \hat{b})$-robustness with

$$\kappa \leq \frac{8\hat{b}}{m - \hat{b}}(\kappa' + 1).$$

Allouah et al. (2023) also shows that CWTM aggregation satisfies $(m, \kappa', \hat{b})$-robustness with

$$\kappa' = \frac{6\hat{b}}{m - 2\hat{b}} \left(1 + \frac{\hat{b}}{m - 2\hat{b}}\right).$$

For completeness, we recall the definition of the NNM preaggregation given m inputs $x_1, \ldots, x_m$ among which b are Byzantine.

$$NNM(x_1, \ldots, x_m) = \frac{1}{m - b} \sum_{j=1}^{m-b} x_{i:j},$$

with $x_{i:j}$ is the j^{th} nearest vector to x_i among the inputs.

B Simulating the Effective Adversarial fraction

Algorithm 3 shows the pseudo-code of the procedure we use to choose s and $\hat{b}$. The algorithm makes use of the CDF of hypergeometric distributions, available in multiple statistical computing libraries, including Scipy which we used in our code.

Algorithm 3 Hyperparameters selection

Require: total number of nodes n, total number of attackers b, number of iterations T, grid of s values G_s, the inverse of hypergeometric law $g_s : x \mapsto F^{-1}(x, n-1, b, s)$, desired fraction threshold $\frac{b}{n-1} \leq q < \frac{1}{2}$, a confidence threshold p.

1: $h = n - b$
2: Compute $u = p^{\frac{1}{hT}}$
3: **for** $s \in G_s$ **do**
4: Compute $\kappa_s = \frac{g_s(u)}{s}$
5: **end for**
6: **return** $(s, \hat{b}_s)$ such that s is the smallest satisfying $\kappa_s \leq q$

Remark. This algorithm always returns a tuple $(s, \hat{b})$, provided the grid G_s is large enough. For instance, if $n - 1 \in G_s$, we know that $\hat{b}_{n-1} = b$ and $\kappa_{n-1} = \frac{b}{n-1} \leq q$.

To see why the algorithm works, notice that for $u = p^{\frac{1}{hT}}$, for a given s, noting $Y = \max(X_1, \ldots, X_{hT})$ for $X_i \sim HG(n-1, b, s)$ the distribution of the maximum of hT Hypergeometric distributions, we have:

$$\begin{aligned}
\mathbb{P}(Y \leq g_s(u)) &= \mathbb{P}(Y \leq F^{-1}(u, n-1, b, s)) \\
&= \mathbb{P}(F(Y) \leq p^{\frac{1}{hT}}) \\
&= \mathbb{P}(F(Y)^{hT} \leq p) \\
&= p
\end{aligned}$$

Where the last line follows from the fact that the CDF of Y is F^{hT}.

C Experiments

In the section, we provide additional details about our experiments for reproducibility. Our experiments test Algorithm 2-the synchronous version of RPEL-on multiple datasets, and for different scenarios. It is important to note that from an experiment point of view, since testing Algorithm 1 in the worst case scenario would require including all Byzantine updates in the $s - \hat{b}$ received ones, running Algorithm 1 is somewhat equivalent to running Algorithm 2 with $s' = s + \hat{b}$. It is exactly equivalent in the case when the attacker does not control which honest nodes to exclude from the set $\hat{S}_i^t$. This is the case, for instance, when all the honest nodes have the same probability of being included in $\hat{S}_i^t$.

We do include, nonetheless, two experiments with Algorithm 1 in Section C.3.

C.1 Experimental details

To run our experiments, we used a server with the following specifications:

- HPe DL380 Gen10
- 2 x Intel(R) Xeon(R) Platinum 8358P CPU @ 2.60GHz
- 128 GB of RAM
- 740 GB ssd disk
- 2 Nvidia A10 GPU cards

For the model architectures, we use the compact notation that can be found in El Mhamdi et al. (2021b); Allouah et al. (2023):

L(#outputs) represents a fully-connected linear layer, R stands for ReLU activation, S stands for log-softmax, M stands for 2D-maxpool (kernel size 2), B stands for batch-normalization, and D represents dropout (with fixed probability 0.25). C(#channels) represents a 2D-convolutional layer with (kernel size 3, padding 1, stride 1) for CIFAR-10 and (kernel size 5, padding 0, stride 1) for MNIST and FEMNIST.

Dataset	MNIST	CIFAR-10
Model Type	CNN	CNN
Model Architecture	C(20)-R-M-C(20)-R-M-L(500)-R-L(10)-S	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
Weight L2 regularization	10^{-4}	10^{-2}
Learning Rate	0.5	0.5 for $0 \leq t \leq 500$ 0.1 for $500 \leq t \leq 1000$, 0.02 for $1000 \leq t \leq 1500$, 0.004 for $1500 \leq t \leq 2000$
Batch size	25	50
Momentum	0.9	0.99
Number of seeds	2	2

Table 1: Experimental Details for MNIST and CIFAR-10

For the fixed graph experiments with MNIST, we used 5 seeds instead of 2.

C.2 Comparison to baselines

We compare our method with the following baselines:

- **CS+** algorithm from Gaucher et al. (2025).
- The adaptive clipping version of **ClippedGossip** from He et al. (2022).
- **GTS**, the adaptation of **NNA** algorithm from Farhadkhani et al. (2023) to sparse graphs, as implemented by Gaucher et al. (2025).

Dataset	FEMNIST
Total number of nodes	30
Number of attackers	3
Heterogeneity	$\alpha = 10$
Model Type	CNN
Model Architecture	C(64)-R-M-C(128)-R-M-L(1024)-R-L(62)-S
Weight L2 regularization	10^{-4}
Learning Rate	0.1
Batch size	50
Momentum	0.99
Number of iterations	500
Number of seeds	2

Table 2: Experimental Details for FEMNIST

Dataset	AG's News
Total number of nodes	$20, 30, 50$
Number of attackers	$2, 3, 4$
Heterogeneity	$\alpha = 1$
Model Type	fastText
Learning Rate	0.5
Batch size	64
Momentum	0.9
Number of iterations	400
Number of seeds	2

Table 3: Experimental Details for AG's News

- **RTC** algorithm from Yang and Ghaderi (2024).

We use ALIE attack, previously presented in the paper, and the Dissensus attack by He et al. (2022). The latter is more adapted to fixed graphs, as it leverages the structure of the gossip update to be made by the honest nodes.

For fixed n, s and b, since our method uses $K = n \times s$ model exchanges, we generate a random connected graph with $K/2$ edges. To generate such a graph, we first generate a random spanning tree, using the Networkx implementation. Next, we add random edges to the graph until reaching the desired connectivity.

Remark C.1. The graph construction here is more realistic than the one used in previous works (He et al., 2022; Gaucher et al., 2025), where the honest subgraph is generated first before the Byzantine connections are added. As a result, the honest subgraph may not satisfy the connectivity assumptions laid out by He et al. (2022) and Gaucher et al. (2025).

Remark C.2. The previous works on fixed-graph robust learning (He et al., 2022; Gaucher et al., 2025) assume that each honest node can have at most a certain number of Byzantine neighbors. This quantity is used as a parameter in the algorithm to choose the clipping threshold. Given that in our experiments, we generate random communication graphs, we fixed this quantity to be equal to $\hat{b}$. Theoretically, this should work if the attackers' positions on the graph are random. However, in the case when the attackers' positions are also adversarial, it is necessary to choose this upper bound to be equal to b, the total number of Byzantine nodes, which tremendously reduces the performance, since the honest nodes are required to clip many more neighbors. The honest nodes that have less than $2b$ neighbors in this case would have to clip all the neighbors and learn on their local data only. It is important to note that this limitation is bypassed by our algorithm, which is also robust to the attackers changing their identities from one iteration to another.

Other than the random placement of attackers, we considered the two following placement strategies:

1. **Betweenness centrality maximization.** The attackers are chosen as the nodes with the largest betweenness centrality on the graph (Brandes, 2001), defined for each node v as follows :

$$g(v) = \sum_{s \neq v \neq t} \frac{\sigma_{st}(v)}{\sigma_{st}}, \tag{29}$$

where σ_{st} is the total number of shortest paths from node s to node t and $\sigma_{st}(v)$ is the number of those paths that pass through v.

2. **Vulnerable targets maximization.** Greedily places the attackers in the neighborhood of the most vulnerable nodes. Placing the b byzantine nodes takes b steps where at the t'th step if the set of attackers is A_t, the most vulnerable honest node is the node that maximizes $\frac{|N(i) \cap A_t|}{|N(i)|}$ and the attacking node is chosen to be one of the neighbors.

Analysis The performance gain of RPEL with respect to the baselines is most impressive when the connectivity is low (bottom-left picture in Figure 5 and Figure 6) or when the heterogeneity is high (Figure 8 and Figure 9). The worst client performance, which tracks the accuracy of the honest client who got the worst final accuracy, shows a clear and consistent advantage of using RPEL.

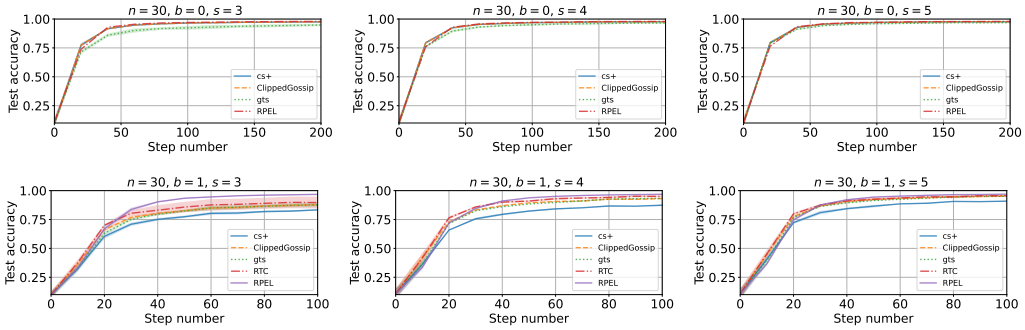


Figure 5: Average test accuracy on MNIST - ALIE attack - Random placement ($\alpha = 1$)

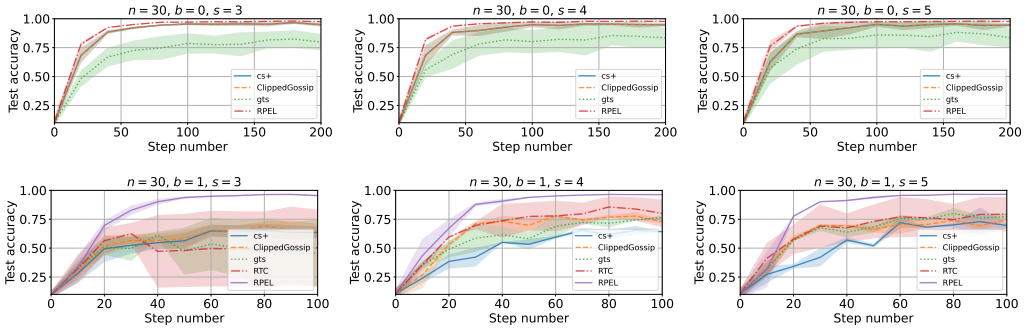


Figure 6: Worst test accuracy on MNIST - ALIE attack - Random placement ($\alpha = 1$)

Local aggregation complexity

Table 4 shows the computational complexity of the robust aggregation. Using logarithmic sampling, the per-round aggregation cost across the network is

$$\mathcal{O}\left(nd\log^2(n)\right),$$

for fixed T, p, and a Byzantine fraction bounded away from the breakdown threshold.

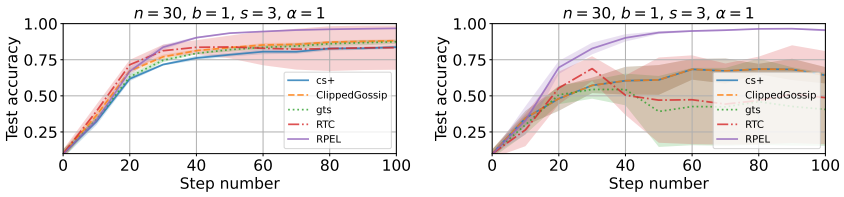


Figure 7: MNIST - (left) Average test accuracy (right) Worst test accuracy - Vulnerable targets placement ($\alpha = 1$)

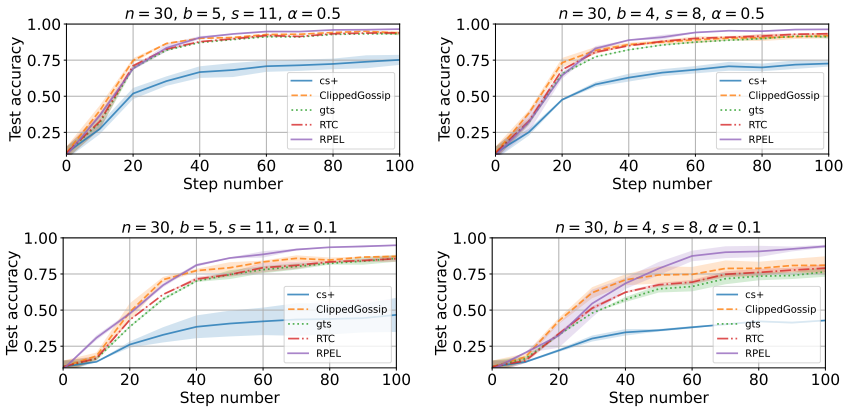


Figure 8: Average test accuracy on MNIST - Higher heterogeneity - Adversarial placement

Operation	Computational complexity
NNM	$\mathcal{O}(m^2 d)$
CWTM	$\mathcal{O}(dm \log m)$
Combined aggregation	$\mathcal{O}(m^2 d)$

Table 4: Computational complexity of the used robust aggregations.

C.3 Other experiments

In this section, we include a few additional experiments.

Asynchronous evaluation

Figure 10 shows the accuracy of Algorithm 1 on MNIST with $n = 30, b = 5, s = 15$. As stated before, we consider the worst case in which all the selected Byzantine attackers are included in the received set of $s - \hat{b}$ updates used for aggregation. Furthermore, noting $b_i^t \leq \hat{b}$ the actual number of Byzantine nodes selected by honest node i at iteration t, we consider two response patterns: when the first $s - \hat{b} - b_i^t$ responding honest nodes are selected uniformly at random out of $S_i^t \cap \mathcal{H}$, and when the honest nodes with the largest local datasets are systematically excluded. These results suggest that the theoretical robustness of the asynchronous protocol translates to practical performance even under biased response availability.

We further evaluate Algorithm 1 on CIFAR-10 with $n = 20$ clients, including $b = 3$ Byzantine clients, and $s = 10$ sampled peers per honest client. We set $\hat{b} = 3$, so each honest client aggregates its own model with the first $s - \hat{b} = 7$ received models using nearest-neighbor mixing followed by coordinate-wise trimmed mean (NNM+CWTM). Since all clients are simulated on a single server, we emulate asynchrony through virtual response delays rather than actual waiting. Honest response delays follow a log-normal distribution with log-space parameters $\mu = 0$ and $\sigma = 0.5$.

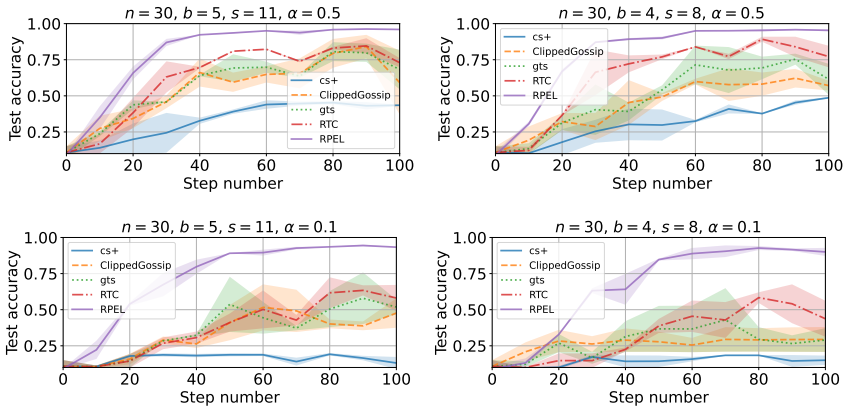


Figure 9: Worst test accuracy on MNIST - Higher heterogeneity - Adversarial placement

We consider two settings: (i) random delays, where an independent delay is sampled for each honest response at every iteration, and (ii) persistent heterogeneous speeds, where each honest client is assigned a fixed latency multiplier throughout training: 1 for 10 fast clients, 5 for five medium-speed clients, and 20 for two slow clients. In both settings, sampled Byzantine clients are assigned zero delay, ensuring that their responses are always retained among the first seven. The attackers perform a receiver-specific ALIE attack, constructing malicious vectors based on the honest peer models retained by each receiver. We compare both asynchronous settings with synchronous RPEL under the same configuration. Table 5 reports mean test accuracy and standard deviation over four seeds, both at the final iteration ($T = 1000$) and averaged over the last three evaluations ($T = 900, 950, 1000$). For each seed, the worst client is defined as the honest client with the lowest test accuracy at $T = 1000$; its last-3 accuracy is computed by averaging the accuracies of this same client at the three evaluation checkpoints. The average last-3 accuracies are 73.15% and 71.96% under random delays and persistent heterogeneous speeds, respectively, compared with 74.77% under synchrony. These results demonstrate that RPEL maintains competitive accuracy under both asynchronous response-delay models, even when Byzantine responses are systematically prioritized.

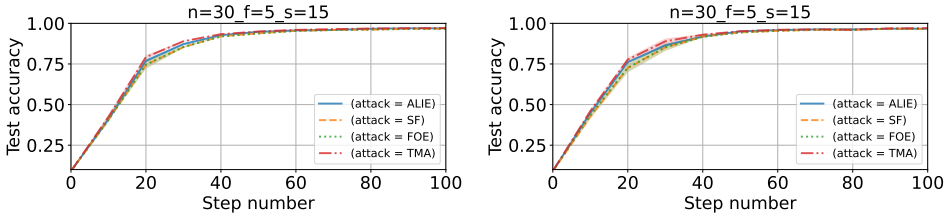


Figure 10: RPEL in asynchrony - (left) random honest nodes (right
Honest nodes with the smallest dataset first

Table 5: Accuracy under simulated asynchrony for CIFAR-10.

Setting	**Average final accuracy (%)**	**Average last-3 accuracy (%)**	**Worst-client final accuracy (%)**	**Worst-client last-3 accuracy (%)**
Random delays	71.77 ± 0.87	73.15 ± 0.28	59.24 ± 5.53	69.58 ± 2.97
Persistent speeds	72.68 ± 2.07	71.96 ± 1.50	64.50 ± 6.75	66.39 ± 1.06
Synchronous	75.34 ± 1.35	74.77 ± 0.54	65.66 ± 3.55	69.10 ± 1.19

Local steps We conduct some experiments on CIFAR-10, for the same default setup, but with 3 local steps at each iteration. Figure 25, Figure 26, and Figure 27 show the test accuracies for $s = 6$, $s = 10$ and $s = 19$ respectively. Unsurprisingly, the performance is better than using a single local

step before communication (Allouah et al., 2024a). The algorithm converges much faster to $75\%+$ accuracy. However, still in this scenario, using only 6 neighbors results in comparable performance with all-to-all communication.

Higher heterogeneity

We run a few other experiments with higher heterogeneity (lower values for α) on the CIFAR-10 and MNIST datasets. Figure 11 shows the performance on MNIST for different heterogeneity levels. This experiment shows that RPEL still performs well and remains robust to the different attacks. Figure 12 shows the accuracy results for CIFAR-10 for extreme levels of heterogeneity. As expected, the performance does degrade as heterogeneity increases. However, this degradation affects both RPEL and the all-to-all robust methods. Interestingly, for $\alpha = 0.1$, increasing the number of sampled nodes to $s = 10$ is enough to get a comparable performance with the all-to-all method. For illustration purposes, Figure 13,Figure 14, Figure 15 and Figure 16 show the label class distribution over the honest neighbors, in the case of the CIFAR-10 experiment, for the different levels of heterogeneity explored in the experiments $\alpha = 1, 0.5, 0.3, 0.1$. It is worth noticing that $\alpha = 1$ already shows high heterogeneity between the different clients. Below $\alpha = 0.5$, the heterogeneity can be considered extreme and pathological.

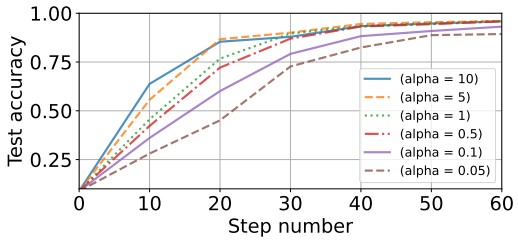


Figure 11: Effect of heterogeneity - MNIST

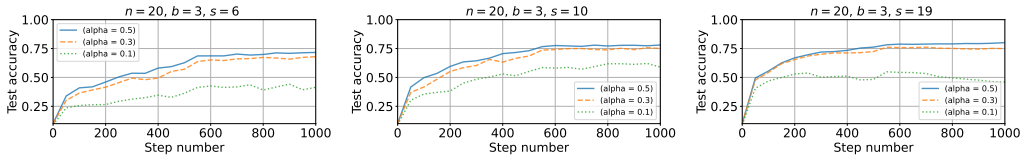


Figure 12: Effect of heterogeneity - CIFAR-10

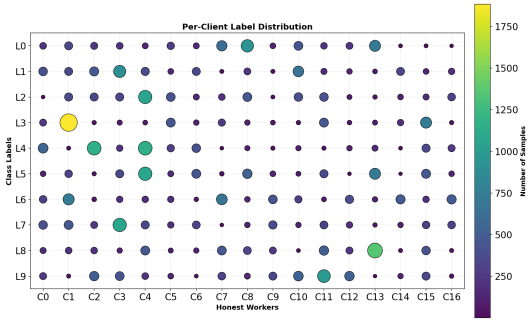


Figure 13: Client label distribution CIFAR-10 - $n = 20, b = 3, \alpha = 1$

Higher Byzantine ratio

Figure 18 shows the Effective Byzantine fraction simulations, similar to Figure 4. Consistent with the theory, RPEL remains robust at higher Byzantine fractions, provided the sampling size is increased accordingly.

Large-scale synthetic experiment

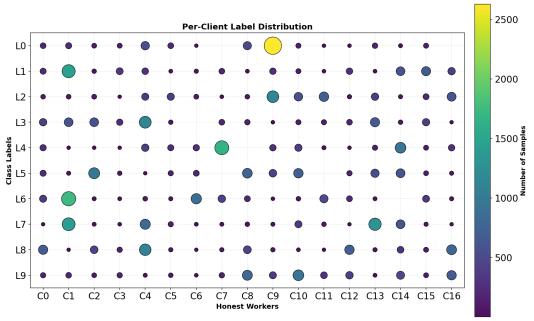


Figure 14: Client label distribution CIFAR-10 - $n = 20, b = 3, \alpha = 0.5$

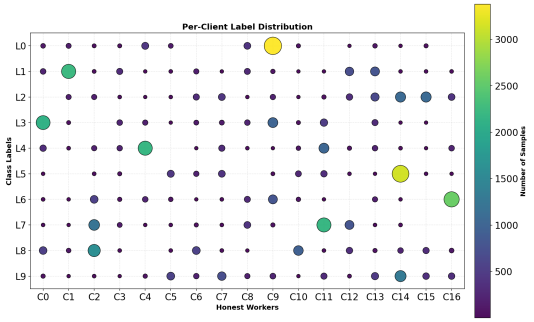


Figure 15: Client label distribution CIFAR-10 - $n = 20, b = 3, \alpha = 0.3$

We evaluate RPEL on a synthetic heterogeneous binary-classification task with up to $10{,}000$ clients, 10% of which are Byzantine. Because all-to-all decentralized baselines are impractical at this scale, we use centralized federated-learning baselines as references.

Clients belong to $K = 10$ latent groups, whose proportions are sampled from a symmetric Dirichlet distribution with parameter 1. For each group k, we sample a feature center $\mu_k \sim \mathcal{N}(0, \tau_\mu^2 I)$ and a scalar label bias $c_k \sim \mathcal{N}(0, \tau_c^2)$. Each client i additionally receives a client-specific feature shift $\delta_i \sim \mathcal{N}(0, \tau_{\text{client}}^2 I)$. Its examples are generated according to

$$x_{i,j} \sim \mathcal{N}(\mu_{g_i} + \delta_i, I), \qquad y_{i,j} \sim \text{Bernoulli}\big(\sigma(w_\star^\top x_{i,j} + c_{g_i})\big)\,,$$

where the shared vector $w_\star$ is sampled once and normalized.

We set $\tau_\mu = 2$, $\tau_c = 1$, and $\tau_{\text{client}} = 0.5$. Each client has between 40 and 400 examples, split equally between training and testing. We train a one-hidden-layer MLP with architecture $20 \to 32 \to 1$. During each communication round, every honest client performs one stochastic-gradient step on a minibatch of 32 examples.

The Byzantine clients use the ALIE attack. We evaluate coordinate-wise trimmed mean (CWTM), both alone and after nearest-neighbor mixing (NNM+CWTM). Each honest RPEL client samples s peers and aggregates their models together with its own model. We select s and $\hat{b}$ so that the effective Byzantine tolerance $\hat{b}/(s+1)$ is approximately 40%.

The Robust FL baseline uses full client participation under the ALIE attack and sets $\hat{b} = b = 0.1n$. The No-attack FL baseline uses the same centralized training procedure without Byzantine clients and serves as an accuracy reference.

Table 6 reports the peer-sampling and robust aggregation parameters.

Table 7 reports pooled test accuracy at rounds 10, 50, and 100.

These experiments provide feasibility evidence that RPEL remains effective as the number of clients increases from $1{,}000$ to $10{,}000$. Although the population grows by a factor of ten, the number of

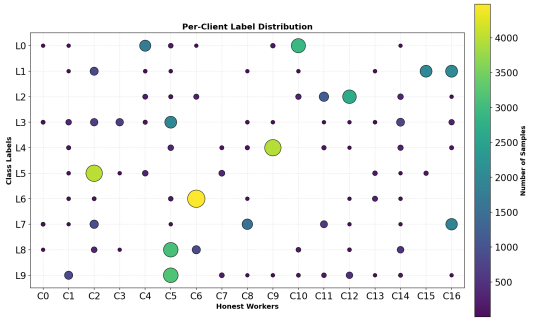


Figure 16: Client label distribution CIFAR-10 - $n = 20, b = 3, \alpha = 0.1$

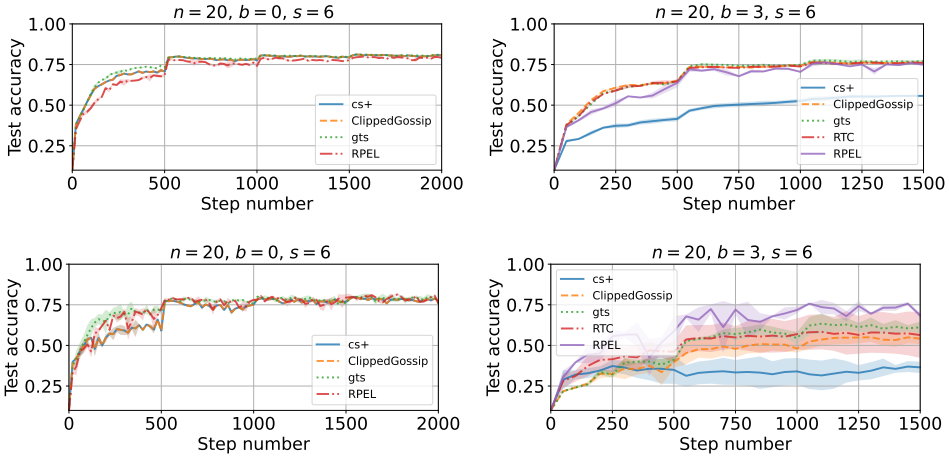


Figure 17: Test accuracies obtained on CIFAR-10 with Dissensus attack, $\alpha = 1$ and 3 local step. (Top row) Average performance. (Bottom row) Worst performance.

peers sampled by each client increases only from 35 to 50. Final pooled accuracy with NNM+CWTM remains between 71.5% and 74.5%.

The $n = 10{,}000$ experiment illustrates the computational motivation for local randomized sampling. Full-participation NNM requires pairwise distances among all $10{,}000$ submitted models, whereas RPEL applies the same pre-aggregation procedure to only 51 models at each client. RPEL with NNM+CWTM obtains 71.5% final accuracy, numerically close to the 71.3% obtained by centralized Robust FL with CWTM alone.

Table 6: RPEL sampling for large-scale synthetic experiments.

Clients n	Peers s	$\hat{b}$	$\hat{b}/(s+1)$
1,000	35	14	38.9%
2,000	40	16	39.0%
5,000	45	18	39.1%
10,000	50	19	37.3%

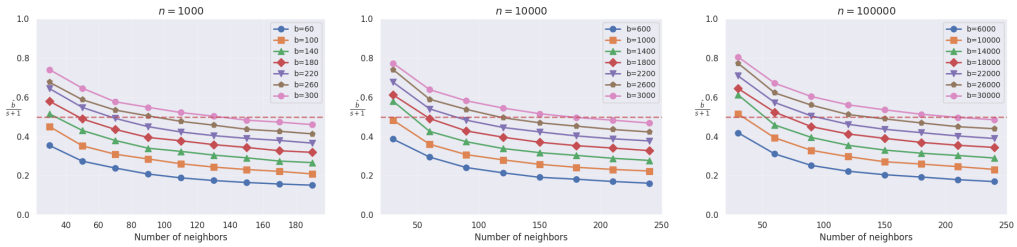


Figure 18: Effective adversarial fraction simulation - higher Byzantine fraction

Table 7: Pooled held-out test accuracy (%) on the synthetic task. Each configuration uses one fixed seed. A dash denotes a configuration that was not run.

n	Method	Aggregation	$T = 10$	$T = 50$	$T = 100$
1,000	RPEL	CWTM	59.3	69.0	70.6
1,000	RPEL	NNM+CWTM	66.8	71.1	73.2
1,000	Robust FL	CWTM	62.9	70.3	71.6
1,000	Robust FL	NNM+CWTM	67.3	71.1	72.2
1,000	No-attack FL	No attack	68.4	71.7	74.1
2,000	RPEL	CWTM	59.9	70.2	72.5
2,000	RPEL	NNM+CWTM	67.8	72.9	74.5
2,000	Robust FL	CWTM	60.7	66.0	67.3
2,000	Robust FL	NNM+CWTM	63.7	66.5	67.6
2,000	No-attack FL	No attack	68.7	72.6	74.8
5,000	RPEL	CWTM	57.1	68.0	70.9
5,000	RPEL	NNM+CWTM	64.5	70.1	71.9
5,000	Robust FL	CWTM	62.5	68.6	70.0
5,000	Robust FL	NNM+CWTM	65.9	69.3	70.7
5,000	No-attack FL	No attack	71.9	74.0	75.9
10,000	RPEL	CWTM	55.3	68.7	70.9
10,000	RPEL	NNM+CWTM	66.5	69.9	71.5
10,000	Robust FL	CWTM	61.4	69.8	71.3
10,000	Robust FL	NNM+CWTM	—	—	—
10,000	No-attack FL	No attack	68.7	71.7	73.5

Partial views

We additionally evaluate the effect of partial views on MNIST and CIFAR-10. Each node i is assigned a fixed random membership view V_i of size v, and at each round samples s peers uniformly from V_i. This protocol is not RPEL as analyzed in the paper: it is a hybrid between a fixed communication topology and RPEL 's randomized sampling, and our current convergence guarantee does not apply to it.

On MNIST with $n = 200$, $b = 20$, and $s = 20$, we reduce the view size from $v = 199$ to $v = 100$ and $v = 50$. On CIFAR-10 with $n = 20$, $b = 3$, and $s = 6$, we log both the final accuracy and the average accuracies over last 3 checkpoints $T = 900, 950, 1000$ for $v = 19$ and $v = 10$. Table 8 and Table 9 show the results over two seeds.

We observe only a slight degradation under these randomly generated views for the CIFAR-10 experiment, while the MNIST experiments shows stable results. This is not unexpected. When the view itself is sampled uniformly and the Byzantine positions are not chosen adversarially relative to the views, the peer set selected in an individual round remains marginally uniform over the global population. Restricting the view mainly introduces topology-dependent mixing. The experiment therefore shows that benign random partial views need not harm performance, but it should not be interpreted as robustness to arbitrary partial views.

Table 8: Effect of view size on MNIST accuracy.

View size	Average final accuracy	Worst-client final accuracy
$m = 199$	94.7 ± 0.3	94.1 ± 0.5
$m = 100$	94.7 ± 0.3	94.4 ± 0.3
$m = 50$	94.7 ± 0.3	94.1 ± 0.7

Table 9: Effect of view size on CIFAR-10 accuracy.

View size	Average final accuracy (%)	Average last-3 accuracy (%)	Worst-client final accuracy (%)	Worst-client last-3 accuracy (%)
19	72.28 ± 1.17	72.52 ± 1.04	60.79 ± 4.53	67.80 ± 4.39
10	70.28 ± 1.80	71.62 ± 1.27	55.71 ± 4.96	64.62 ± 0.64

AG's News

We tested our algorithm on the textual dataset AG's News. Figure 19 shows the results for the three following configurations: (i) $n = 20, b = 2, s = 6$, (ii) $n = 30, b = 3, s = 9$, (iii) $n = 50, b = 5, s = 15$. Keeping the Byzantine fraction constant at 10%, the algorithm reaches the same final accuracy as in the benign setting Figure 20.

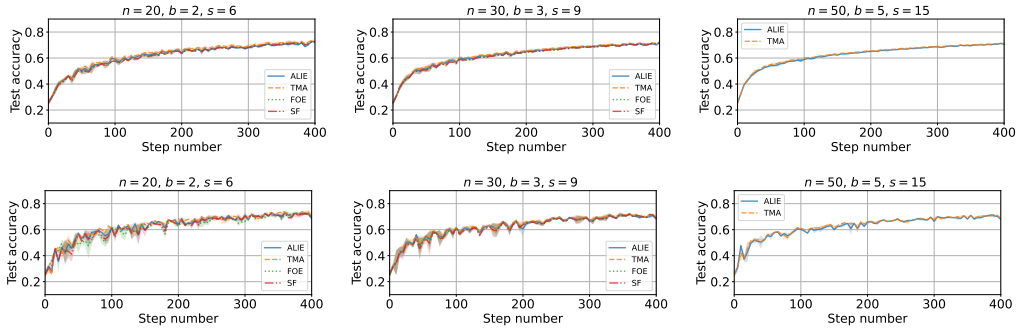


Figure 19: Test accuracy on AG's News dataset. (Top row) Average performance. (Bottom row) Worst performance.

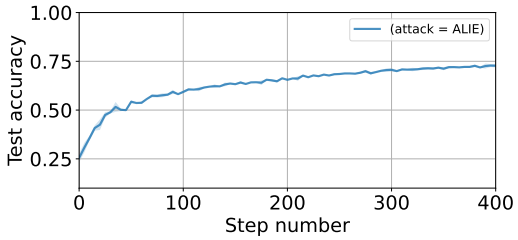


Figure 20: Test accuracy on AG's News dataset. Benign setting $b = 0$.

FEMNIST Figure 30 shows the convergence results for FEMNIST dataset with $n = 30$ and $b = 3$. For comparison, we provide the results in the absence of attackers in Figure 28 and Figure 29. The same conclusions about the good performance of RPEL can be drawn here. Implementation details for FEMNIST experiments are included in Table 2.

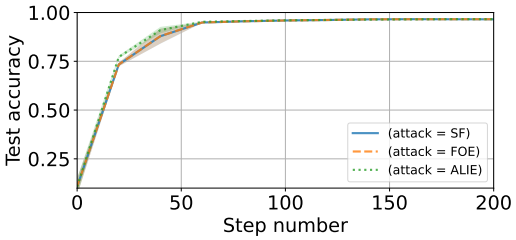


Figure 21: MNIST, $n = 100$, $b = 8$, $s = 15$

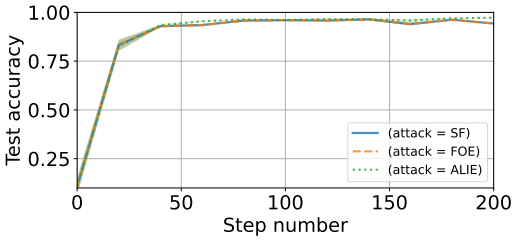


Figure 22: MNIST, $n = 30$, $b = 5$, $s = 15$

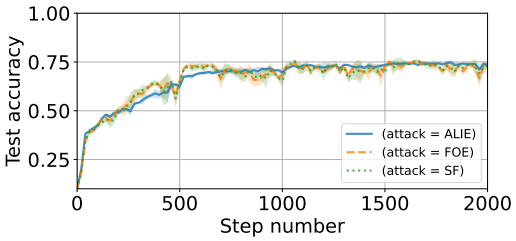


Figure 23: CIFAR-10, $n = 20$, $b = 2$, $s = 6$

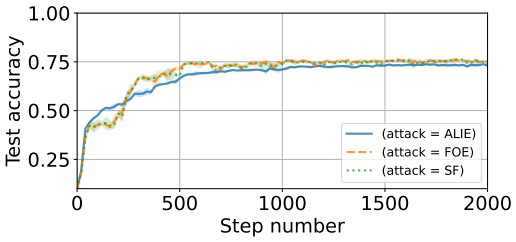


Figure 24: CIFAR-10, $n = 20$, $b = 2$, $s = 19$

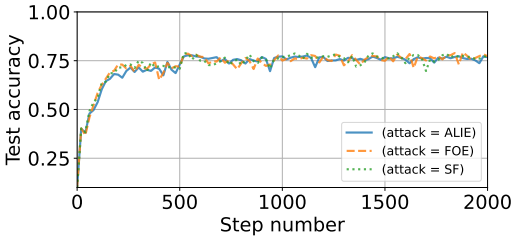


Figure 25: CIFAR-10, $n = 20$, $b = 3$, $s = 6$, 3 local steps

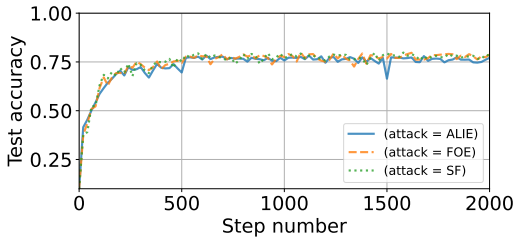


Figure 26: CIFAR-10, $n = 20$, $b = 3$, $s = 10$, 3 local steps

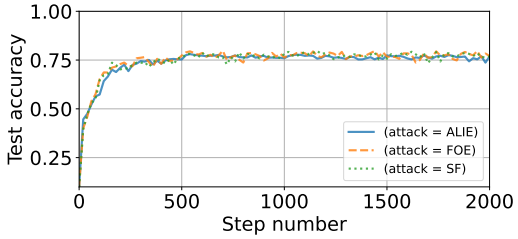


Figure 27: CIFAR-10, $n = 20$, $b = 3$, $s = 19$, 3 local steps

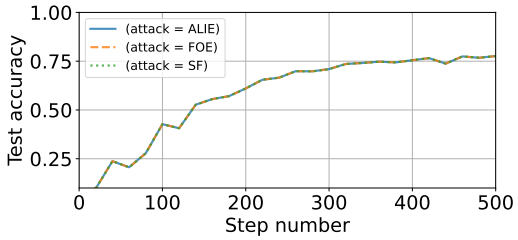


Figure 28: FEMNIST, $n = 30$, $b = 0$, $s = 6$

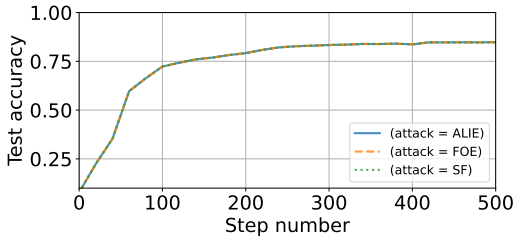


Figure 29: FEMNIST, $n = 30$, $b = 0$, $s = 6$, 3 local steps

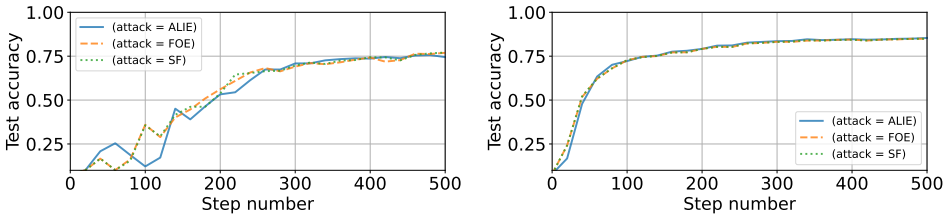


Figure 30: FEMNIST, $n = 30$, $b = 3$, $s = 6$. (Left) 1 local step, (Right) 3 local steps.

C.4 Limitations

While RPEL provides theoretical convergence guarantees and favorable empirical results, several limitations remain. First, our analysis assumes that each client can uniformly sample peers from the global population. Although clients communicate with only a small subset of peers at each iteration, they must be able to discover and address arbitrary participants. This assumption may not hold in networks with restricted connectivity or partial membership views. Our preliminary experiments with fixed random membership views suggest that RPEL can tolerate some restrictions on peer discovery, but extending the theoretical guarantees to arbitrary or adversarially constructed views requires topology-dependent analysis. Second, our analysis assumes a fixed population with a bounded number of Byzantine clients. Dynamic membership, client churn, and Sybil attacks, which may affect the sampling distribution or the effective adversarial fraction, are not covered by our guarantees. Finally, while our experiments with up to $10\,000$ clients on simple synthetic learning tasks suggest that the benefits of randomized communication may extend to larger populations, evaluating RPEL on realistic learning tasks at this scale remains computationally prohibitive with our current resources. Moreover, these experiments do not capture deployment-level aspects such as peer discovery overhead, real network latency, or bandwidth constraints. Our asynchronous experiments emulate response delays rather than actual distributed execution. Investigating these practical considerations and extending RPEL to dynamic or partially connected networks are important directions for future work.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Our claims in the abstract and introduction are precise and consistent with our findings.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitations of the work in a dedicated paragraph Section C.4.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: We provide self-contained proofs for most of the propositions and theorems included in this paper. For the few remaining ones, we refer to some well-recognized papers in the literature.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Justification: We provide detailed guidance on how to reproduce the experiments, including all the used models and hyperparameters, in Section C.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We give open access to our code as well as clear instructions on how to reproduce our experimental results.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: We provide a dedicated section in the appendix (Section C) detailing our experimental setups (e.g. models being used, number of iterations, learning rates).

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We provide error bars for the experimental figures we show in the paper.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Justification: We include a precise description of the compute resources in Section C.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research conducted in this paper respects the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: This work aims at advancing the field of robust distributed machine learning. We highlight the potential societal benefits of enhancing robustness against adversarial machines. We do not foresee any direct negative impacts specific to our work beyond those generally associated with machine learning and artificial intelligence.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.

- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: No new data is released.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Yes. All datasets used in this paper—MNIST, CIFAR-10, FEMNIST, and AG News—are standard public benchmarks. We properly cite the original sources of these datasets in the paper. These datasets are publicly available for research use, and our usage complies with their respective licenses and terms of use.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.

- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: The paper does not release new assets.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: LLMs are solely used for editing and formatting purposes in this paper.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.