Byzantine-Robust Decentralized Learning via Self-Centered Clipping

Lie He * 1  Sai Praneeth Karimireddy * 1 2  Martin Jaggi 1

Abstract

In this paper, we study the challenging task of Byzantine-robust decentralized training on arbitrary communication graphs. Unlike federated learning where workers communicate through a server, workers in the decentralized environment can only talk to their neighbors, making it harder to reach consensus. We identify a novel dissensus attack in which few malicious nodes can take advantage of information bottlenecks in the topology to poison the collaboration. To address these issues, we propose a Self-Centered Clipping (SCCLIP) algorithm for Byzantine-robust consensus and optimization, which is the first to provably converge to an \(O(\delta \max \zeta^2 / \gamma^2)\) neighborhood of the stationary point for non-convex objectives under standard assumptions. Finally, we demonstrate the encouraging empirical performance of SCCLIP under a large number of attacks.

1. Introduction

"Divide et impera".

Many machine learning tasks involve training models on decentralized data due to privacy constraints, distributed training arise as an important topic (McMahan et al., 2017; Kairouz et al., 2019). As the server-worker paradigm may suffer from single point of failure, there are growing amount of works on training in the absence of servers (Lian et al., 2017; Nedic, 2020; Koloskova et al., 2020b). There are two regimes of decentralized training: one allows direct communication between any two workers, such as in a data center; on the other hand, the available communication links are often significantly constrained by physical factors, such as in a sensor network where each device can only communicate with devices nearby, or when one has to respect a given general network topology (e.g. the internet).

Failures—from malfunctioning or even malicious participants—are ubiquitous in all kinds of distributed computing. A Byzantine adversarial worker can deviate from the prescribed algorithm and send arbitrary messages, and is assumed to have knowledge of the whole system. The second assumption indicates that Byzantine workers can collude and know the information sent by all regular workers. However, they cannot compromise the messages sent between two connected regular workers. When the communication topology is constrained and there is no trusted setup, as in the decentralized case, the known secure broadcast algorithms (Pease et al., 1980; Dolev & Strong, 1983; Hirt & Raykov, 2014) are not applicable. A Byzantine worker could even send different values to several of its regular neighbors, compromising the system (Pasquini et al., 2021).

In this work, we address the Byzantine-robustness of decentralized training in a constrained topology. The main contributions of our paper are summarized as follows:

• We identify a new information bottleneck for consensus, characterized in terms of the spectral gap of the communication graph (\(\gamma\)) and number of Byzantine workers (\(\delta\)) and empirically demonstrate that common Byzantine-robust aggregators fail to reach consensus in training.

• We propose a novel DISSENSUS attack against consensus and decentralized optimization which hones in on such information bottlenecks in the topology and amplifies existing disagreements among the workers.

• We address the above issues by proposing a novel Byzantine-robust aggregator for decentralized learning, termed Self-Centered Clipping (SCCLIP), and empirically verify its superior performance.

• We provide, for the first time, the proof of convergence to a \(O(\delta \max \zeta^2 / \gamma^2)\) neighborhood of a stationary point for non-convex objectives under standard assumptions and give the feasibility conditions in terms of \(\gamma\) and \(\delta\).

• Along the way, we also obtain the regular (non-Byzantine) fastest convergence rates for decentralized stochastic non-convex optimization, improving upon (Koloskova et al., 2020b) by using local worker momentum.

2. Related work

Recently there have been extensive works on Byzantine-resilient distributed learning with a trustworthy server. The statistics-based robust aggregation methods cover a wide...
spectrum of works including median (Chen et al., 2017; Blanchard et al., 2017; Yin et al., 2018; Mhamdi et al., 2018; Xie et al., 2018; Yin et al., 2019), geometric median (Pillutla et al., 2019), signSGD (Bernstein et al., 2019; Li et al., 2019; yong Sohn et al., 2020), clipping (Karimireddy et al., 2021a;b), and concentration filtering (Alistarh et al., 2018; Allen-Zhu et al., 2020; Data & Diggavi, 2021). If the server in addition has training dataset, then one can: 1) leverage it to score each input gradient and filter the abnormal ones (Xie et al., 2020a; Regattì et al., 2020); 2) create redundancy to achieve exact Byzantine resilience (Su & Vaidya, 2016b; Chen et al., 2018; Rajput et al., 2019; Gupta et al., 2021). The state-of-the-art attacks take advantage of the variance of good gradients and exert bias over time (Baruch et al., 2019; Xie et al., 2019). In order to provably defend against such attacks in the server case, Karimireddy et al. (2021a); El Mhamdi et al. (2021) propose to use momentum to reduce variance and Allen-Zhu et al. (2021) propose to use concentration filtering.

Byzantine-robustness is challenging when the training is combined with other constraints. For example, asynchronous distributed training allows Byzantine workers to send more gradients to the server and potentially compromise the whole system. This issue is addressed in a few works such as KARDAM (Damaskinos et al., 2018), ZENO++ (Xie et al., 2020b), BASGD (Yang & Li, 2021). The data heterogeneity makes it much harder to distinguish Byzantine gradients from good (regular) gradients. Popular solutions including bucketing-based methods (Karimireddy et al., 2021b; Peng & Ling, 2020), RSA (Li et al., 2019) and concentration filtering (Data & Diggavi, 2021). He et al. (2020); Burkhalter et al. (2021) address the issue of combining Byzantine-resilience with privacy. However, these works all assume there exists a central server which can communicate with all regular workers.

Decentralized machine learning is widely studied (Lian et al., 2017; Koloskova et al., 2020b; Li et al., 2021; Ying et al., 2021b; Lin et al., 2021; Kong et al., 2021; Yuan et al., 2021; Kovalev et al., 2021). Many works focus on compression-techniques (Koloskova et al., 2019; 2020a; Vogels et al., 2020), data heterogeneity (Tang et al., 2018; Vogels et al., 2021; Koloskova et al., 2021), and communication topology (Assran et al., 2019; Ying et al., 2021a). However, combining Byzantine robustness with decentralized machine learning is less studied. If all pairs of workers have a direct communication link, then there exists secure broadcast protocols (Pease et al., 1980; Dolev & Strong, 1983; Hirt & Raykov, 2014) which forces Byzantine workers to send same values to all workers (Gorbunov et al., 2021; El-Mhamdi et al., 2021). On the other hand, if not all workers have direct communication link (the setting we are interested in), then a Byzantine worker is more powerful since it can distort messages being passed and setup a man-in-the-middle type attack. One line of work constructs a Public-Key Infrastructure (PKI) so that the message from each worker can be authenticated using digital signatures. However, this is very inefficient requiring quadratic communication (Abraham et al., 2020). Further, it also requires every worker to have a globally unique identifier which is known to every other worker. This assumption is rendered impossible on general communication graphs, motivating our work to explicitly address the graph topology in decentralized training.

More related to the approaches we study, Su & Vaidya (2016a); Yang & Bajwa (2019b;a) use trimmed mean at each worker to aggregate models of its neighbors. This approach only works when each good worker has a honest majority among its neighbors. Guo et al. (2021) evaluate the incoming models of a good worker with its local samples and only keep those well-perform models for its local update step. However, the method only works for IID data. Peng & Ling (2020) reformulate the original problem by adding TV-regularization and propose a GossipSGD type algorithm which works for strongly convex and non-IID objectives. However, its convergence guarantees are inferior to non-parallel SGD. In this work, we address all of the above issues and are able to provably relate the communication graph (spectral gap) with the fraction of Byzantine workers.

3. Setup

3.1. Threat model

Consider an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where $\mathcal{V} = \{1, \ldots, n\}$ denotes the set of workers and $\mathcal{E}$ denotes the set of edges. Let $\mathcal{N}_i \subset \mathcal{V}$ be the neighbors of node $i$ and $\overline{\mathcal{N}}_i := \mathcal{N}_i \cup \{i\}$. In addition we assume there are no self-loops. Let $\mathcal{V}_B \subset \mathcal{V}$ be the set of Byzantine workers and the set of regular (good) workers is $\mathcal{V}_R := \mathcal{V} \setminus \mathcal{V}_B$.

Let $\mathcal{G}_R$ be the subgraph of $\mathcal{G}$ induced by the regular nodes $\mathcal{V}_R$. If the reduced graph $\mathcal{G}_R$ is disconnected, then there exist two regular workers who cannot reliably exchange information. In such a setting, the goal of training on the combined data of all the good workers is impossible. Hence, we make the following necessary assumption.

(A1) Connectivity. $\mathcal{G}_R$ is connected.

Remark 1. The reduced graph assumption in (Su & Vaidya, 2016a) not only assumes the regular workers are connected but also assume each regular worker receives models with honest majority. (Yang & Bajwa, 2019b;a) follow the same assumption as (Su & Vaidya, 2016a) and additionally assumes that the reduced graph has size of at least $b + 1$. Our work and (Peng & Ling, 2020) use (A1) which is weaker.

Note that Sybil attacks are an important orthogonal issue where a single Byzantine node can create innumerable “fake
nodes” overwhelming the network (cf. recent overview by Ford (2021)). While truly decentralized solutions to this are challenging and sometimes rely on heavy machinery such as blockchains (Poupko et al., 2021) or Proof-of-Personhood (Borge et al., 2017), in practice authentication protocols such as OAuth (Hammer-Lahav et al., 2010; Hardt et al., 2012) and Terms of Service (TOS) agreements suffice.

### 3.2. Decentralized optimization

We study the general distributed optimization problem

$$\min_{x \in \mathbb{R}^d} f(x) := \frac{1}{|\mathcal{V}_R|} \sum_{i \in \mathcal{V}_R} \{ f_i(x) := \mathbb{E}_{\xi_i \sim D_i} F_i(x; \xi_i) \}$$

on heterogeneous (non-iid) data, where $f_i$ is the local objective on worker $i$ with data distribution $D_i$ and independent noise $\xi_i$. We assume that the gradients computed over these data distributions satisfy the following properties.

**A2. Bounded noise and heterogeneity.** Assume that for all $i \in \mathcal{V}_R$ and $x \in \mathbb{R}^d$, we have

$$\mathbb{E}_{\xi_i \sim D_i} \| \nabla F_i(x; \xi_i) - \nabla f_i(x) \|^2 \leq \sigma^2$$

and

$$\mathbb{E}_{j \sim \mathcal{V}_R} \| \nabla f_j(x) - \nabla f(x) \|^2 \leq \zeta^2.$$  

**A3. L-smooth.** For $i \in \mathcal{V}_R$, $f_i(x) : \mathbb{R}^d \rightarrow \mathbb{R}$ is differentiable and there exists a constant $L \geq 0$ such that for each $x, y \in \mathbb{R}^d$,

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

Let $x_t^i \in \mathbb{R}^d$ be the state of worker $i$ at time $t$. The inter-worker communication can be modeled by schemes $x_{t+1}^i = \sum_{j=1}^{n} W_{ij} x_j^t$, where $W \in \mathbb{R}^{n \times n}$ is a non-negative mixing matrix whose positive entry $W_{ij} > 0$ for $j \in \mathcal{N}_i$. In addition, we assume

**A4.** $W$ is symmetric and doubly stochastic: $\forall i, j \in [n]$

$$W_{ij} = W_{ji}, \sum_{i=1}^{n} W_{ij} = 1, \sum_{j=1}^{n} W_{ij} = 1.$$  

Note that $W$ is determined before training and is fixed throughout the training. However, only weights between regular workers are reliable — Byzantine workers could violate $W$ in our threat model. For each regular node $i \in \mathcal{V}_R$, define $\delta_i := \sum_{j \in \mathcal{V}_R} W_{ij}$ to be the total weight of Byzantine edges around $i$. As regular nodes can trust and utilize their own updates, we have $W_{ij} > 0$ and $\delta_i \leq 1 - W_{ii}$. Therefore, we can assume that

**A5.** There exists $\delta_{\max} \in [0, 1)$ s.t. $\delta_i \leq \delta_{\max}, \forall i \in \mathcal{V}_R.$

**Remark 2.** In the decentralized setting, the total fraction of Byzantine nodes is irrelevant. Instead, what matters is the fraction of the edge weights they control which are adjacent to regular nodes (as defined by $\delta_i$ and $\delta_{\max}$). This is because a Byzantine worker can send different messages along each edge. Thus, a single Byzantine worker connected to all other workers with large edge weights can have a large influence on all the other workers. Similarly, a potentially very large number of Byzantine workers may overall have very little effect—if the edges they control towards good nodes have little weight. When we have a uniformly connected graph (such as in the centralized setting), the two notions of bad nodes & edges become equivalent.

We also define $\tilde{W} \in \mathbb{R}^{(n-b) \times (n-b)}$ for $\mathcal{G}_R$ to simulate the adversary-free communication. Its entry $i, j \in \mathcal{V}_R$ is defined by

$$\tilde{W}_{ij} = \begin{cases} W_{ij} & \text{if } i \neq j \\ W_{ii} + \delta_i & \text{if } i = j. \end{cases}$$

**Remark 3.** By the construction of $\tilde{W}$, (A4) implies that $\tilde{W}$ is also doubly stochastic. Further, by (A1), we can guarantee that $\tilde{W}$ has a positive spectral gap which we will use next.

**A6. Graph parameter.** Assume there exists $p \in [0, 1]$ such that for all $x \in \mathbb{R}^{n-b}$ and $\tilde{x} = \frac{1}{n-b} \mathbb{E}_{i \sim \mathcal{V}_R} x_i \in \mathbb{R}^{n-b}$

$$\| \tilde{W} x - \tilde{x} \|_2^2 \leq (1 - p) \| x - \tilde{x} \|_2^2.$$  

The parameter $p$ relates to the spectral gap $\gamma(W)$ of the graph as $\gamma := 1 - \sqrt{1 - p}$. In the rest of the paper we thus use $\gamma$ or $p$ interchangeably.

### 4. Robust Decentralized Consensus

One of the fundamental question in distributed computing is to achieve consensus among regular workers. First consider $\delta = 0$ and each worker holds a vector $x_i$, by recursively applying the following gossip averaging step on $i \in [n]$

$$x_{t+1}^i := \sum_{j=1}^{n} W_{ij} x_j^t, \quad t = 0, 1, \ldots$$

eventually we have for all $x_t^\infty = \tilde{x} = \frac{1}{n} \sum_{j=1}^{n} x_j^0$ for all $i \in [n]$. Such consensus is also called average consensus. Reaching consensus in the presence of Byzantine workers is much more challenging, with a long history of study (Lamport et al., 2019; Su & Vaidya, 2016a).

### 4.1. Limits due to information bottlenecks

The communication between workers can be constrained for many reasons, such as physical distances. If a subset of workers are clustered while loosely connected to the rest, then the link between two clusters become crucial for information diffusion. The existence of such bottleneck make it easier for the attackers to compromise the communication, especially when clusters are heterogeneous.
Consider two clusters A and B with n nodes each, connected by an edge to each other and to a Byzantine node, c.f. Figure 1. Suppose that we know all nodes have values in [0, 1]. Let all nodes in A have value 0. Now consider two settings:

**World 1.** All B nodes also have value 0. However, the Byzantine node pretends to be part of a cluster identical to B which it simulates, except that every node has a value 1. The true consensus average is 0.

**World 2.** All B nodes have value 1. This time the Byzantine node simulates cluster B with value 0. The true consensus average here is 0.5.

From the perspective of cluster A, the two worlds are identical—it seems to be connected to one cluster with value 0 and another with value 1. Thus, it must make $\Omega(1)$ error at least in one of the worlds. This proves that consensus is impossible in this setting. While arguments above are similar to classical lowerbounds in decentralized consensus (Fischer et al., 1986), in our case there is only 1 Byzantine node (out of 2n + 1 regular nodes) which controls only 2 edges. This impossibility result drives home the added challenge restrictive communication topology introduces.

### 4.2. Dissensus attack

In this section, inspired by our impossibility construction above, we introduce a novel DISSENSUS attack whose goal is to prevent worker models from reaching consensus. It does this by amplifying already present variance among the workers. In a gossip averaging step, regular worker i moves its value to the weighted average of its closed neighborhood. If there is no attacker, then consensus distance drop by (A6). Thus the goal of the DISSENSUS attackers around worker i is to send updates in the opposite direction of the regular neighbors of i. Then in the aggregation step, consensus distance drops slower or even grows which motivates the name “dissensus”.

We can parameterize the attack through hyperparameter $\varepsilon_i$, and summarize the attack as follows

**Definition A (DISSENSUS attack.).** Let $\varepsilon_i > 0$ for all $i \in V_{B}$. The Byzantine node $j \in N_i \cap V_{B}$ sends

$$x_{ij} := x_i - \varepsilon_i \frac{\sum_{k \in N_i \cap V_{B}} W_{ij} (x_k - x_i)}{\sum_{j \in N_i \cap V_{B}} W_{ij}}. \tag{8}$$

The $\varepsilon_i$ determines the behavior of the attack. By taking smaller $\varepsilon$, updates from Byzantine workers can be closer to the target updates i and difficult to be detected. On the other hand, a larger $\varepsilon$ pulls the model away from the consensus. Note that this attack requires omniscience since it exploits model information from across the network. If the attackers in addition can choose which node to attack, then they can choose either to spread about the attack across the network or focus on the targeting information bottleneck, that is min-cut of the graph.

**Proposition I.** (i) For all $i \in V_{B}$, under the dissensus attack with $\varepsilon_i = 1$, the gossip averaging step (7) is equivalent to no communication on i, $x_{i,t+1} = x_{i,t}$. Secondly, (ii) If the graph is fully connected, gossip averaging recovers the correct consensus even in the presence of dissensus attack.

**Proof.** For the first part, by definition (7) we know that

$$x_{i,t+1} = \sum_{j=1}^{n} W_{ij} x_{j} = x_{i,t} + \sum_{j \in N_i} W_{ij} (x_{j} - x_{i,t})$$

By setting $\varepsilon_i = 1$ in the attack (8), the second term 0 and therefore $x_{i,t+1} = x_{i,t}$. For part (ii), note that in a fully connected graph the gossip average is the same as standard average. Averaging all the pertubations introduced by the dissensus attack gives

$$-\varepsilon \sum_{i,j \in V_{B}} W_{ij} (x_{j} - x_{i,t}) = 0.$$ 

All terms cancel and sum to 0 by symmetry. Thus, in a fully connected graph the dissensus perturbations cancel out and the gossip average returns the correct consensus.

The above proposition illustrates two interesting aspects of the attack. Firstly, dissensus works by negating the progress that would be made by gossip. The attack in (Peng & Ling, 2020) also satisfies this property (see Appendix for additional discussion). Secondly, it is a uniquely decentralized attack and has no effect in the centralized setting. Hence, its effect can be used to measure the additional difficulty posed due to the restricted communication topology.

### 4.3. Decentralized defenses

Robust aggregators in the federated learning setup can be applied directly on each regular node and replace gossip.
averaging (7). Let’s take geometric median and coordinate-wise trimmed mean for example.

- **Geometric median (GM).** Pillutla et al. (2019) implements the geometric median
  \[
  \text{GM}(x_1, \ldots, x_n) := \arg \min_v \sum_{i=1}^n \|v - x_i\|_2.
  \]

- **Coordinate-wise trimmed mean (TM).** Yin et al. (2018); Yang & Bajwa (2019a) computes the \(k\)-th coordinate of TM as
  \[
  [\text{TM}(x_1, \ldots, x_n)]_k := \frac{1}{(1 - \frac{n - \beta}{n})} \sum_{i \in U_k} [x_i]_k
  \]
  where \(U_k\) is a subset of \([n]\) obtained by removing the largest and smallest \(\beta\)-fraction of its elements.

These aggregators don’t take advantage of the trusted local information and treat all vectors equally. Note that MOZI algorithm proposed in (Guo et al., 2021) does not fit in this framework and thus we discuss it in the Appendix A.2. \(^1\) A subtle additional challenge faced by previous methods is the difficulty in incorporating weights. For ease of comparison, all our graphs are symmetric with uniform edge weights.

**Our method.** Motivated by (Karimireddy et al., 2021a) for federated learning, we introduce a novel decentralized aggregator, termed **self centered clipping** (SCClip), which uses the local update as center and clips all received neighbor vectors. Formally, for CLIP \((z, \tau) := \min(1, \tau/\|z\|) \cdot z\), we define

\[
\text{SCClip}_i(x_1, \ldots, x_n) := \frac{1}{|\mathcal{V}_k|} \sum_{i \in \mathcal{V}_k} W_{ij}(x_i + \text{CLIP}(x_j - x_i, \tau_i)).
\]

**Theorem II.** Let \(\bar{x} := \frac{1}{|\mathcal{V}_k|} \sum_{i \in \mathcal{V}_k} x_i\) be the average iterate over unknown set of regular nodes and choose \(\tau_i\) to

\[
\tau_i = \sqrt{\frac{1}{\mathcal{V}_k} \sum_{j \in \mathcal{V}_k} W_{ij} E \|x_i - x_j\|_2^2}.
\]

If the initial consensus distance is bounded as

\[
\frac{1}{|\mathcal{V}_k|} \sum_{i \in \mathcal{V}_k} E \|x_i - \bar{x}\|_2^2 \leq \rho^2,
\]

then for all \(i \in \mathcal{V}_k\), SCClip outputs \(\hat{x}_i\) such that

\[
\frac{1}{|\mathcal{V}_k|} \sum_{i \in \mathcal{V}_k} E \|\hat{x}_i - \bar{x}\|_2^2 \leq (1 - \gamma + c\sqrt{\delta_{\max}})\rho^2
\]

where the expectation is over the random variable \(\{x_i\}_{i \in \mathcal{V}_k}\) and \(c > 0\) is a constant.

We inspect Theorem II on corner cases for sanity checks. If regular workers already reach consensus before aggregation \((\rho = 0)\), then \(\tau_i = 0\) chosen by (9) for all \(i \in \mathcal{V}_k\) ensures consensus after aggregation. If there is no Byzantine worker \((\delta_{\max} = 0)\), then the robust aggregator must reduce the consensus distance by a factor of \((1 - \gamma)^2 = 1 - p\) which recovers (A6). For the complete graph \((\gamma = 1)\) SCClip satisfies the centralized notion of \((\delta_{\max}, \epsilon^2)\)-robust aggregator in (Karimireddy et al., 2021a, Definition C).

More importantly, if the topology is poorly connected such that the spectral gap \(\gamma < c\sqrt{\delta_{\max}}\) then there is no guarantee that the consensus distance will reduce after aggregation. This is in line with our impossibility result in Section 4.1—if the connectivity is poor then the effect of Byzantine workers is significantly amplified.

**4.4. Simulations**

In Figure 3 show the final consensus error by three defenses to the dissensus attack with varying \(\gamma\) and \(\delta_{\max}\). The points in the figure refer to the last iterates of all experiments. The x-axis is in log scale and the y-axis measures the mean square error of the last iterate to the average consensus. The TM and MEDIAN generate same outputs in this setup. The red dashed lines characterize the output of SCClip depending on \(\delta_{\max}/\gamma^2\) where in the rightmost region SCClip perform similar to TM and MEDIAN.

\begin{itemize}
  \item \(\delta_{\max} = 0\), then the robust aggregator must reduce the consensus distance by a factor of \((1 - \gamma)^2 = 1 - p\)
  \item \((\rho = 0)\), then the Byzantine workers already reach consensus before aggregation.
\end{itemize}

\(\rho\) and large \(\gamma\). On the other hand, the right sub-figure shows that SCClip reaches consensus for \(\delta_{\max}/\gamma^2\) smaller than a threshold, and has error increasing almost linearly with \(\delta_{\max}/\gamma^2\) in log scale until \(\delta_{\max}/\gamma^2\) is too large such that no meaningful progress could be made. This observation matches with our expectation in Theorem II. This phenomenon is however not observed by looking at \(\gamma^{-2}\) or \(\delta_{\max}\) alone, validating our theoretical analysis.

**5. Robust Decentralized Optimization**

The general decentralized training can be formulated as
**Byzantine-robust decentralized learning via self-centered clipping**

**Algorithm 1** Self-Centered Clipping (SCCLIP)

Require: $x^0 \in \mathbb{R}^d$, $\alpha$, $\eta$, $\{\tau^t\}$, $m^t_i = g_i(x^t)$

1: for $t = 0, 1, \ldots$ do
2: for all $i = 1, \ldots, n$ do
3: $m^{t+1}_i = (1 - \alpha)m^{t}_i + \alpha g_i(x^t)$
4: $x^t_i + 1 = x^t_i - \eta m^{t+1}_i$ if $i \in V_\chi$ else *
5: Exchange $x^t_i + 1$ with $N_i$
6: $x^t_i + 1 = \text{SCCLIP}_i(x^t_i + 1/2, \ldots, x^t_i + 1/2; \tau_i^{t+1})$
7: end for
8: end for

$$
x^t_{i + 1/2} := \begin{cases} x^t_i - \eta g_i(x^t_i) & i \in V_\chi \\
\ast & i \in V_\gamma
\end{cases}

x^{t+1}_i := \text{AGG}_i(\{x^{t+1/2}_k : k \in N_i\})
$$

where $\eta$ is the learning rate, $g_i(x) := \nabla F(x, \xi_i)$ is the stochastic gradient, and $\xi_i \sim \mathcal{D}_i$ is the random batch at time $t$ on worker $i$. The message $x^t_i + 1/2$ can be arbitrary for Byzantine nodes $i \in V_\gamma$. Replacing AGG with (7) recovers the standard gossip SGD (Koloskova et al., 2019). In order to improve robustness of training, we can replace AGG with SCCLIP and use local worker momentum to reduce the variance $\sigma^2$ (Karimireddy et al., 2021a). The full procedure is described in Algorithm 1.

**5.1. Analysis**

**Theorem III.** Suppose Assumptions 1–6 hold and $\delta_{\text{max}} \leq \gamma^2 / (10 \cdot 2^{10})$. Define clipping radius to be

$$
\tau_i^{t+1} = \sqrt{\frac{1}{\delta_i} \sum_{j \in V_\chi} W_{ij} \mathbb{E} \left\| x^t_{i + 1/2} - x^t_{j + 1/2} \right\|^2_2}, \quad (10)
$$

By taking $\eta \leq \frac{1}{4\delta_i}$ and $\alpha := 3\eta L$, we have

$$
\frac{1}{T + 1} \sum_{t=0}^{T} \| \nabla f(x^t) \|^2_2 \leq 200^2 \gamma^2 \frac{\delta_{\text{max}}^2}{\xi^2}
+ 2 \left( \frac{3}{\delta_i} \frac{L}{T + 1} \right)^{1/4} + 2 \left( \frac{L}{T + 1} \right)^{3/4} + 3 \frac{L}{T + 1}
$$

where $r_0 := f(x^0) - f^\star$.

Note that the first term $\mathcal{O}(\delta_{\text{max}}^2 \xi^2)$ is also a lower bound (Karimireddy et al., 2021b, Theorem III). We summarize the comparison between our analysis and existing works for non-convex objectives in Table 1.

**Regular decentralized training.** If there is no Byzantine worker ($\delta_{\text{max}} = 0$), our robust convergence rate is slightly faster than that of standard gossip SGD (Koloskova et al., 2020b). The difference being out 3rd term $\mathcal{O}(\frac{\delta_{\text{max}}^2}{\gamma^2 \xi^2})$ is faster than their $\mathcal{O}(\frac{\delta_{\text{max}}^2}{\gamma^2 \xi^2})$ for large $\gamma$ and small $\epsilon$. This is because we use local momentum which reduces the effect of variance $\sigma$. Thus momentum has is doubly useful.

**Byzantine-robust federated learning.** We compare our analysis on fully connected graph $(\gamma = 1)$ with state of the art Byzantine-robust aggregator in federated learning (Karimireddy et al., 2021b). Both algorithms converge to $\Theta(\delta \xi^2)$-neighborhood of stationary point and share same leading term. We incur additional higher order terms $\mathcal{O}(\frac{\xi}{\gamma^2 \xi^2} + \frac{\sigma^2}{\gamma^2 \xi^2})$ as a penalty of the generality of our analysis. This shows that the trusted server of federated learning can be removed without significant slowdowns.

**Byzantine-robust decentralized SGD with fully connected topology.** We limit our analysis to a special case of fully connected graph $(\gamma = 1)$ and IID $(\xi = 0)$. Our rate has same leading term as (Gorbunov et al., 2021) which enjoys the scaling of total number of regular nodes. The second term $\mathcal{O}(\frac{n}{m \delta \xi^2})$ of (Gorbunov et al., 2021) is better than our $\mathcal{O}(\frac{\xi}{\gamma^2 \xi^2} + \frac{\sigma^2}{\gamma^2 \xi^2})$ for small $\epsilon$ because they additionally validates $m$ random updates after each training step. However, it relies on secure protocols which do not easily generalize to constrained communication.

**Byzantine-robust decentralized SGD with constrained communication.** Guo et al. (2021) do not provide theoretical analysis on convergence and Yang & Bajwa (2019a) only prove for full gradient, not SGD. Peng & Ling (2020) don’t prove a rate for non-convex objective; but (Gorbunov et al., 2021) which show (Peng & Ling, 2020) on strongly convex objectives is inferior to parallel SGD. In contrast, our convergence rate matches standard stochastic analysis.

Finally, we point some avenues for further improvement: our results depend on the worst-case $\delta_{\text{max}}$. We believe it is possible to replace it with a (weighted) average of the $\{\delta_i\}$ instead. Also, extending our protocols to time-varying topologies would greatly increase their practicality. Another limitation is the choice of clipping radius:

**Remark 4** (Choice of clipping radius $\tau$). The ideal $\tau_i^{t+1}$ defined in (10) cannot be computed directly due to the subset of Byzantine workers is unknown. In practice, one can choose top $\delta_{\text{max}}$ percentile of $\{\| x^t_i - x^t_j \| \}_{j \in N_i}$ to avoid additional hyperparameter tuning.

**6. Experiments**

In this section, we empirically demonstrate successes and failures of decentralized training in the presence of Byzantine workers, and compare the performance of SCCLIP with existing robust aggregators: 1) geometric median GM (Pillutla et al., 2019); 2) coordinate-wise trimmed mean TM (Yang & Bajwa, 2019a); 3) Mozi (Guo et al., 2020).
Table 1. Comparison of convergence rates for non-convex objectives. For Byzantine-robust decentralized SGD, (Peng & Ling, 2020) is excluded from comparison as they didn’t prove a rate for non-convex objectives. For our clipping-based Byzantine-robust Federated Learning, the convergence rate is provided to the $O(\delta \zeta^2)$-neighborhood of stationary point.

| Reference | Setting | Convergence to $\varepsilon$-accuracy |
|-----------|---------|--------------------------------------|
| Decentralized SGD | Regular Koloskova et al. (2020b) | $\delta = 0$ | $O\left(\frac{\sigma^2}{n\varepsilon^2} + \frac{\zeta}{\gamma \varepsilon^3} + \frac{\sigma}{\sqrt{\gamma \varepsilon^3}} + \frac{1}{\gamma \varepsilon^4}\right)$ |
| Byzantine-robust Decentralized SGD | Guo et al. (2021) | - | $O\left(\frac{\sigma^2}{n\varepsilon^2} + \frac{m \sigma^2}{n \gamma \varepsilon^3} + \frac{1}{\gamma \varepsilon^4}\right)$ |
| Byzantine-robust Federated Learning | Karimireddy et al. (2021b) | $\gamma = 1$, $\zeta = 0$ | $O\left(\frac{\sigma^2}{n^2 \varepsilon^2} + \frac{\delta \sigma^2}{n \varepsilon} + \frac{1}{\varepsilon}\right)$ |

Coordinate-wise median (Yin et al., 2018) and Krum (Blanchard et al., 2017) usually perform worse than GM so we exclude them in the experiments. For attacks, we implement our disensus along with some state of the art federated attacks Inner product manipulation (IPM) (Xie et al., 2019) and A little is enough (ALIE) (Baruch et al., 2019). More details on the adaptation of FL attacks to the decentralized setup is provided in Appendix B.²

All implementations are based on PyTorch (Paszke et al., 2019) and evaluated on different graph topologies, with a distributed MNIST dataset (LeCun & Cortes, 2010).

Topology and data distribution. The communication topology among regular workers is a “dumbbell”, c.f. Figure 1, which consists of two fully connected cliques (i.e., A and B) of equal size with an edge (i.e., “bridge”) connecting them. The Byzantine workers are added to the endpoints of the bottleneck edge to influence communication. As non-IID data distribution, we split the training dataset by labels such that workers in clique A are training on digits 0 to 4 while workers in clique B are training on digits 5 to 9. This entanglement of topology and data distribution is motivated by realistic geographic constraints such as continents with dense intra-connectivity but sparse inter-connection links e.g. through an underwater cable.

6.1. Decentralized defenses without attackers

The first experiment investigates the impact of topology and data distribution over aggregators. In Figure 4 we show the accuracy of the averaged model in clique A. Unlike the IID data setup where all aggregators perform similarly well (i.e. 1st subplot), the non-IID setup alone (i.e. 2nd subplot) leads to either slower convergence (gossip averaging, SCCLIP) or stuck at 50% accuracy (GM, TM, MOZI), even without Byzantine nodes.

²The code is available at this url.

The reason for the stark contrast between aggregator performances is that when both cliques have the same data distribution, workers in clique A do not rely on the “bridge” to access the full spectrum of data and thus the “bridge” is not the information bottleneck; when clique A and B have distinct data distributions, their sparse connection becomes the information bottleneck, slowing down gossip averaging. SCCLIP attains similar performance as gossip averaging, matching this information upper bound.

On the other hand, the low 50% accuracy attained by GM, TM, and MOZI suggest that these aggregators completely ignored the updates from clique B. This is due to receiving a many similar updates from clique A and few updates from clique B vice the sparse connection, making these aggregators falsely treat clique B updates as outliers (suspected Byzantine), thus failing to incorporate any information from B.

To address these issues, we locally employ the bucketing technique of (Karimireddy et al., 2021b) for heterogeneous data aggregation in the 3rd subplot. Plots 4 and 5 demonstrate the impact of one additional edge between the cliques to alleviate the communication bottleneck.

- The bucketing technique randomly inputs received vectors into buckets of equal size, averages the vectors in each bucket, and finally feeds the averaged vectors to the aggregator. While bucketing helps TM to overcome 50% accuracy, TM is still behind SCCLIP. GM only improves by 1% while MOZI remains at almost the same accuracy.
- Adding one more random edge between two cliques improves the spectral gap $\gamma$ from 0.0286 to 0.0154. The SCCLIP and gossip averaging converge faster as the theory predicts. However, TM, GM, and MOZI still stuck at 50% for same heterogeneity reason.
- Combining both bucketing and an additional random edge helps all aggregators achieve higher than 50% accuracy.
6.2. Effect of the number of Byzantine workers

We investigate the effect of Byzantine workers on a dumbbell topology with fixed graph parameter $\gamma$ and heterogeneity $\zeta^2$, but different $\delta_{\text{max}}$ fraction of Byzantine edges under the dissensus attack. The resulting Figure 5 shows that with increasing $\delta_{\text{max}}$ the model quality drops significantly. This is in line with our proven robust convergence rate in terms of dependency in $\delta_{\text{max}}$. It is interesting to see that for large $\delta_{\text{max}}$, the model averaged over all workers performs even worse than those averaged within cliques. It means the models in two cliques are essentially disconnected and are converging to different local minima or stationary points of a non-convex landscape. We defer details to Appendix A.2.2.

6.3. Defense without honest majority

In the decentralized environment, the common honest majority assumption in the federated learning setup can be strengthened to honest majority everywhere, meaning all regular workers have an honest majority of neighbors (Su & Vaidya, 2016b; Yang & Bajwa, 2019b,a). Considering a ring of 5 regular workers with IID data, and adding 2 Byzantine workers to each node will still satisfy the honest majority assumption everywhere. Now adding one more Byzantine worker to a node will break the assumption.

7. Discussion

In this paper, we demonstrated that data heterogeneity and information bottlenecks from the graph topology make many existing aggregators fail to converge even in absence of Byzantine workers. We then proposed decentralized Self-Centered Clipping (SCClip) to address these issues and proved its convergence under standard assumptions. In addition, we found that using bucketing aggregation for heterogeneous data and adding few random edges can empirically help existing aggregators to improve robust convergence.

A main takeaway from our work is that ill-connected communication topologies can vastly magnify the effect of bad actors. Given that decentralized consensus has been proposed as a backbone for digital democracy (Bulteau et al., 2021), and that decentralized learning is touted to be an
alternative to current centralized training paradigms, our findings are significant. A simple strategy we recommend (along with using SCCCLIP) is adding random edges in order to improve the connectivity and robustify the network.

**Acknowledgements.** This project was supported by SNSF grant 200020_200342. We are indebted to the master thesis of Cappelletti (2021) for initiating this project and analyzing the Byzantine-free setting. We also thank Anastasiia Koloskova and Lê Nguyên Hoang for fruitful discussions on optimization and authentication.

**Bibliography**

Abraham, I., Chan, T.-H. H., Dolev, D., Nayah, K., Pass, R., Ren, L., and Shi, E. Communication complexity of byzantine agreement, revisited, 2020.

Alistarh, D., Allen-Zhu, Z., and Li, J. Byzantine stochastic gradient descent. In Bengio, S., Wallach, H. M., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018*, NeurIPS 2018, December 3-8, 2018, Montréal, Canada, pp. 4618–4628. 2018. URL https://proceedings.neurips.cc/paper/2018/hash/a07c2f3b3907aaf8436a26cc6d77f0a2-Abstract.html.

Allen-Zhu, Z., Ebrahimian, F., Li, J., and Alistarh, D. Byzantine-resilient non-convex stochastic gradient descent. *arXiv preprint arXiv:2012.14368*, 2020.

Allen-Zhu, Z., Ebrahimianghazani, F., Li, J., and Alistarh, D. Byzantine-resilient non-convex stochastic gradient descent. In *9th International Conference on Learning Representations, ICLR 2021*, Virtual Event, Austria, May 3-7, 2021. OpenReview.net, 2021. URL https://openreview.net/forum?id=PbEHqVFtcS.

Assran, M., Loizou, N., Ballas, N., and Rabbat, M. Stochastic gradient push for distributed deep learning, 2019.

Baruch, G., Baruch, M., and Goldberg, Y. A little is enough: Circumventing defenses for distributed learning. In Wallach, H. M., Larochelle, H., Beygelzimer, A., d’Alché-Buc, F., Fox, E. B., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019*, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada, pp. 8632–8642, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/e1c59141046cd1866bbbcdf6ae31d4-Abstract.html.

Bernstein, J., Zhao, J., Azizzadenesheli, K., and Anandkumar, A. signedsgd with majority vote is communication efficient and fault tolerant, 2019.

Blanchard, P., Mhamdi, E. M. E., Guerraoui, R., and Stainer, J. Machine learning with adversaries: Byzantine tolerant gradient descent. In Guyon, I., von Luxburg, U., Bengio, S., Wallach, H. M., Fergus, R., Vishwanathan, S. V. N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017*, December 4-9, 2017, Long Beach, CA, USA, pp. 119–129, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/f4b9ec30ad9f68f88b29639786cb62ef-Abstract.html.

Borge, M., Kokoris-Kogias, E., Jovanovic, P., Gasser, L., Guilly, N., and Ford, B. Proof-of-personhood: Redemocratizing permissionless cryptocurrencies. In 2017 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW), pp. 23–26. IEEE, 2017.

Bultheau, L., Shahaf, G., Shapiro, E., and Talmon, N. Aggregation over metric spaces: Proposing and voting in elections, budgeting, and legislation. *Journal of Artificial Intelligence Research*, 70:1413–1439, 2021.

Burkhalter, L., Lycklama, H., Vian, A., Küchler, N., and Hithnawi, A. Roll: Attestable robustness for secure federated learning, 2021.

Cappelletti, W. Byzantine-robust decentralized optimization for Machine Learning. Master Thesis, EPFL, https://williamcappelletti.github.io/assets/pdf/Cappelletti-ByzantineRobustDeSGD.pdf, 2021.

Chen, L., Wang, H., Charles, Z. B., and Papailiopoulos, D. S. DRACO: Byzantine-resilient distributed training via redundant gradients. In Dy, J. G. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning, ICML 2018*, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018, volume 80 of *Proceedings of Machine Learning Research*, pp. 902–911. PMLR, 2018. URL http://proceedings.mlr.press/v80/chench21.html.

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

Damaskinos, G., Mhamdi, E. M. E., Guerraoui, R., Patra, R., and Taziki, M. Asynchronous byzantine machine learning (the case of SGD). In Dy, J. G. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning, ICML 2018*, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018, volume 80 of *Proceedings of Machine Learning Research*, pp. 1153–1162. PMLR, 2018. URL http://proceedings.mlr.press/v80/damaskinos18a.html.

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

Dolev, D. and Strong, H. R. Authenticated algorithms for byzantine agreement revisited, 2020.

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

El Mhamdi, E. M., Guerraoui, R., and Rouault, S. L. A. Distributed momentum for Byzantine-resilient stochastic gradient descent. In *9th International Conference on Learning Representations (ICLR)*, number CONF, 2021.
Byzantine-robust decentralized learning via self-centered clipping

Fischer, M. J., Lynch, N. A., and Merritt, M. Easy impossibility proofs for distributed consensus problems. *Distributed Computing*, 1(1):26–39, 1986.

Ford, B. 10. technologizing democracy or democratizing technology? a layered-architecture perspective on potentials and challenges. In *Digital Technology and Democratic Theory*, pp. 274–321. University of Chicago Press, 2021.

Gorbunov, E., Borzunov, A., Diskin, M., and Ryabinin, M. Secure distributed training at scale, 2021.

Guo, S., Zhang, T., Xie, X., Ma, L., Xiang, T., and Liu, Y. Byzantine-resilient decentralized stochastic gradient descent, 2021.

Hastings, W. K. Monte carlo sampling methods using markov chains and their applications. 1970.

He, L., Karimireddy, S. P., and Jaggi, M. Secure byzantine-robust machine learning. *arXiv* 2006.04747, 2020.

Hammer-Lahav, E., Recordon, D., and Hardt, D. The oauth 1.0 protocol. Technical report, RFC 5849, April, 2010.

Hardt, D. et al. The oauth 2.0 authorization framework, 2012.

Koloskova, A., Lin, T., Stich, S. U., and Jaggi, M. Decentralized deep learning with arbitrary communication compression, 2020a.

Koloskova, A., Loizou, N., Boreiri, S., Jaggi, M., and Stich, S. U. A unified theory of decentralized SGD with changing topology and local updates. In *Proceedings of the 37th International Conference on Machine Learning*, ICML 2020, 13-18 July 2020, Virtual Event, volume 119 of *Proceedings of Machine Learning Research*, pp. 5381–5393. PMLR, 2020b. URL: http://proceedings.mlr.press/v119/koloskova20a.html.

Koloskova, A., Lin, T., and Stich, S. U. An improved analysis of gradient tracking for decentralized machine learning. *Advances in Neural Information Processing Systems*, 34, 2021.

Kong, L., Lin, T., Koloskova, A., Jaggi, M., and Stich, S. U. Consensus control for decentralized deep learning. *arXiv preprint arXiv:2102.04828*, 2021.

Kovalev, D., Koloskova, A., Jaggi, M., Richtarik, P., and Stich, S. A linearly convergent algorithm for decentralized optimization: Sending less bits for free! In *International Conference on Artificial Intelligence and Statistics*, pp. 4087–4095. PMLR, 2021.

Lampert, L., Shostak, R., and Pease, M. The byzantine generals problem. In *Concurrence: the Works of Leslie Lamport*, pp. 203–226, 2019.

LeCun, Y. and Cortes, C. MNIST handwritten digit database. 2010. URL: http://yann.lecun.com/exdb/mnist/.

Li, L., Xu, W., Chen, T., Giannakis, G. B., and Ling, Q. RSA: Byzantine-robust stochastic aggregation methods for distributed learning from heterogeneous datasets. In *The Thirty-Third AAAI Conference on Artificial Intelligence*, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 – February 1, 2019, pp. 1544–1551. AAAI Press, 2019. doi: 10.1609/aaai.v33i01.33011544. URL: https://doi.org/10.1609/aaai.v33i01.33011544.

Li, X., Yang, W., Wang, S., and Zhang, Z. Communication-efficient local decentralized sgd methods, 2021.

Lian, X., Zhang, C., Zhang, H., Hsieh, C.-J., Zhang, W., and Liu, J. Can decentralized algorithms outperform centralized algorithms? A case study for decentralized parallel stochastic gradient descent, 2017.

Lin, T., Karimireddy, S. P., Stich, S. U., and Jaggi, M. Quasi-global momentum: accelerating decentralized deep learning on heterogeneous data. *arXiv preprint arXiv:2102.04761*, 2021.

McMahan, B., Moore, E., Ramage, D., Hampson, S., and y Arcas, B. A. Communication-efficient learning of deep networks from decentralized data. In Singh, A. and Zhu, X. J. (eds.), *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, AISTATS 2017, 20-22 April 2017, Fort Lauderdale, FL, USA, volume 54 of *Proceedings of Machine Learning Research*, pp. 1273–1282. PMLR, 2017. URL: http://proceedings.mlr.press/v54/mcmahan17a.html.
**Byzantine-robust decentralized learning via self-centered clipping**

Mhamdi, E. M. E., Guerraoui, R., and Rouault, S. The hidden vulnerability of distributed learning in byzantium. In Dy, J. G. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning*, ICML 2018. Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018, volume 80 of *Proceedings of Machine Learning Research*, pp. 3518–3527. PMLR. 2018. URL: http://proceedings.mlr.press/v80/mhamdi18a.html.

Nedic, A. Distributed gradient methods for convex machine learning problems in networks: Distributed optimization. *IEEE Signal Processing Magazine*, 37(3):92–101, 2020.

Pasquini, D., Francati, D., and Ateniese, G. Eluding secure aggregation in federated learning via model inconsistency, 2021.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, A., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. PyTorch: An imperative style, high-performance deep learning library. In Wallach, H. M., Larochelle, H., Beygelzimer, A., d’Alché-Buc, F., Fox, E. B., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019*, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada, pp. 8024–8035, 2019. URL: https://proceedings.neurips.cc/paper/2019/hash/bdca288fee7f92f2bfa9f7012727740-Abstract.html.

Pease, M. C., Shostak, R. E., and Lamport, L. Reaching agreement in the presence of faults. *J. ACM*, 27:228–234, 1980.

Peng, J. and Ling, Q. Byzantine-robust decentralized stochastic optimization. In 2020 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2020, Barcelona, Spain, May 4-8, 2020, pp. 5935–5939. IEEE. 2020. doi: 10.1109/ICASSP40776.2020.9054377. URL: https://doi.org/10.1109/ICASSP40776.2020.9054377.

Pillutla, K., Kakade, S. M., and Harchaoui, Z. Robust aggregation for federated learning. *arXiv preprint arXiv:1912.13445*, 2019.

Poupko, O., Shahaf, G., Shapiro, E., and Talmon, N. Building a sybil-resilient digital community utilizing trust-graph connectivity. *IEEE/ACM Transactions on Networking*, 2021.

Rajput, S., Wang, H., Charles, Z. B., and Papailiopoulos, D. S. DETOX: A redundancy-based framework for faster and more robust gradient aggregation. In Wallach, H. M., Larochelle, H., Beygelzimer, A., d’Alché-Buc, F., Fox, E. B., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019*, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada, pp. 10320–10330, 2019. URL: https://proceedings.neurips.cc/paper/2019/hash/415185ea244ea2b2bedeb0449b926802-Abstract.html.

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

Su, L. and Vaidya, N. H. Robust multi-agent optimization: coping with byzantine agents with input redundancy. In *International Symposium on Stabilization, Safety, and Security of Distributed Systems*, pp. 368–382. Springer, 2016b.

Tang, H., Lian, X., Yan, M., Zhang, C., and Liu, J. D\textsuperscript{2}: Decentralized training over decentralized data, 2018.

Vogels, T., Karimireddy, S. P., and Jaggi, M. Practical low-rank communication compression in decentralized deep learning. In *NeurIPS*, 2020.

Vogels, T., He, L., Koloskova, A., Lin, T., Karimireddy, S. P., Stich, S. U., and Jaggi, M. Relaysum for decentralized deep learning on heterogeneous data, 2021.

Xie, C., Koyejo, O., and Gupta, I. Generalized byzantine-tolerant sgd. *arXiv 1802.10116*, 2018.

Xie, C., Koyejo, O., and Gupta, I. Fall of empires: Breaking byzantine-tolerant SGD by inner product manipulation. In Globerson, A. and Silva, R. (eds.), *Proceedings of the Thirty-Fifth Conference on Uncertainty in Artificial Intelligence*, UAI 2019, Tel Aviv, Israel, July 22-25, 2019, volume 115 of *Proceedings of Machine Learning Research*, pp. 261–270. AUAI Press, 2019. URL: http://proceedings.mlr.press/v115/xie20a.html.

Xie, C., Koyejo, S., and Gupta, I. Zeno++: Robust fully asynchronous SGD. In *Proceedings of the 37th International Conference on Machine Learning*, ICML 2020, 13-18 July 2020, Virtual Event, volume 119 of *Proceedings of Machine Learning Research*, pp. 10495–10503. PMLR, 2020a. URL: http://proceedings.mlr.press/v119/xie20c.html.

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

Yang, Y. and Li, W. BASGD: buffered asynchronous SGD for byzantine learning. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning*, ICML 2021, 18-24 July 2021, Virtual Event, volume 139 of *Proceedings of Machine Learning Research*, pp. 11751–11761. PMLR, 2021. URL: http://proceedings.mlr.press/v139/yang21e.html.

Yang, Z. and Bajwa, W. U. Bridge: Byzantine-robust decentralized gradient descent. *arXiv 1908.08098*, 2019a.

Yang, Z. and Bajwa, W. U. Byrdie: Byzantine-robust distributed coordinate descent for decentralized learning. *IEEE Transactions on Signal and Information Processing over Networks*, 2019b.

Yin, D., Chen, Y., Ramchandran, K., and Bartlett, P. L. Byzantine-robust distributed learning: Towards optimal statistical rates. In Dy, J. G. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning*, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018, volume 80 of *Proceedings of Machine Learning Research*, pp. 5636–5645. PMLR, 2018. URL: http://proceedings.mlr.press/v80/yin18a.html.

Yin, D., Chen, Y., Kannan, R., and Bartlett, P. Defending against saddle point attack in byzantine-robust distributed learning. In *International Conference on Machine Learning*, pp. 7074–7084. PMLR, 2019.
Ying, B., Yuan, K., Chen, Y., Hu, H., Pan, P., and Yin, W. Exponential graph is provably efficient for decentralized deep training. *Advances in Neural Information Processing Systems*, 34, 2021a.

Ying, B., Yuan, K., Hu, H., Chen, Y., and Yin, W. Bluefog: Make decentralized algorithms practical for optimization and deep learning. *arXiv preprint arXiv:2111.04287*, 2021b.

Yong Sohn, J., Han, D.-J., Choi, B., and Moon, J. Election coding for distributed learning: Protecting signsgd against byzantine attacks, 2020.

Yuan, K., Chen, Y., Huang, X., Zhang, Y., Pan, P., Xu, Y., and Yin, W. Decentlam: Decentralized momentum sgd for large-batch deep training. *arXiv preprint arXiv:2104.11981*, 2021.
Appendices

A. Experiments

We list the setups and results of experiments for consensus in Appendix A.1 and optimization in Appendix A.2.

A.1. Attacks against consensus

In this section, we provide detailed setups for Figure 3. The Figure 7 demonstrates the topology for the experiment. The 4 regular workers are connected with two of them holding value 0 and the others holding 200. Then the average consensus is 100 with initial mean square error equals 10000. Two Byzantine workers are connected to two regular workers in the middle. We can tune the weights of each edge to change the mixing matrix and $\gamma$. Then we can decide the weight $\delta$ on the Byzantine edge. The set of $\gamma$ and $\delta$ used

- $p = 1 - (1 - \gamma)^2 \in [0.06, 0.05, 0.04, 0.03, 0.02, 0.01, 0.005, 0.0014, 0.00037, 0.0001, 0.000001]$
- $\delta \in [0.05, 0.1, 0.2, 0.3, 0.4, 0.5]$

where non-compatible combination of $\gamma$ and $\delta$ are ignored in the Figure 3. The dissensus attack is applied with $\varepsilon = 0.05$. The hyperparameter $\beta$ of trimmed mean (TM) is set to the actual number of Byzantine workers around the regular worker. The clipping radius of SCCLIP is chosen according to (10).

![Figure 7. The topology for the attacks on consensus. Black workers are regular while red ones are Byzantine.](image)

In Figure 8, we show the iteration-to-error curves for all possible combinations of $\gamma$ and $\delta$. In addition, we provide a version of TM and MEDIAN which takes the mixing weight into account. As we can see, the naive TM, MEDIAN, and MEDIAN* cannot bring workers closer because of the data distribution we constructed. The TM* is performing better than the other baselines but worse than SCCLIP especially on the challenging cases where $\gamma$ is small and $\delta$ is large. For SCCLIP, it matches with our intuition that for a fixed $\gamma$ the convergences is worse with increasing $\delta$ while for a fixed $\delta$ the convergence is worse with decreasing $\gamma$.

A.2. Challenges of decentralized optimization

The default experiment setup is listed in Table 2 and per-experiment setup in Table 3. The default hyperparameters of the aggregators are summarized as follows

| Aggregators | Hyperparameters |
|-------------|-----------------|
| GM          | $T = 8$         |
| TM          | $\beta = \delta_{\text{max}}$ |
| SCCLIP      | $\tau = 1$     |
| MOZI        | $\alpha = 0.5$ where $\alpha$ is the model averaging hyperparameter |

We summarize the running environment of this paper as in Table 4.
Figure 8. The iteration-to-error curves for defenses under dissensus attack. The TM* and MEDIAN* refer to the version of TM and MEDIAN which considers mixing weight.

**Mixing matrix.** Let $d_{\text{max}}$ be the maximum degree of nodes in a graph. We use the following naive construction

\[
W_{ij} = \begin{cases} 
\frac{1}{d_{\text{max}} + 1} & j \in \mathcal{N}_i, \\
1 - \frac{|\mathcal{N}_i|}{d_{\text{max}} + 1} & j = i, \\
0 & \text{Otherwise}.
\end{cases}
\]  

(11)

Note that this construction of mixing matrix usually have worse spectral gap than the common Metropolis-Hastings weight (Hastings, 1970). However, the Metropolis-Hastings give different weight to each edge based on the degree of each node which is not justified in the Byzantine-resilient sense. We leave the study of better mixing matrix to future work.

**Mozi.** Guo et al. (2021) applies two screening steps on worker $i \in \mathcal{V}_R$

\[
\mathcal{N}_i^* := \arg \min_{\mathcal{N}_i^* \subseteq \mathcal{N}_i} \sum_{j \in \mathcal{N}_i^*} \| x_i - x_j \|
\]

\[
\mathcal{N}_i^+ := \mathcal{N}_i^* \cap \{ j \in [n] : \ell(x_j, \xi_i) \leq \ell(x_i, \xi_i) \}
\]

where $\xi_i \sim \mathcal{D}_i$ is a random sample. If $\mathcal{N}_i^+ = \emptyset$, then redefine $\mathcal{N}_i^+ := \{ \arg \min_j \ell(x_j, \xi_i) \}$. Then they update the model with

\[
x_{i+1} := \alpha x_i + \frac{1-\alpha}{|\mathcal{N}_i^+|} \sum_{j \in \mathcal{N}_i^+} x_j^i - \eta \nabla F_i(x_i^i; \xi_i^i)
\]

where $\alpha \in [0, 1]$ is an hyperparameter.
Table 2. Default experimental settings for MNIST

| Dataset       | MNIST          |
|---------------|----------------|
| Architecture  | CONV-CONV-DROPOUT-FC-DROPOUT-FC |
| Training obj. | Negative log likelihood loss |
| Eval. obj.    | Top-1 accuracy |
| Batch size    | 32             |
| Momentum      | 0.9            |
| Learn. rate   | 0.01           |
| LR decay      | No             |
| LR warmup     | No             |
| Weight decay  | No             |
| Repetitions   | 1              |
| Reported metric | Mean test accuracy over the last 150 iterations |

Table 3. Setups for each optimization experiment. The learning rate $\eta = 0.01$ throughout all experiments and momentum $\alpha = 0.9$.

| Topology | n   | b      | #Iters | Results  |
|----------|-----|--------|--------|----------|
| Exp 1    | Fig. 1 | 20  | 0  | 900 | Fig. 4 |
| Exp 2    | Fig. 9 | 12 | 1  | 1500 | Fig. 5 |
| Exp 3    | Fig. 10 | 16/15 | 11/10 | 900 | Fig. 6 |

Table 4. Runtime hardwares and softwares.

| CPU          | Intel (R) Xeon (R) Gold 6132 CPU @ 2.60 GHz |
|--------------|---------------------------------------------|
| # CPU(s)     | 56                                          |
| NUMA node(s) | 2                                           |
| GPU          | Tesla V100-SXM2-32GB                         |
| CUDA Version | 11.0                                        |
| PyTorch      | 1.7.1                                       |

A.2.1. EXP 1: DECENTRALIZED DEFENSES WITHOUT ATTACKERS

| Aggregators | Hyperparameters |
|-------------|-----------------|
| GM          | $T = 8$         |
| TM          | $\beta = \delta_{\text{max}}$ |
| SCCLIP      | $\tau = 1$     |
| MOZI        | $\alpha = 0.5$ where $\alpha$ is the model averaging hyperparameter. $\rho_i = 0.99$ |
| Bucketing   | Each bucket holds at most $s = 2$ vectors |

A.2.2. EXP 2: EFFECTS OF THE NUMBER OF BYZANTINE WORKERS

In this experiment we choose $n - b = 11$ and $b = 0, 1, 2, 3$. Therefore their $\tilde{W}$ and $p$ remain the same for all these $b$. Then we can easily investigate the relation between $\delta_{\text{max}} \in [0, \frac{b}{b+2}]$ and $p$ by varying $b$. 
Byzantine-robust decentralized learning via self-centered clipping

| Attack       | Hyperparameters |
|--------------|-----------------|
| DISSENSUS    | $\varepsilon = 1.5$ |
| Aggregators  | Hyperparameters |
| SCCLIP       | $\tau = 1$ |

Figure 9. Example of variant of dumbbell topology which consists of two cliques A and B (black) connected by an edge.

A.2.3. EXP 3: DEFENSE WITHOUT HONEST MAJORITY

| Attack   | Hyperparameters |
|----------|-----------------|
| DISSENSUS| $\varepsilon = 1.5$ |
| IPM      | $\varepsilon = 0.1$ |
| ALIE     | Determined in (12) |

| Aggregators | Hyperparameters |
|-------------|-----------------|
| GM          | $T = 8$         |
| TM          | $\beta = \delta_{\text{max}}$ |
| SCCLIP      | $\tau = 0.1$ |
| MOZI        | $\alpha = 0.5$ where $\alpha$ is the model averaging hyperparameter, $\rho_i = 0.4$ |

Figure 10. Ring topology used in experiment 3. Note that if the edge with question mark exists, then we don’t have honest majority.

B. Attacks in the decentralized environment

In this section, we describe attacks in the centralized environment and further explain how to transform them into attacks in the decentralized environment.
A little is enough (ALIE): The attackers estimate the mean $\mu_N$, and standard deviation $\sigma_N$, of the regular models, and send $\mu_N - z\sigma_N$ to regular worker $i$ where $z$ is a small constant controlling the strength of the attack (Baruch et al., 2019). The hyperparameter $z$ for ALIE is computed according to (Baruch et al., 2019)

$$z = \max_z \left( \phi(z) < \frac{n - b - s}{n - b} \right)$$

where $s = \lfloor \frac{n}{2} + 1 \rfloor - b$ and $\phi$ is the cumulative standard normal function.

**Inner product manipulation attack.** The inner product manipulation (IPM) attack is proposed in (Xie et al., 2019) which lets all attackers send same corrupted gradient $u$ based on the good gradients

$$u_j = -\varepsilon \text{Avg} (\{v_i : i \in V_k\}) \quad \forall j \in V_b.$$ 

If $\varepsilon$ is small enough, then $u_j$ can be detected as good by the defense, thus circumventing the defense. In contrast, there are 3 main differences when we conduct attacks in the decentralized environment:

1. Byzantine workers may not connected to the same good worker.
2. The model vectors are transmitted instead of gradients.
3. The Avg should be replaced by its equivalent gossip form.

Therefore, each Byzantine worker should compute their own corrupted model updates. More specifically, for $j \in V_b$ and $j' \in V_R$ be the only neighbor of $j$, then

$$u_j = v_j' - \varepsilon \sum_{i \in V_b \cap N_i} W_{j'i}(v_i - v_j').$$

Then the gossip average at node $j'$ gives

$$\sum_{i \in V_b \cap N_i} W_{j'i}v_i + \sum_{j \in V_b \cap N_i} W_{j'j}u_j = \sum_{i \in V_b \cap N_i} W_{j'i}v_i + \left( v_j' - \varepsilon \sum_{i \in V_b \cap N_i} W_{j'i}(v_i - v_j') \right) \sum_{j \in V_b \cap N_i} W_{j'j}$$

$$= v_j' + \left( \sum_{i \in V_b \cap N_i} W_{j'i}(v_i - v_j') \right) \left( 1 - \varepsilon \sum_{j \in V_b \cap N_i} W_{j'j} \right)$$

Note that

$$v_j' + \text{GOSSIP AVG}_{j'}(\{v_i - v_j' : i \in V_R \cap N_i\}).$$

**Relation with zero-sum attack and disensus.** Peng & Ling (2020) propose the “zero-sum” attack which achieves similar effects as Proposition I part (i). This attack is defined for $j \in V_b$

$$x_j := -\frac{\sum_{k \in V_b \cap V_R} x_k}{|V_b \cap V_R|}.$$ 

The key difference between zero-sum attack and our proposed attack is three-folded. First, zero-sum attack ensures $\sum_{j \in N_i} x_j = 0$ which means the Byzantine models have to be far away from $x_i^t$ and therefore easy to detect. This attack pull the aggregated model to 0. On the other hand, our attack ensures

$$\frac{1}{\sum_{j \in N_i} W_{ij}} \sum_{j \in N_i} W_{ij} x_j = x_i^t$$

and the Byzantine updates can be very close to $x_i^t$ and it is more difficult to be detected. Second, our proposed attack considers the gossip averaging which is prevalent in decentralized training (Koloskova et al., 2020b) while the zero-sum attack only targets simple average. Third, our attack has an additional parameter $\varepsilon$ controlling the strength of the attack with $\varepsilon > 1$ further compromise the model quality while zero-sum attack is fixed to training alone.
C. Analysis

Let us restate the core equations in Algorithm 1 at time $t$ on worker $i$

$$m_{i}^{t+1} = (1 - \alpha)m_{i}^{t} + \alpha g_{i}(x_{i}^{t})$$  \hspace{1cm} (13)
$$x_{i}^{t+1/2} = x_{i}^{t} - \eta m_{i}^{t+1}$$  \hspace{1cm} (14)
$$z_{j ightarrow i}^{t+1} = x_{i}^{t+1/2} + \text{CLIP}(x_{j}^{t+1/2} - x_{i}^{t+1/2}, \tau_{i})$$  \hspace{1cm} (15)
$$x_{i}^{t+1} = \sum_{j=1}^{n} W_{ij} z_{j ightarrow i}^{t+1}$$  \hspace{1cm} (16)

In addition, we define the following virtual iterates on the set of good nodes $V_{R}$

- $\bar{x}^{t} = \frac{1}{|V_{R}|} \sum_{i \in V_{R}} x_{i}^{t}$ the average of good iterates.
- $\bar{m}^{t} = \frac{1}{|V_{R}|} \sum_{i \in V_{R}} m_{i}^{t}$ the average of momentum iterates.

In this section, we show that the convergence behavior of the virtual iterates $\bar{x}^{t}$. The structure of this section is as follows:

- In Appendix C.1, we give common quantities, simplified notations and list common equalities/inequalities used in the proof.
- In Appendix C.2, we provide all auxiliary lemmas necessary for the proof. Among these lemmas, Lemma 7 is the key sufficient descent lemma.
- In Appendix C.3, we provide the proof of the main theorem.

C.1. Definitions, and inequalities

Notations for the proof. We use the following variables to simplify the notation

- Sub-optimality:
  $$r^{t} := f(\bar{x}^{t}) - f^{*}$$

- Consensus distance:
  $$\Xi^{t} := \frac{1}{|V_{R}|} \sum_{i \in V_{R}} \|x_{i}^{t} - \bar{x}^{t}\|_{2}^{2}$$

- The distance between the idea gradient and actual averaged momentum
  $$e_{1}^{t+1} := \mathbb{E} \|\nabla f(\bar{x}^{t}) - \bar{m}^{t+1}\|_{2}^{2}$$

- Similar distance between the idea gradient and individual momentum
  $$\tilde{e}_{1}^{t+1} := \frac{1}{|V_{R}|} \sum_{i \in V_{R}} \mathbb{E} \|\nabla f(\bar{x}^{t}) - m_{i}^{t+1}\|_{2}^{2}$$

- Similar distance between the idea gradient and individual momentum
  $$\tilde{e}_{1}^{t+1} := \frac{1}{|V_{R}|} \sum_{i \in V_{R}} \mathbb{E} \|\sum_{j \in V_{R}} \tilde{W}_{ij}(\nabla f_{j}(\bar{x}^{t}) - m_{j}^{t+1})\|_{2}^{2}$$

- Similar we have
  $$e_{I}^{t+1} := \frac{1}{|V_{R}|} \sum_{i \in V_{R}} \mathbb{E} \|m_{i}^{t+1} - \nabla f_{i}(\bar{x}^{t})\|_{2}^{2}.$$
Then the virtual iterate updates

\[ e_2^{t+1} := \frac{1}{|V_k|} \sum_{i \in V_k} \mathbb{E} \left[ \left( \sum_{j \in V_k} W_{ij} (z_{j \rightarrow i}^{t+1} - \bar{x}_{i}^{t+1/2}) + \sum_{j \in V_a} W_{ij} (z_{j \rightarrow i}^{t+1} - x_{i}^{t+1/2}) \right)^2 \right]. \]

Lemma 5 (Common equalities and inequalities). We use the following equalities and inequalities

- The cosine theorem: \( \forall \, x, y \in \mathbb{R}^d \)
  \[
  \langle x, y \rangle = -\frac{1}{2} \|x - y\|_2^2 + \frac{1}{2} \|x\|_2^2 + \frac{1}{2} \|y\|_2^2
  \]
  (17)

- Young’s inequality: For \( \varepsilon > 0 \) and \( x, y \in \mathbb{R}^d \)
  \[
  \|x + y\|_2^2 \leq (1 + \varepsilon)\|x\|_2^2 + (1 + \varepsilon^{-1})\|y\|_2^2
  \]
  (18)

- If \( f \) is convex, then for \( \alpha \in [0, 1] \) and \( x, y \in \mathbb{R}^d \)
  \[
  f(\alpha x + (1 - \alpha)y) \leq \alpha f(x) + (1 - \alpha)f(y)
  \]
  (19)

- Cauchy-Schwarz inequality
  \[
  \langle x, y \rangle \leq \|x\|_2 \|y\|_2
  \]
  (20)

- Let \( \{x_i : i \in [m]\} \) be independent random variables and \( \mathbb{E} x_i = 0 \) and \( \mathbb{E} \|x_i\|^2 = \sigma^2 \) then
  \[
  \mathbb{E} \left[ \frac{1}{m} \sum_{i=1}^{m} x_i \right]^2 = \frac{\sigma^2}{m}
  \]
  (21)

C.2. Lemmas

The following lemma establish the update rule for \( x^t \).

Lemma 6. Assume (A4). Let \( \Delta^{t+1} \) be the error incurred by clipping and \( V_a \)

\[
\Delta^{t+1} := \frac{1}{|V_k|} \sum_{i \in V_k} \left( \sum_{j \in V_k} W_{ij} (z_{j \rightarrow i}^{t+1} - \bar{x}_{i}^{t+1/2}) + \sum_{j \in V_a} W_{ij} (z_{j \rightarrow i}^{t+1} - x_{i}^{t+1/2}) \right).
\]

(22)

Then the virtual iterate updates

\[
x^{t+1} = x^t - \eta \hat{m}^{t+1} + \Delta^{t+1}.
\]

(23)

Proof. Expand \( x^{t+1} \) with the definition of \( x_i^{t+1} \) in (16) yields

\[
x^{t+1} = \frac{1}{|V_k|} \sum_{i \in V_k} x_i^{t+1} = \frac{1}{|V_k|} \sum_{i \in V_k} \left( \sum_{j \in V_k} W_{ij} z_{j \rightarrow i}^{t+1} + \sum_{j \in V_a} W_{ij} z_{j \rightarrow i}^{t+1} \right)
\]
\[
\quad + \frac{1}{|V_k|} \sum_{i \in V_k} \left( \sum_{j \in V_k} W_{ij} (z_{j \rightarrow i}^{t+1} - x_{i}^{t+1/2}) + \sum_{j \in V_a} W_{ij} z_{j \rightarrow i}^{t+1} \right)
\]
\[
\quad + \frac{1}{|V_k|} \sum_{i \in V_k} \left( \sum_{j \in V_a} W_{ij} (z_{j \rightarrow i}^{t+1} - x_{i}^{t+1/2}) + \sum_{j \in V_a} W_{ij} x_{i}^{t+1/2} \right).
\]

Reorganize the terms to form $\Delta^{t+1}$

$$
\bar{x}^{t+1} = \frac{1}{\sum_{i \in \mathcal{V}_k}} \sum_{i \in \mathcal{V}_k} \left( \sum_{j \in \mathcal{V}_k} W_{ij} x_j^{t+1/2} + \sum_{j \in \mathcal{V}_k} W_{ij} x_i^{t+1/2} \right) + \Delta^{t+1}
$$

$$
= \frac{1}{\sum_{i \in \mathcal{V}_k}} \sum_{i \in \mathcal{V}_k} \left( (1 - \delta_j) x_j^{t+1/2} + \frac{1}{\sum_{i \in \mathcal{V}_k}} \delta_j x_i^{t+1/2} + \Delta^{t+1} \right)
$$

$$
= \frac{1}{\sum_{i \in \mathcal{V}_k}} \sum_{i \in \mathcal{V}_k} x_i^{t+1/2} + \Delta^{t+1} = \frac{1}{\sum_{i \in \mathcal{V}_k}} \sum_{i \in \mathcal{V}_k} \left( x_i^{t} - \eta m_i^{t+1} \right) + \Delta^{t+1}
$$

$$
= x^{t} - \eta \bar{m}^{t+1} + \Delta^{t+1}. \tag{23}
$$

Note that the $\Delta^{t+1}$ can be written as the follows

$$
\Delta^{t+1} = \frac{1}{|\mathcal{V}_k|} \sum_{i \in \mathcal{V}_k} \left( x_i^{t+1} - \sum_{j \in \mathcal{V}_k} W_{ij} x_j^{t+1/2} \right) = x^{t+1} - \frac{1}{|\mathcal{V}_k|} \sum_{i \in \mathcal{V}_k} x_i^{t+1/2}.
$$

where measures the error introduced to $\bar{x}^{t+1}$ considering the impact of Byzantine workers and clipping. Therefore when $|\mathcal{V}_y| = 0$ and $\tau$ is sufficiently large, $\Delta^{t+1} = 0$ and $\bar{x}^{t+1}$ converge at the same rate as the centralized SGD with momentum.

Recall that $e_1^{t+1} := \mathbb{E}[\|\nabla f(\bar{x}^t) - \bar{m}^{t+1}\|^2]$. The key descent lemma is stated as follow

**Lemma 7 (Sufficient decrease).** Assume (A3) and $\eta \leq \frac{1}{2\tau}$, then

$$
\mathbb{E}[f(\bar{x}^{t+1})] \leq f(\bar{x}^t) - \frac{\eta}{2} \mathbb{E}[\|\nabla f(\bar{x}^t)\|^2] - \frac{\eta}{4} \mathbb{E}[\|\bar{m}^{t+1}\|^2] - \frac{\eta}{\bar{\eta}} \mathbb{E}[\|\Delta^{t+1}\|^2] + \frac{\eta}{\bar{\eta}} e_1^{t+1},
$$

**Proof.** Use smoothness (A3) and expand it with (23)

$$
f(\bar{x}^{t+1}) \leq f(\bar{x}^t) - \langle \nabla f(\bar{x}^t), \eta \bar{m}^{t+1} - \Delta^{t+1} \rangle + \frac{L}{2} \eta^2 \|\bar{m}^{t+1} - \Delta^{t+1}\|^2
$$

Applying cosine theorem (17) to the inner product $\eta \langle \nabla f(\bar{x}^t), \bar{m}^{t+1} - \Delta^{t+1} \rangle$ yields

$$
\mathbb{E}[f(\bar{x}^{t+1})] \leq f(\bar{x}^t) - \frac{\eta}{2} \mathbb{E}[\|\nabla f(\bar{x}^t)\|^2] - \left( \frac{\eta - \frac{L}{2} \bar{\eta}}{2} \right) \mathbb{E}[\|\bar{m}^{t+1}\|^2] - \frac{\eta}{\bar{\eta}} \mathbb{E}[\|\Delta^{t+1}\|^2] + \frac{\eta}{\bar{\eta}} e_1^{t+1}.
$$

If step size $\eta \leq \frac{1}{2\tau}$, then $-\frac{\eta - \frac{L}{2} \bar{\eta}}{2} \leq -\frac{\eta}{4}$. Applying inequality (20) to the last term

$$
\frac{\eta}{2} \mathbb{E}[\|\nabla f(\bar{x}^t) - \bar{m}^{t+1}\|^2] + \frac{1}{\eta} \mathbb{E}[\|\Delta^{t+1}\|^2] \leq \eta \mathbb{E}[\|\nabla f(\bar{x}^t) - \bar{m}^{t+1}\|^2] + \frac{1}{\eta} \mathbb{E}[\|\Delta^{t+1}\|^2].
$$

Since $e_1^{t+1} := \mathbb{E}[\|\nabla f(\bar{x}^t) - \bar{m}^{t+1}\|^2] + \mathbb{E}[\|\Delta^{t+1}\|^2] \leq e_2^{t+1}$, then we have

$$
\mathbb{E}[f(\bar{x}^{t+1})] \leq f(\bar{x}^t) - \frac{\eta}{2} \mathbb{E}[\|\nabla f(\bar{x}^t)\|^2] - \frac{\eta}{4} \mathbb{E}[\|\bar{m}^{t+1}\|^2] - \frac{\eta}{\bar{\eta}} \mathbb{E}[\|\Delta^{t+1}\|^2] + \frac{\eta}{\bar{\eta}} e_2^{t+1}.
$$

In the next lemma, we establish the recursion for the distance between momentums and gradients

**Lemma 8.** Assume (A2) to (A4). For any doubly stochastic mixing matrix $A \in \mathbb{R}^{n \times n}$

$$
e_1^{t+1} = \frac{1}{|\mathcal{V}_k|} \sum_{i \in \mathcal{V}_k} \mathbb{E}[\sum_{j \in \mathcal{V}_k} A_{ij}(m_j^{t+1} - \nabla f_j(\bar{x}^t))|^2],
$$

then we have the following recursion

$$
e_1^{t+1} \leq (1 - \alpha) e_1^t + \frac{\alpha^2 \sigma^2}{|\mathcal{V}_k|} \mathbb{E}[A|^2_{\mathcal{V}_k} + 2\alpha L^2 \mathcal{E}^t + \frac{2L^2 \eta^2}{\alpha} \bar{m}^{t+1} - \frac{1}{\bar{\eta}} \mathbb{E}[\|\bar{m}^{t+1}\|^2].
$$

where we define $A^2_{\mathcal{V}_k} := \sum_{i \in \mathcal{V}_k} \sum_{j \in \mathcal{V}_k} A_{ij}^2$. Therefore,
Byzantine-robust decentralized learning via self-centered clipping

- If $A_{ij} = \frac{1}{|V_k|}$ for all $i, j \in V_k$, then $e_A^{t+1} = e_1^{t+1}$ and $\|A\|^2_{F, V_k} = 1$.
- If $A = \tilde{W}$, then $e_A^{t+1} = e_1^{t+1}$ and $\|A\|^2_{F, V_k} = \sum_{i \in V_k} \sum_{j \in V_k} \tilde{W}_{ij} \leq |V_k|$.
- If $A = I$, then $\|A\|^2_{F, V_k} = |V_k|$. In addition, $e_1^{t+1} \leq 2e_I^{t+1} + 2\xi^2$

where $A = I$.

Proof. Expand $e_A^{t+1}$ by expanding $m_j^{t+1}$

$$e_A^{t+1} = \frac{1}{|V_k|} \sum_{i \in V_k} \sum_{j \in V_k} A_{ij}((1-\alpha)m_j^t + \alpha g_j(x_j^t) - \nabla f_j(x^t)) ||^2$$

$$\leq \frac{1}{|V_k|} \sum_{i \in V_k} \sum_{j \in V_k} \sum_{i \in V_k} \sum_{j \in V_k} A_{ij}((1-\alpha)m_j^t + \alpha g_j(x_j^t) - \nabla f_j(x^t)) ||^2$$

Extract the stochastic term $g_j(x_j^t) - \nabla f_j(x_j^t)$ inside the norm and use that $\sum_{i \in V_k} \sum_{j \in V_k} g_j(x_j^t) = \nabla f_j(x^t)$,

$$e_A^{t+1} = \frac{1}{|V_k|} \sum_{i \in V_k} \sum_{j \in V_k} A_{ij}((1-\alpha)m_j^t + \alpha \nabla f_j(x_j^t) - \nabla f_j(x^t)) ||^2$$

$$\leq \frac{1}{|V_k|} \sum_{i \in V_k} \sum_{j \in V_k} \sum_{i \in V_k} \sum_{j \in V_k} A_{ij}((1-\alpha)m_j^t + \alpha \nabla f_j(x_j^t) - \nabla f_j(x^t)) ||^2$$

Then we can use (A2) for the last term to get

$$e_A^{t+1} = \frac{1}{|V_k|} \sum_{i \in V_k} \sum_{j \in V_k} A_{ij}((1-\alpha)m_j^t + \alpha \nabla f_j(x_j^t) - \nabla f_j(x^t)) ||^2 + \frac{\alpha^2 \sigma^2}{|V_k|} \|A\|^2_{F, V_k}.$$
Then we can apply smoothness (A.3) and use \((1 - \alpha)^2 \leq 1\)

\[
e^{t+1}_A \leq (1 - \alpha)e^{t}_A + \frac{\alpha^2 \sigma^2}{|V_k|} \|A\|^2_{F,V_k} + 2\alpha L^2 \Xi^2 + \frac{2L^2 \eta^2}{\alpha} \|m^t - \frac{1}{n} \Delta^t\|^2_2.
\]

Besides, consider \(\tilde{e}^{t+1}_1\)

\[
\tilde{e}^{t+1}_1 = \frac{1}{|V_k|} \sum_{i \in V_k} E \|m^{t+1}_i - \nabla f(\bar{x}^t)\|_2^2 = \frac{1}{|V_k|} \sum_{i \in V_k} E \|m^{t+1}_i \pm \nabla f_i(\bar{x}^t) - \nabla f(\bar{x}^t)\|_2^2 \\
\leq 2\frac{1}{|V_k|} \sum_{i \in V_k} E \|m^{t+1}_i - \nabla f_i(\bar{x}^t)\|_2^2 + 2\frac{1}{|V_k|} \sum_{i \in V_k} \|\nabla f_i(\bar{x}^t) - \nabla f(\bar{x}^t)\|^2_2 \\
= 2e^{t+1}_2 + 2\zeta^2.
\]

As we know that \(\|\Delta^{t+1}\|^2_2 \leq e^{t+1}_2\), then we need to finally bound \(e^{t+1}_2\)

**Lemma 9 (Bound on \(e^{t+1}_2\)).** For \(\delta_{\text{max}} := \max_{i \in V_k} \delta_i\), if

\[
\tau^{t+1}_i = \sqrt{\frac{1}{\delta_i} \sum_{j \in V_k} W_{ij} E \|x^{t+1/2}_i - x^{t+1/2}_j\|^2_2},
\]

then we have

\[
e^{t+1}_2 \leq 32\delta_{\text{max}}(2\eta^2(\tau^{t+1}_i + \zeta^2) + \Xi^t).
\]

**Proof.** Use Young’s inequality (18) to bound \(e^{t+1}_2\) by two parts

\[
e^{t+1}_2 = \frac{1}{|V_k|} \sum_{i \in V_k} E \left( \sum_{j \in V_k} W_{ij} (z^{t+1}_{j \rightarrow i} - x^{t+1/2}_j) + \sum_{j \in V_k} W_{ij} (x^{t+1}_{j \rightarrow i} - x^{t+1/2}_i) \right)^2 \\
\leq 2\frac{1}{|V_k|} \sum_{i \in V_k} E \left( \sum_{j \in V_k} W_{ij} (z^{t+1}_{j \rightarrow i} - x^{t+1/2}_j) \right)^2 + 2\frac{1}{|V_k|} \sum_{i \in V_k} E \left( \sum_{j \in V_k} W_{ij} (x^{t+1}_{j \rightarrow i} - x^{t+1/2}_i) \right)^2.
\]

Look at the first term use triangular inequality of \(\|\cdot\|\) and the definition of \(\tau^{t+1}_i\)

\[
A_1 \leq 2\frac{1}{|V_k|} \sum_{i \in V_k} \left( \sum_{j \in V_k} W_{ij} E \|z^{t+1}_{j \rightarrow i} - x^{t+1/2}_j\|^2_2 \right)^2 \\
\leq 2\frac{1}{|V_k|} \sum_{i \in V_k} \left( \tau^{t+1}_i \sum_{j \in V_k} W_{ij} E \|x^{t+1/2}_i - x^{t+1/2}_j\|^2_2 \right)^2.
\]

On the other hand,

\[
A_2 \leq 2\frac{1}{|V_k|} \sum_{i \in V_k} \left( \sum_{j \in V_k} W_{ij} E \|z^{t+1}_{j \rightarrow i} - x^{t+1/2}_j\|^2_2 \right)^2 \leq 2\frac{1}{|V_k|} \sum_{i \in V_k} \left( \sum_{j \in V_k} W_{ij} (\tau^{t+1}_i) \right)^2 \\
= 2\frac{1}{|V_k|} \sum_{i \in V_k} \delta^2_i (\tau^{t+1}_i)^2.
\]
Theorem II. Let $x_i^t$ be the average iterate over unknown set of regular nodes and choose $\tau_i$ to

$$\tau_i = \sqrt{\frac{1}{\delta_i} \sum_{j \in V_k} W_{ij} E \|x_i^t - x_j^t\|^2_2}.$$  

Then we come to the following bound

$$e_i^{t+1} \leq 4 \sum_{i \in V_k} \delta_i \sum_{j \in V_k} W_{ij} E \|x_i^t - x_j^t\|^2_2.$$  

Then we expand the norm as follows

$$E \left\| x_i^{t+1} - x_j^{t+1} \right\|^2_2 = E \left\| x_i^t - \eta m_i^{t+1} + x_j^t + \eta m_j^{t+1} \right\|^2_2$$

$$\leq 4\eta^2 E \|m_i^{t+1} - \eta \nabla f(x_i^t)\|_2^2 + 4\eta^2 E \|m_j^{t+1} - \eta \nabla f(x_j^t)\|_2^2$$

Then minimizing the RHS of $e_i^{t+1}$ by tuning radius for clipping

$$r_i^{t+1} = \sqrt{\frac{1}{\delta_i} \sum_{j \in V_k} W_{ij} E \left\| x_i^{t+1/2} - x_j^{t+1/2} \right\|^2_2}.$$  

Then we come to the following bound

$$e_i^{t+1} \leq \frac{4}{|V_k|} \sum_{i \in V_k} \delta_i W_{ij} E \left\| x_i^{t+1/2} - x_j^{t+1/2} \right\|^2_2.$$  

Proof. We can consider the 1-step consensus problem as 1-step of optimization problem with $\rho^2 = \Xi^t$ and $\eta = 0$. Then we look for the upper bound of $\frac{1}{|V_k|} \sum_{i \in V_k} E \|x_i^t - \bar{x}\|_2^2$ in terms of $\rho^2$, $p$, and $\delta_{\text{max}}$.

$$\frac{1}{|V_k|} \sum_{i \in V_k} E \|x_i^{t+1} - \bar{x}\|_2^2 = \frac{1}{|V_k|} \sum_{i \in V_k} E \left\| \sum_{j=1}^n W_{ij} x_j^{t+1} - \sum_{j \in V_k} W_{ij} x_j^t \right\|^2_2$$

$$= \frac{1}{|V_k|} \sum_{i \in V_k} E \left\| \left( \sum_{j \in V_k} \tilde{W}_{ij} x_j^t - \bar{x} \right) + \left( \sum_{j=1}^n W_{ij} x_j^{t+1} - \sum_{j \in V_k} W_{ij} x_j^t \right) \right\|^2_2.$$  

Theorem II. Let $\bar{x} := \frac{1}{|V_k|} \sum_{i \in V_k} x_i$ be the average iterate over unknown set of regular nodes and choose $\tau_i$ to

$$(9) \tau_i = \sqrt{\frac{1}{\delta_i} \sum_{j \in V_k} W_{ij} E \|x_i - x_j\|^2_2}.$$  

If the initial consensus distance is bounded as

$$\frac{1}{|V_k|} \sum_{i \in V_k} E \|x_i - \bar{x}\|^2 \leq \rho^2,$$  

then for all $i \in V_k$, SCCLIP outputs $\hat{x}_i$ such that

$$\frac{1}{|V_k|} \sum_{i \in V_k} E \|\hat{x}_i - \bar{x}\|^2 \leq (1 - \gamma + c\sqrt{\delta_{\text{max}}}) \rho^2$$  

where the expectation is over the random variable $\{x_i\}_{i \in V_k}$ and $c > 0$ is a constant.

Proof. We can consider the 1-step consensus problem as 1-step of optimization problem with $\rho^2 = \Xi^t$ and $\eta = 0$. Then we look for the upper bound of $\frac{1}{|V_k|} \sum_{i \in V_k} E \|x_i^{t+1} - \bar{x}\|_2^2$ in terms of $\rho^2$, $p$, and $\delta_{\text{max}}$.
Apply \((18)\) with \(\varepsilon > 0\) and use the expected improvement \((A6)\)

\[
\frac{1}{|V_x|} \sum_{i \in V_x} \mathbb{E} \| \mathbf{x}_i^{t+1} - \bar{x}_t \|^2_2
\]

\[
\leq \frac{1 + \varepsilon}{|V_x|} \sum_{i \in V_x} \| \sum_{j \in V_x} \bar{W}_{ij} \mathbf{x}_j^t - \bar{x}_t \|^2_2 + \frac{1 + \varepsilon}{|V_x|} \sum_{i \in V_x} \mathbb{E} \| \sum_{j=1}^n W_{ij} \mathbf{z}_{j \rightarrow i}^{t+1} - \sum_{j \in V_x} \bar{W}_{ij} \mathbf{x}_j^t \|^2_2
\]

\[
\leq \left(1 + \varepsilon \right) \left(1 - p \right) \frac{1}{|V_x|} \sum_{i \in V_x} \| \sum_{j \in V_x} \mathbf{x}_i^t - \bar{x}_t \|^2_2 + \frac{1 + \varepsilon}{|V_x|} \sum_{i \in V_x} \mathbb{E} \| \sum_{j=1}^n W_{ij} \mathbf{z}_{j \rightarrow i}^{t+1} - \sum_{j \in V_x} \bar{W}_{ij} \mathbf{x}_j^t \|^2_2
\]

\[
\leq \left(1 + \varepsilon \right) \left(1 - p \right) \Xi^t + \frac{1 + \varepsilon}{|V_x|} \sum_{i \in V_x} \mathbb{E} \| \sum_{j=1}^n W_{ij} \mathbf{z}_{j \rightarrow i}^{t+1} - \sum_{j \in V_x} \bar{W}_{ij} \mathbf{x}_j^{t+1/2} \|^2_2
\]

Replace \(\mathbf{x}_j^t = \mathbf{x}_j^{t+1/2} + \eta \mathbf{m}_j^{t+1}\) using \((14)\), then apply \((20)\) and \(\eta = 0\)

\[
\frac{1}{|V_x|} \sum_{i \in V_x} \mathbb{E} \| \mathbf{x}_i^{t+1} - \bar{x}_t \|^2_2 \leq \left(1 + \varepsilon \right) \left(1 - p \right) \Xi^t + \frac{1 + \varepsilon}{|V_x|} \sum_{i \in V_x} \mathbb{E} \| \sum_{j=1}^n W_{ij} \mathbf{z}_{j \rightarrow i}^{t+1} - \sum_{j \in V_x} \bar{W}_{ij} \mathbf{x}_j^{t+1/2} \|^2_2.
\]

Recall the definition of \(e_2^{t+1}\)

\[
e_2^{t+1} := \frac{1}{|V_x|} \sum_{i \in V_x} \mathbb{E} \| \sum_{j=1}^n W_{ij} \mathbf{z}_{j \rightarrow i}^{t+1} - \sum_{j \in V_x} \bar{W}_{ij} \mathbf{x}_j^{t+1/2} \|^2_2.
\]

Then use Lemma 8 with the case \(A = \bar{W}\) and apply Lemma 9 with \(\eta = 0\)

\[
\frac{1}{|V_x|} \sum_{i \in V_x} \mathbb{E} \| \mathbf{x}_i^{t+1} - \bar{x}_t \|^2_2 \leq \left(1 + \varepsilon \right) \left(1 - p \right) \Xi^t + \left(1 + \varepsilon \right) e_2^{t+1} \leq \left(1 + \varepsilon \right) \left(1 - p \right) \Xi^t + \left(1 + \varepsilon \right) 32 \delta_{\max} \Xi^t.
\]

Let’s minimize the right hand side of the above inequality by taking \(\varepsilon\) such that \(\varepsilon \left(1 - p \right) = \frac{32 \delta_{\max}}{\varepsilon}\) which leads to \(\varepsilon = \frac{32 \delta_{\max}}{1 - p}\), then the above inequality becomes

\[
\frac{1}{|V_x|} \sum_{i \in V_x} \mathbb{E} \| \mathbf{x}_i^{t+1} - \bar{x}_t \|^2_2 \leq \left(1 - p + 32 \delta_{\max} + 2 \sqrt{32 \delta_{\max} \left(1 - p \right)} \right) \Xi^t \leq \left(\sqrt{1 - p} + \sqrt{32 \delta_{\max}}\right)^2 \Xi^t.
\]

The consensus distance to the average consensus is only guaranteed to reduce if \(\sqrt{1 - p} + \sqrt{32 \delta_{\max}} < 1\) which is

\[
\delta_{\max} < \frac{1}{32} \left(1 - \sqrt{1 - p}\right)^2.
\]

Finally, we complete the proof by simplifying the notation to spectral gap \(\gamma := 1 - \sqrt{1 - p}\).

Recall that

\[
e_2^{t+1} := \frac{1}{|V_x|} \sum_{i \in V_x} \left\| \sum_{j \in V_x} W_{ij} (\mathbf{x}_j^{t+1} - \bar{x}_t^{t+1/2}) + \sum_{j \in V_x} W_{ij} (\mathbf{x}_j^{t+1} - \bar{x}_i^{t+1/2}) \right\|^2_2. \tag{25}
\]

Next we consider the bound on consensus distance \(\Xi^t\).

**Lemma 10** (Bound consensus distance \(\Xi^t\)). Assume \((A6)\), then \(\Xi^t\) has the following iteration

\[
\Xi^{t+1} \leq \left(1 + \varepsilon \right) \left(1 - p \right) \Xi^t + 5 \left(1 + \frac{1}{\varepsilon} \right) \left( \varepsilon e_2^{t+1} + \eta^2 \varepsilon_2^{t+1} + \eta^2 \varepsilon_2^{t+1} + \eta^2 \| \nabla f(\bar{x}) \|_2^2 + \eta^2 \mathbb{E} \| \mathbf{m}_t^{t+1} - \mathbf{m}_t^{t+1} \|^2_2 \right).
\]

where \(\varepsilon > 0\) is determined later.
Proof. Expand the consensus distance at time $t+1$

$$
\Xi^{t+1} = \frac{1}{|\mathcal{V}_k|} \sum_{i \in \mathcal{V}_k} \mathbb{E} \left[ \| x^t_{j+1} - \bar{x}^{t+1} \|^2 \right] = \frac{1}{|\mathcal{V}_k|} \sum_{i \in \mathcal{V}_k} \mathbb{E} \left[ \sum_{j=1}^{n} \mathbf{W}_{ij} \mathbf{z}_{j+1} - \bar{x}^{t+1} \right]^2
$$
$$
= \frac{1}{|\mathcal{V}_k|} \sum_{i \in \mathcal{V}_k} \mathbb{E} \left[ \sum_{j=1}^{n} \mathbf{W}_{ij} \mathbf{z}_{j+1} - \bar{x}^t + \bar{x}^t - \bar{x}^{t+1} \| \right]^2
$$
$$
= \frac{1}{|\mathcal{V}_k|} \sum_{i \in \mathcal{V}_k} \mathbb{E} \left[ (\sum_{j \in \mathcal{V}_k} \tilde{W}_{ij} x_j^t - \bar{x}^t) + (\sum_{j=1}^{n} \mathbf{W}_{ij} \mathbf{z}_{j+1} - \sum_{j \in \mathcal{V}_k} \tilde{W}_{ij} x_j^t) + \bar{x}^t - \bar{x}^{t+1} \right]^2.
$$

Apply (18) with $\varepsilon = \sqrt{\frac{32 \delta_{\max}}{1-p}}$, like the proof of Theorem II, and use the expected improvement (A6)

$$
\Xi^{t+1} \leq \frac{1 + \varepsilon}{|\mathcal{V}_k|} \sum_{i \in \mathcal{V}_k} \mathbb{E} \left[ \sum_{j \in \mathcal{V}_k} \tilde{W}_{ij} x_j^t - \bar{x}^t \right]^2
$$
$$
+ \frac{1 + \varepsilon}{|\mathcal{V}_k|} \sum_{i \in \mathcal{V}_k} \mathbb{E} \left[ \sum_{j=1}^{n} \mathbf{W}_{ij} \mathbf{z}_{j+1} - \sum_{j \in \mathcal{V}_k} \tilde{W}_{ij} x_j^t + \bar{x}^t - \bar{x}^{t+1} \right]^2
$$
$$
\leq \frac{(1 + \varepsilon)(1 - p)}{|\mathcal{V}_k|} \sum_{i \in \mathcal{V}_k} \mathbb{E} \left[ \sum_{j=1}^{n} \mathbf{W}_{ij} \mathbf{z}_{j+1} - \sum_{j \in \mathcal{V}_k} \tilde{W}_{ij} x_j^t + \bar{x}^t - \bar{x}^{t+1} \right]^2
$$
$$
\leq (1 + \varepsilon)(1 - p) \Xi^t + \frac{1 + \varepsilon}{|\mathcal{V}_k|} \sum_{i \in \mathcal{V}_k} \mathbb{E} \left[ (\sum_{j=1}^{n} \mathbf{W}_{ij} \mathbf{z}_{j+1} - \sum_{j \in \mathcal{V}_k} \tilde{W}_{ij} x_j^t) + \bar{x}^t - \bar{x}^{t+1} \right]^2
$$

Replace $x_j^t = x_{j+1}^{t+1/2} + \eta m_j^{t+1}$ using (14), then apply (20)

$$
T_1 = \frac{1 + \varepsilon}{|\mathcal{V}_k|} \sum_{i \in \mathcal{V}_k} \mathbb{E} \left[ \sum_{j=1}^{n} \mathbf{W}_{ij} \mathbf{z}_{j+1} - \sum_{j \in \mathcal{V}_k} \tilde{W}_{ij} x_j^{t+1/2} - \eta \sum_{j \in \mathcal{V}_k} \tilde{W}_{ij} m_j^{t+1} + \bar{x}^t - \bar{x}^{t+1} \right]^2
$$
$$
\leq \frac{5}{|\mathcal{V}_k|} \sum_{i \in \mathcal{V}_k} \mathbb{E} \left[ \sum_{j=1}^{n} \mathbf{W}_{ij} \mathbf{z}_{j+1} - \sum_{j \in \mathcal{V}_k} \tilde{W}_{ij} x_j^{t+1/2} \right]^2 + \frac{\eta^2}{|\mathcal{V}_k|} \sum_{i \in \mathcal{V}_k} \mathbb{E} \left[ \sum_{j \in \mathcal{V}_k} \tilde{W}_{ij} (m_j^{t+1} - \nabla f_j(\bar{x}^t)) \right]^2
$$
$$
+ \frac{\eta^2}{|\mathcal{V}_k|} \sum_{i \in \mathcal{V}_k} \mathbb{E} \left[ \sum_{j \in \mathcal{V}_k} \tilde{W}_{ij} \nabla f_j(\bar{x}^t) - \nabla f(\bar{x}^t) \right]^2 + \eta^2 \| \nabla f(\bar{x}^t) \|^2 + \mathbb{E} \| \bar{x}^t - \bar{x}^{t+1} \|^2 \right].
$$

Recall the definition of $\epsilon_{2}^{t+1}$

$$
\epsilon_{2}^{t+1} = \frac{1}{|\mathcal{V}_k|} \sum_{i \in \mathcal{V}_k} \mathbb{E} \left[ \sum_{j \in \mathcal{V}_k} \mathbf{W}_{ij} (\mathbf{z}_{j+1} - x_j^{t+1/2}) + \sum_{j \in \mathcal{V}_k} \mathbf{W}_{ij} (x_j^{t+1/2} - x_j^{t+1/2}) \right]^2
$$
$$
= \frac{1}{|\mathcal{V}_k|} \sum_{i \in \mathcal{V}_k} \mathbb{E} \left[ \sum_{j=1}^{n} \mathbf{W}_{ij} \mathbf{z}_{j+1} - \sum_{j \in \mathcal{V}_k} \tilde{W}_{ij} x_j^{t+1/2} \right]^2
$$

Then use Lemma 8 with the case $A = \tilde{W}$,

$$
T_1 \leq 5 \left(1 + \frac{1}{\varepsilon} \right) \left( \epsilon_{2}^{t+1} + \eta^2 \epsilon_{1}^{t+1} + \frac{\eta^2}{|\mathcal{V}_k|} \sum_{i \in \mathcal{V}_k} \mathbb{E} \left[ \sum_{j \in \mathcal{V}_k} \tilde{W}_{ij} \nabla f_j(\bar{x}^t) - \nabla f(\bar{x}^t) \right]^2 + \eta^2 \| \nabla f(\bar{x}^t) \|^2 + \mathbb{E} \| \bar{x}^t - \bar{x}^{t+1} \|^2 \right).
$$
We can extend (Koloskova et al., 2020b, Lemma 15)) to the following lemma

\[ T_1 \leq 5(1 + \frac{1}{\varepsilon}) \left( e^{t+1}_2 + \eta^2 e^{t+1}_1 + \eta^2 \zeta^2 + \eta^2 \| \nabla f(\bar{x}^t) \|^2 + \mathbb{E} \| \bar{x}^t - x^{t+1} \|^2 \right). \]

Use (23) for the last term

\[ T_1 \leq 5(1 + \frac{1}{\varepsilon}) \left( e^{t+1}_2 + \eta^2 e^{t+1}_1 + \eta^2 \zeta^2 + \eta^2 \| \nabla f(\bar{x}^t) \|^2 + \eta^2 \mathbb{E} \| \bar{m}^{t+1} - \frac{1}{\eta} \Delta^{t+1} \|^2 \right). \]

Finally, by the definition of \( e^{t+1}_1 \), we have

\[ \Xi^{t+1} \leq (1 + \varepsilon)(1 - p) \Xi^t + 5(1 + \frac{1}{\varepsilon}) \left( e^{t+1}_2 + \eta^2 e^{t+1}_1 + \eta^2 \zeta^2 + \eta^2 \| \nabla f(\bar{x}^t) \|^2 + \eta^2 \mathbb{E} \| \bar{m}^{t+1} - \frac{1}{\eta} \Delta^{t+1} \|^2 \right). \]

We can extend (Koloskova et al., 2020b, Lemma 15)) to the following lemma

**Lemma 11** (Tuning stepsize).

\[ \Psi_T := \frac{r_0}{\eta(T + 1)} + b\eta + e\eta^2 + f\eta^3 \leq 2 \left( \frac{br_0}{T + 1} \right)^{\frac{1}{2}} + 2e^{\frac{1}{4}} \left( \frac{r_0}{T + 1} \right)^{\frac{1}{2}} + 2f^{\frac{1}{4}} \left( \frac{r_0}{T + 1} \right)^{\frac{1}{2}} + \frac{dr_0}{T + 1}, \]

**Proof.** Choosing \( \eta = \min \left\{ \left( \frac{r_0}{b(T + 1)} \right)^{\frac{1}{2}}, \left( \frac{r_0}{e(T + 1)} \right)^{\frac{1}{2}}, \left( \frac{r_0}{f(T + 1)} \right)^{\frac{1}{2}} \right\} \leq \frac{1}{\sqrt{T}} \) we have four cases

- \( \eta = \frac{1}{\sqrt{T}} \) and is smaller than \( \left( \frac{r_0}{b(T + 1)} \right)^{\frac{1}{2}}, \left( \frac{r_0}{e(T + 1)} \right)^{\frac{1}{2}}, \left( \frac{r_0}{f(T + 1)} \right)^{\frac{1}{2}} \), then

  \[ \Psi_T \leq \frac{dr_0}{T + 1} + \frac{b}{T} + \frac{e}{T^2} + \frac{f}{T^3} \leq \frac{dr_0}{T + 1} + \left( \frac{br_0}{T + 1} \right)^{\frac{1}{2}} + e^{1/3} \left( \frac{r_0}{T + 1} \right)^{\frac{1}{2}} + f^{1/4} \left( \frac{r_0}{T + 1} \right)^{\frac{1}{2}}. \]

- \( \eta = \left( \frac{r_0}{b(T + 1)} \right)^{\frac{1}{2}} < \min \left\{ \left( \frac{r_0}{e(T + 1)} \right)^{\frac{1}{2}}, \left( \frac{r_0}{f(T + 1)} \right)^{\frac{1}{2}} \right\} \), then

  \[ \Psi_T \leq 2 \left( \frac{br_0}{T + 1} \right)^{\frac{1}{2}} + e \left( \frac{r_0}{b(T + 1)} \right)^{\frac{1}{2}} \leq 2 \left( \frac{br_0}{b(T + 1)} \right)^{\frac{1}{2}} + e^{1/3} \left( \frac{r_0}{T + 1} \right)^{\frac{1}{2}} + f^{1/4} \left( \frac{r_0}{T + 1} \right)^{\frac{1}{2}}. \]

- \( \eta = \left( \frac{r_0}{e(T + 1)} \right)^{\frac{1}{2}} < \min \left\{ \left( \frac{r_0}{b(T + 1)} \right)^{\frac{1}{2}}, \left( \frac{r_0}{f(T + 1)} \right)^{\frac{1}{2}} \right\} \), then

  \[ \Psi_T \leq 2e^{1/3} \left( \frac{r_0}{T + 1} \right)^{\frac{1}{2}} + b \left( \frac{r_0}{e(T + 1)} \right)^{\frac{1}{2}} + \frac{fr_0}{e(T + 1)} \leq \left( \frac{br_0}{T + 1} \right)^{\frac{1}{2}} + e^{1/3} \left( \frac{r_0}{T + 1} \right)^{\frac{1}{2}} + f^{1/4} \left( \frac{r_0}{T + 1} \right)^{\frac{1}{2}}. \]

- \( \eta = \left( \frac{r_0}{f(T + 1)} \right)^{\frac{1}{2}} < \min \left\{ \left( \frac{r_0}{b(T + 1)} \right)^{\frac{1}{2}}, \left( \frac{r_0}{e(T + 1)} \right)^{\frac{1}{2}} \right\} \), then

  \[ \Psi_T \leq 2f^{1/4} \left( \frac{r_0}{T + 1} \right)^{\frac{1}{2}} + e \left( \frac{r_0}{f(T + 1)} \right)^{\frac{1}{2}} \leq \left( \frac{br_0}{T + 1} \right)^{\frac{1}{2}} + e^{1/3} \left( \frac{r_0}{T + 1} \right)^{\frac{1}{2}} + 2f^{1/4} \left( \frac{r_0}{T + 1} \right)^{\frac{1}{2}}. \]

Then, take the uniform upper bound of the upper bound gives the result. \( \square \)
C.3. Proof of the main theorem

**Theorem III.** Suppose Assumptions 1–6 hold and $\delta_{\text{max}} \leq \gamma^2/10 \cdot 2^{10}$. Define clipping radius to be

$$
\tau_{t+1}^i = \sqrt{\frac{1}{N} \sum_{j \in V_x} W_{ij} E \left\| x_{i}^{t+1/2} - x_{j}^{t+1/2} \right\|^2}.
$$

(10)

By taking $\eta \leq \frac{\gamma}{40L}$ and $\alpha := 3\eta L$, we have

$$
\frac{1}{T} \sum_{t=0}^{T} \| \nabla f(\bar{x}^t) \|_2^2 \leq \frac{200^2}{\alpha^2} \delta_{\text{max}} \zeta^2 + 2 \left( \frac{\eta^2}{|V_x|} + \frac{320^2}{\alpha^2} \delta_{\text{max}} \right)^{1/2} \left( \frac{3L\sigma^2 r_0}{T+1} \right)^{1/2} + 2 \left( \frac{600}{\alpha^2} \right)^{1/4} \left( \frac{ro L}{T+1} \right)^{3/4} + \frac{d_0 r_0}{T+1}.
$$

where $r_0 := f(x^0) - f^\ast$.

**Proof.** Denote the terms of average $t$ from 0 to $T$ as follows

$$
C_1 := \frac{1}{1 + T} \sum_{t=0}^{T} \| \nabla f(\bar{x}^t) \|_2^2, C_2 := \frac{1}{1 + T} \sum_{t=0}^{T} \| m^{t+1} - \Delta^{t+1} \|_2^2, D_1 := \frac{1}{1 + T} \sum_{t=0}^{T} \Xi^{t+1},
$$

$$
E_1 := \frac{1}{1 + T} \sum_{t=0}^{T} e_1^{t+1}, E \Xi := \frac{1}{1 + T} \sum_{t=0}^{T} e_1^{t+1}, E_1 := \frac{1}{1 + T} \sum_{t=0}^{T} e_1^{t+1}, E_2 := \frac{1}{1 + T} \sum_{t=0}^{T} e_2^{t+1},
$$

First we apply average to Lemma 9

$$
E_2 \leq 32\delta_{\text{max}} (2\eta^2 (E_\Xi + \zeta^2) + D_1).
$$

(26)

Then we rewrite key Lemma 7 as

$$
\| \nabla f(\bar{x}^t) \|_2^2 + \frac{1}{2} E \| m^{t+1} - \Delta^{t+1} \|_2^2 \leq \frac{2}{\eta^2} (r^t - r^{t+1}) + 2e_1^{t+1} + \frac{2}{\eta^2} e_2^{t+1},
$$

and further average over time $t$

$$
C_1 + \frac{1}{2} C_2 \leq \frac{2r_0}{\eta(T+1)} + 2E_1 + \frac{2}{\eta^2} E_2
$$

where we use $-f(x^{T+1}) \leq -f^\ast$. Combined with (26) gives

$$
C_1 + \frac{1}{2} C_2 \leq \frac{2r_0}{\eta(T+1)} + 2E_1 + 128\delta_{\text{max}} E_\Xi + 128\delta_{\text{max}} \zeta^2 + \frac{64\delta_{\text{max}}}{\eta^2} D_1
$$

(27)

Now we also average Lemma 8 for $e_{1}^{t+1}$ over $t$ gives

$$
\frac{1}{1 + T} \sum_{t=0}^{T} e_1^{t+1} \leq \frac{1}{1 + T} \sum_{t=0}^{T} e_1^t + 2\alpha L^2 D_1 + \frac{\alpha^2 \sigma^2}{|V_x|} + \frac{2L^2 \eta^2}{\alpha} \frac{1}{1 + T} \sum_{t=0}^{T} \| m^t - \frac{\Delta}{\eta} \|_2^2
$$

$$
\leq \frac{1}{1 + T} \sum_{t=0}^{T} e_1^{t+1} + 2\alpha L^2 D_1 + \frac{\alpha^2 \sigma^2}{|V_x|} + \frac{2L^2 \eta^2}{\alpha} C_2
$$

where we use $\Xi^0 = e_1^0 = 0$ and $m^0 = \Delta^0 = 0$. Then let $\beta_1 := \frac{2L^2 \eta^2}{\alpha^2}

$$
E_1 \leq 2L^2 D_1 + \frac{\alpha \sigma^2}{|V_x|} + \beta_1 C_2.
$$

(28)
Similarly, Lemma 8 for $\epsilon_I^{t+1}$ the only difference is that we don’t have $\frac{1}{n}$ for $\sigma^2$ 

\[ E_I \leq 2L^2 D_1 + \alpha \sigma^2 + \beta_1 C_2. \]  

(29) 

Similarly, let's call $\beta_2 := \frac{1}{|V_s|} \sum_{i \in V_s} \sum_{j \in V_s} \tilde{W}_{ij} \geq 1$ 

\[ \tilde{E}_1 \leq 2L^2 D_1 + \beta_2 \alpha \sigma^2 + \beta_1 C_2. \]  

(30) 

The consensus distance Lemma 10 has 

\[ D_1 \leq \frac{(1 + \varepsilon)(1 - p)}{1 + T} \sum_{t=0}^{T} \varepsilon^t + 5(1 + \frac{1}{\varepsilon}) E_2 + 5(1 + \frac{1}{\varepsilon}) \eta^2 (E_I^{t+1} + \zeta^2 + C_1 + C_2) \]

\[ \leq (1 + \varepsilon)(1 - p) D_1 + 5(1 + \frac{1}{\varepsilon}) E_2 + 5(1 + \frac{1}{\varepsilon}) \eta^2 (\bar{E}_1^{t+1} + \zeta^2 + C_1 + C_2). \]

Replace $E_2$ using (26) gives 

\[ D_1 \leq (1 + \varepsilon)(1 - p) D_1 + 5(1 + \frac{1}{\varepsilon}) (32\delta_{\max}(2\eta^2(E_I^{t+1} + \zeta^2) + D_1)) + 5(1 + \frac{1}{\varepsilon}) \eta^2 (\bar{E}_1^{t+1} + \zeta^2 + C_1 + C_2) \]

\[ \leq ((1 + \varepsilon)(1 - p) + 160(1 + \frac{1}{\varepsilon}) \delta_{\max}) D_1 + 5(1 + \frac{1}{\varepsilon}) \eta^2 (64\delta_{\max}E_I^{t+1} + \bar{E}_1^{t+1} + (1 + 64\delta_{\max}) \zeta^2 + C_1 + C_2). \]

Now replace $\bar{E}_1$, $E_I$ with (30), (29), then 

\[ D_1 \leq ((1 + \varepsilon)(1 - p) + 5(1 + \frac{1}{\varepsilon}) (32\delta_{\max}(1 + 4L^2 \eta^2) + 2L^2 \eta^2)) \]

\[ + 5(1 + \frac{1}{\varepsilon}) \eta^2 (64\delta_{\max} + \beta_2) \alpha \sigma^2 + (64\delta_{\max} + 1) \zeta^2 + ((64\delta_{\max} + 1) \beta_1 + 1) C_2 + C_1). \]

By enforcing $\eta \leq \sqrt{\frac{\gamma}{4T}}$ and $\delta_{\max} \leq \sqrt{\frac{\sigma^2}{160n}}$ we have 

\[ 10L^2 \eta^2 \leq \gamma^2 / 8 \]

\[ 160 \delta_{\max} (1 + 4L^2 \eta^2) \leq \gamma^2 / 8 \]

we can achieve 

\[ \sqrt{160 \delta_{\max} (1 + 4L^2 \eta^2)} + 10L^2 \eta^2 \leq \frac{\gamma}{2}. \]

Then 

\[ D_1 \leq ((1 + \varepsilon)(1 - p) + (1 + \frac{1}{\varepsilon}) \frac{\gamma^2}{4}) D_1 \]

\[ =: T_2 \]

\[ + 5(1 + \frac{1}{\varepsilon}) \eta^2 (64\delta_{\max} + \beta_2) \alpha \sigma^2 + (64\delta_{\max} + 1) \zeta^2 + ((64\delta_{\max} + 1) \beta_1 + 1) C_2 + C_1). \]

Let us minimize the the coefficients of $D_1$ on the right hand side of inequality by having 

\[ \varepsilon(1 - p) = \frac{1}{\varepsilon} \frac{\gamma^2}{4}, \]

that is $\varepsilon = \sqrt{\frac{\gamma^2}{4(1 - p)}}$. Then the coefficient becomes 

\[ T_2 = (1 + \varepsilon)(1 - p) + (1 + \frac{1}{\varepsilon}) \frac{\gamma^2}{4} \]

\[ = (\sqrt{1 - p} + \frac{\gamma}{2})^2 \]

\[ = (1 - \frac{\gamma}{2})^2. \]

Then we use $\frac{1}{\varepsilon} = \sqrt{\frac{4(1 - p)}{\gamma}} \leq \frac{2}{\gamma}$ and $1 + \frac{1}{\varepsilon} \leq \frac{3}{\gamma}$ 

\[ D_1 \leq \frac{20\eta^2}{\gamma^2} ((64\delta_{\max} + \beta_2) \alpha \sigma^2 + (64\delta_{\max} + 1) \zeta^2 + ((64\delta_{\max} + 1) \beta_1 + 1) C_2 + C_1). \]
This leads to $64\delta_{\text{max}} \leq \frac{\gamma^2}{20} \leq 1$ and $\beta_2 \leq 1$, then we know

$$D_1 \leq \frac{20\eta^2}{\gamma^2} (2\alpha\sigma^2 + 2\zeta^2 + C_1 + (1 + 2\beta_1)C_2)$$ (31)

Finally, we combine (27), (28), (30)

$$C_1 + \frac{1}{2}C_2 \leq \frac{2r_0}{\eta(T + 1)} + 2E_1 + 128\delta_{\text{max}}E_4 + 128\delta_{\text{max}}\zeta^2 + \frac{64\delta_{\text{max}}}{\eta^2}D_1$$

$$\leq \frac{2r_0}{\eta(T + 1)} + (4L^2D_1 + \frac{2\alpha\sigma^2}{\|v_{\xi}\|} + 2\beta_1C_2) + 64\delta_{\text{max}}(4L^2D_1 + 2\beta_2\sigma^2 + 2\beta_1C_2)$$

$$+ 128\delta_{\text{max}}\zeta^2 + \frac{64\delta_{\text{max}}}{\eta^2}D_1$$

$$\leq \frac{2r_0}{\eta(T + 1)} + (4L^2 + 256\delta_{\text{max}}L^2 + \frac{64\delta_{\text{max}}}{\eta^2})D_1 + \left(\frac{1}{\|v_{\xi}\|} + 64\delta_{\text{max}}\right)2\alpha\sigma^2$$

$$+ 4\beta_1C_2 + 128\delta_{\text{max}}\zeta^2$$

Then we replace $D_1$ with (31)

$$C_1 + \frac{1}{2}C_2 \leq \frac{2r_0}{\eta(T + 1)} + \left(\frac{1}{\|v_{\xi}\|} + 64\delta_{\text{max}}\right)2\alpha\sigma^2 + 4\beta_1C_2 + 128\delta_{\text{max}}\zeta^2$$

$$+ (4L^2\eta^2 + 256\delta_{\text{max}}L^2\eta^2 + 64\delta_{\text{max}})\frac{20}{\gamma} (2\alpha\sigma^2 + 2\zeta^2 + C_1 + (1 + 2\beta_1)C_2)$$

To have a valid bound on $C_1$, there are two constraints on the coefficient of the RHS $C_1$ and $C_2$. 

$$\left(4L^2\eta^2 + 256\delta_{\text{max}}L^2\eta^2 + 64\delta_{\text{max}}\right)\frac{20}{\gamma} < 1$$

$$\left(4L^2\eta^2 + 256\delta_{\text{max}}L^2\eta^2 + 64\delta_{\text{max}}\right)\frac{20}{\gamma} (1 + 2\beta_1) + 4\beta_1 \leq \frac{1}{2}.$$ 

We can strength the first requirement to

$$\left(4L^2\eta^2 + 256\delta_{\text{max}}L^2\eta^2 + 64\delta_{\text{max}}\right)\frac{20}{\gamma} \leq \frac{1}{4}.$$ (33)

Then, apply this inequality to the second inequality gives

$$\frac{1}{4} + \frac{1}{2}\beta_1 + 4\beta_1 \leq \frac{1}{2}$$

which requires $\eta \leq \frac{\gamma}{20\sqrt{2}}$. Next (33) can be achieved by requiring $\delta_{\text{max}} \leq \frac{\gamma^2}{20120}$

$$(4 + 256\delta_{\text{max}})L^2\eta^2 + 64\delta_{\text{max}} \leq 8L^2\eta^2 + 64\delta_{\text{max}} \leq \frac{\gamma^2}{80}$$

which requires $8\eta^2L^2 \leq \frac{\gamma^2}{1000}$, and we can simplify it to $\eta \leq \frac{\gamma}{40\sqrt{2}}$. Now we can simplify (32) with (33)

$$\frac{3}{4}C_1 \leq \frac{2r_0}{\eta(T + 1)} + \left(\frac{1}{\|v_{\xi}\|} + 64\delta_{\text{max}}\right)3\alpha\sigma^2 + 128\delta_{\text{max}}\zeta^2$$

$$+ (4L^2\eta^2 + 256\delta_{\text{max}}L^2\eta^2 + 64\delta_{\text{max}})\frac{20}{\gamma} \left(2\alpha\sigma^2 + 2\zeta^2\right)$$

Multiply both sides with $\frac{3}{4}$ and relax constant $\frac{3}{4} \cdot 2 \leq 3$. Then by taking $\eta \leq \frac{\gamma}{40\sqrt{2}}$, we have that

$$C_1 \leq \frac{3r_0}{\eta(T + 1)} + \left(\frac{1}{\|v_{\xi}\|} + \frac{151}{\gamma} 64\delta_{\text{max}}\right)3\alpha\sigma^2 + \frac{200}{\gamma^2} \delta_{\text{max}}\zeta^2 + \frac{200}{\gamma^2} \left(\alpha\sigma^2 + \zeta^2\right)L^2\eta^2$$

By taking $\alpha := 3\eta L$ and relax the constants we have

$$C_1 \leq \frac{3r_0}{\eta(T + 1)} + \left(\frac{3^2}{\|v_{\xi}\|} + \frac{320}{\gamma^2} \delta_{\text{max}}\right)L\sigma^2\eta + \frac{200}{\gamma^2} \left(\alpha\sigma^2 + \zeta^2\right)L^2\eta^2 + \frac{200}{\gamma^2} \delta_{\text{max}}\zeta^2.$$
Minimize the right hand side by tuning step size Lemma 11 we have

\[
\frac{1}{T+1} \sum_{t=0}^{T} \| \nabla f(\bar{x}^t) \|^2 \leq \frac{200^2}{\gamma^2} \delta_{\text{max}} \epsilon^2 + 2 \left( \frac{\gamma^2}{2} + \frac{320^2 \delta_{\text{max}}}{T+1} \right)^{\frac{1}{2}}
\]

\[
+ 2 \left( \frac{200}{\gamma^2} \epsilon^2 \right)^{\frac{3}{2}} \left( \frac{\gamma L}{T+1} \right)^{\frac{3}{2}} + 2 \left( \frac{600 \sigma}{\gamma T+1} \right)^{\frac{1}{4}} \left( \frac{\gamma L}{T+1} \right)^{\frac{3}{4}} + \frac{d_0 r_0}{T+1}
\]

where \( \eta \leq \min \{ \frac{1}{4L}, \frac{7}{40L}, \frac{7}{40L} \} = \frac{7}{40L} =: \frac{1}{d_0}. \]