Muffliato: Peer-to-Peer Privacy Amplification for Decentralized Optimization and Averaging

Anonymous Author(s)
Affiliation
Address
email

Abstract

Decentralized optimization is increasingly popular in machine learning for its scalability and efficiency. Intuitively, it should also provide better privacy guarantees, as nodes only observe the messages sent by their neighbors in the network graph. But formalizing and quantifying this gain is challenging: existing results are typically limited to Local Differential Privacy (LDP) guarantees that overlook the advantages of decentralization. In this work, we introduce pairwise network differential privacy, a relaxation of LDP that captures the fact that the privacy leakage from a node u to a node v may depend on their relative position in the graph. We then analyze the combination of local noise injection with (simple or randomized) gossip averaging protocols on fixed and random communication graphs. We also derive a differentially private decentralized optimization algorithm that alternates between local gradient descent steps and gossip averaging. Our results show that our algorithms amplify privacy guarantees as a function of the distance between nodes in the graph, matching the privacy-utility trade-off of the trusted curator, up to factors that explicitly depend on the graph topology. Finally, we illustrate our privacy gains with experiments on synthetic and real-world datasets.

1 Introduction

Training machine learning models traditionally requires centralizing data in a single server, raising issues of scalability and privacy. An alternative is to use Federated Learning (FL), where each user keeps her data on device [41, 33]. In *fully decentralized* FL, the common hypothesis of a central server is also removed, letting users, represented as nodes in a graph, train the model via peer-to-peer communications along edges. This approach improves scalability and robustness to central server failures, enabling lower latency, less power consumption and quicker deployment [40, 10, 48, 46, 1, 39, 36].

Another important dimension is privacy, as a wide range of applications deal with sensitive and personal data. The gold standard to quantify the privacy leakage of algorithms is Differential Privacy (DP) [18]. DP typically requires to randomly perturb the data-dependent computations to prevent the final model from leaking too much information about any individual data point (e.g., through data memorization). However, decentralized algorithms do not only reveal the final model to the participating nodes, but also the results of some intermediate computations. A solution is to use Local Differential Privacy (LDP) [34, 17], where random perturbations are performed locally by each user, thus protecting against an attacker that would observe everything that users share. This can be easily combined with decentralized algorithms, as done for instance in [31, 5, 13, 53, 51]. Unfortunately, LDP requires large amounts of noise, and thus provides poor utility.

In this work, we show that the LDP guarantees give a very pessimistic view of the privacy offered by decentralized algorithms. Indeed, there is no central server receiving all messages, and the

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

participating nodes can only observe the messages sent by their neighbors in the graph. So, a given node should intuitively leak less information about its private data to nodes that are far away. We formally quantify this privacy amplification for the fundamental brick of communication at the core of decentralized optimization: gossip algorithms. Calling *Muffliato* the combination of local noise injection with a gossip averaging protocol, we precisely track the resulting privacy leakage between each pair of nodes. Through gossiping, the private values and noise terms of various users add up, obfuscating their contribution well beyond baseline LDP guarantees: as their distance in the graph increases, the privacy loss decreases. We then show that the choice of graph is crucial to enforce a good privacy-utility trade-off while preserving the scalability of gossip algorithms.

Our results are particularly attractive in situations where nodes want stronger guarantees with respect to some (distant) peers. For instance, in social network graphs, users may have lower privacy expectations with respect to close relatives than regarding strangers. In healthcare, a patient might trust her family doctor more than she trusts other doctors, and in turn more than employees of a regional agency and so on, creating a hierarchical level of trust that our algorithms naturally match.

Contributions and outline of the paper

(i) We introduce *pairwise network DP*, a relaxation of Local Differential Privacy inspired by the definitions of Cyffers and Bellet [15], which is able to quantify the privacy loss of a decentralized algorithm for each pair of distinct users in a graph.

(ii) We propose *Muffliato*[1], a privacy amplification mechanism composed of local Gaussian noise injection at the node level followed by gossiping for averaging the private values. It offers privacy amplification that increases as the distance between two nodes increases. Informally, the locally differentially private value shared by a node u is mixed with other contributions, to the point that the information that leaks to another node v can have a very small sensitivity to the initial value in comparison to the accumulated noise.

(iii) We analyze both synchronous gossip [16] and randomized gossip [10] under a unified privacy analysis with arbitrary time-varying gossip matrices. We show that the magnitude of the privacy amplification is significant: the average privacy loss over all the pairs in this setting reaches the optimal utility-privacy of a trusted aggregator, up to a factor $\frac{d}{\sqrt{\lambda_W}}$, where λ_W is the weighted graph eigengap and d the maximum degree of the graph. Remarkably, this factor can be of order 1 for expanders, yielding a sweet spot in the privacy-utility-scalability trade-off of gossip algorithms. Then, we study the case where the graph is itself random and private, and derive stronger privacy guarantees.

(iv) Finally, we develop and analyze differentially private decentralized Gradient Descent (GD) and Stochastic Gradient Descent (SGD) algorithms to minimize a sum of local objective functions. Building on *Muffliato*, our algorithms alternate between rounds of differentially private gossip communications and local gradient steps. We prove that they enjoy the same privacy amplification described above for averaging, up to factors that depend on the regularity of the global objective.

(v) We show the usefulness of our approach and analysis through experiments on synthetic and real-world datasets and network graphs, illustrating how privacy is amplified between nodes in the graph as a function of their distance.

Related work

Gossip algorithms and decentralized optimization. Gossip algorithms [9, 16] were introduced to compute the global average of local vectors through peer-to-peer communication, and are at the core of many decentralized optimization algorithms. Classical decentralized optimization algorithms alternate between gossip communications and local gradient steps [44, 35, 36], or use dual formulations and formulate the consensus constraint using gossip matrices to obtain decentralized dual or primal-dual algorithms [48, 29, 22, 23, 37, 1]. We refer the reader to [45] for a broader survey on decentralized optimization. Our algorithms are based on the general analysis of decentralized SGD in [36].

LDP and amplification mechanisms. Limitations of LDP for computing the average of the private values of n users have been studied, showing that for a fixed privacy budget, the expected squared error in LDP is n times larger than in central DP [11]. More generally, LDP is also known to significantly reduce utility for many learning problems [54, 50], which motivates the study of intermediate trust

[1] The name is borrowed from the Harry Potter series: it designates a "spell that filled the ears of anyone nearby with an unidentifiable buzzing", thereby concealing messages from unintended listeners through noise injection.

models. Cryptographic primitives, such as secure aggregation [19, 49, 8, 12, 32, 4, 47] and secure shuffling [14, 21, 3, 28, 27], as well as additional mechanisms such as amplification by subsampling [2] or amplification by iteration [25], can offer better utility for some applications, but cannot be easily applied in a fully decentralized setting, as they require coordination by a central server.

Amplification through decentralization. The idea that decentralized communications can provide differential privacy guarantees was initiated by [6] in the context of rumor spreading. Closer to our work, [15] showed privacy amplification for random walk algorithms on complete graphs, where the model is transmitted from one node to another sequentially. While we build on their notion of Network DP, our work differs from [15] in several aspects: (i) our analysis holds for any graph and explicitly quantifies its effect, (ii) instead of worst-case privacy across all pairs of nodes, we prove pairwise guarantees that are stronger for nodes that are far away from each other, and (iii) unlike random walk approaches, gossip algorithms allow parallel computation and thus better scalability.

2 Setting and Pairwise Network Differential Privacy

We study a decentralized model where n nodes (users) hold private datasets and communicate through gossip protocols, that we describe in Section 2.1.In Section 2.2, we recall differential privacy notions and the two natural baselines for our work, central and local DP. Finally, we introduce in Section 2.3 the relaxation of local DP used throughout the paper: the *pairwise network DP*.

2.1 Gossip Algorithms

We consider a connected graph $G = (\mathcal{V}, \mathcal{E})$ on a set $\mathcal{V}$ of n users. An edge $\{u, v\} \in \mathcal{E}$ indicates that u and v can communicate (we say they are neighbors). Each user $v \in \mathcal{V}$ holds a local dataset $\mathcal{D}_v$ and we aim at computing averages of private values. This averaging step is a key building block for solving machine learning problems in a decentralized manner, as will be discussed in Section 4. From a graph, we derive a gossip matrix.

Definition 1 (Gossip matrix)**.** *A gossip matrix over a graph G is a* symmetric *matrix $W \in \mathbb{R}^{\mathcal{V}\times\mathcal{V}}$ with non-negative entries, that satisfies $W\mathbb{1} = \mathbb{1}$* i.e. *W is stochastic ($\mathbb{1} \in \mathbb{R}^{\mathcal{V}}$ is the vector with all entries equal to 1), and such that for any $u, v \in \mathcal{V}$, $W_{u,v} > 0$ implies that $\{u, v\} \in \mathcal{E}$ or $u = v$.*

The iterates of synchronous gossip [16] are generated through a recursion of the form $x^{t+1} = Wx^t$, and converge to the mean of initial values at a linear rate $e^{-t\lambda_W}$, with λ_W defined below.

Definition 2 (Spectral gap)**.** *The spectral gap λ_W associated with a gossip matrix W is $\min_{\lambda\in \mathrm{Sp}(W)\setminus\{1\}}(1 - |\lambda|)$, where $\mathrm{Sp}(W)$ is the spectrum of W.*

The inverse of λ_W is the relaxation time of the random walk on G with transition probabilities W, and is closely related to the connectivity of the graph: adding edges improve mixing properties (λ_W increases), but can reduce scalability by increasing node degrees (and thus the per-iteration communication complexity). The rate of convergence can be accelerated to $e^{-t\sqrt{\lambda_W}}$ using re-scaled Chebyshev polynomials, leading to iterates of the form $x^t = P_t(W)x^0$ [7].

Definition 3 (Re-scaled Chebyshev polynomials)**.** *The re-scaled Chebyshev polynomials $(P_t)_{t\geqslant 0}$ with scale parameter $\gamma \in [1, 2]$ are defined by second-order linear recursion:*

$$P_0(X) = 1\,, \quad P_1(X) = X\,, \quad P_{t+1}(X) = \gamma X P_t(X) + (1-\gamma)P_{t-1}(X)\,,\, t \geqslant 2\,. \tag{1}$$

2.2 Rényi Differential Privacy

Differential Privacy (DP) quantifies how much the output of an algorithm $\mathcal{A}$ leaks about the dataset taken as input [18]. DP requires to define an adjacency relation between datasets. In this work, we adopt a user-level relation [42] which aims to protect the whole dataset $\mathcal{D}_v$ of a given user represented by a node $v \in \mathcal{V}$. Formally, $\mathcal{D} = \cup_{v\in\mathcal{V}}\mathcal{D}_v$ and $\mathcal{D}' = \cup_{v\in\mathcal{V}}\mathcal{D}'_v$ are adjacent datasets, denoted by $\mathcal{D} \sim \mathcal{D}'$, if there exists $v \in \mathcal{V}$ such that only $\mathcal{D}_v$ and $\mathcal{D}'_v$ differ. We use $\mathcal{D} \sim_v \mathcal{D}'$ to denote that $\mathcal{D}$ and $\mathcal{D}'$ differ only in the data of user v.

We use Rényi Differential Privacy (RDP) [43] to measure the privacy loss, which allows better and simpler composition than the classical (ε, δ)-DP. Note that any (α, ε)-RDP algorithm is also $(\varepsilon + \ln(1/\delta)/(\alpha - 1), \delta)$-DP for any $0 < \delta < 1$ [43].

Definition 4 (Rényi Differential Privacy)**.** *An algorithm $\mathcal{A}$ satisfies (α, ε)-Rényi Differential Privacy (RDP) for $\alpha > 1$ and $\varepsilon > 0$ if for all pairs of neighboring datasets $\mathcal{D} \sim \mathcal{D}'$:*

$$D_\alpha\left(\mathcal{A}(\mathcal{D})||\mathcal{A}(\mathcal{D}')\right) \leqslant \varepsilon\,, \tag{2}$$

where for two random variables X and Y, $D_\alpha\big(X\,||\,Y\big)$ is the Rényi divergence *between X and Y:*

$$D_\alpha\big(X\,||\,Y\big) = \tfrac{1}{\alpha-1}\ln\int\left(\tfrac{\mu_X(z)}{\mu_Y(z)}\right)^\alpha \mu_Y(z)dz\,.$$

with μ_X and μ_Y the respective densities of X and Y.

Without loss of generality, we consider gossip algorithms with a single real value per node (in that case, $\mathcal{D}_v = \{x_v\}$ for some $x_v \in \mathbb{R}$), and we aim at computing a private estimation of the mean $\bar{x} = (1/n)\sum_v x_v$. The generalization to vectors is straightforward, as done subsequently for optimization in Section 4. In general, the value of a (scalar) function g of the data can be privately released using the Gaussian mechanism [18, 43], which adds $\eta \sim \mathcal{N}(0, \sigma^2)$ to $g(\mathcal{D})$. It satisfies $(\alpha, \alpha\Delta_g^2/(2\sigma^2))$-RDP for any $\alpha > 1$, where $\Delta_g = \sup_{\mathcal{D}\sim\mathcal{D}'}\|g(\mathcal{D}) - g(\mathcal{D}')\|$ is the sensitivity of g. We focus on the Gaussian mechanism for its simplicity (similar results could be derived for other DP mechanisms), and thus assume an upper bound on the L_2 inputs sensitivity.

Assumption 1. *There exists some constant $\Delta > 0$ such that for all $u \in \mathcal{V}$ and for any adjacent datasets $\mathcal{D} \sim_u \mathcal{D}'$, we have $\|x_u - x'_u\| \leqslant \Delta$.*

In central DP, a trusted aggregator can first compute the mean $\bar{x}$ (which has sensitivity Δ/n) and then reveal a noisy version with the Gaussian mechanism. On the contrary, in local DP where there is no trusted aggregator and everything that a given node reveals can be observed, each node must locally perturb its input (which has sensitivity Δ), deteriorating the privacy-utility trade-off. Formally, to achieve (α, ε)-DP, one cannot have better utility than:

$$\mathbb{E}\left[\left\|x^{\text{out}} - \bar{x}\right\|^2\right] \leqslant \frac{\alpha\Delta^2}{2n\varepsilon} \quad \text{for local DP}\,, \quad \text{and} \quad \mathbb{E}\left[\left\|x^{\text{out}} - \bar{x}\right\|^2\right] \leqslant \frac{\alpha\Delta^2}{2n^2\varepsilon} \quad \text{for central DP}\,,$$

where x^{out} is the output of the algorithm. This $1/n$ gap motivates the study of relaxations of local DP.

2.3 Pairwise Network Differential Privacy

We relax local DP to take into account privacy amplification between nodes that are distant from each other in the graph. We define a decentralized algorithm $\mathcal{A}$ as a randomized mapping that takes as input a dataset $\mathcal{D} = \cup_{v\in\mathcal{V}}(\mathcal{D}_v)$ and outputs the transcript of all messages exchanged between users in the network. A message between neighboring users $\{u, v\} \in \mathcal{E}$ at time t is characterized by the tuple $(u, m(t), v)$: user u sent a message with content $m(t)$ to user v, and $\mathcal{A}(\mathcal{D})$ is the set of all these messages. Each node v only has a partial knowledge of $\mathcal{A}(\mathcal{D})$, captured by its *view*:

$$\mathcal{O}_v\big(\mathcal{A}(\mathcal{D})\big) = \{(u, m(t), v) \in \mathcal{A}(\mathcal{D}) \quad \text{such that} \quad \{u, v\} \in \mathcal{E}\}\,.$$

This subset corresponds to direct interactions of v with its neighbors, which provide only an indirect information on computations in others parts of the graph. Thus, we seek to express privacy constraints that are personalized for each pair of nodes. This is captured by our notion of Pairwise Network DP.

Definition 5 (Pairwise Network DP)**.** *For $f : \mathcal{V} \times \mathcal{V} \to \mathbb{R}^+$, an algorithm $\mathcal{A}$ satisfies (α, f)-Pairwise Network DP (PNDP) if for all pairs of distinct users $u, v \in \mathcal{V}$ and neighboring datasets $D \sim_u D'$:*

$$D_\alpha\big(\mathcal{O}_v(\mathcal{A}(\mathcal{D}))\,||\,\mathcal{O}_v(\mathcal{A}(\mathcal{D}'))\big) \leqslant f(u, v)\,. \tag{3}$$

We note $\varepsilon_{u\to v} = f(u, v)$ the privacy leaked to v from u and say that u is $(\alpha, \varepsilon_{u\to v})$-PNDP with respect to v if only inequality (3) holds for $f(u, v) = \varepsilon_{u\to v}$.

By taking f constant in Definition 5, we recover the definition of Network DP [15]. Our pairwise variant refines Network DP by allowing the privacy guarantee to depend on u and v (typically, on their distance in the graph). We assume that users are *honest but curious*: they truthfully follow the protocol, but may try to derive as much information as possible from what they observe. We refer to Appendix G for an adaptation of our definition and results to the presence of colluding nodes.

In addition to pairwise guarantees, we will use the *mean privacy loss* $\overline{\varepsilon}_v = \frac{1}{n}\sum_{u\in\mathcal{V}\setminus\{v\}} f(u, v)$ to compare with baselines LDP and trusted aggregator by enforcing $\overline{\varepsilon} = \max_{v\in\mathcal{V}}\overline{\varepsilon}_v \leqslant \varepsilon$. The value $\overline{\varepsilon}_v$ is the average of the privacy loss from all the nodes to v and thus does not correspond to a proper privacy guarantee, but it is a convenient way to summarize our gain, noting that distant nodes — in ways that will be specified — will have better privacy guarantee than this average, while worst cases will remain bounded by the baseline LDP guarantee provided by local noise injection.

Algorithm 1: MUFFLIATO

Input: local values $(x_v)_{v\in\mathcal{V}}$ to average, gossip matrix W on a graph G, in T iterations, noise variance σ^2

$\gamma \leftarrow 2\frac{1-\sqrt{\lambda_W(1-\frac{\lambda_W}{4})}}{(1-\lambda_W/2)^2}$

for *all nodes v in parallel* **do**
  $x_v^0 \leftarrow x_v + \eta_v$ where $\eta_v \sim \mathcal{N}(0,\sigma^2)$
for $t = 0$ *to* $T-1$ **do**
  for *all nodes v in parallel* **do**
    for *all neighbors w defined by W* **do**
      Send x_v^t, receive x_w^t
    $x_v^{t+1} \leftarrow (1-\gamma)x_v^{t-1} + \gamma\sum_{w\in\mathcal{N}_v} W_{v,w}x_w^t$

Algorithm 2: RANDOMIZED MUFFLIATO

Input: local values $(x_v)_{v\in\mathcal{V}}$ to average, activation intensities $(p_{\{v,w\}})_{\{v,w\}\in\mathcal{E}}$, in T iterations, noise variance σ^2

for *all nodes v in parallel* **do**
  $x_v^0 \leftarrow x_v + \eta_v$ where $\eta_v \sim \mathcal{N}(0,\sigma^2)$
for $t = 0$ *to* $T-1$ **do**
  Sample $\{v_t, w_t\} \in \mathcal{E}$ with probability $p_{\{v_t,w_t\}}$
  v_t and w_t exchange $x_{v_t}^t$ and $x_{w_t}^t$
  Local averaging: $x_{v_t}^{t+1} = x_{w_t}^{t+1} = \frac{x_{v_t}^{t+1}+x_{w_t}^{t+1}}{2}$
  For $v \in \mathcal{V}\setminus\{v_t,w_t\}$, $x_v^{t+1} = x_v^t$

3 Private Gossip Averaging

In this section, we analyze a generic algorithm with arbitrary time-varying communication matrices for averaging. Then, we instantiate and discuss these results for synchronous communications with a fixed gossip matrix, communications using randomized gossip [10], and with Erdös-Rényi graphs.

3.1 General Privacy Analysis of Gossip Averaging

We consider gossip over time-varying graphs $(G_t)_{0\leqslant t\leqslant T}$, defined as $G_t = (\mathcal{V}, \mathcal{E}_t)$, with corresponding gossip matrices $(W_t)_{0\leqslant t\leqslant T}$. The *generic Muffliato* algorithm $\mathcal{A}^T$ over T iterations for averaging $x = (x_v)_{v\in\mathcal{V}}$ corresponds to an initial noise addition followed by gossip steps. Writing $W_{0:t} = W_{t-1}\dots W_0$, the iterates of $\mathcal{A}^T$ are thus defined by:

$$\forall v \in \mathcal{V}, x_v^0 = x_v + \eta_v \text{ with } \eta_v \sim \mathcal{N}(0,\sigma^2), \quad \text{and } x^{t+1} = W_t x^t = W_{0:t+1}(x+\eta)\,. \tag{4}$$

Note that the update rule at node $v\in\mathcal{V}$ writes as $x_v^{t+1} = \sum_{w\in\mathcal{N}_t(v)}(W_t)_{v,w}x_w^t$ where $\mathcal{N}_t(v)$ are the neighbors of v in G_t, so for the privacy analysis, the view of a node is:

$$\mathcal{O}_v\big(\mathcal{A}^T(\mathcal{D})\big) = \big\{\big(W_{0:t}(x+\eta)\big)_w \,\big|\quad \{v,w\}\in\mathcal{E}_t\,,\quad 0\leqslant t\leqslant T-1\big\} \cup \{x_v\}\,. \tag{5}$$

Theorem 1. *Let $T \geqslant 1$ and denote by $\mathcal{P}^T_{\{v,w\}} = \{s < T : \{v,w\}\in\mathcal{E}_s\}$ the set of time-steps with communication along edge $\{v,w\}$. Under Assumption 1, $\mathcal{A}^T$ is (α, f)-PNDP with:*

$$f(u,v) = \frac{\alpha\Delta^2}{2\sigma^2}\sum_{w\in\mathcal{V}}\sum_{t\in\mathcal{P}^T_{\{v,w\}}}\frac{(W_{0:t})^2_{u,w}}{\|(W_{0:t})_w\|^2}\,. \tag{6}$$

This theorem, proved in Appendix B, gives a tight computation of the privacy loss between every pair of nodes and can easily be computed numerically (see Section 5). Since distant nodes correspond to small entries in $W_{0:t}$, Equation 6 suggests that they reveal less to each other. We will characterize this precisely for the case of fixed communication graph in the next subsection.

Another way to interpret the result of Theorem 1 is to derive the corresponding mean privacy loss:

$$\overline{\varepsilon}_v = \frac{\alpha\Delta^2 T_v}{2n\sigma^2}\,,$$

where T_v is the total number of communications node v was involved with up to time T. Thus, in comparison with LDP, the mean privacy towards v is n/T_v times smaller. In other words, a node learns much less than in LDP as long as it communicates $o(n)$ times.

3.2 Private Synchronous *Muffliato*

We now consider *Muffliato* over a fixed graph (Algorithm 1) and start by analyzing its utility. The utility decomposes as an averaging error term vanishing exponentially fast, and a *bias* term due to the noise. General convergence rates are given in Appendix C, from which we extract the following result.

Table 1: Utility of *Muffliato* for several topologies under the constraint $\bar{\varepsilon} \leqslant \varepsilon$ for the classic gossip matrix where $W_{v,w} = \min(1/d_v, 1/d_w)$ and d_v is the degree of node v. Constant and logarithmic factors are hidden. Recall that utility is $\alpha\Delta^2/n\varepsilon$ for LDP and $\alpha\Delta^2/n^2\varepsilon$ for a trusted aggregator.

Graph	Arbitrary	Expander	D-Torus	Complete	Ring
Algorithm 1	$\frac{\alpha\Delta^2 d}{n^2\varepsilon\sqrt{\lambda_W}}$	$\frac{\alpha\Delta^2}{n^2\varepsilon}$	$\frac{\alpha\Delta^2 D}{n^{2-1/D}\varepsilon}$	$\frac{\alpha\Delta^2}{n\varepsilon}$	$\frac{\alpha\Delta^2}{n\varepsilon}$
Algorithm 2	$\frac{\alpha\Delta^2}{n^2\varepsilon\lambda_W}$	$\frac{\alpha\Delta^2}{n^2\varepsilon}$	$\frac{\alpha\Delta^2}{n^{2-2/D}\varepsilon}$	$\frac{\alpha\Delta^2}{n^2\varepsilon}$	$\frac{\alpha\Delta^2}{n\varepsilon}$

Theorem 2 (Utility analysis)**.** *Let λ_W be the spectral gap of W. Muffliato (Algorithm 1) verifies:*

$$\frac{1}{2n}\sum_{v\in\mathcal{V}}\mathbb{E}\left[\left\|x_v^{T^{\text{stop}}} - \bar{x}\right\|^2\right] \leqslant \frac{3\sigma^2}{n}\,, \quad \textit{for } T^{\text{stop}} \leqslant \frac{1}{\sqrt{\lambda_W}}\ln\left(\frac{n}{\sigma^2}\max\left(\sigma^2, \frac{1}{n}\sum_{v\in\mathcal{V}}\left\|x_v^0 - \bar{x}\right\|^2\right)\right).$$

For the privacy guarantees, Theorem 1 still holds as accelerated gossip can be seen as a simple post-processing of the non-accelerated version. We can derive a more explicit formula.

Corollary 1. *Algorithm 1 satisfies $(\alpha, \varepsilon^T_{u\to v}(\alpha))$-PNDP for node u with respect to v, with:*

$$\varepsilon^T_{u\to v}(\alpha) \leqslant \frac{\alpha\Delta^2 n}{2\sigma^2}\max_{\{v,w\}\in\mathcal{E}} W_{v,w}^{-2}\sum_{t=1}^{T}\mathbb{P}\left(X^t = v|X^0 = u\right)^2\,,$$

where $(X^t)_t$ is the random walk on graph G, with transitions W.

This result allows us to directly relate the privacy loss from u to v to the probability that the random walk on G with transition probabilities given by the gossip matrix W goes from u to v in a certain number of steps. It thus captures a notion of distance between nodes in the graph. We also report the utility under fixed mean privacy loss $\bar{\varepsilon} \leqslant \varepsilon$ in Table 1 for various graphs, where one can see a utility-privacy trade-off improvement of $n\sqrt{\lambda_W}/d$, where d is the maximum degree, compared to LDP. Using expanders closes the gap with a trusted aggregator up to constant and logarithmic terms. Remarkably, we see that topologies that make gossip averaging efficient (i.e. with big $\sqrt{\lambda_W}/d$), such as exponential graphs or hypercubes [52], are also the ones that achieve optimal privacy amplification (up to logarithmic factors). In other words, *privacy, utility and scalability are compatible*.

3.3 Private Randomized *Muffliato*

Synchronous protocols require global coordination between nodes, which can be costly or even impossible. On the contrary, asynchronous protocols only requires separated activation of edges: they are thus are more resilient to stragglers nodes and faster in practice. In asynchronous gossip, at a given time-step a single edge $\{u, v\}$ is activated independently from the past with probability $p_{\{u,v\}}$, as described by Boyd et al. [10]. In our setting, randomized *Muffliato* (Algorithm 2) corresponds to instantiate our general analysis with $W^t = W_{\{v_t,w_t\}} = I_n - (e_{v_t} - e_{w_t})(e_{v_t} - e_{w_t})^\top/2$ if $\{v_t, w_t\}$ is sampled at time t. The utility analysis is similar to the synchronous case.

Theorem 3 (Utility analysis)**.** *Let $\lambda(p)$ be the spectral gap of graph G with weights $(p_{\{v,w\}})_{\{v,w\}\in\mathcal{E}}$. Randomized* Muffliato *(Algorithm 2) verify:*

$$\frac{1}{2n}\sum_{v\in\mathcal{V}}\mathbb{E}\left[\left\|x_v^{T^{\text{stop}}} - \bar{x}\right\|^2\right] \leqslant \frac{2\sigma^2}{n}\,, \quad \textit{for } T^{\text{stop}} \leqslant \frac{1}{\lambda(p)}\ln\left(\frac{n}{\sigma^2}\max\left(\sigma^2, \frac{1}{n}\sum_{v\in\mathcal{V}}\left\|x_v^0 - \bar{x}\right\|^2\right)\right).$$

To compare with synchronous gossip (Algorithm 1), we note that activation probabilities can be derived from a gossip matrix W by taking $p_{\{u,v\}} = 2W_{\{u,v\}}/n$ implying that $\lambda(p) = 2\lambda_W/n$, thus requiring n times more iterations to reach the same utility than by applying in a synchronous way matrix W. However, for a given time-horizon T and node v, the number of communications v can be bounded with high probability by a T/n multiplied by a constant whereas Algorithm 1 requires $d_v T$ communications. Consequently, as reported in Table 1, for a fixed privacy mean $\bar{\varepsilon}_v$, Algorithm 2 has the same utility as Algorithm 1, up to two differences: the degree factor d_v is removed, while $\sqrt{\lambda_W}$

degrades to λ_W as we do not accelerate randomized gossip.[2] Randomized gossip can thus achieve an optimal privacy-utility trade-off with large-degree graphs, as long as the spectral gap is small enough.

3.4 Erdös-Rényi Graphs

So far the graph was considered to be public and the amplification only relied on the secrecy of the messages. In practice, the graph may be sampled randomly and the nodes need only to know their direct neighbors. We show that we can leverage this through the weak convexity of Rényi DP to amplify privacy between non-neighboring nodes. We focus on Erdös-Rényi graphs, which can be built without central coordination by picking each edge independently with the same probability q. For $q = c\ln(n)/n$ where $c > 1$, Erdös-Rényi graphs are good expanders with node degrees $d_v = \mathcal{O}(\log n)$ and λ_W concentrating around 1 [30], and we obtain the following privacy guarantee.

Theorem 4 (*Muffliato* on a random graph)**.** *Let $\alpha > 1$, $T \geqslant 0$, $\sigma^2 \geqslant \frac{\Delta^2\alpha(\alpha-1)}{2}$ and $q = c\frac{\ln(n)}{n}$ for $c > 1$. Let $u, v \in \mathcal{V}$ be distinct nodes. After running Algorithm 1 with these parameters, node u is $(\alpha, \varepsilon^T_{u\to v}(\alpha))$-PNDP with respect to v, with:*

$$\varepsilon^T_{u\to v}(\alpha) \leqslant \begin{cases} \dfrac{\alpha\Delta^2}{2\sigma^2} & \textit{with probability } q\,, \\ \dfrac{\alpha\Delta^2}{\sigma^2}\dfrac{Td_v}{n-d_v} & \textit{with probability } 1-q\,. \end{cases}$$

This results shows that with probability q, u and v are neighbors and there is no amplification compared to LDP. The rest of the time, with probability $1-q$, the privacy matches that of a trusted aggregator up to a degree factor $d_v = \mathcal{O}(\log n)$ and $T = \tilde{\mathcal{O}}(1/\sqrt{\lambda_W}) = \tilde{\mathcal{O}}(1)$ [30].

4 Private Decentralized Optimization

We now build upon *Muffliato* to design decentralized optimization algorithms. Each node $v \in \mathcal{V}$ possesses a data-dependent function $\phi_v : \mathbb{R}^d \to \mathbb{R}$ and we wish to *privately* minimize the function

$$\phi(\theta) = \frac{1}{n}\sum_{v\in\mathcal{V}}\phi_v(\theta)\,, \quad \text{with } \phi_v(\theta) = \frac{1}{|\mathcal{D}_v|}\sum_{x_v\in\mathcal{D}_v}\ell_v(\theta, x_v)\,, \quad \theta\in\mathbb{R}^d\,, \tag{7}$$

where $\mathcal{D}_v$ is the (finite) dataset corresponding to user v for data lying in a space $\mathcal{X}_v$, and $\ell_v : \mathbb{R}^d \times \mathcal{X}_v \to \mathbb{R}$ a loss function. We assume that ϕ is μ-strongly convex, and each ϕ_v is L-smooth, and denote $\kappa = L/\mu$. Denoting by $\theta^\star$ the minimizer of ϕ, for some non-negative $(\zeta_v^2)_{v\in\mathcal{V}}$, $(\rho_v^2)_{v\in\mathcal{V}}$ and all $v \in \mathcal{V}$, we assume:

$$\|\nabla\phi_v(\theta^\star) - \nabla\phi(\theta^\star)\|^2 \leqslant \zeta_v^2 \quad , \quad \mathbb{E}\left[\|\nabla\ell_v(\theta^\star, x_v) - \nabla\phi(\theta^\star)\|^2\right] \leqslant \rho_v^2\,, \quad x_v \sim \mathcal{L}_v\,,$$

where $\mathcal{L}_v$ is the uniform distribution over $\mathcal{D}_v$. We write $\bar\rho^2 = \frac{1}{n}\sum_{v\in\mathcal{V}}\rho_v^2$ and $\bar\zeta^2 = \frac{1}{n}\sum_{v\in\mathcal{V}}\zeta_v^2$.

We introduce Algorithm 3, a private version of the classical decentralized SGD algorithm studied in [36]. Inspired by the optimal algorithm MSDA of Scaman et al. [48] that alternates between K Chebychev gossip communications and expensive dual gradient computations, our Algorithm 3 alternates between K Chebychev communications and local stochastic gradient steps. This alternation reduces the total number of gradients leaked, a crucial point for achieving good privacy. Note that in Algorithm 3, each communication round uses a potentially different gossip matrix W_t. In the results stated below, we fix $W_t = W$ for all t and defer the more general case to Appendix F, where different independent Erdös-Rényi graphs with same parameters are used at each communication round.

Remark 1. *Our setting encompasses both GD and SGD.* MUFFLIATO-GD *is obtained by removing the stochasticity,* i.e.*, setting $\ell_v(\cdot) = \phi_v(\cdot)$. In that case, $\bar\rho^2 = 0$.*

Theorem 5 (Utility analysis of Algorithm 3)**.** *For suitable step-size parameters, for a total number of T^{stop} computations and $T^{\text{stop}}K$ communications, with:*

$$T^{\text{stop}} = \tilde{\mathcal{O}}(\kappa)\,, \quad \textit{and} \quad K = \left\lceil \sqrt{\lambda_W}^{-1}\ln\left(\max\left(n, \frac{\bar\zeta^2}{\sigma^2+\bar\rho^2}\right)\right)\right\rceil\,,$$

[2] One could also accelerate randomized gossip as described by Even et al. [23], obtaining $\sqrt{\lambda(p)/|\mathcal{E}|}$ instead of $\lambda(p)$ in all our results.

Algorithm 3: MUFFLIATO-SGD and MUFFLIATO-GD

Input: initial points θ_i^0, number of iterations T, step sizes $\nu > 0$, noise variance $\sigma \geqslant 0$, mixing matrices $(W_t)_{t\geqslant 0}$, local functions ϕ_v, number of communication rounds K

for $t = 0$ *to* $T-1$ **do**

  for *all nodes* v *in parallel* **do**

    Compute $\hat{\theta}_v^t = \theta_v^t - \nu \nabla_\theta \ell_v(\theta_v^t, x_v^t)$ where $x_v^t \sim \mathcal{L}_v$

  $\theta_v^{t+1} = \text{MUFFLIATO}\big((\hat{\theta}_v^t)_{v\in\mathcal{V}}, W_t, K, \nu^2\sigma^2\big)$

the iterates $(\theta^t)_{t\geqslant 0}$ *generated by Algorithm 3 verify* $\mathbb{E}\left[\phi(\tilde{\theta}^{\text{out}}) - \phi(\theta^\star)\right] = \tilde{\mathcal{O}}(\frac{\sigma^2+\bar{\rho}^2}{\mu T^{\text{stop}}})$ *where* $\tilde{\theta}^{\text{out}}$ *is a weighted average of the* $\bar{\theta}^t = \frac{1}{n}\sum_{v\in\mathcal{V}} \theta_v^t$ *until* T^{stop}.

For the following privacy analysis, we need a bound on the sensitivity of gradients with respect to the data. To this end, we assume that for all v and x_v, $\ell_v(\cdot, x_v)$ is $\Delta_\phi/2$ Lipschitz[3].

Theorem 6 (Privacy analysis of Algorithm 3)**.** *Let u and v be two distinct nodes in $\mathcal{V}$. After T iterations of Algorithm 3 with $K \geqslant 1$, node u is $(\varepsilon^T_{u\to v}(\alpha), \alpha)$-PNDP with respect to v, with:*

$$\varepsilon^T_{u\to v}(\alpha) \leqslant \frac{T\Delta_\phi^2 \alpha}{2\sigma^2} \sum_{k=0}^{K-1} \sum_{w:\{v,w\}\in\mathcal{E}} \frac{(W^k)^2_{u,w}}{\|(W^k)_w\|^2} . \tag{8}$$

Thus, for any $\varepsilon > 0$, Algorithm 3 with $T^{\text{stop}}(\kappa, \sigma^2, n)$ steps and for K as in Theorem 5, there exists f such that the algorithm is (α, f)-pairwise network DP, with:

$$\forall v \in \mathcal{V}, \quad \bar{\varepsilon}_v \leqslant \varepsilon \quad \textit{and} \quad \mathbb{E}\left[\phi(\tilde{\theta}^{\text{out}}) - \phi(\theta^\star)\right] \leqslant \tilde{\mathcal{O}}\left(\frac{\alpha\Delta_\phi^2 d_v}{n\mu\varepsilon\sqrt{\lambda_W}} + \frac{\bar{\rho}^2}{nL}\right) .$$

The term $\frac{\bar{\rho}^2}{nL}$ above is privacy independent, and typically dominated by the first term. Comparing Theorem 6 with the privacy guarantees of *Muffliato* (Section 3.2), the only difference lies in the factor Δ_ϕ^2/μ. While Δ_ϕ^2 plays the role of the sensitivity Δ^2, μ is directly related to the complexity of the optimization problem through the condition number κ: the easier the problem is, the more private our algorithm becomes. Finally, the same discussion as after Corollary 1 applies here, up to the above optimization-related factors that do not affect the influence of the graph.

5 Experiments

In this section, we show that pairwise network DP provides significant privacy gains in practice even for moderate size graphs. We use synthetic graphs and real-world graphs for gossip averaging. For decentralized optimization, we solve a logistic regression problem on real-world data with time-varying Erdos-Renyi graphs, showing in each case clear gains of privacy compared to LDP.

Synthetic graphs. We generate synthetic graphs with $n = 2048$ nodes and define the corresponding gossip matrix according to the Hamilton scheme. Note that the privacy guarantees of *Muffliato* are deterministic for a fixed W, and defined by Equation 4. For each graph, we run *Muffliato* for the theoretical number of steps required for convergence, and report in Figure 1(a) the pairwise privacy guarantees aggregated by shortest path lengths between nodes, along with the LDP baseline for comparison. *Exponential graph* (generalized hypercubes): this has shown to be an efficient topology for decentralized learning [52]. Consistently with our theoretical result, privacy is significiantly amplified. The shortest path completely defines the privacy loss, so there is no variance. *Erdos-Renyi graph* with $q = c\log n/n$ $(c \geqslant 1)$ [20], averaged over 5 runs: this has nearly the same utility-privacy trade-off as the exponential graph but with significant variance, which motivates the time-evolving version mentioned in Section 4. *Grid:* given its larger mixing time, it is less desirable than the two previous graphs, emphasizing the need for careful design of the communication graph. *Geometric random graph:* two nodes are connected if and only if their distance is below a given threshold, which models for instance Bluetooth communications (effective only in a certain radius). We sample

[3]This assumption can be replaced by the more general Assumption 2 given in Appendix F

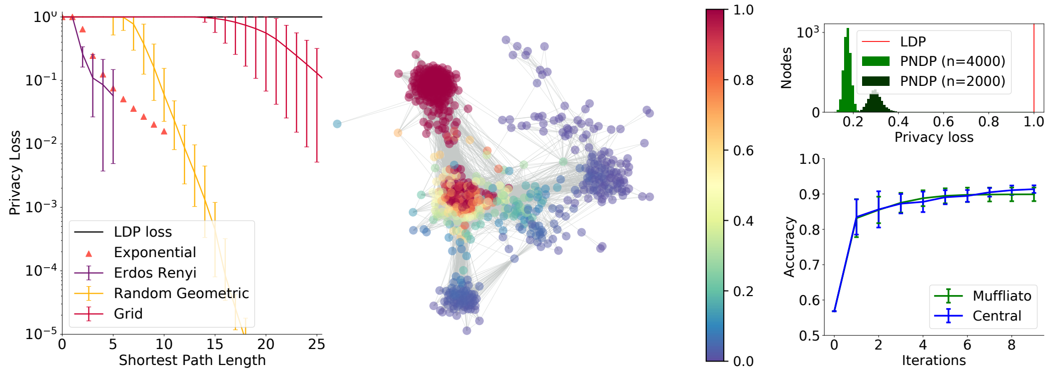

Figure 1: (a) Left: Privacy loss of *Muffliato* in pairwise NDP on synthetic graphs (best, worst and average in error bars over nodes at a given distance), confirming a significant privacy amplification as the distance increases. (b) Middle: Privacy loss of *Muffliato* from a node chosen at random on a Facebook ego graph, showing that leakage is limited outside the node's own community. (c) Right: Privacy loss and utility of *Muffliato*-GD compared to a baseline based on a trusted aggregator.

nodes uniformly at random in the square unit and choose a radius ensuring full connectivity. While the shortest path is a noisy approximation of the privacy loss, the Euclidean distance is a very good estimator as shown in Appendix H.

Real-world graphs. We consider the graphs of the Facebook ego dataset [38], where nodes are the friends of a given user (this central user is not present is the graph) and edges encode the friendship relation between these nodes. Ego graphs typically induce several clusters corresponding to distinct communities: same high school, same university, same hobbies... For each graph, we extract the giant connected component, choose a user at random and report its privacy loss with respect to other nodes. The privacy loss is often limited to the cluster of direct neighbors and fades quickly in the other communities, as seen in Figure 1(b). We observe this consistently across other ego graphs (see Appendix H). This is in line with one of our initial motivation: our pairwise guarantees are well suited to situations where nodes want stronger privacy with respect to distant nodes.

Logistic regression on real-world data. Logistic regression corresponds to minimizing Equation 7 with $\ell(\theta; x, y) = \ln(1 + \exp(-y\theta^\top x))$ where $x \in \mathbb{R}^d$ and $y \in \{-1, 1\}$. We use a binarized version of UCI Housing dataset.[4] We standardize the features and normalize each data point x to have unit L_2 norm so that the logistic loss is 1-Lipschitz for any (x, y). We split the dataset uniformly at random into a training set (80%) and a test set and further split the training set across users. For each gossiping step, we draw at random an Erdos-Renyi graph of same parameter q and run the theoretical number of steps required for convergence. For each node, we keep track of the privacy loss towards the first node (note that all nodes play the same role). We compute an equivalent in federated learning setting as drawn in Figure 1(c), where updates are aggregated by a trusted central server, with the same parameters, showing that we do observe the same behavior. We report the privacy loss per node for $n = 2000$ and $n = 4000$, showing clear gains over LDP that increase with the number of nodes.

6 Conclusion

We showed that gossip protocols amplify the LDP guarantees provided by local noise injection as values propagate in the graph. Despite the redundancy of gossip that, at first sight could be seen, as an obstacle to privacy, the amplification turns out to be significant: it can nearly match the optimal privacy-utility trade-off of the trusted curator. From the fundamental building block — noise injection followed by gossip — that we analyzed under the name *Muffliato*, one can easily extend the analysis to other decentralized algorithms. Our results are motivated by the typical relation between proximity in the communication graph and lower privacy expectations. Other promising directions are to assume that close people are more similar, which leads to smaller individual privacy accounting [24], or to design new notions of similarity between nodes in graphs that match the privacy loss variations.

[4] https://www.openml.org/d/823

References

[1] Sulaiman A. Alghunaim and Ali H. Sayed. Linear convergence of primal-dual gradient methods and their performance in distributed optimization, 2019. URL https://arxiv.org/abs/1904.01196.

[2] Borja Balle, Gilles Barthe, and Marco Gaboardi. Privacy Amplification by Subsampling: Tight Analyses via Couplings and Divergences. In *NeurIPS*, 2018.

[3] Borja Balle, James Bell, Adrià Gascón, and Kobbi Nissim. Differentially Private Summation with Multi-Message Shuffling. Technical report, arxiv:1906.09116, 2019.

[4] James Henry Bell, Kallista A. Bonawitz, Adrià Gascón, Tancrède Lepoint, and Mariana Raykova. Secure single-server aggregation with (poly)logarithmic overheads. In *CCS*, 2020.

[5] Aurélien Bellet, Rachid Guerraoui, Mahsa Taziki, and Marc Tommasi. Personalized and Private Peer-to-Peer Machine Learning. In *AISTATS*, 2018.

[6] Aurélien Bellet, Rachid Guerraoui, and Hadrien Hendrikx. Who started this rumor? Quantifying the natural differential privacy guarantees of gossip protocols. In *DISC*, 2020.

[7] Raphaël Berthier, Francis Bach, and Pierre Gaillard. Accelerated gossip in networks of given dimension using jacobi polynomial iterations. *SIAM Journal on Mathematics of Data Science*, 2(1): 24–47, 2020. doi: 10.1137/19M1244822. URL https://doi.org/10.1137/19M1244822.

[8] Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H. Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. Practical Secure Aggregation for Privacy-Preserving Machine Learning. In *CCS*, 2017.

[9] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah. Randomized gossip algorithms. *IEEE Transactions on Information Theory*, 52(6):2508–2530, 2006. doi: 10.1109/TIT.2006.874516.

[10] Stephen Boyd, Arpita Ghosh, Balaji Prabhakar, and Devavrat Shah. Randomized gossip algorithms. *IEEE transactions on information theory*, 52(6):2508–2530, 2006.

[11] T.-H. Hubert Chan, Elaine Shi, and Dawn Song. Optimal Lower Bound for Differentially Private Multi-party Aggregation. In *ESA*, 2012.

[12] T.-H. Hubert Chan, Elaine Shi, and Dawn Song. Privacy-preserving stream aggregation with fault tolerance. In *Financial Cryptography*, 2012.

[13] Hsin-Pai Cheng, Patrick Yu, Haojing Hu, Syed Zawad, Feng Yan, Shiyu Li, Hai Helen Li, and Yiran Chen. Towards Decentralized Deep Learning with Differential Privacy. In *CLOUD*, 2019.

[14] Albert Cheu, Adam D. Smith, Jonathan Ullman, David Zeber, and Maxim Zhilyaev. Distributed Differential Privacy via Shuffling. In *EUROCRYPT*, 2019.

[15] Edwige Cyffers and Aurélien Bellet. Privacy amplification by decentralization, 2020. URL https://arxiv.org/abs/2012.05326.

[16] A. G. Dimakis, S. Kar, J. M. F. Moura, M. G. Rabbat, and A. Scaglione. Gossip algorithms for distributed signal processing. *Proceedings of the IEEE*, 98(11):1847–1864, 2010.

[17] John C Duchi, Michael I Jordan, and Martin J Wainwright. Local privacy and statistical minimax rates. In *FOCS*, 2013.

[18] Cynthia Dwork and Aaron Roth. The Algorithmic Foundations of Differential Privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3-4):211–407, 2014.

[19] Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. Our Data, Ourselves: Privacy Via Distributed Noise Generation. In *EUROCRYPT*, 2006.

[20] P. Erdös and A. Rényi. On random graphs i. *Publicationes Mathematicae Debrecen*, 6:290, 1959.

[21] Ulfar Erlingsson, Vitaly Feldman, Ilya Mironov, Ananth Raghunathan, and Kunal Talwar. Amplification by Shuffling: From Local to Central Differential Privacy via Anonymity. In *SODA*, 2019.

[22] Mathieu Even, Hadrien Hendrikx, and Laurent Massoulié. Asynchrony and acceleration in gossip algorithms. Technical report, arXiv:2011.02379, 2020.

[23] Mathieu Even, Raphaël Berthier, Francis Bach, Nicolas Flammarion, Hadrien Hendrikx, Pierre Gaillard, Laurent Massoulié, and Adrien Taylor. Continuized accelerations of deterministic and stochastic gradient descents, and of gossip algorithms. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 28054–28066. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper/2021/file/ec26fc2eb2b75aece19c70392dc744c2-Paper.pdf.

[24] Vitaly Feldman and Tijana Zrnic. Individual Privacy Accounting via a Rényi Filter. In *NeurIPS*, 2021.

[25] Vitaly Feldman, Ilya Mironov, Kunal Talwar, and Abhradeep Thakurta. Privacy Amplification by Iteration. In *FOCS*, 2018.

[26] Vitaly Feldman, Ilya Mironov, Kunal Talwar, and Abhradeep Thakurta. Privacy amplification by iteration. *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, Oct 2018. doi: 10.1109/focs.2018.00056. URL http://dx.doi.org/10.1109/FOCS.2018.00056.

[27] Vitaly Feldman, Audra McMillan, and Kunal Talwar. Hiding Among the Clones: A Simple and Nearly Optimal Analysis of Privacy Amplification by Shuffling. Technical report, arXiv:2012.12803, 2020.

[28] Badih Ghazi, Noah Golowich, Ravi Kumar, Pasin Manurangsi, Rasmus Pagh, and Ameya Velingker. Pure Differentially Private Summation from Anonymous Messages. Technical report, arXiv:2002.01919, 2020.

[29] Hadrien Hendrikx, Francis Bach, and Laurent Massoulié. An accelerated decentralized stochastic proximal algorithm for finite sums. In *Advances in Neural Information Processing Systems*, 2019.

[30] Christopher Hoffman, Matthew Kahle, and Elliot Paquette. Spectral gaps of random graphs and applications. *International Mathematics Research Notices*, 2021(11):8353–8404, May 2019. ISSN 1687-0247. doi: 10.1093/imrn/rnz077. URL http://dx.doi.org/10.1093/imrn/rnz077.

[31] Zhenqi Huang, Sayan Mitra, and Nitin Vaidya. Differentially Private Distributed Optimization. In *ICDCN*, 2015.

[32] Bargav Jayaraman, Lingxiao Wang, David Evans, and Quanquan Gu. Distributed learning without distress: Privacy-preserving empirical risk minimization. In *NeurIPS*, 2018.

[33] Peter Kairouz, H. Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, Rafael G. L. D'Oliveira, Hubert Eichner, Salim El Rouayheb, David Evans, Josh Gardner, Zachary Garrett, Adrià Gascón, Badih Ghazi, Phillip B. Gibbons, Marco Gruteser, Zaid Harchaoui, Chaoyang He, Lie He, Zhouyuan Huo, Ben Hutchinson, Justin Hsu, Martin Jaggi, Tara Javidi, Gauri Joshi, Mikhail Khodak, Jakub Konecný, Aleksandra Korolova, Farinaz Koushanfar, Sanmi Koyejo, Tancrède Lepoint, Yang Liu, Prateek Mittal, Mehryar Mohri, Richard Nock, Ayfer Özgür, Rasmus Pagh, Hang Qi, Daniel Ramage, Ramesh Raskar, Mariana Raykova, Dawn Song, Weikang Song, Sebastian U. Stich, Ziteng Sun, Ananda Theertha Suresh, Florian Tramèr, Praneeth Vepakomma, Jianyu Wang, Li Xiong, Zheng Xu, Qiang Yang, Felix X. Yu, Han Yu, and Sen Zhao. Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning*, 14(1–2):1–210, 2021.

[34] Shiva Prasad Kasiviswanathan, Homin K. Lee, Kobbi Nissim, Sofya Raskhodnikova, and Adam D. Smith. What Can We Learn Privately? In *FOCS*, 2008.

[35] Anastasia Koloskova, Sebastian Stich, and Martin Jaggi. Decentralized stochastic optimization and gossip algorithms with compressed communication. In *International Conference on Machine Learning*, volume 97, pages 3478–3487. PMLR, 2019.

[36] Anastasia Koloskova, Nicolas Loizou, Sadra Boreiri, Martin Jaggi, and Sebastian Stich. A unified theory of decentralized SGD with changing topology and local updates. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 5381–5393. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/koloskova20a.html.

[37] Dmitry Kovalev, Elnur Gasanov, Alexander Gasnikov, and Peter Richtarik. Lower bounds and optimal algorithms for smooth and strongly convex decentralized optimization over time-varying networks. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 22325–22335. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper/2021/file/bc37e109d92bdc1ea71da6c919d54907-Paper.pdf.

[38] Jure Leskovec and Julian Mcauley. Learning to discover social circles in ego networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. URL https://proceedings.neurips.cc/paper/2012/file/7a614fd06c325499f1680b9896beedeb-Paper.pdf.

[39] Xiangru Lian, Ce Zhang, Huan Zhang, Cho-Jui Hsieh, Wei Zhang, and Ji Liu. Can decentralized algorithms outperform centralized algorithms? a case study for decentralized parallel stochastic gradient descent. In *NIPS*, 2017.

[40] Cassio G. Lopes and Ali H. Sayed. Incremental adaptive strategies over distributed networks. *IEEE Transactions on Signal Processing*, 55(8):4064–4077, 2007. doi: 10.1109/TSP.2007.896034.

[41] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-Efficient Learning of Deep Networks from Decentralized Data. In Aarti Singh and Jerry Zhu, editors, *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pages 1273–1282. PMLR, 20–22 Apr 2017.

[42] H. Brendan McMahan, Daniel Ramage, Kunal Talwar, and Li Zhang. Learning Differentially Private Recurrent Language Models. In *ICLR*, 2018.

[43] Ilya Mironov. Renyi differential privacy. *CoRR*, abs/1702.07476, 2017. URL http://arxiv.org/abs/1702.07476.

[44] Angelia Nedic and Asuman Ozdaglar. Distributed subgradient methods for multi-agent optimization. *IEEE Transactions on Automatic Control*, 54(1):48–61, 2009. doi: 10.1109/TAC.2008.2009515.

[45] Angelia Nedic, Alex Olshevsky, and Michael G. Rabbat. Network topology and communication-computation tradeoffs in decentralized optimization. *Proceedings of the IEEE*, 106(5):953–976, May 2018.

[46] Giovanni Neglia, Gianmarco Calbi, Don Towsley, and Gayane Vardoyan. The role of network topology for distributed machine learning. In *INFOCOM*, 2019.

[47] César Sabater, Aurélien Bellet, and Jan Ramon. Distributed Differentially Private Averaging with Improved Utility and Robustness to Malicious Parties. Technical report, arXiv:2006.07218, 2020.

[48] Kevin Scaman, Francis Bach, Sébastien Bubeck, Yin Tat Lee, and Laurent Massoulié. Optimal algorithms for smooth and strongly convex distributed optimization in networks. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 3027–3036. PMLR, 06–11 Aug 2017. URL https://proceedings.mlr.press/v70/scaman17a.html.

[49] Elaine Shi, T.-H. Hubert Chan, Eleanor G. Rieffel, Richard Chow, and Dawn Song. Privacy-Preserving Aggregation of Time-Series Data. In *NDSS*, 2011.

[50] Di Wang, Marco Gaboardi, and Jinhui Xu. Empirical Risk Minimization in Non-interactive Local Differential Privacy Revisited. In *NeurIPS*, 2018.

[51] Jie Xu, Wei Zhang, and Fei Wang. A(dp)^{2}sgd: Asynchronous decentralized parallel stochastic gradient descent with differential privacy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[52] Bicheng Ying, Kun Yuan, Yiming Chen, Hanbin Hu, Pan Pan, and Wotao Yin. Exponential graph is provably efficient for decentralized deep training. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021.

[53] Xueru Zhang, Mohammad Mahdi Khalili, and Mingyan Liu. Improving the Privacy and Accuracy of ADMM-Based Distributed Algorithms. In *ICML*, 2018.

[54] Kai Zheng, Wenlong Mou, and Liwei Wang. Collect at Once, Use Effectively: Making Non-interactive Locally Private Learning Possible. In *ICML*, 2017.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] We define our relaxation of Local DP formally in Section 2, the gossip averaging is analyzed in Section 3 and show substantial amplification, supported by the experiments in Section 5. We define and prove guarantees for decentralized optimization in Section 4.
   (b) Did you describe the limitations of your work? [Yes] We discuss each theorem after in its subsection and show the effective magnitude of the privacy amplification in the experiments (Section 5).
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] We have included a broader impact statement at the end of the supplementary.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] All our theorems and corollaries have their complete set of assumptions.
   (b) Did you include complete proofs of all theoretical results? [Yes] , except when proving similar results, we do not repeat the full proof but only explain the main differences.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] We provide the code needed to reproduce the results in the supplementary material.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] Most of our experiments have no hyperparameters to tune, and we provide the hyperparameters in Annex for Figure 1(c).
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] see Figures 1(c) and 1(a).
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [No] All of the simulations ran in a few minutes on a regular laptop.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] We use Houses Dataset and Facebook Ego dataset and cite them.

(b) Did you mention the license of the assets? [Yes] The dataset Houses is in the public domain, as indicated on the link provided and the Facebook ego dataset in under BSD license.
(c) Did you include any new assets either in the supplemental material or as a URL? [Yes] We have included our code in the supplementary.
(d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]

5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]