DIFFERENTIALLY PRIVATE CONSENSUS AND OPTIMIZATION ON COMMUNICATION CONSTRAINED DIRECTED GRAPHS

MONICA RIBERO, YIYUE CHEN AND HARIS VIKALO

Abstract. The rise of machine learning has been accompanied by growing privacy concerns and demands to protect users' sensitive data. At the same time, the amount of data that needs to be processed has been rapidly growing, bringing forth concerns related to limited communication resources available in practical settings. To this end, in this paper we study decentralized versions of consensus and convex optimization problems over directed graphs with communication and privacy constraints. Leveraging a local differential privacy model, we provide provable privacy guarantees for decentralized algorithmic frameworks that rely on sparsification to reduce the communication cost; while motivated by meeting communication constraints, sparsification is interpreted and exploited as a privacy amplification mechanism. To our knowledge, these are the first consensus and decentralized optimization frameworks that provide differential privacy for decentralized learning on directed graphs under communication constraints. The proposed scheme is tested on the consensus model with synthetic datasets, and a tag prediction model with logistic regression over a realistic Stackoverflow dataset. The experiments validate theoretical results and demonstrate efficacy of the proposed differentially private schemes.

Key words. Distributed optimization, Consensus, Machine Learning,Federated Learning, Communication efficiency, Differential Privacy

AMS subject classifications. 68Q25, 68R10, 68U05

1. Introduction.

Decentralized consensus and convex optimization have been studied in a number of fields including machine learning, signal processing, and control [42, 47, 46]. They have emerged as attractive alternatives to centralized solutions limited by latency challenges [12, 26], high cost of communicating data to the central server [27], and, in many settings, privacy concerns that prohibit central data aggregation [24, 45, 48]. In consensus, a set of n nodes, each one with a data vector $\mathbf{x}_i \in \mathbb{R}^d$, for $i \in [n] := \{1, ..., n\}$, aims to find the mean vector $\bar{\mathbf{x}} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{x}_i$. In decentralized optimization, the nodes collaborate to minimize the finite sum of local objective functions $f_i : \mathbb{R}^d \to \mathbb{R}$ over a convex compact constraint set $\mathcal{X}$, i.e., solve

$$\min_{\mathbf{x}\in\mathcal{X}} \left[f(\mathbf{x}) := \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{x}) \right]. \tag{1.1}$$

Recent multiagent applications over IoT networks, mobile devices and federated learning systems have reignited attention to problems wherein a number of nodes locally collects data and a peer-to-peer interaction allows them to estimate a parameter or optimize a function [8, 21]. However, to deliver potential benefits of decentralized solutions, two issues need to be addressed. First, even if data remains local, shared updates may be high dimensional, e.g., in federated learning where the updates are deep learning models and the number of parameters to be exchanges may be in the millions [18]; in such settings, limited energy and bandwidth typical of practical systems create a communication bottleneck. Second, in recent applications such as recommender systems, federated learning, and online learning, users' privacy may be compromised if sharing information potentially sufficient for identification [33, 1, 30]. Previous solutions to these problems have focused on fixed topologies as in federated learning where a server communicates with all agent nodes in the network – the setting equivalent to a fully connected star network topology [30]. More general decentralized optimization topologies that have previously been studied include undirected graphs

[46, 38, 52, 34, 14, 24, 28]. However, real-world networks are known to be time-varying and support directed communication between the nodes; while there exist communication-efficient protocols for decentralized consensus and optimization in such setting [10], no prior work providing privacy guarantees therein exist.

In this work, we consider differential privacy [15, 16] – a statistical framework that enables trade-off between data utility and privacy loss. The privacy loss of a query to a database is defined as the probability of identifying an individual record in the database from the output of the query. This requires trusting a central aggregator to compute the query output and mechanism, which is not available in the fully distributed and decentralized setting. We consider the local differential privacy model, introduced first by [23], where each node has to protect its outputs by perturbing any shared data or message. We propose (to our knowledge, the first) convergent algorithms for decentralized consensus and optimization over directed graphs that satisfy both communication and differential privacy constraints.

In particular, our contributions can be summarized as follows:

- We propose algorithms for decentralized consensus and convex optimization over time-varying directed graphs with communication, and local and record-level differential privacy constraints.
- We provide convergence analysis for both algorithms; for the consensus algorithm, we establish linear convergence; for the optimization algorithm, we show $O(\ln T/\sqrt{T})$ convergence rate if the global objective function is convex and $O(\ln T/T)$ convergence rate if, in addition, local objective functions are strongly convex.
- We provide a tight privacy analysis and show record-level differential privacy guarantees, with a utility-privacy trade-off of $O\left(\frac{dn}{\epsilon r} + \frac{\sqrt{nd^3}}{\epsilon r}\right)$for convex functions, and $O\left(\frac{p^2nd^2}{\epsilon^2r^2}\right)$ for strongly convex local objectives.
- We perform extensive numerical studies under various communication and privacy settings, and investigate accuracy/communication/privacy trade-offs.

Notation. We represent vectors by lowercase bold letters and matrices by uppercase letters. $[A]_{ij}$ represents the (i,j) element of matrix A. $\|\cdot\|$ represents the standard Euclidean norm. For convenience, the symbols used in the paper are summarized in a table in the supplementary document, Sec A.

1.1. Related work and significance. Prior work on consensus algorithms considered both directed and undirected graphs [5, 44], as well as time-varying graphs [20, 53, 43, 7]. By leveraging compressed communication, [24] developed the first linearly convergent communication-efficient consensus algorithm over undirected time-invariant graphs. [10] established the same type of results for directed time-varying graphs. Prior work on decentralized optimization includes a gradient descent scheme [38], the alternating direction method of multipliers (ADMM) [52], and decentralized dual averaging methods [14, 34]. More recently, [28] studied how decentralization and asynchronous SGD affect convergence. [24, 45] introduced a decentralized convex scheme with limited communication and convergence guarantees; note that all of the above prior work is limited to the undirected settings.

The subgradient-push [35] and Directed Distributed Gradient Descent (D-DGD) [55] address optimization over directed graphs; they achieve $O(\frac{\ln T}{\sqrt{T}})$ convergence rate. In the same setting, [37] improve the convergence rate to linear for smooth and strongly convex functions. However, these approaches do not consider limited communication settings. [10] studies directed networks under communication constraints and proves

$O(\frac{\ln T}{\sqrt{T}})$ convergence rate. In the non-convex setting, [3] proposes Overlap Stochastic Gradient Push which combines the push-sum algorithm with stochastic gradient updates and proves the same sub-linear rate as in SGD.

An approach to reducing communication by taking advantage of local SGD rounds is studied in [49]; this work is extended in [25] to the non-i.i.d. case. [50] proposes MATCHA, an algorithm that improves over previous approaches and reduces communication and computation costs by randomly sampling clients. They also show that local updates in distributed optimization can accelerate convergence rate of the algorithm.

Decentralized optimization under privacy constraints has been studied in [11], which provides an overview of privatization mechanisms for data exchanged in networks. This works shows an imposibility of convergence, under a different privacy model and where all messages in all iterations are noised. In contrast, we rely in the post-processing property of differential privacy to design our algorithm and achieve a better accuracy-privacy tradeoff. Authors in [19, 51] study consensus under different definitions of privacy. The focus of our work is on providing differential privacy guarantees for decentralized consensus and optimization over communication-constrained time-varying networks; related prior work includes [4] which incorporates differential privacy guarantees but considers a setting wherein each user has a set of personalized parameters. In contrast, the parameters in our problem are shared across nodes and the aim is to achieve consensus or minimize the finite sum of local objective functions.

None of the above considers both privacy and communication constraints, often present simultaneously in practice. *cpSGD*, introduced in [2], takes communication and privacy into account by using efficient quantization via random rotations, and introducing a binomial privacy mechanism that reduces the communication overhead. That work is inspired by federated learning and the obtained guarantees are valid only for the fully connected, undirected network topologies. Concurrent to our work, [9] studied the trade-off between communication and privacy, but they only consider the centralized model.

Finally, it is worth pointing out that satisfying differential privacy constraints is typically more challenging in iterative settings, including those commonly used in optimization algorithms; the iterative nature of such algorithms requires splitting the privacy budget across iterations. [1, 54, 29] proposed privacy techniques that account for noisy SGD which adds Gaussian noise to the gradient before updating the parameters. This is further refined in [32] by leveraging Renyi differential privacy [31], a relaxation of the traditional (ϵ, δ)-differential privacy. In our work we analyse composition across iterations and dimensions using strong composition [22], that allows for a more interpretable bound. In our experiments we use Renyi-DP to provide a tighter, more realistic accounting of privacy.

Organization. We start by introducing some preliminary definitions in subsection 2.1, followed by the consensus algorithm and analysis in subsection 2.2. We continue with optimization algorithms and analysis in subsection 2.3, and experimental results in section 3. We include detailed proofs in section 5 and finish with a discussion in section 6.

2. Private and Communication Efficient Decentralized Algorithms.

2.1. Preliminaries.

Communication-constrained networks. We model the connectivity in a network with n nodes by a time-varying directed graph. At time t, the in-neighbor connectivity matrix (row-stochastic), W_{in}^t, and the out-neighbor connectivity matrix (column-stochastic), W^t_{out}, are defined as

$$[W^t_{in}]_{ij} = \begin{cases} > 0, & j \in \mathcal{N}^t_{in,i} \\ 0, & \text{otherwise} \end{cases}, \quad [W^t_{out}]_{ij} = \begin{cases} > 0, & i \in \mathcal{N}^t_{out,j} \\ 0, & \text{otherwise} \end{cases} \tag{2.1}$$

where $\mathcal{N}^t_{in,i}$ is the set of nodes that can send messages to node i (including i) and $\mathcal{N}^t_{out,j}$ is the set of nodes that can receive messages from node j (including j) at time t. Node i knows both $\mathcal{N}^t_{in,i}$ and $\mathcal{N}^t_{out,i}$, which is sufficient for the construction of W^t_{in} and W^t_{out}.

To comply with the communication constraints, the nodes in a network may need to limit their communication to only a fraction of a full message, which we facilitate by applying sparsification methods. To this end, let us introduce a sparsification operator $Q : \mathcal{R}^d \to \mathcal{R}^d$; applying Q to a d-dimensional real vector returns a sparsified version of that vector. If node i can communicate k out of d entries of a message, the probability of any given entry actually being communicated is $\frac{k}{d}$.

Differential privacy. Differential privacy was first introduced in [15] as a mechanism to prevent output queries to databases from disclosing the inclusion of a particular record on the dataset. Formally, it is a bound on the probability of losing a record's privacy by including it in the computation of a query.

Definition 2.1 ((approximate) Differential Privacy). *We say that a randomized algorithm $\mathcal{M}$ satisfies (ϵ, δ)-differential privacy ((ϵ, δ)-DP) if for any pair of datasets $\mathcal{D}$ and $\mathcal{D}'$ differing by only one record and any subset of outcomes $S \in range(\mathcal{M})$ it holds that*

$$Pr(\mathcal{M}(\mathcal{D}) \in S) \leq e^{\epsilon} \cdot Pr(\mathcal{M}(\mathcal{D}') \in S) + \delta.$$

Theorem 2.2 (Theorem A.1 in [16]). *The ℓ_2-sensitivity of a query f evaluated on a dataset D with range in ^{d} is defined as $\Delta_2(f) := \max_{\mathcal{D},\mathcal{D}'} \|f(\mathcal{D}) - f(\mathcal{D}')\|_2$, where $\mathcal{D}$ and $\mathcal{D}'$ are datasets differing in only one record. The Gaussian mechanism with parameter σ, $G_{f,\sigma}$, adds zero-mean Gaussian noise with variance σ^2 to all d coordinates in query f; formally, $\mathcal{G}_{f,\sigma}(\mathcal{D}) = f(\mathcal{D}) + N(0, \sigma^2 I_d)$, allowing an abuse of notation. $G_{f,\sigma}(\mathcal{D})$ is (ϵ, δ)-DP if $\sigma \geq \frac{\Delta_2(f)}{\epsilon}\sqrt{2\log(\frac{1.25}{\delta})}$.*

The Sampled Gaussian Mechanism with sampling rate p and noise variance σ^2, $SGM_{p,\sigma}$, first generates a subset of $\mathcal{D}$ by selecting points independently at random with probability p, computes the query on this random subset, and adds to it samples from a zero-mean Gaussian distribution with variance σ^2. Further details and illustrations of this mechanism are in the supplementary material, Sec B. Differential privacy assumes there is a trusted central aggregator that possesses all users' data and computes private output queries, allowing the aggregation to add less noise. In the decentralized setting each user has it's data locally, thus all users' outcomes have to be noised, making the problem harder. In our case, all nodes share record-level differentially private messages. This means that we protect users against an attack wishing to learn if a specific record is in their data. To illustrate the different models, consider the setting where each node i has r records and X_i is a query to node i, and $\Delta_2(X_i) = \frac{1}{r}$. Assume we want to disclose the mean of $X_1, X_2, ..., X_n$. In the central model, the sensitivity of query $\bar{x}$ is $\frac{1}{nr}$ so we only add gaussian noise with variance $O\left(\frac{L}{\epsilon nr}\right)$. In the local model, we would have to add for each node i, thus noise standard deviation is augmented by a factor of n, $n = O\left(\frac{L}{\epsilon r}\right)$.

2.2. Differentially-Private Communication-Efficient Consensus. In general, applying sparsification methods to existing consensus schemes, e.g. [6, 7, 35], does not guarantee convergence since the sparsification operator causes non-vanishing error. In [10], authors rely on entry-wise sparsification of a message vector and the structure of the underlying connectivity matrices to propose a convergent consensus algorithm. This is accomplished by splitting the vector-valued consensus problem into d scalar-valued sub-problems with connectivity matrices $\{W_{in,m}\}_{m=1}^{d}$ and $\{W_{out,m}\}_{m=1}^{d}$, re-normalized according to the sparsification patterns. We assume node i has access to vector $\mathbf{x}_i$ and, following [10], introduce an auxiliary "surplus" vector $\mathbf{y}_i \in \mathcal{R}^d$; at iteration t, $\mathbf{y}_i^t$ records the change between consecutive state vectors, $\mathbf{x}_i^t - \mathbf{x}_i^{t-1}$. Both $\mathbf{x}_i^t$ and $\mathbf{y}_i^t$ can be communicated to the out-neighbors of node i. To simplify the notations, we introduce $\mathbf{z}_i^t \in \mathcal{R}^d$ defined as

$$\mathbf{z}_i^t = \begin{cases} \mathbf{x}_i^t, & i \in \{1, ..., n\} \\ \mathbf{y}_{i-n}^t, & i \in \{n+1, ..., 2n\}. \end{cases} \tag{2.2}$$

Let $Q(\mathbf{z}_i^t)$ denote a vector obtained by sparsifying $\mathbf{z}_i^t$, and $[Q(\mathbf{z}_i^t)]_m$ be the m-th entry of $Q(\mathbf{z}_i^t)$. The re-normalization of the in-neighbor and out-neighbor connectivity matrices is performed according to

$$[A_m^t]_{ij} = \begin{cases} \frac{[W_{in}^t]_{ij}}{\sum_{j \in \mathcal{S}_m^t(i,j)} [W_{in}^t]_{ij}} & \text{if } j \in \mathcal{S}_m^t(i,j) \\ 0 & \text{otherwise}, \end{cases}, [B_m^t]_{ij} = \begin{cases} \frac{[W_{out}^t]_{ij}}{\sum_{i \in \mathcal{T}_m^t(i,j)} [W_{out}^t]_{ij}} & \text{if } i \in \mathcal{T}_m^t(i,j) \\ 0 & \text{otherwise}, \end{cases} \tag{2.3}$$

respectively, where $\mathcal{S}_m^t(i,j) = \{j | j \in \mathcal{N}_{in,i}^t, [Q(z_j^t)]_m \neq 0\} \cup \{i\}$ and $\mathcal{T}_m^t(i,j) = \{i | i \in \mathcal{N}_{out,j}^t, [Q(z_i^t)]_m \neq 0\} \cup \{j\}$. The connectivity and communication weights across the network are summarized by mixing matrices; in particular, the m-th mixing matrix at iteration t is defined as

$$\bar{M}_m^t = \begin{bmatrix} A_m^t & \mathbf{0} \\ I - A_m^t & B_m^t \end{bmatrix}, \tag{2.4}$$

Having defined $\mathbf{z}_i^t$ and $\bar{M}_m^t$, respectively, we now introduce the update for $\mathbf{z}_i^t$ in the communication efficient and (ϵ, δ)-DP consensus algorithm (Algorithm 2.1):

$$z_{im}^{t+1} = \sum_{j=1}^{2n} [\bar{M}_m^t]_{ij} [Q(\mathbf{z}_j^t)]_m + \mathbb{1}_{\{t \mod \mathcal{B} = \mathcal{B}-1\}} \gamma [F]_{ij} z_{jm}^{\mathcal{B}\lfloor t/\mathcal{B} \rfloor}, \tag{2.5}$$

where $F = \begin{bmatrix} \mathbf{0} & I \\ \mathbf{0} & -I \end{bmatrix}$ and m represents the coordinate index. Vectors $\mathbf{z}_i^t$ are updated according to the sparsification and multiplication of the mixing matrices at all time steps except those that are multiples of $\mathcal{B}$, i.e.,

$$t \mod \mathcal{B} = \mathcal{B} - 1. \tag{2.6}$$

where $\mathcal{B}$ is the window size parameter indicating that starting from any time $t = k\mathcal{B}$ for all integers $k \geq 0$, the union graph over $\mathcal{B}$ consecutive time steps forms a strongly connected graph (See Assumption 2.4).

In the update (2.5), when the time step t satisfies (2.6), vectors $\mathbf{z}_i^{\mathcal{B}\lfloor t/\mathcal{B} \rfloor}$, stored at time $\mathcal{B}\lfloor t/\mathcal{B} \rfloor$, are also included in the update. The term $\sum_{j=1}^{2n} [F]_{ij} z_{jm}^{\mathcal{B}\lfloor t/\mathcal{B} \rfloor}$ in the update facilitates to form the following update over $\mathcal{B}$ time steps:

$$\mathbf{z}_{im}^{(k+1)\mathcal{B}} = \sum_{j=1}^{2n} [\bar{M}_m((k+1)\mathcal{B} - 1 : k\mathcal{B}) + \gamma F]_{ij} Q(\mathbf{z}_j^{k\mathcal{B}})_m \tag{2.7}$$

, where $\bar{M}_m((k+1)\mathcal{B}-1:k\mathcal{B})$ represents the product of mixing matrices from time $k\mathcal{B}$ to time $(k+1)\mathcal{B}-1$. Since neither any single mixing matrix nor this product can ensure a non-zero spectral gap, γF is added to ensure a non-zero spectral gap. In addition, F has its $(1,1)$ and $(2,1)$ block equal to zero matrix and the rest two diagonal blocks and therefore the stored vectors are not communicated, leading to the update (2.5) above.

Our proposed (ϵ,δ) differentially private procedure is summarized as Algorithm 2.1; it builds upon [10] to incorporate the Gaussian mechanism and aplifies it with sparsification. As we prove in Theorem 2.6, adding privacy guarantees has no detrimental effect on the convergence of decentralized consensus with sparsified updates, while Theorem 2.3 establishes that all messages guarantee (ϵ,δ) differential privacy.

2.2.1. Privacy guarantees. In the consensus case, we assume the "honest-but-curious" security model [40] where all nodes execute the algorithm honestly but may attempt to learn additional information from the received messages. In the first iteration, each node receives a perturbed version of individual data vectors from its neighbors.

Our adopted privacy mechanism adds a zero-mean Gaussian noise to query outputs, i.e., user vectors, where the noise standard deviation is proportional to the L_2-sensitivity of the query.

In the first iteration of our consensus algorithm, the sensitivity of the query at node i is $\Delta_2(\mathbf{z}_i^1)=\sqrt{d}C$, where C denotes a bound on the magnitude of the components of the original message $\mathbf{x_i^0}$.

Note that in the consensus problem it is only necessary to add noise to the initial vector, before the first iteration of the algorithm. This is the key insight to our differentially private algorithm: posterior messages do not touch the data again.

This is because subsequent messages, being functions of the initial noisy message, are already privatized by the post-processing property of differential privacy [16]. This allows us to achieve a better accuracy-privacy trade-off than [39], [11].

Algorithm 2.1 Communication Efficient and (ϵ,δ)-DP Consensus Algorithm

Input: Time horizon T, Initial state $\mathbf{x}^0$, Initialize $\mathbf{y}^0=\mathbf{0}$, noise variance σ^2, network connectivity parameter $\mathcal{B}$ and γ
Noise initial data $\mathbf{x}_i^0 \leftarrow \mathbf{x}_i^0+\mathbf{b}_i$ for $\mathbf{b}_i \sim N(0,\sigma^2 I_{d\times d})$, $i=1,...,n$
Initialize $\mathbf{z}^0$
for $t=1,...,T$ **do**
  Generate non-negative matrices W_{in}^t, W_{out}^t
  for $m=1,...,d$ **do**
    construct a row-stochastic A_m^t and a column-stochastic B_m^t according to (2.3)
    construct $\bar{M}_m^t$ according to definition (2.4)
    for $i=1,...,2n$ **do**
      $z_{im}^{t+1}=\sum_{j=1}^{2n}[\bar{M}_m^t]_{ij}[Q(\mathbf{z}_j^t)]_m+\mathbb{1}_{\{t \mod \mathcal{B}=\mathcal{B}-1\}}\gamma[F]_{ij}z_{jm}^{\mathcal{B}\lfloor t/\mathcal{B}\rfloor}$
    end for
  end for
end for
Result: Local consensus values $\mathbf{z}_i^T$ for nodes $i=1,...,n$

Before starting the loop, we define $z_j^0=[\mathbf{x}_j^0+\mathbf{b}_j,y_j^0]$, where $\mathbf{b}_j\sim N(0\sigma^2)$ in the first iteration of the algorithm, node i receives

$$z_{im}^1 = \sum_{j=1}^{2n} [M_m^t]_{ij}[Q(z_j^t)]_m + N(0, \sigma^2). \tag{2.8}$$

To protect user j we only need to ensure sensitivity of z_{im}^1, the first aggregated message received by i from other nodes.

THEOREM 2.3. *Set* $\sigma = \frac{\sqrt{d}C}{\epsilon}\sqrt{2\log(\frac{1.25}{\delta})}$, *Algorithm 2.1 is* $(\epsilon, \delta)-$ *differentially private.*

Proof. Note that the sensitivity of $\mathbf{x}_i^0$ is $\Delta_2(\mathbf{x}_i^0) = \sqrt{d}C$, thus (2.8) states the Gaussian mechanism for the first step and hence this iteration is (ϵ, δ)-DP. Since further iterations depend only on the original DP queries (without further access to raw data), the overall (ϵ, δ)-DP of the algorithm follows from the post-processing property of differential privacy [16]. ∎

2.2.2. Convergence guarantees. Having provided privacy guarantees for Algorithm 2.1, we next study its convergence properties. In particular, we prove that the addition of privacy mechanism does not adversely affect convergence rate. We begin by making assumptions needed for the analysis.

Assumption 2.4. The graph induced by sparsification is $\mathcal{B}$-jointly connected, i.e., starting from any time step $t = k\mathcal{B}$ where $k = 0, 1, \cdots$, the union graph over $\mathcal{B}$ consecutive time steps, $\bigcup_{t=k\mathcal{B}}^{(k+1)\mathcal{B}-1} \mathcal{G}(t)$ is a strongly connected directed graph.

This is a common assumption for algorithms on directed networks [37].

Assumption 2.5. Given $\gamma \in (0, 1)$, the set of all possible mixing matrices $\{M_m^t\}$, $\mathcal{U}_M$, is finite.

We are now ready state the convergence result. We differ all proofs to section section 5

THEOREM 2.6. *Suppose Assumption 2.4 and 2.5 hold. Fix*

$$\gamma \in (0, \min_m \left\{ \frac{1}{(20+8n)^n}(1 - |\lambda_3(\bar{M}_m((k+1)\mathcal{B} - 1 : k\mathcal{B}))|)^n \right\}),$$

and let $\tau = \max_{C\in\mathcal{U}_M} |\lambda_2(C)| < 1$, $\bar{\mathbf{x}} = \frac{1}{n}\sum_{i=1}^n \mathbf{x}_i^0$, *and* $t \geq 0$. *Then running Algorithm 2.1 for* t *iterations, suppose* $t = k\mathcal{B} - 1 + t'$, *where* $t' = 0, \cdots, \mathcal{B} - 1$ *and it holds that for any* $i \in [n]$ *and* $t \geq 1$,

$$\begin{aligned} \|\mathbf{x}_i^t - \bar{\mathbf{z}}^t\| &\leq \sqrt{2nd}(\tau^{1/\mathcal{B}})^{t-(t'-1)} \sum_{j=1}^{2n}\sum_{m=1}^{d} |z_{jm}^0|, \\ \|\mathbf{y}_i^t\| &\leq \sqrt{2nd}(\tau^{1/\mathcal{B}})^{t-(t'-1)} \sum_{j=1}^{2n}\sum_{m=1}^{d} |z_{jm}^0|, \end{aligned} \tag{2.9}$$

where $\bar{\mathbf{z}}^t = \frac{1}{n}\sum_{i=1}^n \mathbf{x}_i^t + \frac{1}{n}\sum_{i=1}^n \mathbf{y}_i^t$. *Further,* $E[\mathbf{x}_i^t]$ *converges to* $\bar{\mathbf{x}}$ *at a linear rate* $O(\tau^{t/\mathcal{B}})$.

Theorem 2.6 implies that the local state vector $\mathbf{x}_i^t$ converges (at a linear rate) to the averaging consensus vector $\bar{\mathbf{z}}^t$, while the surplus vectors $\mathbf{y}_i^t$ vanishes to zero.

Since the noise added in the first iteration is unbiased, it always holds that $E[\bar{\mathbf{z}}^t] = \bar{\mathbf{z}}^0$. Therefore, by setting $\mathbf{y}_i^0 = \mathbf{0}$ in the initialization of Algorithm 2.1, $\bar{\mathbf{z}}^0 = \bar{\mathbf{x}}$ and we are able to guarantee linear convergence to $\bar{\mathbf{x}}$ in expectation. Note that the fastest consensus algorithms for directed time-varying graphs, be they for full communication as in the [13, 6] or with sparsification of messages as in [10], enjoy linear convergence. Unlike our Algorithm 2.1, however, those schemes do not provide privacy guarantees.

2.3. Differentially-Private Communication-Efficient Optimization. Unlike consensus, decentralized optimization typically requires algorithms to access local raw data whenever gradients are computed; consequently, all iterations with gradient computation need to be protected. To this end, as in DP-SGD we deploy a privacy mechanism by perturbing the gradient

$$\mathbf{g}_i^t = \begin{cases} \nabla f_i(\mathbf{x}_i^t), & i \in \{1, ..., n\} \\ \mathbf{0}, & i \in \{n+1, ..., 2n\} \end{cases} \tag{2.10}$$

with a noise term sampled from a Gaussian $N(0, \sigma^2 D^2)$ distribution where D is an upper bound on the magnitude of the components of $\mathbf{g}_i^t$. Note that (2.10) implies the state vectors are updated via decentralized gradient descent while the surplus vectors are updated the same way as in Algorithm 2.1.

Our proposed differentially private decentralized optimization scheme is formalized as Algorithm 2.2. Differential privacy of the iterates is enforced in line 8 of the algorithm. Despite incorporating privacy mechanism by adding noise to gradients, the algorithm converges when a decreasing learning rate is used. Formal privacy and convergence guarantees are provided below. Further, we study the trade-off between these two.

2.3.1. Privacy guarantees. While the consensus algorithm only access the original data on the first iteration, gradient descent access it at every iteration to compute the current gradient. We have to take again a local DP approach but now ensure that each message is privatized, and use the composition theorem to ensure the track the overall algorithm DP guarantees.

Traditionally, the full gradient is considered as one query and privatized with a gaussian vector b_t drawn from $b_t \sim \mathcal{N}(0, \sigma^2 I_d)$. Notice the expected norm of the noise is $\mathbb{E}[\|b_t\|_2] = \sqrt{d}\sigma^2$. Thanks to the sparsification, we only have to take into account $(1-q)d$ dimensions, and get a privacy amplification factor of $(1-q)$ and the privacy guarantee.

THEOREM 2.7. *Assuming $\sigma = O(\frac{D\sqrt{T(1-q)d}\log(1/\delta}{\epsilon})$, after T iterations, Algorithm 2.2 satisfies (ϵ, δ) differential privacy.*

Proof. Assuming $|g_{im}^t| \leq D$, which is readily achieved by the Lipschitz assumption, the sensitivity of z_{im}^{t+1} is given by $\Delta_2(\alpha_t g_{im}) = \alpha_t D$. We use advanced composition (See Corollary 1 in [31]) over dimensions $(1-q)d$, and over T iterations, and obtain the desired result. ∎

2.3.2. Convergence guarantees. We now shift our attention from the privacy to convergence guarantees. To facilitate the analysis, we first study a broader problem: we focus on convergence under arbitrary gaussian noise variance σ. Then, we study the utility - privacy trade-off by imposing necessary assumptions for privacy on the noise variance, that guarantee we meet privacy requirements.

We can now state the convergence result.

Algorithm 2.2 Communication-Efficient and (ϵ,δ)-DP Decentralized Optimization Algorithm

Input: Time horizon T Initial state $\mathbf{x}^0$, Initialize $\mathbf{y}^0 = \mathbf{0}$, sparsification level q, noise variance σ^2, network connectivity parameter $\mathcal{B}$ and γ
set $\mathbf{z}^0 = [\mathbf{x}^0; \mathbf{y}^0]$
for $t = 1, ..., T$ **do**
  Generate non-negative matrices W_{in}^t, W_{out}^t
  for $m = 1, ..., d$ **do**
    construct a row-stochastic A_m^t and a column-stochastic B_m^t according to 2.3
    construct $\bar{M}_m^t$ according to definition
    for $i = 1, ..., 2n$ **do**

$$\begin{aligned} z_{im}^{t+1} = &\sum_{j=1}^{2n} [\bar{M}_m^t]_{ij}[Q(\mathbf{z}_j^t)]_m + \mathbb{1}_{\{t \mod \mathcal{B}=\mathcal{B}-1\}}\gamma[F]_{ij} z_{jm}^{\mathcal{B}\lfloor t/\mathcal{B}\rfloor} \\ &- \mathbb{1}_{\{t \mod \mathcal{B}=\mathcal{B}-1\}}\alpha_{\lfloor t/\mathcal{B}\rfloor}(g_{im}^{\mathcal{B}\lfloor t/\mathcal{B}\rfloor} + N(0,\sigma^2)), \end{aligned}$$

    end for
  end for
end for
Result: Local optimum values $\mathbf{z}_i^T$ for nodes $i = 1, ..., n$

THEOREM 2.8. *Suppose Assumptions 2.4-2.5 on mixing matrices hold, and fix $\gamma \in (0, \min_m \left\{\frac{1}{(20+8n)^n}(1 - |\lambda_3(\bar{M}_m((k+1)\mathcal{B}-1 : k\mathcal{B}))|)^n\right\})$. Assume $|g_{im}^t| \le D$. If the objective function f is convex, and the step size α_t decays as $O(1/\sqrt{T})$, Algorithm 2.2 converges at a rate of $O(\ln T/\sqrt{T})$, Concretely,*

$$(E[f_{\min,T}] - f^*) \le \frac{C_1}{\sum_{k=0}^{\lfloor T/\mathcal{B}\rfloor}\alpha_k} + \frac{C_2\sum_{k=0}^{\lfloor T/\mathcal{B}\rfloor}\alpha_k^2}{\sum_{k=0}^{\lfloor T/\mathcal{B}\rfloor}\alpha_k} \le \frac{C_1}{\sqrt{T/\mathcal{B}-1}} + \frac{C_2\ln(T/\mathcal{B})}{\sqrt{T/\mathcal{B}-1}} \tag{2.11}$$

where $f_{\min,T} := \min_{t=1,\cdots,T} f(\bar{\mathbf{z}}^t)$ and f^* is the optimal value, and

$$C_1 = \frac{n\sqrt{d}D^2}{2} + \frac{\sqrt{2}dD}{\sqrt{n}}\sum_{j=1}^{2n}\frac{\|\mathbf{z}_j^0\|}{1-\tau^2}, \tag{2.12}$$

$$C_2 = \frac{(d+\sigma^2)nD^2}{2} + \frac{\sqrt{2}dD}{\sqrt{n}}\sum_{j=1}^{2n}\|\mathbf{z}_j^0\| + \frac{2\sqrt{2nd^2(d+\sigma^2)}D^2}{1-\tau} + \frac{4\sqrt{d(d+\sigma^2)}D^2}{n} \tag{2.13}$$

THEOREM 2.9. *Suppose Assumptions 2.4-2.5 on mixing matrices hold, fix $\gamma \in (0, \min_m \left\{\frac{1}{(20+8n)^n}(1 - |\lambda_3(\bar{M}_m((k+1)\mathcal{B}-1 : k\mathcal{B}))|)^n\right\})$. Assume $|g_{im}^t| \le D$ and let $\hat{\mathbf{x}}_i{}^T = \frac{\sum_{t=1}^{\lfloor T/\mathcal{B}\rfloor}(t-1)\mathbf{x}_i^t}{t(t-1)/2}$ for $T \ge \mathcal{B}$ and $K = \lfloor T/\mathcal{B}\rfloor$. If the objective function f is convex and the local objective function f_i is μ_i strongly-convex, then there exists some constant*

$C_3 > 0, C_4 > 0$ such that for all i

(2.14)

$$E[f(\hat{\mathbf{x}}_i^T) - f(\mathbf{x}^*)] \leq \frac{C_3}{K}\sum_{j=1}^n \|\mathbf{x}_j^0\|_1 + \frac{C_4}{K}(1+\ln(K-1)) + \frac{p^2nD^2d}{K}(1+\sigma^2)$$

$$E[\sum_{j=1}^n \mu_j\|\hat{\mathbf{x}}_j^T - \mathbf{x}^*\|^2] \leq \frac{C_3}{K}\sum_{j=1}^n \|\mathbf{x}_j^0\|_1 + \frac{C_4}{K}(1+\ln(K-1)) + \frac{p^2nD^2d}{K}(1+\sigma^2)$$

where the step size $\alpha_t = \frac{p}{t}$ for $p \geq \frac{4n}{\sum_{i=1}^n \mu_i}$, $\mathbf{x}^*$ is the optimal solution, $C_3 = \frac{5\sqrt{2n}ndD}{1-\tau}$ and $C_4 = \frac{5\sqrt{2nd(1+\sigma^2)}n^2dD^2}{1-\tau}$.

THEOREM 2.10. *Suppose Assumptions 2.4-2.5 on mixing matrices hold for a specific γ. Assume $|g_{im}^t| \leq D = O(1)$. If the objective function f is convex, and the step size α_t decays as $O(1/\sqrt{T})$, Algorithm 2.2 converges at a rate of $O(\ln T/\sqrt{T})$. Concretely,*

$$(E[f_{\min,T}] - f^*) \leq O\left(\frac{C_1}{\sqrt{T}} + \frac{C_2\ln T}{\sqrt{T}}\right) \tag{2.15}$$

where $f_{\min,T} := \min_{t=1,\cdots,T} f(\bar{\mathbf{z}}^t)$ and f^* is the optimal value, and

$$C_1 = O(n\sqrt{d} + \sqrt{n}d), \quad C_2 = O\left((d+\sigma^2)n + \sqrt{n}dD + \sqrt{nd^2(d+\sigma^2)}\right) \tag{2.16}$$

THEOREM 2.11. *Suppose Assumptions 2.4-2.5 on mixing matrices hold, fix $\gamma \in (0, \min_m \left\{\frac{1}{(20+8n)^n}(1-|\lambda_3(\bar{M}_m((k+1)\mathcal{B}-1:k\mathcal{B}))|)^n\right\})$. Assume $|g_{im}^t| \leq D$ and let $\hat{\mathbf{x}_i}^T = \frac{\sum_{t=1}^{\lfloor T/\mathcal{B}\rfloor}(t-1)\mathbf{x}_i^t}{t(t-1)/2}$ for $T \geq \mathcal{B}$ and $K = \lfloor T/\mathcal{B}\rfloor$. If the objective function f is convex and the local objective function f_i is μ_i strongly-convex, then there exists some constant $C_3 > 0, C_4 > 0$ such that for all i*

(2.17)

$$E[f(\hat{\mathbf{x}}_i^T) - f(\mathbf{x}^*)] \leq \frac{C_3}{K}\sum_{j=1}^n \|\mathbf{x}_j^0\|_1 + \frac{C_4}{K}(1+\ln(K-1)) + \frac{p^2nD^2d}{K}(1+\sigma^2)$$

$$E[\sum_{j=1}^n \mu_j\|\hat{\mathbf{x}}_j^T - \mathbf{x}^*\|^2] \leq \frac{C_3}{K}\sum_{j=1}^n \|\mathbf{x}_j^0\|_1 + \frac{C_4}{K}(1+\ln(K-1)) + \frac{p^2nD^2d}{K}(1+\sigma^2)$$

where the step size $\alpha_t = \frac{p}{t}$ for $p \geq \frac{4n}{\sum_{i=1}^n \mu_i}$, $\mathbf{x}^*$ is the optimal solution, $C_3 = \frac{5\sqrt{2n}ndD}{1-\tau}$ and $C_4 = \frac{5\sqrt{2nd(1+\sigma^2)}n^2dD^2}{1-\tau}$.

Remark 2.12. It is of interest to explore the impact of q (defined in) on the convergence speed. As the mixing matrices are constructed over sparsified graphs, q affects the number of non-zero entries in the matrix and further affects the second largest magnitude of eigenvalues. Specifically, when the graph connectivity parameter $\mathcal{B}$ is fixed, greater q leads to greater τ and further slows down the convergence process.

Theorem 2.8 and 2.9 provide convergence results for Algorithm 2.2 with different assumptions: convexity of the global function (Theorem 2.8) and, in addition, strong-convexity of local functions (Theorem 2.9). The resulting convergence rates match those of the full communication gradient-push and D-DGD algorithms [35, 55], the communication-efficient algorithm in [10], and the stochastic gradient-push [36], under respective assumptions.

In the following section we study relevant instances where these assumptions hold, including linear regression and logistic regression.

2.3.3. Utility - privacy tradeoff. In this section we provide explicit trade-offs between privacy and utility of optimization algorithms. We state our results in **??**

COROLLARY 2.13. *Assume the setting of Theorem 2.8 holds,particularly, f is a D-Lipschitz function, and assume $D = \|\mathbf{z}_j^0\| = O(1)$. Let r be the minimum number of records each node has. Setting $\sigma = O\left(\frac{\sqrt{T(1-q)d}\log(1/\delta)}{\epsilon r}\right)$, after $T = \epsilon^2 r^2$ iterations algorithm Algorithm 2.2 is$(\epsilon,\delta)-$ differentially private and the empirical risk is bounded by*

$$E[f_T - f^*] \leq O\left(\frac{dn}{\epsilon r} + \frac{\sqrt{nd^3}}{\epsilon r}\right)$$

COROLLARY 2.14. *Assume the same setting of Theorem 2.9, and r be the minimum number of records each user has. Let $\sigma = O\left(\frac{\sqrt{T(1-q)d}\log(1/\delta)}{\epsilon r}\right)$, then Algorithm 2.2 as $T \to \infty$,*

$$E[f_T - f^*] \leq O\left(\frac{p^2 n d^2(1-q)}{\epsilon^2 r^2}\right)$$

The proof of both corollaries follows by replacing σ in Theorem 2.8 and **??** with the appropriate value of σ.

3. Numerical Results.

In this section, we demonstrate performance of the proposed privacy-preserving algorithms for decentralized consensus and optimization. In both settings we show that, as expected, privacy and communication constraints slow down convergence but the developed methods ultimately achieve performance similar to that of non-private and full-communication algorithms. We start our numerical studies with a network system having 10 nodes, and generate its edges randomly while preserving the strong connectivity.

The construction begins with the Erdős–Rényi model [17] with edge probability parameter equal to 0.9; then, 2 directed edges are dropped from each strongly connected graph, leading to directed graphs. Building upon this basic structure, we can construct networks with different connectivity. Recall that the window size parameter $\mathcal{B}$, introduced in Assumption 2.4, implies that the union graph over $\mathcal{B}$ consecutive time steps, starting from any time that is a multiple of $\mathcal{B}$, forms an almost-surely strongly connected Erdős–Rényi graph. When $\mathcal{B} = 1$, the network is strongly connected at each time step. We then apply sparsification such that the communication throughput is brought down to various sparsity levels q (larger q means more entries are sparsified, $q = 0$ means full communication). For privacy accounting in optimization we use the TensorFlow Privacy library.[1]

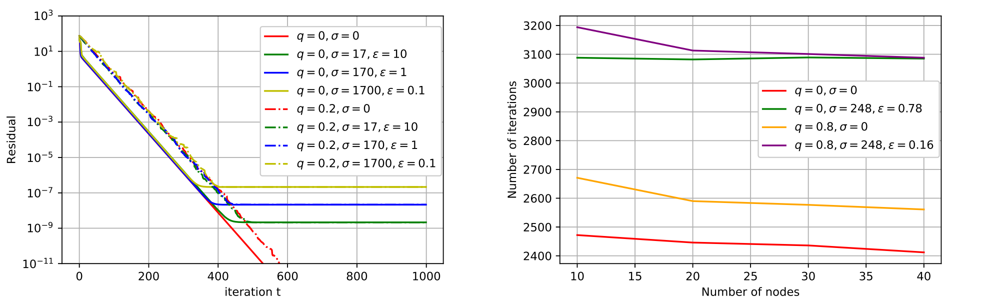
3.1. Consensus.

In the consensus problem, each node has access to a local vector of dimension $d = 64$. Components of the initial local vector at node i, $\mathbf{x}_i^0$, are generated uniformly at random from $[-5, 5]$. To illustrate the effect of the privacy mechanism, in Figure 1 we compare the performance of our Algorithm 1 for different levels of noise σ and sparsity q, and show the corresponding privacy guarantee ϵ. In Fig. 1a, we show the residual as a function of the number of iterations t. We observe that sparsity and noise added to provide privacy only delay the convergence without

[1] https://github.com/tensorflow/privacy

affecting its rate, matching the results of Theorem 2.6. As expected, higher values of q result in slower convergence since less information is communicated; however, higher q achieves higher privacy for fixed σ because the probability of observing an entry is lower. Fig. 1b shows that for a fixed sparsification level, the convergence becomes faster as the number of nodes increases. Further results for varied values of parameters and network topologies are in the supplementary material, Sec C.

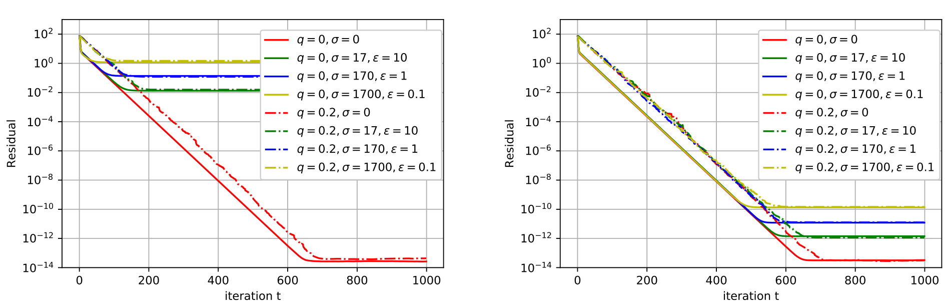
We observe the noise effectively does not affect the final residual, neither the rate of convergence: in Figure 1 it is clear that for all values of q and σ the speed of convergence is the same, although the initial residual might differ depending on the communication and privacy parameters. Eventually all methods converge.



(a) Residual vs. iterations for a 10-node network, $\mathcal{B} = 5$ and $\delta = 10^{-4}$.

(b) The number of iterations needed for residual to drop below 10^{-10} as a function of the number of nodes.

Fig. 1: Convergence of Algorithm 1 for varied parameters q and σ, and the privacy loss ϵ achieved. In (a), we see sparsity and noise delay convergence in early iterations but the convergence rate is unaffected. In (b) we show that for a fixed sparsification level, the convergence becomes faster as the number of nodes increases.



(a)

(b)

Fig. 2: Residual vs. iterations for a 10-node network, $\mathcal{B} = 5$ and $\delta = 10^{-4}$.

3.2. Decentralized Optimization Problems. Next, we test performance of Algorithm 2 on a multi-class tag classification task with a logistic regression model, which leads to the optimization problem with features $\mathbf{m}_{ij}$ and corresponding label y_{ij} of the form

$$\min_{\mathbf{x}} \left\{ \frac{\mu}{2}\|\mathbf{x}\|^2 + \sum_{i=1}^{n}\sum_{j=1}^{N} \ln(1+\exp(-(\mathbf{m}_{ij}^T\mathbf{x}_i)y_{ij})) \right\}. \tag{3.1}$$

with regularization prameter μ.

The model is trained and tested on the Stackoverflow dataset, a language modelling dataset with questions and answers collected from 342477 unique users. The objective is to tag each sentence with appropriate categories. We present detailed preprocessing of the dataset in the supplementary, Sec C. Following prior work [41], we use a build vocabulary with 10000 frequent words and restrict each user's dataset to have at most 128 sentences. We rely on padding and truncation to enforce 20 word sentences, and represent them with index sequences corresponding to the vocabulary words, out of vocabulary words, beginning and end of sentences.

The $150,000$ data points are randomly split into 10 groups of equal size, where each group is interpreted as being the local data for one of the nodes in the network. Each node uses $13,500$ data points as training data and the remaining 1500 points as the validation set. The testing data contains 37640 data points.

We consider a network with 10 nodes and evenly split $150,000$ data points at random into 10 groups, each one representing one node in the network. We leave 1500 points for validation for each node. For this problem we use a noise variance of $\sigma D = 30$, a the step size $\alpha_t = \frac{0.02}{t}$ and the privacy parameter $\delta = 10^{-4}$.

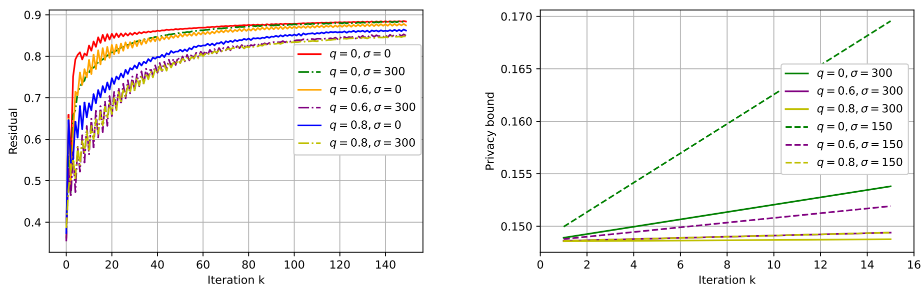
In **??** we observe the effect of privacy and sparsity. Both constraints slightly delay and affect convergence. However, Figure 3a shows that this has minimal impact on the accuracy, and that after a few rounds all models reach a similar level of accuracy. Finally, Figure 3b shows that we are able to maintain a fair privacy budget ($\epsilon < 10$) for models, even at the end of the training; this is a reasonable budget for iterative procedures in literature [1], showing that our algorithms are able to achieve very good performance while guaranteeing privacy and meeting communication constraints.

4. Additional experiments.

4.1. Consensus. In Figure 1 of the main paper, we show convergence results for the proposed consensus algorithm (algorithm 1). With the same parameter setups, Figure 4 in this document illustrates the relationship between sparsity level and privacy bound. As expected, smaller sparsity level q and smaller σ lead to larger privacy loss.

4.2. Linear Regression. We test the linear regression problem where the goal is to minimize the objective $f(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n}\|\mathbf{y}_i - D_i\mathbf{x}\|^2$, where $D_i \in \mathbb{R}^{200\times 5}$ and $\mathbf{y}_i \in \mathbb{R}^{200}$ denote the local measurement matrix and local measurement vector at node i, respectively. To generate the data we synthesize the optimal solution $\mathbf{x}^*$ from the normal distribution. Then, $\mathbf{y}_i$ is formed as $\mathbf{y}_i = D_i\mathbf{x}^* + \xi_i$, where ξ_i denotes the noise added to the local measurement at node i. For all i, D_i is drawn at random from the standard normal distribution and then normalized so that its rows sum to 1; ξ_i is generated from a Gaussian distribution with zero mean and variance 0.01. Local vectors $\mathbf{x}_i^0$ are randomly initialized; the stepsize decreases with the iterations and is set to $\alpha_k = \frac{0.2}{k}$ in the k-th iteration.

In the implementation of the proposed algorithm, the gradient bound is set to $D = 10$ and the privacy parameter δ is set to $\delta = 10^{-5}$. We compute the residual for



(a) Accuracy (b) Privacy bound

Fig. 3: Results on logistic regression on Stackoverflow. In (a) we observe sparsity and privacy delay convergence but they do not affect performance. In (b) we show the privacy loss over several iterations; we are able to maintain a reasonable budget for all combinations of parameters.

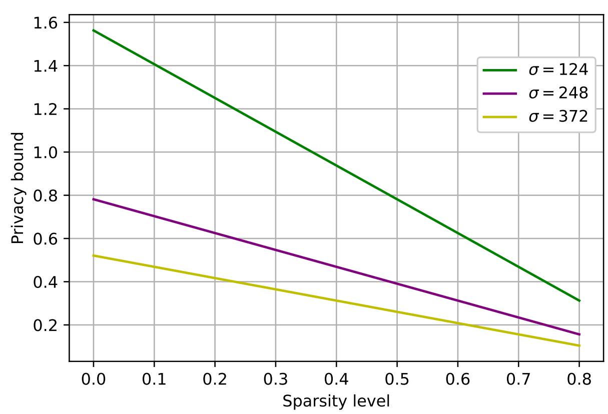


Fig. 4: Privacy bound for varied sparsity levels.

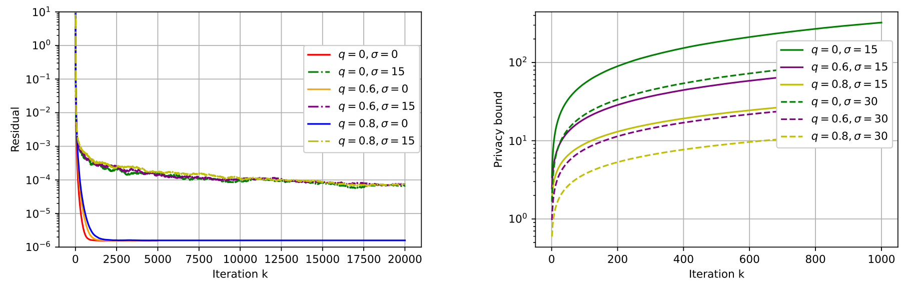
each iteration and show the results in Figure 5a. We observe that both sparsity and privacy slow down the convergence.

Privacy bound for schemes with varied values of parameters are shown in Figure 5b, illustrating how privacy degrades over iterations.

4.3. Logistic Regression.

4.3.1. Datasets.

Stackoverflow. Following prior work [41], we use a build vocabulary with 10000 frequent words and restrict each user's dataset to have at most 128 sentences. We rely on padding and truncation to enforce 20 word sentences, and represent them with index sequences corresponding to the vocabulary words, out of vocabulary words, beginning and end of sentences.



(a) Residual

(b) Residual and corresponding privacy bound for varied parameters; $\delta = 10^{-5}$.

Fig. 5: Results of algorithm 2 on linear regression for synthetic data. $B = 5$. In (a) we show the sparsification and privacy will delay the convergence. In (b) we show how the privacy bound increases as we increase sparsity level and noise standard deviation.

The $150,000$ data points are randomly split into 10 groups of equal size, where each group is interpreted as being the local data for one of the nodes in the network. Each node uses $13,500$ data points as training data and the remaining 1500 points as the validation set. The testing data contains 37640 data points.

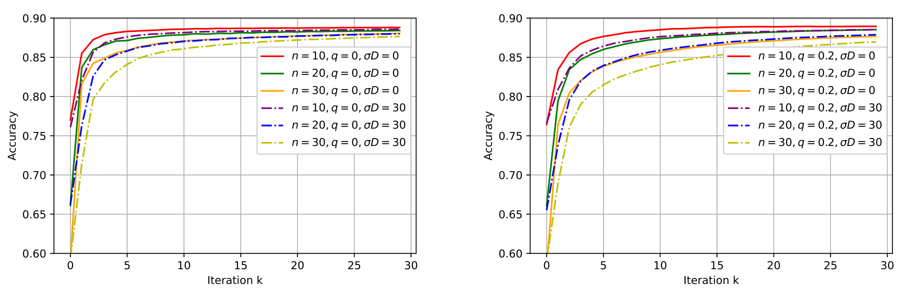


Fig. 6: Accuracy for varied network size.

4.3.2. Varying the network size. In this experiment we explore how the size of the network affects the performance of the proposed algorithm 2. Here the total number of data points is fixed and the number of local data points is inversely proportional to the number of nodes in the network. In Figure 6, we see that the increasing the network size and adding more noise delays the convergence without having much effect on the final accuracy.

4.3.3. Different topology. So far, the network topology is randomly generated at each iteration according to Erdős–Rényi model with some removing edges to make the graph directed. Next, we consider a different type of the generative model. In particular, we consider a topology periodically varying between the two networks shown in Figure 7. This is a much sparser network than the previous ones, rendering the algorithms slower as reflected by the results in Figure 5.

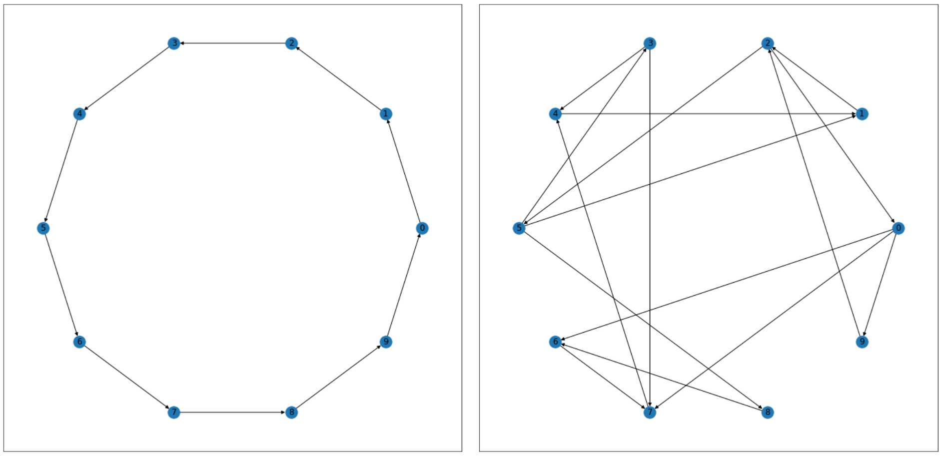
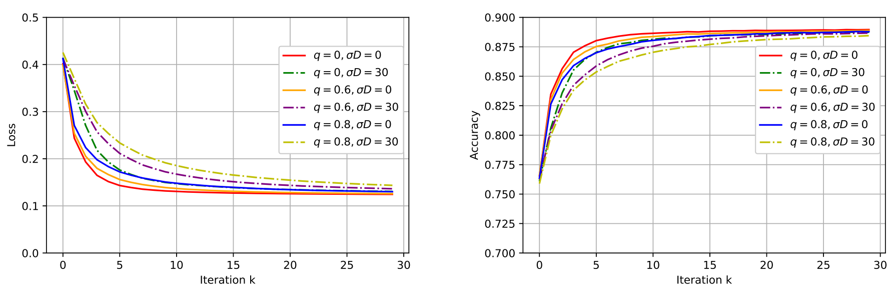


Fig. 7: Topology for periodically changing network



(a) Loss (b) Accuracy

Fig. 8: Results of running the proposed algorithm 2 in the decentralized logistic regression model on a periodic network in Figure 4.

In particular, Figure 5 shows the loss and accuracy for the logistic regression model of algorithm 2; all the remaining parameters of the experiment are the same as in the main paper: the standard deviation $\sigma D = 30$, the step size $\alpha_t = \frac{0.02}{t}$ and the privacy parameter $\delta = 10^{-4}$.

Both privacy and sparsity constraints delay the convergence but do not affect the final loss and accuracy.

5. Formal convergence theorems and proofs.

THEOREM 2.4. *(Theorem 2.4 in main body). Suppose Assumption 2.4 and 2.5 hold. Fix*

$$\gamma \in (0, \min_m \left\{ \frac{1}{(20+8n)^n} (1 - |\lambda_3(\bar{M}_m((k+1)\mathcal{B} - 1 : k\mathcal{B}))|)^n \right\}),$$

and let $\tau = \max_{C \in \mathcal{U}_M} |\lambda_2(C)| < 1$, $\bar{\mathbf{x}} = \frac{1}{n}\sum_{i=1}^n \mathbf{x}_i^0$, *and* $t \geq 0$. *Then running Algorithm 2.1 for* t *iterations, suppose* $t = k\mathcal{B} - 1 + t'$, *where* $t' = 0, \cdots, \mathcal{B} - 1$ *and it* *holds that for any $i \in [n]$ and $t \geq 1$,*

$$\begin{aligned} \|\mathbf{x}_i^t - \bar{\mathbf{z}}^t\| &\leq \sqrt{2nd}(\tau^{1/\mathcal{B}})^{t-(t'-1)} \sum_{j=1}^{2n}\sum_{m=1}^{d} |z_{jm}^0|, \\ \|\mathbf{y}_i^t\| &\leq \sqrt{2nd}(\tau^{1/\mathcal{B}})^{t-(t'-1)} \sum_{j=1}^{2n}\sum_{m=1}^{d} |z_{jm}^0|, \end{aligned} \tag{5.1}$$

where $\bar{\mathbf{z}}^t = \frac{1}{n}\sum_{i=1}^n \mathbf{x}_i^t + \frac{1}{n}\sum_{i=1}^n \mathbf{y}_i^t$. Further, $E[\mathbf{x}_i^t]$ converges to $\bar{\mathbf{x}}$ at a linear rate $O(\tau^{t/\mathcal{B}})$.

Proof. To start with, we observe that the update in Algorithm 1 can be simplified as

$$\begin{aligned} z_{im}^{t+1} &= \sum_{j=1}^{2n} [\bar{M}_m^t]_{ij} [Q(\mathbf{z}_j^t)]_m + \mathbb{1}_{\{t \mod \mathcal{B} = \mathcal{B}-1\}} \gamma [F]_{ij} z_{jm}^{\mathcal{B}\lfloor t/\mathcal{B} \rfloor} \\ &= \sum_{j=1}^{2n} [\bar{M}_m^t]_{ij} z_{jm}^t + \mathbb{1}_{\{t \mod \mathcal{B} = \mathcal{B}-1\}} \gamma [F]_{ij} z_{jm}^{\mathcal{B}\lfloor t/\mathcal{B} \rfloor} \end{aligned} \tag{5.2}$$

This holds because the mixing matrix is constructed such that its entries which multiply zero-valued (i.e., "sparsified") entries of $Q(z_j^t)$ are set to be zero themselves. Next, we review the following lemma which help complete the proof after the incorporation of the noise expectation.

LEMMA 5.1. *[Theorem 2.4 in [10]] Suppose Assumptions 2.4 and 2.5 hold, and instate the notations and hypotheses above. Then, there exist $\sigma \in (0,1)$ and $\Gamma = \sqrt{2nd}$ such that the following statements hold.*
(a) For $1 \leq i \leq n$ and $t = k\mathcal{B} - 1 + t'$, where $t' = 0, \cdots, \mathcal{B}-1$,

$$\|\mathbf{z}_i^t - \bar{\mathbf{z}}\| \leq \Gamma(\tau^{1/\mathcal{B}})^{t-(t'-1)} \sum_{j=1}^{2n}\sum_{m=1}^{d} |z_{jm}^0|, \tag{5.3}$$

where $\bar{\mathbf{z}} = \frac{1}{n}\sum_{i=1}^n \mathbf{x}_i^0 + \frac{1}{n}\sum_{i=1}^n \mathbf{y}_i^0$;
(b) For $1+n \leq i \leq 2n$ and $t = k\mathcal{B} - 1 + t'$, where $t' = 0, \cdots, \mathcal{B}-1$,

$$\|\mathbf{z}_i^t\| \leq \Gamma(\tau^{1/\mathcal{B}})^{t-(t'-1)} \sum_{j=1}^{2n}\sum_{m=1}^{d} |z_{jm}^0|. \tag{5.4}$$

Now we can continue the proof of our theorem. In particular, using Lemma 5.1 above, we have the first part (inequality) in the theorem proved, and establish $\bar{\mathbf{z}}^t = \bar{\mathbf{z}}^0$ for all $t \geq 0$. Since the noise added in the initialization part is unbiased, we have that $E[\bar{\mathbf{z}}^0] = \mathbf{z}^0$ where $\mathbf{z}^0$ represents the initialization without noise. Then we can conclude $\lim_{t\to\infty} E[\mathbf{x}_i^t] = \bar{\mathbf{z}}^0 = \bar{\mathbf{x}}$ and the convergence rate is $O(\tau^{t/\mathcal{B}})$. ∎

THEOREM 2.6. *(Theorem 2.6 in main body) Suppose Assumptions 2.4-2.5 on mixing matrices hold, fix $\gamma \in (0, \min_m \left\{\frac{1}{(20+8n)^n}(1 - |\lambda_3(\bar{M}_m((k+1)\mathcal{B}-1 : k\mathcal{B}))|)^n\right\})$. Assume $|g_{im}^t| \leq D$. If the objective function f is convex, and the step size α_t decays as $O(1/\sqrt{T})$, Algorithm 2.2 converges at a rate of $O(\ln T/\sqrt{T})$, Concretely,*

$$(5.5)\quad (E[f_{\min,T}]-f^*) \le \frac{C_1}{\sum_{k=0}^{\lfloor T/\mathcal{B}\rfloor}\alpha_k} + \frac{C_2\sum_{k=0}^{\lfloor T/\mathcal{B}\rfloor}\alpha_k^2}{\sum_{k=0}^{\lfloor T/\mathcal{B}\rfloor}\alpha_k} \le \frac{C_1}{\sqrt{T/\mathcal{B}}-1} + \frac{C_2\ln(T/\mathcal{B})}{\sqrt{T/\mathcal{B}}-1}$$

where $f_{\min,T} := \min_{t=1,\cdots,T} f(\bar{\mathbf{z}}^t)$ *and* f^* *is the optimal value, and*

$$(5.6)\qquad C_1 = \frac{n\sqrt{d}D^2}{2} + \frac{\sqrt{2}dD}{\sqrt{n}}\sum_{j=1}^{2n}\frac{\|\mathbf{z}_j^0\|}{1-\tau^2},$$

$$(5.7)\quad C_2 = \frac{(d+\sigma^2)nD^2}{2} + \frac{\sqrt{2}dD}{\sqrt{n}}\sum_{j=1}^{2n}\|\mathbf{z}_j^0\| + \frac{2\sqrt{2nd^2(d+\sigma^2)}D^2}{1-\tau} + \frac{4\sqrt{d(d+\sigma^2)}D^2}{n}$$

Proof. Similar to the steps in the consensus case, we start by the following observation to simplify the update in Algorithm 2:

$$\begin{aligned} z_{im}^{t+1} &= \sum_{j=1}^{2n}[\bar{M}_m^t]_{ij}[Q(\mathbf{z}_j^t)]_m + \mathbb{1}_{\{t \mod \mathcal{B}=\mathcal{B}-1\}}\gamma[F]_{ij}z_{jm}^{\mathcal{B}\lfloor t/\mathcal{B}\rfloor} \\ &\quad - \mathbb{1}_{\{t \mod \mathcal{B}=\mathcal{B}-1\}}\alpha_{\lfloor t/\mathcal{B}\rfloor}(g_{im}^{\mathcal{B}\lfloor t/\mathcal{B}\rfloor} + N(0,\sigma^2)) \\ &= \sum_{j=1}^{2n}[\bar{M}_m^t]_{ij}z_{jm}^t + \mathbb{1}_{\{t \mod \mathcal{B}=\mathcal{B}-1\}}\gamma[F]_{ij}z_{jm}^{\mathcal{B}\lfloor t/\mathcal{B}\rfloor} \\ &\quad - \mathbb{1}_{\{t \mod \mathcal{B}=\mathcal{B}-1\}}\alpha_{\lfloor t/\mathcal{B}\rfloor}(g_{im}^{\mathcal{B}\lfloor t/\mathcal{B}\rfloor} + N(0,\sigma^2)) \end{aligned}$$

Let $[\nabla\tilde{f}_i(\mathbf{z}_i^t)]_m = g_{im}^t + N(0,\sigma^2D^2)\mathbb{1}_{(i\le n)}$ be the m^{th} entry of $\nabla\tilde{f}_i(\mathbf{z}_i^t)$ and we then compute

$$(5.8)\quad \|\bar{\mathbf{z}}^{(k+1)\mathcal{B}}-\mathbf{x}^*\|^2 = \|\bar{\mathbf{z}}^{k\mathcal{B}}-\mathbf{x}^*\|^2 + \|\frac{\alpha_k}{n}\sum_{i=1}^n\nabla\tilde{f}_i(\mathbf{z}_i^{k\mathcal{B}})\|^2 - 2\frac{\alpha_k}{n}\sum_{i=1}^n\langle\bar{\mathbf{z}}^{k\mathcal{B}}-\mathbf{x}^*, \nabla\tilde{f}_i(\mathbf{z}_i^{k\mathcal{B}})\rangle.$$

$$\begin{aligned} E[\|\bar{\mathbf{z}}^{(k+1)\mathcal{B}}-\mathbf{x}^*\|^2|\mathcal{F}_{k\mathcal{B}}] &= \|\bar{\mathbf{z}}^{k\mathcal{B}}-\mathbf{x}^*\|^2 + E[\|\frac{\alpha_k}{n}\sum_{i=1}^n\nabla\tilde{f}_i(\mathbf{z}_i^{k\mathcal{B}})\|^2|\mathcal{F}_{k\mathcal{B}}] \\ &\quad - 2\frac{\alpha_k}{n}\sum_{i=1}^n\langle\bar{\mathbf{z}}^{k\mathcal{B}}-\mathbf{x}^*, \nabla f_i(\mathbf{z}_i^{k\mathcal{B}})\rangle \\ &\le \|\bar{\mathbf{z}}^{k\mathcal{B}}-\mathbf{x}^*\|^2 + E[\|\frac{\alpha_k}{n}\sum_{i=1}^n\nabla\tilde{f}_i(\mathbf{z}_i^{k\mathcal{B}})\|^2|\mathcal{F}_{k\mathcal{B}}] \\ &\quad - 2\frac{\alpha_k}{n}\sum_{i=1}^n(-2\sqrt{d}D\|\bar{\mathbf{z}}^{k\mathcal{B}}-\mathbf{z}_i^{k\mathcal{B}}\| + f_i(\bar{\mathbf{z}}^{k\mathcal{B}}) - f_i(\mathbf{x}^*)) \end{aligned}$$

where the last inequality is derived from

$$\begin{aligned} \langle\bar{\mathbf{z}}^{k\mathcal{B}}-\mathbf{x}^*, \nabla f_i(\mathbf{z}_i^{k\mathcal{B}})\rangle &\ge \langle\bar{\mathbf{z}}^{k\mathcal{B}}-\mathbf{z}_i^{k\mathcal{B}}, \nabla f_i(\mathbf{z}_i^{k\mathcal{B}})\rangle + f_i(\mathbf{z}_i^{k\mathcal{B}}) - f_i(\mathbf{x}^*) \\ &\ge -\sqrt{d}D\|\bar{\mathbf{z}}^{k\mathcal{B}}-\mathbf{z}_i^{k\mathcal{B}}\| + f_i(\mathbf{z}_i^{k\mathcal{B}}) - f_i(\bar{\mathbf{z}}^{k\mathcal{B}}) + f_i(\bar{\mathbf{z}}^{k\mathcal{B}}) - f_i(\mathbf{x}^*) \\ &\ge -2\sqrt{d}D\|\bar{\mathbf{z}}^{k\mathcal{B}}-\mathbf{z}_i^{k\mathcal{B}}\| + f_i(\bar{\mathbf{z}}^{k\mathcal{B}}) - f_i(\mathbf{x}^*) \end{aligned}$$

Now, the unconditional expectation satisfies

$$E[\|\bar{\mathbf{z}}^{(k+1)\mathcal{B}} - \mathbf{x}^*\|^2] \le E[\|\bar{\mathbf{z}}^{k\mathcal{B}} - \mathbf{x}^*\|^2] + E[\|\frac{\alpha_k}{n}\sum_{i=1}^{n} \nabla \tilde{f}_i(\mathbf{z}_i^{k\mathcal{B}})\|^2] + 4\frac{\alpha_k\sqrt{d}D}{n}\sum_{i=1}^{n} E[\|\bar{\mathbf{z}}^{k\mathcal{B}} - \mathbf{z}_i^{k\mathcal{B}}\|]$$
$$- 2\frac{\alpha_k}{n}\sum_{i=1}^{n} E[f_i(\bar{\mathbf{z}}^{k\mathcal{B}})] - f_i(x^*).$$

Summing over k from 0 to ∞ and rearranging yields

$$\begin{aligned} 2\sum_{k=0}^{\lfloor T/\mathcal{B}\rfloor} \alpha_k(E[f(\bar{\mathbf{z}}^{k\mathcal{B}})] - f^*) &\le \|\bar{\mathbf{z}}^0 - \mathbf{x}^*\|^2 + n(d+\sigma^2)D^2 \sum_{k=0}^{\lfloor T/\mathcal{B}\rfloor} \alpha_k^2 \\ &+ \frac{4\sqrt{d}D}{n}\sum_{i=1}^{n}\sum_{k=0}^{\lfloor T/\mathcal{B}\rfloor} \alpha_k E[\|\bar{\mathbf{z}}^{k\mathcal{B}} - \mathbf{z}_i^{k\mathcal{B}}\|]. \end{aligned} \tag{5.9}$$

Similar to the derivations for Lemma 3 in [10], we next obtain an upper bound for the last term in (5.9).

Since the update in Algorithm 2.2 implies

$$z_{im}^{k\mathcal{B}} = \sum_{j=1}^{n}[M_m(k\mathcal{B}-1:0)]_{ij} z_{jm}^0 - \sum_{r=1}^{k-1}\sum_{j=1}^{2n}[M_m((k-1)\mathcal{B}-1:(r-1)\mathcal{B})]_{ij}\alpha_{r-1}[\nabla\tilde{f}_j(\mathbf{z}_j^{(r-1)\mathcal{B}})]_m$$
$$- \alpha_{k-1}[\nabla\tilde{f}_i(\mathbf{z}_i^{(k-1)\mathcal{B}})]_m$$

for $i \in \{1, \cdots, 2n\}$ and $m \in \{1, \cdots, d\}$ and using the fact that the mixing matrix and its product have column sum equal to 1,

$$\begin{aligned} \bar{z}_m^{k\mathcal{B}} &= \frac{1}{n}\sum_{j=1}^{2n} z_{jm}^{k\mathcal{B}} \\ &= \frac{1}{n}\sum_{j=1}^{2n} z_{jm}^0 - \frac{1}{n}\sum_{r=1}^{(k-1)\mathcal{B}}\sum_{j=1}^{2n}[\nabla\tilde{f}_j(\mathbf{z}_j^{(r-1)\mathcal{B}})]_m - \frac{1}{n}\sum_{j=1}^{2n}\alpha_{k-1}[\nabla f_j(\tilde{\mathbf{z}_j^{(k-1)\mathcal{B}}})]_m \end{aligned}$$

Then using the gradient norm bound and noise variance, we derive the following upper bound for the last term in (5.9)

$$\begin{aligned} \sum_{i=1}^{n}\sum_{k=0}^{\lfloor T/\mathcal{B}\rfloor} \alpha_k E[\|\bar{\mathbf{z}}^{k\mathcal{B}} - \mathbf{z}_i^{k\mathcal{B}}\|] &\le \sqrt{2nd}\sum_{j=1}^{2n}\|\mathbf{z}_j^0\| \sum_{k=1}^{\lfloor T/\mathcal{B}\rfloor} \alpha_k\tau^k \\ &+ \sqrt{2nd(d+\sigma^2)}nD\sum_{k=1}^{\lfloor T/\mathcal{B}\rfloor}\sum_{r=1}^{k-1}\tau^{k-r}\alpha_k\alpha_{r-1} \\ &+ 2\sqrt{d+\sigma^2}D\sum_{k=0}^{\lfloor T/\mathcal{B}\rfloor-1}\alpha_k^2 \end{aligned} \tag{5.10}$$

Applying $ab \le \frac{1}{2}(a+b)^2$, we have following bounds:

$$\sum_{k=1}^{\lfloor T/\mathcal{B}\rfloor} \alpha_k\tau^k \le \frac{1}{2}\sum_{k=1}^{\lfloor T/\mathcal{B}\rfloor}\alpha_k^2 + \frac{1}{1-\tau^2} \tag{5.11}$$

$$\sum_{k=1}^{\lfloor T/\mathcal{B}\rfloor}\sum_{r=1}^{k-1}\tau^{k-r}\alpha_k\alpha_{r-1} \leq \frac{1}{2}\sum_{k=1}^{\lfloor T/\mathcal{B}\rfloor}\alpha_t^2\sum_{r=1}^{k-1}\tau^{k-r}+\frac{1}{2}\sum_{r=1}^{\lfloor T/\mathcal{B}\rfloor-1}\alpha_{r-1}^2\sum_{k=r+1}^{\lfloor T/\mathcal{B}\rfloor-1}\tau^{k-r} \leq \frac{1}{1-\tau}\sum_{k=1}^{\lfloor T/\mathcal{B}\rfloor}\alpha_k^2 \tag{5.12}$$

Then

$$E[\min_{t=1,\cdots,T} f(\bar{\mathbf{z}}^t)] \to \frac{1}{n}\sum_{i=1}^{n} f_i(x^*) = f^*.$$

Defining $f_{\min} := \min_t f(\bar{\mathbf{z}}^t)$, we have

$$(E[f_{\min}] - f^*)\sum_{k=0}^{\lfloor T/\mathcal{B}\rfloor}\alpha_k \leq \sum_{k=0}^{\lfloor T/\mathcal{B}\rfloor}\alpha_k(f(\bar{\mathbf{z}}^{k\mathcal{B}}) - f^*) \leq C_1 + C_2\sum_{k=0}^{\lfloor T/\mathcal{B}\rfloor}\alpha_k^2, \tag{5.13}$$

where

$$C_1 = \frac{n\sqrt{d}D^2}{2} + \frac{\sqrt{2}dD}{\sqrt{n}}\sum_{j=1}^{2n}\frac{\|\mathbf{z}_j^0\|}{1-\tau^2}, \tag{5.14}$$

$$C_2 = \frac{(d+\sigma^2)nD^2}{2} + \frac{\sqrt{2}dD}{\sqrt{n}}\sum_{j=1}^{2n}\|\mathbf{z}_j^0\| + \frac{2\sqrt{2nd^2(d+\sigma^2)}D^2}{1-\tau} + \frac{4\sqrt{d(d+\sigma^2)}D^2}{n} \tag{5.15}$$

Note that we can express (5.13) equivalently as

$$(E[f_{\min}] - f^*) \leq \frac{C_1}{\sum_{k=0}^{\lfloor T/\mathcal{B}\rfloor}\alpha_k} + \frac{C_2\sum_{k=0}^{\lfloor T/\mathcal{B}\rfloor}\alpha_k^2}{\sum_{k=0}^{\lfloor T/\mathcal{B}\rfloor}\alpha_k}. \tag{5.16}$$

If we select the schedule of stepsizes according to $\alpha_t = O(1/\sqrt{t})$, the first term on the right hand side of (5.16) satisfies

$$\frac{C_1}{\sum_{t=0}^{T}\alpha_t} = C_1\frac{1/2}{\sqrt{T}-1} \tag{5.17}$$

while for the second term it holds that

$$\frac{C_2\sum_{t=0}^{T}\alpha_t^2}{\sum_{t=0}^{T}\alpha_t} = C_2\frac{\ln T}{2(\sqrt{T}-1)} \tag{5.18}$$

Recall that $\sigma = \mathcal{O}(\frac{\sqrt{T}\ln(1/\delta)}{\epsilon})$, then

$$C_2\frac{\ln T}{2(\sqrt{T}-1)} = \mathcal{O}(\frac{\sqrt{d}}{1-\tau}\frac{\ln T}{(\sqrt{T}-1)} + (d + \frac{T(\ln(1/\delta))^2}{\epsilon^2})\frac{\ln T}{(\sqrt{T}-1)})$$

□

THEOREM 2.8. *(Theorem 2.8 in main body) Suppose Assumptions 2.4-2.5 on mixing matrices hold, fix* $\gamma \in (0, \min_m\left\{\frac{1}{(20+8n)^n}(1-|\lambda_3(\bar{M}_m((k+1)\mathcal{B}-1:k\mathcal{B}))|)^n\right\})$. *Assume* $|g_{im}^t| \leq D$ *and let* $\hat{\mathbf{x}}_i{}^T = \frac{\sum_{t=1}^{\lfloor T/\mathcal{B}\rfloor}(t-1)\mathbf{x}_i^t}{t(t-1)/2}$ *for* $T \geq \mathcal{B}$ *and* $K = \lfloor T/\mathcal{B}\rfloor$. *If the* *objective function f is convex and the local objective function f_i is μ_i strongly-convex, then there exists some constant $C_3 > 0, C_4 > 0$ such that for all i*

(5.19)
$$E[f(\hat{\mathbf{x}}_i^T) - f(\mathbf{x}^*)] \le \frac{C_3}{K}\sum_{j=1}^{n}\|\mathbf{x}_j^0\|_1 + \frac{C_4}{K}(1+\ln(K-1)) + \frac{p^2nD^2d}{K}(1+\sigma^2)$$
$$E[\sum_{j=1}^{n}\mu_j\|\hat{\mathbf{x}}_j^T - \mathbf{x}^*\|^2] \le \frac{C_3}{K}\sum_{j=1}^{n}\|\mathbf{x}_j^0\|_1 + \frac{C_4}{K}(1+\ln(K-1)) + \frac{p^2nD^2d}{K}(1+\sigma^2)$$

where the step size $\alpha_t = \frac{p}{t}$ for $p \ge \frac{4n}{\sum_{i=1}^n \mu_i}$, $\mathbf{x}^*$ is the optimal solution, $C_3 = \frac{5\sqrt{2n}ndD}{1-\tau}$ and $C_4 = \frac{5\sqrt{2nd(1+\sigma^2)}n^2dD^2}{1-\tau}$.

Proof. Let $\mathbf{v} \in \mathcal{R}^d$ be any arbitrary vector and ξ_j^t be the noise vector at $\nabla f_j(\mathbf{x}_j^t)$ (denoted as ∇f_j^t shortly). Since $\mathbf{g}_j^t = \nabla f_j(\mathbf{x}_j^t) = \nabla f_j^t$ when $i \in \{1, \cdots, n\}$, for all $t \ge 0$,

$$\begin{aligned}
z_{im}^{t+1} &= \sum_{j=1}^{2n}[\bar{M}_m^t]_{ij}[Q(\mathbf{z}_j^t)]_m + \mathbb{1}_{\{t \mod \mathcal{B}=\mathcal{B}-1\}}\gamma[F]_{ij}z_{jm}^{\mathcal{B}\lfloor t/\mathcal{B}\rfloor} \\
&\quad - \mathbb{1}_{\{t \mod \mathcal{B}=\mathcal{B}-1\}}\alpha_{\lfloor t/\mathcal{B}\rfloor}(g_{im}^{\mathcal{B}\lfloor t/\mathcal{B}\rfloor} + \xi_{jm}^{\mathcal{B}\lfloor t/\mathcal{B}\rfloor}\mathbb{1}_{(i\le n)}) \\
&= \sum_{j=1}^{2n}[\bar{M}_m^t]_{ij}z_{jm}^t + \mathbb{1}_{\{t \mod \mathcal{B}=\mathcal{B}-1\}}\gamma[F]_{ij}z_{jm}^{\mathcal{B}\lfloor t/\mathcal{B}\rfloor} \\
&\quad - \mathbb{1}_{\{t \mod \mathcal{B}=\mathcal{B}-1\}}\alpha_{\lfloor t/\mathcal{B}\rfloor}(g_{im}^{\mathcal{B}\lfloor t/\mathcal{B}\rfloor} + \xi_{jm}^{\mathcal{B}\lfloor t/\mathcal{B}\rfloor}\mathbb{1}_{(i\le n)})
\end{aligned}$$

and, moreover,

$$\begin{aligned}
\bar{z}_m^{t+1} &= \frac{1}{n}\sum_{j=1}^{2n}z_{jm}^t - \frac{1}{n}\alpha_{\lfloor t/\mathcal{B}\rfloor}\sum_{j=1}^{2n}\mathbb{1}_{\{t \mod \mathcal{B}=\mathcal{B}-1\}}(g_{im}^{\mathcal{B}\lfloor t/\mathcal{B}\rfloor} + \xi_{jm}^{\mathcal{B}\lfloor t/\mathcal{B}\rfloor}\mathbb{1}_{(i\le n)}) \\
&= \bar{z}_m^t - \frac{\alpha_{\lfloor t/\mathcal{B}\rfloor}}{n}\sum_{j=1}^{2n}(g_{jm}^{\mathcal{B}\lfloor t/\mathcal{B}\rfloor} + \xi_j^{\mathcal{B}\lfloor t/\mathcal{B}\rfloor}\mathbb{1}_{(j\le n)}).
\end{aligned} \tag{5.20}$$

Then,

(5.21)
$$\|\bar{\mathbf{z}}^{(k+1)\mathcal{B}} - \mathbf{v}\|^2 = \|\bar{\mathbf{z}}^{k\mathcal{B}} - \mathbf{v}\|^2 - \frac{\alpha_k}{n}\sum_{j=1}^{2n}(\mathbf{g}_j^{k\mathcal{B}} + \xi_j^{k\mathcal{B}}\mathbb{1}_{(j\le n)})'(\bar{\mathbf{z}}^{k\mathcal{B}} - \mathbf{v}) + \frac{\alpha_k^2}{n^2}\|\sum_{j=1}^{2n}\mathbf{g}_j^{k\mathcal{B}} + \xi_j^{k\mathcal{B}}\mathbb{1}_{(j\le n)}\|^2$$

(5.22)
$$= \|\bar{\mathbf{z}}^{k\mathcal{B}} - \mathbf{v}\|^2 - \frac{\alpha_k}{n}\sum_{j=1}^{n}(\nabla f_j^{k\mathcal{B}} + \xi_j^{k\mathcal{B}})'(\bar{\mathbf{z}}^{k\mathcal{B}} - \mathbf{v}) + \frac{\alpha_k^2}{n^2}\|\sum_{j=1}^{n}\nabla f_j^{k\mathcal{B}} + \xi_j^{k\mathcal{B}}\|^2. \qquad \blacksquare$$

Now, we rewrite each cross-term $(\nabla f_j^t)'(\bar{\mathbf{z}}^t - \mathbf{v})$ as

$$\begin{aligned}(\nabla f_j^t)'(\bar{\mathbf{z}}^t - \mathbf{v}) &= (\nabla f_j^t)'(\bar{\mathbf{z}}^t - \mathbf{x}_j^t) + (\nabla f_j^t)'(\mathbf{x}_j^t - \mathbf{v}) \\ &\geq -\sqrt{d}D\|\bar{\mathbf{z}}^t - \mathbf{x}_j^t\| + f_j(\mathbf{x}_j^t) - f_j(\mathbf{v}) + \frac{\mu_j}{2}\|\mathbf{x}_j^t - \mathbf{v}\|^2 \\ &= -\sqrt{d}D\|\bar{\mathbf{z}}^t - \mathbf{x}_j^t\| + (f_j(\mathbf{x}_j^t) - f_j(\bar{\mathbf{z}}^t)) + (f_j(\bar{\mathbf{z}}^t) - f_j(\mathbf{v})) \\ &\quad + \frac{\mu_j}{2}\|\mathbf{x}_j^t - \mathbf{v}\|^2 \\ &\geq -2\sqrt{d}D\|\bar{\mathbf{z}}^t - \mathbf{x}_j^t\| + (f_j(\bar{\mathbf{z}}^t) - f_j(\mathbf{v})) + \frac{\mu_j}{2}\|\mathbf{x}_j^t - \mathbf{v}\|^2.\end{aligned} \tag{5.23}$$

Using $f(\mathbf{x}) = \frac{1}{n}\sum_{j=1}^n f_j(\mathbf{x})$,

$$\sum_{j=1}^n (\nabla f_j^{k\mathcal{B}})'(\bar{\mathbf{z}}^{k\mathcal{B}} - \mathbf{v}) \geq n(f(\mathbf{x}^{k\mathcal{B}}) - f(\mathbf{v})) + \frac{1}{2}\sum_{j=1}^n \mu_j\|\mathbf{x}_j^{k\mathcal{B}} - \mathbf{v}\|^2 - 2\sum_{j=1}^n \sqrt{d}D\|\bar{\mathbf{z}}^{k\mathcal{B}} - \mathbf{x}_j^{k\mathcal{B}}\|. \tag{5.24}$$

Hence, we have shown that

$$\begin{aligned}E[\|\bar{\mathbf{z}}^{(k+1)\mathcal{B}} - \mathbf{v}\|^2 | \mathcal{F}_{k\mathcal{B}}] &\leq \|\bar{\mathbf{z}}^{k\mathcal{B}} - \mathbf{v}\|^2 - 2\alpha_k(f(\bar{\mathbf{z}}^{k\mathcal{B}}) - f(\mathbf{v})) - \frac{\alpha_k}{n}\sum_{j=1}^n \mu_j\|\mathbf{x}_j^{k\mathcal{B}} - \mathbf{v}\|^2 \\ &\quad + \frac{4\alpha_k}{n}\sum_{j=1}^n \sqrt{d}D\|\mathbf{x}_j^{k\mathcal{B}} - \bar{\mathbf{z}}^{k\mathcal{B}}\| + \frac{\alpha_k^2}{n^2}\sum_{j=1}^n (\sqrt{d}D + \sigma\sqrt{d}D)^2 \\ &\leq \|\bar{\mathbf{z}}^{k\mathcal{B}} - \mathbf{v}\|^2 - 2\alpha_k(f(\bar{\mathbf{z}}^{k\mathcal{B}}) - f(\mathbf{v})) - \frac{\alpha_k}{n}\sum_{j=1}^n \mu_j\|\mathbf{x}_j^{k\mathcal{B}} - \mathbf{v}\|^2 \\ &\quad + \frac{4\alpha_k}{n}\sum_{j=1}^n \sqrt{d}D\|\mathbf{x}_j^{k\mathcal{B}} - \bar{\mathbf{z}}^{k\mathcal{B}}\| + \frac{\alpha_k^2}{n^2}\sum_{j=1}^n d(1+\sigma^2)D^2.\end{aligned}$$

Then we can replace $\mathbf{v}$ by the optimal solution $\mathbf{x}^*$ and this gives

$$\begin{aligned}E[\|\bar{\mathbf{z}}^{(k+1)\mathcal{B}} - \mathbf{x}^*\|^2 | \mathcal{F}_{k\mathcal{B}}] &\leq \|\bar{\mathbf{z}}^{k\mathcal{B}} - \mathbf{x}^*\|^2 - 2\alpha_k(f(\bar{\mathbf{z}}^{k\mathcal{B}}) - f(\mathbf{x}^*)) - \frac{\alpha_k}{n}\sum_{j=1}^n \mu_j\|\mathbf{x}_j^{k\mathcal{B}} - \mathbf{x}^*\|^2 \\ &\quad + \frac{4\alpha_k}{n}\sum_{j=1}^n \sqrt{d}D\|\mathbf{x}_j^{k\mathcal{B}} - \bar{\mathbf{z}}^{k\mathcal{B}}\| + \frac{\alpha_k^2}{n^2}\sum_{j=1}^n d(1+\sigma^2)D^2.\end{aligned} \tag{5.25}$$

Since $f(\mathbf{x}) = \frac{1}{n}\sum_{j=1}^n f_j(\mathbf{x})$ is convex, each local objective function f_i is μ_i-strongly-convex and the upper bound on the gradient magnitude is $|g_{im}^t| \leq D$, the following two inequalities hold:

(a)

$$f(\bar{\mathbf{z}}^t) - f(\mathbf{x}^*) \geq \frac{1}{2n}(\sum_{j=1}^n \mu_j)\|\bar{\mathbf{z}}^t - \mathbf{x}^*\| \tag{5.26}$$

(b)

$$f(\bar{\mathbf{z}}^t) - f(\mathbf{x}^*) \geq -\frac{L}{n}\|\mathbf{x}_i^t - \bar{\mathbf{z}}^t\| + f(\mathbf{x}_i^t) - f(\mathbf{x}^*) \tag{5.27}$$

where $L = n\sqrt{d}D$, for any $i = 1, \cdots, n$.

The above imply that for all $i = 1, \dots, n$,

$$2(f(\bar{\mathbf{z}}^t) - f(\mathbf{x}^*)) \geq \frac{1}{2}(\frac{1}{n}\sum_{j=1}^{n}\mu_j)\|\bar{\mathbf{z}}^t - \mathbf{x}^*\| - \frac{L}{n}\|\mathbf{x}_i^t - \bar{\mathbf{z}}^t\| + f(\mathbf{x}_i^t) - f(\mathbf{x}^*). \tag{5.28}$$

Now, for each $i = 1, \dots, n$,

$$E[\|\bar{\mathbf{z}}^{(k+1)\mathcal{B}} - \mathbf{x}^*\|^2|\mathcal{F}_{k\mathcal{B}}] \leq \|\bar{\mathbf{z}}^{k\mathcal{B}} - \mathbf{x}^*\|^2 - \frac{\alpha_k}{n}(\frac{1}{2}(\sum_{j=1}^{n}\mu_j)\|\bar{\mathbf{z}}^{k\mathcal{B}} - \mathbf{x}^*\| - L\|\mathbf{x}_i^{k\mathcal{B}} - \bar{\mathbf{z}}^{k\mathcal{B}}\| + n(f(\mathbf{x}_i^{k\mathcal{B}}) - f(\mathbf{x}^*)))$$
$$- \frac{\alpha_k}{n}\sum_{j=1}^{n}\mu_j\|\mathbf{x}_j^{k\mathcal{B}} - \mathbf{x}^*\|^2 + \frac{4\alpha_k}{n}\sum_{j=1}^{n}\sqrt{d}D\|\mathbf{x}_j^{k\mathcal{B}} - \bar{\mathbf{z}}^{k\mathcal{B}}\| + \frac{\alpha_k^2}{n}\sum_{j=1}^{n}(\sqrt{d}D + \sigma\sqrt{d}D)^2.$$

Let $\alpha_k = \frac{p}{k+1}$; since $p\frac{\sum_{i=1}^{n}\mu_i}{n} \geq 4$,

$$E[\|\bar{\mathbf{z}}^{(k+1)\mathcal{B}} - \mathbf{x}^*\|^2|\mathcal{F}_{k\mathcal{B}}] \leq (1 - \frac{2}{k+1})\|\bar{\mathbf{z}}^{k\mathcal{B}} - \mathbf{x}^*\|^2 - \frac{p}{(k+1)}(f(\mathbf{x}_i^{k\mathcal{B}}) - f(\mathbf{x}^*))$$
$$+ \frac{pL}{n(k+1)}\|\mathbf{x}_i^{k\mathcal{B}} - \bar{\mathbf{z}}^{k\mathcal{B}}\| - \frac{p}{n(t+1)}\sum_{j=1}^{n}\mu_j\|\mathbf{x}_j^{k\mathcal{B}} - \mathbf{x}^*\|^2$$
$$+ \frac{4p}{n(k+1)}\sum_{j=1}^{n}\sqrt{d}D\|\mathbf{x}_j^{k\mathcal{B}} - \bar{\mathbf{z}}^{k\mathcal{B}}\| + \frac{p^2}{n(k+1)^2}\sum_{j=1}^{n}(\sqrt{d}D + \sigma\sqrt{d}D)^2.$$

Multiply both sides of the above inequality by $k(k+1)$ and taking the expectation yields for all $T \geq \mathcal{B}$, let $K = \lfloor T/\mathcal{B} \rfloor$,

(5.29)

$$K(K-1)E[\|\bar{\mathbf{z}}^{K\mathcal{B}} - \mathbf{x}^*\|^2] \leq -\frac{p}{n}\sum_{k=1}^{K-1} tE[n(f(\mathbf{x}_i^{k\mathcal{B}}) - f(\mathbf{x}^*)) + \sum_{j=1}^{n}\mu_j\|\mathbf{x}_j^{k\mathcal{B}} - \mathbf{x}^*\|^2]$$
$$+ \frac{pL}{n}\sum_{k=1}^{K-1} kE[\|\mathbf{x}_i^{k\mathcal{B}} - \bar{\mathbf{z}}^{k\mathcal{B}}\|] - \frac{p}{n(k+1)}(\sum_{j=1}^{n}\mu_j)E[\|\mathbf{x}_j^{k\mathcal{B}} - \mathbf{x}^*\|^2]$$
$$+ \frac{4p}{n}\sum_{k=1}^{K-1} t\sum_{j=1}^{n}\sqrt{d}DE[\|\mathbf{x}_j^{k\mathcal{B}} - \bar{\mathbf{z}}^{k\mathcal{B}}\|] + \frac{p^2}{n}\sum_{j=1}^{n}(\sqrt{d}D + \sigma\sqrt{d}D)^2\sum_{k=1}^{K-1}\frac{k}{k+1}.$$

To derive the upper bound on $E[\sum_{k=1}^{K-1}\|\mathbf{x}_i^{k\mathcal{B}} - \bar{\mathbf{z}}^{k\mathcal{B}}\|]$, we refer to Lemma 3 in [10] and Corollary 1 and 2 in [36]. In particular, we have the following

$$E[\sum_{k=1}^{K}\|\mathbf{x}_i^{k\mathcal{B}} - \bar{\mathbf{z}}^{k\mathcal{B}}\|] \leq C_3'\sum_{j=1}^{n}\|\mathbf{x}_j^0\|_1 + C_4'(1 + \ln K). \tag{5.30}$$

where

$$C_3' = \frac{\sqrt{2nd}}{1-\tau}, \quad C_4' = \frac{\sqrt{2nd^2(1+\sigma^2)}nD}{1-\tau} \tag{5.31}$$

Therefore,

$$\frac{1}{K(K-1)}\sum_{k=1}^{K-1} tE[n(f(\mathbf{x}_i^{k\mathcal{B}})-f(\mathbf{x}^*))+\sum_{j=1}^{n}\mu_j\|\mathbf{x}_j^{k\mathcal{B}}-\mathbf{x}^*\|^2] \le \frac{5L}{T}(C_3'\sum_{j=1}^{n}\|\mathbf{x}_j^0\|_1 + C_4'(1+\ln(K-1))) + \frac{p^2}{K}\sum_{j=1}^{n} d(1+\sigma^2)D^2 \tag{5.32}$$

Hence,

$$\frac{1}{K(K-1)}\sum_{k=1}^{K-1} kE[(f(\mathbf{x}_i^{k\mathcal{B}})-f(\mathbf{x}^*))+\sum_{j=1}^{n}\mu_j\|\mathbf{x}_j^{k\mathcal{B}}-\mathbf{x}^*\|^2] \le \frac{5L}{K}(C_3'\sum_{j=1}^{n}\|\mathbf{x}_j^0\|_1 + C_4'(1+\ln(K-1)))+ \frac{p^2}{K}\sum_{j=1}^{n} d(1+\sigma^2)D^2 \tag{5.33}$$

By convexity, for each $i \in [n]$ it holds that

$$\frac{2}{K(K-1)}\sum_{k=1}^{K-1} t(f(\mathbf{x}_i^{k\mathcal{B}})-f(\mathbf{x}^*))+\sum_{j=1}^{n}\mu_j\|\mathbf{x}_j^{k\mathcal{B}}-\mathbf{x}^*\|^2 \ge f(\hat{\mathbf{x}}^K)-f(\mathbf{x}^*)+\sum_{j=1}^{n}\mu_j\|\hat{\mathbf{x}}^K-\mathbf{x}^*\|^2, \tag{5.34}$$

∎

where $\hat{\mathbf{x}_i}^K = \frac{\sum_{k=1}^K (k-1)\mathbf{x}_i^k}{k(k-1)/2}$ for $K \ge 2$.

Setting $C_3 = 5LC_3' = \frac{5\sqrt{2n}ndD}{1-\tau}$ and $C_4 = 5LC_4' = \frac{5\sqrt{2nd(1+\sigma^2)}n^2dD^2}{1-\tau}$ completes the proof.

6. Conclusion.

In this paper we propose differentially private and communication efficient algorithms for decentralized consensus and optimization over directed time-varying graphs. Our results introduce these techniques to a large class of real world applications operating under resource constraints. We provide theoretical guarantees and numerical validation of the proposed methods in several settings.

Future work includes extending these results to non-convex settings with more sophisticated tasks such as language modelling and speech processing. Moreover, it is of interest to study stochastic gradient methods as they will reduce local computations. Finally, an orthogonal direction worth exploring involves security models that account for adversarial attacks.

REFERENCES

[1] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang, *Deep learning with differential privacy*, in Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, 2016, pp. 308–318.

[2] N. Agarwal, A. T. Suresh, F. X. X. Yu, S. Kumar, and B. McMahan, *cpsgd: Communication-efficient and differentially-private distributed sgd*, in Advances in Neural Information Processing Systems 31, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, eds.,

Curran Associates, Inc., 2018, pp. 7564–7575, http://papers.nips.cc/paper/7984-cpsgd-communication-efficient-and-differentially-private-distributed-sgd.pdf.

[3] M. Assran, N. Loizou, N. Ballas, and M. Rabbat, *Stochastic gradient push for distributed deep learning*, arXiv preprint arXiv:1811.10792, (2018).

[4] A. Bellet, R. Guerraoui, M. Taziki, and M. Tommasi, *Personalized and private peer-to-peer machine learning*, arXiv preprint arXiv:1705.08435, (2017).

[5] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, *Gossip algorithms: design, analysis and applications*, in Proceedings IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies., vol. 3, 2005, pp. 1653–1664 vol. 3.

[6] K. Cai and H. Ishii, *Average consensus on general strongly connected digraphs*, Automatica, 48 (2012), pp. 2750–2761.

[7] K. Cai and H. Ishii, *Average consensus on arbitrary strongly connected digraphs with time-varying topologies*, IEEE Transactions on Automatic Control, 59 (2014), pp. 1066–1071.

[8] T.-H. Chang, M. Hong, and X. Wang, *Multi-agent distributed optimization via inexact consensus admm*, IEEE Transactions on Signal Processing, 63 (2014), pp. 482–497.

[9] W.-N. Chen, P. Kairouz, and A. Özgür, *Breaking the communication-privacy-accuracy trilemma*, arXiv preprint arXiv:2007.11707, (2020).

[10] Y. Chen, A. Hashemi, and H. Vikalo, *Communication-efficient algorithms for decentralized optimization over directed graphs*, 2020, https://arxiv.org/abs/2005.13189.

[11] J. Cortés, G. E. Dullerud, S. Han, J. Le Ny, S. Mitra, and G. J. Pappas, *Differential privacy in control and network systems*, in 2016 IEEE 55th Conference on Decision and Control (CDC), 2016, pp. 4252–4272.

[12] J. Davin and P. J. Modi, *Impact of problem centralization in distributed constraint optimization algorithms*, in Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems, 2005, pp. 1057–1063.

[13] A. D. Domínguez-García and C. N. Hadjicostis, *Distributed strategies for average consensus in directed graphs*, in 2011 50th IEEE Conference on Decision and Control and European Control Conference, IEEE, 2011, pp. 2124–2129.

[14] J. C. Duchi, A. Agarwal, and M. J. Wainwright, *Dual averaging for distributed optimization: Convergence analysis and network scaling*, IEEE Transactions on Automatic Control, 57 (2012), pp. 592–606.

[15] C. Dwork, F. McSherry, K. Nissim, and A. Smith, *Calibrating noise to sensitivity in private data analysis*, Springer, 2006, pp. 265–284.

[16] C. Dwork, A. Roth, et al., *The algorithmic foundations of differential privacy*, Foundations and Trends® in Theoretical Computer Science, 9 (2014), pp. 211–407.

[17] P. Erdös, A. Rényi, et al., *On random graphs*, Publicationes mathematicae, 6 (1959), pp. 290–297.

[18] A. Hard, K. Rao, R. Mathews, S. Ramaswamy, F. Beaufays, S. Augenstein, H. Eichner, C. Kiddon, and D. Ramage, *Federated learning for mobile keyboard prediction*, arXiv preprint arXiv:1811.03604, (2018).

[19] J. He, L. Cai, C. Zhao, P. Cheng, and X. Guan, *Privacy-preserving average consensus: Privacy analysis and algorithm design*, IEEE Transactions on Signal and Information Processing over Networks, 5 (2019), pp. 127–138.

[20] A. Jadbabaie, Jie Lin, and A. S. Morse, *Coordination of groups of mobile autonomous agents using nearest neighbor rules*, IEEE Transactions on Automatic Control, 48 (2003), pp. 988–1001.

[21] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings, et al., *Advances and open problems in federated learning*, arXiv preprint arXiv:1912.04977, (2019).

[22] P. Kairouz, S. Oh, and P. Viswanath, *The composition theorem for differential privacy*, in International conference on machine learning, PMLR, 2015, pp. 1376–1385.

[23] S. P. Kasiviswanathan, H. K. Lee, K. Nissim, S. Raskhodnikova, and A. Smith, *What can we learn privately?*, SIAM Journal on Computing, 40 (2011), pp. 793–826.

[24] A. Koloskova, T. Lin, S. U. Stich, and M. Jaggi, *Decentralized deep learning with arbitrary communication compression*, arXiv preprint arXiv:1907.09356, (2019).

[25] X. Li, W. Yang, S. Wang, and Z. Zhang, *Communication efficient decentralized training with multiple local updates*, arXiv preprint arXiv:1910.09126, (2019).

[26] Y. Li, L. Guo, and S. K. Prasad, *An energy-efficient distributed algorithm for minimum-latency aggregation scheduling in wireless sensor networks*, in 2010 IEEE 30th International Conference on Distributed Computing Systems, 2010, pp. 827–836.

[27] X. Lian, C. Zhang, H. Zhang, C.-J. Hsieh, W. Zhang, and J. Liu, *Can decentralized algorithms outperform centralized algorithms? a case study for decentralized parallel stochastic*

gradient descent, in Advances in Neural Information Processing Systems 30, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, eds., Curran Associates, Inc., 2017, pp. 5330–5340.

[28] X. Lian, C. Zhang, H. Zhang, C.-J. Hsieh, W. Zhang, and J. Liu, *Can decentralized algorithms outperform centralized algorithms? a case study for decentralized parallel stochastic gradient descent*, in Advances in Neural Information Processing Systems, 2017, pp. 5330–5340.

[29] H. B. McMahan, G. Andrew, U. Erlingsson, S. Chien, I. Mironov, N. Papernot, and P. Kairouz, *A general approach to adding differential privacy to iterative training procedures*, arXiv preprint arXiv:1812.06210, (2018).

[30] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, et al., *Communication-efficient learning of deep networks from decentralized data*, arXiv preprint arXiv:1602.05629, (2016).

[31] I. Mironov, *Rényi differential privacy*, in 2017 IEEE 30th Computer Security Foundations Symposium (CSF), IEEE, 2017, pp. 263–275.

[32] I. Mironov, K. Talwar, and L. Zhang, *R\'enyi differential privacy of the sampled gaussian mechanism*, arXiv preprint arXiv:1908.10530, (2019).

[33] A. Narayanan and V. Shmatikov, *Robust de-anonymization of large sparse datasets*, in 2008 IEEE Symposium on Security and Privacy (sp 2008), IEEE, 2008, pp. 111–125.

[34] A. Nedić, S. Lee, and M. Raginsky, *Decentralized online optimization with global objectives and local communication*, in 2015 American Control Conference (ACC), IEEE, 2015, pp. 4497–4503.

[35] A. Nedić and A. Olshevsky, *Distributed optimization over time-varying directed graphs*, IEEE Transactions on Automatic Control, 60 (2014), pp. 601–615.

[36] A. Nedić and A. Olshevsky, *Stochastic gradient-push for strongly convex functions on time-varying directed graphs*, IEEE Transactions on Automatic Control, 61 (2016), pp. 3936–3947.

[37] A. Nedic, A. Olshevsky, and W. Shi, *Achieving geometric convergence for distributed optimization over time-varying graphs*, SIAM Journal on Optimization, 27 (2017), pp. 2597–2633.

[38] A. Nedic and A. Ozdaglar, *Distributed subgradient methods for multi-agent optimization*, IEEE Transactions on Automatic Control, 54 (2009), pp. 48–61.

[39] E. Nozari, P. Tallapragada, and J. Cortés, *Differentially private average consensus: Obstructions, trade-offs, and optimal algorithm design*, Automatica, 81 (2017), pp. 221–231.

[40] A. Paverd, A. Martin, and I. Brown, *Modelling and automatically analysing privacy properties for honest-but-curious adversaries*.

[41] S. Reddi, Z. Charles, M. Zaheer, Z. Garrett, K. Rush, J. Konečný, S. Kumar, and H. B. McMahan, *Adaptive federated optimization*, arXiv preprint arXiv:2003.00295, (2020).

[42] W. Ren and R. W. Beard, *Consensus seeking in multiagent systems under dynamically changing interaction topologies*, IEEE Transactions on automatic control, 50 (2005), pp. 655–661.

[43] W. Ren, H. Chao, W. Bourgeous, N. Sorensen, and Y. Chen, *Experimental validation of consensus algorithms for multivehicle cooperative control*, IEEE Transactions on Control Systems Technology, 16 (2008), pp. 745–752.

[44] D. Shah, *Gossip algorithms*, Now Publishers Inc, 2009.

[45] S. U. Stich, J.-B. Cordonnier, and M. Jaggi, *Sparsified sgd with memory*, in Advances in Neural Information Processing Systems, 2018, pp. 4447–4458.

[46] A. T. Suresh, F. X. Yu, S. Kumar, and H. B. McMahan, *Distributed mean estimation with limited communication*, in Proceedings of the 34th International Conference on Machine Learning-Volume 70, JMLR. org, 2017, pp. 3329–3337.

[47] H. Tang, S. Gan, C. Zhang, T. Zhang, and J. Liu, *Communication compression for decentralized training*, in Advances in Neural Information Processing Systems 31, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, eds., Curran Associates, Inc., 2018, pp. 7652–7662, http://papers.nips.cc/paper/7992-communication-compression-for-decentralized-training.pdf.

[48] H. Tang, S. Gan, C. Zhang, T. Zhang, and J. Liu, *Communication compression for decentralized training*, in Advances in Neural Information Processing Systems, 2018, pp. 7652–7662.

[49] J. Wang and G. Joshi, *Cooperative sgd: A unified framework for the design and analysis of communication-efficient sgd algorithms*, arXiv preprint arXiv:1808.07576, (2018).

[50] J. Wang, A. K. Sahu, Z. Yang, G. Joshi, and S. Kar, *Matcha: Speeding up decentralized sgd via matching decomposition sampling*, arXiv preprint arXiv:1905.09435, (2019).

[51] Y. Wang, *Privacy-preserving average consensus via state decomposition*, IEEE Transactions

on Automatic Control, 64 (2019), pp. 4711–4716.

[52] E. Wei and A. Ozdaglar, *Distributed alternating direction method of multipliers*, in 2012 IEEE 51st IEEE Conference on Decision and Control (CDC), IEEE, 2012, pp. 5445–5450.

[53] Wei Ren and R. W. Beard, *Consensus seeking in multiagent systems under dynamically changing interaction topologies*, IEEE Transactions on Automatic Control, 50 (2005), pp. 655–661.

[54] X. Wu, F. Li, A. Kumar, K. Chaudhuri, S. Jha, and J. Naughton, *Bolt-on differential privacy for scalable stochastic gradient descent-based analytics*, ACM, 2017, pp. 1307–1322.

[55] C. Xi, Q. Wu, and U. A. Khan, *On the distributed optimization over directed networks*, Neurocomputing, 267 (2017), pp. 508–515.