Dropout-Robust Mechanisms for Differentially Private and Fully Decentralized Mean Estimation

César Sabater
CNRS, INSA Lyon
Villeurbanne, France
cesar.sabater@insa-lyon.fr

Sonia Ben Mokhtar
CNRS, INSA Lyon
Villeurbanne, France
sonia.ben-mokhtar@cnrs.fr

Jan Ramon
INRIA Lille
Villeneuve d'Ascq, France
jan.ramon@inria.fr

Abstract

Achieving differentially private computations in decentralized settings poses significant challenges, particularly regarding accuracy, communication cost, and robustness against information leakage. While cryptographic solutions offer promise, they often suffer from high communication overhead or require centralization in the presence of network failures. Conversely, existing fully decentralized approaches typically rely on relaxed adversarial models or pairwise noise cancellation, the latter suffering from substantial accuracy degradation if parties unexpectedly disconnect. In this work, we propose IncA, a new protocol for fully decentralized mean estimation, a powerful primitive in data-intensive processing. Our protocol, which enforces differential privacy, requires no central orchestration and employs low-variance correlated noise, achieved by incrementally injecting sensitive information into the computation. First, we theoretically demonstrate that, when no parties permanently disconnect, our protocol achieves accuracy comparable to that of a centralized setting—already an improvement over most existing decentralized differentially private techniques. Second, we empirically show that our use of low-variance correlated noise significantly mitigates the accuracy loss experienced by existing techniques in the presence of dropouts.

Keywords

differential privacy, decentralized mean estimation, decentralized optimization

1 Introduction

The training of machine learning models commonly relies on the centralized processing of large datasets. While this approach offers simplicity, it presents substantial privacy implications when sensitive data is involved, or when data aggregation is limited by sharing constraints. For instance, private companies or hospitals might be reluctent to share their data for jointly training prediction models due to competitive secrecy or juridic regulations. For these reasons, algorithms where the data remains local to their owners have gained significant popularity in the last decade. One prominent example is Federated Learning [47, 52], a framework in which participating entities iteratively perform local model training and subsequently transmit intermediate model updates to a central server for aggregation. While this architecture enhances data governance, it nonetheless introduces new challenges in terms of privacy and robustness. Indeed, computational processes relying on a large participant base give rise to vulnerabilities to participant behavior, including failures or corruption by adversaries. This may compromise both the integrity of the outcome and participant privacy through inference attacks [33, 49, 53, 55, 59, 65].

Crucially, Federated Learning still relies on centralized coordination. This substantially concentrates the security and robustness of the computation on a single entity, which is often a server owned by a company or another kind of organization. This has two important drawbacks. The first is applicability, as a party that is computationally capable of processing the messages of all parties and can be trusted not to crash or disconnect is not always available. The second is robustness, as compromising this single entity creates significant privacy and security risks due to its central control and full visibility over the computation.

For these reasons, a central coordinator may be undesirable or simply unavailable in certain settings. To overcome this limitation, entirely decentralized machine learning algorithms have been developed [9, 39]. In this context, decentralized averaging algorithms, which are the primary focus of this paper, constitute a fundamental component enabling parties to average their model parameters or gradients towards model convergence. Specifically, we concentrate on the canonical task of averaging a set of private values held by participants. Despite its apparent simplicity, averaging serves as a critical primitive in decentralized learning and various other machine learning and data mining tasks, including recommendation systems[58], clustering, matrix factorization[57], decision trees, empirical cumulative distribution functions[6] and linear regression. Fundamentally, its applicability extends to any task amenable to decomposition into local computations followed by private averaging.

Our objective is then to analyze the privacy of decentralized averaging algorithms through the lens of *differential privacy (DP)* [28]. This framework has emerged as a gold standard in privacy research due to its solid theoretical guarantees. Nevertheless, demonstrating that an algorithm satisfies DP is often challenging, typically necessitating the injection of noise into computations, which degrades accuracy. This challenge is particularly acute within the local differential privacy (LDP) framework [15, 25, 45, 48, 50]. In LDP, where data is fully privatized before it is used in collaborative computations, the required noise levels are frequently prohibitively high, rendering computations practically unusable in many real-world scenarios.

The integration of cryptographic primitives, such as secure multiparty computation (specifically, Secure Aggregation [8, 12]) or fully homomorphic encryption, into decentralized computations offers a promising solution to substantially reduce the noise levels

Proceedings on Privacy Enhancing Technologies 2026(2), 472–498

https://doi.org/10.56553/popets-2026-0057

required for achieving differential privacy (DP) guarantees. This approach can yield privacy-accuracy trade-offs comparable to those observed in central DP settings [2, 16, 27, 40, 44, 46], where a single trusted party or curator is responsible for noise injection. However, a significant drawback of these cryptographic methods is their high computational cost, particularly as the number of involved parties increases, or their reliance on a trusted central entity for orchestration and fault tolerance (e.g., handling participant dropouts).

New challenges arise when privacy-preserving computations lack a central coordinator trusted to follow the protocol. In a decentralized setting, each party has a partial view of the protocol. The lack of a single consistent view of the full set of parties' interactions reduces the capability of recovery from failures and the detection of actively corrupted parties.

Several techniques can reduce the noise of local DP and be executed without central coordination. These solutions consist of gossip algorithms in which exchanges are similar to the protocols proposed in [13]. However, they often rely on DP relaxations [18, 20], which can limit practical applicability, or necessitate correlated bounded noise (e.g., pairwise canceling Gaussian noise [4, 61, 66]). The latter approaches either incur high communication costs or demand the cancellation of high-variance noise, which substantially impacts accuracy in the presence of dropouts.

In our work, we propose IncA, a protocol designed to minimize the impact of inconsistencies in the computation, such as failures or connection problems of participants. This is achieved by splitting information injection into multiple iterations and reducing the impact of inconsistent messages. Our approach is particularly advantageous when failure-recovery mechanisms are too costly or infeasible. Therefore, it is a good candidate in the decentralized setting. IncA (i) provides the same privacy-accuracy trade-offs as Central DP in the absence of dropouts, (ii) satisfies the classical notion of DP, (iii) is robust against a fraction of parties colluding and attempting to learn information about honest participants, and (iv) bounds impact of dropouts through the use of low variance correlated noise.

1.1 Contributions

Our contributions are the following:

(1) We propose a novel protocol for private averaging. It involves iterative exchanges of messages in a way similar to the gossip protocols in [13] with the addition of correlated noise. The latter ensures privacy with a minimal harm in accuracy. We provide a generic construction in which the private values as well as the noise injected in the computation are customizable, allowing to choose the most adequate strategy for different scenarios. Our approach differs from the common technique of previous correlated noise approaches [4, 61, 66] and Secure Aggregation [7, 8, 12, 27, 64], that are based in pairwise noise masks shared between users. In our approach, each user knows its own noise terms and cancels them in progressive updates.
(2) When parties do not drop out of the protocol, we prove differential privacy guarantees even if the adversary (i) corrupts a proportion of the participants to passively share their gathered information, (ii) observes a proportion the exchanges, and (iii) knows all the network interactions (i.e., who communicated with whom). While proven to be differentially private, our protocol matches the utility of Secure Aggregation combined with local noise [2, 16, 40, 46] without the use cryptographic primitives or a requiring a central orchestrator. This privacy-utility trade-off is comparable to Central DP. We provide two characterizations of our privacy guarantees: the first is more accurate while the second is more interpretable. For the latter, we show that our guarantees hold when a sufficiently large and diverse set of exchanges have been performed between honest parties.
(3) We prove that, given the graph that models the hidden exchanges between honest parties, certain topologies provide sufficient conditions for privacy or the lack of it. We provide positive and negative results. The former are for strongly connected topologies. The latter are for graphs where no party changes their neighbors across iterations and the adversary continually observes at least two honest parties.
(4) We extend our privacy analysis to the case of dropouts, where correlated noise does not cancel as when all participants finish their contribution correctly. We show that our approach is especially resilient to dropouts when the injection of private values is done incrementally among the noise, therefore keeping correlated DP noise smaller in size. The analysis is done both in theory and in practice. First, we show theoretically that DP noise that is not supposed to cancel is bounded. We compare IncA with alternative decentralized [20, 61, 66] and centrally coordinated techniques [8] under the same threat model. We empirically show that our protocol exhibits a similar cost to other decentralized techniques while requiring correlated noise of smaller variance. Additionally, we measure the cost and accuracy gap between IncA and centrally coordinated ones.

1.2 Structure of the Paper

The rest of the paper is organized as follows. We first present preliminaries (Section 2). Then we present our protocol (Section 3) and its privacy analysis (Section 4). We further present our empirical evaluation (Section 5), related work (Section 6), and conclude the paper (Section 7).

2 Preliminaries

We denote the set of integers between a and b with $[a, b] \coloneqq \{a, a + 1 \dots b\}$ and the set of first k positive integers by $[k] \coloneqq [1, k]$. We define $\mathbf{1} \coloneqq (1, \dots, 1)^\top$ and b_i is a vector of all 0s except for the ith coordinate which is equal to 1. The dimensions of $\mathbf{1}$ and b_i can be inferred from the context. Unless explicitly stated differently, all vectors are column vectors. $\mathbb{I}[\cdot]$ is the indicator function, i.e., $\mathbb{I}[True] = 1$ and $\mathbb{I}[False] = 0$. For an $n \times m$ matrix M and subsets of indices $J \subseteq [n]$ and $K \subseteq [m]$, $M_{J,K}$ is the matrix obtained from M by taking the subsets of rows and columns indicated by J and K respectively. This extends analogously to vectors and other kinds of tensors. When used as sub-indexes ':' means the set of all indices and $-i$ means the set of all indices minus i, e.g., $M_{:,K} = M_{[n],K}$ and $M_{-i,K} = M_{[n]\setminus\{i\},K}$. When we use ":" instead of a column index of a matrix, the result is a row vector (e.g., $M_{i,:}$ is a row vector).

2.1 Problem Statement

We consider a set of parties $P = [n]$. Each party $i \in P$ has a private value $\mathsf{x}_i \in \mathcal{X}$ where $\mathcal{X}$ is a convex set in a vector space $\mathbb{X}$. These parties want to collaboratively estimate $\bar{\mathsf{x}} = \frac{1}{n}\sum_{i\in P}\mathsf{x}_i$ while keeping each value x_i (differentially) private. While for simplicity of our explanation we average scalars, our approach can be easily extended to vectors (such as machine learning models or gradients).

2.2 Differential Privacy

We evaluate the privacy of our protocols using the differential privacy framework (DP) [29]. This framework allows one to quantify how distinguishable is the output of a randomized algorithm on two *neighboring* datasets. More formally, a dataset is a vector of elements of $\mathcal{X}$, we denote the space of all datasets by $\mathcal{X}^*$. Two datasets $\mathsf{x}^{(A)}, \mathsf{x}^{(B)} \in \mathcal{X}^*$ are neighboring if there are $\mathsf{x}^{(0)} \in \mathcal{X}^*$, permutation matrices $Q^{(A)}$ and $Q^{(B)}$, and $u^{(A)}, u^{(B)} \in \mathcal{X}$ such that $\mathsf{x}^{(A)} = Q^{(A)}\begin{bmatrix} u^{(A)} \\ \mathsf{x}^{(0)} \end{bmatrix}$ and $\mathsf{x}^{(B)} = Q^{(B)}\begin{bmatrix} u^{(B)} \\ \mathsf{x}^{(0)} \end{bmatrix}$.

Definition 2.1. Let $\varepsilon > 0$ and $\delta \in [0, 1]$. A randomized algorithm $\mathcal{A} : \mathcal{X}^* \to \mathcal{Y}$ is (ε, δ)-*DP* if for any pair of neighboring datasets D, D' and any subset $Y \subseteq \mathcal{Y}$ of possible outputs we have that

$$\Pr(\mathcal{A}(D) \in Y) \leq \exp(\varepsilon)\Pr(\mathcal{A}(D') \in Y) + \delta \tag{1}$$

where the probability is taken over the randomness of $\mathcal{A}$.

In our problem, we set $\mathcal{X} = \{x \in \mathbb{R} : 0 \leq x \leq 1\}$ for simplicity of explanation. A dataset is a vector $\mathsf{x} = (\mathsf{x}_i)_{i=1}^n \in \mathcal{X}^n$. The output of $\mathcal{A}$ is the set of all observations an adversary makes, which may exceed the intended input and could include intercepted messages or information obtained from corrupted parties. This output therefore depends on the threat model considered. Consider a particular party $i \in P$. One can consider two extreme cases, which we briefly describe below and provide details in Appendix D.1. First, if the threat model allows for all parties $P \setminus \{i\}$ to be corrupted, then the best i can do is to privatize x_i before starting anything, and the best utility which can be obtained for a given privacy level (ε, δ) is the utility of Local DP [24]. Second, if the threat model makes more favorable assumptions, the best possible utility which can be reached for a given privacy level is the utility of Central DP, where there exist a trusted entity to whom all parties can send their messages and the adversary cannot see them until this entity releases his output.

Gaussian Mechanism. Our protocols use Gaussian noise to achieve differential privacy. Using the Gaussian Mechanism[30], it is possible to compute (ε, δ)-DP estimations of averages with a mean squared error of $2\ln(1.25/\delta)/\varepsilon^2 n^2$ in the central model. In the local model this error is n times bigger.

2.3 Threat Model

We consider two possible threat models: Collusion DP and Eavesdrop DP. Both of them correspond to a passive adversary: regardless of whether parties are corrupted or not, all of them follow the protocol as prescribed. However, they might involuntarily crash or disconnect (we discuss failures in Section 2.4).

Collusion DP has been widely adopted in the literature and is the main focus of our work. It considers that the adversary corrupts a subset of parties $C \subset P$. If a party $c \in C$ is corrupted, the adversary can learn both its private value x_i and its incoming/outgoing messages. We denote $P^H = P \setminus C$ the set of $n_H = n - |C|$ parties not controlled by the adversary. We assume that parties communicate via secure channels, and therefore the content of messages between honest parties is not known by the adversary if both sender and receiver are honest parties. However, we assume that the sender and receiver of each message are known by the adversary. This allows us to analyze the worst-case situations of network traffic tracking.

More formally, let $\mathcal{T}(\mathcal{A}, D) = (y_k)_{k=1}^M$ be the transcript of all M messages exchanged by a stochastic decentralized algorithm $\mathcal{A}$ with input dataset D, where $y_k = ((i, j), v) \in (P \times P) \times \{0, 1\}^*$ means that party i sent value v to j.

Definition 2.2 (Collusion DP). We say that a decentralized algorithm $\mathcal{A}$ is (ε, δ, C)-Collusion DP (or (ε, δ, C)-C-DP) if it is (ε, δ)-DP for the output $\mathcal{A}(D^H) = (\mathcal{W}, \mathcal{V}_{val})$, where the input dataset D^H is the set of private values of honest parties $(\mathsf{x}_i)_{i\in P^H}$, $\mathcal{W} = (p)_{(p,v)\in\mathcal{T}(\mathcal{A},D)}$ is the tuple of pairs sender-receiver of all user interactions, and $\mathcal{V}_{val} = \{((i, j), v) \in \mathcal{T}(\mathcal{A}, D) : i \in C \text{ or } j \in C\}$.

The view of the adversary in C-DP is similar to other correlated noise techniques such as those presented in [4, 61] and secure aggregation [8], where the set of interactions $\mathcal{W}$ is known. It is stronger than the view of the adversary in Network DP [18] which only knows $\mathcal{V}_{val}$ and ignores $\mathcal{W}$. We remark that the knowledge of $\mathcal{W}$ is a dangerous piece of information as it can be exploited to completely compromise privacy [55]. Our adversary is even stronger than Pairwise Network DP [20] where $\mathcal{W}$ is also unknown and the reported privacy loss of a party is the average over all possible placements of the adversary, whereas it corresponds to the worst-case in C-DP. Unlike our model, some pairwise noise and secure aggregation approaches [7, 8, 61] consider adversaries that can deviate from the protocol in some ways. We discuss the dangers of active attacks to our protocol in Appendix E.

We consider Eavesdrop DP as an alternative model where the adversary does not corrupt parties but compromises the security of some channels and observes a subset of messages $\mathcal{V}_{val} \subseteq \mathcal{T}(\mathcal{A}, D)$. This might come from intercepting messages, exploiting side channels, or other attacks. As in Collusion DP, it also knows the sender and receiver of all interactions $\mathcal{W}$.

Definition 2.3 (Eavesdrop DP). We say that a decentralized algorithm $\mathcal{A}$ satisfies $(\varepsilon, \delta, \mathcal{V}_{val})$-Eavesdrop DP (or $(\varepsilon, \delta, \mathcal{V}_{val})$-E-DP) if it satisfies (ε, δ)-DP for the output $\mathcal{A}(D) = (\mathcal{W}, \mathcal{V}_{val})$.

Note that $(\varepsilon, \delta, \mathcal{T}(D))$-E-DP (i.e., where all messages are known by the adversary) is equivalent to Local DP. We remark that parties executing algorithm $\mathcal{A}$ do not have knowledge of the observed messages $\mathcal{V}_{val}$ or of the corrupted parties C.

2.4 Communication Model

Synchronization and Communication Structure. We study decentralized protocols in the synchronous setting as defined in [36]. In this setting, parties perform exchanges in T iterations and there exists a known finite time bound for a message to reach its destination.

At each iteration $t \in [T]$, all parties wake up and interact, among others sending a message to a set of neighbors. We first model this communication structure. For every iteration $t \in [T]$, let $E_t \subseteq P \times P$ such that $(i, j) \in E_t$ if and only if party i sends a message to j in iteration t. If $E_t = E_1$ for all $t \in [T]$, we will say that our protocol has *static* exchanges, otherwise these are *dynamic*. For $i \in P$ and $t \in [T]$, let $N_{i\leftarrow}^{(t)} = \{j \mid (j, i) \in E_t\}$ and $N_{i\rightarrow}^{(t)} = \{j \mid (i, j) \in E_t\}$ to be respectively the sets of incoming and outgoing neighbors of i.

Failures. We consider that, at each iteration, parties could be unexpectedly absent of the computation due to a temporary or permanent crash, disconnection from the network, or other problems. We call this behavior a *dropout*. We denote by $O^{(t)} \in P$ to the set of parties that did not drop out in the computation at iteration $t \in [T]$. We consider that a party $i \in P$ dropped out *permanently* if he is not present in the last iteration (i.e., $i \notin O^{(T)}$). We assume that messages have bounded delays [36] and that there is sufficient time in an iteration that parties can confirm they received a message. Therefore, if a party i send a message to j in an iteration where j dropped out, i will detect it.

Decentralization. We assume that no party is exempt from the risk of failure. Therefore, it is not possible to designate a party that consistently coordinates the protocol as is done in Federated Learning and other centrally coordinated techniques. In Section 6, we discuss the additional challenges of the decentralized setting in comparison with centrally coordinated protocols.

3 Protocol

In this section, we present our protocol. The base protocol is described in Section 3.1 and the dropout resistant version is presented in Section 3.2.

3.1 Base Protocol

We introduce IncA, our protocol for incremental averaging in Algorithm 1. It consist on three phases: Initialization, Mixing and Dissemination.

Protocol. The Initialization Phase is devoted to sample DP noise and compute the initial messages of each party $i \in P$. In particular, i draws a Gaussian noise sample $\eta_i^\star \sim \mathcal{N}(0, \sigma_\star^2)$ and a $(T+1)$-vector $\left(z_{i,t}\right)_{t=0}^{T}$ from a noise distribution $\mathcal{D}(\mathsf{x}_i + \eta_i^\star)$ (line 4). For all $u \in \mathbb{X}$, $\mathcal{D}(u)$ satisfies the following property:

$$\sum_{t=0}^{T} v_t = u, \quad \text{for all } (v_t)_{t=0}^{T} \sim \mathcal{D}(u). \tag{2}$$

The goal of the $T+1$ terms $\left(z_{i,t}\right)_{t=0}^{T}$ is to inject the value $\mathsf{x}_i + \eta_i^\star$ into the system incrementally in a randomized way, such that (a) it is hard for an adversary to infer this value and (b) if party i would drop out and would not inject the rest of the value into the system the resulting error remains bounded. The goal of $\eta_i^\star$ is to prevent leakages from the exact average, by outputing only a differentially private approximation of it. The first message $y_i^{(0)}$ of each party $i \in P$ is set to $z_{i,0}$ (line 5).

Exchanges to ensure privacy are done in the Mixing Phase. At each iteration $t \in [T]$, each party $i \in P$ starts by sending its message $y_i^{(t-1)}$ to outgoing neighbors $N_{i\rightarrow}^{(t)}$. In the spirit of gossip algorithms, one can view $y_i^{(t-1)}$ as i's current estimate of the final output with the information it has so far. Then, it computes its new message (updated estimate of the final output) $y_i^{(t)}$ by aggregating incoming messages, his current message and a new part of its private value to inject $z_{i,t}$. For each incoming neighbor $j \in N_{i\leftarrow}^{(t)}$, message $y_j^{(t-1)}$ is aggregated with weight $W_{t;i,j}$ and $y_i^{(t)}$ with $W_{t;i,i}$ (line 10). At the end of the Mixing Phase, each party $i \in P$ will have an estimate $y_i^{(T)}$. By the time iteration T is reached, the final messages $(y_i^{(T)})_{i\in P}$ of this phase sufficiently hide the private values due to the gossip mixing and the noise terms $(\eta_i^\star)_{i\in P}$ and $(z_{i,t})_{i,t\in P\times[T]}$ (see Section 4 for details).

In the Dissemination Phase, since $(y_i^{(T)})_{i\in P}$ is differentially private, the parties can compute $\frac{1}{n}\sum_{i\in P} y_i^{(T)}$ in the clear. There are many practical ways to perform this computation. One of them is to use a gossip averaging protocol [13], this preserves a similar structure as the previous phases of our protocols.

Utility. If for all $t \in [T]$ we have that W_t satisfies

$$\forall j \in P, \quad \sum_{i\in P} W_{t;i,j} = 1 \quad \text{(column stochasticity)} \tag{3}$$

then

$$\begin{aligned}
\frac{1}{n}\sum_{i\in P} y_i^{(T)} &= \frac{1}{n}\sum_{i\in P} z_{i,T} + \sum_{j\in P} W_{T;i,j}\, y_j^{(T-1)} \\
&= \frac{1}{n}\sum_{i\in P} z_{i,T} + \frac{1}{n}\sum_{j\in P}\left(\sum_{i\in P} W_{T;i,j}\right) y_j^{(T-1)} \\
\text{(by Eq. (3))} &= \frac{1}{n}\sum_{i\in P} z_{i,T} + \frac{1}{n}\sum_{j\in P} y_j^{(T-1)}
\end{aligned} \tag{4}$$

By applying Equation (4) recursively on its second term, we have that

$$\begin{aligned}
\frac{1}{n}\sum_{i\in P} y_i^{(T)} &= \frac{1}{n}\sum_{i\in P}\sum_{t=0}^{T} z_{i,t} \\
\text{(by Eq. (2))} &= \frac{1}{n}\sum_{i\in P} \mathsf{x}_i + \eta_i^\star.
\end{aligned} \tag{5}$$

Equation (5) shows that the only noise that remains in the final estimate is $\frac{1}{n}\sum_{i\in P}\eta_i^\star$, whose variance is $\sigma_\star^2/n$. This shows that, if matrices $(W_t)_{t\in[T]}$ are column stochastic, then as long as $\mathcal{D}$ satisfies Equation (2) the injected noise of this distribution will not affect accuracy.

Algorithm 1 IncA Protocol

Input: $T \in \mathbb{N}$, $\mathsf{x} = (x_1, \ldots, x_n)^\top \in \mathcal{X}^n$, $W_1, \ldots, W_T \in \mathbb{R}^{n \times n}$, $\sigma_\star^2 \in \mathbb{R}$, $\mathcal{D}(\cdot) : \mathbb{X} \to \mathcal{P}\left(\mathbb{X}^{T+1}\right)$
Initialization Phase
for all $i \in P$ **do**
  Sample $\eta_i^\star \sim \mathcal{N}(0, \sigma_\star^2)$ and $(z_{i,0,\ldots} z_{i,T}) \sim \mathcal{D}(x_i + \eta_i^\star)$
  $y_i^{(0)} \leftarrow z_{i,0}$
end for
Mixing Phase
for $t \in \{1 \ldots T\}$ **do**
  for all $i \in P$ **do**
    $y_i^{(t)} \leftarrow \left(\sum_{j \in P} W_{t;i,j} y_j^{(t-1)}\right) + z_{i,t}$
  end for
end for
Dissemination Phase: Parties jointly compute $\frac{1}{n} \sum_{i \in P} y_i^{(T)}$

Distribution $\mathcal{D}$. There are several ways to model the distribution $\mathcal{D}(\cdot)$. We first show two examples to illustrate the core principle of the distribution.

Example 3.1 (Early Injection: $\mathcal{D}_{\text{EI}}$). For some variance $\sigma_\Delta^2 > 0$ and $v \in \mathbb{X}$, $(v_t)_{t=1}^T \sim \mathcal{D}_{\text{EI}}(v)$ is defined as follows:

(1) draw T i.i.d. samples $\eta_1, \ldots, \eta_T$ with distribution $\mathcal{N}(0, \sigma_\Delta^2)$
(2) set $v_0 = v + \sum_{t=1}^T \eta_t$ and $v_t = -\eta_t$ for each $t \in [T]$

Let us instantiate $\mathcal{D}$ of Algorithm 1 by $\mathcal{D}_{\text{EI}}$. First, each party $i \in P$ will draw T i.i.d noise terms $\eta_{i,0}, \ldots, \eta_{i,T}$ from $\mathcal{N}(0, \sigma_\Delta^2)$. Then we have that

$$y_i^{(0)} = \mathsf{x}_i + \eta_i^\star + \sum_{t=1}^{T} \eta_{i,T} \quad \text{and that}$$

$$y_i^{(t)} = \left(\sum_{j \in P} W_{t;i,j} y_j^{(t-1)}\right) - \eta_{i,t}$$

for all $t \in [T]$. In this example, parties inject all the private value and noise at the beginning of the execution to $y_i^{(0)}$, and gradually remove each noise term in subsequent messages while mixing them.

Note that, in addition to the independent noise $\eta_i^\star$, each party i injects T noise terms. One might naively think that privacy could be obtained by just adding a single extra noise term and progressively removing it instead of adding many terms as done in the example. However, such approach is likely to compromise privacy. With the knowledge of interactions and weights matrices $(W_1, \ldots, W_T)$, the adversary can easily construct a system of linear equations where private values and noise terms are unknowns and each observed message in $\mathcal{V}_{val}$ can be used to construct one equation of the system (see Section 4.2). This attack strategy has shown to be dangerous in [55]. Therefore, it is crucial that each party has a sufficient number of noise terms to hide private values from adversarial observations.

Example 3.2 (Incremental Injection: $\mathcal{D}_{\text{Inc}}$). For some variance $\sigma_\Delta^2 > 0$ and $v \in \mathbb{X}$, $(v_t)_{t=1}^T \sim \mathcal{D}_{\text{Inc}}(v)$ is defined as follows:

(1) draw $\eta_1, \ldots, \eta_T$ as in Example 3.1
(2) set $v_0 = v/(T+1) + \eta_1$, $v_t = v/(T+1) - \eta_t + \eta_{t+1}$ for all $t \in [T-1]$ and $v_T = v/(T+1) - \eta_T$

In Example 3.2, v is spread over all vectors $(v_t)_{t=1}^T$ and one noise term is added and canceled at each iteration. When $\mathcal{D}_{\text{Inc}}$ is used in Algorithm 1, each party $i \in P$ draws i.i.d. noise terms $\eta_{i,0}, \ldots, \eta_{i,T}$ as in the previous example. Then we have that

$$y_i^{(0)} = \mathsf{x}_i/(T+1) + \eta_{i,1},$$

$$y_i^{(t)} = \left(\sum_{j \in P} W_{t;i,j} y_j^{(t-1)}\right) - \eta_{i,t} + \eta_{i,t+1} \quad \forall t \in [T-1] \text{ and}$$

$$y_i^{(T)} = \left(\sum_{j \in P} W_{t;i,j} y_j^{(t-1)}\right) - \eta_{i,t}.$$

Party i injects $1/(T+1)$-th of x_i per iteration, renewing the noise each time. At iteration T no new noise term is injected. The sensitivity of each noisy term $z_{i,t}$ to changes in x_i is in the worst case smaller than in Example 3.1 where the private values are injected all at once in the first iteration. This reduces the variance σ_Δ^2 required to satisfy differential privacy. As shown in Equation (5), noise added by $\mathcal{D}$ cancels and its variance σ_Δ^2 does not impact accuracy in the absence of failures. However, parties can disconnect during the computation. Therefore, it is important to remain σ_Δ^2 as low as possible.

Multivariate Gaussians. Examples 3.1 and 3.2 are part of a more general types of distributions for which we prove differential privacy guarantees. We define it below.

Definition 3.3 (Multivariate Gaussian). We call a distribution $\mathcal{D}$ is a (c, Z)-*Gaussian* if for some $c \in \mathbb{R}^{T+1}$ and $Z \in \mathbb{R}^{(T+1) \times T}$ there holds $\mathcal{D}(u) = \mathcal{D}_{Gauss}^{(c,Z)}(u)$, where sampling $(v_0, \ldots, v_T) \sim \mathcal{D}_{Gauss}^{(c,Z)}(u)$ is achieved by

(1) Sampling $\eta_k \sim \mathcal{N}(0, \sigma_k^2)$ for each $k \in [1, T]$
(2) Setting $v_t = c_t u + \sum_{k=1}^T Z_{t,k} \eta_k$ for each $t \in [0, T]$.

One can see that $\mathcal{D}_{Gauss}^{(c,Z)}(v)$ is the multivariate Gaussian distribution $\mathcal{N}(cv, Z\Sigma_g Z^\top)$ where $\Sigma_g = \text{diag}(\sigma_1^2, \ldots, \sigma_T^2)$. To prove our privacy guarantees, we will need *Valid* (c, Z)-Gaussians:

Definition 3.4 (Valid (c, Z)-Gaussian). A (c, Z)-*Gaussian* $\mathcal{D}$ is *Valid* if (i) it satisfies the property of Equation (2), i.e., $\sum_{t=0}^T v_t = u$ independently of the η_k, (ii) the matrix $(c, Z) \in \mathbb{R}^{(T+1) \times (T+1)}$ is invertible and (iii) $Z_{-T,:} \in \mathbb{R}^{T \times T}$ is invertible.

Condition (i) above implies that $\sum_{t=0}^T c_t = 1$ and $\forall k : \sum_{t=0}^T Z_{t,k} = 0$. Conditions (ii) and (iii) prevent individual noise terms from being guessed if the adversary does not observe all the messages of a party.

Lemma 3.5. *$\mathcal{D}_{\text{EI}}$ and $\mathcal{D}_{\text{Inc}}$ are Valid Gaussians.*

We prove Lemma 3.5 it in Appendix A.1. In practice we'll adopt $\mathcal{D}_{\text{Inc}}$ due to its good properties against dropouts.

3.2 Protocol for Dropouts

We now present our protocol in the presence of temporary and permanent dropouts. For all $t \in [1, T]$, we denote by $O^{(t)}$ the set of parties that did not dropout at iteration t. Our protocol is described in Algorithm 2. We consider that all parties in P have at least started the protocol and computed its first message (i.e., $O^{(0)} = P$), otherwise they are not considered as part of the protocol.

The Initialization Phase only requires minor changes with respect to Algorithm 1 on the sampling of correlated noise. Since a party $i \in P$ might dropout, not all samples $(z_{t,i})_{t\in[0,T]}$ are generated in advance, as it is more convenient to do it per iteration, depending on the online history i. Only the first term $z_{i,0}$ is sampled at Initialization Phase.

At each iteration $t \in [1,T]$ of the Mixing Phase, only active parties in $O^{(t)}$ exchange messages. Each party $i \in O^{(t)}$ receives its incoming messages of online parties, sends its outgoing messages and detects that its outgoing messages are not received by $j \in N_{i\to}^{(t)} \setminus O^{(t)}$ due to dropout. Matrix W_t^O has the weights of real exchanges, correcting the attempted exchanges W_t according to $O^{(t)}$. The weights of incoming neighbors that did not drop out remain unchanged (line 11), otherwise they are set to 0 (line 12). To keep the column stochasticity of W_t^O, the weight $W_{t;i,i}^O$ of its own message $y_i^{(t-1)}$ is increased with the weights of value that i could not send due to dropout (line 13).

Next, each party i samples a new term $z_{i,t}$ from the distribution $\mathcal{D}^O(i,t)$ (line 14). Each party adapts $\mathcal{D}$ to its online history. At the end of the Mixing phase, noise terms $(z_{i,t})_{t\in[0,T]}$ follow distribution $\mathcal{D}^O(i,:)$, which we will explain later.

After that, party i aggregates incoming messages and noise terms to compute $y_i^{(t)}$ as in the original protocol (line 15). If party i dropped out, it will not be able to perform any exchange or sample any noise term. Therefore it will consider the last message he computed as his current message (line 17). This concludes the Mixing Phase.

In the Dissemination phase, parties average the values $(y_i^{(T)})_{i\in O^{(T)}}$. When a party i drops out during an iteration, it only partially injects its value x_i. Moreover, if i permanently drops out at some iteration $t \in [1,T]$, a part of the private values of (a subset of) all parties is lost due to the mixing nature of our protocol, along with the message $y_i^{(t)}$, which is never sent. Therefore, when computing the average of $(y_i^{(T)})_{i\in O^{(T)}}$, dividing $\sum_{i\in O^{(T)}} y_i^{(T)}$ by n is less accurate than dividing by the total weight of private values injected.

We adopt the latter approach, which is feasible by adding some extra information to the messages. Even if $(\mathrm{x}_i)_{i\in P}$ are secret, parties can still track the total weight of private values in each message if they share their own weight alongside their messages and update it when they inject part of their private values or aggregate incoming messages. This does not constitute a privacy breach, as such weights are already known to the adversary (as encoded in Equation (21) and explained in Section 4.1) and only depend on the status of parties being online of offline rather than the private values themselves.

We assume that $(y_i^{(T)})_{i\in O^{(T)}}$ is visible to the adversary. For resilience to dropouts, instead of using the method in [13], one can employ other non-private epidemic dissemination protocols [22, 60] that have negligible probability of information loss due to dropout. In the unlikely case that a message from party i is lost in the Dissemination phase due to a dropout, we consider $i \notin O^{(T)}$.

Distribution $\mathcal{D}^O$. Now we describe how parties use multivariate Gaussians when dropouts occur. Let $\mathcal{D} = \mathcal{D}_{Gauss}^{(c,Z)}(v)$ and $O = (O^{(i)})_{i\in P}$ be the dropout history. Then, for each party $i \in P$, $\mathcal{D}^O(i,:) = \mathcal{D}_{Gauss}^{(c^{(i)},Z^{(i)})}(\mathrm{x}_i + \eta_i^\star)$ where

$$
\begin{aligned}
&c_t^{(i)} = 0, && Z_{t,:}^{(i)} = 0 && \text{if } i \notin O^{(t)} \text{ and } t \in [0,T]\\
&c_t^{(i)} = c_{t'}, && Z_{t',:}^{(i)} = Z_{t,:} && \text{if } i \in O^{(t)} \text{ and } t \in [0,T-1]\\
&c_T^{(i)} = c_{T'}, && Z_{T,:}^{(i)} = -\textstyle\sum_{t'=0}^{T'-1} Z_{t',:} && \text{if } i \in O^{(T)}
\end{aligned}
$$

and $t' = \sum_{k=0}^{t} \mathbb{I}\left[i \in O^{(k)}\right]$ for all $t \in [0,T]$. Essentially, parties do not inject anything into the system if they dropped out and, if they are online in the last iteration, they cancel all previously injected noise terms.

Algorithm 2 IncA for dropouts

1: **Input:** $T \in \mathbb{N}$, $\mathrm{x} = (x_1,\dots,x_n)^\top \in \mathcal{X}^n$, $W_1,\dots,W_T \in \mathbb{R}^{n\times n}$, $\sigma_\star^2 \in \mathbb{R}$, $\mathcal{D} : \mathbb{X} \to \mathcal{P}\left(\mathbb{X}^{T+1}\right)$
2: *Initialization Phase*
3: **for all** $i \in P$ **do**
4: Sample $\eta_i^\star \sim \mathcal{N}(0,\sigma_\star^2)$ and $z_{i,0} \sim \mathcal{D}^O(i,0)$
5: $y_i^{(0)} \leftarrow z_{i,0}$
6: **end for**
7: *Mixing Phase*
8: **for** $t \in \{1\dots T\}$ **do**
9: **for all** $i \in P$ **do**
10: **if** $i \in O^{(t)}$ **then**
11: Set $W_{t;i,j}^O \leftarrow W_{t;i,j}$ for all $j \in O^{(t)}$
12: Set $W_{t;i,j}^O \leftarrow 0$ for all $j \in P \setminus O^{(t)} \setminus \{i\}$
13: Set $W_{t;i,i}^O \leftarrow W_{t;i,i} + \sum_{j\in P\setminus O^{(t)}} W_{t;j,i}$
14: Sample $z_{i,t} \sim \mathcal{D}^O(i,t)$
15: $y_i^{(t)} \leftarrow \left(\sum_{j\in P} W_{t;i,j}^O y_j^{(t-1)}\right) + z_{i,t}$
16: **else**
17: $y_i^{(t)} \leftarrow y_i^{(t-1)}$
18: **end if**
19: **end for**
20: **end for**
21: *Dissemination Phase*: Parties jointly average $(y_i^{(T)})_{i\in O^{(T)}}$.

Replacing $z_{i,t}$ using the definition of $\mathcal{D}^O$, we have the following equations for the messages:

$$y_i^{(0)} = c_0^{(i)}(\mathrm{x}_i + \eta_i^\star) + \sum_{k=1}^{T} Z_{0,k}^{(i)} \eta_{i,k} \tag{6}$$

$$y_i^{(t)} = \sum_{j=1}^{n} W_{t;i,j}^O y_j^{(t)} + c_t^{(i)}(\mathrm{x}_i + \eta_i^\star) + \sum_{k=1}^{T} Z_{t,k}^{(i)} \eta_{i,k} \tag{7}$$
$$\text{for all } t \in [1,T].$$

These equations determine the view of the adversary.

For each $i \in O^{(T)}$, we have that

$$\sum_{t=0}^{T} z_{i,t} = w_i(\mathrm{x}_i + \eta_i^\star) \tag{8}$$

where $w \in \mathbb{R}^n$ is such that

$$w_i = \sum_{t=0}^{T} c_t^{(i)}. \tag{9}$$

If party i did not permanently drop out, she/he will be able to inject a proportion w_i of its private value x_i plus independent noise $\eta_i^\star$.

Utility. If all parties have the same probability of dropout, our estimate will be unbiased. If there are only temporary drop outs, i.e. $O^{(T)} = P$, all correlated noise is canceled. By following a similar analysis as the one in Equation (5), we have that

$$\sum_{i=1}^{n} y_i^{(T)} = \sum_{i=1}^{n} w_i(x_i + \eta_i^\star).$$

If permanent dropouts occur, two main disruptions happen. First, parts of the private values with non-zero coefficients in the dropped messages are lost. Second, correlated noise is not completely canceled. Therefore, the properties in equations (3) and (8) do not hold. We empirically show the final accuracy in Section 5.2.

4 Privacy Analysis

We now present our privacy results. In Section 4.1 we show how the adversary's knowledge can be structured as a set of linear equations. Next, in Section 4.2 we present an abstract result that accurately accounts for the privacy loss of IncA. After that, we present in Section 4.3 interpretable results on the conditions to obtain differential privacy with accuracy comparable to that of Central DP when no dropouts occur. Finally, in Section 4.4, we present sufficient positive and negative conditions on the graph that models parties interactions to obtain privacy.

4.1 Knowledge of the Adversary

We prove Eavesdrop DP and Collusion DP guarantees. Hence, we assume that the adversary knows the set $\mathcal{W} = \{W_1, \ldots, W_T\}$ and online activity $O = (O^{(t)})_{t\in[1,T]}$. For the subset of values of messages it knows, we will slightly abuse notation with respect to the definition of $\mathcal{V}_{val}$ in E-DP and C-DP so that $\mathcal{V}_{val} \subseteq P \times [0, T]$ denotes the subset of observed iteration-party messages. Hence the messages $\{((i,t), y_i^{(t)})\}_{(i,t)\in\mathcal{V}_{val}}$ will be known.

For E-DP, we assume that the adversary can observe the final messages of the Mixing Phase of all parties, i.e.,

$$\mathcal{V}_{val} \supseteq \{(i, T)\}_{i\in P}.$$

For C-DP, they will additionally see all messages seen by corrupted parties. That is,

$$\mathcal{V}_{val} = \{(i,t) \in P\times[0,T-1] : (\{i\}\cup N_{i\rightarrow}^{(t+1)})\cap C \neq \emptyset\}\cup\{(i,T)\}_{i\in P}.$$

Knowledge as Linear Equations. The knowledge of the adversary can be structured as a set of linear equations where the unknowns are private values x and noise terms $(\eta^\star, \eta)$. We show below how these equations are constructed.

Let $k_H := n_H T$ be the number of canceling noise terms unknown by the adversary before they make any observations. We denote by $x^H := (x_i)_{i\in P^H} \in \mathcal{X}^{n_H}$, $\eta^{\star H} := (\eta_i^\star)_{i\in P^H} \in \mathbb{X}_H^n$ and $\eta_{(:)}^H := (\eta_{i,:})_{i\in P^H} \in \mathbb{X}^{k_H}$ the vectors of private values and noise terms of honest parties. Then, from an execution $\mathcal{E} = (\mathcal{W}, O, \mathcal{V}_{val})$ of Algorithm 2, the adversary can construct a set of linear equations described by

$$B(x^H + \eta^{\star H}) + A\eta_{(:)}^H = y_{\mathcal{V}} \tag{10}$$

where for $m := |\mathcal{V}_{val}| \leq n_H T$, $B \in \mathbb{R}^{m\times n_H}$ and $A \in \mathbb{R}^{m\times k_H}$ are matrices with public coefficients such that (B, A) is a full rank, $y_{\mathcal{V}} \in \mathbb{X}^m$ is the set of messages observed by the adversary after removing known constants and redundant information (e.g., linearly dependent equations), and $(x^H, \eta^{\star H}, \eta_{(:)}^H)$ are the unknowns. In Appendix A.2 we show the details on how to construct the system of Equation (10).

Discussion. Let's inspect the system of Equation (10). For each $i \in P^H$, variables x_i and $\eta_i^\star$ only appear coupled in the term $x_i + \eta_i^\star$. Therefore, the adversary will never be able to isolate them and we can consider them as a single variable $\tilde{x}_i = x_i + \eta_i^\star$. In the worst case, the adversary observes all the messages of honest parties, meaning that $m = n_H + k_H$ and that (B, A) is a square invertible matrix. Then, the adversary is able to recover $\tilde{x}_i$ for all $i \in P^H$ by computing $(B, A)^{-1} y_{\mathcal{V}}$. In that case, the only way to protect private values is to set $\eta_i^\star$ large enough, i.e. using the same noise required for Local DP. Hence, we emphasize the importance of setting $n_H + k_H > m$ to prevent this situation, which justifies the large amount of noise terms (one per iteration) each party generates.

4.2 Abstract Result

We first present an abstract privacy result that provides a relationship between the variance of added noise terms and (ε, δ)-DP guarantees for a given execution of Algorithm 2. We define $\Sigma_\eta \in \mathbb{X}^{(n_H+k_H)\times(n_H+k_H)}$ to be the covariance matrix of $\begin{pmatrix} \eta^{\star H} \\ \eta_{(:)}^H \end{pmatrix}$. Note that Σ_η is diagonal as all noise terms are independent.

Theorem 4.1. *Let $\mathcal{E} = (\mathcal{W}, O, \mathcal{V}_{val})$ be an execution of Algorithm 2 where $\mathcal{D}$ is a Valid (c, Z)-Gaussian and $\mathcal{V}_{val}$ be defined by the observations of the adversary (respectively by the corruption of a set C of parties). Let B, A be derived from $\mathcal{E}$ as described in Equation* (10). *Let Σ_η be the covariance matrix of noise terms $\begin{pmatrix} \eta^{\star H} \\ \eta_{(:)}^H \end{pmatrix}$, which have positive variance. Then, execution $\mathcal{E}$ is $(\varepsilon, \delta, \mathcal{V}_{val})$-E-DP (respectively (ε, δ, C)-C-DP) if all noise terms have positive variance and*

$$h^\top \Sigma^{-1} h \leq \frac{\varepsilon^2}{c^2} \qquad \text{for each column } h \text{ of } B \tag{11}$$

where $c^2 > 2\ln(1.25/\delta)$ and

$$\Sigma := (B, A)\Sigma_\eta (B, A)^\top \in \mathbb{R}^{m\times m}. \tag{12}$$

We prove this result in Appendix A.4. Note that the invertibility of Σ is not an additional constraint: given that (B, A) is full rank, Σ is always invertible if all noise terms have positive variance. Using Theorem 4.1, one can set variances and obtain ε and δ. We now show the inverse direction, i.e., how to obtain Σ_η given ε and δ using convex optimization.

A symmetric matrix M is positive semi-definite, denoted $M \succeq 0$, if $x^\top M x \geq 0$ for every non-zero vector x. M is positive definite, or $M \succ 0$, if $x^\top M x > 0$ for every non-zero vector x.

Corollary 4.2. *Let $\varepsilon, \delta \in (0, 1)$. Let B, A be associated with an execution of Algorithm 2 be derived from $(\mathcal{W}, O, \mathcal{V}_{val})$ (or from $(\mathcal{W}, O, C)$) as defined in Theorem 4.1. Let Σ also be defined as in*

Theorem 4.1. Then, the execution is (ε, δ, $\mathcal{V}_{val}$)-E-DP (respectively (ε, δ, C)-C-DP) if all noise terms have positive variance and

$$\begin{pmatrix} \Sigma & h \\ h^\top & \varepsilon^2/c^2 \end{pmatrix} \succeq 0 \tag{13}$$

for each column h of B, where $c^2 > 2\ln(1.25/\delta)$.

We prove it in Appendix A.4. It implies that for fixed (ε, δ) the possible values of Σ_η can be computed with a convex program.

4.3 Bounded Independent Noise $\eta^\star$

In this section, we show how to achieve such guarantees with bounded variance $\sigma_\star^2$ of the independent noise $\eta^\star$. This is important, as $\sigma_\star^2$ completely determines the accuracy of Algorithm 1 (see Equation 2 and significantly influences that of Algorithm 2, as $\eta^\star$ are the only terms that IncA does not attempt to cancel.

From an execution $\mathcal{E} = (\mathcal{W}, \mathcal{O}, \mathcal{V}_{val})$ of Algorithm 2, the view of the adversary is completely determined by the values of x^H, $\eta^{\star H}$ and $\eta^H_{(:)}$ as described by Equation (10). We denote this view by

$$\mathcal{V}_\mathcal{E}\left(\mathsf{x}^H + \eta^{\star H}, \eta^H\right) = y_\mathcal{V}.$$

We also sometimes slightly abuse notation by using the matrix $\eta^H = (\eta_{i,:})_{i\in P^H} \in \mathbb{X}^{n_H \times T}$ instead of vector $\eta^H_{(:)}$ (which contains the same elements in a different shape) as the last parameter of $\mathcal{V}_\mathcal{E}$.

We start by presenting Lemma 4.3, which relates the change required in noise terms $\eta^{\star H}$ and $\eta^H_{(:)}$ such that executing the protocol with neighboring datasets $\mathsf{x}^{(A)}$ or $\mathsf{x}^{(B)}$ remain indistinguishable to the adversary and the required variance of $\eta^{\star H}$ and $\eta^H_{(:)}$ to satisfy (ε, δ)-DP guarantees.

Lemma 4.3. *Let $\varepsilon, \delta \in (0,1)$. Let $\mathcal{E} = (\mathcal{W}, \mathcal{O}, \mathcal{V}_{val})$ be defined as in Theorem 4.1. For a pair of neighboring datasets $\mathsf{x}^{(A)}, \mathsf{x}^{(B)} \in \mathbb{X}^{n_H}$, let $\xi^\star \in \mathbb{X}^{n_H}$ and $\Delta_{(:)} \in \mathbb{X}^{k_H}$ be such that*

$$\mathcal{V}_\mathcal{E}\left(\mathsf{x}^{(A)} + \eta^{\star H}, \eta^H_{(:)}\right) = \mathcal{V}_\mathcal{E}\left(\mathsf{x}^{(B)} + \eta^{\star H} + \xi^\star, \eta^H_{(:)} + \Delta_{(:)}\right)$$

for any $\eta^{\star H}$, $\eta^H_{(:)}$. Recall that Σ_η is the covariance matrix of $\begin{pmatrix} \eta^{\star H} \\ \eta^H_{(:)} \end{pmatrix}$. Then, execution $\mathcal{E}$ is (ε, δ, $\mathcal{V}_{val}$)-E-DP (or (ε, δ, C)-C-DP) if for all such neighboring datasets $\mathsf{x}^{(A)}$ and $\mathsf{x}^{(B)}$

$$\left((\xi^\star)^\top, \Delta_{(:)}^\top\right) \Sigma_\eta^{-1} \begin{pmatrix} \xi^\star \\ \Delta_{(:)} \end{pmatrix} \le \frac{\varepsilon^2}{c^2}$$

where $c^2 > 2\ln(1.25/\delta)$.

We can think of $\xi^\star$ as a maximum difference between data points in the domain (and hence neighboring datasets), i.e., a measure closely related to sensitivity. In particular, for the sensitivity $\Delta(\text{IncA})$ there holds $\Delta(\text{IncA}) = \max_{\mathsf{x}^{(A)},\mathsf{x}^{(B)}} \|\text{IncA}(\mathsf{x}^{(A)}, \mathsf{x}^{(B)})\| = \max_{\xi^\star} \|\text{IncA}(\xi^\star)\|$.

The main techniques to prove Lemma 4.3 have been studied in [61, Theorem 1]. We adapt the proof to our setting in Appendix A.5. This lemma establishes the relation between the required (ε, δ)-DP parameters and magnitude of the change $\xi^\star$ in independent noise term $\eta^\star$ and the change $\Delta_{(:)}$ in canceling noise terms $\eta_{(:)}$ such that the view the of adversary remains indistinguishable when the input is changed from $\mathsf{x}^{(A)}$ to $\mathsf{x}^{(B)}$.

Correlated Noise Variance σ_Δ^2. In the rest of the paper, all noise terms of vector η^H have variance σ_Δ^2. Hence the condition to satisfy (ε, δ)-DP in Lemma 4.3 is equivalent to

$$\frac{\|\xi^\star\|_2^2}{\sigma_\star^2} + \frac{\|\Delta_{(:)}\|_2^2}{\sigma_\Delta^2} \le \frac{\varepsilon^2}{c^2} \tag{14}$$

where $c^2 > 2\ln(1.25/\delta)$.

From Equation (14), we can see that a bounded $\sigma_\star$ is achievable for sufficiently large $\sigma_\Delta > \|\Delta_{(:)}\|c/\varepsilon$ as long as $\xi^\star$ is bounded. We break down our problem, first analyzing in Lemma 4.4 how privacy is amplified by each message that the adversary does not see and in subsequent theorems the conditions on unseen messages to obtain a bounded $\xi^\star$ (and consequently $\sigma_\star$).

Lemma 4.4. *Let $\mathcal{E} = (\mathcal{W}, \mathcal{O}, \mathcal{V}_{val})$ be defined as in Theorem 4.1. Then if $y_i^{(t)}$ is not observed by the adversary (i.e., $(i,t) \in P^H \times [0, T-1] \setminus \mathcal{V}_{val}$) and $i \in \mathcal{O}^{(t)}$, then there exists $a^{(i,t)} \in \mathbb{X}^n$ and $G^{(i,t)} \in \mathbb{X}^{n\times T}$ such that for any $\beta \in \mathbb{R}$ we have*

$$\mathcal{V}_\mathcal{E}\left(\tilde{\mathsf{x}}^H, \eta^H\right) = \mathcal{V}_\mathcal{E}\left(\tilde{\mathsf{x}}^H + \beta a^{(i,t)}, \eta^H - \beta G^{(i,t)}\right).$$

Moreover, $a_i^{(i,t)} = \frac{W^{\mathcal{O}}_{t+1;i,i}{}^{-1}}{w_i}$ and $a_j^{(i,t)} = \frac{W^{\mathcal{O}}_{t+1;j,i}}{w_j}$ for all $j \in P^H \setminus \{i\}$, where $w \in \mathbb{R}^n$ is defined as in Equation (9).

Lemma 4.4 establishes that if $y_i^{(t)}$ is not part of the adversary's observations and was effectively transmitted to other parties, a vector in the nullspace of Equation (10), i.e., a vector which added to a solution gives another solution, is $(a^{(i,t)}, -G^{(i,t)})$. We prove this in Appendix A.6.

In the following theorems bounds on $\sigma_\star^2$ are the result of constructing a bounded $\xi^\star$, from the possible solutions $\mathsf{x} + \eta^\star$ of Equation (10). The more messages are unobserved the easier to find smaller values of $\xi^\star$ and $\Delta_{(:)}$ of Lemma (4.3).

First we show this when IncA is executed without dropouts.

Theorem 4.5. *Let $\varepsilon, \delta \in (0,1)$. Let $\mathcal{E} = (\mathcal{W}, \mathcal{O}, \mathcal{V}_{val})$ be defined as in Theorem 4.1 and associated with an execution without dropouts (i.e., $\mathcal{O}^{(t)} = P$ for all $t \in [0,T]$). Let $\mathcal{H} = P^H \times [0,T] \setminus \mathcal{V}_{val}$ be the set of pairs (i,t) such that $y_i^{(t)}$ is not seen by the adversary. For all $(i,t) \in \mathcal{H}$, let $a^{(i,t)}$ be as defined in Lemma 4.4. If $\{a^{(i,t)}\}_{(i,t)\in\mathcal{H}}$ has at least $n_H - 1$ independent vectors, there exist $\xi^\star$ and $\Delta_{(:)}$ as defined in Lemma 4.3 and $\mathcal{E}$ satisfies (ε, δ, $\mathcal{V}_{val}$)-E-DP (or (ε, δ, C)-C-DP) for any*

$$\sigma_\star^2 > \frac{c^2}{n_H \varepsilon^2} \qquad \text{and} \qquad \sigma_\Delta^2 \ge \frac{\|\Delta_{(:)}\|_2^2}{\frac{\varepsilon^2}{c^2} - \frac{1}{n_H \sigma_\star^2}},$$

where $c^2 > 2\ln(1.25/\delta)$.

We prove this theorem in Appendix A.7. By Equation (5), we have that Algorithm 1 produces an unbiased estimate of $\frac{1}{n}\sum_{i\in P} \mathsf{x}_i$ with variance $\sigma_\star^2/n$. Under the conditions of Theorem 4.5 this variance of Algorithm 1's output can be made arbitrarily close to

$$\frac{c^2}{n_H n \varepsilon^2}.$$

The above matches the error of Secure Aggregation plus DP noise [2, 16, 46] and Central DP when $n_H = n$. In the case where $n_H < n$, IncA order-wisely matches the $O(1/n^2)$ mean squared error of Central DP. Protocols that exactly obtain Central DP accuracy in

the presence of corrupted parties require a large communication cost [27, 44, 62]. In the remaining of Section 5.2 we show a bound on $\sigma_\star^2$ in the presence of dropouts.

Theorem 4.6. *Let $\varepsilon, \delta \in (0,1)$. Let $\mathcal{E} = (\mathcal{W}, \mathcal{O}, \mathcal{V}_{val})$ be defined as in Theorem 4.1. Let $w_{\mathcal{O}} = \sum_{i \in P^H} w_i$, where w is defined in Equation (9). Let $\mathcal{H} = P^H \times [0,T] \setminus \mathcal{V}_{val}$ be the set of pairs (i,t) such that $y_i^{(t)}$ is not seen by the adversary. For all $(i,t) \in \mathcal{H}$, let $a^{(i,t)}$ be as defined in Lemma 4.4. If $\{a^{(i,t)}\}_{(i,t)\in\mathcal{H}}$ has at least $|n_H| - 1$ independent vectors, then there exist $\xi^\star$ and $\Delta_{(:)}$ as defined in Lemma 4.3 such that $\mathcal{E}$ satisfies $(\varepsilon, \delta, \mathcal{V}_{val})$-E-DP (or (ε, δ, C)-C-DP) for any*

$$\sigma_\star^2 > \frac{(n_H - 1)c^2}{(w_{\mathcal{O}} - 1)^2 \varepsilon^2} \qquad \text{and} \qquad \sigma_\Delta^2 \geq \frac{\|\Delta_{(:)}\|_2^2}{\frac{\varepsilon^2}{c^2} - \frac{(n_H - 1)}{(w_{\mathcal{O}} - 1)^2 \sigma_\star^2}},$$

where $c^2 > 2\ln(1.25/\delta)$.

Theorem 4.6 shows the needed increase of $\sigma_\star^2$ due to dropout. We prove it in Appendix A.7. As w_i is the proportion of x_i injected to the computation for each $i \in P^H$, $w_{\mathcal{O}} \in [n_H/T, n_H]$ is the total amount for all private values. It is proportional to the online time of all parties. As $w_{\mathcal{O}} \in O(n)$, then $\sigma_\star^2 \in O(c^2/n\varepsilon^2)$ which order-wisely matches $\sigma_\star^2$ in the setting without dropouts.

If a honest party drops-out permanently in an early iteration, it is possible that it was not able to send a sufficient number of messages in order for $\{a^{(i,t)}\}_{(i,t)\in\mathcal{H}}$ to have at least $|n_H| - 1$ independent vectors. This would make Theorem 4.6 inapplicable. However, as it has not completely canceled his correlated noise due to the permanent dropout, it will remain protected by the uncanceled correlated noise terms even if $\sigma_\star^2$ is bounded. We reflect that possibility in the following theorem.

Theorem 4.7. *Let $\varepsilon, \delta \in (0,1)$. Let $\mathcal{E} = (\mathcal{W}, \mathcal{O}, \mathcal{V}_{val})$ be defined as in Theorem 4.1. Let $J \subseteq P^H$ be a coalition that contain all honest parties that did not drop out permanently (i.e., $\mathcal{O}^{(T)} \cap P^H \subseteq J$). Let $w_{\mathcal{O}} = \sum_{i \in P^H} w_i$, where w is defined in Equation (9) and*

$$\mathcal{H} = \{(i,t) \in P^H \times [0,T] \setminus \mathcal{V}_{val} : i \in J \wedge N_{i\rightarrow}^{(t)} \subseteq J\}.$$

For all $(i,t) \in \mathcal{H}$, let $a^{(i,t)}$ be as defined in Lemma 4.4. If $\{a^{(i,t)}\}_{(i,t)\in\mathcal{H}}$ has at least $|J| - 1$ independent vectors, then there exist $\xi^\star$ and $\Delta_{(:)}$ as defined in Lemma 4.3 and $\mathcal{E}$ satisfies $(\varepsilon, \delta, \mathcal{V}_{val})$-E-DP (or (ε, δ, C)-C-DP) for any

$$\sigma_\star^2 > \frac{(|J| - 1)c^2}{(w_{\mathcal{O}} - 1)^2 \varepsilon^2} \qquad \text{and} \qquad \sigma_\Delta^2 \geq \frac{\|\Delta_{(:)}\|_2^2}{\frac{\varepsilon^2}{c^2} - \frac{(|J| - 1)}{(w_{\mathcal{O}} - 1)^2 \sigma_\star^2}},$$

where $c^2 > 2\ln(1.25/\delta)$.

We prove this theorem in Appendix A.7. Here, $\mathcal{H}$ is the set of pairs (i,t) such that $y_i^{(t)}$ has not been seen by the adversary and only sent between the members of J. Theorem 4.7 shows how $\sigma_\star^2$ increases if honest parties cannot join the coalition J, which is an "adequately connected component" of honest users. However, if over time a party performs sufficient exchanges with other parties, they are likely to be part of J.

So far we have shown conditions under which DP can be achieved while $\sigma_\star^2$, which is an important factor to determine output accuracy, remains bounded. However, σ_Δ^2 still depends on $\Delta_{(:)}$. A large σ_Δ^2 implies a larger risk (in terms of additional error) in case a party drops out permanently. One can reduce σ_Δ^2 by increasing T. Indeed, more iterations means that in every iteration a smaller fraction of x_i is injected, and hence the needed $\|\xi^\star\|$ and $\Delta_{(:)}$ become smaller. While this dependency on T isn't explicit from the above theorems, in Section 5.2, we empirically show how σ_Δ^2 depends on T.

4.4 Topological Conditions for Privacy

We show that certain conditions on the underlying topology of the parties' exchanges already determine the number of linearly independent vectors of $\{a^{(i,t)}\}_{(i,t)\in\mathcal{H}}$ for Theorems 4.5 and 4.6.

For all $t \in [1,T]$, let

$$E_t^H = \{(i,j) \in P^H \times P^H : W_{t;j,i}^{\mathcal{O}} > 0 \wedge (i,t) \notin \mathcal{V}_{val}\}$$

be the set of edges whose messages the adversary has not seen. Let

$$G^H = \left(P^H, \bigcup_{t=1}^{T} E_t^H\right) \tag{15}$$

be the graph induced by these edges, which combines the unseen exchanges across iterations.

Theorem 4.8. *Let $\varepsilon, \delta \in (0,1)$. Let $\mathcal{E} = (\mathcal{W}, \mathcal{O}, \mathcal{V}_{val})$ be defined as in Theorem 4.1, $w_{\mathcal{O}} = \sum_{i \in P^H} w_i$ and G^H as defined in Equation (15). If G^H is strongly connected, then $\mathcal{E}$ satisfies $(\varepsilon, \delta, \mathcal{V}_{val})$-E-DP (or (ε, δ, C)-C-DP) for $\sigma_\star^2$ and σ_Δ^2 defined as in Theorem 4.6.*

We prove the theorem in Appendix A.8. Essentially, we show that if G^H is strongly connected, $\{a^{(i,t)}\}_{(i,t)\in\mathcal{H}}$ has at least $n_H - 1$ linearly independent vectors and therefore theorems 4.5 and 4.6 are applicable [1]. Analogous results can be obtained if the graph induced by the unseen interactions inside a coalition J is strongly connected for Theorem 4.7 to apply.

Finally, we provide a negative result on static interactions: that is, when all parties interact with the same neighbors in all iterations, then it is difficult to obtain privacy using our theorems. In particular if all the messages of two parties are observed, the preconditions of our main results are not met.

Theorem 4.9. *Let $\mathcal{E} = (\mathcal{W}, \mathcal{O}, \mathcal{V}_{val})$ be defined as in Theorem 4.1 Let $\mathcal{H} = P^H \times [0,T] \setminus \mathcal{V}_{val}$ be the set of pairs (i,t) such that $y_i^{(t)}$ is not seen by the adversary. For all $(i,t) \in \mathcal{H}$, let $a^{(i,t)}$ be as defined in Lemma 4.4. If parties do not change neighbors across iterations, (i.e., $W_t^{\mathcal{O}} = W_1^{\mathcal{O}}$ for all $t \in [2,T]$) and there exist $i, j \in P^H$ such that $(i,t) \in \mathcal{V}_{val}$ and $(j,t) \in \mathcal{V}_{val}$ for all $t \in [0,T]$, we have that $\{a^{(i,t)}\}_{(i,t)\in\mathcal{H}}$ has less than $n_H - 1$ independent vectors.*

We prove it in Appendix A.8. Essentially if all messages of two honest parties are observed and no party changes neighbors in the entire execution, it is not possible to meet the conditions for privacy with bounded $\sigma_\star^2$. However, changes in the neighborhood can occur accidentally with dropouts. In that case, negative conditions do not apply. Note that when parties do not drop out, negative conditions are obtained easily if at least one corrupted party exchanges messages with two honest parties. Therefore, to increase the amount of different vectors in $\{a^{(i,t)}\}_{(i,t)\in\mathcal{H}}$, it is beneficial that parties change neighbors as much as possible. This aligns with [65], which shows that dynamic networks are beneficial for privacy in decentralized learning as they increase the mixing speed of messages.

[1] Theorem 4.5 is a special case of Theorem 4.6, therefore the same reasoning applies.

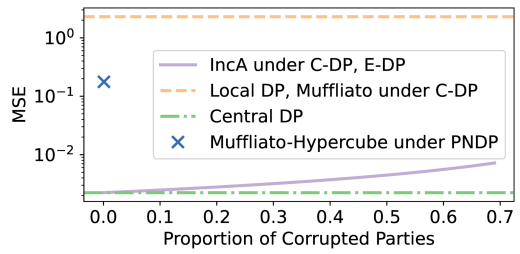


Figure 1: MSE of Local DP, Central DP, IncA and Muffliato using a Hypercube graph for $n = 2^{10}$, $\varepsilon = 0.1$ and $\delta = 10^{-5}$. The MSE of IncA is in function of n_H in the x-axis.

5 Empirical Evaluation

In this section, we empirically evaluate IncA. The main question we aim to answer is

How does the utility and cost of IncA compare with existing differentially private techniques in the same adversarial setting?

We perform our evaluation under different dropout regimes and communication parameters of IncA. In all cases, parties will generate their per-iteration neighbors by choosing k random parties from P to be their outgoing neighbors. Message weights are chosen evenly: for all $t \in [1, T]$, $W_{t;j,i} = 1/(k+1)$ for each $j \in N_{i\rightarrow}^{(t)}$ and $W_{t;i,i} = 1/(k+1)$. We compare IncA with both decentralized techniques and centrally coordinated techniques. For utility, we measure the mean squared error (MSE) of the privacy-preserving estimations with respect to the ground truth over a set of samples. For comparisons with other decentralized techniques, we measure the cost of a protocol by counting the number of exchanges and iterations that it performs. As existing centrally coordinated techniques rely on cryptographic primitives and perform operations that are substantially different in nature from that of decentralized protocols, we provide a more refined analysis that includes runtime and the total number of transferred bits.

We perform our evaluation when parties do not drop out in Section 5.1, and under dropout in sections 5.2 and 5.3. We compare with decentralized techniques in Section 5.2 and with centrally coordinated techniques in Section 5.3.

5.1 Performance Without Dropouts

We analyze the performance of Algorithm 1 in the absence of dropouts. We start by discussing the error of IncA, given by Theorem 4.5. After that, we show the communication effort required such that preconditions to apply this theorem are met.

Mean Squared Error. As a result of Theorem 4.5, we concluded in Section 4.3 that Algorithm 1 produces an estimate with MSE equal to $\frac{c^2}{n_H n \varepsilon^2}$. In Figure 1, we illustrate this for $n = 1024$, comparing IncA, Central DP, Local DP and Muffliato[20]. The latter is an approach for differentially private averaging that relaxes the adversary by reporting the mean privacy loss under Pairwise Network DP (PNDP). We show the error of IncA under E-DP, where there are no corrupted parties and C-DP, where the proportion of corrupted parties varies in the x-axis from 0 to 0.7 (the error under E-DP correspond to the left extreme of the IncA curve). This shows the degradation of IncA due to collusion, which allows the adversary to know a proportion of independent noise terms $\{\eta_i^\star\}_{i\in C}$. The error of IncA is much closer to Central DP than to Local DP even when more than 70% of the parties are corrupted. For Muffliato, we use hypercube topologies (which provide the best privacy-utility trade-offs) under no collusion. As discussed in Section 2.3, the adversary of PNDP is substantially weaker than E-DP and C-DP. We can see that the error of Muffliato is significantly higher than IncA, even with a weaker adversary. Moreover, when executed under E-DP or C-DP, Muffliato matches the error of Local DP. We provide more details on the calculations for Muffliato in Appendix B. When executed without dropouts, existent correlated noise techniques have the same error than ours [61, 66]. However, they are significantly impacted by dropouts, as we show in Section 5.2.

Communication Cost. The more iterations T that IncA performs, the higher the likelihood is to obtain $n_H - 1$ linearly independent vectors in $\{a^{(i,t)}\}_{(i,t)\in\mathcal{H}}$ in order to have the error given by Theorem 4.5. In Appendix C.1, we evaluate IncA under different topology parameters. We vary the neighbors per iteration $k \in \{1, 2, 3, 4, 5\}$ and the iterations $T \in \{2, 4, 6, 8, 10\}$. We set 50% of observed messages for Eavesdrop DP and 30% of corrupted parties for Collusion DP. We show that, under collusion, the best choice is $k = 1$ which reduces the number of observations of the adversary. Under Eavesdrop DP, increasing k reduces the round complexity but still increases communication.

For our next experiment, we focus on Collusion DP, our main threat model. We set $k = 1$ to obtain the best results. We show how the number of parties affects the communication cost, analyzing the number of messages required to apply Theorem 4.5 when n increases. For all sets of parameters, we run the protocol 10^5 times and in all cases we obtain 100% success. We focus on Collusion DP with 50% of corrupted parties. As seen in our previous experiment, the best k under C-DP is equal to 1 as it reduces the number of messages and T. In Figure 2, the number of parties n is chosen from $\{100, 500, 1000, 5000\}$. Even with 5000 parties, the communication cost is low: the amount of messages (and T) required is slightly higher than 15. For this experiment, we made a slight modification in the way parties made exchanges: at each iteration, each party chose an outgoing neighbor that he/she had not chosen in any previous iteration.

5.2 Comparison With Decentralized Techniques

We now compare IncA with decentralized techniques in the presence of dropouts under the Collusion DP threat model. We assume that a proportion $\gamma \in (0, 1)$ of parties permanently drop out at a random iteration. We use the $\mathcal{D}_{\text{Inc}}$ (Example 3.2) as parameter for $\mathcal{D}$, in which private values are evenly injected across all iterations, thereby reducing the required σ_Δ^2. The error of IncA under dropouts depends on several factors. They consist on

(1) the variance $\sigma_\star^2$ of terms $\eta^\star$, which are not canceled
(2) the variance of σ_Δ^2 terms η
(3) the amount of uncanceled terms η
(4) the coefficients w of $\mathrm{x} + \eta^\star$

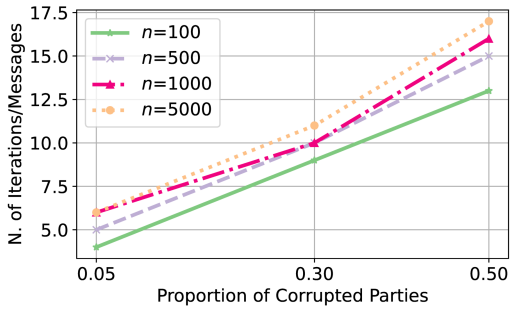


Figure 2: Minimum number of iterations (and messages since $k = 1$) of IncA such that it obtains 100% success over 10^5 runs of the protocol for $n \in \{100, 500, 1000, 5000\}$ considering a proportion of corrupted nodes between 0.05% and 0.5.

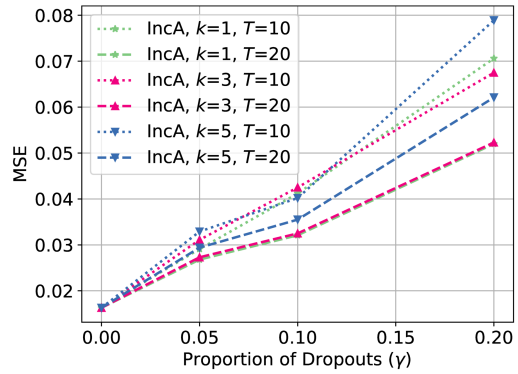
Theorems 4.6 and 4.7 provide bounds on $\sigma_\star^2$ (factor 1). We experimentally measure the impact of factors 1-4 in two experiments. The first experiment determines σ_Δ^2 (factor 2) and in the second we jointly assess the relation of all factors.

First Experiment. For each run, we fix parameters ε, δ, n, k, T, the proportion of corrupted parties $\rho \in [0, 1)$, the dropout rate γ, and the independent noise variance $\sigma_\star^2$. Then, we compute the required value of σ_Δ^2 such that IncA satisfies (ε, δ, C)-C-DP according to Theorem 4.7. We set $\sigma_\star^2 = \alpha 2\ln(1.25/\delta)/n_O\varepsilon^2$ where $n_O = n(1-\gamma-\rho)$ and a degradation parameter α. The quantity $2\ln(1.25/\delta)/n_O\varepsilon^2$ is the theoretical lower bound of $\sigma_\star^2$ according to Theorem 4.7 when n_O honest users are online all iterations. As $\sigma_\star^2$ must be bigger than the theoretical bound, we chose $\alpha = 1.3$, providing a good balance between $\sigma_\star^2$ and σ_Δ^2. For each set of parameters, we perform 1000 runs for each parameter set and keep the worst case σ_Δ^2.

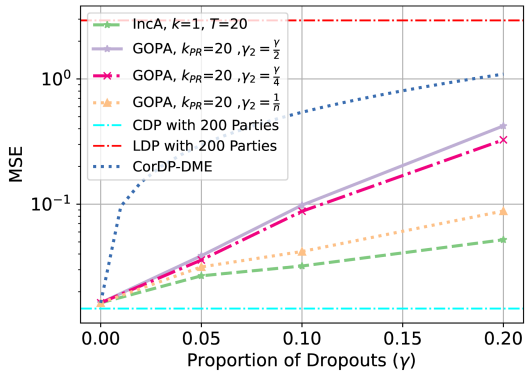
Second Experiment. For the second experiment, each run the protocol estimates the average of a random vector $\mathbf{x} \in [0, 1]^n$ using the same sets of parameters than the first and worst case values of σ_Δ^2. We perform 1000 runs and compute the MSE of the estimations with respect to the true average.

Comparison of Different Topologies. Figure 3a shows the results for $n = 200$, $\varepsilon = 0.2$, $\delta = 10^{-5}$ and proportion of corrupted parties $\rho = 0.1$. We tested IncA with parameters $k \in \{1, 3, 5\}$ and $T \in \{10, 20\}$. It is clear that, for the same amount of permanent dropouts, higher T decreases the error. Parameter k impacts different factors that determine the MSE. First, as for the case without dropouts, higher k increases the number of observed messages, which result in a larger σ_Δ^2 to achieve DP guarantees. Second, higher k also increases the mixing speed of the protocol reducing σ_Δ^2. In experiments, these effects compensate for $k \in \{1, 3, 5\}$, providing similar results, although smaller k yields smaller error and fewer amount of messages per iteration.

Comparison With Related Techniques. We compare our protocol with GOPA [61] and Cor-DP-DME [66], which are fully decentralized and provide similar MSE when there are no dropouts (see Appendix D.3 for more details on these protocols). The main differences between GOPA and CorDP-DME are that the former is



(a) Parameters of IncA



(b) IncA and Previous Work

Figure 3: Mean Squared Error (MSE) in function of the proportion γ of dropouts for IncA, GOPA and CorDP-DME protocols, over 1000 runs where $n = 200$, $\varepsilon = 0.2$, $\delta = 10^{-5}$ and proportion of corrupted parties $\rho = 0.1$. As references we include the MSE of Central DP (CDP) and Local DP (LDP) with n parties.

agnostic to the communication topology and provides a dropout mitigation phase, while CorDP-DME focuses on the complete graph (i.e., all parties communicate with each other). The mitigation phase of GOPA consists of parties rolling back the uncanceled noise due to dropouts. However, if additional dropouts occur during recovery, some uncanceled terms remain.

Figure 3b shows the comparison of IncA, GOPA and Cor-DP-DME. For GOPA, k_{PR} is the number of exchanges at each round. In Cor-DP-DME each party performs n messages per round. Similarly to IncA, GOPA is evaluated with two experiments of 1000 runs, first determining worst-case empirical variance and then the MSE. Both experiments are done in 1000 runs. For Cor-DP-DME, we ignore the error induced by missing private values due to dropout and only compute the error induced by the variance of noise terms when they are uncanceled, which is a lower bound on the real error and can be computed theoretically.

We evaluate all of the protocols for a proportion of dropouts $\gamma \in \{0.05, 0.1, 0.2\}$, and set GOPA and IncA to send 20 messages:

$k_{PR} = 20$ for GOPA and $(k, T) = (1, 20)$ for IncA. GOPA is very sensitive to the distribution of dropouts over its two rounds. We denote by $\gamma_2 \in [0, \gamma]$ to the proportion of dropouts that happen in the rollback round. We consider $\gamma_2 \in \{\frac{\gamma}{2}, \frac{\gamma}{4}, \frac{1}{n}\}$, note that $\gamma_2 = \frac{1}{n}$ assumes an extremely optimistic scenario for GOPA, in which only a single party drops out in the rollback round. In all cases, we observe that IncA outperforms the other protocols for the same proportion of dropouts. If $\gamma_2 = \frac{1}{n}$ GOPA approaches IncA, but significantly degrades if $\gamma_2 \in \{\frac{\gamma}{2}, \frac{\gamma}{4}\}$. Cor-DP-DME provides the worst performance even if sends substantially more messages, showing the impact when noise terms are not rolled back.

5.3 Comparison With Centralized Techniques

We conclude our experimental section by comparing IncA with centrally-coordinated techniques. As a baseline, we consider secure aggregation with locally added Gaussian noise in [46] (SAgg). We use the secure aggregation primitive proposed by [8], which offers good scalability properties with respect to the number of parties and the input dimension in our semi-honest setting (see Appendix D.3 for more details on secure aggregation). For simplicity, we ignore the errors introduced in the Gaussian noise by the finite precision required for secure aggregation.

SAgg extensively uses cryptographic primitives such as pseudo-random generation and secret sharing. Therefore, to perform more accurate comparisons, our evaluation measures the runtime and the number of transferred bits in both protocols. To do so, we perform simulations in ABIDES [14], a discrete event simulation framework recently adopted to deploy secure aggregation primitives [37, 51]. We consider the runtime of a protocol to be the average time a single party spends performing computations or blocked waiting. However, we do not include the delays of messages in the network. For communication, we measure the total number of bits sent by all parties. For each set of parameters we perform 10 simulations, reporting the average and standard deviation.

Using the same principles as in previous experiments, for a given setting (n, corrupted proportion ρ and dropout proportion γ) we select parameters of IncA and SAgg that do not compromise privacy over 1000 runs. For both protocols, dropouts are evenly distributed across rounds. We consider each party's private values to be d-dimensional vectors of 32-bit values. All experiments were conducted in a machine with Ubuntu 22.04, a 12th Gen Intel® Core™ i9-12950HX processor (24 cores) and 32 GB of memory.

Figure 4 shows our evaluation for varying the number of dimensions d of the input vectors and the number of parties n. For both cases, $\rho = 0.1$ and $\gamma = 0.1$. IncA is executed with $(k, T) = (1, 40)$ and a Dissemination Phase using gossip averaging [13] of $T_D = 20$ iterations. SAgg is set such that each party interacts with $k_{SA} = 16$ other parties. In the top plots, we show the costs for $d \in \{10^i\}_{i\in[0,6]}$ and $n = 200$. SAgg transmits a significantly lower amount of bits for $d > 10^2$ due to the use of seeds to compress messages. IncA is faster than SAgg for $d < 10^5$ as it does not use cryptographic primitives and does not remain blocked by the computations of the server. Once d becomes the dominant factor, both protocols exhibit similar runtime, but the cost of IncA increases at a greater factor. In the bottom plots of this figure, we vary $n \in \{200, 300, 400, 500\}$ and

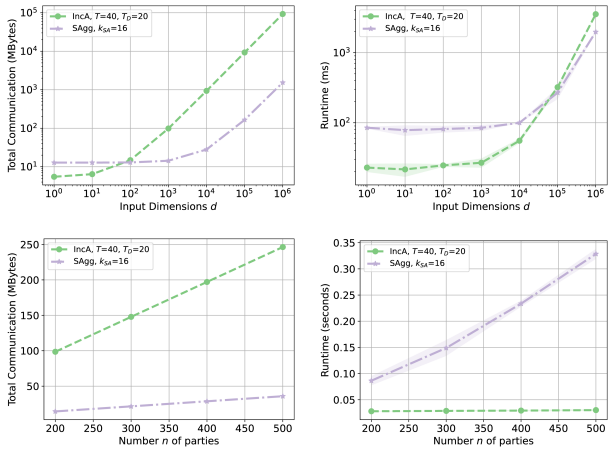


Figure 4: Communication cost in MegaBytes and computational cost in milliseconds (ms) or seconds of IncA and SAgg. Costs vary as a function of the number of dimensions d of input vectors on the top plots and as a function of n in the bottom plots.

fix $d = 10^3$. We see a similar advantage of SAgg in terms of communication cost due to seed compression. In terms of computation, we can see that the runtime of IncA remains constant. On the other hand, in SAgg the computational work of the server is linear in n, as is the waiting time experienced by each party. In Appendix C.2, we show performance when varying the total number of iterations in IncA and compare its accuracy with respect to that of SAgg for many combinations of ρ and γ.

6 Related Work

Many distributed data processing tasks can be accomplished through local computations and subsequent aggregation of their outcomes across participating entities [6, 11, 47, 52, 57, 58]. This makes averaging a simple, yet powerful primitive. Averages can be computed in a federated way [11, 52] or in a decentralized way using gossip protocols [10, 13, 22, 60]. However, many early approaches required that parties share their data in the clear, which may be exploited by external attackers [33, 49, 53, 55, 59, 65].

Differentially Private Mean Estimation. We study approaches that prevent information leakage by providing differential privacy guarantees [28]. Satisfying DP guarantees typically requires to perturb the computation with noise, which compromises accuracy. In the central model of DP (CDP), where a trusted curator performs the computation, one can achieve a mean squared error of $O(1/n^2)$ induced by privacy noise [30]. If such a party is not available, parties could add noise before sharing their sensitive information, which further compromises accuracy. In the local model of DP (LDP)[15, 25, 45, 48, 50], where parties fully privatize their data before using it in a collaborative computation, the estimation error is a factor of n greater than in CDP and only yields acceptable accuracy when the number of parties is massive [23, 32]. In Appendix D.1, we describe LDP and CDP in more detail.

Cryptographic Primitives. Instead of a trusted curator, one can also use cryptographic primitives such as Multiparty Computation (MPC) [21, 68] or other models such as Shuffling [42]. These techniques can recover similar privacy utility trade-offs of Central DP [2, 5, 16, 17, 34, 40, 41, 46] or exactly match them [27, 44, 62]. However, besides secure aggregation which we explain below, they incur large communication costs or substantially relax trust assumptions by considering multiple non-colluding servers or trusted parties which anonymize messages by shuffling them.

Central Coordination. Secure Aggregation [7, 8, 12, 37, 51, 64] is a special MPC primitive tailored to compute sums. Notably, [7, 8, 37, 63] offer scalability to a large number of participants as they are constant round and each party only has to communicate with an $O(poly\text{-}\log(n))$ number of other parties. These approaches rely on a central coordinator that acts as an intermediary on all communications and has a global view of them. This design has two important advantages. First, it allows for single round failure detection and message dissemination. Second, it plays a key role when recovering from failures of cryptographic noise based protocols, where even minor inconsistencies can make the output random noise and highly structured approaches are required to ensure security. However, this central coordinator must not fail during the computation and must process the messages of all parties. We provide a detailed description of these approaches in Appendix D.3, and experimentally compare IncA to [8] in Section 5.3.

Decentralized/Gossip Protocols. Decentralized protocols are applicable when such a central orchestrator is not an option. However, they present several challenges. First, each participant has a partial and delayed view of the failures of other parties. Second, if scalability to the number of parties is desired (i.e., with a sub-linear cost with respect to n) this knowledge further narrows to small communication neighborhoods. Therefore, the global dissemination of computations and protocol state (e.g., which parties are currently online) take multiple rounds, making structured recovery strategies challenging to implement, especially when the global state evolves in a single round.

Certain approaches for decentralized differentially private averaging focus on the relaxation of the adversary's view due to decentralization [18–20]. However, these relaxations are aggressive as they consider average-case instead of worst-case privacy loss [20], a constant number of colluders instead of a proportion of n [18–20], and do not consider that the communication structure is public. The techniques proposed in [18, 19] are based on random walks, which forces sequential communication and a very large number of rounds. We discuss these threat models in more detail in Section 2.3, a comparison to [20] (the only work that focuses on mean estimation) in Section 5.1, and a more detailed description of gossip protocols in Appendix D.2.

Another type of decentralized protocols obtains an accuracy that is comparable to central DP with a similar strategy to centrally coordinated secure aggregation, but replacing cryptographic pairwise noise with Gaussian noise [4, 61, 66]. The resilience against dropouts relies on the bounded impact of the non-cryptographic noise terms [4, 66] and on recovery strategies that are simpler than the ones used in secure aggregation [61]. Moreover, [4] focuses on decentralized SGD and shows that ML updates can be applied to ML models that have been obfuscated with bounded correlated noise without experiencing an asymptotically significant impact on convergence. [61, 66] focus on averaging, and [61] shows that only $O(\log(n))$ messages per party are required to obtain robust topologies and bounded correlated noise. However, accuracy still degrades significantly if parties disconnect in the middle of the computation as pairwise noise does not cancel, even when applying dropout mitigation techniques. [66] is restricted to the case where all parties communicate with each other, which offers poor scalability in n. We present these protocols in detail in Appendix D.3 and compare to them in Section 5.2.

Achieving Malicious Security. IncA considers passive attackers, while some works are resilient against parties that arbitrarily deviate from the protocol. [61] offers a variant for these kinds of attacks. However, it requires a centralized structure with similar assumptions as the central coordinator. Alternatively, one can implement this structure in the decentralized setting with a blockchain [56] or use general purpose maliciously secure MPC [21] under heavy computation and communication costs. If scalability is desired, the centrally coordinated protocols previously discussed can prevent some types of malicious behavior. However, these types remain narrow (e.g., a server faking dropouts or out-of-range inputs) and their privacy is significantly degraded. Finally, [3] mitigates the impact of poisoned contributions, but does not ensure that actively corrupted parties follow the protocol.

In Appendix E, we expand our discussion to alternatives, identifying advantages and limitations of our approach in terms of computational and communication cost, dropout resilience and security.

7 Conclusion and Future Work

We propose IncA, a protocol for fully decentralized mean estimation, a key primitive for decentralized and privacy-preserving computation. We demonstrate that IncA not only satisfies theoretical privacy guarantees but also maintains accuracy despite party failures and collusion, all while incurring reasonable communication costs. Directions of future work include the evaluation of our technique within larger systems such as the computation of federated learning models, the derivation of explicit bounds on the variance of correlated noise and the theoretical quantification of the probability that our conditions hold under random communication graphs.

Acknowledgments

We thank the anonymous reviewers for their valuable feedback, which helped improve this paper. This work was supported by the French government managed by the Agence Nationale de la Recherche (ANR) through France 2030 program with the reference ANR-23-PEIA-005 (REDEEM project).

References

[1] Gergely Ács and Claude Castelluccia. 2011. I have a dream!(differentially private smart metering). In *International Workshop on Information Hiding*. Springer, 118–132.

[2] Naman Agarwal, Peter Kairouz, and Ziyu Liu. 2021. The skellam mechanism for differentially private federated learning. *Advances in Neural Information Processing Systems* 34 (2021), 5052–5064.

[3] Youssef Allouah, Rachid Guerraoui, and John Stephan. 2025. Towards Trustworthy Federated Learning with Untrusted Participants. *arXiv preprint arXiv:2505.01874* (2025).

[4] Youssef Allouah, Anastasia Koloskova, Aymane El Firdoussi, Martin Jaggi, and Rachid Guerraoui. 2024. The Privacy Power of Correlated Noise in Decentralized Learning. *arXiv preprint arXiv:2405.01031* (2024).

[5] Borja Balle, James Bell, Adria Gascón, and Kobbi Nissim. 2020. Private summation in the multi-message shuffle model. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*. 657–676.

[6] Antoine Barczewski, Amal Mawass, and Jan Ramon. 2025. Differentially Private Empirical Cumulative Distribution Functions. *arXiv preprint arXiv:2502.06651* (2025).

[7] James Bell, Adrià Gascón, Tancrède Lepoint, Baiyu Li, Sarah Meiklejohn, Mariana Raykova, and Cathie Yun. 2023. {ACORN}: input validation for secure aggregation. In *32nd USENIX Security Symposium (USENIX Security 23)*. 4805–4822.

[8] James Henry Bell, Kallista A Bonawitz, Adrià Gascón, Tancrède Lepoint, and Mariana Raykova. 2020. Secure single-server aggregation with (poly) logarithmic overhead. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*. 1253–1269.

[9] Enrique Tomás Martínez Beltrán, Mario Quiles Pérez, Pedro Miguel Sánchez Sánchez, Sergio López Bernal, Gérôme Bovet, Manuel Gil Pérez, Gregorio Martínez Pérez, and Alberto Huertas Celdrán. 2023. Decentralized federated learning: Fundamentals, state of the art, frameworks, trends, and challenges. *IEEE Communications Surveys & Tutorials* 25, 4 (2023), 2983–3013.

[10] Raphaël Berthier, Francis Bach, and Pierre Gaillard. 2020. Accelerated gossip in networks of given dimension using jacobi polynomial iterations. *SIAM Journal on Mathematics of Data Science* 2, 1 (2020), 24–47.

[11] Akash Bharadwaj and Graham Cormode. 2024. Federated computation: a survey of concepts and challenges. *Distributed and Parallel Databases* 42, 3 (2024), 299–335.

[12] Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. 2017. Practical secure aggregation for privacy-preserving machine learning. In *proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. 1175–1191.

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

[14] David Byrd, Maria Hybinette, and Tucker Hybinette Balch. 2019. Abides: Towards high-fidelity market simulation for ai research. *arXiv preprint arXiv:1904.12066* (2019).

[15] Wei-Ning Chen, Peter Kairouz, and Ayfer Ozgur. 2020. Breaking the communication-privacy-accuracy trilemma. *Advances in Neural Information Processing Systems* 33 (2020), 3312–3324.

[16] Wei-Ning Chen, Ayfer Ozgur, and Peter Kairouz. 2022. The poisson binomial mechanism for unbiased federated learning with secure aggregation. In *International Conference on Machine Learning*. PMLR, 3490–3506.

[17] Albert Cheu, Adam Smith, Jonathan Ullman, David Zeber, and Maxim Zhilyaev. 2019. Distributed differential privacy via shuffling. In *Advances in Cryptology–EUROCRYPT 2019: 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Darmstadt, Germany, May 19–23, 2019, Proceedings, Part I 38*. Springer, 375–403.

[18] Edwige Cyffers and Aurélien Bellet. 2022. Privacy amplification by decentralization. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 5334–5353.

[19] Edwige Cyffers, Aurélien Bellet, and Jalaj Upadhyay. 2024. Differentially private decentralized learning with random walks. *arXiv preprint arXiv:2402.07471* (2024).

[20] Edwige Cyffers, Mathieu Even, Aurélien Bellet, and Laurent Massoulié. 2022. Muffliato: Peer-to-peer privacy amplification for decentralized optimization and averaging. *Advances in Neural Information Processing Systems* 35 (2022), 15889–15902.

[21] Ivan Damgård, Marcel Keller, Enrique Larraia, Valerio Pastro, Peter Scholl, and Nigel P Smart. 2013. Practical covertly secure MPC for dishonest majority–or: breaking the SPDZ limits. In *Computer Security–ESORICS 2013: 18th European Symposium on Research in Computer Security, Egham, UK, September 9-13, 2013. Proceedings 18*. Springer, 1–18.

[22] Alan Demers, Dan Greene, Carl Hauser, Wes Irish, John Larson, Scott Shenker, Howard Sturgis, Dan Swinehart, and Doug Terry. 1987. Epidemic algorithms for replicated database maintenance. In *Proceedings of the sixth annual ACM Symposium on Principles of distributed computing*. 1–12.

[23] Bolin Ding, Janardhan Kulkarni, and Sergey Yekhanin. 2017. Collecting telemetry data privately. *Advances in Neural Information Processing Systems* 30 (2017).

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

[25] John C Duchi, Michael I Jordan, and Martin J Wainwright. 2013. Local privacy, data processing inequalities, and statistical minimax rates. *arXiv preprint arXiv:1302.3203* (2013).

[26] Sanghamitra Dutta, Gauri Joshi, Soumyadip Ghosh, Parijat Dube, and Priya Nagpurkar. 2018. Slow and stale gradients can win the race: Error-runtime tradeoffs in distributed SGD. In *International conference on artificial intelligence and statistics*. PMLR, 803–812.

[27] Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. 2006. Our data, ourselves: Privacy via distributed noise generation. In *Advances in cryptology-EUROCRYPT 2006: 24th annual international conference on the theory and applications of cryptographic techniques, st. Petersburg, Russia, May 28-June 1, 2006. proceedings 25*. Springer, 486–503.

[28] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. 2006. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography: Third Theory of Cryptography Conference, TCC 2006, New York, NY, USA, March 4-7, 2006. Proceedings 3*. Springer, 265–284.

[29] Cynthia Dwork and Aaron Roth. 2014. The Algorithmic Foundations of Differential Privacy. *Foundations and Trends in Theoretical Computer Science* 9, 3–4 (2014), 1–277.

[30] Cynthia Dwork, Aaron Roth, et al. 2014. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science* 9, 3–4 (2014), 211–407.

[31] Tariq Elahi, George Danezis, and Ian Goldberg. 2014. Privex: Private collection of traffic statistics for anonymous communication networks. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*. 1068–1079.

[32] Úlfar Erlingsson, Vasyl Pihur, and Aleksandra Korolova. 2014. Rappor: Randomized aggregatable privacy-preserving ordinal response. In *Proceedings of the 2014 ACM SIGSAC conference on computer and communications security*. 1054–1067.

[33] Jonas Geiping, Hartmut Bauermeister, Hannah Dröge, and Michael Moeller. 2020. Inverting gradients-how easy is it to break privacy in federated learning? *Advances in neural information processing systems* 33 (2020), 16937–16947.

[34] Badih Ghazi, Ravi Kumar, Pasin Manurangsi, and Rasmus Pagh. 2020. Private counting from anonymous messages: Near-optimal accuracy with vanishing communication overhead. In *International Conference on Machine Learning*. PMLR, 3505–3514.

[35] Slawomir Goryczka and Li Xiong. 2015. A comprehensive comparison of multiparty secure additions with differential privacy. *IEEE transactions on dependable and secure computing* 14, 5 (2015), 463–477.

[36] Rachid Guerraoui and Luis Rodrigues. 2006. *Introduction to reliable distributed programming*. Springer Science & Business Media.

[37] Yue Guo, Antigoni Polychroniadou, Elaine Shi, David Byrd, and Tucker Balch. 2024. MicroSecAgg: Streamlined Single-Server Secure Aggregation. *Proceedings on Privacy Enhancing Technologies* (2024).

[38] Suyog Gupta, Wei Zhang, and Fei Wang. 2016. Model accuracy and runtime tradeoff in distributed deep learning: A systematic study. In *2016 IEEE 16th International Conference on Data Mining (ICDM)*. IEEE, 171–180.

[39] István Hegedűs, Gábor Danner, and Márk Jelasity. 2021. Decentralized learning works: An empirical comparison of gossip learning and federated learning. *J. Parallel and Distrib. Comput.* 148 (2021), 109–124.

[40] Mikko A Heikkilä. 2025. On Using Secure Aggregation in Differentially Private Federated Learning with Multiple Local Steps. *Transactions on Machine Learning Research* (2025).

[41] Hafiz Imtiaz, Jafar Mohammadi, and Anand D Sarwate. 2019. Distributed differentially private computation of functions with correlated noise. *arXiv preprint arXiv:1904.10059* (2019).

[42] Yuval Ishai, Eyal Kushilevitz, Rafail Ostrovsky, and Amit Sahai. 2006. Cryptography from anonymity. In *2006 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS'06)*. IEEE, 239–248.

[43] Rob Jansen and Aaron Johnson. 2016. Safely measuring tor. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. 1553–1567.

[44] Bargav Jayaraman, Lingxiao Wang, David Evans, and Quanquan Gu. 2018. Distributed learning without distress: Privacy-preserving empirical risk minimization. *Advances in neural information processing systems* 31 (2018).

[45] Peter Kairouz, Keith Bonawitz, and Daniel Ramage. 2016. Discrete distribution estimation under local privacy. In *International Conference on Machine Learning*. PMLR, 2436–2444.

[46] Peter Kairouz, Ziyu Liu, and Thomas Steinke. 2021. The distributed discrete gaussian mechanism for federated learning with secure aggregation. In *International Conference on Machine Learning*. PMLR, 5201–5212.

[47] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. 2021. Advances and open problems in federated learning. *Foundations and trends® in machine learning* 14, 1–2 (2021), 1–210.

[48] Peter Kairouz, Sewoong Oh, and Pramod Viswanath. 2014. Extremal mechanisms for local differential privacy. *Advances in neural information processing systems* 27 (2014).

[49] Sanjay Kariyappa, Chuan Guo, Kiwan Maeng, Wenjie Xiong, G Edward Suh, Moinuddin K Qureshi, and Hsien-Hsin S Lee. 2023. Cocktail party attack: Breaking aggregation-based privacy in federated learning using independent component

analysis. In *International Conference on Machine Learning*. PMLR, 15884–15899.
[50] Shiva Prasad Kasiviswanathan, Homin K Lee, Kobbi Nissim, Sofya Raskhodnikova, and Adam Smith. 2011. What can we learn privately? *SIAM J. Comput.* 40, 3 (2011), 793–826.
[51] Yiping Ma, Jess Woods, Sebastian Angel, Antigoni Polychroniadou, and Tal Rabin. 2023. Flamingo: Multi-round single-server secure aggregation with applications to private federated learning. In *2023 IEEE Symposium on Security and Privacy (SP)*. IEEE, 477–496.
[52] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*. PMLR, 1273–1282.
[53] Luca Melis, Congzheng Song, Emiliano De Cristofaro, and Vitaly Shmatikov. 2019. Exploiting unintended feature leakage in collaborative learning. In *2019 IEEE symposium on security and privacy (SP)*. IEEE, 691–706.
[54] Ilya Mironov. 2017. Rényi differential privacy. In *2017 IEEE 30th computer security foundations symposium (CSF)*. IEEE, 263–275.
[55] Abdellah El Mrini, Edwige Cyffers, and Aurélien Bellet. 2024. Privacy Attacks in Decentralized Learning. *arXiv preprint arXiv:2402.10001* (2024).
[56] Satoshi Nakamoto. 2008. Bitcoin: A peer-to-peer electronic cash system. *Available at SSRN 3440802* (2008).
[57] Julien Nicolas, César Sabater, Mohamed Maouche, Sonia Ben Mokhtar, and Mark Coates. 2024. Differentially private and decentralized randomized power method. *arXiv preprint arXiv:2411.01931* (2024).
[58] Julien Nicolas, César Sabater, Mohamed Maouche, Sonia Ben Mokhtar, and Mark Coates. 2025. Secure Federated Graph-Filtering for Recommender Systems. *arXiv preprint arXiv:2501.16888* (2025).
[59] Dario Pasquini, Mathilde Raynal, and Carmela Troncoso. 2023. On the (in) security of peer-to-peer decentralized machine learning. In *2023 IEEE Symposium on Security and Privacy (SP)*. IEEE, 418–436.
[60] Boris Pittel. 1987. On spreading a rumor. *SIAM J. Appl. Math.* 47, 1 (1987), 213–223.
[61] César Sabater, Aurélien Bellet, and Jan Ramon. 2022. An accurate, scalable and verifiable protocol for federated differentially private averaging. *Machine Learning* 111, 11 (2022), 4249–4293.
[62] César Sabater, Florian Hahn, Andreas Peter, and Jan Ramon. 2023. Private sampling with identifiable cheaters. *Proceedings on Privacy Enhancing Technologies* 2023, 2 (2023).
[63] Jinhyun So, Başak Güler, and A Salman Avestimehr. 2021. Turbo-aggregate: Breaking the quadratic aggregation barrier in secure federated learning. *IEEE Journal on Selected Areas in Information Theory* 2, 1 (2021), 479–489.
[64] Riccardo Taiello, Melek Önen, Clémentine Gritti, and Marco Lorenzi. 2024. Let Them Drop: Scalable and Efficient Federated Learning Solutions Agnostic to Stragglers. In *Proceedings of the 19th International Conference on Availability, Reliability and Security*. 1–12.
[65] Ousmane Touat, Jezekael Brunon, Yacine Belal, Julien Nicolas, Mohamed Maouche, César Sabater, and Sonia Ben Mokhtar. 2024. Scrutinizing the Vulnerability of Decentralized Learning to Membership Inference Attacks. *arXiv preprint arXiv:2412.12837* (2024).
[66] Sajani Vithana, Viveck R Cadambe, Flavio P Calmon, and Haewon Jeong. 2025. Correlated privacy mechanisms for differentially private distributed mean estimation. In *2025 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*. IEEE, 590–614.
[67] Jianyu Wang, Anit Kumar Sahu, Zhouyi Yang, Gauri Joshi, and Soummya Kar. 2019. MATCHA: Speeding up decentralized SGD via matching decomposition sampling. In *2019 Sixth Indian Control Conference (ICC)*. IEEE, 299–300.
[68] Andrew C Yao. 1982. Protocols for secure computations. In *23rd annual symposium on foundations of computer science (sfcs 1982)*. IEEE, 160–164.

A Missing Theoretical Material

A.1 Proof of Lemma 3.5

Lemma 3.5. $\mathcal{D}_{\text{EI}}$ and $\mathcal{D}_{\text{Inc}}$ are Valid Gaussians.

Proof. $\mathcal{D}_{\text{EI}}$ is a (c, Z)-Gaussian where c and Z are as follows:

- $c_t = 1$ for $t = 0$ and $c_t = 0$ for $t \in [1, T]$,
- $Z_{0,:} = \mathbf{1}^\top$,
- for all $t \in [1, T]$,

$$Z_{t,k} = \begin{cases} -1 & \text{if } k = t \\ 0 & \text{if } k \in [1, T] \setminus \{t\}. \end{cases}$$

$\mathcal{D}_{\text{Inc}}$ is a (c, Z)-Gaussian where

- $c = \mathbf{1}/(T+1)$
- for all $t \in [0, T]$ and $k \in [1, T]$

$$Z_{t,k} = \begin{cases} 1 & \text{if } k = t+1, \\ -1 & \text{if } k = t, \\ 0 & \text{otherwise.} \end{cases}$$

For both $\mathcal{D}_{\text{EI}}$ and $\mathcal{D}_{\text{Inc}}$, (c, Z) and $Z_{-T,:}$ are invertible matrices. □

A.2 Knowledge in Linear Relations

We now show how to construct the system of linear equations presented in Equation (10). Recall that in our privacy proofs, each party $i \in P$ uses $\mathcal{D}_{Gauss}^{(c^{(i)}, Z^{(i)})}(\mathrm{x}_i + \eta_i^\star)$ as the parameter for $\mathcal{D}^O(i, :)$ in Algorithm 2. Recall that $c^{(i)} \in \mathbb{R}^{T+1}$ and $Z^{(i)} \in \mathbb{R}^{(T+1)\times T}$ dictate how private values and canceling noise terms are introduced at each iteration for party i.

Vectorized Gossip. Let $y^{(t)} \coloneqq (y_1^{(t)}, \ldots, y_n^{(t)})^\top \in \mathbb{X}^n$ for all $t \in \{0, \ldots, T\}$, where $y_i^{(t)}$ is defined for all $i \in P$ and $t \in [0, T]$ in equations (6) and (7). Recall that b_i is a column vector with a 1 in the ith coordinate and 0s elsewhere. Let $D_c^{(t)} = \mathrm{diag}(c_t^{(1)}, \ldots, c_t^{(n)}) \in \mathbb{R}^{n\times n}$ for all $t \in [0, T]$. The vectorized version of the Initialization Phase (line 5) is

$$y^{(0)} \coloneqq D_c^{(0)}(\mathrm{x} + \eta^\star) + \sum_{i=1}^{n} \left(Z_{0,:}^{(i)} \eta_{i,:}^\top\right) \mathrm{b}_i \tag{16}$$

and the vectorized version of the Mixing Phase (line 15) is

$$y^{(t)} = W_t^O y^{(t-1)} + D_c^{(t)}(\mathrm{x} + \eta^\star) + \sum_{i=1}^{n} \left(Z_{t,:}^{(i)} \eta_{i,:}^\top\right) \mathrm{b}_i \tag{17}$$

for all $t \in [1, T]$. Note that for all $t \in [0, T]$ and $i \in P$, $\left(Z_{t,:}^{(i)} \eta_{i,:}^\top\right) \mathrm{b}_i$ is a vector where only the i-th value is different from 0.

Let $k \coloneqq nT$ be the total number of canceling noise terms and recall that $\eta_{(:)} \coloneqq (\eta_{1,:}, \ldots, \eta_{n,:})^\top \in \mathbb{X}^k$. For all $t \in [0, T]$, let

$$\zeta_t \coloneqq \left(\mathrm{b}_1 Z_{t,:}^{(1)}, \quad \ldots, \quad \mathrm{b}_n Z_{t,:}^{(n)}\right) \in \mathbb{R}^{n\times k}$$

be a block matrix formed by horizontally concatenating n blocks, each of size $n \times T$ with only one nonzero row (i.e., the i-th block only has the i-th row with non-zero values). Equations (16) and (17) are equivalent to

$$y^{(0)} = D_c^{(0)}(\mathrm{x} + \eta^\star) + \zeta_0 \eta_{(:)} \tag{18}$$

and

$$y^{(t)} = W_t^O y^{(t-1)} + D_c^{(t)}(\mathrm{x} + \eta^\star) + \zeta_t \eta_{(:)} \tag{19}$$

respectively.

Linear Equations. From Equation (18) and applying Equation (19) recursively we have that

$$B_t(\mathrm{x} + \eta^\star) + A_t \eta_{(:)} = y^{(t)} \tag{20}$$

for all $t \in [0, \ldots, T]$ where

$$B_0 \coloneqq D_c^{(0)}, \qquad B_t \coloneqq W_t^O B_{t-1} + D_c^{(t)} \in \mathbb{R}^{n\times n}, \quad \forall t \in [T]$$
$$A_0 \coloneqq \zeta_0, \qquad A_t \coloneqq W_t^O A_{t-1} + \zeta_t \in \mathbb{R}^{n\times k}, \quad \forall t \in [T].$$

Let

$$B^* \coloneqq \begin{pmatrix} B_0 \\ \vdots \\ B_T \end{pmatrix} \in \mathbb{R}^{k\times n}, \quad A^* \coloneqq \begin{pmatrix} A_0 \\ \vdots \\ A_T \end{pmatrix} \in \mathbb{R}^{k\times k}, \text{ and } y \coloneqq \begin{pmatrix} y^{(0)} \\ \vdots \\ y^{(T)} \end{pmatrix} \in \mathbb{R}^k.$$

The we can summarize Equation (20) for all $t \in \{0, \dots, T\}$ by

$$B^*(\mathrm{x} + \eta^\star) + A^*\eta_{(:)} = y \tag{21}$$

Knowledge of the Adversary. Recall that C is the set of corrupted parties (if we are under the C-DP setting), $\mathcal{V}_{val}$ is the set of observations made by the adversary, P^H is the set of honest parties with size n_H and k_H the total number of canceling noise terms generated by honest users. Additionally, recall that x^H, $\eta^{\star H}$ and $\eta^H_{(:)}$ are the private vectors and noise terms of them. Let x^C, $\eta^{\star C}$ and $\eta^C_{(:)}$ be the vectors of values and noise terms generated by corrupted parties in C and $[i,t]$ be the index of y such that $y_{[i,t]} = y_i^{(t)}$. Then,

$$B^{\mathcal{V}}(\mathrm{x} + \eta^\star) + A^{\mathcal{V}}\eta_{(:)} = y'_{\mathcal{V}}$$

is the linear system obtained by only keeping the rows with indexes $\{[i,t] : (i,t) \in \mathcal{V}_{val}\}$, which are observable by the adversary. Rearranging terms to separate the unknowns of the adversary, we generate the equivalent system

$$\begin{pmatrix} B & B^C \end{pmatrix} \begin{pmatrix} \mathrm{x}^H + \eta^{\star H} \\ \mathrm{x}^C + \eta^{\star C} \end{pmatrix} + \begin{pmatrix} A & A^C \end{pmatrix} \begin{pmatrix} \eta^H_{(:)} \\ \eta^C_{(:)} \end{pmatrix} = y'_{\mathcal{V}}.$$

Above, we assume that rows which are linear combinations of the others are removed such that (B, A) is full rank. Removing these rows does not reduce the adversary's knowledge. The remaining system is also equivalent to

$$B(\mathrm{x}^H + \eta^{\star H}) + A\eta^H_{(:)} = y'_{\mathcal{V}} - B^C(\mathrm{x}^C + \eta^{\star C}) - A^C\eta^C_{(:)}$$

which is exactly the system we described in Equation (10) for $y_{\mathcal{V}} = y'_{\mathcal{V}} - B^C(\mathrm{x}^C + \eta^{\star C}) - A^C\eta^C_{(:)}$, a vector known to the adversary.

A.3 Preliminary Lemma

Before presenting our privacy proofs, we prove a preliminary lemma.

Lemma A.1. *Given $\varepsilon, \delta \in (0,1)$, Equations*

$$\varepsilon - \frac{1}{2}\theta \geq \sqrt{\theta} \tag{22}$$

and

$$\frac{1}{2}\frac{(\varepsilon - \frac{1}{2}\theta)^2}{\theta} \geq \ln\left(\frac{2}{\delta\sqrt{2\pi}}\right) \tag{23}$$

are satisfied for

$$\theta \leq \frac{\varepsilon^2}{c^2} \tag{24}$$

where $c^2 > 2\ln(1.25/\delta)$.

Proof. This lemma has been proven as part of the proof of bounds of the Gaussian Mechanism [30, Theorem A.1]. For $\theta \leq \frac{\varepsilon^2}{c^2}$, we have that Equation (22) is implied if

$$\begin{aligned} &\varepsilon - \frac{\varepsilon^2}{2c^2} > \frac{\varepsilon}{c} \\ \iff &\varepsilon > \frac{\varepsilon}{c} + \frac{\varepsilon^2}{2c^2} \\ \iff &\frac{c}{\varepsilon}\varepsilon > \frac{c}{\varepsilon}\left(\frac{\varepsilon}{c} + \frac{\varepsilon^2}{2c^2}\right) \\ \iff &c - \frac{\varepsilon}{2c} > 1. \end{aligned}$$

For $\varepsilon \leq 1$, the above is implied when $c > 3/2$. Moreover, for $\theta \leq \frac{\varepsilon^2}{c^2}$ Equation (23) is implied if

$$\begin{aligned} &\frac{1}{2}\frac{\left(\varepsilon - \frac{\varepsilon^2}{2c^2}\right)^2}{\frac{\varepsilon^2}{c^2}} \geq \ln\left(\frac{2}{\delta\sqrt{2\pi}}\right) \\ \iff &\frac{1}{2}\left(\varepsilon - \frac{\varepsilon^2}{2c^2}\right)^2 \geq \ln\left(\frac{2}{\delta\sqrt{2\pi}}\right)\frac{\varepsilon^2}{c^2} \\ \iff &\frac{1}{2}\left(\varepsilon^2 - \frac{\varepsilon^3}{c^2} + \frac{\varepsilon^4}{4c^4}\right) \geq \ln\left(\frac{2}{\delta\sqrt{2\pi}}\right)\frac{\varepsilon^2}{c^2} \\ \iff &c^2 - \varepsilon + \frac{\varepsilon^2}{4c^2} \geq 2\ln\left(\frac{2}{\delta\sqrt{2\pi}}\right). \end{aligned}$$

For $c > 3/2$ and $\varepsilon \leq 1$, the derivative of $c^2 - \varepsilon + \frac{\varepsilon^2}{4c^2}$ is positive. Therefore

$$c^2 - \varepsilon + \frac{\varepsilon^2}{4c^2} > c^2 - 8/9$$

and thus Equation (23) is satisfied if $c^2 > 2\ln(1.25/\delta)$. □

A.4 Proof of Theorem 4.1 and Corollary 4.2

Now we prove Theorem 4.1.

Theorem 4.1. *Let $\mathcal{E} = (\mathcal{W}, O, \mathcal{V}_{val})$ be an execution of Algorithm 2 where $\mathcal{D}$ is a Valid (c, Z)-Gaussian and $\mathcal{V}_{val}$ be defined by the observations of the adversary (respectively by the corruption of a set C of parties). Let B, A be derived from $\mathcal{E}$ as described in Equation* (10). *Let Σ_η be the covariance matrix of noise terms $\begin{pmatrix} \eta^{\star H} \\ \eta^H_{(:)} \end{pmatrix}$, which have positive variance. Then, execution $\mathcal{E}$ is $(\varepsilon, \delta, \mathcal{V}_{val})$-E-DP (respectively (ε, δ, C)-C-DP) if all noise terms have positive variance and*

$$h^\top \Sigma^{-1} h \leq \frac{\varepsilon^2}{c^2} \qquad \text{for each column } h \text{ of } B \tag{25}$$

where $c^2 > 2\ln(1.25/\delta)$ and

$$\Sigma \coloneqq (B, A)\Sigma_\eta(B, A)^\top \in \mathbb{R}^{m\times m}. \tag{26}$$

Proof. We start our analysis from Equation (21). Matrices A, B and x^H are fixed, then honest parties draw $\eta^{\star H}$, $\eta^H_{(:)}$ and the adversary observes $y_{\mathcal{V}}$. Let $\mathrm{x}^{(A)}$ and $\mathrm{x}^{(B)}$ be two possible values of x^H that are neighboring as precised in Definition 2.1. We prove that

$$\Pr\left(y_{\mathcal{V}} \mid \mathrm{x}^{(A)}\right) \leq \varepsilon \Pr\left(y_{\mathcal{V}} \mid \mathrm{x}^{(B)}\right) + \delta$$

for ε and δ according to Equation (25). Recalling that $\mathbb{I}[\cdot]$ is the indicator function, we have that

$$\Pr\left(y_{\mathcal{V}} \mid \mathrm{x}^H\right) = \int_{\binom{\eta^{\star H}}{\eta^H_{(:)}} \in \mathbb{X}^{n_H+k_H}} \mathbb{I}\left[B(\mathrm{x}^H + \eta^{\star H}) + A\eta^H_{(:)} = y_{\mathcal{V}}\right] \Pr\binom{\eta^{\star H}}{\eta^H_{(:)}} \mathrm{d}\binom{\eta^{\star H}}{\eta^H_{(:)}}$$

As (B, A) is full rank, then $(B, A)\binom{\eta^{\star H}}{\eta^H_{(:)}}$ covers the complete space $\mathbb{X}^m$. Recall that Σ is the covaraince matrix of $(B, A)\binom{\eta^{\star H}}{\eta^H_{(:)}}$. The probability of a certain value of $(B, A)\binom{\eta^{\star H}}{\eta^H_{(:)}} = y_{\mathcal{V}} - B\mathrm{x}^H$ is given by the Gaussian distribution

$$\Pr\left((B, A)\binom{\eta^{\star H}}{\eta^H_{(:)}} = v\right) = \frac{\exp\left(-v^\top \Sigma^{-1} v/2\right)}{\sqrt{(2\pi)^m \det(\Sigma)}}$$

For two possible values $v_A = y_{\mathcal{V}} - B\mathrm{x}^{(A)}$ and $v_B = y_{\mathcal{V}} - B\mathrm{x}^{(B)}$ we get the ratio:

$$\frac{\Pr\left((B, A)\binom{\eta^{\star H}}{\eta^H_{(:)}} = v_A\right)}{\Pr\left((B, A)\binom{\eta^{\star H}}{\eta^H_{(:)}} = v_B\right)} = \frac{\exp\left(-v_A^\top \Sigma^{-1} v_A/2\right)}{\exp\left(-v_B^\top \Sigma^{-1} v_B/2\right)} = \exp\left(-(v_A + v_B)^\top \Sigma^{-1}(v_A - v_B)/2\right).$$

Now we adapt the strategy of [29] for proving the general gaussian mechanism, computing first this ratio and showing it is bounded by e^ε with probability $1-\delta/2$ (and above $e^{-\varepsilon}$ with probability $1-\delta/2$). We have that

$$\ln\left|\frac{\Pr\left((B, A)\binom{\eta^{\star H}}{\eta^H_{(:)}} = v_A\right)}{\Pr\left((B, A)\binom{\eta^{\star H}}{\eta^H_{(:)}} = v_B\right)}\right| > \varepsilon$$
$$\iff \left|-\frac{1}{2}(v_A + v_B)^\top \Sigma^{-1}(v_A - v_B)\right| > \varepsilon. \tag{27}$$

Now note that

$$v_A - v_B = B(\mathrm{x}^{(A)} - \mathrm{x}^{(B)})$$
$$v_A + v_B = 2y_{\mathcal{V}} - B(\mathrm{x}^{(A)} + \mathrm{x}^{(B)}).$$

Without loss of generality, we assume that our datasets $\mathrm{x}^{(A)}$ and $\mathrm{x}^{(B)}$ differ on the value of one party as much as possible. Let $\Delta\mathrm{x} \coloneqq \mathrm{x}^{(A)} - \mathrm{x}^{(B)}$. As private values lie in the interval $[0, 1]$, we have that $\Delta\mathrm{x}$ has one component equal to 1 and 0 in all the other components. Equation (27) is equivalent to

$$\left|\frac{1}{2}\left(2y_{\mathcal{V}} - B(\mathrm{x}^{(A)} + \mathrm{x}^{(B)})\right)^\top \Sigma^{-1} B\Delta\mathrm{x}\right| > \varepsilon$$
$$\iff \left|\frac{1}{2}\left(2y_{\mathcal{V}} - 2B\mathrm{x}^{(A)} + B\Delta\mathrm{x}\right)^\top \Sigma^{-1} B\Delta\mathrm{x}\right| > \varepsilon$$
$$\iff \left|\frac{1}{2}\left(2\,(B, A)\binom{\eta^{\star H}}{\eta^H_{(:)}} + B\Delta\mathrm{x}\right)^\top \Sigma^{-1} B\Delta\mathrm{x}\right| > \varepsilon$$
$$\iff \left|\left((B, A)\binom{\eta^{\star H}}{\eta^H_{(:)}}\right)^\top \Sigma^{-1} B\Delta\mathrm{x} + \frac{1}{2}(B\Delta\mathrm{x})^\top \Sigma^{-1} B\Delta\mathrm{x}\right| > \varepsilon.$$

We explore the conditions where

$$\left((B, A)\binom{\eta^{\star H}}{\eta^H_{(:)}}\right)^\top \Sigma^{-1} B\Delta\mathrm{x} + \frac{1}{2}(B\Delta\mathrm{x})^\top \Sigma^{-1} B\Delta\mathrm{x} > \varepsilon$$

with probability smaller than $\delta/2$ (as said, the other side of the bound is analog). Note that

$$h \coloneqq B\Delta\mathrm{x} \in \mathbb{R}^m$$

is the ith column of B, where i is the coordinate of $\Delta\mathrm{x}$ that is equal to 1. The above is equivalent to

$$\left((B, A)\binom{\eta^{\star H}}{\eta^H_{(:)}}\right)^\top \Sigma^{-1} h > \varepsilon - \frac{1}{2} h^\top \Sigma^{-1} h.$$

To bound the probability of the above to hold, we will use the tail bound

$$\Pr(w > \gamma) \leq \frac{\sigma_w}{\gamma\sqrt{2\pi}} \exp(-\gamma^2/2\sigma_w^2) \tag{28}$$

where σ_w is the standard deviation of w. We set

$$w = \left((B, A)\binom{\eta^{\star H}}{\eta^H_{(:)}}\right)^\top \Sigma^{-1} h \tag{29}$$

$$\gamma = \varepsilon - \frac{1}{2} h^\top \Sigma^{-1} h. \tag{30}$$

and start by computing σ_w:

$$\sigma_w^2 = \mathrm{var}\left(\left((B, A)\binom{\eta^{\star H}}{\eta^H_{(:)}}\right)^\top \Sigma^{-1} h\right) = h^\top \Sigma^{-1} \mathrm{var}\left((B, A)\binom{\eta^{\star H}}{\eta^H_{(:)}}\right) \Sigma^{-1} h = h^\top \Sigma^{-1} h. \tag{31}$$

By Equation (28), proving that

$$\frac{\sigma_w}{\gamma\sqrt{2\pi}} \exp\left(\frac{-\gamma^2}{2\sigma_w^2}\right) \leq \frac{\delta}{2}$$

implies our privacy guarantee. The above is equivalent to

$$\frac{\gamma}{\sigma_w} \exp\left(\frac{\gamma^2}{2\sigma_w^2}\right) \geq \frac{2}{\delta\sqrt{2\pi}}$$

and, applying logarithms to both sides, to

$$\ln\left(\frac{\gamma}{\sigma_w}\right) + \frac{1}{2}\left(\frac{\gamma}{\sigma_w}\right)^2 \geq \ln\left(\frac{2}{\delta\sqrt{2\pi}}\right).$$

The above is implied if

$$\ln\left(\frac{\gamma}{\sigma_w}\right) \geq 0 \tag{32}$$

and

$$\frac{1}{2}\left(\frac{\gamma}{\sigma_w}\right)^2 \geq \ln\left(\frac{2}{\delta\sqrt{2\pi}}\right). \tag{33}$$

Below, we replace γ and σ_w using Equations (30) and (31). Then by noticing that Equation (32) is equivalent to $\gamma \geq \sigma_w$, we have that Equations (32) and (33) can be rewritten as

$$\varepsilon - \frac{1}{2} h^\top \Sigma^{-1} h > \sqrt{h^\top \Sigma^{-1} h} \tag{34}$$

and

$$\frac{1}{2}\frac{\left(\varepsilon - \frac{1}{2} h^\top \Sigma^{-1} h\right)^2}{h^\top \Sigma^{-1} h} \geq \ln\left(\frac{2}{\delta\sqrt{2\pi}}\right) \tag{35}$$

respectively. Finally, using Lemma A.1, equations (34) and (35) are implied for

$$h^\top \Sigma^{-1} h \le \frac{\varepsilon^2}{c^2}.$$

where $c^2 > 2\ln(1.25/\delta)$. □

Now we prove Corollary 4.2.

Corollary 4.2. *Let $\varepsilon, \delta \in (0,1)$. Let B, A be associated with an execution of Algorithm 2 be derived from $(\mathcal{W}, O, \mathcal{V}_{val})$ (or from $(\mathcal{W}, O, C)$) as defined in Theorem 4.1. Let Σ also be defined as in Theorem 4.1. Then, the execution is $(\varepsilon, \delta, \mathcal{V}_{val})$-E-DP (respectively (ε, δ, C)-C-DP) if all noise terms have positive variance and*

$$\begin{pmatrix} \Sigma & h \\ h^\top & \varepsilon^2/c^2 \end{pmatrix} \succeq 0 \tag{36}$$

for each column h of B, where $c^2 > 2\ln(1.25/\delta)$.

Proof. Given that (B, A) is full rank and Σ_η is a positive diagonal matrix, then $\Sigma \succ 0$. Given that

$$\begin{pmatrix} \Sigma & h \\ h^\top & \varepsilon^2/c^2 \end{pmatrix} \succeq 0,$$

by Schur's complement we have that $\varepsilon^2/c^2 - h^\top \Sigma^{-1} h \ge 0$ for each column h of B. This is equivalent to the condition of Theorem 4.1 to obtain (ε, δ)-DP. □

A.5 Proof of Lemma 4.3

Lemma 4.3. *Let $\varepsilon, \delta \in (0,1)$. Let $\mathcal{E} = (\mathcal{W}, O, \mathcal{V}_{val})$ be defined as in Theorem 4.1. For a pair of neighboring datasets $\mathrm{x}^{(A)}, \mathrm{x}^{(B)} \in \mathbb{X}^{n_H}$, let $\xi^\star \in \mathbb{X}^{n_H}$ and $\Delta_{(:)} \in \mathbb{X}^{k_H}$ be such that*

$$\mathcal{V}_{\mathcal{E}}\left(\mathrm{x}^{(A)} + \eta^{\star H}, \eta^H_{(:)}\right) = \mathcal{V}_{\mathcal{E}}\left(\mathrm{x}^{(B)} + \eta^{\star H} + \xi^\star, \eta^H_{(:)} + \Delta_{(:)}\right)$$

for any $\eta^{\star H}, \eta^H_{(:)}$. Recall that Σ_η is the covariance matrix of $\begin{pmatrix} \eta^{\star H} \\ \eta^H_{(:)} \end{pmatrix}$. Then, execution $\mathcal{E}$ is $(\varepsilon, \delta, \mathcal{V}_{val})$-E-DP (or (ε, δ, C)-C-DP) if for all such neighboring datasets $\mathrm{x}^{(A)}$ and $\mathrm{x}^{(B)}$

$$\left((\xi^\star)^\top, \Delta_{(:)}^\top\right) \Sigma_\eta^{-1} \begin{pmatrix} \xi^\star \\ \Delta_{(:)} \end{pmatrix} \le \frac{\varepsilon^2}{c^2}$$

where $c^2 > 2\ln(1.25/\delta)$.

Proof. It follows directly from [61, Theorem 1]. We adapt the proof to our setting. Let $\mathrm{x}^{(A)}$ and $\mathrm{x}^{(B)}$ be two neighboring datasets. Let $y_{\mathcal{V}}$ be the view of the adversary after execution $\mathcal{E}$. Let

$$S^{(A)} = \{(\eta^{\star H}, \eta^H_{(:)}) \in \mathbb{X}^{n_H + k_H} : \mathcal{V}_{\mathcal{E}}\left(\mathrm{x}^{(A)} + \eta^{\star H}, \eta^H_{(:)}\right) = y_{\mathcal{V}}\}$$

and analogously

$$S^{(B)} = \{(\eta^{\star H}, \eta^H_{(:)}) \in \mathbb{X}^{n_H + k_H} : \mathcal{V}_{\mathcal{E}}\left(\mathrm{x}^{(B)} + \eta^{\star H}, \eta^H_{(:)}\right) = y_{\mathcal{V}}\}.$$

Let $t = (\xi^\star, \Delta_{(:)})$. The precondition of the lemma implies that $S^{(A)} + t = S^{(B)}$. Our (ε, δ)-differential privacy guarantee given by

$$\Pr(y_{\mathcal{V}} | \mathrm{x}^{(A)}) \le e^\varepsilon \Pr(y_{\mathcal{V}} | \mathrm{x}^{(B)}) + \delta$$

is implied if

$$\Pr((\eta^{\star H}, \eta^H_{(:)})) \le e^\varepsilon \Pr((\eta^{\star H}, \eta^H_{(:)}) + t) + \delta \tag{37}$$

Indeed if Equation (37) holds we have

$$\begin{aligned}
\Pr(y_{\mathcal{V}} | \mathrm{x}^{(A)}) &= \int_{(\eta^{\star H}, \eta^H_{(:)}) \in S^{(A)}} \Pr((\eta^{\star H}, \eta^H_{(:)})) \mathrm{d}\eta^{\star H} \mathrm{d}\eta^H_{(:)} \\
&\le \int_{(\eta^{\star H}, \eta^H_{(:)}) \in S^{(A)}} (e^\varepsilon \Pr((\eta^{\star H}, \eta^H_{(:)}) + t) + \delta) \mathrm{d}\eta^{\star H} \mathrm{d}\eta^H_{(:)} \\
&= \int_{(\eta^{\star H}, \eta^H_{(:)}) - t \in S^{(A)}} (e^\varepsilon \Pr((\eta^{\star H}, \eta^H_{(:)})) + \delta) \mathrm{d}\eta^{\star H} \mathrm{d}\eta^H_{(:)} \\
&= \int_{(\eta^{\star H}, \eta^H_{(:)}) \in S^{(B)}} (e^\varepsilon \Pr((\eta^{\star H}, \eta^H_{(:)})) + \delta) \mathrm{d}\eta^{\star H} \mathrm{d}\eta^H_{(:)} \\
&= e^\varepsilon \Pr(y_{\mathcal{V}} | \mathrm{x}^{(B)}) + \delta.
\end{aligned}$$

Therefore, it suffices to prove Equation (37) to prove our (ε, δ)-DP guarantee.

We will do that by proving that $\Pr((\eta^{\star H}, \eta^H_{(:)})) \le e^\varepsilon \Pr((\eta^{\star H}, \eta^H_{(:)}) + t)$ with probability at least $1 - \delta$. Denoting $\gamma = (\eta^{\star H}, \eta^H_{(:)})$ for convenience, we need to prove that with probability $1 - \delta$ it holds that

$$|\log(\Pr(\gamma)/\Pr(\gamma + t))| \le \varepsilon.$$

We have that

$$\begin{aligned}
\left|\log \frac{\Pr(\gamma)}{\Pr(\gamma + t)}\right| &= \left| -\frac{1}{2}\gamma^\top \Sigma_\eta^{-1} \gamma + \frac{1}{2}(\gamma + t)^\top \Sigma_\eta^{-1}(\gamma + t) \right| \\
&= \left| \frac{1}{2}(2\gamma + t)^\top \Sigma_\eta^{-1} t \right|.
\end{aligned}$$

To ensure that $|\log(\Pr(\gamma)/\Pr(\gamma + t))| \le e^\varepsilon$ holds with probability at least $1 - \delta$, since we are interested in the absolute value, we show that

$$\Pr\left(\frac{1}{2}(2\gamma + t)^\top \Sigma_\eta^{-1} t \ge \varepsilon\right) \le \delta/2,$$

the proof of the other direction is analogous. This is equivalent to

$$\Pr(\gamma \Sigma_\eta^{-1} t \ge \varepsilon - t^\top \Sigma_\eta^{-1} t/2) \le \delta/2. \tag{38}$$

The variance of $\gamma \Sigma_\eta^{-1} t$ is

$$\begin{aligned}
\mathrm{var}(\gamma \Sigma_\eta^{-1} t) &= t^\top \Sigma_\eta^{-\top} \mathrm{var}(\gamma) \Sigma_\eta^{-1} t \\
&= t^\top \Sigma_\eta^{-\top} \Sigma_\eta \Sigma_\eta^{-1} t \\
&= t^\top \Sigma_\eta^{-1} t.
\end{aligned}$$

For any centered Gaussian random variable Y with variance σ_Y^2, we have that

$$\Pr(Y \ge \lambda) \le \frac{\sigma_Y}{\lambda\sqrt{2\pi}} \exp\left(-\lambda^2/2\sigma_Y^2\right). \tag{39}$$

Let $Y = \gamma \Sigma_\eta^{-1} t$, $\sigma_Y^2 = t^\top \Sigma_\eta^{-1} t$ and $\lambda = \varepsilon - t^\top \Sigma_\eta^{-1} t/2$, then satisfying

$$\frac{\sigma_Y}{\lambda\sqrt{2\pi}} \exp\left(-\lambda^2/2\sigma_Y^2\right) \le \delta/2 \tag{40}$$

implies Equation (38). Equation (40) is equivalent to

$$\frac{\lambda}{\sigma_Y} \exp\left(\lambda^2/2\sigma_Y^2\right) \ge 2/\delta\sqrt{2\pi},$$

or, after taking logarithms on both sides, to

$$\log\left(\frac{\lambda}{\sigma_Y}\right) + \frac{1}{2}\left(\frac{\lambda}{\sigma_Y}\right)^2 \ge \log\left(\frac{2}{\delta\sqrt{2\pi}}\right).$$

To make this inequality hold, we require

$$\log\left(\frac{\lambda}{\sigma_Y}\right) \ge 0 \tag{41}$$

and

$$\frac{1}{2}\left(\frac{\lambda}{\sigma_Y}\right)^2 \geq \log\left(\frac{2}{\delta\sqrt{2\pi}}\right). \tag{42}$$

Equation (41) is equivalent to $\lambda \geq \sigma_Y$. Substituting λ and σ_Y we get

$$\varepsilon - t^\top \Sigma_\eta^{-1} t/2 \geq (t^\top \Sigma_\eta^{-1} t)^{1/2}. \tag{43}$$

Substituting λ and σ_Y in Equation (42) gives

$$\frac{1}{2}\frac{(\varepsilon - \frac{1}{2}t^\top \Sigma_\eta^{-1} t)^2}{t^\top \Sigma_\eta^{-1} t} \geq \ln\left(\frac{2}{\delta\sqrt{2\pi}}\right) \tag{44}$$

By Lemma A.1, equations (43) and (44) are satisfied for $t^\top \Sigma_\eta^{-1} t \leq \frac{\varepsilon^2}{c^2}$ for $c^2 = 2\ln(1.25/\delta)$.

□

A.6 Proof of Lemma 4.4

Lemma 4.4. *Let $\mathcal{E} = (\mathcal{W}, O, \mathcal{V}_{val})$ be defined as in Theorem 4.1. Then if $y_i^{(t)}$ is not observed by the adversary (i.e., $(i,t) \in P^H \times [0, T-1] \setminus \mathcal{V}_{val}$) and $i \in O^{(t)}$, then there exists $a^{(i,t)} \in \mathbb{X}^n$ and $G^{(i,t)} \in \mathbb{X}^{n\times T}$ such that for any $\beta \in \mathbb{R}$ we have*

$$\mathcal{V}_{\mathcal{E}}\left(\tilde{\mathrm{x}}^H, \eta^H\right) = \mathcal{V}_{\mathcal{E}}\left(\tilde{\mathrm{x}}^H + \beta a^{(i,t)}, \eta^H - \beta G^{(i,t)}\right).$$

Moreover, $a_i^{(i,t)} = \frac{W^O_{t+1;i,i}-1}{w_i}$ and $a_j^{(i,t)} = \frac{W^O_{t+1;j,i}}{w_j}$ for all $j \in P^H \setminus \{i\}$, where $w \in \mathbb{R}^n$ is defined as in Equation (9).

Proof. For clarity, within the proof we ignore the superscript H of x^H, $\eta^{\star H}$ and η^H and refer to them as x, $\eta^\star$ and η.

Let $\tilde{\mathrm{x}} = \mathrm{x} + \eta^\star$. We want to prove that for any $\beta \in \mathbb{R}$,

$$\mathcal{V}_{\mathcal{E}}(\tilde{\mathrm{x}}, \eta) = \mathcal{V}_{\mathcal{E}}(\tilde{\mathrm{x}}', \eta') \tag{45}$$

where $\tilde{\mathrm{x}}' = \tilde{\mathrm{x}} + \beta a^{(i,t)}$ and $\eta' = \eta - \beta G^{(i,t)}$ for some $a^{(i,t)}$ and $G^{(i,t)}$. To achieve this, we will set $\tilde{\mathrm{x}}'_i = \tilde{\mathrm{x}}_i + \beta$ and compute the values of $\tilde{\mathrm{x}}'$ and η' such that the view $\mathcal{V}$ remains unchanged.

Let $Y^{(y)} \in \mathbb{X}^{(T+1)\times n}$ be

$$Y^{(y)}_{t',:} := \begin{cases} 0 & \text{for } t' = 0 \\ \left(W^O_{t'} y^{(t'-1)}\right)^\top & \text{for } t' \in [1, T]. \end{cases}$$

The vector $y_j^{(:)} \in \mathbb{X}^{T+1}$ of messages of any party $j \in P$ follows

$$Y^{(y)}_{:,j} + c^{(j)}\tilde{\mathrm{x}}_j + Z^{(j)}\eta^\top_{j,:} = y_j^{(:)}. \tag{46}$$

Let $y' = (y_j^{(t')\prime})^\top_{i\in P, t'\in[0,T]}$ be the vector of messages of all parties with input $\tilde{\mathrm{x}}', \eta'$. For all $j \in P$ let

$$O_j = \{t' \in [T] : j \in O^{(t')}\}$$

be the set of active iterations of a party. We are assuming that the only message that is not visible to the adversary is $y_i^{(t)}$. Therefore Equation (45) is implied if

$$y_i^{(t')\prime} = y_i^{(t')} \quad \text{for all } t' \in O_i \setminus \{t\} \tag{47}$$

and

$$y_j^{(t')\prime} = y_j^{(t')}, \quad \text{for all } j \in P \setminus \{i\}, t' \in O_j, \tag{48}$$

which means that all messages remain unchanged except for $y_i^{(t)}$ when the input changes from $\tilde{\mathrm{x}}, \eta$ to $\tilde{\mathrm{x}}', \eta'$.

Temporary Dropouts. We first analyze the case where there are no permanent dropouts. We will start by computing $y_i^{(t)\prime} - y_i^{(t)}$, which we will use for the remainder of the proof. By Equation (47), have that

$$\sum_{t'\in O_i} y_i^{(t')\prime} - \sum_{t'\in O_i} y_i^{(t')} = y_i^{(t)\prime} - y_i^{(t)}. \tag{49}$$

By equations (6) and (7) we have that for all $j \in P$

$$\sum_{t\in O_j} y_j^{(t')} = \sum_{t'\in O_j\setminus\{0\}} W^O_{t';j,:} y^{(t'-1)} + \sum_{t'\in O_j} c^{(j)}_{t'}\tilde{\mathrm{x}}_j + Z^{(j)}_{t',:}(\eta_{j,:})^\top. \tag{50}$$

By definition of $\mathcal{D}^O$ and Equation (8), we know that

$$\sum_{t'\in[0,T]} c^{(j)}_{t'}\tilde{\mathrm{x}}_j + Z^{(j)}_{t',:}(\eta_{j,:})^\top = \sum_{t'\in O_j} c^{(j)}_{t'}\tilde{\mathrm{x}}_j + Z^{(j)}_{t',:}(\eta_{j,:})^\top = w_j\tilde{\mathrm{x}}_j$$

when there are only temporary dropouts. Then Equation (50) becomes equivalent to

$$\sum_{t\in O_j} y_j^{(t')} = \left(\sum_{t'\in O_j\setminus\{0\}} W^O_{t';j,:} y^{(t'-1)}\right) + w_j\tilde{\mathrm{x}}_j \tag{51}$$

for all $j \in P$. Then by equations (51), (47) and (48) we also have that

$$\sum_{t'\in O_j} y_j^{(t')\prime} - \sum_{t'\in O_j} y_j^{(t')} = W^O_{t+1;j,i}(y_i^{(t)\prime} - y_i^{(t)}) + w_j(\tilde{\mathrm{x}}'_j - \tilde{\mathrm{x}}_j) \tag{52}$$

for all $j \in P$. By combining the above with Equation (49) and since $\beta = \tilde{\mathrm{x}}'_i - \tilde{\mathrm{x}}_i$ have that

$$y_i^{(t)\prime} - y_i^{(t)} = W^O_{t+1;i,i}(y_i^{(t)\prime} - y_i^{(t)}) + w_i\beta$$

The above is equivalent to

$$y_i^{(t)\prime} - y_i^{(t)} = \frac{w_i\beta}{1 - W^O_{t+1;i,i}} \tag{53}$$

which is the quantity we wanted to compute as a first step of our proof.

Now we will compute $\eta'_{i,:} - \eta_{i,:}$. Using equations (46) and (47), we have that for all $t' \in O_i \setminus \{t\}$:

$$\begin{aligned} 0 &= y_i^{(t')\prime} - y_i^{(t')} \\ &= W^O_{t';i,:}(y^{(t'-1)\prime} - y^{(t'-1)}) + c^{(i)}_{t'}(\tilde{\mathrm{x}}'_i - \tilde{\mathrm{x}}_i) + Z^{(i)}_{t',:}(\eta'_{i,:} - \eta_{i,:})^\top \end{aligned} \tag{54}$$

The above is equivalent to

$$Z^{(i)}_{t',:}(\eta'_{i,:} - \eta_{i,:})^\top = -c^{(i)}\beta - W^O_{t';i,:}(y^{(t'-1)\prime} - y^{(t'-1)}) \tag{55}$$

for all $t' \in O_i \setminus \{t\}$.

Let $\hat{t}$ be the smallest iteration of O_i that is greater than t (it always exists, as $t < T$ and $T \in O_i$). We have that

$$W^O_{t';i,:}(y^{(t'-1)\prime} - y^{(t'-1)}) = 0$$

for all $t' \in O_i \setminus \{t, \hat{t}\}$. If i dropped out from iterations $t+1$ to $\hat{t}$, $y^{(t')} = y^{(t+1)}$ for all $t' \in [t+1, \hat{t}]$, and no extra noise terms and

private values were added due to inactivity. Therefore, by Equation (53) we have

$$W^{O}_{\hat{t};i,:}(y^{(\hat{t}-1)\prime} - y^{(\hat{t}-1)}) = W^{O}_{t+1;i,i}(y_i^{(t)\prime} - y_i^{(t)}) = \frac{\beta w_i W^{O}_{t+1;i,i}}{1 - W^{O}_{t+1;i,i}}.$$

From the above and Equation (55) we have that

$$Z^{(i)}_{t',:}(\eta'_{i,:} - \eta_{i,:})^\top = \beta h_{t'} \qquad \text{for all } t' \in O_i \setminus \{t\} \tag{56}$$

where $h_{t'} = -c^{(i)}$ for $t' \in O_i \setminus \{t, \hat{t}\}$ and

$$h_{\hat{t}} = -c^{(i)} - \frac{w_i W^{O}_{t+1;i,i}}{1 - W^{O}_{t+1;i,i}}.$$

Given that $\mathcal{D}^O$ is Valid, we have that $\hat{Z} = (Z^{(i)}_{t',:})_{t' \in O_i \setminus \{t\}}$ is full row rank. Hence, there exist $v \in \mathbb{X}^T$ that satisfies

$$\hat{Z}v = (h_{t'})_{t' \in O_i \setminus \{t\}}.$$

Setting $(\eta'_{i,:} - \eta_{i,:})^\top = \beta v$ satisfies Equation (56).

We now compute $\tilde{\mathrm{x}}'_j$ and $\eta'_{j,:}$ for $j \in P \setminus \{i\}$. Similarly to Equation (54), by equations (46) and (48), we have that

$$\begin{aligned} 0 &= y_j^{(t')\prime} - y_j^{(t')} \\ &= W^{O}_{t';j,:}(y^{(t'-1)\prime} - y^{(t'-1)}) + c^{(j)}_{t'}(\tilde{\mathrm{x}}'_j - \tilde{\mathrm{x}}_j) + Z^{(j)}_{t',:}(\eta'_{j,:} - \eta_{j,:})^\top. \end{aligned} \tag{57}$$

for all $t' \in O_j$.

Let $k_{t'} = -W^{O}_{t';j,:}(y^{(t'-1)\prime} - y^{(t'-1)})$ for all $t' \in O_j$. Let $\tilde{t}$ be the earliest iteration of O_j that is bigger than t (it always exist, as before, as $T \in O_j$). $k_{t'}$ is equal to 0 for $t' \in O_j \setminus \{\tilde{t}\}$ and by Equation (53)

$$k_{\tilde{t}} = W^{O}_{t+1;j,i}(y_i^{(t)} - y_i^{(t)\prime}) = -\frac{\beta w_i W^{O}_{t+1;j,i}}{1 - W^{O}_{t+1;i,i}}.$$

For $\tilde{Z} = (c^{(j)}_{t'}, Z^{(j)}_{t',:})_{t \in O_j}$, Equation (57) is equivalent to

$$\tilde{Z}\begin{pmatrix} \tilde{\mathrm{x}}'_j - \tilde{\mathrm{x}}_j \\ \eta'_{j,:} - \eta_{j,:} \end{pmatrix} = \beta(k_{t'})_{t' \in O_j}. \tag{58}$$

As $\mathcal{D}^O$ is a Valid Gaussian, we have that $\tilde{Z}$ full row rank. Therefore there exist $v \in \mathbb{X}^{T+1}$ such that $\tilde{Z}v = (k_{t'})_{t' \in [0,T]}$. Setting

$$\begin{pmatrix} \tilde{\mathrm{x}}'_j - \tilde{\mathrm{x}}_j \\ \eta'_{j,:} - \eta_{j,:} \end{pmatrix} = \beta v$$

satisfies Equation (58). Then, setting $a^{(i,t)} = \tilde{\mathrm{x}}' - \tilde{\mathrm{x}}$ and $G^{(i,t)} = \eta' - \eta$ finishes the first part of our claim.

We now prove the remaining part of the claim, which is that there exist $\hat{a}_i^{(i,t)}$ such that

$$\hat{a}_i^{(i,t)} = (W^{O}_{t;i,i} - 1)/w_i$$

and

$$\hat{a}_j^{(i,t)} = W^{O}_{t;j,i}/w_j \quad \text{for all } j \in P^H \setminus \{j\}$$

satisfies the lemma.

For $K \in \mathbb{R}$, any re-scaling $(Ka^{(i,t)}, KG^{(i,t)})$ of $(a^{(i,t)}, G^{(i,t)})$ satisfies the lemma. We will prove the claim for a re-scaling $\hat{a}^{(i,t)}$ of $a^{(i,t)}$.

For all $j \in P^H \setminus \{i\}$, by equations (52) and (48) we have that

$$\begin{aligned} 0 &= \sum_{t' \in O_j} y_j^{(t')\prime} - \sum_{t' \in O_j} y_j^{(t')} \\ &= W^{O}_{t+1;j,i}(y_i^{(t)\prime} - y_i^{(t)}) + w_j(\tilde{\mathrm{x}}'_j - \tilde{\mathrm{x}}_j) \\ \text{(by Eq. (53))} &= W^{O}_{t+1;j,i}\frac{w_i\beta}{1 - W^{O}_{t+1;i,i}} + w_j(\tilde{\mathrm{x}}'_j - \tilde{\mathrm{x}}_j). \end{aligned} \tag{59}$$

The above is equivalent to

$$\tilde{\mathrm{x}}'_j - \tilde{\mathrm{x}}_j = \beta\frac{w_i}{W^{O}_{t+1;i,i} - 1}\frac{W^{O}_{t+1;j,i}}{w_j} = \beta a_j^{(i,t)} \tag{60}$$

for all $j \in P^H \setminus \{i\}$. Let $K = \frac{W^{O}_{t+1;i,i}-1}{w_i}$ and $\hat{a}^{(i,t)} = Ka^{(i,t)}$. We deduce that $\hat{a}^{(i,t)}$ satisfies the lemma as it is a re-scaling of $a^{(i,t)}$. Recalling that $\tilde{\mathrm{x}}'_i - \tilde{\mathrm{x}}_i = \beta$ we know that $a_i^{(i,t)} = 1$ and hence

$$\hat{a}_i^{(i,t)} = Ka_i^{(i,t)} = \frac{W^{O}_{t+1;i,i} - 1}{w_i}.$$

By equation (60) we have that

$$\hat{a}_j^{(i,t)} = Ka_j^{(i,t)} = \frac{W^{O}_{t+1;j,i}}{w_j}.$$

for all $j \in P^H \setminus \{i\}$. The last two equations are what we wanted to prove.

Permanent Dropouts. For simplicity, in the case of permanent dropouts we will assume that for each permanently dropped party $i' \in P \setminus O^{(T)}$ the adversary knows a special message:

$$y_{i'}^{(T)} = \sum_{j \in P} W^{O}_{T;i',j} y_j^{(i')} + \sum_{k=1}^{T} \bar{Z}^{(i')}_{T,k}\eta_{i',k} \tag{61}$$

where

$$\bar{Z}^{(i')}_{T,k} = -\sum_{t'=0}^{T-1} \bar{Z}^{(i')}_{t',k} \quad \text{for all } k \in [1, T].$$

The special message of Equation (61) cancels remaining noise that party i' didn't cancel due to dropout. Note that in the original setting, the adversary does not know $y_{i'}^{(T)}$ as party i' dropped out. Therefore if we prove that there exist $a^{(i,t)}$ and $G^{(i,t)}$ that satisfy the lemma with the extra knowledge of Equation (61) then this will also hold with the original knowledge of the adversary where these special messages are unknown. Then, for each party $i' \in P \setminus O^{(T)}$ we define $\bar{Z}^{(i')} \in \mathbb{R}^{(T+1)\times T}$ such that $\bar{Z}^{(i')}_{t,:} = Z^{(i')}_{t,:}$ for all $t \in [0, T-1]$ and $\bar{Z}^{(i')}_{T,:}$ defined as above.

Running Algorithm 2 with distributions

$$\mathcal{D}^O(i',:) = \mathcal{D}^{(c^{(i')},\bar{Z}^{(i')})}_{Gauss}(\mathrm{x}_{i'} + \eta^{\star}_{i'})$$

for all $i' \in P \setminus O^{(T)}$ is equivalent to an execution without permanent dropouts. Hence we can compute $a^{(i,t)}$ and $G^{(i,t)}$ as in the temporary dropout case. Note that the value of $a^{(i,t)}$ only depends on $W_{t+1;:,i}$ and w which only depends on $(c^{(i')})_{i' \in P}$. Therefore its value is not modified by the changes introduced by $\bar{Z}^{(i')}$ for all $i' \in P$. □

A.7 Proofs of Theorems 4.5, 4.6 and 4.7

In this appendix, we prove theorems 4.5, 4.6 and 4.7. We start by proving Lemma A.2. Then we prove Theorem 4.7 and show that theorems 4.5 and 4.6 are a special case of Theorem 4.7.

Lemma A.2. *Let $\mathcal{E} = (\mathcal{W}, \mathcal{O}, \mathcal{V}_{val})$ be defined as in Theorem 4.1. Let J, w, $\mathcal{H}$, $\{a^{(i,t)}\}_{(i,t)\in\mathcal{H}}$ be as defined in Theorem 4.7. If $\{a^{(i,t)}\}_{(i,t)\in\mathcal{H}}$ has at least $|J|-1$ independent vectors, then for any x, $\eta^\star$, x' and $(\eta^\star)'$ such that*

$$\sum_{i\in J} w_i(\mathrm{x}_i + \eta_i^\star) = \sum_{i\in J} w_i(\mathrm{x}'_i + (\eta_i^\star)').$$

there exist $\Delta \in \mathbb{X}^{n_H \times T}$ such that

$$\mathcal{V}_{\mathcal{E}}\left(\mathrm{x} + \eta^\star, \eta\right) = \mathcal{V}_{\mathcal{E}}\left(\mathrm{x}' + (\eta^\star)', \eta + \Delta\right)$$

for any η.

Proof. We first define

$$\mathcal{K}^{(\tilde{\mathrm{x}})} = \left\{ \tilde{\mathrm{x}}' \in \mathbb{X}^{n_H} : \sum_{i\in J} w_i \tilde{\mathrm{x}}'_i = \sum_{i\in J} w_i \tilde{\mathrm{x}}_i \right\}.$$

and

$$\mathcal{H}^{(\tilde{\mathrm{x}})} = \left\{ \tilde{\mathrm{x}} + \sum_{(i,t)\in\mathcal{H}} \beta_{(i,t)} a^{(i,t)} \in \mathbb{X}^{n_H} : \beta_{(i,t)} \in \mathbb{R} \quad \forall (i,t) \in \mathcal{H} \right\}.$$

By definition, $W^{\mathcal{O}}_{t+1;j,i} \neq 0$ only if $j \in N^{(t)}_{i\rightarrow}$ or if $j = i$. By Equation (3) and since $N^{(t)}_{i\rightarrow} \in J$ we have that for any $(i,t) \in \mathcal{H}$,

$$\sum_{j\in J} W^{\mathcal{O}}_{t+1;j,i} = \sum_{j\in P} W^{\mathcal{O}}_{t+1;j,i} = 1. \tag{62}$$

Then for any $(i,t) \in \mathcal{H}$, $\tilde{\mathrm{x}} \in \mathbb{X}^{n_H}$ and $\beta \in \mathbb{R}$,

$$\begin{aligned}
\sum_{j\in J} w_j(\tilde{\mathrm{x}}_j + \beta a^{(i,t)}_j) &= \sum_{j\in J} w_j \tilde{\mathrm{x}}_j + \sum_{j\in J} w_j a^{(i,t)}_j \\
&= \sum_{j\in J} w_j \tilde{\mathrm{x}}_j + w_i a^{(i,t)}_i + \sum_{j\in J\setminus\{i\}} a^{(i,t)}_j \\
\text{(by Lemma 4.4)} &= \sum_{j\in J} w_j \tilde{\mathrm{x}}_j + w_i a^{(i,t)}_i + \sum_{j\in J\setminus\{i\}} w_j \frac{W^{\mathcal{O}}_{t+1;j,i}}{w_j} \\
&= \sum_{j\in J} w_j \tilde{\mathrm{x}}_j + w_i \frac{W^{\mathcal{O}}_{t+1;i,i} - 1}{w_i} + \sum_{j\in J\setminus\{i\}} W^{\mathcal{O}}_{t+1;j,i} \\
&= \sum_{j\in J} w_j \tilde{\mathrm{x}}_j - 1 + \sum_{j\in J} W^{\mathcal{O}}_{t+1;j,i} \\
\text{(by Eq. (62))} &= \sum_{j\in J} w_j \tilde{\mathrm{x}}_j.
\end{aligned}$$

Therefore $\tilde{\mathrm{x}} + \beta a^{(i,t)} \in \mathcal{K}^{(\tilde{\mathrm{x}})}$. This implies that $\mathcal{H}^{(\tilde{\mathrm{x}})} \subseteq \mathcal{K}^{(\tilde{\mathrm{x}})}$. We know that $\mathcal{H}^{(\tilde{\mathrm{x}})}$ has dimension $|J|-1$. Since $\mathcal{K}^{(\tilde{\mathrm{x}})}$ also has dimension $|J|-1$, it must be that $\mathcal{H}^{(\tilde{\mathrm{x}})} = \mathcal{K}^{(\tilde{\mathrm{x}})}$.

This means that for any $\tilde{\mathrm{x}}' \in \mathcal{K}^{(\tilde{\mathrm{x}})}$ there exists $(\beta_{(i,t)})_{(i,t)\in\mathcal{H}} \in \mathbb{R}^{|\mathcal{H}|}$ such that

$$\tilde{\mathrm{x}} + \sum_{(i,t)\in\mathcal{H}} \beta_{(i,t)} a^{(i,t)} = \tilde{\mathrm{x}}'.$$

By successively applying Lemma 4.4 for each $(i,t) \in \mathcal{H}$ we have that

$$\mathcal{V}_{\mathcal{E}}(\tilde{\mathrm{x}}, \eta) = \mathcal{V}_{\mathcal{E}}(\tilde{\mathrm{x}}', \eta + \Delta)$$

for any η, where $\Delta = -\sum_{(i,t)\in\mathcal{H}} \beta_{(i,t)} G^{(i,t)}$ and $G^{(i,t)}$ as defined in Lemma 4.4. This completes the proof. □

Theorem 4.7. *Let $\varepsilon, \delta \in (0,1)$. Let $\mathcal{E} = (\mathcal{W}, \mathcal{O}, \mathcal{V}_{val})$ be defined as in Theorem 4.1. Let $J \subseteq P^H$ be a coalition that contain all honest parties that did not drop out permanently (i.e., $\mathcal{O}^{(T)} \cap P^H \subseteq J$). Let $w_{\mathcal{O}} = \sum_{i\in P^H} w_i$, where w is defined in Equation (9) and*

$$\mathcal{H} = \{(i,t) \in P^H \times [0,T] \setminus \mathcal{V}_{val} : i \in J \wedge N^{(t)}_{i\rightarrow} \subseteq J\}.$$

For all $(i,t) \in \mathcal{H}$, let $a^{(i,t)}$ be as defined in Lemma 4.4. If $\{a^{(i,t)}\}_{(i,t)\in\mathcal{H}}$ has at least $|J|-1$ independent vectors, then there exist $\xi^\star$ and $\Delta_{(:)}$ as defined in Lemma 4.3 and $\mathcal{E}$ satisfies (ε, δ, $\mathcal{V}_{val}$)-E-DP (or (ε, δ, C)-C-DP) for any

$$\sigma^2_\star > \frac{(|J|-1)c^2}{(w_{\mathcal{O}}-1)^2\varepsilon^2} \qquad \text{and} \qquad \sigma^2_\Delta \geq \frac{\|\Delta_{(:)}\|_2^2}{\frac{\varepsilon^2}{c^2} - \frac{(|J|-1)}{(w_{\mathcal{O}}-1)^2\sigma^2_\star}},$$

where $c^2 > 2\ln(1.25/\delta)$.

Proof. Let $\mathrm{x}^{(A)}$ and $\mathrm{x}^{(B)}$ be two neighboring datasets. Let $i \in P^H$ be the coordinate in which $\mathrm{x}^{(B)}$ and $\mathrm{x}^{(A)}$ differ. Without loss of generality, we assume $\mathrm{x}^{(B)}_i - \mathrm{x}^{(A)}_i = 1$. We split the proof into two cases, depending on whether $i \in J$ or not. We start with the case where $i \in J$.

Define $\xi^\star$ such that $\xi^\star_j = -\frac{w_i}{\|w\|_2^2} w_j$ if $j \in J$ and $\xi^\star_j = 0$ otherwise. For any $\eta^\star$, we have that

$$\sum_{j\in P^H} w_j(\mathrm{x}^{(A)}_j + \eta^\star_j) = \sum_{j\in P^H} w_j(\mathrm{x}^{(B)}_j + \eta^\star_j + \xi^\star_j).$$

By Lemma A.2, there exists Δ such that

$$\mathcal{V}_{\mathcal{E}}\left(\mathrm{x}^{(A)} + \eta^\star, \eta\right) = \mathcal{V}_{\mathcal{E}}\left(\mathrm{x}^{(B)} + \eta^\star + \xi^\star, \eta + \Delta\right). \tag{63}$$

for any η. Therefore we can apply Lemma 4.3. We have that execution $\mathcal{E}$ is (ε,δ)-DP if $\sigma^2_\star$, σ^2_Δ, $\|\xi^\star\|_2^2$, $\|\Delta\|_2^2$, ε and δ satisfy Equation (14).

We have that

$$\|\xi^\star\|_2^2 = \frac{w_i^2}{\|(w_i)_{i\in J}\|_2^2}$$

From the above, Equation (14) is implied if

$$\frac{\|\Delta\|_2^2}{\sigma^2_\Delta} \leq \frac{\varepsilon^2}{c^2} - \frac{w_i^2}{\|(w_i)_{i\in J}\|_2^2\sigma^2_\star}.$$

We know that

$$\sigma^2_\star > \frac{w^2_{max}}{\|(w_i)_{i\in J}\|_2^2}\frac{c^2}{\varepsilon^2}.$$

Then, we can deduce that

$$\frac{\varepsilon^2}{c^2} - \frac{w_i^2}{\|(w_i)_{i\in J}\|_2^2\sigma^2_\star} > 0.$$

The above is equivalent to

$$\sigma^2_\Delta \geq \frac{\|\Delta\|_2^2}{\frac{\varepsilon^2}{c^2} - \frac{w_i^2}{\|(w_i)_{i\in J}\|_2^2\sigma^2_\star}}. \tag{64}$$

In other words, with σ_Δ^2 lower bounded by the above expression execution $\mathcal{E}$ is (ε, δ)-DP as Equation (14) is satisfied.

We now prove the lemma when $i \notin J$. It must be that i dropped out permanently, which means that $Z^{(i)}_{T,:} = 0$. As $\mathcal{D}$ is a Valid (c, Z)-Gaussian, we have that $Z^{(i)}_{-T,:}$ is full row rank. The T-th row of $Z^{(i)}$ is equal to 0. Then, $Z^{(i)}$ and $Z^{(i)}_{-T,:}$ share the same solutions. Hence $Z^{(i)}$ is also full row rank.

We first prove that for any $\eta^\star$ and η there exist $\xi^\star$ and Δ such that

$$\|\xi^\star\|_2^2 \le \frac{w_i^2}{\|(w_i)_{i\in J}\|_2^2}$$

and Equation (63) is satisfied. The former requirement is met by setting

$$\xi_i^\star = \frac{w_i^2}{\|(w_j)_{j\in J}\|_2^2}$$

and $\xi_j^\star = 0$ for all $j \in P^H \setminus \{i\}$. Equation (63) holds if $y_i^{(t)}$ does not change for all $t \in [0, T]$ and the algorithm is executed with input $(\mathrm{x}^{(A)}, \eta^\star, \eta)$ or with input $(\mathrm{x}^{(B)}, \eta^\star + \xi^\star, \eta + \Delta)$. This is equivalent to proving that

$$\begin{aligned} &c^{(i)}(\mathrm{x}_i^{(A)} + \eta_i^\star) + Z^{(i)}(\eta_{i,:})^\top \\ &= c^{(i)}(\mathrm{x}_i^{(B)} + \eta_i^\star + \xi_i^\star) + Z^{(i)}(\eta_{i,:} + \Delta_{i,:})^\top. \end{aligned}$$

The above is equivalent to

$$Z^{(i)}(\Delta_{i,:})^\top = -c^{(i)}(1 + \xi_i^\star).$$

As $Z^{(i)}$ is full row rank, there exists $\Delta_{i,:}$ that satisfies such equation.

Then by setting $\Delta_{j,:} = 0$ for all $j \in P^H \setminus \{i\}$ we apply Lemma 4.3 for $\xi^\star$ and Δ as defined above. We do the same reasoning as the last part of the derivation where $i \in J$ (after Equation (63)), showing that Equation (14) is satisfied for

$$\sigma_\star^2 > \frac{w_{max}^2}{\|(w_i)_{i\in J}\|_2^2} \cdot \frac{c^2}{\varepsilon^2}$$

and σ_Δ^2 as in Equation (64). This concludes the proof. □

Theorem 4.5. *Let $\varepsilon, \delta \in (0, 1)$. Let $\mathcal{E} = (\mathcal{W}, \mathcal{O}, \mathcal{V}_{val})$ be defined as in Theorem 4.1 and associated with an execution without dropouts (i.e., $\mathcal{O}^{(t)} = P$ for all $t \in [0, T]$). Let $\mathcal{H} = P^H \times [0, T] \setminus \mathcal{V}_{val}$ be the set of pairs (i, t) such that $y_i^{(t)}$ is not seen by the adversary. For all $(i, t) \in \mathcal{H}$, let $a^{(i,t)}$ be as defined in Lemma 4.4. If $\{a^{(i,t)}\}_{(i,t)\in\mathcal{H}}$ has at least $n_H - 1$ independent vectors, there exist $\xi^\star$ and $\Delta_{(:)}$ as defined in Lemma 4.3 and $\mathcal{E}$ satisfies $(\varepsilon, \delta, \mathcal{V}_{val})$-E-DP (or (ε, δ, C)-C-DP) for any*

$$\sigma_\star^2 > \frac{c^2}{n_H \varepsilon^2} \qquad \text{and} \qquad \sigma_\Delta^2 \ge \frac{\|\Delta_{(:)}\|_2^2}{\frac{\varepsilon^2}{c^2} - \frac{1}{n_H \sigma_\star^2}},$$

where $c^2 > 2\ln(1.25/\delta)$.

Proof. If there are no dropouts, then $w_i = 1$ for all $i \in P^H$ and $\mathcal{O}^{(t)} = P$ for all $t \in [0, T]$. Then the proof is a direct application of Theorem 4.7 with $J = P^H$. □

Theorem 4.6. Let $\varepsilon, \delta \in (0, 1)$. Let $\mathcal{E} = (\mathcal{W}, \mathcal{O}, \mathcal{V}_{val})$ be defined as in Theorem 4.1. Let $w_O = \sum_{i\in P^H} w_i$, where w is defined in Equation (9). Let $\mathcal{H} = P^H \times [0, T] \setminus \mathcal{V}_{val}$ be the set of pairs (i, t) such that $y_i^{(t)}$ is not seen by the adversary. For all $(i, t) \in \mathcal{H}$, let $a^{(i,t)}$ be as defined in Lemma 4.4. If $\{a^{(i,t)}\}_{(i,t)\in\mathcal{H}}$ has at least $|n_H| - 1$ independent vectors, then there exist $\xi^\star$ and $\Delta_{(:)}$ as defined in Lemma 4.3 such that $\mathcal{E}$ satisfies $(\varepsilon, \delta, \mathcal{V}_{val})$-E-DP (or (ε, δ, C)-C-DP) for any

$$\sigma_\star^2 > \frac{(n_H - 1)c^2}{(w_O - 1)^2\varepsilon^2} \qquad \text{and} \qquad \sigma_\Delta^2 \ge \frac{\|\Delta_{(:)}\|_2^2}{\frac{\varepsilon^2}{c^2} - \frac{(n_H-1)}{(w_O-1)^2\sigma_\star^2}},$$

where $c^2 > 2\ln(1.25/\delta)$.

Proof. We have that

$$\frac{w_{max}^2}{\|w\|_2^2} \le \frac{n_H - 1}{(w_O - 1)^2}$$

for all w such that $\sum_{i\in P^H} w_i = w_O$. Then, it follows directly by applying Theorem 4.7 with $J = P^H$. □

A.8 Proof of Theorems 4.8 and 4.9

Theorem 4.8. *Let $\varepsilon, \delta \in (0, 1)$. Let $\mathcal{E} = (\mathcal{W}, \mathcal{O}, \mathcal{V}_{val})$ be defined as in Theorem 4.1, $w_O = \sum_{i\in P^H} w_i$ and G^H as defined in Equation (15). If G^H is strongly connected, then $\mathcal{E}$ satisfies $(\varepsilon, \delta, \mathcal{V}_{val})$-E-DP (or (ε, δ, C)-C-DP) for $\sigma_\star^2$ and σ_Δ^2 defined as in Theorem 4.6.*

Proof. We first construct a weighted adjacency matrix W^H of graph G^H. For each party $i \in P^H$, let

$$F^{(i)} = \left\{ W^O_{t+1;P^H,i} \in \mathbb{R}^{n_H} : (i, t) \in P^H \times [0, T-1] \setminus \mathcal{V}_{val} \right\}$$

the set whose elements are columns

$$W^O_{t+1;P^H,i} = (W^O_{t+1;j,i})_{j\in P^H} \in \mathbb{R}^{n_H},$$

which determine the weights of outgoing messages $y_i^{(t)}$ that have not been seen by the adversary.

For each party $i \in P^H$, we set

$$W^H_{:,i} = \frac{1}{|F^{(i)}|} \sum_{v\in F^{(i)}} v.$$

For all $i, j \in P^H$, we have that $W^H_{i,j} > 0$ if $(i, j) \in E(G^H)$ and $W^H_{i,j} = 0$ otherwise. Therefore, W^H is a weighted adjacency matrix of G^H. As G^H is strongly connected then W^H is irreducible. This means that for all $i, j \in P^H$, $(W^H)^k_{i,j} > 0$ for a sufficiently large integer k.

Therefore we can apply the Perron-Frobenius Theorem for non-negative irreducible matrices. This implies that the largest eigenvalue of W^H is smaller or equal to 1 and has multiplicity 1. The same applies to $(W^H)^\top$.

As columns of W^H are the average of column-stochastic matrices, then W^H is column stochastic. Therefore,

$$(W^H)^\top \mathbf{1} = \mathbf{1}.$$

This means that 1 is an eigenvalue of $(W^H)^\top$ associated to the eigenvector $\mathbf{1}$.

By the results of our application of the Perron-Frobenius Theorem, it must be that 1 is the largest eigenvalue of $(W^H)^\top$ and has multiplicity 1. This means that the nullspace of $(W^H)^\top - I$ has dimension 1, which implies that $(W^H)^\top - I$ has rank $n_H - 1$. Then

$$\bar{W} = W^H - I$$

also has rank $n_H - 1$. Let

$$F_I^{(i)} = \left\{ W^O_{t+1;P^H,i} - \mathrm{b}_i \in \mathbb{R}^{n_H} : (i, t) \in P^H \times [0, T-1] \setminus \mathcal{V}_{val} \right\}.$$

We can deduce that

$$F_I = \bigcup_{i \in P^H} F_I^{(i)} = \left\{ w_{P^H}^\top a_{P^H}^{(i,t)} : (i,t) \in P^H \times [0, T-1] \setminus \mathcal{V}_{val} \right\}$$

where w is defined in Equation (9), $a^{(i,t)}$ in Lemma 4.4,

$$w_{P^H} = (w_j)_{j \in P^H} \in \mathbb{R}^{n_H}$$

and

$$a_{P^H}^{(i,t)} = (a_j^{(i,t)})_{j \in P^H} \in \mathbb{R}^{n_H}.$$

We can also deduce that for each $i \in P^H$

$$\bar{W}_{:,i} = \frac{1}{|F_I^{(i)}|} \sum_{v \in F_I^{(i)}} v.$$

Given that $\bar{W}$ has rank $n_H - 1$, F_I contains at least $n_H - 1$ linearly independent vectors. F_I can be obtained by multiplying each vector of

$$\hat{F}_I = \left\{ a_{P^H}^{(i,t)} : (i,t) \in P^H \times [0, T-1] \setminus \mathcal{V}_{val} \right\}$$

by $w_{P^H}^\top$. Therefore, $\hat{F}_I$ also has $n_H - 1$ linearly independent vectors. For all $(i,t) \in P^H \times [0, T-1] \setminus \mathcal{V}_{val}$, we have that the vector $a_j^{(i,t)} = 0$ for all $j \in C$. This means that we can obtain

$$\tilde{F}_I = \left\{ a^{(i,t)} : (i,t) \in P^H \times [0, T-1] \setminus \mathcal{V}_{val} \right\}$$

by adding 0-valued entries to each vector $\hat{F}_I$ in the same indexes. Therefore, the number of linearly independent vectors in $\tilde{F}_I$ is the same as that of $\hat{F}_I$. Given that $\tilde{F}_I$ has $n_H - 1$ linearly independent vectors, we can apply Theorem 4.6, which gives the required bound on $\sigma_\star^2$ to achieve (ε, δ)-DP, for sufficiently large σ_Δ^2. □

Now we prove Theorem 4.9.

Theorem 4.9. *Let $\mathcal{E} = (\mathcal{W}, O, \mathcal{V}_{val})$ be defined as in Theorem 4.1 Let $\mathcal{H} = P^H \times [0, T] \setminus \mathcal{V}_{val}$ be the set of pairs (i,t) such that $y_i^{(t)}$ is not seen by the adversary. For all $(i,t) \in \mathcal{H}$, let $a^{(i,t)}$ be as defined in Lemma 4.4. If parties do not change neighbors across iterations, (i.e., $W_t^O = W_1^O$ for all $t \in [2, T]$) and there exist $i, j \in P^H$ such that $(i,t) \in \mathcal{V}_{val}$ and $(j,t) \in \mathcal{V}_{val}$ for all $t \in [0, T]$, we have that $\{a^{(i,t)}\}_{(i,t) \in \mathcal{H}}$ has less than $n_H - 1$ independent vectors.*

Proof. According to Lemma 4.4, for each $(i,t) \in \mathcal{H}$, $a^{(i,t)}$ only depends on $W_{t+1:,i}^O$ and w. If $W_t^O = W_1^O$ for all $t \in [1, T]$, we have that for all $i \in P^H$, if $(i,t) \in \mathcal{H}$ then $a^{(i,t)} = a^{(i,0)}$ for all $t \in [0, T-1]$. In addition, from the theorem we know that there exist $i, j \in P^H$ such that $(i,t) \in \mathcal{V}_{val}$ and $(j,t) \in \mathcal{V}_{val}$ for all $t \in [0, T]$. Therefore $\{a^{(i,t)}\}_{(i,t) \in \mathcal{H}}$ is contained in $\{a^{(i,0)}\}_{i \in P^H \setminus \{i,j\}}$, which has at most $n_H - 2$ vectors. □

B Calculations for Muffliato

In this appendix, we show how the accuracy of Muffliato is computed for the experiment in Figure 1.

Essentially, for a given function $f : P \times P \to \mathbb{R}^+$ and $\alpha_M > 1$, a mechanism satisfies (α_M, f)-Pairwise Network DP (PNDP) [20, Definition 5] if for all pairs of parties $i, j \in P$, it satisfies $(\alpha_M, f(i,j))$-Renyi DP (RDP) [54] assuming the adversary has access only to party j's view.

Results of [20, Theorem 1] defines the mean privacy loss $\bar{\varepsilon}_i$ of a party i. Given that private values are within $[0, 1]$ we have that the local sensitivity is given by $\Delta_M = 1$ and for any party $i \in P^H$ we have that

$$\bar{\varepsilon}_i = \frac{\alpha_M T d_i}{2n\sigma_M^2} \tag{65}$$

by [20, Equation (7)], where d_i is the degree of party i, α_M is the desired parameter α of Renyi DP, T is the number of iterations and σ_M^2 is the variance of the Gaussian noise that each party adds to its private value. Muffliato reports $\bar{\varepsilon} = \max_{i \in P} \bar{\varepsilon}_i$ as its measure of privacy loss.

For the sake of comparison, we convert PNDP guarantees given by $\bar{\varepsilon}$, which are essentially the mean privacy loss of a RDP guarantee, to a mean privacy loss of an (ε, δ)-DP. By [54, Proposition 3], a mechanism that satisfies $(\alpha, \bar{\varepsilon})$-RDP it also satisfies (ε, δ)-DP with

$$\varepsilon = \bar{\varepsilon} + \frac{\ln(1/\delta)}{\alpha_M - 1}. \tag{66}$$

Typically, when guarantees are satisfied for multiple pairs $(\alpha, \bar{\varepsilon})$ one uses the pairs that minimizes ε for a fixed δ. This optimizes the privacy/accuracy trade-offs. In our experiments of Figure 1, we fix $(\varepsilon, \delta) = (0.1, 10^{-5})$. Here we do the same and choose $\bar{\varepsilon}$, α_M such that it minimizes σ_M^2. By Equation (66), we set

$$\alpha_M = \frac{\log(1/\delta)}{\varepsilon - \bar{\varepsilon}} + 1. \tag{67}$$

We use the same parameters as those in Figure 1 and set Muffliato under hypercube graphs, which give the smallest $\bar{\varepsilon}$ (see [20, Figure 1a]). For $n = 2^{10} = 1024$, the degree of each node i in a hypercube is $d_i = 10$ for all $i \in P$. By plugging Equation (67) into Equation (65) we have that

$$\sigma_M^2 = \left(\frac{\ln(1/\delta)}{\varepsilon - \bar{\varepsilon}} + 1 \right) \frac{10T}{2n\bar{\varepsilon}}. \tag{68}$$

We set T according to [20, Theorem 2]. Then we have that

$$T = \left\lceil \frac{\ln(n)}{\sqrt{\lambda_2}} \right\rceil$$

when $\sigma_M^2 \geq 1$. λ_2 is the second largest eigenvalue of the gossip matrix of a hypercube of $n = 1024$ nodes and is equal to $\frac{9}{11}$. Given these parameters, we deduce that $T = 8$.

With the current parameters, σ_M^2 is always greater than 180. We ignore the gossip error by assuming that each party converged to the true average of noisy values. Therefore we approximate the MSE of Muffliato by σ_M^2/n which is at least 0.1764.

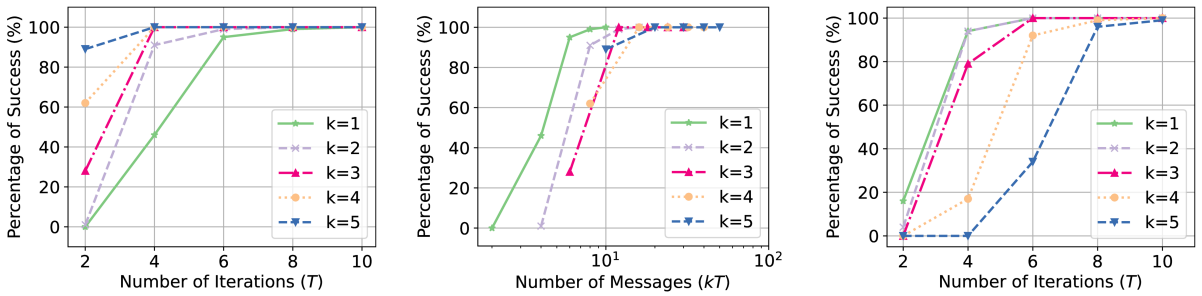
C Additional Experiments

In this appendix, we provide additional experiments related to Section 5. First, we show the impact of varying the parameter k in Appendix C.1, and then we present a comparison with secure aggregation in Appendix C.2.

C.1 Additional Experiments Varying k

We analyze the communication effort required to obtain the privacy conditions described in Section 5.1, under the experimental setup given therein.

We evaluate IncA when each party communicates with $k \in \{1, 2, 3, 4, 5\}$ neighbors per iteration. We consider $T \in \{2, 4, 6, 8, 10\}$, where 50% of messages per iteration chosen randomly are observed



(a) 50% of observed messages (b) 50% of observed messages (c) 30% of corrupted parties

Figure 5: Percentage of success in function of the number of iterations (figures 5a and 5c) and the number of messages (Figure 5a) to meet the preconditions of Theorem 4.5 for $k \in \{1, 2, 3, 4, 5\}$, where the adversary only observes messages or corrupts a subset of parties. The total number of parties is $n = 100$.

by the adversary (Eavesdrop DP) or 30% of the parties are chosen randomly to collude with the adversary (Collusion DP). For each parameter combination, we run the protocol 100 times and count how many times the preconditions of Theorem 4.5 are met.

Our results are shown for $n = 100$ in Figure 5, with the success rate as a function of the number of iterations (T) or total number of messages (kT) per party. The left and center figures show results when the adversary only observes messages (Eavesdrop DP). In Figure 5a, we can see that the higher k is, the lower is the number of iterations T required for IncA to get 100% success. However, as shown in Figure 5b, lower k requires less number of messages in total. In Figure 5c, we switch to Collusion DP, where parties are corrupted. When k is lower, it is easier to achieve higher success rate. This happens because bigger values of k reduce the size of $(a^{(i,t)})_{(i,t)\in\mathcal{H}}$ (and therefore the possibility that this set has $n_H - 1$ linearly independent vectors), as the corrupted nodes receive more messages. This results hold for any Valid (c, Z)-Gaussian $\mathcal{D}$, as when there are no dropouts, $\{a^{(i,t)}\}_{(i,t)\in\mathcal{H}}$ only depends on the communication matrices $\mathcal{W}$.

C.2 Additional Comparisons With Secure Aggregation

We show additional experiments with secure aggregation (SAgg). The experimental setup is described in Section 5.3. First, we evaluate the computation and communication cost of IncA when we vary the total number of iterations and compare it to SAgg. Second, we compare the MSE of IncA and SAgg for different proportions of parties that collude (ρ) or drop out (γ).

Varying the Number of Iterations. Figure 6 shows the computational and communication cost of IncA with $k = 1$ for a total number of iterations from the set $\{10, 20, 40, 60, 80\}$. We compare this cost with that of SAgg with $k_{SA} = 24$ (see Appendix D.3 for details on k_{SA}). For both protocols, we set the number of dimensions $d = 10^3$, $\rho = 0.1$ and $\gamma = 0.2$. We can observe that both the communication cost and runtime of IncA increase linearly with the number of iterations. The computational and communication gaps between the protocols is consistent with the observations of Figure 4 in Section 5.3.

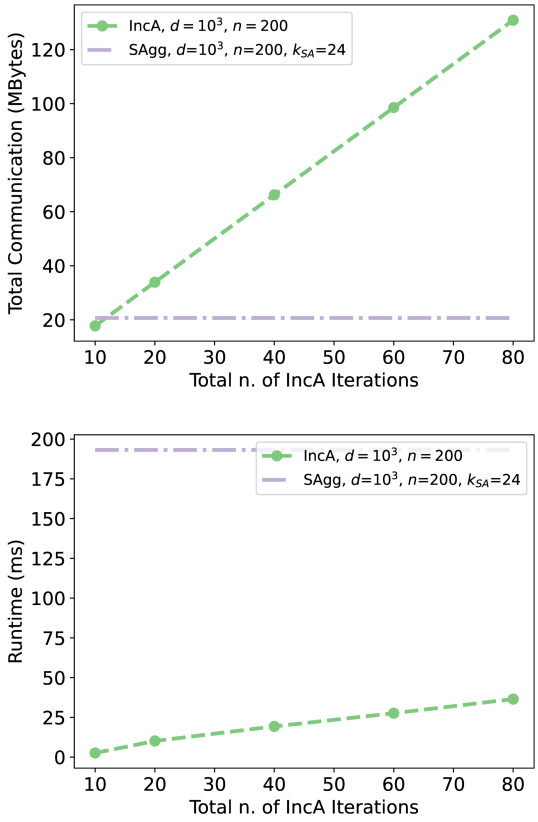


Figure 6: Computation and communication costs for IncA with respect to the total number of iterations.

Varying ρ and γ. Figure 7 shows the MSE of IncA and SAgg with $n = 200$ for $\rho \in \{0.05, 0.1, 0.15\}$ and $\gamma \in \{0.05, 0.1, 0.15, 0.2\}$. For IncA, we set $T = 40$ and $k = 1$ and consider that the final dissemination phase is done in $T_D = 20$ iterations with $k = 1$. We can see that the MSE of both protocols increases uniformly when ρ

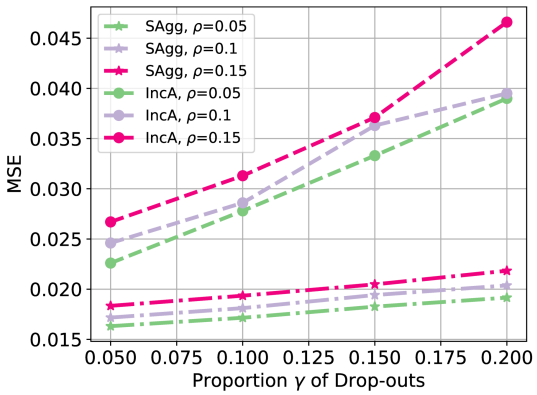


Figure 7: MSE as a function of the proportion γ of dropouts and varying the number of corrupted parties ρ for IncA and SAgg with $n = 200$. For IncA, $T = 40$ and $k = 1$. Dropouts are evenly distributed across rounds for both protocols.

or γ increase. We can see that IncA has higher MSE values, steeper growth and greater instability due to the gossip averaging in the Dissemination Phase compared to SAgg.

D Alternatives to Our Protocol

We describe in detail the most relevant alternatives to our protocol that were presented in Section 6. Central DP and Local DP are discussed in Appendix D.1, gossip protocols in Appendix D.2, and pairwise-noise approaches in Appendix D.3.

D.1 Central DP and Local DP

We first describe Central DP, then Local DP.

Central DP assumes the existence of a trusted party (or trusted curator) that does not compromise the obtained information. During a multi-party computation, each party can send their private values to the trusted curator, which will use this input to produce an outcome and add DP noise before revealing it. For mean estimation, the trusted curator computes

$$f_{CDP}(\mathsf{x}_1, \ldots, \mathsf{x}_n) = \frac{1}{n} \sum_{i \in P} \mathsf{x}_i,$$

the only information that will be exposed to the adversary and that therefore needs to be privatized. Local DP [25, 50] considers a much stronger adversary. All messages sent by any party are exposed to it, and therefore each party has individually privatize a component of

$$f_{LDP}(\mathsf{x}_1, \ldots, \mathsf{x}_n) = (\mathsf{x}_1, \ldots, \mathsf{x}_n)$$

before sharing it.

To satisfy DP, a mechanism adds an amount of noise that is proportional to the sensitivity of the outcome with respect to the change in the input. More formally, for any computation f in the domain $\mathcal{X}^n$, the ℓ_k-sensitivity $\Delta_k f$ is defined to be the maximum change between outputs on neighboring datasets. That is,

$$\Delta_k f = \max_{\substack{\mathsf{x},\mathsf{x}' \in \mathcal{X}^n \\ \mathsf{x} \sim \mathsf{x}'}} \|f(\mathsf{x}) - f(\mathsf{x}')\|_k$$

where $\mathsf{x} \sim \mathsf{x}'$ means that x and x' are neighboring datasets (see Section 2.2), and $\|\cdot\|_k$ is the ℓ_k-norm.

For the Gaussian mechanism, the noise required to privatize each coordinate of the outcome vector of f to satisfy (ε, δ)-DP must have a variance of at least

$$\sigma^2(f) = \frac{2\ln(1.25/\delta)(\Delta_2 f)^2}{\varepsilon^2} \quad \text{[30, Theorem 3.22]}.$$

If private values are in the range $[0, 1]$, the ℓ_2-sensitivity of f_{CDP} and f_{LDP} are $1/n$ and 1 respectively. This gives

$$\sigma^2(f_{LDP}) = n^2 \sigma^2(f_{CDP}). \tag{69}$$

For computing an average in Local DP, after adding noise, we get the estimate

$$\frac{1}{n} \sum_{i \in P} \left(\mathsf{x}_i + \mathcal{N}(0, \sigma^2(f_{LDP}))\right)$$

that has a variance equal to $\sigma^2(f_{LDP})/n$. In Central DP, the estimate has variance $\sigma^2(f_{CDP})$ as the noise is added right after the computation. Therefore, given Equation (69) this variance a factor of n smaller than in Local DP.

Due to the utility obtained from revealing only the final output, Central DP is considered as the best possible privacy-accuracy trade-off, whereas Local DP gives the strongest privacy guarantees.

D.2 Gossip Protocols

Gossip protocols are designed for the dissemination and computation of information in a decentralized way[10, 13, 22, 60]. They offer good scalability with the number of participants as they only require each party to communicate with a small number of neighbors. When computations do not require privacy constraints, they are already known to have good resilience to failures.

The work of [13] studies the averaging task presented in Algorithm 3. The authors show that if matrices $W_1, \ldots, W_T$ are doubly stochastic, for all $i \in P$ the convergence rate of $y_i^{(T)}$ to $\frac{1}{n}\sum_{i \in P} \tilde{\mathsf{x}}_i$ when $T \to \infty$ depends on the spectral properties of these matrices.

Muffliato, a recent approach by Cyffers et al. [20] builds on [13] and proposes differentially private gossip protocols. Muffliato executes Algorithm 3, but with $\tilde{\mathsf{x}}_i = \mathsf{x}_i + \eta_i^g$ for all $i \in P$, where x_i is the private value and $\eta_i^g \sim \mathcal{N}(0, \sigma_g^2)$ for some $\sigma_g^2 > 0$.

In our setting, Muffliato satisfies DP with the same trade-offs as Local DP. However, relaxing the assumptions on the knowledge of the adversary, these trade-offs can be improved (see Section 5.1 and Appendix B for more details).

The works of [18] and [19] have used gossip protocols for decentralized differentially private computations. Specifically, they use random walks to compute machine learning models with differential privacy guarantees.

Algorithm 3 Gossip Averaging [13]

Input: $T \in \mathbb{N}$, $\tilde{\mathbf{x}} = (\tilde{\mathbf{x}}_1, \ldots, \tilde{\mathbf{x}}_n)^\top \in \mathbb{X}^n$, $W_1, \ldots, W_T \in \mathbb{R}^{n \times n}$, T
Initialization Phase
for all $i \in P$ **do**
  $y_i^{(0)} \leftarrow \tilde{\mathbf{x}}_i$
end for
for $t \in \{1 \ldots T\}$ **do**
  for all $i \in P$ **do**
    $y_i^{(t)} \leftarrow \sum_{j \in P} W_{t;i,j} y_j^{(t-1)}$
  end for
end for

D.3 Pairwise Noise Approaches

Pairwise-noise masks has been one of the main techniques for privacy preserving aggregation in our setting. We present it below.

The core idea for differentially private averaging is described in Algorithm 4. Each party $i \in P$ communicates with a set of neighbors $N_p(i) = \{j : \{i, j\} \in E_p\}$ which define an undirected communication graph G_p. Jointly with each neighbor $j \in N_p(i)$, party i draws a pairwise noise term $p_{i,j}$. After that, one of $\{i, j\}$ adds $p_{i,j}$ and the other adds $-p_{i,j}$ to their private values. Since all pairwise-noise terms cancel each other, the value computed in the Aggregation Phase is

$$\begin{aligned}\frac{1}{n}\sum_{i\in P} \mathsf{x}_i^p &= \frac{1}{n}\sum_{i\in P} \mathsf{x}_i + \eta_i^p + \sum_{\substack{j\in N_p(i)\\ j<i}} p_{i,j} - \sum_{\substack{j\in N_p(i)\\ j>i}} p_{i,j}\\ &= \frac{1}{n}\sum_{i\in P} \mathsf{x}_i + \eta_i^p + \frac{1}{n}\sum_{\{i,j\}\in E_p} p_{i,j} - p_{i,j}\\ &= \frac{1}{n}\sum_{i\in P} \mathsf{x}_i + \eta_i^p.\end{aligned}$$

The DP noise terms $(\eta_i^p)_{i\in P}$ ensure that the final result satisfies differential privacy, while each individual term x_i is additionally protected by $(p_{i,j})_{j\in N_p(i)}$.

Algorithm 4 Pairwise Noise Aggrebation

Input: Vector of private values $\mathsf{x} \in \mathcal{X}^n$, noise distribution $\mathcal{M}_p$, communication graph $G_p = (P, E_p)$, DP noise variance σ_p^2
Randomization Phase:
for all $i \in P$ **do**
  for all j such that $\{i, j\} \in E_p$ **do**
    i and j jointly sample $p_{i,j} \sim \mathcal{M}_p$
  end for
  Sample $\eta_i^p \sim \mathcal{N}(0, \sigma_p^2)$
  Compute $\mathsf{x}_i^p = \mathsf{x}_i + \eta_i^p + \sum_{\substack{j\in N_p(i)\\ j<i}} p_{i,j} - \sum_{\substack{j\in N_p(i)\\ j>i}} p_{i,j}$
end for
Aggregation Phase:
Each party $i \in P$ reveals x_i^p and all parties jointly compute $\sum_{i\in P} \mathsf{x}_i^p / n$

Let's analyze the same setting as in Collusion DP where a set of parties is corrupted by an adversary. If the sub-graph of G_p induced by honest parties is connected, then (i) not all of a party's neighbors are corrupted by the adversary and (ii) partial sums of the set $(\mathsf{x}_i^p)_{i\in P}$ do not reveal partial sums of $(\mathsf{x}_i)_{i\in P}$. In this case, the only information revealed without the protection of terms $(p_{i,j})_{(i,j)\in E_p}$ is the final average plus $(\eta_i^p)_{i\in P}$. In that case and if $(p_{i,j})_{(i,j)\in E_p}$ has sufficiently large variance, pairwise noise protocols obtain differentially private average estimates with an MSE that only depends on the variance σ_p^2 of independent noise terms $(\eta_i^p)_{i\in P}$. These terms only need to protect the final average, therefore they can be calibrated to have in total the same variance as in Central DP, obtaining the most accurate possible result.

There is a large body of work that studies a variant of Algorithm 4. It was first proposed in [1] and subsequently studied in [31, 35, 43]. The former works mainly strengthen the resilience to failures. The work of [12] became widely adopted due to its strong security guarantees. However, this work still requires that each party to communicate with each other ($O(n)$ messages per party). [8] and [61] obtain good scalability to the number of participants by proving that $O(\log(n))$ messages per party were sufficient to achieve strong privacy guarantees.

Except for [61], $\mathcal{M}_p$ is a uniform distribution over a finite group. In this case, all noise masks are added to private values via the (modular) addition of this group. These protocols, if proven secure, are commonly known as secure aggregation (SAgg) protocols[2].

The kind of pairwise masks used by SAgg leak no information, but they are extremely vulnerable to failures. If, during the execution, parties drop out of the protocol and even a single pairwise mask remains uncanceled, then the entire computation converts into modular noise. Therefore, these approaches use Shamir's secret sharing to hide pairwise masks while providing sufficient redundancy to reconstruct uncanceled masks due to dropout, and compress pairwise noise messages to a small seed using pseudo-random generators. However, they require structured and centrally coordinated techniques to consistently reconstruct uncanceled noise and ensure a negligible probability of that a mask is not reconstructed.

In GOPA [61], all noise is Gaussian. Therefore, pairwise masks leak some information, which makes the proof of privacy guarantees more involved. However, uncanceled noise terms have a bounded impact on the estimation and simpler dropout-recovery techniques that can be performed without central coordination, are applied to repair the damage. Parties in GOPA [61] perform a roll-back of uncanceled noise terms in the Aggregation Phase when they realize their neighbors are inactive. However, if some parties that are supposed to roll back the uncanceled noise have themselves dropped out, some pairwise noise terms remain. CorDP-DME [66] also uses Gaussian pairwise noise and minimizes its variance required to obtain DP guarantees. However, the protocol achieves this by maximizing communication, requiring $O(n)$ messages per party.

More recent extensions of secure aggregation improve the robustness against malicious participants and computational complexity in [8]. However both [7, 8] offer only a narrow protection to a malicious server. Other recent works improve the performance for multiple executions of the protocol [37, 51]. This is done either at

[2] In our paper we use secure aggregation combined with DP noise in a similar way as it is done in [46]. However, secure aggregation computes the exact average and is not required to satisfy DP guarantees.

the cost of degrading scalability to the number of parties [51] or to the dimension of input vectors [37].

Parameters for Experimental Comparison. We compare our protocol with the decentralized and semi-honest variants of GOPA, CorDP-DME and SAgg. For GOPA, each party communicates with k_{PR} random neighbors at randomization phase and after that, each party performs a decentralized Aggregation Phase based in gossip averaging (Algorithm 3), performing T_{PD} iterations and each party communicating with k_{PD} neighbors per iteration. However, we only include the cost Aggregation Phase in the theoretical analysis of Appendix E. In Section 5.2, when we empirically compare GOPA and IncA, mostly considering Randomization and dropout recovery phases for GOPA and Mixing Phase for IncA. We don't include gossip Aggregation and Dissemination phases of both protocols in the empirical comparison because they are identical. Moreover, since GOPA uses correlated noise of higher variance than IncA, dropouts in this phase will be more detrimental for the former. As CorDP-DME only differs from GOPA in that $k_{PR} = n$ and does not have a dropout recovery method, we proceed similarly.

For centrally-coordinated techniques, we compare with the secure aggregation protocol of [8] as a baseline which achieves good scalability in terms of number of parties and input dimension. We consider that each party interacts with k_{SA} neighbors. In our comparison, we include the cost of Dissemination Phase for IncA.

E Discussion

In this appendix, we discuss the advantages and limitations of our approach, with emphasis on the comparison with pairwise noise approaches, the alternatives most related to IncA (see Appendix D.3 for details). We discuss the communication and computational costs, the resilience to dropout and the impact of active attackers.

Round Complexity. Our protocol requires T rounds for the Mixing Phase and T_D rounds for the Dissemination Phase. It performs $O(dk(T + T_D))$ computations per party. As shown in the experiments, the number of messages of IncA remains similar to pairwise noise approaches when $k = 1$. Acceptable values of T and T_D slightly increase with n and more significantly with the proportion of corrupted or dropped out parties. Evidence shows a logarithmic behavior of T in function of n (see Figure 2), although better utility can be obtained increasing this parameter.

Pairwise noise approaches in the decentralized setting require a constant round randomization phase where each party interacts with k_{PR} other parties and a T_{PD}-round aggregation phase which is the same as the Dissemination Phase of IncA and requires interaction with k_{PD} parties per round (see Appendix D.3 for details). Overall, these protocols have a $O(T_{PD})$ round complexity. Therefore, considering that T_D and T_{PD} have similar values due to the similarity of Dissemination and Randomization phases, pairwise noise approaches require less rounds than IncA. However, the Randomization Phase of the former requires k_{PR} interactions, while each round of IncA works the best with a single interaction per round in Dissemination Phase. With a central coordinator, pairwise noise approaches can reduce the Aggregation Phase (and therefore the overall protocol) to a constant number of rounds.

Communication Cost. Our methodology relies in incrementally mixing private vectors with noise. If private vectors have dimension d and parties communicate with $k = 1$ neighbors per round (a setting shown to be optimal under dropouts and collusion) then each party sends $T + T_D$ messages and $O(d(T + T_D))$ bits. GOPA [61], which scales well to n in the decentralized setting, requires $O(d(k_{PR}+k_{PD}T_{PD}))$ bits per client. Centrally coordinated approaches compress messages of randomization phase using small seeds and only require constant-round dissemination phase, reducing the cost to $O(d + k_{SA})$ bits per client, where k_{SA} is the number of parties each client interacts. However, they require the server to send $O(n(d + k_{SA}))$ bits. Although not proposed in [61], GOPA might also partially benefit from the seed compression of secure aggregation in the decentralized setting, removing the multiplicative factor d in the randomization phase.

Both k_{SA} and k_{PR} are logarithmic in n. From [13], terms T_D, T_{PD}, k_{PD} can also be logarithmic in n for good mixing graphs. Therefore evidence suggests that GOPA and IncA require the same communication complexity of $O(d\text{poly-log}\, n)$ bits per party and secure aggregation of $O(d + \text{poly-log}\, n)$ per client. Overall, IncA is less applicable when the number of rounds is particularly costly or parties have low bandwidth (i.e., when larger messages will have a higher impact in the overall time).

Computational Cost. IncA performs $O(d(T + T_D))$ computations per client. In the centralized setting, pairwise noise approaches require a computational work of $O(d + k_{SA})$ per client and $O(n(d + k_{SA}))$ at the server [7]. Moreover, clients are blocked during the server's computation, which results in a linear runtime for all parties. In the decentralized setting, these approaches require $O(d(k_{PR}+ k_{PD}T_{PD}))$ computations per party and both scale gracefully with n as IncA.

Dropout Distribution. Each protocol may experience a different number and distribution of dropouts. In practice, this is influenced by factors such as the runtime complexity (also discussed in [26, 38, 67]) and the ability of parties to rejoin the computation. The former depends on the number of rounds ($T + T_D$ for IncA) and their duration. The duration of a round is influenced by the number of computations and messages (k for IncA). IncA allows dropped parties to rejoin the computation. Therefore, permanent dropouts do not necessarily increase with T, as longer runtime may increase the probability of a party to rejoin. Pairwise noise approaches require only two rounds in the randomization phase, but substantially more messages per iteration than IncA and rejoining is only allowed until parties start the rollback phase.

Impact of Active Attacks. We consider that corrupted parties are passive. If they would deviate from the protocol, the utility of IncA would be compromised. However, our privacy guarantees depend on the structure of interactions between honest parties. Therefore, when the latter choose their outgoing neighbors as in our experiments of Section 5, active attacks do not compromise the privacy guarantees of our protocol.