Differential Privacy on Trust Graphs

Badih Ghazi ✉ ⬤
Google Research, Mountain View, CA, USA

Ravi Kumar ✉ ⬤
Google Research, Mountain View, CA, USA

Pasin Manurangsi ✉ ⬤
Google Research, Bangkok, Thailand

Serena Wang ✉ ⬤
Google Research, Mountain View, CA, USA
Harvard University, Cambridge, MA, USA

─── **Abstract** ───

We study differential privacy (DP) in a multi-party setting where each party only trusts a (known) subset of the other parties with its data. Specifically, given a *trust graph* where vertices correspond to parties and neighbors are mutually trusting, we give a DP algorithm for aggregation with a much better privacy-utility trade-off than in the well-studied local model of DP (where each party trusts no other party). We further study a robust variant where each party trusts all but an unknown subset of at most t of its neighbors (where t is a given parameter), and give an algorithm for this setting. We complement our algorithms with lower bounds, and discuss implications of our work to other tasks in private learning and analytics.

1 Introduction

Differential privacy (DP) [22, 21] is a rigorous privacy notion that has seen extensive study (e.g., [23, 52]) and widespread adoption in analytics and learning (e.g., [20, 2, 46, 51]). It dictates that the output of a randomized algorithm remains statistically indistinguishable if the data of a single user changes. The most widely studied models of DP are the *central* model where a trusted curator is given access to the raw data and required to output a DP estimate of the function of interest, and the *local* model [26, 37] where every message leaving each user's device is required to be DP. While the latter is compelling in that each user needs to place minimal trust in other users, it is known to suffer from a significantly higher utility degradation compared to the former (e.g., [15]).

In practice, data sharing settings often include situations where a user is willing to place more trust in a *subset* of other users. For example, many people have different privacy sensitivities depending on their relationships with others: Alice might be willing to share her location data with family and close friends, but unwilling to have her location data be recoverable by strangers from a public channel. This relates to philosophical conceptualizations of privacy as *control over personal information*, in that individuals may specify with whom they are willing share their information [49].

Trust Graph DP Models

In this work, we model such relationships as a *trust graph* where vertices correspond to different users and neighboring vertices are mutually trusting (see Figure 1 for a simple example). We define and study DP over such trust graphs, whereby the DP guarantee is enforced on the messages exchanged by each vertex or its trusted neighbors on one hand, and the other non-trusted vertices on the other hand.
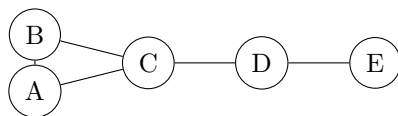
■ **Figure 1** Simple example trust graph. User A is only willing to share their data with users B and C, and user C is additionally willing to share their data with D. We introduce a privacy model (TGDP) in which users D and E cannot identify user A's data based on any communication exchanged.

Specifically, we define notions of *Trust Graph DP (TGDP)* that generalize existing definitions of local DP [37] and central DP, and effectively interpolate between them. Informally, TGDP requires that the distribution of all messages exchanged by each vertex v or one of its neighbors with any vertices that are not trusted by v should remain statistically indistinguishable if the input data held by v changes; we formalize this in Definition 7.

We further extend TGDP to capture *robustness* to potentially compromised neighbors. Namely, the above privacy guarantee can break if a single neighbor of a vertex turns out to be untrustworthy. Thus, we introduce the notion of *Robust Trust Graph DP (RTGDP)*, which maintains the privacy guarantee even if some unknown subset of neighbors of a given size is compromised; see Definition 14.

Our Results

Having defined these trust graph-based models of DP, we give algorithms for the basic *aggregation* primitive that satisfy both TGDP and RTGDP notions. Notably, we propose algorithms that depend on linear programming formulations that can be computed in polynomial time (Theorems 9 and 15). We complement our algorithms with lower bounds on the error that depend on combinatorial properties of the graph (Theorems 10 and 16). Although closing the gap between our upper and lower bounds is still open, we obtain a *bi-criteria* result showing that the upper bound is not much larger than the lower bound when we slightly increase the robustness parameter t (Theorem 17).

The aggregation primitive we study in this work is a basic building block. Indeed, our work implies new DP algorithms over trust graphs for other problems in learning and analytics (see Appendix 5).

We supplement the theory with evaluation on nine real network datasets including email communication networks, social networks, and cryptocurrency trust networks. Our results show that the utility degradation when satisfying TGDP and RTGDP can be significantly lower than that of local DP. Thus, when trust relationships exist, accounting for these relationships when incorporating DP into a system can considerably improve overall utility.

1.1 Technical Overview

We first give a TGDP mechanism for the integer aggregation problem with a mean-squared error (MSE) that scales linearly in the size of any dominating set[1] of the trust graph. The idea of the protocol is the following: Each user identifies one user in the dominating set that they trust and to whom they send their input. Then, each user in the dominating set simply runs the central (discrete) Laplace mechanism to privatize the data. The final estimate is the sum of all these sub-estimates in the dominating set. The MSE grows with the dominating set size, as formalized in Theorem 8.

A disadvantage of the protocol described above is that, in order to minimize the error, it requires the knowledge of a minimum dominating set. However, computing a minimum dominating set is NP-hard and even hard to approximate [27]. So we give (in Theorem 9) a protocol that not only is efficient but can also reduce the error by up to $O(\log n)$ factor. This protocol only requires a solution to a linear program (LP), which, unlike the minimum dominating set, can be computed in polynomial time. At a high-level, there are two key ideas in this protocol:

(i) *Input Splitting:* Instead of sending the input to a single vertex as in the previous section, each user will split their input into (random) additive shares and send it to all its neighbors. The input splitting idea originated in cryptography [36] and has recently found applications in the shuffle model of DP [6, 32], although the nature of how we use it here is quite different from those previous works.

(ii) *Distributed Noise Addition:* Similar to the previous protocol, each user again broadcasts the sum of all messages they receive with some noise added. The main difference here is that, instead of using the discrete Laplace noise, we use the negative binomial noise designed in such a way that, when sufficiently many of them are summed up, they guarantee DP. This helps reduce the amount of noise required in the protocol. (The idea of distributed noise generation dates back to the early works on DP [21], but the distributions we use here are from [6].)

What we gain by applying the input splitting is that, due to the properties of random additive shares, the only way the adversary learns anything about x_v is to sum up all the messages broadcast from its neighbors. By a careful design of the distributed noise distribution, we can ensure that this sum contains sufficient noise to provide DP guarantees.

Our lower bound on integer aggregation with TGDP (Theorem 10) shows that the MSE grows with the packing number of the trust graph. The main idea is to transform any TGDP protocol to a local DP (LDP) protocol with the same privacy and utility, but with a number of users equal to the packing number. The "packing" property ensures that the users are "isolated" from each other in the reduction step.

For our results in the RTGDP model, we consider the same LP but impose a stricter constraint to ensure DP guarantees even when some neighbors of each vertex are compromised. To prove our bi-criteria tightness (Theorem 17), we study the dual of the LP and apply randomized rounding to convert the fractional solution into an integral one.

1.2 Related Work

Secure multiparty computation (SMPC) [55, 56, 34] can be leveraged to allow users to achieve central DP utility without relying on a trusted curator; this is done via cryptographic protocols whose security relies on computational hardness assumptions [21, 8, 11, 9]. An

---

[1] See Section 3 for the formal definitions of a dominating set and a packing.

important distinction between our model and SMPC is that while the privacy of SMPC protocols relies on computational hardness assumptions, the privacy guaranteed in our proposed TGDP model is information-theoretic (and thus stronger). Still, our proposed TGDP model can also be thought of as relaxing the SMPC threat model to assume that a subset of at least a certain number of users is trustworthy (or would execute the algorithm faithfully). Specifically, the proposed TGDP notion could enable higher-utility protocols than the state-of-the-art in SMPC, by making a stronger (but realistic) assumption that different users fully trust some subset of other users (namely, their neighbors in the trust graph).

We also point out that the multi-central (a.k.a. multi-server) model of DP [50, 17] is a special case of our RTGDP model. Another intermediate model between local and central DP is the shuffle model [10, 25, 16], where aggregation has been extensively studied (e.g., [5, 30, 31]); but this model does not capture mutually trusting relationships between different pairs of users. Finally, the network DP model [18] is a different relaxation of local DP where the (DP) communication between users is restricted to the edges of a given graph; this is in contrast to our proposed TGDP model where a given graph encodes trust relationships.

Organization

We start with some background in Section 2. In Section 3, we formally define the notion of DP on trust graphs, and give algorithms and a lower bound for solving the aggregation task under this privacy guarantee. The RTGDP notion is defined in Section 4 where we also give an algorithm and lower bounds for aggregation under this robust notion. We conclude with some interesting future directions in Section 6. Missing proofs are in Appendix B. In Appendix C, we include experiments in which we report our given upper and lower bounds on real network datasets.

2 Preliminaries

Let n users be represented as vertices $V = \{1, ..., n\}$ of a graph $G = (V, E)$, where $E \subseteq V^2$ corresponds to the set of pairs (i, j) of users, where i and j are willing to share their data with each other.[2] Let $\mathcal{X}$ be any domain and suppose each user $i \in V$ has data $x_i \in \mathcal{X}$, and the (full) input dataset is given by $(x_1, \ldots, x_n) = \mathbf{x} \in \mathcal{X}^n$. Let $N(v)$ be the *neighborhood* of v in $G = (V, E)$, i.e., $N(v) = \{u \mid (u, v) \in E\}$ and let $N[v]$ be the *closed neighborhood* of v, i.e., $N[v] = N(v) \cup \{v\}$.

2.1 Differential Privacy Definitions and Tools

▶ **Definition 1** (DP; [22, 21]). *A randomized mechanism $M : \mathcal{X}^n \to \mathcal{O}$ is (ε, δ)-differentially private ((ϵ, δ)-DP) if for all pairs $\mathbf{x}, \mathbf{x}' \in \mathcal{X}^n$ of datasets that differ only in the data of a single user, and for all subsets $S \subseteq \mathcal{O}$, $\Pr[M(\mathbf{x}) \in S] \leq e^\varepsilon \Pr[M(\mathbf{x}') \in S] + \delta$.*

For brevity, we write $(\varepsilon, 0)$-DP as ε-DP (a.k.a., *pure*-DP).

In *non-interactive local DP*, each user has to randomize their own input and send it to the server (or, alternatively, publish it). In this case, each user's randomized output is required to be DP:

---

[2] For simplicity, we focus only on the "symmetric" notion of sharing. It is relatively simple to extend all of our algorithms to the "asymmetric" version as well.

▶ **Definition 2** (Non-Interactive Local DP; [37]). *A randomized mechanism $M : \mathcal{X} \to \mathcal{O}$ is a* non-interactive (ε, δ)*-local DP randomizer if for any pair $x, x' \in \mathcal{X}$, and for all subsets $S \subseteq \mathcal{O}$, $\Pr[M(x) \in S] \leq e^\varepsilon \Pr[M(x') \in S] + \delta$.*

To define DP properties of possibly interactive protocols, we follow the approach of [8]. First, we define the notion of a protocol view.

▶ **Definition 3** (View). *The* view *of a protocol P at vertex u for an input dataset $\mathbf{x} \in \mathcal{X}^n$, denoted $\text{VIEW}_P^u(\mathbf{x})$, consists of the input x_u and all messages received and sent (together with the corresponding source/destination) by the vertex u.*

The view of the protocol P for a subset $S \subseteq V$ of vertices is defined as $\text{VIEW}_P^S(\mathbf{x}) := (\text{VIEW}_P^u(\mathbf{x}))_{u \in S}$. Let $\mathcal{O}$ be the set of all possible views. For any $T \subseteq V$, we write $\mathbf{x}_{-T}$ as a shorthand for $\mathbf{x}_{V \smallsetminus T}$ and, for any $v \in V$, $\mathbf{x}_{-v}$ as a shorthand for $\mathbf{x}_{-\{v\}}$. When the protocol is interactive, local DP can be defined as follows [8]:

▶ **Definition 4** (Interactive Local DP). *A protocol P satisfies (ϵ, δ)-local DP ((ϵ, δ)-LDP) if for each vertex $v \in V$, $\text{VIEW}_P^{V \smallsetminus \{v\}}(\mathbf{x})$ satisfies (ε, δ)-DP with respect to the input x_v for all values of $\mathbf{x}_{-v}$. I.e., for all pairs $x_v, x'_v \in \mathcal{X}$, all values of $\mathbf{x}_{-v}$, and all subsets $S \subseteq \mathcal{O}$,*

$$\Pr[\text{VIEW}_P^{V \smallsetminus \{v\}}(x_v, \mathbf{x}_{-v}) \in S] \leq e^\varepsilon \Pr[\text{VIEW}_P^{V \smallsetminus \{v\}}(x'_v, \mathbf{x}_{-v}) \in S] + \delta.$$

For pure-DP, it is useful to define $D_\infty(\mathcal{P} \parallel \mathcal{P}') := \max_{o \in \text{supp}(\mathcal{P})} \ln\left(\frac{\Pr_{X \sim \mathcal{P}}[X = o]}{\Pr_{X' \sim \mathcal{P}}[X' = o]}\right)$ for distributions $\mathcal{P}, \mathcal{P}'$. We will sometimes use random variables and distributions interchangeably.

It is well-known that DP is robust to post-processing. This fact will be useful in our privacy analysis.

▶ **Lemma 5** (Post-Processing). *For any random variables X, X' and a (possibly randomized) function f, we have $D_\infty(f(X) \parallel f(X')) \leq D_\infty(X \parallel X')$.*

We will use the following distributions for the noise:
- The *negative binomial distribution* $\text{NB}(r, p)$ with parameters $r > 0, p \in (0, 1)$ is supported on $\mathbb{Z}_{\geq 0}$ with density $\Pr[X = k] = \binom{k+r-1}{k}(1-p)^k p^r$, where $X \sim \text{NB}(r, p)$. Its variance is $r(1-p)/p^2$.
- Let $\text{sNB}(r, p)$ be the distribution of $X - X'$ where $X, X' \sim \text{NB}(r, p)$ are i.i.d.
- The *discrete Laplace distribution* $\text{DLap}(b)$ with parameter $b > 0$ is supported on $\mathbb{Z}$ and its density is given by $\Pr[X = k] \propto \exp(-|k|/b)$ for $X \sim \text{DLap}(b)$.

We will use the following facts in our analysis:
- If $X_1 \sim \text{sNB}(r_1, p)$ and $X_2 \sim \text{sNB}(r_2, p)$, then $X_1 + X_2 \sim \text{sNB}(r_1 + r_2, p)$.
- $\text{DLap}(b)$ is the same distribution as $\text{sNB}(1, 1 - e^{-1/b})$.

The discrete Laplace mechanism is well-known to guarantee DP in the central setting [33]. Below, we state a slightly more general version of this for sNB that will be convenient for our analysis.

▶ **Lemma 6.** *For any $x, x' \in \{0, \dots, \Delta\}$, let $Z \sim \text{sNB}(r, 1 - e^{-\varepsilon/\Delta})$ where $r \geq 1$. Then, we have $D_\infty(Z + x \parallel Z + x') \leq \varepsilon$.*

In the privacy analysis, we often consider $\text{VIEW}_P^S(x_v, \mathbf{x}_{-v})$ and $\text{VIEW}_P^S(x'_v, \mathbf{x}_{-v})$ for $x_v, x'_v \in \mathcal{X}$. For convenience, we will write $\text{VIEW}_P^S(x)$ for $x \in \mathcal{X}$ as a shorthand for $\text{VIEW}_P^S(x_v, \mathbf{x}_{-v})$ when $x_v = x$. Similarly, for a quantity y that depends on x_v, we will write $y(x)$ to denote y when $x = x_v$.

3 Trust Graph Differential Privacy

We model trust relationships across users as a network where vertices correspond to users, and undirected edges connect users who are mutually willing to share their data (see Figure 1). We focus on undirected trust graphs, though extensions of our results to directed graphs are possible. For a given trust graph, we define a general notion of Trust Graph DP, provide algorithms for achieving it, and analyze upper and lower bounds on the error for the integer aggregation problem.

▶ **Definition 7.** *(Trust Graph DP) Let $G = (V, E)$. A protocol P satisfies (ϵ, δ, G)-Trust Graph DP $((\epsilon, \delta, G)$-TGDP) if for each vertex $v \in V$, $\text{VIEW}_P^{V \smallsetminus N[v]}(\mathbf{x})$ satisfies (ε, δ)-DP with respect to the input x_v for all values of $\mathbf{x}_{-v}$. I.e., for all pairs $x_v, x'_v \in \mathcal{X}$, all values of $\mathbf{x}_{-v}$, and all subsets $S \subseteq \mathcal{O}$,*

$$\Pr[\text{VIEW}_P^{V \smallsetminus N[v]}(x_v, \mathbf{x}_{-v}) \in S] \leq e^\varepsilon \Pr[\text{VIEW}_P^{V \smallsetminus N[v]}(x'_v, \mathbf{x}_{-v}) \in S] + \delta.$$

Referring back to Figure 1 as an example, Definition 7 says that even if users D and E pooled their messages together, their collective view would still be DP with respect to the data for user A.

Notably, the proposed TGDP model generalizes both the central DP model and the local DP model. The central DP model is captured when G is a star graph in which all vertices (the users) entrust their data a single central vertex (the analyst): each user's data is private relative to the view of all other users, but all users trust the same central analyst. The local DP model, on the other hand, is captured when G simply has no edges between any vertices. Our TGDP model thus introduces a flexibility to capture intermediate trust relationships, perhaps involving several local analysts, or more general trust graphs arising from social networks.

As before, we write $(\varepsilon, 0, G)$-TGDP as (ε, G)-TGDP. Note that the (ε, G)-DP condition in Definition 7 can be written as $D_\infty \left(\text{VIEW}_P^{V \smallsetminus N[v]}(x_v) \, \big\| \, \text{VIEW}_P^{V \smallsetminus N[v]}(x'_v) \right) \leq \varepsilon$.

Recall that for a graph $G = (V, E)$, a *dominating set* is a subset $U \subseteq V$ such that for every $v \in V \smallsetminus U$, there is a $u \in U$ such that $(u, v) \in E$; the size of a minimum dominating set is the *domination number* $\gamma(G)$. A *packing* of G is a subset $U \subseteq V$ such that for any distinct $u, u' \in U$, $N[u]$ and $N[u']$ are disjoint; the size of a maximum packing is the *packing number* $\rho(G)$.

Aggregation

We consider the *integer aggregation* problem. Let each individual have a value $x_i \in \{0, \ldots, \Delta\}$. The goal is to compute an estimate $\tilde{a}$ of $a = \sum_{i=1}^n x_i$. We measure the *mean-square error* (MSE), which is defined as $\mathbb{E}[(\tilde{a} - a)^2]$, where the expectation is over the randomness of the protocol. In central DP, the standard Laplace mechanism [22] achieves an error of $2\Delta^2/\varepsilon^2$. In local DP, the local version of the Laplace mechanism achieves an error of $2\Delta^2 n/\varepsilon^2$. Both of these are known to be asymptotically optimal.

3.1 Algorithm via Dominating Set

We start by giving a protocol for the integer aggregation problem using the graph's dominating set.

▶ **Theorem 8.** *There is an (ε, G)-TGDP mechanism for the aggregation problem with MSE at most $2\Delta^2|T|/\varepsilon^2$, where T is any dominating set of G.*

Proof. The protocol works as follows:

- First, each user $v \in V$ picks an arbitrary vertex $u_v \in T \cap N[v]$. (The intersection is not empty since T is a dominating set.) Then, the user sends x_v to u_v.
- Each user $u \in T$ broadcasts the sum of all numbers it receives together with a noise drawn from $\mathrm{DLap}(\Delta/\varepsilon)$. More formally, the user broadcasts $a_u = \sum_{\substack{v \in V \\ u_v = u}} x_v + z_u$, where $z_u \sim \mathrm{DLap}(\Delta/\varepsilon)$.
- Finally, the estimate is $\tilde{a} = \sum_{u \in T} a_u$.

Privacy Analysis. Consider any $v \in V$ and $\mathbf{x}_{-v} \in \{0, \dots, \Delta\}^{V \smallsetminus \{v\}}$. We write $\mathbf{a}$ as a shorthand for $(a_u)_{u \in T}$. Let $S_v := \{w \in N[v] \mid u_w \in N[v]\}$ denote the nodes in $N[v]$ whose message in the first step is sent to a node in $N[v]$. Notice that $\mathrm{VIEW}_P^{V \smallsetminus N[v]}(x)$ is exactly $(\mathbf{x}_{-S_v}, \mathbf{a}(x))$. We claim that this is a post-processing of $z_{u_v} + x$. This is simply because $\mathbf{x}_{-S_v}, (a_u)_{u \in T \smallsetminus \{u_v\}}$ do not depend on $x_v = x$ at all and are independent of $z_{u_v} + x$; finally, note that $a_{u_v}(x)$ is a post-processing of $z_{u_v} + x$ since $a_{u_v}(x) = (z_{u_v} + x) + \sum_{\substack{v' \in V \smallsetminus \{v\} \\ u_{v'} = u_v}} x_{v'}$.

Consider any $x_v, x_v' \in \{0, \dots, \Delta\}$. By Lemma 5 and Lemma 6, we have

$$D_\infty \left(\mathrm{VIEW}_P^{V \smallsetminus N[v]}(x_v) \, \middle\| \, \mathrm{VIEW}_P^{V \smallsetminus N[v]}(x_v') \right) \leq D_\infty(z_{u_v} + x_v \, \| \, z_{u_v} + x_v') \leq \varepsilon,$$

where the second inequality is due to Lemma 6. Thus, the protocol satisfies (ε, G)-TGDP as desired.

Utility Analysis. The MSE is $\mathbb{E}\left[(\tilde{a} - a)^2\right] = \sum_{u \in T} \mathbb{E}[z_u^2] \leq |T| \cdot \frac{2\Delta^2}{\varepsilon^2}$. ◀

3.2 Improved Algorithm via Linear Programming

A disadvantage of the protocol from Section 3.1 is that to minimize the error, it requires the knowledge of a minimum dominating set. Computing minimum dominating set is NP-hard and even hard to approximate [27]. In this section, we give a protocol that is efficient to compute and furthermore can reduce the error by up to $O(\log n)$ factor in certain graphs. To describe our protocol, recall the linear programming (LP) relaxation of the dominating set problem:

$$\min \sum_{u \in V} y_u \qquad \text{s.t.} \quad \sum_{u \in N[v]} y_u \geq 1 \quad \forall v \in V; \qquad 0 \leq y_u \leq 1 \quad \forall u \in V. \tag{1}$$

To see that this is a relaxation of the dominating set problem, note that any dominating set $T \subseteq V$ gives a solution by setting $y_v = \mathbf{1}[v \in T]$. Due to this, the optimum of this LP is no more than the size of the dominating set. In fact, the LP optimum can be smaller than the minimum dominating set size by an $O(\log n)$ factor [45].

The main result is a protocol whose MSE scales with the LP optimum instead of dominating set:

▶ **Theorem 9.** *There is an (ε, G)-TGDP mechanism for the aggregation problem with MSE at most $2\Delta^2 \cdot \mathrm{OPT}_{\mathrm{LP}}/\varepsilon^2$, where $\mathrm{OPT}_{\mathrm{LP}}$ denotes the value of the optimal solution to the LP in (1).*

Proof. Let $\mathbf{y} = (y_u)_{u \in V}$ denote any solution to the LP in (1). The protocol works as follows:

- Let $q = 2n\Delta$.

- For every user $v \in V$, pick $\{s_v^u\}_{u \in N[v]} \subseteq \mathbb{Z}_q$ uniformly at random among those that satisfy $\sum_{u \in N[v]} s_v^u \equiv x_v \mod q$. Then, for every $u \in N[v]$, user v sends s_v^u to u.

- For every $u \in V$, sample $z_u \sim \text{sNB}(y_u, 1 - e^{-\varepsilon/\Delta})$; broadcast $a_u \equiv z_u + \sum_{v \in N[u]} s_v^u \mod q$.

- Compute $a' \equiv \sum_u a_u \mod q$. Then, output $\tilde{a} = \begin{cases} a' & \text{if } a' \leq q/2, \\ a' - q & \text{otherwise.} \end{cases}$

Privacy Analysis. Throughout the analysis, we assume that the addition is modulo q unless stated otherwise. Consider any $v \in V$ and $\mathbf{x}_{-v} \in \{0, \dots, \Delta\}^{V \smallsetminus \{v\}}$. We write $\mathbf{a}$ as a shorthand for $(a_u)_{u \in V}$. Notice that $\text{VIEW}_P^{V \smallsetminus N[v]}(x)$ is exactly $(\mathbf{x}_{-N[v]}, \mathbf{a}(x), (s_{v'}^u)_{u \in V \smallsetminus N[v], v' \in V})$. We claim that this is a post-processing of $(z_u + s_v^u)_{u \in N[v]}$. This is simply because $\mathbf{x}_{-N[v]}, (a_u)_{u \in V \smallsetminus N[v]}, (s_{v'}^u)_{u \in V \smallsetminus N[v], v' \in V}$ do not depend on $x_v = x$ at all and are independent of $(z_u + s_v^u)_{u \in N[v]}$; finally, note that $(a_u(x))_{u \in N[v]}$ is a post-processing of $(z_u + s_v^u)_{u \in N[v]}$ since $a_u(x) = (z_u + s_v^u) + \sum_{\substack{v' \in N[v] \smallsetminus \{v\} \\ u_{v'} = u}} s_{v'}^u$ for all $u \in N[v]$.

For any $x_v, x_v' \in \{0, \dots, \Delta\}$, Lemma 5 implies that

$$D_\infty \left(\text{VIEW}_P^{V \smallsetminus N[v]}(x_v) \,\Big\|\, \text{VIEW}_P^{V \smallsetminus N[v]}(x_v') \right) \leq D_\infty((z_u + s_v^u(x_v))_{u \in N[v]} \,\|\, (z_u + s_v^u(x_v'))_{u \in N[v]}).$$

Now, since $(s_v^u(x))_{u \in N[v]}$ are random elements of $\mathbb{Z}_q$ that sum to x, we also have that $(z_u + s_v^u(x))_{u \in N[v]}$ are random elements of $\mathbb{Z}_q$ that sum to $x + \sum_{u \in N[v]} z_u$. In other words, $(z_u + s_v^u(x_v))_{u \in N[v]}$ is a post-processing of $x + \sum_{u \in N[v]} z_u$. Again, Lemma 5 implies that

$$D_\infty((z_u + s_v^u(x_v))_{u \in N[v]} \,\|\, (z_u + s_v^u(x_v'))_{u \in N[v]}) \leq D_\infty \left(x_v + \sum_{u \in N[v]} z_u \,\Big\|\, x_v' + \sum_{u \in N[v]} z_u \right).$$

Finally, $Z := \sum_{u \in N[v]} z_u$ is distributed as $\text{sNB}\left(\sum_{u \in N[v]} y_u, 1 - e^{-\varepsilon/\Delta}\right)$. Since $\mathbf{y}$ is feasible in (1), we have $\sum_{u \in N[v]} y_u \geq 1$. Thus, we can apply Lemma 6 to conclude that the RHS above is $\leq \varepsilon$.

Utility Analysis. Note $\tilde{a} \equiv a + \left(\sum_{u \in V} z_u\right) \mod q$. Since $a \in [0, q/2]$ and $\tilde{a} \in (-q/2, q/2]$, we have $|a - \tilde{a}| \leq |\sum_{u \in V} z_u|$. Thus, the MSE is $\leq \sum_{u \in T} \mathbb{E}[z_u^2] \leq \sum_{u \in T} \frac{2y_u}{(\varepsilon/\Delta)^2} = \frac{2\Delta^2 \text{OPT}_{\text{LP}}}{\varepsilon^2}$, where the last equality is from our assumption that $(y_u)_{u \in V}$ is an optimal solution to the LP in (1). ◄

We remark that, in the proof above, the privacy guarantee holds even for non-optimal LP solution $\mathbf{y}$, as long as it satisfies the constraints. Similarly, the error guarantee holds where OPT_{LP} is replaced with the objective value of the solution. This is helpful for practical applications where we may only have an approximately optimal LP solution.

3.3 Lower Bound

We now give a lower bound for integer aggregation, where the MSE grows with the packing number.

▶ **Theorem 10.** *For any $\varepsilon \leq O(1)$, any (ε, G)-TGDP protocol for integer aggregation incurs MSE $\Omega(\Delta^2 \cdot \rho(G))$, where $\rho(G)$ denotes the packing number of the trust graph G.*

In fact, we give the following reduction that transforms any TGDP protocol to an LDP protocol with the same privacy parameter and MSE, but only on $\rho(G)$ users (instead of n users). Applying the known $\Omega(\Delta^2 n)$ lower bound for integer aggregation in LDP [15][3] immediately yields Theorem 10.

▶ **Lemma 11.** *Suppose that there is an (ε, G)-TGDP protocol for integer aggregation. Then, there exists an ε-local DP protocol for integer aggregation for $\rho(G)$ users with the same MSE as the (ε, G)-TGDP protocol, where $\rho(G)$ denotes the packing number of G.*

Proof. Let $U = \{u_1, \dots, u_m\} \subseteq V$ be the largest packing in G where $m = \rho(G)$. To avoid ambiguity, let $\tilde{\mathbf{x}} = (\tilde{x}_1, \dots, \tilde{x}_m)$ be the input to the LDP protocol (that we construct below).

To construct the LDP protocol, let $Q_1 \cup \cdots \cup Q_m$ be any partition of V such that $N[u_i] \subseteq Q_i$ for all $i \in [m]$. Such a partition exists because $N[u_1], \dots, N[u_m]$ are disjoint by the definition of packing. Let P be any (ε, G)-TGDP protocol for integer aggregation. Our LDP protocol $\tilde{P}$ runs the protocol P where each $\tilde{P}$'s user $i \in [m]$ assumes the role of all P's users in Q_i, where the input to P is defined as $x_u = \begin{cases} \tilde{x}_i & \text{if } u = u_i \\ 0 & \text{otherwise,} \end{cases} \quad \forall u \in Q_i$. We then output the estimate as produced by P. The MSE of $\tilde{P}$ is obviously the same as that of P.

To see that $\tilde{P}$ satisfies ε-LDP, consider any $i \in [m], \tilde{\mathbf{x}}_{-i} \in \mathcal{X}^{[m] \smallsetminus \{i\}}$, we have $\mathrm{VIEW}_{\tilde{P}}^{[m] \smallsetminus \{i\}}(\tilde{x}) = \mathrm{VIEW}_P^{V \smallsetminus Q_i}(\mathbf{x}(\tilde{x}))$, where $\mathbf{x}(\tilde{x})$ is the input to P as defined above. Since $V \smallsetminus Q_i \subseteq V \smallsetminus N[u_i]$, $\mathrm{VIEW}_P^{V \smallsetminus Q_i}(\mathbf{x}(\tilde{x}))$ is a post-processing of $\mathrm{VIEW}_P^{V \smallsetminus N[u_i]}(\mathbf{x}(\tilde{x}))$, Lemma 5 implies that

$$D_\infty \left(\mathrm{VIEW}_{\tilde{P}}^{[m] \smallsetminus \{i\}}(\tilde{x}_i) \,\middle\|\, \mathrm{VIEW}_{\tilde{P}}^{[m] \smallsetminus \{i\}}(\tilde{x}_i') \right) \leq D_\infty \left(\mathrm{VIEW}_P^{V \smallsetminus N[u_i]}(\tilde{x}_i) \,\middle\|\, \mathrm{VIEW}_P^{V \smallsetminus N[u_i]}(\tilde{x}_i') \right) \leq \varepsilon,$$

where the last inequality is due to P being an (ε, G)-TGDP protocol. Hence, $\tilde{P}$ is ε-LDP. ◀

Unfortunately, the lower bound in Theorem 10 is not tight with respect to the upper bounds in Theorems 8 and 9. Indeed, the following is a example of a graph that has a large gap between the domination number and the packing number [14]. Let $V = [k] \times [k]$ for $k \in \mathbb{N}$. There is an edge between any $(x, y) \in V, (x', y') \in V$ iff $x = x'$ or $y = y'$. For this graph, $\mathrm{OPT}_{\mathrm{LP}}$ is $\Omega(\sqrt{|V|})$ since every vertex has degree $O(k) = O(\sqrt{|V|})$ whereas the maximal packing has size exactly one (see Figure 2 for $k = 4$). We can also show that this instance exhibits an asymptotically optimal gap:

▶ **Theorem 12.** *For any graph G, $\mathrm{OPT}_{\mathrm{LP}} \leq \rho(G) \cdot \sqrt{n}$ where $\mathrm{OPT}_{\mathrm{LP}}$ denote the value of the optimal solution to the LP in* (1).

In other words, our upper bound based on the LP (Theorem 9) and our lower bound (Theorem 10) on the MSE has a gap of at most $O(\sqrt{n})$. To the best of our knowledge, the bound in Theorem 12 was not known before; we give the full proof in Section D.

Note that the above instance also gives a gap of $\Omega(\sqrt{|V|})$ between the domination number and the packing number. Since it is known [45] that $\gamma(G) \leq O(\log n) \cdot \mathrm{OPT}_{\mathrm{LP}}$, Theorem 12 implies the following corollary:

▶ **Corollary 13.** *For any graph G, $\gamma(G) \leq \rho(G) \cdot O(\sqrt{n} \cdot \log n)$.*

That is, the above gap instance is tight up to a logarithmic factor. Furthermore, this also means that our upper bound based on the dominating set (Theorem 8) and our lower bound (Theorem 10) on the MSE has a gap of at most $O(\sqrt{n})$.

---

[3] Note that [15] state their lower bound for $\Delta = 1$ but the case $\Delta > 1$ follows by scaling up the input.

4 Robust Trust Graph Differential Privacy

In the previous section, we assumed that each user u trusts all of their neighbors $N(u)$. Although this is certainly a reasonable assumption, it might pose a security risk. For example, if one of the neighbors of u is compromised, then u's data might be leaked as the model offers no protection with respect to the view of u's neighbors. Indeed, in the dominating set protocol (Theorem 8), the user sends their raw data to one of their neighbors; if this neighbor is compromised, then the user's data is leaked in the clear. To mitigate this, we propose a revised trust graph DP model that is more robust to such leakage. In particular, for each user u, the DP protection remains as long as at most t_u of their neighbors are compromised, where t_u is some predefined number. This is formalized below.

▶ **Definition 14.** *(Robust Trust Graph DP)* Let $G = (V, E)$ and $\mathbf{t} = (t_v)_{v \in V} \in \mathbb{Z}_{\geq 0}^V$. A *protocol P satisfies $(\epsilon, \delta, G, \mathbf{t})$-Robust Trust Graph DP $((\epsilon, \delta, G, \mathbf{t})$-RTGDP) if for each vertex $v \in V$ and every set $T \subseteq N(v)$ of size at most t_v, VIEW$_P^{V \smallsetminus (N[v] \smallsetminus T)}(\mathbf{x})$ satisfies (ε, δ)-DP with respect to the input x_v for all values of $\mathbf{x}_{-v}$. I.e., for all pairs $x_v, x_v' \in \mathcal{X}$, all values of $\mathbf{x}_{-v}$, and all subsets $S \subseteq \mathcal{O}$,*

$$\Pr[\textit{VIEW}_P^{V \smallsetminus (N[v] \smallsetminus T)}(x_v, \mathbf{x}_{-v}) \in S] \leq e^{\varepsilon} \Pr[\textit{VIEW}_P^{V \smallsetminus (N[v] \smallsetminus T)}(x_v', \mathbf{x}_{-v}) \in S] + \delta.$$

4.1 Integer Aggregation Protocol

We start by giving an integer aggregation protocol that is again based on an LP. We adapt the LP in (1) by imposing a stricter constraint to ensure DP guarantees even when up to t_v of v's neighbor are compromised. This results in the following LP where the only difference compared to (1) is the stricter first constraint.[4] Note that when $\mathbf{t} = 0$, the two LPs coincide.

$$\min \sum_{u \in V} y_u \qquad \text{s.t.} \sum_{u \in (N[v] \smallsetminus T)} y_u \geq 1 \quad \forall v \in V,\ T \in \binom{N(v)}{\leq t_v}; \quad 0 \leq y_u \leq 1 \quad \forall u \in V. \quad (2)$$

While (2) can have exponential size due to the presence of "$\forall T \in \binom{N(v)}{\leq t_v}$" in the constraint, it can still be solved in polynomial time because an efficient separation oracle exists for the first constraint. Specifically, for each $v \in V$, we can check the first constraint by letting T be the t_v maximum values and check the inequality for just that T (instead of enumerating over all $T \in \binom{N(v)}{\leq t_v}$). From the above LP, we can derive an algorithm that uses exactly the same protocol as in Theorem 9.

▶ **Theorem 15.** *There is an $(\varepsilon, G, \mathbf{t})$-RTGDP protocol for the aggregation problem with MSE at most $2\Delta^2 \cdot \mathrm{OPT}_{\mathrm{LP}}^{\mathbf{t}}/\varepsilon^2$, where $\mathrm{OPT}_{\mathrm{LP}}^{\mathbf{t}}$ denotes the value of the optimal solution to the LP in (2).*

4.2 Lower Bound

We define a $(2, \mathbf{t})$-robust packing of $G = (V, E)$ as a pair $U \subseteq V$ and $(T_u)_{u \in U}$ such that (i) $N[u] \smallsetminus T_u$ are disjoint for all $u \in U$ and (ii) $T_u \in \binom{N(u)}{\leq t_u}$ for all $u \in U$. The size of the robust packing is $|U|$. Let $\rho(G)^{\mathbf{t}}$ denote the largest size of a $(2, \mathbf{t})$-robust packing of G.

---

[4] $\binom{S}{\leq t}$ denotes the collection of all subsets of S of size $\leq t$.

Note that, when $\mathbf{t} = \mathbf{0}$, $(2, \mathbf{t})$-robust packing coincides with the standard notion of packing we used in the previous section. We prove a lower bound for integer aggregation in the $(\varepsilon, G, \mathbf{t})$-RTGDP model that grows with the size of the maximum $(2, \mathbf{t})$-robust packing:

▶ **Theorem 16.** *For any $\varepsilon \leq O(1)$ and $\mathbf{t} \in \mathbb{N}^V$, any $(\varepsilon, G, \mathbf{t})$-RTGDP protocol for integer aggregation must incur MSE at least $\Omega(\Delta^2 \cdot \rho(G)^{\mathbf{t}})$.*[5]

4.3 Bi-criteria Tightness of the Bounds

Although we are not aware in general how large the gap between our upper (Theorem 15) and lower bounds (Theorem 16) are, we can show the following *bi-criteria* result, that the upper bound is not much larger than the lower bound when we increase $\mathbf{t}$ slightly. This is stated and proved below.

We write $\lceil \alpha \cdot \mathbf{t} \rceil$ for some $\alpha > 0$ as a shorthand for the vector $(\lceil \alpha \cdot t_u \rceil)_{u \in U}$. Furthermore, let $\deg_U$ denote the vector of degrees of the vertices, i.e., $(\deg(u))_{u \in U}$.

▶ **Theorem 17.** *For any $\alpha \in (0, 1)$, $\rho(G)^{\mathbf{t} + \lceil \alpha \cdot \deg_U \rceil} \geq \frac{\alpha}{8} \cdot \mathrm{OPT}_{\mathrm{LP}}^{\mathbf{t}}$, where $\rho(G)^{\mathbf{t}}, \mathrm{OPT}_{\mathrm{LP}}^{\mathbf{t}}$ are as defined in Theorems 15 and 16.*

To show this, we consider the dual of LP in (2), which turns out to be a relaxation for $(2, \mathbf{t})$-robust packing where there is a variable $w_{v,T} \in [0, 1]$ for all $v \in V, T \subseteq \binom{N(v)}{\leq t_v}$ representing whether (v, T) should be included in the robust packing. To turn such a fractional solution to an integral one, we employ *randomized rounding* – a standard technique in approximation algorithms (e.g., [54, Chapter 5]). More precisely, we include (v, T) in our solution with probability proportional to $w_{v,T}$. Unfortunately, this does not work yet as the produced solution may not be a $(2, \mathbf{t})$-robust packing, i.e., it might contain $w_{v,T}$ and $w_{v',T'}$ such that $(N[v] \smallsetminus T)$ and $(N[v'] \smallsetminus T')$ are not disjoint. Due to this, we need to apply a correction procedure on top of this randomized solution. Roughly speaking, we try to enlarge T until we are sure that such an intersection is avoided. This is indeed the reason why we need the slight increase in $\mathbf{t}$. However, even with this increase, we still have to be careful as sometimes T might become too large, i.e., larger than $t_v + \lceil \alpha \cdot \deg(v) \rceil$. We deal with this by simply removing such a pair (v, T) from the solution. A careful analysis shows that (in expectation) only a small fraction of the solution will get removed this way; see Section B.

5 Machine Learning with Trust Graph DP

While the main body of our work focuses on integer aggregation, it is a primitive on which we can build many more complex algorithms. First of all, we can easily use it to perform *real number aggregation* by re-scaling and discretization [6]. In particular, suppose that each user now has $x_i \in [0, 1]$. They can pick $\Delta \in \mathbb{N}$ and perform integer aggregation on $y_i = \mathrm{round}(\Delta x_i)$ where $\mathrm{round}(x)$ randomly round x to either $\lfloor x \rfloor$ or $1 + \lfloor x \rfloor$ with probabilities $1 - (x - \lfloor x \rfloor)$ and $x - \lfloor x \rfloor$ respectively. Once we have run the integer aggregation protocol, the answer is scaled by a factor of $\frac{1}{\Delta}$. If our integer aggregation protocol has MSE $\Delta^2 \xi^2$, then this results in a real aggregation protocol with error $\xi^2 + \frac{n}{4\Delta^2}$ [6]. Picking Δ to be sufficiently large (e.g., $\omega(\sqrt{n})$), the second term becomes negligible.

Real number aggregation allows us to perform statistical queries (SQ) [38]. While a simple family, statistical queries have wide variety of applications in learning theory. One specific work we wish to highlight is that of [29], who showed that convex optimization problems can be solved using statistical queries. As a result, we can apply their algorithm

---

[5] Again, this theorem is shown via a reduction to the LDP model; see Section B for the proof.

to our setting and obtain convex optimization algorithms with Trust Graph DP. We also remark that statistical queries are also useful for (non-ML) data analytic tasks; e.g., [29] provides SQ-based (vector) mean estimation, and other statistics such as quantiles are also known to be computable via SQs [28].

5.1 Vector Summation with Trust Graph DP

Although SQ-based algorithms can be used in our model, we will sketch a more direct algorithm for the task of *vector summation* with Trust Graph DP. This is not only more efficient but also provide better error guarantees for subsequent tasks such as convex empirical risk minimization (ERM).

In the *vector summation* (with ℓ_2-norm bound) problem, each user input x_i is a vector in $\mathbb{R}^d$ such that $\|x_i\|_2 \leq \Delta$, where Δ is a norm bound known to the algorithm. The goal is again to compute an estimate $\tilde{a}$ to the sum $a = \sum_{i \in [n]} x_i$. The ℓ_2^2-error is defined as $\mathbb{E}[\|\tilde{a} - a\|_2^2]$. Furthermore, we say that the estimator is unbiased if $\mathbb{E}[\tilde{a}] = a$.

It will be easiest to state the algorithms in terms of *zero-concentrated differential privacy (zCDP)* [13, 24]. To do so, we first define α-Renyi divergence for $\alpha > 1$ between distributions $\mathcal{P}, \mathcal{P}'$ for two distributions $\mathcal{P}, \mathcal{P}'$ to be $D_\alpha(\mathcal{P} \parallel \mathcal{P}') := \frac{1}{\alpha-1} \ln \left(\mathbb{E}_{x \sim \mathcal{P}} \left[\left(\frac{\mathcal{P}(x)}{\mathcal{P}'(x)} \right)^{\alpha-1} \right] \right)$. We note that $\lim_{\alpha \to \infty} D_\alpha(\mathcal{P} \parallel \mathcal{P}')$ is indeed equal to $D_\infty(\mathcal{P} \parallel \mathcal{P}')$ that we defined in Section 2.

zCDP can now be defined as follows.

▶ **Definition 18** (zCDP; [13]). *A randomized mechanism $M : \mathcal{X}^n \to \mathcal{O}$ is ρ-zero concentrated DP (ρ-zCDP) if for all pairs $\mathbf{x}, \mathbf{x}' \in \mathcal{X}^n$ of datasets that differ only in the data of a single user and all $\alpha > 1$,*

$$D_\alpha\left(M(\mathbf{x}) \parallel M(\mathbf{x}')\right) \leq \alpha\rho. \tag{3}$$

ZCDP can be easily converted to DP:

▶ **Lemma 19** ([13]). *For any $\rho > 0, \delta \in (0, 1/2)$, any ρ-zCDP algorithm is $(\rho + 2\sqrt{\rho \cdot \ln(1/\delta)}, \delta)$-DP.*

It also has a simple composition theorem:

▶ **Lemma 20** ([13]). *Let M be a mechanism that just runs subroutines that are ρ_1-zCDP, ..., ρ_m-zCDP. Then, M is $(\rho_1 + \cdots + \rho_m)$-zCDP.*

Trust Graph zCDP is simply zCDP on the view of the non-neighbors:

▶ **Definition 21** (Trust Graph zCDP). *Let $G = (V, E)$. A protocol P satisfies (ρ, G)-Trust Graph zCDP ((ρ, G)-TGzCDP) if for each vertex $v \in V$, $VIEW_P^{V \smallsetminus N[v]}(\mathbf{x})$ satisfies ρ-zCDP with respect to the input x_v for all values of $\mathbf{x}_{-v}$. I.e., for all pairs $x_v, x_v' \in \mathcal{X}$, all values of $\mathbf{x}_{-v}$ and all $\alpha > 1$,*

$$D_\alpha\left(VIEW_P^{V \smallsetminus N[v]}(x_v, \mathbf{x}_{-v}) \,\Big\|\, VIEW_P^{V \smallsetminus N[v]}(x_v', \mathbf{x}_{-v})\right) \leq \alpha\rho.$$

We can now state the algorithm for vector summation, which is similar to the algorithm in Theorem 8.

▶ **Theorem 22.** *There is an (ρ, G)-TGzCDP mechanism for the aggregation problem with ℓ_2^2-error $2d\Delta^2 |T|/\rho$, where T is any dominating set of G. Furthermore, the estimate is unbiased.*

Proof. Let $\sigma = \Delta\sqrt{\frac{1}{2\rho}}$. The protocol works as follows:

- First, each user $v \in V$ picks an arbitrary vertex $u_v \in T \cap N[v]$.
- Each user $u \in T$ broadcasts the sum of all vectors it receives together with a noise drawn from $\mathcal{N}(0, \sigma^2 I_d)$. More formally, the user broadcasts $a_u = \sum_{\substack{v \in V \\ u_v = u}} x_v + z_u$, where $z_u \sim \mathcal{N}(0, \sigma^2 I_d)$.
- Finally, the estimate is $\tilde{a} = \sum_{u \in T} a_u$.

Privacy Analysis   Consider any $v \in V$ and $\mathbf{x}_{-v} \in \mathcal{X}^{V \smallsetminus \{v\}}$. Similar to the proof of Theorem 8, the view is a post-processing of $z_u + x$. Thus, for any $x_v, x_v' \in \mathcal{X}$, we have

$$D_\alpha\left(\text{VIEW}_P^{V \smallsetminus N[v]}(x_v) \,\middle\|\, \text{VIEW}_P^{V \smallsetminus N[v]}(x_v')\right) \le D_\alpha(z_{u_v} + x_v \,\|\, z_{u_v} + x_v') \le \alpha \cdot \rho,$$

where the second inequality follows from the zCDP guarantee of the Gaussian mechanism (e.g., Proposition 16 of [13]). Thus, the protocol satisfies (ρ, G)-TGzCDP as desired.

Utility Analysis   It is clear that the estimate is unbiased. The ℓ_2^2-error is

$$\mathbb{E}\left[(\tilde{a} - a)^2\right] = \sum_{u \in T} \mathbb{E}[\|z_u\|^2] = |T| \cdot (\sigma^2 d) = 2d\Delta^2 |T|/\rho. \qquad \blacktriangleleft$$

Using Lemma 19, we immediately get the following corollary. For $|T| = \Theta(1)$, this guarantee (asymptotically) matches the known lower bound in central DP (see, e.g., [7]), while for $|T| = \Theta(n)$, this guarantee nearly matches the known lower bound in local DP [4].

▶ **Corollary 23.** *For any $\varepsilon < O(\log(1/\delta))$, there is an (ε, δ, G)-TGDP mechanism for the aggregation problem with ℓ_2^2-error $O\left(\frac{d\Delta^2 |T| \log(1/\delta)}{\varepsilon^2}\right)$, where T is any dominating set of G. Furthermore, the estimate is unbiased.*

5.2   From Vector Summation to Convex Optimizaiton

Using the above vector summation protocol, we can immediately implement several DP ML algorithms in the literature, such as the DP-SGD algorithm [1]. We can also obtain formal guarantee for convex optimization problems from these algorithms. For instance, let us consider the convex ERM problem [7]. Here there is a loss function $\ell : \mathcal{W} \times \mathcal{X} \to \mathbb{R}$ which is L-Lipschitz on the first parameter and suppose that the diameter of $\mathcal{W}$ is at most R. The goal is to minimize the empirical loss $\mathcal{L}(w, \mathbf{x}) := \frac{1}{n} \sum_{v \in V} \ell(w, x_v)$. Using the above vector summation algorithm, we can arrive at the following:

▶ **Theorem 24.** *For $0 < \rho \le O(1)$, there is an (ρ, G)-TGzCDP mechanism for the convex ERM problem with expected excess risk $O\left(\frac{RL\sqrt{d|T|/\rho}}{n}\right)$, where T is any dominating set of G.*

Proof Sketch. The algorithm uses the (stochastic) mirror descent (see, e.g., [12, Section 6.1]) over n^2 steps. In each step, the gradient is computed by running our (ρ', G)-TGzCDP vector summation protocol with $\rho' = \rho/n^2, \Delta = G$ and scale the answer by a factor of $\frac{1}{n}$. The composition theorem (Lemma 20) implies that the algorithm is ρ-TGzCDP. The excess risk guarantee follows from standard analysis of stochastic mirror descent (e.g., [12, Theorem 6.1]). ◀

Again, using Lemma 19, we immediately get the following corollary for (ε, δ)-DP. For $|T| = \Theta(1)$, this guarantee (asymptotically) matches the known lower bound in central DP [7].

▶ **Corollary 25.** *For any $\varepsilon < O(\log(1/\delta))$, there is an (ε, δ, G)-TGDP mechanism for the convex ERM problem with expected excess risk $O\left(\frac{RL\sqrt{d|T|\log(1/\delta)}}{\varepsilon n}\right)$, where T is any dominating set of G.*

6   Conclusion and Future Directions

We have proposed a new model of privacy given a graph of trust relationships between users. Our model generalizes central and local DP, and further captures intermediate trust structures such as social networks or multiple curators. A significant open theoretical problem is to close the gap between the upper and lower bounds for TGDP, though our experiments suggest that this gap may be small in practice.

References

1 Martín Abadi, Andy Chu, Ian J. Goodfellow, H. Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *CCS*, pages 308–318, 2016. `doi:10.1145/2976749.2978318`.

2 John M Abowd. The US Census Bureau adopts differential privacy. In *KDD*, pages 2867–2867, 2018.

3 Akshay Agrawal, Robin Verschueren, Steven Diamond, and Stephen Boyd. A rewriting system for convex optimization problems. *Journal of Control and Decision*, 5(1):42–60, 2018. `doi:10.1080/23307706.2017.1397554`.

4 Hilal Asi, Vitaly Feldman, and Kunal Talwar. Optimal algorithms for mean estimation under local differential privacy. In *ICML*, pages 1046–1056, 2022. URL: `https://proceedings.mlr.press/v162/asi22b.html`.

5 Borja Balle, James Bell, Adrià Gascón, and Kobbi Nissim. The privacy blanket of the shuffle model. In *CRYPTO*, pages 638–667, 2019. `doi:10.1007/978-3-030-26951-7_22`.

6 Borja Balle, James Bell, Adrià Gascón, and Kobbi Nissim. Private summation in the multi-message shuffle model. In *CCS*, pages 657–676, 2020. `doi:10.1145/3372297.3417242`.

7 Raef Bassily, Adam D. Smith, and Abhradeep Thakurta. Private empirical risk minimization: Efficient algorithms and tight error bounds. In *FOCS*, pages 464–473, 2014. `doi:10.1109/FOCS.2014.56`.

8 Amos Beimel, Kobbi Nissim, and Eran Omri. Distributed private data analysis: Simultaneously solving how and what. In *CRYPTO*, pages 451–468, 2008. `doi:10.1007/978-3-540-85174-5_25`.

9 James Henry Bell, Kallista A Bonawitz, Adrià Gascón, Tancrède Lepoint, and Mariana Raykova. Secure single-server aggregation with (poly) logarithmic overhead. In *CCS*, pages 1253–1269, 2020. `doi:10.1145/3372297.3417885`.

10 Andrea Bittau, Úlfar Erlingsson, Petros Maniatis, Ilya Mironov, Ananth Raghunathan, David Lie, Mitch Rudominer, Ushasree Kode, Julien Tinnés, and Bernhard Seefeld. Prochlo: Strong privacy for analytics in the crowd. In *SOSP*, pages 441–459, 2017. `doi:10.1145/3132747.3132769`.

11 Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. Practical secure aggregation for privacy-preserving machine learning. In *CCS*, pages 1175–1191, 2017.

12 Sébastien Bubeck. Convex optimization: Algorithms and complexity. *Found. Trends Mach. Learn.*, 8(3-4):231–357, 2015. `doi:10.1561/2200000050`.

13 Mark Bun and Thomas Steinke. Concentrated differential privacy: Simplifications, extensions, and lower bounds. In *TCC*, pages 635–658, 2016. `doi:10.1007/978-3-662-53641-4_24`.

14 Alewyn P. Burger, Michael A. Henning, and Jan H. van Vuuren. On the ratios between packing and domination parameters of a graph. *Discrete Mathematics*, 309:2473–2478, 2009. `doi:10.1016/J.DISC.2008.05.030`.

15 T.-H. Hubert Chan, Elaine Shi, and Dawn Song. Optimal lower bound for differentially private multi-party aggregation. In *ESA*, pages 277–288, 2012. `doi:10.1007/978-3-642-33090-2_25`.

16 Albert Cheu, Adam D. Smith, Jonathan Ullman, David Zeber, and Maxim Zhilyaev. Distributed differential privacy via shuffling. In *EUROCRYPT*, pages 375–403, 2019. `doi:10.1007/978-3-030-17653-2_13`.

17 Albert Cheu and Chao Yan. Necessary conditions in multi-server differential privacy. In *ITCS*, 2023.

18 Edwige Cyffers and Aurélien Bellet. Privacy amplification by decentralization. In *AISTATS*, pages 5334–5353, 2022. URL: `https://proceedings.mlr.press/v151/cyffers22a.html`.

19 Steven Diamond and Stephen Boyd. CVXPY: A Python-embedded modeling language for convex optimization. *JMLR*, 17(83):1–5, 2016. URL: `https://jmlr.org/papers/v17/15-408.html`.

20 Bolin Ding, Janardhan Kulkarni, and Sergey Yekhanin. Collecting telemetry data privately. *NIPS*, 30, 2017.

21 Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. Our data, ourselves: Privacy via distributed noise generation. In *EUROCRYPT*, pages 486–503, 2006. `doi:10.1007/11761679_29`.

22 Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam D. Smith. Calibrating noise to sensitivity in private data analysis. In *TCC*, pages 265–284, 2006. `doi:10.1007/11681878_14`.

23 Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4):211–407, 2014. `doi:10.1561/0400000042`.

24 Cynthia Dwork and Guy N. Rothblum. Concentrated differential privacy. *CoRR*, abs/1603.01887, 2016. `arXiv:1603.01887`.

25 Úlfar Erlingsson, Vitaly Feldman, Ilya Mironov, Ananth Raghunathan, Kunal Talwar, and Abhradeep Thakurta. Amplification by shuffling: From local to central differential privacy via anonymity. In *SODA*, pages 2468–2479, 2019. `doi:10.1137/1.9781611975482.151`.

26 Alexandre Evfimievski, Johannes Gehrke, and Ramakrishnan Srikant. Limiting privacy breaches in privacy preserving data mining. In *PODS*, pages 211–222, 2003. `doi:10.1145/773153.773174`.

27 Uriel Feige. A threshold of ln n for approximating set cover. *JACM*, 45(4):634–652, 1998. `doi:10.1145/285055.285059`.

28 Vitaly Feldman. Dealing with range anxiety in mean estimation via statistical queries. In *ALT*, pages 629–640, 2017. URL: `http://proceedings.mlr.press/v76/feldman17b.html`.

29 Vitaly Feldman, Cristóbal Guzmán, and Santosh S. Vempala. Statistical query algorithms for mean vector estimation and stochastic convex optimization. In *SODA*, pages 1265–1277, 2017. `doi:10.1137/1.9781611974782.82`.

30 Badih Ghazi, Ravi Kumar, Pasin Manurangsi, and Rasmus Pagh. Private counting from anonymous messages: Near-optimal accuracy with vanishing communication overhead. In *ICML*, pages 3505–3514, 2020. URL: `http://proceedings.mlr.press/v119/ghazi20a.html`.

31 Badih Ghazi, Ravi Kumar, Pasin Manurangsi, Rasmus Pagh, and Amer Sinha. Differentially private aggregation in the shuffle model: Almost central accuracy in almost a single message. In *ICML*, pages 3692–3701, 2021. URL: `http://proceedings.mlr.press/v139/ghazi21a.html`.

32 Badih Ghazi, Pasin Manurangsi, Rasmus Pagh, and Ameya Velingker. Private aggregation from fewer anonymous messages. In *EUROCRYPT*, pages 798–827, 2020. `doi:10.1007/978-3-030-45724-2_27`.

33 Arpita Ghosh, Tim Roughgarden, and Mukund Sundararajan. Universally utility-maximizing privacy mechanisms. *SICOMP*, 41(6):1673–1693, 2012. `doi:10.1137/09076828X`.

34 Oded Goldreich, Silvio Micali, and Avi Wigderson. How to play any mental game, or a completeness theorem for protocols with honest majority. In *Providing Sound Foundations for Cryptography: On the Work of Shafi Goldwasser and Silvio Micali*, pages 307–328. Association for Computing Machinery, 2019. `doi:10.1145/3335741.3335755`.

35 Magnús M. Halldórsson, Jan Kratochvíl, and Jan Arne Telle. Independent sets with domination constraints. *Discret. Appl. Math.*, 99(1-3):39–54, 2000. `doi:10.1016/S0166-218X(99)00124-9`.

36 Yuval Ishai, Eyal Kushilevitz, Rafail Ostrovsky, and Amit Sahai. Cryptography from anonymity. In *FOCS*, pages 239–248, 2006. `doi:10.1109/FOCS.2006.25`.

37 Shiva Prasad Kasiviswanathan, Homin K Lee, Kobbi Nissim, Sofya Raskhodnikova, and Adam Smith. What can we learn privately? *SICOMP*, 40(3):793–826, 2011. `doi:10.1137/090756090`.

38 Michael J. Kearns. Efficient noise-tolerant learning from statistical queries. *JACM*, 45(6):983–1006, 1998. `doi:10.1145/293347.293351`.

39 Bryan Klimt and Yiming Yang. The Enron corpus: A new dataset for email classification research. In *ECML*, pages 217–226, 2004. `doi:10.1007/978-3-540-30115-8_22`.

40 Srijan Kumar, Bryan Hooi, Disha Makhija, Mohit Kumar, Christos Faloutsos, and VS Subrahmanian. Rev2: Fraudulent user prediction in rating platforms. In *WSDM*, pages 333–341, 2018.

41 Srijan Kumar, Francesca Spezzano, VS Subrahmanian, and Christos Faloutsos. Edge weight prediction in weighted signed networks. In *ICDM*, pages 221–230, 2016.

42 Jure Leskovec, Jon Kleinberg, and Christos Faloutsos. Graph evolution: Densification and shrinking diameters. *TKDD*, 1(1), 2007. `doi:10.1145/1217299.1217301`.

43 Jure Leskovec and Andrej Krevl. SNAP Datasets: Stanford large network dataset collection. `http://snap.stanford.edu/data`, June 2014.

44 Jure Leskovec and Julian Mcauley. Learning to discover social circles in ego networks. *NIPS*, 25, 2012.

45 László Lovász. On the ratio of optimal integral and fractional covers. *Discret. Math.*, 13(4):383–390, 1975. `doi:10.1016/0012-365X(75)90058-8`.

46 Carey Radebaugh and Ulfar Erlingsson. Introducing TensorFlow Privacy: Learning with Differential Privacy for Training Data, March 2019. URL: `blog.tensorflow.org`.

47 Matthew Richardson, Rakesh Agrawal, and Pedro Domingos. Trust management for the semantic web. In *ISWC*, pages 351–368, 2003. `doi:10.1007/978-3-540-39718-2_23`.

48 Benedek Rozemberczki, Carl Allen, and Rik Sarkar. Multi-scale attributed node embedding. *CoRR*, abs/1909.13021, 2019. `arXiv:1909.13021`.

49 Daniel J Solove. Conceptualizing privacy. *Calif. L. Rev.*, 90:1087, 2002.

50 Thomas Steinke. Multi-central differential privacy. *CoRR*, abs/2009.05401, 2020. `arXiv:2009.05401`.

51 Davide Testuggine and Ilya Mironov. PyTorch Differential Privacy Series Part 1: DP-SGD Algorithm Explained, August 2020. URL: `medium.com`.

52 Salil Vadhan. The complexity of differential privacy. *Tutorials on the Foundations of Cryptography: Dedicated to Oded Goldreich*, pages 347–450, 2017. `doi:10.1007/978-3-319-57048-8_7`.

53 Stanley L Warner. Randomized response: A survey technique for eliminating evasive answer bias. *JASA*, 60(309):63–69, 1965.

54 David P. Williamson and David B. Shmoys. *The Design of Approximation Algorithms*. Cambridge University Press, 2011.

55 Andrew C-C Yao. Protocols for secure computations. In *FOCS*, pages 160–164, 1982.

56 Andrew C-C Yao. How to generate and exchange secrets. In *FOCS*, pages 162–167, 1986.

A   Example Graph

Figure 2 gives an example of a graph with a significant gap between the domination number and the packing number, which importantly illustrates the gap between the upper and lower bounds on MSE under TGDP in Theorems 8, 9, and 10.
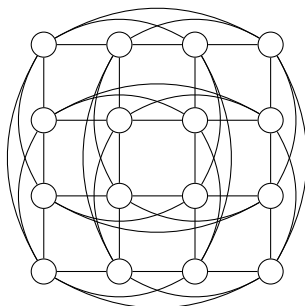
■ **Figure 2** A graph with a gap between the domination number (4) and the packing number (1). The relaxed LP solution $\text{OPT}_{\text{LP}} = 16/7 \approx 2.285$.

B   Missing Proofs

Proof of Lemma 6. We may write $Z = Z^1 + Z^2$ where $Z^1 \sim \text{sNB}\left(1, 1 - e^{-\varepsilon/\Delta}\right) = \text{DLap}(\Delta/\varepsilon)$ and $Z^2 \sim \text{sNB}\left(r - 1, 1 - e^{-\varepsilon/\Delta}\right)$ are independent. We again can think of $x + Z$ and $x' + Z$ as a post-processing of $x + Z^1$ and $x' + Z^1$, respectively. This implies that

$$
\begin{aligned}
&D_\infty\left(x + Z \parallel x + Z\right) \\
&\leq D_\infty\left(x + Z_1 \parallel x' + Z_1\right) \\
&= \max_{o \in \mathbb{N}} \ln\left(\frac{\Pr_{Z_1 \sim \text{DLap}(\Delta/\varepsilon)}[x + Z_1 = o]}{\Pr_{Z_1 \sim \text{DLap}(\Delta/\varepsilon)}[x' + Z_1 = o]}\right) \\
&= \frac{\varepsilon}{\Delta} \cdot \left(|o - x| - |o - x'|\right) \\
&\leq \frac{\varepsilon}{\Delta} \cdot |x' - x| \leq \varepsilon,
\end{aligned}
$$

where the last two inequalities follow from the triangle inequality and since $0 \leq x, x' \leq \Delta$ respectively.   ◄

Proof of Theorem 15. Let $\mathbf{y} = (y_u)_{u \in V}$ denote any solution to LP in (2). We use exactly the same protocol as in Theorem 9. The utility analysis is also similar to before. Thus, we only provide the privacy analysis below.

Privacy Analysis

Consider any $v \in V, T \in \binom{N(v)}{\leq t_v}$ and $\mathbf{x}_{-v} \in \{0, \ldots, \Delta\}^{V \setminus \{v\}}$. We write $\mathbf{a}$ as a shorthand for $(a_u)_{u \in V}$. Notice that $\text{VIEW}_P^{V \setminus (N[v] \setminus T)}(x)$ is exactly $(\mathbf{x}_{-(N[v] \setminus T)}, \mathbf{a}(x), (s_{v'}^u)_{u \in V \setminus (N[v] \setminus T), v' \in V})$. We claim that this is a post-processing of $(z_u + s_v^u)_{u \in (N[v] \setminus T)} \cup (s_v^u)_{u \in T}$. This is simply because $\mathbf{x}_{-(N[v] \setminus T)}$, $(a_u)_{u \in V \setminus N[v]}$,

$(s^u_{v'})_{u\in V\smallsetminus N[v], v'\in V}, (s^u_{v'})_{u\in T, v'\in V\smallsetminus\{v\}}$ do not depend on $x_v = x$ at all and are independent of $(z_u + s^u_v)_{u\in(N[v]\smallsetminus T)} \cup (s^u_v)_{u\in T}$; finally, note that $(a_u(x))_{u\in N[v]}$ is a post-processing of $(z_u + s^u_v)_{u\in N[v]}$ since $a_u(x) = (z_u + s^u_v) + \sum_{\substack{v'\in N[V]\smallsetminus\{v\}\\ u_{v'}=u}} s^u_{v'}$.

Now, since $(s^u_v(x))_{u\in N[v]}$ are random elements of $\mathbb{Z}_q$ that sums to x, we also have that $(z_u + s^u_v)_{u\in(N[v]\smallsetminus T)} \cup (s^u_v)_{u\in T}$ are random elements of $\mathbb{Z}_q$ that sums to $x + \sum_{u\in N[v]} z_u$. In other words, $(z_u + s^u_v)_{u\in(N[v]\smallsetminus T)} \cup (s^u_v)_{u\in T}$ is a post-processing of $x + \sum_{u\in(N[v]\smallsetminus T)} z_u$. The remainder of the privacy proof then proceed similarly to that in Theorem 9, where we now use the fact that $\sum_{u\in(N[v]\smallsetminus T)} z_u$ is distributed as $\mathrm{sNB}\left(\sum_{u\in(N[v]\smallsetminus T)} y_u, 1 - e^{-\varepsilon/\Delta}\right)$ and $\sum_{u\in(N[v]\smallsetminus T)} y_u \geq 1$ from (2). ◄

Theorem 16 is an immediate consequence of the following reduction to the LDP model, similar to Theorem 10 and the corresponding reduction (Lemma 11).

▶ **Lemma 26.** *Suppose that there is an $(\varepsilon, G, \mathbf{t})$-RTGDP protocol for integer aggregation. Then, there exists an ε-local DP protocol for integer aggregation with the same MSE for $\rho(G)^{\mathbf{t}}$ users, where $\rho(G)^{\mathbf{t}}$ denotes the size of the largest $(2, \mathbf{t})$-robust packing G.*

Proof of Lemma 26. Let $U = \{u_1, \ldots, u_m\} \subseteq V$ and $(T_v)_{v\in U}$ be the largest $(2, \mathbf{t})$-robust packing in G where $m = \rho(G)$. To avoid ambiguity, we use $\tilde{\mathbf{x}} = (\tilde{x}_1, \ldots, \tilde{x}_m)$ to denote the input to the local DP protocol.

To construct the LDP protocol, let $Q_1, \ldots, Q_m \subseteq V$ be any partition of V such that $(N[u_i]\smallsetminus T_{u_i}) \subseteq Q_i$ for all $i \in [m]$. Such a partition exists because $N[u_1]\smallsetminus T_{u_1}, \ldots, N[u_m]\smallsetminus T_{u_m}$ are disjoint by definition of a packing. Let P be any $(\varepsilon, G, \mathbf{t})$-RTGDP protocol for integer aggregation. Our LDP protocol $\tilde{P}$ works simply by running the protocol P where each user $i \in [m]$ assumes the role of all users in Q_i where the input is defined as

$$
x_u = \begin{cases} x_i & \text{if } u = u_i \\ 0 & \text{otherwise.} \end{cases}
$$

for all $u \in Q_i$. We then output the estimate as produced by P. The MSE of $\tilde{P}$ is obviously the same as that of P.

To see that this satisfies ε-LDP, consider any $i \in [m], \tilde{\mathbf{x}}_{-i} \in \mathcal{X}^{[m]\smallsetminus\{i\}}$, we have $\mathrm{VIEW}_{\tilde{P}}^{[m]\smallsetminus\{i\}}(\tilde{x}) = \mathrm{VIEW}_P^{V\smallsetminus Q_i}(\mathbf{x}(\tilde{x}))$ where $\mathbf{x}(\tilde{x})$ is the input to P as defined above. Since $V\smallsetminus Q_i \subseteq V\smallsetminus(N[u_i]\smallsetminus T_{u_i})$, $\mathrm{VIEW}_P^{V\smallsetminus Q_i}(\mathbf{x}(\tilde{x}))$ is a post-processing of $\mathrm{VIEW}_P^{V\smallsetminus(N[u_i]\smallsetminus T_{u_i})}(\mathbf{x}(\tilde{x}))$. Thus, for every $\tilde{x}_i, \tilde{x}'_i \in \mathcal{X}$, Lemma 5 implies that

$$
\begin{aligned}
&D_\infty\left(\mathrm{VIEW}_{\tilde{P}}^{[m]\smallsetminus\{i\}}(\tilde{x}_i) \,\|\, \mathrm{VIEW}_{\tilde{P}}^{[m]\smallsetminus\{i\}}(\tilde{x}'_i)\right) \\
&\leq D_\infty\left(\mathrm{VIEW}_P^{V\smallsetminus(N[u_i]\smallsetminus T_{u_i})}(\tilde{x}_i) \,\|\, \mathrm{VIEW}_P^{V\smallsetminus(N[u_i]\smallsetminus T_{u_i})}(\tilde{x}'_i)\right) \\
&\leq \varepsilon,
\end{aligned}
$$

where the last inequality is due to P being $(\varepsilon, G, \mathbf{t})$-RTGDP protocol.

As a result, $\tilde{P}$ is ε-LDP as desired. ◄

Proof of Theorem 17. We will write $\mathbf{t}'$ as a shorthand for $\mathbf{t} + \lceil \alpha \cdot \deg_U \rceil$.

We claim that $\mathrm{OPT}^{\mathbf{t}'}_{\mathrm{pack}} \geq \gamma \cdot \mathrm{OPT}^{\mathbf{t}}_{\mathrm{LP}}$ for $\gamma = \frac{\alpha}{8}$. To prove this, first observe that LP duality implies that $\mathrm{OPT}^{\mathbf{t}}_{\mathrm{LP}}$ is also equal to the optimal of the following LP:

$$
\max \sum_{v\in V, T\in \binom{N(v)}{\leq t_v}} w_{v,T} \tag{4}
$$

$$\text{s.t.} \sum_{\substack{u \in V, T \in \binom{N(u)}{\leq t_u} \\ (N[u] \smallsetminus T) \ni v}} w_{u,T} \leq 1 \qquad\qquad\qquad \forall v \in V \qquad\qquad (5)$$

$$0 \leq w_{v,T} \leq 1 \qquad\qquad\qquad \forall v \in V, T \in \binom{N(v)}{\leq t_v}.$$

For $T' \subseteq N[v']$, we let $\overline{T}'[v'] := N[v'] \smallsetminus T'$.

Given an optimal solution $(w_{v,T})_{v \in V, T \in \binom{N(v)}{\leq t_v}}$ to the dual LP (4), we construct a $(2, \mathbf{t}')$-robust packing as follows:

- Include each (v, T) in the set R_0 with probability $2\gamma \cdot w_{v,T}$.
- Filter elements in R_0 to create R_1 where R_1 only includes (v, T) such that

$$v \notin \bigcup_{(v',T') \in (R_0 \smallsetminus \{(v,T)\})} \overline{T}'[v'], \qquad\qquad (6)$$

and

$$\left| \bigcup_{(v',T') \in (R_0 \smallsetminus \{(v,T)\})} (N(v) \cap \overline{T}'[v']) \right| \leq \alpha \cdot \deg(v). \qquad\qquad (7)$$

- Construct R_2 by adding $\left(v, T \cup \bigcup_{(v',T') \in R_1 \smallsetminus \{(v,T)\}} (N(v') \cap \overline{T}'[v']) \right)$, for each $(v, T) \in R_1$.

By the conditions imposed when constructing R_1, it is simple to see that R_2 is a valid $(2, \mathbf{t})$-robust packing. Thus, we are only left to argue that it has a large size. To do so, first observe that

$$\mathbb{E}[\,|R_2|\,]$$
$$= \mathbb{E}[\,|R_1|\,]$$
$$= \sum_{v \in V, T \in \binom{N(v)}{\leq t_v}} \Pr[(v,T) \in R_1]$$
$$= \sum_{v \in V, T \in \binom{N(v)}{\leq t_v}} \Pr[(v,T) \in R_0] \cdot \Pr[(v,T) \in R_1 \mid (v,T) \in R_0]$$
$$= \sum_{v \in V, T \in \binom{N(v)}{\leq t_v}} (2\gamma \cdot w_{v,T}) \cdot \Pr[(v,T) \in R_1 \mid (v,T) \in R_0], \qquad\qquad (8)$$

where the last equality is due to the randomized rounding in the first step.

Let us now fix $v \in V$ and $T \in \binom{N(v)}{\leq t_v}$. To bound $\Pr[(v,T) \in R_1 \mid (v,T) \in R_0]$, note that from our procedure, we have

$$\Pr[(v,T) \in R_1 \mid (v,T) \in R_0]$$
$$= \Pr[\text{(6) and (7) both hold}]$$
$$\geq 1 - \Pr[\text{(6) fails}] - \Pr[\text{(7) fails}]. \qquad\qquad (9)$$

We bound each of the two terms separately. For the first term, we have

$$\Pr[\text{(6) fails}]$$
$$= \Pr\left[v \in \bigcup_{(v',T') \in (R^0 \smallsetminus \{(v,T)\})} \overline{T}'[v'] \right]$$

$$= \Pr \left[\exists \begin{matrix} v' \in V, T' \in \binom{N(v')}{\leq t_{v'}} \\ (N[v'] \smallsetminus T') \ni v \end{matrix}, (v', T') \in R^0 \right]$$

$$\leq \sum_{\substack{v' \in V, T' \in \binom{N(v')}{\leq t_{v'}} \\ (N[v'] \smallsetminus T') \ni v}} \Pr[(v', T') \in R^0]$$

$$= \sum_{\substack{v' \in V, T' \in \binom{N(v')}{\leq t_{v'}} \\ (N[v'] \smallsetminus T') \ni v}} 2\gamma \cdot w_{v', T'}$$

$$\overset{(5)}{\leq} 2\gamma$$

$$\leq \frac{1}{4}. \tag{10}$$

Similarly, we have

$$\Pr[(7) \text{ fails}]$$

$$= \Pr \left[\left| \bigcup_{(v', T') \in (R^0 \smallsetminus \{(v, T)\})} (N(v) \cap \overline{T}'[v']) \right| > \alpha \cdot \deg(v) \right]$$

$$\leq \frac{\mathbb{E}\left[\left| \bigcup_{(v', T') \in (R^0 \smallsetminus \{(v, T)\})} (N(v) \cap \overline{T}'[v']) \right| \right]}{\alpha \cdot \deg(v)}$$

$$= \frac{\sum_{u \in N(v)} \Pr\left[u \in \bigcup_{(v', T') \in (R^0 \smallsetminus \{(v, T)\})} \overline{T}'[v'] \right]}{\alpha \cdot \deg(v)}$$

$$= \frac{\sum_{u \in N(v)} \Pr\left[\exists \begin{matrix} v' \in V, T' \in \binom{N(v')}{\leq t_{v'}} \\ (N[v'] \smallsetminus T') \ni u \end{matrix}, (v', T') \in R^0 \right]}{\alpha \cdot \deg(v)}$$

$$\leq \frac{\sum_{u \in N(v)} \sum_{\substack{v' \in V, T' \in \binom{N(v')}{\leq t_{v'}} \\ (N[v'] \smallsetminus T') \ni u}} \Pr[(v', T') \in R^0]}{\alpha \cdot \deg(v)}$$

$$= \frac{\sum_{u \in N(v)} \sum_{\substack{v' \in V, T' \in \binom{N(v')}{\leq t_{v'}} \\ (N[v'] \smallsetminus T') \ni u}} 2\gamma \cdot w_{v', T'}}{\alpha \cdot \deg(v)}$$

$$\overset{(5)}{\leq} \frac{\sum_{u \in N(v)} 2\gamma}{\alpha \cdot \deg(v)}$$

$$= \frac{2\gamma}{\alpha} \leq \frac{1}{4}, \tag{11}$$

where the first inequality is due to Markov and the last inequality is from our choice of γ.

Combining Equations (8)–(11), we get

$$\mathbb{E}[\, |R_2| \,] \geq \sum_{v \in V, T \in \binom{N(v)}{\leq t_v}} \gamma \cdot w_{v, T} = \gamma \cdot \mathrm{OPT}_{\mathrm{LP}}^{\mathbf{t}},$$

which concludes the proof. ◀

C Additional Experiment Details

We next provide additional experiment details. Optimization problems were solved using `cvxpy` [19, 3].

C.1   Additional Dataset Details

Table 1 lists additional details for each dataset. All datasets are publicly available through SNAP [43]. We give additional dataset descriptions below.

▪ **Table 1** Integrality gap comparison of OPT_{LP} to minimum dominating set size $|T|$ for 9 network datasets.

Dataset	Number of nodes n	Number of edges $|E|$	Maximum degree
EU Emails (Core)	$1,005$	$16,706$	347
Bitcoin (Alpha)	$3,783$	$12,972$	507
Facebook	$4,039$	$88,234$	$1,045$
Bitcoin (OTC)	$5,881$	$18,591$	788
Enron Emails	$36,692$	$183,831$	$1,383$
GitHub	$37,700$	$289,003$	$9,458$
Epinions	$75,879$	$405,740$	$3,044$
Twitter	$81,306$	$1,342,310$	$3,383$
EU Emails (All)	$265,214$	$365,570$	$7,636$

EU Emails datasets. This data comes from an email network at a European research institution collected by [42]. We include an undirected edge between sender and receiver who have exchanged an email, though the original dataset contains directed edge information. We consider two subgraphs: *(i)* a "core" subgraph consisting of email addresses within the research institution (which we refer to as *EU Emails (Core)*), and *(ii)* the full graph of all emails contained in the dataset (which we refer to as *EU Emails (All)*).

Enron Emails. This data comes from an email communication network within Enron [39]. The graph contains an undirected edge between a sender and receiver if at least one email was exchanged between them. The original graph is undirected.

Bitcoin datasets. We include network data from two different Bitcoin trading platforms, *Bitcoin OTC* and *Bitcoin Alpha* [41, 40]. In the original data, each user rater another user with a value between -10 and 10, where a negative rating corresponds to mistrust and a positive rating corresponds to trust. In our analysis, we include an undirected edge between users if and only if one user rates another with a value greater than 0. We ignore mistrust ratings. In general, further analysis that includes the mistrust ratings would be interesting to conduct.

Facebook dataset. This data from Facebook was published by [44]. Each undirected edge indicates a friend relationship between two users.

GitHub dataset. This data comes from a social network of GitHub developers collected by [48]. Edges represent mutual follower relationships between two users.

Epinions dataset. This data comes from the consumer review site Epinions.com [47]. We include an undirected edge if one user "trusts" another by giving a positive rating to the other user (which indicates trusting their reviews). The original dataset is a directed graph.

Twitter dataset. This data encodes follower relationships from Twitter [44]. We include an undirected edge if one user has follows another. The original dataset contains directed edges for follower relationships.

C.2   Integrality Gap Between Linear Program Optimum and Minimum Dominating Set

The proposed LP-based mechanism for achieving Trust Graph DP presented in Theorem 9 will always perform at least as well as the dominating set protocol in terms of error. Here we consider whether in practice, the LP-based mechanism is better than the dominating

set protocol. We observed an integrality gap between OPT_{LP} and the size of the minimum dominating set $|T|$ on three out of nine datasets. Notably, we only observed the integrality gap on the email communication datasets, and not on the Bitcoin or social network datasets. Theoretically, it is known that the integrality gap between OPT_{LP} and $|T|$ can be up to a factor of up to $O(\log(n))$ [54]. Further study of the integrality gaps that might arise in other real network settings remains an interesting open question. Table 2 lists the ratio $\frac{\text{OPT}_{\text{LP}}}{|T|}$ for each dataset.

Whether any given graph exhibits an integrality gap or not, a prevailing advantage of the proposed improved algorithm via linear programming for TGDP over a minimum dominating set protocol lies in computational efficiency, as finding the minimum dominating set is NP-hard.

Table 2 Integrality gap comparison of OPT_{LP} to minimum dominating set size $|T|$ for 9 network datasets.

Dataset	n	OPT_{LP}	$|T|$	$\frac{\text{OPT}_{\text{LP}}}{|T|}$
EU Emails (Core)	$1{,}005$	111.97	128	0.8748
Bitcoin (Alpha)	$3{,}783$	686	686	1
Facebook	$4{,}039$	10	10	1
Bitcoin (OTC)	$5{,}881$	$1{,}126$	$1{,}126$	1
Enron Emails	$36{,}692$	$3{,}060.66$	$3{,}062$	0.9996
GitHub	$37{,}700$	$4{,}538$	$4{,}538$	1
Epinions	$75{,}879$	$15{,}734$	$15{,}734$	1
Twitter	$81{,}306$	961	961	1
EU Emails (All)	$265{,}214$	$18{,}074.40$	$18{,}181$	0.9941

C.3 Other Local DP Mechanisms for Aggregation

Note that we focused our comparisons to the LDP version of the Laplace mechanism since the ratio has a simple expression. Nevertheless, we point out that there are other LDP mechanisms for aggregation. For example, one can apply randomized rounding to the input (where we set it to Δ with probability x_i/Δ and set it to zero otherwise) before applying the classic randomized response (RR) mechanism [53]. The MSE of this method is actually input-dependent due to the randomized rounding, making it harder to compare against our method. To give the benefit to this algorithm, let us assume for the moment that all inputs are either 0 or Δ. In this case, there is no error due to the randomized rounding. A simple calculation shows that the error from here is $c_\epsilon \cdot 2\Delta^2 n/\epsilon^2$ where $c_\epsilon = \frac{e^\epsilon \cdot \epsilon^2}{2(e^\epsilon - 1)^2}$. For $\epsilon \le 1$, this constant c_ϵ is at least 0.46. Thus, in all cases, our mechanism still demonstrates a significant improvement over this over-optimistic estimate of the error for randomized rounding + RR.

D Relating Packing Number and Minimum Dominating Set

In this section, we prove Theorem 12. Our proof follows that of Halldorsson et al. [35], who gave a greedy algorithm that provides a $\sqrt{n}$-approximation for the packing number. At a high level, the main difference between our proof and theirs is that we compare the greedy solution with the (optimal) LP solution, whereas they compare it with the (optimal) integral solution. The LP for the packing number turns out to be exactly the dual of the LP for the minimum dominating set (i.e., LP (1)); this then gives us the desired claim.

Proof of Theorem 12. Consider the dual of LP (1), which can be written as follows:

$$\min \sum_{u \in V} y_u \tag{12}$$

$$\text{s.t.} \sum_{u \in N[v]} y_u \le 1 \qquad\qquad \forall v \in V \tag{13}$$

$$0 \le y_u \le 1 \qquad\qquad \forall u \in V.$$

Let $\mathbf{y}^* = (y_u^*)_{u \in V}$ denote an optimal solution of the LP (12). Consider the following greedy algorithm by Halldorsson et al. [35].

- Set $S \leftarrow \emptyset, i \leftarrow 0$, and $V_i \leftarrow V$.
- While $V_i \ne \emptyset$ do:
    - Let v_i be the vertex in V_i with the smallest degree (ties broken arbitrarily).
    - Set $S \leftarrow S \cup \{v_i\}$.
    - Let $Z_i := \{u \in V_i \mid N[u] \cap N[v_i] \ne \emptyset\}$.
    - Set $V_{i+1} \leftarrow V_i \smallsetminus Z_i$.
    - Set $i \leftarrow i + 1$.
- Output S.

It is clear that the output S is indeed a packing; let $q = |S|$. We claim the following for all $i \in [q]$:

$$\sum_{u \in Z_i} y_u^* \le \sqrt{n}. \tag{14}$$

Before we prove (14), let us argue that it implies $\rho(G) \ge \text{OPT}_{\text{LP}}/\sqrt{n}$. Notice that $\{Z_i\}_{i \in [q]}$ is a partition of V. Therefore, we have

$$\text{OPT}_{\text{LP}} = \sum_{u \in V} y_u^* = \sum_{i \in [q]} \sum_{u \in Z_i} y_u^* \overset{(14)}{\le} \sum_{i \in [q]} \sqrt{n} \le \rho(G) \cdot \sqrt{n},$$

as desired.

Finally, we prove (14). Consider two cases based on whether $\deg(v_i) \le \sqrt{n} - 1$.

- Case I: $\deg(v_i) \le \sqrt{n} - 1$. In this case, we have

$$\sum_{u \in Z_i} y_u^* \le \sum_{w \in N[v_i]} \sum_{u \in N[w]} y_u^* \overset{(13)}{\le} \sum_{w \in N[v_i]} 1 = \deg(v_i) + 1 \le \sqrt{n}.$$

- Case II: $\deg(v_i) > \sqrt{n} - 1$. Since we pick v_i to be the vertex with the smallest degree among those in V_i, we have $\deg(u) > \sqrt{n} - 1$ for all $u \in Z_i$. Therefore, we have

$$\sum_{u \in Z_i} y_u^* \le \frac{1}{\sqrt{n}} \cdot \sum_{u \in Z_i} (\deg(u) + 1) \cdot y_u^*$$

$$\le \frac{1}{\sqrt{n}} \cdot \sum_{u \in V} (\deg(u) + 1) \cdot y_u^*$$

$$= \frac{1}{\sqrt{n}} \cdot \sum_{v \in V} \sum_{u \in N[v]} y_u^*$$

$$\overset{(13)}{\le} \frac{1}{\sqrt{n}} \cdot n$$

$$= \sqrt{n}.$$

Thus, in both cases (14) holds, which completes our proof. ◀