High-Accuracy, Poisoning-Resilient Frequency Estimation in the Shuffle Model

Shaoqiang Wu[†], Jingyu Jia[†‡], Yikuan Zhu[†], Xinhao Li[†], Changyu Dong[§], Zheli Liu[†]
[†] *CS&CCS, DISSec, AAIS, Nankai University, Tianjin, China,*
{wushaoqiang, jiajingyu, yikuanzhu, asunalxh}@mail.nankai.edu.cn, liuzheli@nankai.edu.cn
[‡] *Automotive New Technology Research Institute of BYD, Shenzhen, China*
[§] *Guangzhou University, Guangzhou, China, Changyu.dong@gmail.com*

Abstract

We study frequency estimation in the shuffle model of differential privacy under *poisoning attacks*, where corrupted users may deviate from the local randomizer to inject crafted in-domain messages. Existing shuffle-model protocols face a core tension: achieving low estimation error relies on flexible multi-message noise generation, which can amplify poisoning influence once messages are anonymized by shuffling.

To address this tension, we propose a *symmetric binomial-sum noise distribution* (i.e., $\mathrm{Bin}(n/2,p)+\mathrm{Bin}(n/2,1-p)$), which preserves high accuracy while limiting the impact of crafted in-domain messages. We realize this distribution via preprocessing-guided noise generation, which routes a balanced collection of mode flags through the shuffler so that each user receives a randomly assigned mode flag that fixes their noise-sampling behavior prior to shuffling. For binary estimation, our protocol requires a single Bernoulli trial per user and at most 2 messages per user (1.5 on average), while bounding the worst-case poisoning influence of a single corrupted user by $O(1/n)$. We extend the protocol to histograms, including large domains via hashing, and provide formal privacy, accuracy, and robustness guarantees. Experiments on real datasets show that our protocols remain resilient under poisoning and reduce MAE by up to nearly $2\times$ over the strongest baseline at comparable per-user communication on small domains, and match it on large domains.

1 Introduction

Differential privacy (DP) has seen broad real-world deployment by both governments and industry [1, 15, 20]. The shuffle model of differential privacy improves the trust–accuracy trade-off by combining local randomization with *privacy amplification by shuffling*, which anonymizes reports by breaking the linkage between senders and messages [4, 5, 10]. For basic analytics tasks such as frequency estimation and histogram

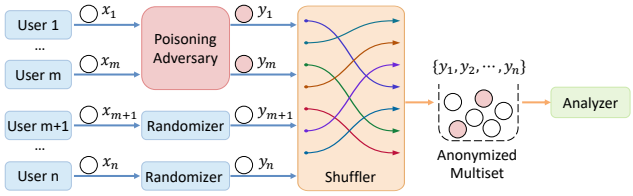

Figure 1: Poisoning attacks in the shuffle model.

construction, a growing body of work develops shuffle protocols that achieve near-central accuracy under similar privacy parameters [2, 11, 18, 19, 25].

After shuffling, the analyzer observes only an unlabeled multiset and, in the standard model, cannot attribute messages to users, a loss that is particularly consequential when some users are adversarial and deviate from the prescribed local randomizer (Fig. 1). While DP bounds how much the output distribution of a fixed mechanism can change under a single-record change, it does not, by itself, imply robustness to arbitrary malicious deviations from the prescribed local randomizer. Many accuracy-oriented shuffle protocols rely on *multi-message* reports to realize low-variance aggregate noise; under anonymity, this multi-message freedom can be exploited for poisoning. Even with a per-user message cap, if many syntactically valid multisets fit within the cap, corrupted users can choose the multiset that maximizes the shift in the estimator's expectation. As a result, existing designs face a tension. Tightly constrained designs admit explicit worst-case poisoning bounds at the cost of higher error, whereas accuracy-oriented multi-message designs achieve lower error but do not provide such bounds in the standard shuffle model [11, 25].

We study shuffle-model frequency estimation under *poisoning attacks* in which an adversary corrupts up to m users and makes them deviate arbitrarily, subject only to the protocol's *per-user message limit*. We quantify robustness by *poisoning influence*: the maximum ℓ_1 shift in the *expected* estimate induced by m corrupted users. Our goal is to design

This is the full version of a paper accepted to the 35th USENIX Security Symposium, 2026.

shuffle protocols that simultaneously (i) satisfy (ε, δ)-DP, (ii) improve on the accuracy of state-of-the-art multi-message shuffle protocols at comparable per-user communication, and (iii) provide an explicit upper bound on poisoning influence, without assuming shuffler capabilities beyond permutation.

Our main technical contribution is a new aggregate-noise design for the shuffle model: the *symmetric binomial-sum noise distribution* $\mathrm{Bin}(n/2, p) + \mathrm{Bin}(n/2, 1-p)$, together with a shuffle-only generation method that is resilient to poisoning. At a high level, we introduce a *data-independent* preprocessing step that assigns each user a mode flag $b_i \in \{0,1\}$ hidden from the analyzer. Concretely, the analyzer submits to the shuffler a multiset containing $n/2$ copies of 0 and $n/2$ copies of 1; the shuffler uniformly permutes these mode flags, and each user receives one permuted mode flag as auxiliary input to the local randomizer, while the analyzer never observes the per-user assignment. Conditioned on b_i, each user samples noise using one of two complementary probabilities (p vs. $1-p$), so that the *aggregate* noise follows the symmetric binomial-sum form. Crucially, the same mechanism that shapes aggregate noise also shrinks the adversary's action space under anonymity, restricting each user's in-domain multiset to a small family (e.g., $\{0,1,2\}$ copies in the binary protocol), which bounds worst-case bias.

This design yields two concrete benefits. First, compared to binomial or skewed binomial-variant noise commonly induced by bounded-influence shuffle protocols [11, 25], our analysis and experiments show that the resulting symmetric binomial-sum noise achieves improved accuracy under comparable privacy and communication. Second, because preprocessing limits each user's feasible in-domain multiset, each additional in-domain message has a bounded marginal effect on the estimate; moreover, our estimators apply a constant shift rather than an amplifying multiplicative debiasing, so adversarial perturbations are not magnified. We provide a complete theoretical analysis of this symmetric binomial-sum noise design, establishing (ε, δ)-DP and explicit accuracy guarantees for binary estimation and for histograms, and deriving explicit poisoning-influence bounds under the standard per-user message-limit premise.

We instantiate the design in a family of frequency-estimation protocols. In the binary setting, each user can contribute only a small in-domain multiset (at most two copies of the symbol 1), and the analyzer estimates the mean via the constant-shift estimator $\tilde{f} = |\mathbf{Y}|/n - 1/2$, where $|\mathbf{Y}|$ is the total number of shuffled messages. The protocol uses at most 2 messages per user (and 1.5 on average). We then extend the design to histograms. A per-bin instantiation incurs $O(d)$ messages per user, so we develop a low-communication protocol in which each user is assigned a single (hidden) noise bin (and a mode) and injects noise only into that bin. This removes the linear-in-d communication cost, and each message carries a $\log d$-bit bin label, while each user perturbs only one bin. Finally, to cover large domains, we give a compressed variant that hashes values into a smaller domain, performs aggregation in the hashed space, and reconstructs unbiased estimates with bounded collision error. Tab. 1 compares our protocols with representative prior work in terms of communication, accuracy, and poisoning influence.

Contributions. We make the following contributions:

- **A new symmetric binomial-sum noise construction.** We introduce a symmetric binomial-sum noise distribution and show how to realize it with a permutation-only shuffler via a data-independent mode assignment, enabling symmetric, low-variance aggregate noise while reducing the poisoning adversary's degrees of freedom under anonymity.
- **Poisoning-resilient protocols with improved accuracy.** Building on the new noise construction, we design binary and histogram protocols that use constant-shift estimators and yield explicit poisoning-influence bounds under the standard per-user message-limit premise, while improving accuracy over binomial or skewed binomial-variant bounded-influence designs.
- **Theory and empirics.** We provide formal analyses of privacy, accuracy, poisoning influence, and message complexity, and empirically validate the resulting accuracy–communication–poisoning-robustness trade-offs on real datasets. Experiments indicate that we reduce MAE by $1.1\times$ to nearly $2\times$ over the best prior protocol [25] at comparable per-user communication and poisoning robustness on the small-domain datasets, and match it on the large-domain dataset.

2 Related Work

Shuffle-model frequency estimation. Frequency estimation and histogram construction have been widely studied in the shuffle model. A first line of work studies *single-message* protocols, where each user locally randomizes their input and sends one report to the shuffler. Shuffling amplifies privacy and can reduce the noise needed relative to purely local mechanisms [4, 10], yet single-message protocols still exhibit a provable accuracy gap to the central model [10].

A second line of work improves accuracy by allowing each user to send *multiple* anonymous messages [2, 5, 17, 18]. To approach central-model error, Balle et al. [5] realize low-variance (e.g., discrete-Laplace-like) noise via split-and-mix messages [22]. Later refinements improve communication by altering the noise-generation/encoding procedure, typically trading a small accuracy loss for fewer messages [18, 19]. However, once messages are anonymized, a corrupted user can choose an arbitrary multiset of in-domain messages within the per-user message limit, which can translate into a large

Table 1: Comparison of shuffle-model frequency-estimation protocols under poisoning attacks. We use (ε, δ) for privacy, n for the number of users, d for the histogram domain size, and d_h for the hashed domain size. *Message per user* is the average number of messages sent by an honest user. *Max error* is a constant-success-probability bound (up to constants) on the ℓ_∞ deviation of the frequency statistic. *Poisoning influence* is the maximum ℓ_1 deviation in the expected frequency statistic under m corrupted users subject to the per-user message limit.

Task	Protocol	Message per user	Max error	Poisoning influence
Binary	CSUZZ [10]	1	$O\left(\frac{\sqrt{\log(1/\delta)}}{\varepsilon n}\right)$	$O\left(\frac{m}{n}/(1-\frac{\sqrt{\log(1/\delta)}}{\varepsilon^2 n})\right)$
Binary	BC [2]	$2-\frac{50\log(2/\delta)}{\varepsilon^2 n}$	$O\left(\frac{\log(1/\delta)}{\varepsilon^2 n}\right)$	$O\left(\frac{m}{n}+\frac{\log(1/\delta)}{\varepsilon^2 n}\right)$
Binary	BBGN [5]	$O(\log\log n)$	$O\left(\frac{\log\log n\sqrt{\log(1/\delta)}}{\varepsilon n}\right)$	$O\left(\frac{m}{n}\cdot(1+\frac{(\log\log n)^2\log(1/\delta)}{n^{2/3}\varepsilon^2})\right)$
Binary	Ours (Thm. 3)	1.5	$O\left(\frac{\sqrt{\log(1/\delta)}}{\varepsilon n}\right)$	$O\left(\frac{m}{n}\right)$
Histogram	CSUZZ [10]	d	$O\left(\frac{\sqrt{\log d\cdot\log(1/\delta)}}{\varepsilon n}\right)$	$O\left(\frac{m}{n}\cdot d/(1-\frac{\sqrt{\log(1/\delta)}}{\varepsilon^2 n})\right)$
Histogram	BC [2]	$1+d-\frac{200d\log(4/\delta)}{\varepsilon^2 n}$	$O\left(\frac{\log(1/\delta)}{\varepsilon^2 n}\right)$	$O\left(\frac{m}{n}\cdot(1+d)\right)$
Histogram	GKMP [18]	$1+O\left(\frac{d\log^2(1/\delta)}{\varepsilon^2 n}\right)$	$O\left(\frac{\log d}{\varepsilon n}\right)$	unbounded†
Histogram	GGKPV [17]	$O\left(\frac{\log^3 d\ \log(\log d/\delta)}{\varepsilon^2}\right)$	$O\left(\frac{\log^{3/2} d\ \sqrt{\log(\log d/\delta)}}{\varepsilon n}\right)$	$O\left(\frac{m}{n}\cdot\frac{\log^2 d\ \log(\log d/\delta)}{\varepsilon^2}\right)$
Histogram	CZ [11]	$1+O\left(\frac{\log(1/\delta)}{\varepsilon^2 n}\right)$	$O\left(\frac{\log d}{n}+\frac{\sqrt{\log d\cdot\log(1/\delta)}}{\varepsilon n}\right)$	$O\left(\frac{m}{n}\cdot d\cdot(1+\frac{\log(1/\delta)}{\varepsilon^2 n}+\frac{\log d}{n})\right)$
Histogram	LWY (small d) [25]	$1+\frac{32d\log(2/\delta)}{\varepsilon^2 n}$	$O\left(\frac{\log d}{n}+\frac{\sqrt{\log d\cdot\log(1/\delta)}}{\varepsilon n}\right)$	$O\left(\frac{m}{n}\cdot(1+\frac{d\log(1/\delta)}{\varepsilon^2 n})\right)$
Histogram	LWY (large d) [25]	$1+\frac{32d_h\log(2/\delta)}{\varepsilon^2 n}$	$O\left(\frac{\log d}{n}+\frac{\sqrt{\log d(\varepsilon^2 n/d_h+\log(1/\delta))}}{\varepsilon n}\right)$	$O\left(\frac{m}{n}\cdot d\cdot(1+\frac{d_h\log(1/\delta)}{\varepsilon^2 n})\right)$
Histogram	Ours (Thm. 7, small d)	$1+O\left(\frac{d\log(1/\delta)}{\varepsilon^2 n}\right)$	$O\left(\frac{\sqrt{\log d\cdot\log(1/\delta)}}{\varepsilon n}\right)$	$O\left(\frac{m}{n}\cdot(1+\frac{d\log(1/\delta)}{\varepsilon^2 n})\right)$
Histogram	Ours (Thm. 8, large d)	$1+O\left(\frac{d_h\log(1/\delta)}{\varepsilon^2 n}\right)$	$O\left(\sqrt{\frac{\log d}{nd_h}}+\frac{\sqrt{\log d\cdot\log(1/\delta)}}{\varepsilon n}\right)$	$O\left(\frac{m}{n}\cdot d\cdot(1+\frac{d_h\log(1/\delta)}{\varepsilon^2 n})\right)$

† GKMP has no deterministic per-user message limit. Its noise count is unbounded, so any external limit distorts honest behavior, while no limit leaves poisoning unbounded. GKMP's message-per-user bound above hides a large constant under its accuracy-favoring privacy-budget split.

worst-case bias of the final estimate. In contrast, another family of multi-message protocols enforces a per-user message limit and a restricted message domain, typically inducing binomial (or binomial-variant) noise. Such restrictions make it possible to derive an explicit per-user influence bound, but may increase estimation error when the induced noise distribution is asymmetric or otherwise mismatched to the target symmetric noise [11, 25].

Closest baselines under the standard shuffle model. Luo et al. [25] provide our closest baseline: a blanket-noise design in which each user sends the true item and, with a data-independent Bernoulli probability, additionally sends a uniformly sampled domain element. Our design instead uses data-independent preprocessing to assign each user a specific noise bin and mode flag, inducing a symmetric binomial-sum noise distribution at each bin. This comparison therefore highlights the effect of symmetric noise generation on accuracy, and the narrower tails of our symmetric binomial-sum noise help explain the lower MAE observed in Tab. 2.

Poisoning and robustness. Poisoning/manipulation attacks have been systematically studied in the local model, showing that adversarial users can exploit local randomizers to submit crafted yet *in-domain* reports that bias the final estimate [7, 9]. Subsequent work extends these attacks to richer tasks (e.g., key–value, mean and variance estimation) and further shows that the effect of privacy strength on manipulation can be

subtle and mechanism-dependent [24,30]. These concerns are also relevant to shuffle protocols built from local randomizers, since by removing sender–message linkage, shuffling can make it harder to enforce or audit per-user reporting structure and to detect structured manipulations once reports appear syntactically valid.

Recent work by Murakami et al. [26] addresses poisoning by moving beyond the standard shuffle setting. They study an *augmented* shuffle model where the shuffler performs additional online operations beyond permutation, namely random sampling and dummy-data addition over user-encrypted inputs. These added capabilities enable local-noise-free constructions and robustness guarantees against poisoning/collusion, but they rely on a stronger shuffler model and incur cryptographic and communication overhead; moreover, the overhead can become challenging to scale to large domains. In contrast, our protocol uses only the shuffler's standard permutation operation, including the components that realize the model extension. The model extension we adopt (Def. 3) is confined to allowing each user to receive a data-independent auxiliary input before the local randomizer applies. Our robustness, in turn, comes from structural constraints on user reports rather than added shuffler functionality, yielding explicit poisoning-influence bounds.

A related but orthogonal line studies *robust shuffle privacy* [3], which concerns privacy robustness. It characterizes how privacy guarantees degrade when only a fraction of users execute the prescribed protocol while the rest may behave arbitrarily. For example, Luo et al. [25] discuss instantiations that tolerate a constant fraction of such users under this privacy notion. In contrast, we focus on bounding the estimator's worst-case bias under adversarial in-domain reports, given a per-user message limit.

Our work occupies a distinct point in the shuffle-protocol design space. First, unlike augmented-shuffle approaches [26], our shuffler is permutation-only (no online sampling or dummy injection) and we derive explicit poisoning-influence bounds under this weaker shuffler capability; we target bounded estimator bias under a per-user message limit, rather than the privacy-degradation framing of robust shuffle privacy. Second, compared to many accuracy-oriented multi-message shuffle protocols [2,5,17,18], where anonymity can leave corrupted users substantial freedom to choose protocol-conforming in-domain message multisets within the message cap, we co-design the noise and report syntax to achieve *both* high accuracy and bounded poisoning influence. Third, within bounded-influence multi-message designs based on binomial or skewed binomial-variant noise [11,25], our symmetric binomial-sum noise avoids the accuracy penalty of such asymmetric noise. In sum, both high accuracy and poisoning resilience follow from this design, not from a stronger shuffler or a looser per-user message limit.

3 Preliminaries

3.1 Differential Privacy

Central and local models. In the central model, a trusted curator collects raw user data and releases randomized outputs satisfying differential privacy (DP). In contrast, the local model eliminates the trusted curator by requiring each user to apply a local randomizer before sending data to the server, resulting in local differential privacy (LDP) guarantees.

Definition 1 (Differential Privacy [13,14])**.** A randomized mechanism $\mathcal{M}: \mathcal{X}^n \to \mathcal{Y}$ satisfies (ε,δ)-DP if, for any neighboring datasets $X, X' \in \mathcal{X}^n$ (differing in exactly one user's record) and any measurable set $Y \subseteq \mathcal{Y}$,

$$\Pr[\mathcal{M}(X) \in Y] \leq e^{\varepsilon} \Pr[\mathcal{M}(X') \in Y] + \delta.$$

We review two important theorems. The post-processing theorem states that an adversary cannot increase the privacy loss by analyzing the mechanism output without the help of additional knowledge of the private dataset.

Theorem 1. (Post-processing Theorem [14]) Let $\mathcal{M}: \mathcal{X}^n \to \mathcal{Y}$ be a randomized mechanism satisfying (ε,δ)-DP and $f: \mathcal{Y} \to \mathcal{Z}$ be any arbitrary randomized mapping. Then $f \circ M: \mathcal{X}^n \to \mathcal{Z}$ satisfies (ε,δ)-DP.

The composition theorem allows the computation of cumulative privacy loss for multiple queries on the raw dataset.

Theorem 2. (Composition Theorem [14]) Let $\mathcal{M}_1: \mathcal{X}^n \to \mathcal{Y}_1$ satisfies (ε_1,δ_1)- differential privacy and $\mathcal{M}_2: \mathcal{X}^n \to \mathcal{Y}_2$ satisfies (ε_2,δ_2)-DP. For any $X \in \mathcal{X}^n$, the mechanism $\mathcal{M} = (\mathcal{M}_1(X), \mathcal{M}_2(X))$ satisfies $(\varepsilon_1+\varepsilon_2, \delta_1+\delta_2)$-DP.

Shuffle model. The shuffle model of differential privacy lies between the local and the central models. A shuffle protocol consists of three components $\mathcal{P} = (\mathcal{R}, \mathcal{S}, \mathcal{A})$. Each user $i \in [n]$ applies a weak local randomizer $\mathcal{R}$ to their input $x_i \in \mathcal{X}$ and sends the resulting one or more messages to a shuffler $\mathcal{S}$. The shuffler applies a uniformly random permutation to all submitted messages and forwards only the resulting *multiset* to an analyzer $\mathcal{A}$ (implemented by the server), thereby removing the linkage between users and individual messages.

Definition 2 (Shuffle-Model DP [10])**.** A protocol $\mathcal{P} = (\mathcal{R}, \mathcal{S}, \mathcal{A})$ satisfies (ε,δ)-DP in the shuffle model if the mechanism

$$(x_1, \ldots, x_n) \mapsto \mathcal{S}\big(\mathcal{R}(x_1), \ldots, \mathcal{R}(x_n)\big)$$

is (ε,δ)-DP as a function from $\mathcal{X}^n$ to shuffled multisets.

To capture our protocols, we extend this model to allow each user to receive a data-independent auxiliary input prior to the local randomizer.

DEFINITION 3 (Shuffle Model with Data-Independent Auxiliary Input; this work)**.** The shuffle model with data-independent auxiliary input is parameterized by an auxiliary-input space $\mathcal{T}$ and a joint distribution $\mathcal{D}_{\text{aux}}$ over $\mathcal{T}^n$ that is independent of the input dataset. A protocol $\mathcal{P} = (\mathcal{R}, \mathcal{S}, \mathcal{A})$ satisfies (ε, δ)-DP in this model if the mechanism

$$(x_1, \ldots, x_n) \mapsto \mathcal{S}\big(\mathcal{R}(x_1, \theta_1), \ldots, \mathcal{R}(x_n, \theta_n)\big),$$

where $(\theta_1, \ldots, \theta_n) \sim \mathcal{D}_{\text{aux}}$, is (ε, δ)-DP as a function from $\mathcal{X}^n$ to shuffled multisets.

Under both the standard model (Def. 2) and our extended model (Def. 3), the shuffler $\mathcal{S}$ performs *no computation beyond permutation*, since it does not drop, insert, modify, or tag reported messages, and it reveals nothing besides the permuted multiset. We treat $\mathcal{S}$ as an ideal shuffler and assume it does not collude with $\mathcal{A}$. In this work, we study shuffle-model protocols for frequency estimation over a domain $\mathcal{X}$ with n users holding inputs $x_i \in \mathcal{X}$, where each user may send a single message or multiple messages under the protocol.

Our protocols realize $\mathcal{D}_{\text{aux}}$ through a data-independent preprocessing step that requires only the shuffler's permutation capability, used to deliver auxiliary inputs from the analyzer to the users. In the binary protocol, $\theta_i = b_i \in \{0,1\}$ is the mode flag and $\mathcal{D}_{\text{aux}}$ is a uniform permutation of $n/2$ copies of 0 and $n/2$ copies of 1; in the histogram protocol, $\theta_i = (j_i, b_i) \in [d] \times \{0,1\}$ specifies the assigned noise bin and the mode flag.

3.2 Poisoning Robustness

Poisoning attacks. We consider a poisoning adversary that corrupts m of n users. Corrupted users may arbitrarily deviate from the prescribed local randomizer. In particular, they may inject *crafted in-domain messages* that are syntactically valid (i.e., lie in the same message domain as honest reports) and therefore cannot be filtered by domain membership once anonymized by shuffling. Accordingly, poisoning influence is governed by how many in-domain messages each corrupted user can contribute per round, and how much each such message perturbs the target statistic once aggregated.

Per-user message-limit premise. All poisoning-influence bounds in this paper are stated under the standard *per-user message limit* premise: in each collection round, the analyzer accepts at most $k+1$ messages from any user (one report plus up to k noise messages), including corrupted users. This premise is essential for bounded-influence robustness in the standard shuffle model. Without an enforceable bound on accepted messages, an adversary could cause arbitrarily many in-domain messages to be accepted, making the poisoning influence unbounded by repetition. We discuss a simple enforcement mechanism as a deployment option in App. F.

Poisoning influence. We quantify poisoning attacks by the worst-case impact on the analyzer's output, measured as the maximum shift in the expected estimate (in ℓ_1 distance).

Let f denote the target statistic. In the binary case or for a single histogram bin, $f \in \mathbb{R}$ is a scalar frequency; for histogram estimation, $f \in \mathbb{R}^d$ is the full frequency vector. Let $\tilde{f}$ be the estimate produced when all users are honest, and let $\tilde{f}^{\mathsf{Adv}}$ be the resulting estimate under attack. We define the *poisoning influence* as

$$\text{Infl}(m) \;:=\; \sup_{\mathsf{Adv}} \big\| \mathbb{E}\big[\tilde{f}\big] - \mathbb{E}\big[\tilde{f}^{\mathsf{Adv}}\big] \big\|_1 ,$$

where the supremum ranges over all poisoning strategies corrupting m users subject to the per-user message limit $k+1$.

Illustrative example. To illustrate how poisoning arises in the shuffle model, consider binary mean estimation via randomized response. Each honest user holds $x \in \{0,1\}$ and reports y such that for some $p \in (1/2, 1)$, $\Pr[y = x] = p$ and $\Pr[y = 1-x] = 1-p$. Given n reports, the standard unbiased estimator is

$$\tilde{f} \;=\; \frac{1}{2p-1}\left(\frac{1}{n}\sum_{i=1}^{n} y_i - (1-p)\right).$$

For any honest user, $\mathbb{E}[y_i] = (2p-1)x_i + (1-p)$, so

$$\mathbb{E}[\tilde{f}] = \frac{1}{n}\sum_{i=1}^{n} x_i.$$

If each corrupted user $i \in \{1, \ldots, m\}$ injects *one* in-domain bit $c_i \in \{0,1\}$ (e.g., always 1), then the poisoned estimator is

$$\tilde{f}^{\mathsf{Adv}} \;=\; \frac{1}{2p-1}\left(\frac{1}{n}\left(\sum_{i=1}^{m} c_i + \sum_{i=m+1}^{n} y_i\right) - (1-p)\right),$$

and the expectation is

$$\mathbb{E}[\tilde{f}^{\mathsf{Adv}}] \;=\; \frac{1}{n}\sum_{i=m+1}^{n} x_i \;+\; \frac{1}{2p-1}\cdot\frac{1}{n}\sum_{i=1}^{m}(c_i - (1-p)).$$

Therefore, the expected estimate is shifted by

$$\text{Infl}(m) \;=\; \frac{m}{n}\cdot\frac{p}{2p-1},$$

demonstrating that even a small number of corrupted users can exert non-negligible influence.

If a corrupted user could inject t accepted in-domain bits, the expected shift would scale linearly with t, underscoring the necessity of an enforceable per-user message limit.

Robustness. Under the standard per-user message limit, poisoning robustness is governed by how much a corrupted user can bias the estimator within this constrained action budget. Concretely, two protocol-induced factors determine $\text{Infl}(m)$: (i) the estimator's per-message effect on the target statistic once messages are aggregated, and (ii) the corrupted user's feasible in-domain message multiset. Intuitively, allowing

richer multi-message reporting can improve accuracy by enabling more expressive noise synthesis, but it may also enlarge the feasible multiset space and thus increase the adversary's degrees of freedom to introduce bias. Our protocol is designed to reconcile these goals, supporting accurate multi-message noise generation while restricting each user's multiset to a structured family, yielding explicit and tunable upper bounds on $\mathrm{Infl}(m)$ under the enforced message-limit premise.

4 Binary Frequency Estimation

We begin with binary frequency estimation (equivalently, binary summation). Each user $i \in [n]$ holds a bit $x_i \in \{0,1\}$, and the analyzer aims to estimate the fraction of ones

$$f = \frac{1}{n}\sum_{i=1}^{n} x_i.$$

For convenience, we assume n is even; this assumption can be removed by rounding and affects only constants.

Design goal. We seek a shuffle-model protocol that simultaneously achieves: (i) (ε,δ)-DP, (ii) low estimation error, and (iii) an explicit bound on worst-case estimator bias (*poisoning influence*) under *poisoning attacks*, where corrupted users may deviate from the local randomizer but are restricted to sending only protocol-admissible in-domain messages (in particular, respecting the per-user message cap).

4.1 Protocol overview

Our binary protocol has the form $\mathcal{P}^B = (\mathcal{R}^B, \mathcal{S}, \mathcal{A}^B)$. The central idea is to realize a low-variance *symmetric binomial-sum noise* distribution while explicitly restricting each user's feasible anonymous message multiset.

Preprocessing: shuffled mode assignment. We introduce a *data-independent* preprocessing step that assigns each user a *mode flag* $b_i \in \{0,1\}$, hidden from the analyzer. Concretely, the analyzer submits to the shuffler a multiset containing $n/2$ copies of 0 and $n/2$ copies of 1. The shuffler uniformly permutes these mode flags, and each user i receives one permuted mode flag b_i as auxiliary input for the local randomizer; the analyzer never observes the per-user assignment. Intuitively, this commits each user to one of two complementary noise-sampling probabilities (p vs. $1-p$) prior to reporting.

Local randomizer. Given input x_i and the assigned mode flag b_i, user i samples one noise bit $\eta_i \leftarrow \mathrm{Ber}(p_{b_i})$, where

$$p_{b_i} = \begin{cases} p, & b_i = 0, \\ 1-p, & b_i = 1, \end{cases} \qquad \text{with } p \le \tfrac{1}{2}.$$

User i then sends $x_i + \eta_i$ copies of the in-domain message 1 to the shuffler. Thus each user sends either 0, 1, or 2 copies of message 1, and in particular at most 2 messages.

Analyzer. Let $|\mathbf{Y}|$ denote the total number of 1-messages received after shuffling. The analyzer outputs

$$\tilde{f} = \frac{|\mathbf{Y}|}{n} - \frac{1}{2}.$$

This is a *constant-shift* estimator.

Noise intuition. By construction, $n/2$ users receive $b_i = 0$ and $n/2$ receive $b_i = 1$. Hence the aggregate noise satisfies

$$\sum_{i=1}^{n} \eta_i \sim \mathrm{Bin}\left(\tfrac{n}{2}, p\right) + \mathrm{Bin}\left(\tfrac{n}{2}, 1-p\right), \quad \mathbb{E}\left[\sum_{i=1}^{n} \eta_i\right] = \tfrac{n}{2}.$$

Thus, $\tilde{f}$ is unbiased for f, and the analyzer applies only a constant shift rather than a multiplicative debiasing factor that could amplify the effect of adversarial message injections.

Communication complexity. Each user sends $x_i + \eta_i$ copies of message 1 and thus at most 2 messages. Moreover, since marginally $\mathbb{E}[\eta_i] = 1/2$ for each user under the balanced mode assignment, the expected number of noise messages is $n/2$. For the all-ones input ($x_i = 1$ for all i), the expected total number of transmitted messages is $n + n/2 = 1.5n$, i.e., 1.5 messages per user on average.

4.2 Algorithms

The local randomizer and analyzer are summarized in Algs. 1 and 2. The parameter p will be instantiated as a function of (n,ε,δ) in the privacy analysis.

Algorithm 1 $\mathcal{R}^B_{b,p}$: randomizer for the binary protocol

Input: $x \in \{0,1\}$, mode flag $b \in \{0,1\}$
Output: multiset $\mathbf{y}$ consisting of copies of symbol 1
if $b = 0$ **then**
    $p_b \leftarrow p$
else
    $p_b \leftarrow 1-p$
end if
Sample $\eta \leftarrow \mathrm{Ber}(p_b)$
Append $x+\eta$ copies of message 1 to $\mathbf{y}$
Return $\mathbf{y}$

Algorithm 2 $\mathcal{A}^B$: analyzer for the binary protocol

Input: shuffled multiset $\mathbf{Y}$ consisting of copies of 1 ▷ all users' messages after shuffling
Output: estimate $\tilde{f} \in \mathbb{R}$
$\tilde{f} \leftarrow \frac{|\mathbf{Y}|}{n} - \frac{1}{2}$
Return $\tilde{f}$

4.3 Main guarantee

We now state the main theorem for the binary protocol.

THEOREM 3. For any $\varepsilon \in (0,1]$, $\delta \in (0,1)$, and $n > \frac{60}{\varepsilon^2}\log\frac{4}{\delta}$, there exists a choice of parameter $p \in (0,1/2]$ such that the protocol $\mathcal{P}^B = (\mathcal{R}^B_{b,p}, \mathcal{S}, \mathcal{A}^B)$ satisfies:

1. **Privacy.** $\mathcal{P}^B$ satisfies (ε,δ)-DP.
2. **Accuracy.** For any $\beta \in (0,1)$, with probability at least $1-\beta$ over the protocol randomness, the error is $\left|\tilde{f} - f\right| = O\left(\frac{1}{\varepsilon n}\sqrt{\log\frac{1}{\delta}\cdot\log\frac{1}{\beta}}\right)$.
3. **Robustness.** Under any poisoning attack corrupting m users, the expected estimate can change by at most $\frac{3m}{2n}$.
4. **Communication.** Each user sends at most 2 messages, and for the all-ones input the expected number of messages is 1.5 per user.

4.4 Privacy Analysis

We prove (ε,δ)-DP by a reduction to a one-dimensional additive-noise mechanism, followed by post-processing [14].

LEMMA 1 (Reduction to a one-bit additive-noise mechanism)**.** If $\mathcal{C}^B_{n,p}$ in Alg. 3 satisfies (ε,δ)-DP, then the shuffle protocol $\mathcal{P}^B = (\mathcal{R}^B_{b,p}, \mathcal{S}, \mathcal{A}^B)$ satisfies (ε,δ)-DP.

Algorithm 3 $\mathcal{C}^B_{n,p}$ (core additive-noise counting mechanism)

Input: $x \in \{0,1\}$
Output: $y \in \mathbb{N}$
$z \leftarrow 0$
for all $i \in [n/2]$ **do**
  Sample $z_1 \sim \mathrm{Ber}(p)$
  Sample $z_2 \sim \mathrm{Ber}(1-p)$
  $z \leftarrow z + z_1 + z_2$
end for
$y \leftarrow x + z$
Return y

Proof. The output is $\mathcal{M}(\mathbf{x}) = \mathcal{S}\big(\mathcal{R}^B_{b_1,p}(x_1), \ldots, \mathcal{R}^B_{b_n,p}(x_n)\big)$, where the mode-flag assignment $(b_1,\ldots,b_n)$, a uniform permutation of $n/2$ copies of 0 and $n/2$ copies of 1 sampled independently of the user data $\mathbf{x}$, serves as the auxiliary input of Def. 3. Under the non-collusion assumption, this assignment is not part of this output, so it suffices to characterize the distribution of the shuffled multiset. Fix any such assignment; given $\mathbf{x} = (x_1,\ldots,x_n) \in \{0,1\}^n$, user i samples $\eta_i \sim \mathrm{Ber}(p_{b_i})$ independently and outputs a multiset of $x_i + \eta_i$ copies of the same symbol 1.

After shuffling, the analyzer observes only the multiset of all messages. Since all shuffled messages are identical, this multiset is fully determined by the message count $|\mathbf{Y}|$,

$$T(\mathbf{x}) := |\mathbf{Y}| = \sum_{i=1}^{n}(x_i + \eta_i) = \Big(\sum_{i=1}^{n} x_i\Big) + Z,$$

where $Z := \sum_{i=1}^{n}\eta_i$ depends only on protocol randomness and is independent of $\mathbf{x}$. Any permutation applied by the shuffler is thus immaterial.

To verify DP, it suffices to prove that $T(\cdot)$ is (ε,δ)-DP, because the full protocol output (including the shuffled multiset and the analyzer output) is a deterministic post-processing of T. Indeed, $\mathcal{A}^B$ depends only on $|\mathbf{Y}| = T(\mathbf{x})$.

Now fix any neighboring datasets $\mathbf{x}, \mathbf{x}'$ that differ in exactly one record, say user j. Let $c := \sum_{i\neq j} x_i = \sum_{i\neq j} x'_i$, which is a constant for this neighboring pair. Then

$$T(\mathbf{x}) = c + \big(x_j + Z\big), \qquad T(\mathbf{x}') = c + \big(x'_j + Z\big).$$

Since adding a constant c is post-processing, it suffices to show that the one-dimensional mechanism $x \mapsto x + Z$ is (ε,δ)-DP for $x \in \{0,1\}$.

Finally, by grouping the n Bernoulli trials into $n/2$ pairs with parameters p and $1-p$, we can write $Z = \sum_{i=1}^{n/2} z_i$ where $z_i \sim \mathrm{Ber}(p) + \mathrm{Ber}(1-p)$ are i.i.d. Thus the mechanism $x \mapsto x + Z$ is exactly $\mathcal{C}^B_{n,p}$ up to post-processing. Therefore, if $\mathcal{C}^B_{n,p}$ is (ε,δ)-DP, then so is $T(\cdot)$, and hence $\mathcal{P}^B$ is (ε,δ)-DP by the post-processing theorem (Theorem 1). □

By Lemma 1, it remains to prove that $\mathcal{C}^B_{n,p}$ satisfies (ε,δ)-DP for a suitable choice of p.

THEOREM 4 (Privacy of the Symmetric Binomial-Sum Mechanism)**.** For $\varepsilon > 0$, $\delta \in (0,1)$, and $n \geq \frac{2L(3+2a)}{3a^2}$, where $a = \tanh\frac{\varepsilon}{2}$ and $L = \log\frac{4}{\delta}$, there exists a choice of parameter $p \in (0,1/2]$ such that $\mathcal{C}^B_{n,p}$ (Alg. 3) satisfies (ε,δ)-DP.

Proof. Recall that $\mathcal{C}^B_{n,p}(x) = x + Z$, where $Z = \sum_{i=1}^{n/2} z_i$, $z_i \sim \mathrm{Ber}(p) + \mathrm{Ber}(1-p)$, independently. Since the input domain is $\{0,1\}$, it suffices to verify DP for the neighboring inputs $x = 0$ and $x = 1$, i.e., for all measurable $Y \subseteq \mathbb{N}$,

$$\begin{aligned}\Pr[\,\mathcal{C}^B_{n,p}(1) \in Y\,] &\leq e^{\varepsilon}\Pr[\,\mathcal{C}^B_{n,p}(0) \in Y\,] + \delta,\\ \Pr[\,\mathcal{C}^B_{n,p}(0) \in Y\,] &\leq e^{\varepsilon}\Pr[\,\mathcal{C}^B_{n,p}(1) \in Y\,] + \delta. \qquad (1)\end{aligned}$$

Step 1: Tail bound. The noise distribution is symmetric with mean $\mathbb{E}[Z] = n/2$. Let $\overline{Z} := Z - n/2$ and $\bar{z}_i := z_i - 1$. Then $\overline{Z} = \sum_{i=1}^{n/2}\bar{z}_i$, where each $\bar{z}_i$ is zero-mean, bounded in $[-1,1]$, and has distribution

$$\Pr[\bar{z} = y] = \begin{cases} p_0 & y = -1,\\ 1-2p_0 & y = 0,\\ p_0 & y = 1,\end{cases} \qquad \text{with } p_0 = p(1-p).$$

Moreover, $\mathbb{E}[\bar{z}^2] = 2p_0$, hence $\sum_{i=1}^{n/2}\mathbb{E}[\bar{z}_i^2] = (n/2)\cdot 2p_0 = np_0$.

Fact 1 (Bernstein inequality)**.** Let $x_1,\ldots,x_m$ be independent zero-mean random variables with $|x_i| \leq M$ for all i. Then for all $t > 0$,

$$\Pr\left[\sum_{i=1}^{m} x_i \geq t\right] \leq \exp\left(-\frac{t^2/2}{\sum_{i=1}^{m}\mathbb{E}[x_i^2] + Mt/3}\right).$$

Applying Fact 1 to $\overline{Z}$ with $m = n/2$, $M = 1$, and $\sum \mathbb{E}[\overline{z}_i^2] = np_0$, we obtain

$$\Pr[\overline{Z} \geq t] \leq \exp\left(-\frac{t^2/2}{np_0 + t/3}\right).$$

By symmetry, this yields

$$\Pr[\,|Z - n/2| \geq t\,] \leq 2\exp\left(-\frac{t^2/2}{np(1-p) + t/3}\right). \quad (2)$$

For a target tail limit $\delta_t \in (0,1)$, define

$$t = \left\lceil \frac{1}{3}\log\frac{2}{\delta_t} + \sqrt{\left(\frac{1}{3}\log\frac{2}{\delta_t}\right)^2 + 2np(1-p)\log\frac{2}{\delta_t}} \right\rceil. \quad (3)$$

Then Eq. (2) implies

$$\Pr[\,|Z - n/2| > t\,] \leq \delta_t. \quad (4)$$

Step 2: Likelihood-ratio bound on the central region. Define the "good" (central) set

$$G := \{k \in \mathbb{N} : |k - n/2| \leq t\}.$$

For $k \in \mathbb{N}$, the DP inequality for the singleton event $\{k\}$ reads

$$\Pr[Z = k-1] \leq e^{\varepsilon}\Pr[Z = k] + \delta(\{k\}),$$

since $\mathcal{C}^B_{n,p}(1) = 1 + Z$ and $\mathcal{C}^B_{n,p}(0) = Z$. Thus, a sufficient condition for Eq. (1) is: (i) the tail probability outside G is at most δ_t, and (ii) for all k with $k, k-1 \in G$,

$$\frac{\Pr[Z = k]}{\Pr[Z = k-1]} \leq e^{\varepsilon_t} \quad \text{for some } \varepsilon_t \leq \varepsilon. \quad (5)$$

Since Z is symmetric about $n/2$, i.e., $\Pr[Z = k] = \Pr[Z = n-k]$, the involution $k \mapsto n+1-k$ maps the adjacent-pair range $G^* := \{k : k, k-1 \in G\}$ to itself and turns each ratio $\Pr[Z = k]/\Pr[Z = k-1]$ into its reciprocal. Hence the ratio bound on G^* controls both directions of the singleton DP inequalities; the two directions are assembled in Step 3.

To establish Eq. (5), we use the following monotonicity lemma (proved in App. A).

Lemma 2 (Monotone likelihood ratio of the symmetric Bernoulli sum)**.** Let $p \in (0, 1/2]$, and for Bernoulli distributions

$$\xi \sim \begin{bmatrix} 0 & 1 \\ p & 1-p \end{bmatrix}, \qquad \eta \sim \begin{bmatrix} 0 & 1 \\ 1-p & p \end{bmatrix},$$

let $X = \sum_{i=1}^N \xi_i + \sum_{j=1}^N \eta_j$, where all summands are independent. Then the ratio $\Pr[X = m+1]/\Pr[X = m]$ is non-increasing in m over all m with $\Pr[X = m] > 0$ and $\Pr[X = m+1] > 0$.

In our setting, Z is a sum of n Bernoulli trials: $n/2$ with parameter p and $n/2$ with parameter $1-p$. Pairing them yields i.i.d. summands $z_i = \mathrm{Ber}(p) + \mathrm{Ber}(1-p)$; equivalently, Z has the same distribution as the variable X in Lemma 2 with $N = n/2$. Hence, $\Pr[Z = m+1]/\Pr[Z = m]$ is non-increasing in m over its support.

Therefore, for every k with $k, k-1 \in G$ and $k \leq n/2$,

$$\frac{\Pr[Z = k]}{\Pr[Z = k-1]} \leq \frac{\Pr[Z = n/2 - t + 1]}{\Pr[Z = n/2 - t]},$$

and by symmetry the worst-case in-G ratio is attained at $k = n/2 - t + 1$. Define

$$\varepsilon_t := \log\left(\frac{\Pr[Z = n/2 - t + 1]}{\Pr[Z = n/2 - t]}\right). \quad (6)$$

Then Eq. (5) holds with this ε_t.

Step 3: The DP inequality. Let $Y \subseteq \mathbb{N}$ be arbitrary. On the central region, the ratio bound Eq. (5) covers every k with $k, k-1 \in G$, that is, $k \in [n/2 - t + 1, n/2 + t]$. For the shifted count $1 + Z$, the only uncovered point of $G + 1 = [n/2 - t + 1, n/2 + t + 1]$ is the right boundary $k = n/2 + t + 1$, whose mass $\Pr[1 + Z = n/2 + t + 1] = \Pr[Z = n/2 + t]$ is at most $\delta_t/2$ by the symmetric tail Eq. (2) (using $\Pr[Z \geq n/2 + t] = \frac{1}{2}\Pr[|Z - n/2| \geq t]$). Together with the tail bound Eq. (4), this gives

$$\begin{aligned}\Pr[1 + Z \in Y] &\leq e^{\varepsilon_t}\Pr[Z \in Y] + \Pr[Z \notin G] + \Pr[Z = n/2 + t] \\ &\leq e^{\varepsilon_t}\Pr[Z \in Y] + \tfrac{3}{2}\delta_t.\end{aligned}$$

The reverse direction $\Pr[Z \in Y] \leq e^{\varepsilon_t}\Pr[1 + Z \in Y] + \frac{3}{2}\delta_t$ follows by applying the above to $Y'' := n + 1 - Y$ and using $\Pr[Z = k] = \Pr[Z = n - k]$, which swaps $\Pr[Z \in Y]$ with $\Pr[1 + Z \in Y]$. Thus, $\mathcal{C}^B_{n,p}$ satisfies $(\varepsilon_t, \frac{3}{2}\delta_t)$-DP. In particular, if we choose parameters so that $\varepsilon_t \leq \varepsilon$ and $\frac{3}{2}\delta_t \leq \delta$, then $\mathcal{C}^B_{n,p}$ is (ε, δ)-DP.

Step 4: Existence of a feasible choice of p and the condition on n. Eq. (3) and Eq. (6) characterize $(\varepsilon_t, \delta_t)$ as functions of (n, p). Intuitively, increasing the noise dispersion (achieved by taking p closer to $1/2$) decreases the worst-case adjacent ratio and hence decreases ε_t. In particular, taking $p = 1/2$ gives $Z \sim \mathrm{Bin}(n, 1/2)$, and then

$$\varepsilon_t = \log\left(\frac{n/2 + t}{n/2 - t + 1}\right).$$

Choosing $\delta_t = \delta/2$ (so that $\frac{3}{2}\delta_t = \frac{3}{4}\delta \leq \delta$ per Step 3, aligning $L = \log\frac{2}{\delta_t} = \log\frac{4}{\delta}$ with the constant used in Theorems 3 and 5) and t as in Eq. (3) (with $p = 1/2$), it suffices to ensure

$$\frac{n/2 + t}{n/2 - t + 1} \leq e^{\varepsilon}.$$

Ignoring the ceiling and the $+1$ constant, consider the relaxed condition

$$\frac{n/2 + t}{n/2 - t} \leq e^{\varepsilon}.$$

This inequality is equivalent to

$$t \le \frac{n}{2}\tanh\frac{\varepsilon}{2}. \tag{7}$$

Substituting the expression for t from Eq. (3) (with $p = 1/2$) into Eq. (7), we require

$$\frac{1}{3}\log\frac{4}{\delta} + \sqrt{\left(\frac{1}{3}\log\frac{4}{\delta}\right)^2 + \frac{n}{2}\log\frac{4}{\delta}} \le \frac{n}{2}\tanh\frac{\varepsilon}{2}. \tag{8}$$

Let $a = \tanh(\varepsilon/2)$ and $L = \log\frac{4}{\delta}$. Isolating the square root in Eq. (8), squaring (valid since $na/2 - L/3 \ge 0$ for $n \ge 2L/(3a)$, implied by the bound below), and rearranging yields the sufficient condition

$$n \;\ge\; \frac{2L(3+2a)}{3a^2}. \tag{9}$$

This is exactly the lower bound stated in the theorem. Therefore, when n satisfies this condition, choosing $p = 1/2$ ensures $\varepsilon_t \le \varepsilon$ and $\delta_t \le \delta$, and the mechanism satisfies (ε,δ)-DP. For larger n, one can choose $p < 1/2$ appropriately, so a suitable $p \in (0, 1/2]$ always exists when n meets the stated bound. As $a \to 0$ (small ε), the bound asymptotes to $n \ge \frac{2L}{a^2} \approx \frac{8}{\varepsilon^2}\log\frac{4}{\delta}$. □

4.5 Accuracy Analysis

This section analyzes the accuracy of our binary protocol.

Claim 1. Fix any $p \in (0, 1/2]$ and $n \in \mathbb{N}$. For any raw data $x_1, \dots, x_n \in \{0,1\}$, the error variance of protocol $\mathcal{P}^B = (\mathcal{R}^B_{b,p}, \mathcal{S}, \mathcal{A}^B)$ is $p(1-p)/n$.

Proof. Each user contributes an independent Bernoulli noise with variance $p(1-p)$. Since the analyzer outputs the average over n users, the variance of the aggregated estimate is $1/n^2 \cdot n \cdot p(1-p) = p(1-p)/n$. □

By standard Bernstein-type concentration bounds (Fact 1 and Eq. (3)), for any $\beta \in (0,1)$, with probability at least $1-\beta$, the estimation error of $\mathcal{P}^B$ satisfies

$$\left|\tilde{f} - f\right| \le \frac{1}{3n}\log\frac{2}{\beta} + \frac{1}{n}\sqrt{\left(\frac{1}{3}\log\frac{2}{\beta}\right)^2 + 2np(1-p)\log\frac{2}{\beta}}. \tag{10}$$

The privacy of $\mathcal{P}^B$ is established by Theorem 4 above via direct Bernstein analysis on $\mathcal{C}^B_{n,p}$, giving the tightest closed-form sample size $n = O(\frac{1}{\varepsilon^2}\log\frac{1}{\delta})$ in the small-ε regime. It does not, however, pin down an explicit value of the noise parameter p as a function of (ε,δ,n), which the high-probability error bound above requires.

To obtain such an explicit p, we introduce an auxiliary mechanism $\mathcal{C}^B_{n,b,p}$ (Alg. 4) that exposes additional structure, with its output (y_1, y_2) reporting the per-group 1-counts of mode-0 and mode-1 users separately, rather than only their sum. Since the message count of $\mathcal{P}^B$ equals $y_1 + y_2$, the protocol $\mathcal{P}^B$ is a deterministic post-processing of this auxiliary mechanism. By the post-processing inequality, any (ε,δ)-DP guarantee therefore transfers to $\mathcal{P}^B$ (Claim 2); this transfer is a self-contained but looser alternative to Theorem 4. Since post-processing cannot improve accuracy, the accuracy of $\mathcal{P}^B$ is no worse than that of the auxiliary mechanism. Theorem 5 below provides this p; substituting it into the high-probability bound above then yields explicit accuracy guarantees for $\mathcal{P}^B$ in terms of (ε,δ).

Algorithm 4 $\mathcal{C}^B_{n,b,p}$

Input: $x_1, \dots, x_n \in \{0,1\}$, mode flags $(b_1, \dots, b_n)$
Output: $y_1, y_2 \in \mathbb{N}$
$y_1 = \sum_{i=1}^n \mathbb{I}(b_i)x_i + Bin(\frac{n}{2}, p)$
$y_2 = \sum_{i=1}^n \mathbb{I}(1-b_i)x_i + Bin(\frac{n}{2}, 1-p)$
Return: (y_1, y_2)

Lemma 3 (Likelihood-Ratio Inequalities)**.** Fix $\varepsilon \in (0,1]$, $p \in (0, 2/5]$, and set $\alpha = \varepsilon/2$. For any real $\mu \ge \frac{12\log 4}{\varepsilon^2}$, define

$$F_+ = \frac{1}{2}\left[\frac{(1-p)(1+\alpha+1/\mu)}{1-p(1+\alpha)} + \frac{1-p(1-\alpha)+p/\mu}{(1-p)(1-\alpha)}\right],$$
$$F_- = \frac{1}{2}\left[\frac{(1-p)(1-\alpha)}{1-p(1-\alpha)+p/\mu} + \frac{1-p(1+\alpha)}{(1-p)(1+\alpha+1/\mu)}\right].$$

Then $F_+ \le e^{\varepsilon}$ and $F_- \ge e^{-\varepsilon}$. The proof is given in App. B.

Theorem 5. For any $\varepsilon,\delta \in (0,1]$ and $n \ge \frac{60}{\varepsilon^2}\log\frac{4}{\delta}$, $\mathcal{C}^B_{n,b,p}$ (Alg. 4) with $p = \frac{24}{\varepsilon^2 n}\log\frac{4}{\delta}$ satisfies (ε,δ)-DP.

Proof. Let $n_0 := n/2$, $L := \log\frac{4}{\delta}$, $\alpha := \varepsilon/2$, and $\mu := n_0 p = \frac{12L}{\varepsilon^2}$. Since $L \ge \log 4$ for $\delta \le 1$, the given conditions imply $\mu \ge \frac{12\log 4}{\varepsilon^2}$; since $n \ge \frac{60L}{\varepsilon^2}$, they imply $p \le 2/5$. The hypotheses of Lemma 3 are therefore met.

Likelihood-ratio formula. We compute the per-output likelihood ratio $R(y_1,y_2)$ explicitly. Let $\xi \sim \mathrm{Bin}(n_0, p)$ and $\eta \sim \mathrm{Bin}(n_0, 1-p)$ be independent, and set $\bar{\eta} := n_0 - \eta \sim \mathrm{Bin}(n_0, p)$. For any $(y_1,y_2) \in \mathbb{N}^2$, the two input cases decompose as

$$\Pr[\mathcal{C}^B_{n,b,p}(0) = (y_1,y_2)] = \Pr[\xi = y_1]\Pr[\eta = y_2],$$
$$\Pr[\mathcal{C}^B_{n,b,p}(1) = (y_1,y_2)] = \frac{1}{2}\big(\Pr[\xi = y_1 - 1]\Pr[\eta = y_2] + \Pr[\xi = y_1]\Pr[\eta = y_2 - 1]\big),$$

where the input-1 case splits into two equiprobable sub-cases according to which group receives the extra count. Applying the binomial ratio identity $\frac{\Pr[\xi = y_1 - 1]}{\Pr[\xi = y_1]} = \frac{1-p}{p}\cdot\frac{y_1}{n_0 - y_1 + 1}$ and the analogous identity for η, we obtain

$$R(y_1,y_2) := \frac{\Pr[\mathcal{C}^B_{n,b,p}(1) = (y_1,y_2)]}{\Pr[\mathcal{C}^B_{n,b,p}(0) = (y_1,y_2)]} = \frac{1}{2}\left[\frac{1-p}{p}\cdot\frac{y_1}{n_0 - y_1 + 1} + \frac{p}{1-p}\cdot\frac{y_2}{n_0 - y_2 + 1}\right]. \tag{11}$$

Good events. We split the privacy analysis into a typical (high-probability) region, where the binomial counts y_1 and $n_0 - y_2$ stay near their common mean μ, and the complementary bad event, whose probability we bound by δ via Chernoff. Define the typical regions

$$G_+ := \{y_1 \le \mu(1+\alpha)+1,\ n_0 - y_2 + 1 \ge \mu(1-\alpha)\},$$
$$G_- := \{y_1 \ge \mu(1-\alpha),\ n_0 - y_2 \le \mu(1+\alpha)\}.$$

G_+ controls the upper tails relevant for the forward direction, where the $+1$ buffer in y_1 absorbs the input-1 shift; G_- controls the symmetric opposite tails for the reverse direction.

Forward direction. We bound the LR on G_+ by bounding each of the two summands in Eq. (11). The constraints $y_1 \le \mu(1+\alpha)+1$ and $n_0 - y_1 + 1 \ge n_0 - \mu(1+\alpha)$ from G_+ give

$$\frac{1-p}{p} \cdot \frac{y_1}{n_0 - y_1 + 1} \le \frac{(1-p)(\mu(1+\alpha)+1)}{p\,(n_0 - \mu(1+\alpha))} = \frac{(1-p)(1+\alpha+1/\mu)}{1-p(1+\alpha)};$$

analogously, the constraints $y_2 \le n_0 - \mu(1-\alpha)+1$ and $n_0 - y_2 + 1 \ge \mu(1-\alpha)$ from G_+ give

$$\frac{p}{1-p} \cdot \frac{y_2}{n_0 - y_2 + 1} \le \frac{1-p(1-\alpha)+p/\mu}{(1-p)(1-\alpha)}.$$

Substituting both bounds into Eq. (11) yields $R(y_1,y_2) \le F_+(\varepsilon,p,\mu) \le e^{\varepsilon}$ on G_+, where the final inequality is Lemma 3.

It remains to bound the probability of the complementary bad event G_+^c under input 1. Multiplicative Chernoff applied to the two violation events gives

$$\begin{aligned}\Pr[y_1 > \mu(1+\alpha)+1] &\le \Pr[\xi > \mu(1+\alpha)] \\ &\le \exp(-\alpha^2\mu/3) = e^{-L} = \delta/4, \\ \Pr[n_0 - y_2 + 1 < \mu(1-\alpha)] &\le \Pr[\bar{\eta} < \mu(1-\alpha)] \\ &\le \exp(-\alpha^2\mu/2) = e^{-3L/2} \le \delta/4.\end{aligned}$$

A union bound therefore yields $\Pr[G_+^c \mid P_1] \le \delta/2 < \delta$, and combining the LR bound on G_+ with this tail bound on G_+^c via the standard good-event/bad-event argument gives the forward DP inequality

$$\Pr[\mathcal{C}^B_{n,b,p}(1) \in Y] \le e^{\varepsilon} \Pr[\mathcal{C}^B_{n,b,p}(0) \in Y] + \delta.$$

Reverse direction. A symmetric derivation on G_- that uses $y_1 \ge \mu(1-\alpha)$ and $n_0 - y_2 \le \mu(1+\alpha)$ yields $R(y_1,y_2) \ge F_-(\varepsilon,p,\mu) \ge e^{-\varepsilon}$ by Lemma 3, and symmetric Chernoff bounds give $\Pr[G_-^c \mid P_0] \le \delta/2 < \delta$, so

$$\Pr[\mathcal{C}^B_{n,b,p}(0) \in Y] \le e^{\varepsilon} \Pr[\mathcal{C}^B_{n,b,p}(1) \in Y] + \delta.$$

The two directions together establish (ε,δ)-DP. □

Claim 2. If $\mathcal{C}^B_{n,b,p}$ (Alg. 4) satisfies (ε,δ)-DP, then $\mathcal{P}^B$ satisfies (ε,δ)-DP and is at least as accurate as $\mathcal{C}^B_{n,b,p}$.

Proof. The protocol $\mathcal{P}^B$ is obtained by post-processing the output of $\mathcal{C}^B_{n,b,p}$ by aggregating $y_1 + y_2$. Post-processing preserves differential privacy and cannot increase the estimation error, which proves the claim. □

Substituting the explicit p from Theorem 5 into the high-probability bound above, via the accuracy transfer of Claim 2, yields the error bound for $\mathcal{P}^B$ stated in Theorem 6 below.

Theorem 6. Under the hypotheses of Theorem 5, with probability at least $1-\beta$, the estimation error of $\mathcal{P}^B$ satisfies

$$\begin{aligned}\left|\tilde{f} - \tfrac{1}{n}\textstyle\sum x_i\right| &\le \frac{1}{3n}\log\frac{2}{\beta} + \frac{1}{n}\sqrt{\left(\frac{1}{3}\log\frac{2}{\beta}\right)^2 + 2np\log\frac{2}{\beta}} \\ &\le \frac{1}{3n}\log\frac{2}{\beta} + \frac{1}{n}\sqrt{\left(\frac{1}{3}\log\frac{2}{\beta}\right)^2 + \frac{48}{\varepsilon^2}\log\frac{4}{\delta}\log\frac{2}{\beta}} \\ &= O\left(\frac{1}{\varepsilon n}\sqrt{\log\frac{1}{\delta}\cdot\log\frac{1}{\beta}}\right).\end{aligned}$$

4.6 Robustness Analysis

We analyze robustness of the binary protocol against a poisoning adversary that corrupts m out of n users. We assume the standard *per-user message limit* holds even for corrupted users, so each user can submit at most 2 messages to the shuffler. (If an adversary could violate this limit, robustness would be unbounded without additional enforcement; see App. F.)

Estimator as a function of the total message count. Recall that every message is the same symbol 1, so the shuffled transcript is fully determined by its cardinality $M := |\mathbf{Y}|$. The estimator can be written as

$$\tilde{f} = \frac{|\mathbf{Y}|}{n} - \frac{1}{2} = \frac{M}{n} - \frac{1}{2}.$$

Therefore, for any attack strategy,

$$\left|\mathbb{E}[\tilde{f}] - \mathbb{E}[\tilde{f}^{\mathsf{Adv}}]\right| = \frac{1}{n}\left|\mathbb{E}[M] - \mathbb{E}[M^{\mathsf{Adv}}]\right|. \tag{12}$$

Decomposing honest and corrupted contributions. Let H be the set of honest users and C the set of corrupted users, with $|C| = m$ and $|H| = n - m$. Decompose the total message count in the honest execution as

$$M = \sum_{i\in H}(x_i + \eta_i) + \sum_{i\in C}(x_i + \eta_i),$$

where $\eta_i \in \{0,1\}$ is the Bernoulli noise bit (drawn according to the protocol). Under a poisoning attack, corrupted users may deviate from the local randomizer but must respect the per-user message limit. Let $t_i \in \{0,1,2\}$ denote the (possibly adversarial) number of messages sent by corrupted user $i \in C$. Then the attacked execution has total count

$$M^{\mathsf{Adv}} = \sum_{i\in H}(x_i + \eta_i) + \sum_{i\in C} t_i.$$

Since the adversary does not affect honest users' local randomness, the honest contribution cancels in expectation:

$$\mathbb{E}[M] - \mathbb{E}[M^{\mathsf{Adv}}] \;=\; \mathbb{E}\left[\sum_{i\in C}(x_i+\eta_i)\right] - \mathbb{E}\left[\sum_{i\in C} t_i\right]. \tag{13}$$

Bounding poisoning influence. For each corrupted user $i \in C$, we have $t_i \in \{0,1,2\}$ deterministically, hence

$$0 \;\le\; \sum_{i\in C} t_i \;\le\; 2m \quad\Rightarrow\quad 0 \;\le\; \mathbb{E}\left[\sum_{i\in C} t_i\right] \;\le\; 2m. \tag{14}$$

In the honest execution, each corrupted user $i \in C$ would contribute $x_i + \eta_i$ messages, where η_i is a Bernoulli bit whose expectation equals $1/2$ (because the preprocessing assigns $b_i \in \{0,1\}$ with equal marginal probability and uses $p_{b_i} \in \{p, 1-p\}$, yielding $\mathbb{E}[\eta_i] = 1/2$ for every i). Therefore,

$$\mathbb{E}\left[\sum_{i\in C}(x_i+\eta_i)\right] \;=\; \sum_{i\in C} x_i \;+\; \frac{m}{2}. \tag{15}$$

Combining Eq. (13), Eq. (14), and Eq. (15),

$$\left|\mathbb{E}[M] - \mathbb{E}[M^{\mathsf{Adv}}]\right| = \left|\left(\sum_{i\in C} x_i + \frac{m}{2}\right) - \mathbb{E}\left[\sum_{i\in C} t_i\right]\right|.$$

Since $\sum_{i\in C} x_i \in [0,m]$ and $\mathbb{E}[\sum_{i\in C} t_i] \in [0,2m]$, we obtain

$$\left|\mathbb{E}[M] - \mathbb{E}[M^{\mathsf{Adv}}]\right| \le \max\left\{\left(\sum_{i\in C} x_i + \frac{m}{2}\right) - 0,\; 2m - \left(\sum_{i\in C} x_i + \frac{m}{2}\right)\right\} = \frac{3m}{2}.$$

Plugging into Eq. (12) yields

$$\left|\mathbb{E}[\tilde{f}] - \mathbb{E}[\tilde{f}^{\mathsf{Adv}}]\right| \le \frac{3m}{2n}. \tag{16}$$

Remark 1 (Flag-free split-trial alternative)**.** $\mathcal{P}^B$ relies on the shuffler delivering the data-independent flag $b_i \in \{0,1\}$ to user i. When this preprocessing step is unavailable, an alternative protocol that fits the standard shuffle model (no auxiliary input) is the *flag-free split-trial*: each user i locally samples two independent noise bits $\eta_i^{(0)} \sim \mathrm{Ber}(p)$ and $\eta_i^{(1)} \sim \mathrm{Ber}(1-p)$ and sends $x_i + \eta_i^{(0)} + \eta_i^{(1)}$ copies of the message 1. The aggregate noise $\mathrm{Bin}(n,p) + \mathrm{Bin}(n,1-p)$ preserves the symmetric binomial-sum structure of $\mathcal{P}^B$, so the likelihood-ratio template of Theorem 5 carries over at effective sample size $2n$. The per-user message cap rises from 2 to 3, the expected number of messages per user from 1.5 to 2, and the worst-case poisoning influence loosens from $3m/(2n)$ to $2m/n$. Therefore, $\mathcal{P}^B$ remains preferable for its tighter communication and influence bounds; the flag-free split-trial serves as a fallback.

5 Histogram Protocol

We now consider frequency estimation over a finite domain $[d]$. Each user $i \in [n]$ holds $x_i \in [d]$, and the analyzer estimates the histogram $f \in [0,1]^d$, where $f_j = \frac{1}{n}\sum_{i=1}^n \mathbf{1}[x_i = j]$. As in the binary case, our designs achieve low variance via a balanced mode assignment (half using p and half using $1-p$).

A naïve histogram extension. As a baseline, one can treat the histogram as d independent binary frequency estimation and run the binary protocol for each bin in parallel, where user i contributes to bin j as if holding the bit $\mathbf{1}[x_i = j]$. This inherits the per-bin privacy/accuracy/robustness guarantees of the binary protocol. The drawback is communication, as each user must locally generate noise for every bin, yielding $O(d)$ messages per user. We therefore seek protocols with communication independent of d (up to $\log d$ bits per message).

5.1 Low-Communication Histogram Protocol

We present a low-communication histogram protocol that reduces per-user communication from the naïve $O(d)$ to $O(k)$ messages, with k typically much smaller than d. The key idea is to use a data-independent preprocessing step to *bind each honest user* to a single noise bin, so that noise is injected in aggregate across users rather than per user across all bins.

Preprocessing (shuffled bin & mode assignment). The analyzer submits to the shuffler a multiset of pairs $(j,b) \in [d] \times \{0,1\}$ such that: (i) each bin j receives the same number of assignments, and (ii) within each bin, half the assignments have $b = 0$ and half have $b = 1$ (up to rounding). The shuffler uniformly permutes these pairs and delivers one permuted pair (j_i, b_i) to each user as auxiliary input; the analyzer never observes the per-user assignment.

Algorithm 5 $\mathcal{R}^{H1}_{b,j,p}$: randomizer for low-comm. histogram

Input: $x \in [d]$, assigned pair $(j,b) \in [d] \times \{0,1\}$
Output: multiset $\mathbf{y}$ of elements from $[d]$
Append x to $\mathbf{y}$ ▷ raw report
if $b = 0$ **then**
  $p_b \leftarrow p$
else
  $p_b \leftarrow 1-p$
end if
for $t \leftarrow 1$ **to** k **do**
  if $\mathrm{Ber}(p_b) = 1$ **then**
    Append j to $\mathbf{y}$ ▷ noise for assigned bin
  end if
end for
Return multiset $\mathbf{y}$ of elements from $[d]$

Algorithm 6 $\mathcal{A}^{H1}$: analyzer for low-comm. histogram

Input: shuffled multiset $\mathbf{Y}$ of elements in $[d]$ ▷ all users' messages after shuffling
Output: estimate $\tilde{f} \in \mathbb{R}^d$
Initialize counts $C_1, \ldots, C_d \leftarrow 0$
for all $y \in \mathbf{Y}$ **do**
    $C_y \leftarrow C_y + 1$
end for
for $j \leftarrow 1$ **to** d **do**
    $\tilde{f}_j \leftarrow \frac{C_j}{n} - \frac{k}{2d}$
end for
Return $\tilde{f}$

Local randomizer. (Alg. 5) User i sends one *raw* message equal to x_i. In addition, user i runs k independent Bernoulli trials with probability $p_{b_i} \in \{p, 1-p\}$ and, on success, sends a *noise* message equal to the assigned bin j_i. Thus, each user sends at most $k+1$ in-domain messages in $[d]$.

Analyzer. (Alg. 6) Let C_j be the total number of received messages equal to j. The analyzer outputs the debiased estimate

$$\tilde{f}_j = \frac{C_j}{n} - \frac{k}{2d} \qquad (j \in [d]),$$

which subtracts the known mean noise contribution and does not attempt to identify individual noise messages.

Main result. We defer the proof to App. C.

THEOREM 7 (Low-Communication Histogram Protocol)**.** For any $\varepsilon \in (0,2]$, $\delta \in (0,1)$, and $n > \frac{120d}{\varepsilon^2} \log \frac{8}{\delta}$, let $k = \left\lceil \frac{240d}{\varepsilon^2 n} \log \frac{8}{\delta} \right\rceil$ and $p = \frac{96d}{\varepsilon^2 nk} \log \frac{8}{\delta}$. Then the histogram protocol $\mathcal{P}^{H1} = (\mathcal{R}^{H1}_{b,j,p}, \mathcal{S}, \mathcal{A}^{H1})$ satisfies:

1. **Privacy.** $\mathcal{P}^{H1}$ satisfies (ε, δ)-DP.
2. **Accuracy.** For any $\beta \in (0,1)$, with probability at least $1-\beta$ over the protocol randomness, for each fixed bin $j \in [d]$, $|\tilde{f}_j - f_j| = O\left(\frac{1}{\varepsilon n}\sqrt{\log \frac{1}{\delta} \cdot \log \frac{1}{\beta}}\right)$. In particular, $\|\tilde{f} - f\|_\infty = O\left(\frac{1}{\varepsilon n}\sqrt{\log \frac{1}{\delta} \cdot \log \frac{d}{\beta}}\right)$
3. **Robustness.** Under any poisoning attack corrupting m users (each limited to at most $k+1$ in-domain messages), the expected estimate of each bin can change by at most $\frac{(k+1)m}{n}$. The expected ℓ_1-deviation of the whole histogram estimate is bounded by $\left\|\mathbb{E}[\tilde{f}] - \mathbb{E}[\tilde{f}^{\mathsf{Adv}}]\right\|_1 \le \frac{2(k+1)m}{n}$.
4. **Communication.** Each user sends at most $k+1$ messages; the expected number is $\frac{k}{2}+1$. Each message carries a bin label in $[d]$ and thus has length $O(\log d)$.

REMARK 2. The lower bound $n > \frac{120d}{\varepsilon^2} \log \frac{8}{\delta}$ confines the protocol to the low-communication regime $k \le 2$ that matches its name; the privacy argument in App. C only requires k and p to be defined as above and establishes the same (ε, δ)-DP guarantee, for $\varepsilon \in (0,2]$, for any $n \in \mathbb{N}$, at the cost of a larger $k = \lceil (240d/(\varepsilon^2 n)) \log(8/\delta) \rceil$, which in turn raises the communication and poisoning-influence bounds.

5.2 Compressed Histogram for Large Domains

When $d \gg n$, even low-communication protocols may require a larger k to provide privacy. We therefore introduce a compressed histogram protocol that hashes values into a smaller domain $[d_h]$ and performs aggregation in the hashed space, followed by a debiasing reconstruction.

Hash family. We use a pairwise-independent hash family $\{h_{u,v}\}$ mapping $[d]$ to $[d_h]$ such that each $h_{u,v}(x)$ is uniform on $[d_h]$ and $\Pr[h_{u,v}(x) = h_{u,v}(y)] = 1/d_h$ for any $x \neq y$.

Algorithm 7 $\mathcal{R}^{H2}_{b,k,p,\gamma}$: randomizer for compressed histogram

Input: $x \in [d]$, assigned mode flag $b \in \{0,1\}$
Output: multiset $\mathbf{y}$ of triples (u, v, w) with hash seed (u, v) and label $w \in [d_h]$
Sample (u, v) uniformly
Append $(u, v, h_{u,v}(x))$ to $\mathbf{y}$
if $b = 0$ **then**
    $p_b \leftarrow p$
else
    $p_b \leftarrow 1 - p$
end if
for $t \leftarrow 1$ **to** k **do**
    if $\mathrm{Ber}(\gamma p_b) = 1$ **then**
        Sample $u, v, w \leftarrow \mathcal{U} \times \mathcal{V} \times [d_h]$ uniformly
        Append (u, v, w) to $\mathbf{y}$
    end if
end for
Return $\mathbf{y}$

Algorithm 8 $\mathcal{A}^{H2}$: analyzer for compressed histogram

Input: shuffled multiset $\mathbf{Y} \subseteq \mathcal{U} \times \mathcal{V} \times [d_h]$
Output: estimate $\tilde{f} \in \mathbb{R}^d$
for $j \in [d]$ **do**
    $C_j \leftarrow \left|\{(u, v, w) \in \mathbf{Y} : h_{u,v}(j) = w\}\right|$
    $\tilde{f}_j \leftarrow \frac{C_j/n - \gamma k/(2d_h) - 1/d_h}{1 - 1/d_h}$
end for
Return $\tilde{f}$

Protocol idea. Each user sends one raw hashed report $(u, v, h_{u,v}(x_i))$. Given mode flag b, let $p_b = p$ if $b = 0$ and $p_b = 1 - p$ otherwise. The user then runs k Bernoulli noise trials, each succeeding with probability γp_b (the rate p_b of $\mathcal{P}^{H1}$ scaled by $\gamma \in (0,1]$); on success it samples a fresh (u, v) and

$w \leftarrow [d_h]$ and sends (u, v, w). Thus the noise messages are data-independent and uniformly distributed over the compressed message space, with γ controlling the effective noise-message rate. The analyzer reconstructs unbiased estimates by subtracting the known mean noise contribution and correcting for hash collisions in expectation.

Main result. We defer the proof to App. D.

THEOREM 8 (Compressed Histogram Protocol)**.** Fix $\varepsilon \in (0,3]$, $\delta \in (0,1)$ and an integer $d_h \geq 2$. Assume the hash family $\mathcal{H} = \{h_{u,v} : [d] \to [d_h]\}$ is pairwise independent in the following exact sense: for every fixed $x \in [d]$, $h_{u,v}(x)$ is uniform in $[d_h]$ over uniform (u, v), and for every $x \neq y$, $\Pr[h_{u,v}(x) = h_{u,v}(y)] = \frac{1}{d_h}$. Let $k = \left\lceil \frac{108 d_h}{\varepsilon^2 n} \log \frac{8}{\delta} \right\rceil$, $\gamma = \frac{1}{k} \cdot \frac{108 d_h}{\varepsilon^2 n} \log \frac{8}{\delta} \in (0,1]$, and $p \in (0, 1/2]$. Then the histogram protocol $\mathcal{P}^{H2} = (\mathcal{R}^{H2}_{b,k,p,\gamma}, \mathcal{S}, \mathcal{A}^{H2})$ satisfies:

1. **Privacy.** $\mathcal{P}^{H2}$ satisfies (ε, δ)-DP.
2. **Accuracy.** For any $\beta \in (0,1)$, with probability at least $1-\beta$ over the protocol randomness, for each fixed item $j \in [d]$, $|\tilde{f}_j - f_j| = O\left(\sqrt{\frac{\log(1/\beta)}{n d_h}} + \frac{1}{\varepsilon n} \sqrt{\log \frac{1}{\delta} \cdot \log \frac{1}{\beta}} \right)$. In particular, $\|\tilde{f} - f\|_\infty = O\left(\sqrt{\frac{\log(d/\beta)}{n d_h}} + \frac{1}{\varepsilon n} \sqrt{\log \frac{1}{\delta} \cdot \log \frac{d}{\beta}} \right)$.
3. **Robustness.** Under any poisoning attack corrupting m users (each still limited to at most $k+1$ in-domain messages), the expected estimate of each item can change by at most $\frac{(k+1)m}{n(1-1/d_h)}$. Moreover, the expected ℓ_1-deviation of the whole estimate satisfies $\left\| \mathbb{E}[\tilde{f}] - \mathbb{E}[\tilde{f}^{\mathsf{Adv}}] \right\|_1 \leq \frac{d(k+1)m}{n(1-1/d_h)}$.
4. **Communication.** Each user sends at most $k+1$ messages; the expected number is $1 + \frac{\gamma k}{2} = 1 + \frac{54 d_h}{\varepsilon^2 n} \log \frac{8}{\delta}$. Each message has length $O(\log d_h)$ up to the seed-length constant.

Whereas $\mathcal{P}^{H1}$ requires $k \propto d$ noise trials, hashing lets $\mathcal{P}^{H2}$ use $k \propto d_h$, decoupling per-user communication from the domain size. The hashed dimension d_h thus tunes an accuracy–communication trade-off, with a larger d_h lowering the hash-collision error ($\propto 1/\sqrt{d_h}$) while raising the expected message count $1 + \gamma k/2$ (linear in d_h).

The compressed histogram protocol $\mathcal{P}^{H2}$ uses our symmetric binomial-sum noise inside the per-message-seed hashing wrapper of Luo et al. [25], where every report carries its own seed and Alg. 8 counts messages whose label matches the query under that message's seed. The large-domain experiment in Sec. 6 therefore compares this noise design against their blanket noise under matched hashing.

6 Experimental Evaluation

We evaluate the proposed histogram protocols on accuracy, communication, and poisoning robustness. The low-communication histogram protocol $\mathcal{P}^{H1}$ is evaluated on the IPUMS *City* dataset [28] ($d = 40$, $n = 155{,}782$), the San Francisco Fire Department *Fire* dataset [16] ($d = 272$, $n = 681{,}174$), and the ACS PUMS *Occupation* dataset [29] ($d = 529$, $n = 123{,}293$). The compressed histogram protocol $\mathcal{P}^{H2}$ is evaluated on the AOL query log [27] with $d = 2^{24}$ and $n = 10^5$. For the three categorical datasets, we fix $\delta = 10^{-6}$, consistent with the common choice $\delta < 1/n$, and vary $\varepsilon \in \{0.25, 0.5, 0.75, 1, 2, 3\}$. For AOL, we set $\delta = 10^{-10}$ and use two hashed sub-domain sizes $d_h \in \{8685, 65\}$, matching LWY's AOL parameter setting [25]. All numbers are means over 100 independent runs.

For $\mathcal{P}^{H1}$, we choose the smallest k for which the joint two-bin numerical condition in Lemma 5 holds for some $p \in (0, 1/2]$. This condition certifies (ε, δ)-DP at each evaluated ε, including $\varepsilon = 3$, which lies beyond the $\varepsilon \leq 2$ range of Theorem 7. In the AOL experiment, we calibrate both protocols with LWY's numerical search rule [25], so that the comparison is made under a single parameter-selection criterion. The resulting message counts are far smaller than implied by the conservative closed-form parameters of Theorem 8.

We measure accuracy by count-level MAE over histogram bins, reported in absolute counts rather than normalized frequencies. Communication is the average number of messages sent by one honest user, counting both data and noise messages. For robustness, we report the empirical poisoning influence, namely the increase in normalized ℓ_1 histogram error after replacing a uniformly random m/n fraction of users by corrupted users that submit the maximum number of syntactically valid messages the protocol accepts, with $m/n \in \{0.1\%, 1\%, 5\%, 10\%, 20\%, 30\%\}$. This is a lower bound on the worst-case influence $\mathrm{Infl}(m)$.

Small-domain evaluation. Tab. 2 reports accuracy (MAE) and communication (messages per user) of $\mathcal{P}^{H1}$ on the three categorical datasets across privacy levels. It shows that $\mathcal{P}^{H1}$ substantially improves accuracy over CSUZZ [10], BC [2], and CZ [11]. Across all three datasets and all reported privacy levels, the MAE reduction is between $4.3\times$ and $13.2\times$. For example, at $\varepsilon = 1$ on City, $\mathcal{P}^{H1}$ has MAE 5.4, compared with 46, 44, and 25 for CSUZZ, BC, and CZ, respectively. The same pattern holds on Fire and Occupation, even though their domain sizes differ by more than an order of magnitude. GKMP [18] attains the smallest MAE but only with a much larger message budget, ranging from thousands to millions of messages per user, placing it at a different accuracy–communication point.

The message counts should also be read together with the per-message format. CSUZZ and BC use scalar messages but require d-scale communication, while CZ sends only two messages but each message is a d-bit vector. The corresponding bit-level cost is reported in App. E.1. Thus, among protocols with short scalar messages and small expected message counts, the closest comparison is LWY [25]. At $\varepsilon = 0.25$, $\mathcal{P}^{H1}$ achieves MAE essentially equal to LWY's across the three datasets. As ε increases, $\mathcal{P}^{H1}$ becomes consistently more accu-

Table 2: Empirical comparison of accuracy and communication for representative shuffle-model protocols across datasets and privacy levels ($\delta = 10^{-6}$ and $\varepsilon \in \{0.25, 0.5, 0.75, 1, 2, 3\}$). Larger ε accentuates the MAE advantage of Ours over LWY (e.g., $\sim$48% reduction at ε=3 on all three datasets), reflecting that our symmetric binomial-sum noise avoids the right skew that LWY's blanket noise develops at small per-bin means.

	MAE						Msgs/user					
Protocol	0.25	0.5	0.75	1	2	3	0.25	0.5	0.75	1	2	3
Dataset: City with d=40 and $n\approx$156k												
CSUZZ [10]	280	101	62	46	22	14	40	40	40	40	40	40
BC [2]	148	85	58	44	22	15	29	38	40	40	41	41
GKMP [18]	7.9	4.0	2.6	1.9	0.9	0.5	416k	107k	49k	29k	8.0k	4.1k
CZ [11]	99	48	32	25	15	12	2	2	2	2	2	2
LWY [25]	21.6	11.1	7.7	6.0	3.8	3.1	1.19	1.05	1.02	1.01	1.01	1.00
Ours	22.0	10.9	7.3	5.4	2.6	1.6	1.5	1.5	1.5	1.5	1.5	1.5
Dataset: Fire with d=272 and $n\approx$681k												
CSUZZ [10]	194	89	59	43	22	14	272	272	272	272	272	272
BC [2]	169	87	58	44	22	15	254	268	271	272	273	273
GKMP [18]	8.0	4.0	2.6	1.9	0.9	0.5	647k	166k	76k	45k	12k	6.3k
CZ [11]	93	46	32	24	15	12	2	2	2	2	2	2
LWY [25]	21.6	11.0	7.6	5.9	3.7	3.1	1.29	1.08	1.04	1.02	1.01	1.01
Ours	21.4	10.8	7.1	5.3	2.6	1.6	2.0	1.5	1.5	1.5	1.5	1.5
Dataset: Occupation with d=529 and $n\approx$123k												
CSUZZ [10]	285	102	62	45	22	14	529	529	529	529	529	529
BC [2]	136	83	57	43	22	15	321	478	507	517	527	529
GKMP [18]	8.0	4.0	2.6	1.9	0.9	0.5	6.9M	1.8M	818k	480k	134k	68k
CZ [11]	102	48	32	25	15	12	2	2	2	2	2	2
LWY [25]	21.5	11.1	7.7	6.0	3.7	3.1	4.15	1.83	1.39	1.24	1.09	1.06
Ours	21.6	10.8	7.2	5.5	2.6	1.6	7.5	3.0	2.0	1.5	1.5	1.5

Table 3: Large-domain accuracy and communication comparison on AOL, with the same experimental setup as LWY [25] (d=2^{24}, n=10^5, δ=10^{-10}, $d_h \in \{8685, 65\}$). Ours matches LWY on both metrics across $\varepsilon \in \{0.25, 0.5, 0.75, 1, 2, 3\}$.

	MAE						Msgs/user					
Protocol	0.25	0.5	0.75	1	2	3	0.25	0.5	0.75	1	2	3
Hashed sub-domain d_h=$n/\log n\approx$8685												
LWY [25]	28.9	15.0	10.5	8.4	5.6	4.9	114	30.7	15.0	9.5	4.2	3.3
Ours	28.9	15.0	10.5	8.4	5.6	4.9	114	30.7	15.0	9.5	4.2	3.3
Hashed sub-domain d_h=$n/\log^3 n\approx$65												
LWY [25]	42.9	34.9	33.2	32.5	31.9	31.8	1.85	1.22	1.11	1.06	1.02	1.02
Ours	42.9	34.9	33.2	32.6	31.9	31.8	1.85	1.22	1.11	1.06	1.02	1.02

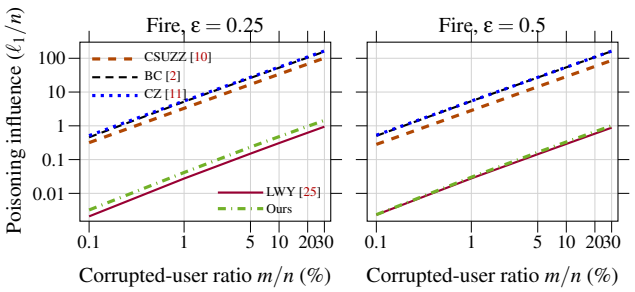

Figure 2: Poisoning influence comparison on the Fire dataset at $\varepsilon \in \{0.25, 0.5\}$. Ours and LWY remain orders of magnitude below CSUZZ, BC, and CZ across the full m/n range. App. E gives the full figure for City, Fire, and Occupation at $\varepsilon \in \{0.25, 0.5, 0.75, 1\}$.

rate. At $\varepsilon = 1$, it reduces MAE relative to LWY by 10.0% on City, 10.2% on Fire, and 8.3% on Occupation. At $\varepsilon = 2$, the reduction is about 29–32%, and at $\varepsilon = 3$, it reaches about 48% across the three datasets. This widening advantage at larger ε is consistent with the different aggregate noise shapes: LWY's centered blanket noise becomes right-skewed when the per-bin mean is small, whereas $\mathcal{P}^{H1}$ uses symmetric binomial-sum noise.

In terms of communication, $\mathcal{P}^{H1}$ remains in the same low-message range as LWY in the evaluated settings. It uses 1.5 messages per user on all three datasets once $\varepsilon \geq 1$; the strongest-privacy regime requires a few additional messages (at most 7.5 per user) to satisfy the DP target. Overall, the small-domain experiments show a favorable trade-off. Our $\mathcal{P}^{H1}$ keeps communication low, improves substantially over CSUZZ/BC/CZ, and improves over LWY at all but $\varepsilon = 0.25$, where the two achieve essentially equal MAE.

Large-domain evaluation. We use AOL ($d = 2^{24}$) to test whether the compressed protocol $\mathcal{P}^{H2}$ preserves LWY's large-domain accuracy–communication trade-off under the same settings. The larger d_h reduces hash-collision error at higher communication, while the smaller favors near-single-message communication. Running both protocols at the same d_h isolates the noise-design contribution by holding the hash-collision rate $1/d_h$ identical across the two protocols. As shown in Tab. 3, $\mathcal{P}^{H2}$ matches LWY's MAE and messages per user under this matched setup.

Poisoning-influence evaluation. Fig. 2 reports representative poisoning-influence curves on the Fire dataset under varying corrupted-user ratios m/n from 0.1% to 30%, and App. E.2 gives the full set of results for all three datasets and $\varepsilon \in \{0.25, 0.5, 0.75, 1\}$. Across these settings, the curves grow approximately linearly in m/n, as predicted by the per-user message-limit analysis. On poisoning influence, our $\mathcal{P}^{H1}$ stays close to LWY [25], the strongest baseline, and is consistently far below CSUZZ [10], BC [2], and CZ [11]. The reason is structural: CSUZZ, BC, and CZ let a corrupted user push every coordinate (via d-bit reports or d-dim vectors), so their poisoning grows with d; LWY and $\mathcal{P}^{H1}$ cap each user at a few scalar messages, bounding the coordinates corruption can touch. For example, at $m/n = 10\%$ and $\varepsilon = 0.25$ on Fire, $\mathcal{P}^{H1}$ reaches 0.48 vs CSUZZ 33.72, BC 52.39, CZ 54.80; LWY is comparable at 0.31, and the gap between $\mathcal{P}^{H1}$ and LWY narrows as ε increases. We omit GKMP [18] from Fig. 2

because it has no deterministic per-user message limit, so a corrupted user could repeat in-domain messages and make the worst-case poisoning influence unbounded by repetition.

Taken together, these results establish that our protocols achieve the best accuracy-communication-robustness trade-off, delivering up to nearly $2\times$ lower small-domain MAE than the closest baseline [25], with comparable per-user communication and poisoning influence.

7 Discussion

7.1 Practical Considerations of Preprocessing

In our protocols, preprocessing is used to instantiate the data-independent auxiliary-input distribution required by Def. 3. We discuss its impact on latency, coordination state, and user dropout in the three subsections that follow.

Latency. Because the auxiliary inputs are independent of user data, they can be generated and delivered before the analytics window starts, or batched with other round-configuration material. Thus, in scheduled collection workflows, preprocessing need not lie on the critical path of user reporting; in interactive settings, it adds one setup phase before the usual collection round and incurs no additional per-round latency.

Coordination state. The additional state is temporary and small. During preprocessing, the shuffler holds the auxiliary-input multiset and the random assignment used for delivery; this state can be erased once delivery completes. Each user stores only its own auxiliary input until reporting finishes. The analyzer needs only the public round parameters and the prescribed auxiliary-input distribution, but must not learn the per-user assignment.

User dropout. A dropped user contributes no report, and the remaining reports are processed normally. This applies whether the user drops out during preprocessing, between preprocessing and reporting, or during reporting itself. As with other shuffle protocols whose guarantees depend on the number of submitted reports, our privacy and accuracy guarantees should be instantiated for the effective reporting population. In practice, a deployment may choose parameters for a conservative lower bound on the number of reporters, or restart the round if participation falls below the required threshold.

7.2 Deployment Compatibility

The main deployment question is whether an implementation provides a setup channel for distributing data-independent auxiliary inputs before users report. Otherwise, the flag-free split-trial variant of Remark 1 remains available under the standard shuffle model, with higher communication and a looser poisoning-influence bound.

Prochlo-style deployments. A Prochlo-style Encode-Shuffle-Analyze architecture [6] is compatible with our design if auxiliary inputs are distributed as configuration material before the Encode step. The analyzer may specify the data-independent auxiliary-input multiset required by Def. 3, but must not learn which user receives which element; preprocessing then amounts to a limited extension of the existing setup channel used to configure each collection round.

Mix-net-based deployments. A canonical mix-net [8] is primarily one-way, with senders submitting messages, the mix-net anonymizes them, and recipients obtain the outputs. It does not by itself expose a shuffler-to-sender delivery path. A strictly one-way mix-net deployment would therefore need a separate setup channel to realize our preprocessing step; otherwise, the flag-free split-trial variant of Remark 1 remains the natural fallback at the trade-offs noted above.

TEE-based deployments. A shuffler implemented inside a trusted execution environment (TEE) [12] can realize the preprocessing step as enclave-internal logic. The enclave obtains or samples the data-independent auxiliary-input multiset during setup and delivers the assigned auxiliary input to each user over a channel established via remote attestation. This requires no separate setup channel, subject to the usual TEE-deployment assumptions (enclave integrity, attestation soundness, and a trusted code base).

8 Conclusion

We proposed shuffle protocols that achieve (ε, δ)-DP and strong accuracy while providing explicit poisoning-influence bounds under the per-user message-limit premise. Our design centers on a *symmetric binomial-sum noise* construction realized via a shuffled assignment of data-independent auxiliary inputs, enabling a simple constant-shift estimator and explicit worst-case bias analysis under anonymity. We provide formal privacy, accuracy, and robustness guarantees for binary estimation and histograms (including a large-domain variant) and validate the resulting accuracy–communication–robustness trade-offs on real datasets. More broadly, our results highlight a protocol-design principle for the shuffle model: beyond calibrating noise for DP, structurally limiting how much each user can report is key to provably bounding worst-case estimator bias under anonymity. From this perspective, symmetric binomial-sum noise provides a reusable design building block. Its known mean shift supports low-variance estimation without multiplicative bias amplification, and it combines with domain-reduction techniques such as hashing while retaining explicit influence bounds under the standard per-user message limit.

Ethical Considerations

Scope and intended use. This work develops and evaluates shuffle-model differential privacy protocols for frequency estimation and histogram construction, with emphasis on mitigat-

ing the impact of poisoning by corrupted users. Our defensive goal is to improve the reliability of privacy-preserving aggregate analytics under realistic adversarial behavior.

Stakeholders, benefits, and potential harms. This work affects several stakeholder groups, directly or indirectly: (i) *data subjects* represented in the public evaluation datasets, and more broadly populations whose aggregate statistics may be released using shuffle-model protocols; (ii) *deploying organizations* that operate shuffler and analyzer components or use such protocols for aggregate analytics; (iii) *downstream consumers* of the released aggregate statistics, including the communities those statistics describe; (iv) the broader *differential-privacy and shuffle-model research community*. The intended benefit across these groups is more reliable privacy-preserving aggregate analytics: stronger robustness against tampering with aggregate releases, and reusable design primitives for future shuffle-model protocols. Potential harms include dual-use concerns arising from the poisoning analysis, over-interpretation of our worst-case bounds as deployment guarantees outside the stated assumptions, deployment without the required per-user message-bound enforcement, and downstream reliance on biased or noisy aggregates.

Datasets, human subjects, and privacy. We do not recruit participants, interact with human subjects, or collect new personal data. Our small-domain evaluation uses three publicly available categorical datasets: IPUMS City (U.S. Census PUMS data, year 2023), SFFD Fire (San Francisco Fire Department call records, year 2023), and ACS PUMS Occupation (U.S. Census ACS PUMS data, year 2023). Each user contributes only a single categorical attribute (*city* for IPUMS, *Alarms* for Fire, and *occupation* for ACS PUMS Occupation), with domain sizes $d = 40$, $d = 272$, and $d = 529$, respectively. According to the dataset documentation and our use, these do not include personally identifiable information (PII) for our purposes, and we use them solely for research and benchmarking consistent with their stated research-use terms.

The large-domain evaluation uses the AOL query log [27], the standard large-domain benchmark in this line of work [25]. Although the 2006 release was withdrawn after users were re-identified from raw query text and per-user metadata, our evaluation reads only a histogram of fixed-length query prefixes over $d = 2^{24}$ keys (no raw queries, identifiers, timestamps, or click-throughs), so it retains none of the fields that enabled re-identification. Matching LWY's dataset keeps the comparison in Tab. 3 valid.

Data minimization and artifact release. To minimize privacy and misuse risks, our released artifacts include no record-level data for any of the four datasets. We release only the aggregated input histograms used by our evaluation: categorical frequency vectors over the natural attribute for the three small-domain datasets, and the 3-character query-prefix histogram described above for AOL. The released vectors contain no auxiliary fields beyond the counts needed for the studied tasks; the preprocessing scripts that read record-level inputs are not redistributed.

Dual-use considerations: poisoning analysis. Because the paper studies poisoning strategies and quantifies their influence, there is a potential dual-use concern that such analyses could inform attempts to bias aggregate statistics. We mitigate this risk by focusing on mitigation mechanisms and bounded-influence designs, and by scoping released code to research reproduction. In particular, our poisoning code is provided only as a simulation module for controlled benchmarking of robustness claims; it is not intended as tooling for attacking deployed systems. On balance, we judge publication to be net beneficial, since poisoning can undermine private aggregate analytics even when individual privacy is preserved, and openly articulating the assumptions, limitations, and bounded-influence mechanisms enables scrutiny, reproducibility, and clearer guidance for future shuffle-model protocol design.

Responsible experimentation. All experiments are conducted offline within our evaluation framework. We do not probe, attack, or measure any real-world systems without authorization, and we do not disclose vulnerabilities in specific deployed services.

Open Science

Artifact availability. We release a permanent, versioned artifact on Zenodo with a citable DOI:

https://doi.org/10.5281/zenodo.20405419

The artifact contains Python implementations of our proposed shuffle protocols and the baselines evaluated in the paper, including LWY, CSUZZ, BC, CZ, and GKMP. It also includes a top-level `README.md` describing the code structure, configuration parameters, and commands needed to reproduce the reported results.

Reproducibility scope. The released scripts reproduce the main evaluation metrics in the paper, including error, messages per user, and robustness under poisoning simulations. Because the protocols and poisoning experiments involve randomized procedures, repeated runs may exhibit small statistical variation.

Acknowledgments

We thank our shepherd and the reviewers for their constructive comments and guidance throughout the revision. This work is supported by the National Cryptologic Science Fund of China under Grant 2025NCSF01010, and the Key Program of the National Natural Science Foundation of China under Grants U25B2028 and 62432012.

References

[1] Apple Differential Privacy Team. Learning with privacy at scale. Apple Machine Learning Research, 2017. URL: https://machinelearning.apple.com/research/learning-with-privacy-at-scale.

[2] Victor Balcer and Albert Cheu. Separating local & shuffled differential privacy via histograms. In *Information-Theoretic Cryptography (ITC)*, 2020.

[3] Victor Balcer, Albert Cheu, Matthew Joseph, and Jieming Mao. Connecting robust shuffle privacy and pan-privacy. In *Symposium on Discrete Algorithms (SODA)*, pages 2384–2403, 2021.

[4] Borja Balle, James Bell, Adrià Gascón, and Kobbi Nissim. The privacy blanket of the shuffle model. In *CRYPTO*, pages 638–667, 2019.

[5] Borja Balle, James Bell, Adria Gascón, and Kobbi Nissim. Private summation in the multi-message shuffle model. In *ACM CCS*, pages 657–676, 2020.

[6] Andrea Bittau, Úlfar Erlingsson, Petros Maniatis, Ilya Mironov, Ananth Raghunathan, David Lie, Mitch Rudominer, Ushasree Kode, Julien Tinnes, and Bernhard Seefeld. Prochlo: Strong privacy for analytics in the crowd. In *Operating Systems Principles*, pages 441–459, 2017.

[7] Xiaoyu Cao, Jinyuan Jia, and Neil Zhenqiang Gong. Data poisoning attacks to local differential privacy protocols. In *USENIX Security*, pages 947–964, 2021.

[8] David L. Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Communications of the ACM*, 24(2):84–90, 1981.

[9] Albert Cheu, Adam Smith, and Jonathan Ullman. Manipulation attacks in local differential privacy. In *IEEE SP*, pages 883–900, 2021.

[10] Albert Cheu, Adam Smith, Jonathan Ullman, David Zeber, and Maxim Zhilyaev. Distributed differential privacy via shuffling. In *EUROCRYPT*, pages 375–403, 2019.

[11] Albert Cheu and Maxim Zhilyaev. Differentially private histograms in the shuffle model from fake users. In *IEEE SP*, pages 440–457, 2022.

[12] Victor Costan and Srinivas Devadas. Intel SGX explained. Cryptology ePrint Archive, Paper 2016/086, 2016. URL: https://eprint.iacr.org/2016/086.

[13] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *TCC*, pages 265–284, 2006.

[14] 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.

[15] Úlfar Erlingsson, Vasyl Pihur, and Aleksandra Korolova. Rappor: Randomized aggregatable privacy-preserving ordinal response. In *ACM CCS*, pages 1054–1067, 2014.

[16] Fire Department Calls for Service. San francisco fire department calls for service [dataset]. https://data.sfgov.org/Public-Safety/Fire-Department-Calls-for-Service/nuek-vuh3/about_data, 2023.

[17] Badih Ghazi, Noah Golowich, Ravi Kumar, Rasmus Pagh, and Ameya Velingker. On the power of multiple anonymous messages: Frequency estimation and selection in the shuffle model of differential privacy. In *EUROCRYPT*, pages 463–488, 2021.

[18] Badih Ghazi, Ravi Kumar, Pasin Manurangsi, and Rasmus Pagh. Private counting from anonymous messages: Near-optimal accuracy with vanishing communication overhead. In *Proceedings of Machine Learning Research (PMLR)*, pages 3505–3514, 2020.

[19] 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 *Proceedings of Machine Learning Research (PMLR)*, pages 3692–3701, 2021.

[20] Michael B Hawes. Implementing differential privacy: Seven lessons from the 2020 united states census. *Harvard Data Science Review*, 2(2), 2020.

[21] S. G. Hoggar. Chromatic polynomials and logarithmic concavity. *Journal of Combinatorial Theory, Series B*, 16:248–254, 1974.

[22] Yuval Ishai, Eyal Kushilevitz, Rafail Ostrovsky, and Amit Sahai. Cryptography from anonymity. In *Foundations of Computer Science (FOCS)*, pages 239–248, 2006.

[23] Oliver Johnson and Christina Goldschmidt. Preservation of log-concavity on summation. *ESAIM: Probability and Statistics*, 10:206–215, 2006. doi:10.1051/ps:2006008.

[24] Xiaoguang Li, Ninghui Li, Wenhai Sun, Neil Zhenqiang Gong, and Hui Li. Fine-grained poisoning attack to local differential privacy protocols for mean and variance estimation. In *USENIX Security*, pages 1739–1756, 2023.

[25] Qiyao Luo, Yilei Wang, and Ke Yi. Frequency estimation in the shuffle model with almost a single message. In *ACM CCS*, pages 2219–2232, 2022.

[26] Takao Murakami, Yuichi Sei, and Reo Eriguchi. Augmented shuffle protocols for accurate and robust frequency estimation under differential privacy. In *IEEE SP*, pages 3892–3911, 2025.

[27] Greg Pass, Abdur Chowdhury, and Cayley Torgeson. A picture of search. In *Proceedings of the 1st International Conference on Scalable Information Systems (InfoScale)*, pages 1–7, 2006.

[28] Sarah Flood Steven Ruggles et al. Ipums usa: Version 14.0 [dataset]. https://www.ipums.org/projects/ipums-usa/d010.V14.0, 2023.

[29] U.S. Census Bureau. American community survey public use microdata sample (pums) [dataset]. https://www.census.gov/programs-surveys/acs/microdata.html, 2023.

[30] Yongji Wu, Xiaoyu Cao, Jinyuan Jia, and Neil Zhenqiang Gong. Poisoning attacks to local differential privacy protocols for key-value data. In *USENIX Security*, pages 519–536, 2022.

A Proof of Lemma 2

Lemma 2: For Bernoulli distributions ξ and η,

$$\xi \sim \begin{bmatrix} 0 & 1 \\ p & 1-p \end{bmatrix} \qquad \eta \sim \begin{bmatrix} 0 & 1 \\ 1-p & p \end{bmatrix},$$

Let $p \in (0,1/2]$ and $X = \sum_{i=1}^{N} \xi_i + \sum_{j=1}^{N} \eta_j$, where all summands are independent. We prove that $\Pr[X = m+1]/\Pr[X = m]$ is non-increasing in m over all m with $\Pr[X = m] > 0$ and $\Pr[X = m+1] > 0$.

Proof. A Bernoulli pmf is log-concave on its contiguous support $\{0,1\}$, and log-concavity of a probability mass function on $\mathbb{Z}$ is preserved under convolution of independent random variables [23, Thm. 1.4 and Example 1.3(3)]; see also [21]. Since X is a sum of $2N$ independent Bernoulli variables, its pmf $\big(\Pr[X = m]\big)_{m=0}^{2N}$ is log-concave:

$$\Pr[X = m]^2 \geq \Pr[X = m-1]\Pr[X = m+1], \qquad 1 \leq m \leq 2N-1.$$

For a positive log-concave sequence this is equivalent to $\Pr[X = m+1]/\Pr[X = m]$ being non-increasing wherever both consecutive probabilities are positive. Because $p \in (0,1/2] \subset (0,1)$, all values $0,\ldots,2N$ have positive probability, so the adjacent ratio is non-increasing for all $m = 0,\ldots,2N-1$. □

B Proof of Lemma 3

Lemma 3: Fix $\varepsilon \in (0,1]$, $p \in (0,2/5]$, and set $\alpha = \varepsilon/2$. For any real $\mu \geq \frac{12\log 4}{\varepsilon^2}$, define

$$F_+ = \frac{1}{2}\left[\frac{(1-p)(1+\alpha+1/\mu)}{1-p(1+\alpha)} + \frac{1-p(1-\alpha)+p/\mu}{(1-p)(1-\alpha)}\right],$$
$$F_- = \frac{1}{2}\left[\frac{(1-p)(1-\alpha)}{1-p(1-\alpha)+p/\mu} + \frac{1-p(1+\alpha)}{(1-p)(1+\alpha+1/\mu)}\right].$$

Then $F_+ \leq e^{\varepsilon}$ and $F_- \geq e^{-\varepsilon}$.

Proof. Throughout, set $\lambda := 1/\log 4$.

Monotonicity reduction. Write $F_\pm = \frac{1}{2}\big(R_1^\pm + R_2^\pm\big)$, where $R_1^\pm, R_2^\pm$ denote the first and second bracketed terms of $F_\pm$ in the lemma statement. On the lemma's domain, the quotient rule gives

$$\partial_p R_1^+ = \frac{(1+\alpha+1/\mu)\alpha}{[1-p(1+\alpha)]^2} > 0,$$
$$\partial_p R_2^+ = \frac{\alpha+1/\mu}{(1-p)^2(1-\alpha)} > 0,$$
$$\partial_p R_1^- = -\frac{(1-\alpha)(\alpha+1/\mu)}{[1-p(1-\alpha)+p/\mu]^2} < 0,$$
$$\partial_p R_2^- = -\frac{\alpha}{(1-p)^2(1+\alpha+1/\mu)} < 0.$$

The partials w.r.t. $1/\mu$ are positive for R_i^+ (numerator dependence) and negative for R_i^- (denominator dependence). Hence F_+ is strictly increasing and F_- strictly decreasing in both p and $1/\mu$; both extrema are attained at $(p_*,\mu_*) := (2/5,\, 12\log 4/\varepsilon^2)$, so it suffices to verify $F_+(\varepsilon,p_*,\mu_*) \leq e^{\varepsilon}$ and $F_-(\varepsilon,p_*,\mu_*) \geq e^{-\varepsilon}$ for $\varepsilon \in [0,1]$.

Reduction to polynomials. At (p_*,μ_*), multiplying through by the positive factor $6(3-\varepsilon)(2-\varepsilon)$ (resp. $B_-D_- > 0$, where B_- and D_- are the denominators of R_1^- and R_2^-) reduces the inequalities to

$$F_+ \leq e^{\varepsilon} \iff \phi(\varepsilon) \leq 0, \qquad F_- \geq e^{-\varepsilon} \iff \psi(\varepsilon) \geq 0,$$

where, after expansion,

$$\phi(\varepsilon) = 36 + \frac{5\lambda-13}{2}\varepsilon^2 - \frac{13\lambda}{12}\varepsilon^3 - e^{\varepsilon}(36-30\varepsilon+6\varepsilon^2),$$
$$\psi(\varepsilon) = P(\varepsilon) - 2e^{-\varepsilon}Q(\varepsilon),$$
$$P(\varepsilon) = 108 + \frac{15\lambda-39}{2}\varepsilon^2 - \frac{13\lambda}{4}\varepsilon^3,$$
$$Q(\varepsilon) = 54 + 45\varepsilon + \big(9 + \frac{15\lambda}{2}\big)\varepsilon^2 + 3\lambda\varepsilon^3 + \frac{\lambda^2}{4}\varepsilon^4.$$

Forward inequality $\phi(\varepsilon) \le 0$ on $[0,1]$. Direct differentiation gives $\phi(0) = 0, \phi(1) = \frac{59}{2} + \frac{17\lambda}{12} - 12e < 0$ (using $e > 2.71$ and $\lambda < 0.73$, so $12e > 32.5$ versus $29.5 + \frac{17\lambda}{12} < 30.6$), and

$$\phi'(\varepsilon) = (5\lambda - 13)\varepsilon - \frac{13\lambda}{4}\varepsilon^2 - 6e^{\varepsilon}(1 - 3\varepsilon + \varepsilon^2),$$
$$\phi''(\varepsilon) = (5\lambda - 13) - \frac{13\lambda}{2}\varepsilon + 6e^{\varepsilon}(2 + \varepsilon - \varepsilon^2).$$

Sub-claim: $\phi'' > 0$ *on* $[0,1]$. Using $e^{\varepsilon} \ge 1 + \varepsilon$ and $(1+\varepsilon)(2+\varepsilon-\varepsilon^2) = 2 + 3\varepsilon - \varepsilon^3$,

$$\phi''(\varepsilon) \ge f(\varepsilon) := (5\lambda - 1) + (18 - \frac{13\lambda}{2})\varepsilon - 6\varepsilon^3.$$

Since $f''(\varepsilon) = -36\varepsilon \le 0$ on $[0,1]$, f is concave and $\min_{[0,1]} f = \min\{f(0), f(1)\} = \min\{5\lambda - 1, 11 - \frac{3\lambda}{2}\} > 1.6 > 0$, giving $\phi'' > 0$.

Since ϕ' is strictly increasing with $\phi'(0) = -6 < 0$ and $\phi'(1) = \frac{7\lambda}{4} - 13 + 6e > 0$, ϕ' has exactly one zero $\varepsilon^* \in (0,1)$; thus ϕ has a unique interior minimum and is monotone on each side. With $\phi(0) = 0$ and $\phi(1) < 0$, $\phi(\varepsilon) \le \max\{\phi(0), \phi(1)\} = 0$ on $[0,1]$, i.e., $F_+(\varepsilon, p_*, \mu_*) \le e^{\varepsilon}$.

Reverse inequality $\psi(\varepsilon) \ge 0$ on $[0,1]$. Direct differentiation gives $\psi(0) = 0, \psi'(0) = 2(Q(0) - Q'(0)) = 2(54 - 45) = 18 > 0, \psi(1) > 0$ and $\psi'(1) < 0$ (both verifiable from $e > 2.71$, $\lambda < 0.73$), and

$$\psi''(\varepsilon) = P''(\varepsilon) - 2e^{-\varepsilon}\big(Q - 2Q' + Q''\big)(\varepsilon).$$

Sub-claim: $\psi'' < 0$ *on* $[0,1]$. Expansion shows $(Q - 2Q' + Q'')(\varepsilon) = q_0 + q_1\varepsilon + q_2\varepsilon^2 + q_3\varepsilon^3 + q_4\varepsilon^4$ with $q_0 = 15\lambda - 18 < 0$ and $q_1, q_2, q_3, q_4 > 0$, so the polynomial is strictly increasing on $[0,1]$ and remains strictly negative throughout (since $(Q - 2Q' + Q'')(1) = 45 - \frac{27\lambda}{2} - \frac{3\lambda^2}{4} - 18 < 0$ by $e > 2.71$, $\lambda < 0.73$). Hence $-2e^{-\varepsilon}(Q - 2Q' + Q'')$ is positive, and as the product of two positive decreasing functions, monotonically decreasing with maximum $2|q_0| = 36 - 30\lambda$ at $\varepsilon = 0$. Since $P''(\varepsilon) = (15\lambda - 39) - \frac{39\lambda}{2}\varepsilon$ is also monotonically decreasing with $P''(0) = 15\lambda - 39$,

$$\begin{aligned}\psi''(\varepsilon) &\le P''(0) + 2|q_0| \\ &= (15\lambda - 39) + (36 - 30\lambda) = -3 - 15\lambda < 0.\end{aligned}$$

Since ψ' is strictly decreasing on $[0,1]$ with $\psi'(0) > 0$ and $\psi'(1) < 0$, ψ' has exactly one zero $\varepsilon^{**} \in (0,1)$; thus ψ has a unique interior maximum and is monotone on each side. With $\psi(0) = 0$ and $\psi(1) > 0$, $\psi(\varepsilon) \ge \min\{\psi(0), \psi(1)\} = 0$ on $[0,1]$, i.e., $F_-(\varepsilon, p_*, \mu_*) \ge e^{-\varepsilon}$. □

C Proof of Theorem 7

C.1 Deriving the choice of k (and p)

Lemma 4 (Choosing k for (ε,δ)-DP)**.** Fix $\varepsilon \in (0,2]$, $\delta \in (0,1)$, and $n \in \mathbb{N}$. Consider the low-communication histogram protocol $\mathcal{P}^{H1}$. Let

$$k := \left\lceil \frac{240d}{\varepsilon^2 n} \log\frac{8}{\delta} \right\rceil \quad \text{and} \quad p := \frac{96d}{\varepsilon^2 nk} \log\frac{8}{\delta}.$$

Then the per-bin mechanism induced by $\mathcal{P}^{H1}$ is $(\varepsilon/2, \delta/2)$-DP for every fixed bin, and consequently the full histogram protocol $\mathcal{P}^{H1}$ is (ε,δ)-DP.

Proof. We instantiate Theorem 5 (the corrected binary analysis, which proves both DP directions via Lemma 3) per bin.

Per-bin reduction. Fix an arbitrary bin $j \in [d]$. By preprocessing, exactly (up to rounding) n/d users are assigned to bin j, and within this group half have mode $b = 0$ and half have mode $b = 1$. Each assigned user performs k Bernoulli trials and, on success, sends the label j. Let

$$M := \frac{nk}{2d}, \qquad n_{\text{bin}} := \frac{nk}{d} = 2M.$$

Then the total noise count contributed to bin j can be written as

$$N_j = \xi_j + \eta_j, \qquad \xi_j \sim \text{Bin}(M, p), \quad \eta_j \sim \text{Bin}(M, 1-p),$$

where ξ_j, η_j are independent. This is the same symmetric binomial-sum noise structure as in the binary mechanism, with effective sample size n_{bin}.

Privacy for a fixed bin. For privacy, we analyze an auxiliary per-bin mechanism that reveals *additional* group information (exactly as in Alg. 4 for the binary case). Specifically, for bin j we consider releasing the pair of groupwise counts

$$(Y_{j,0}, Y_{j,1}) := \Big(T_{j,0} + \text{Bin}(M,p),\ \ T_{j,1} + \text{Bin}(M, 1-p)\Big),$$

where $T_{j,0}$ (resp. $T_{j,1}$) is the number of raw reports equal to j contributed by users with mode $b = 0$ (resp. $b = 1$), so that $T_{j,0} + T_{j,1} = T_j$. Conditioned on the (data-independent) preprocessing assignment, a neighboring change affects at most one user's raw contribution in at most one of these two groups, and the affected group is symmetric (probability $1/2$) due to the random balanced assignment. Hence Theorem 5 applies per bin with parameters

$$\varepsilon' := \varepsilon/2, \qquad \delta' := \delta/2, \qquad n_{\text{bin}} = 2M = \frac{nk}{d}.$$

Provided $\varepsilon \in (0,2]$ (so $\varepsilon' \le 1$) and the n bound below is met, the auxiliary per-bin release $(Y_{j,0}, Y_{j,1})$ is (ε', δ')-DP with the fixed parameter

$$p = \frac{24}{(\varepsilon')^2 n_{\text{bin}}} \log\frac{4}{\delta'} = \frac{96d}{\varepsilon^2 nk} \log\frac{8}{\delta}. \tag{17}$$

The actual per-bin count Z_j used by the analyzer equals $Z_j = Y_{j,0} + Y_{j,1}$, a post-processing of $(Y_{j,0}, Y_{j,1})$, and is therefore also (ε', δ')-DP.

Choice of k. Theorem 5's sample-size condition $n_{\text{bin}} \geq 60/(\varepsilon')^2 \log(4/\delta')$ becomes

$$\frac{nk}{d} \geq \frac{240}{\varepsilon^2} \log \frac{8}{\delta} \quad \Longleftrightarrow \quad k \geq \frac{240\,d}{\varepsilon^2 n} \log \frac{8}{\delta},$$

which is met by

$$k = \left\lceil \frac{240\,d}{\varepsilon^2 n} \log \frac{8}{\delta} \right\rceil.$$

Substituting this k into Eq. (17) gives a fixed $p \leq 2/5 \leq 1/2$ (since $96/240 = 2/5$), which meets Lemma 3's hypothesis, so the per-bin mechanism is $(\varepsilon/2, \delta/2)$-DP.

From one bin to the full histogram. A neighboring change to the input histogram modifies exactly one user's value, so it can affect the raw report in at most two bins: the old bin and the new bin. All other bins depend only on noise and are independent of the change. Applying the above per-bin guarantee with $(\varepsilon/2, \delta/2)$ to each of the at most two affected bins and composing them yields (ε, δ)-DP for the full histogram protocol $\mathcal{P}^{H1}$.

This concludes the proof. □

The closed-form k in Lemma 4 loses a factor of 2 in ε, δ from basic composition over the two affected bins. The following joint two-bin check avoids this loss; our experiments use the smallest k for which it certifies (ε, δ)-DP.

Lemma 5 (Joint two-bin numerical (ε, δ)-DP check for symmetric binomial-sum noise)**.** Fix $M \in \mathbb{N}$, $p \in (0, 1/2]$, $\varepsilon > 0$, and $\delta \in (0,1)$. Let π denote the pmf of $Z = \xi + \eta$ with $\xi \sim \text{Bin}(M, p)$ and $\eta \sim \text{Bin}(M, 1-p)$ independent, and define the boundary-extended log-ratio $g(t) := \log \frac{\pi(t-1)}{\pi(t)}$ for $t \in \{1, \ldots, 2M\}$, with $g(0) := -\infty$ and $g(2M+1) := +\infty$. If

$$\Pr_{Z_A, Z_B \sim \pi} \Big[g(1 + Z_A) - g(Z_B) > \varepsilon \Big] \leq \delta, \tag{18}$$

where Z_A, Z_B are independent, then the histogram protocol $\mathcal{P}^{H1}$ with $M = nk/(2d)$ is (ε, δ)-DP.

Proof. Neighboring datasets D, D' differ in one user's input (A under D, B under D'). Only bins A and B are affected; all other bins have identical conditional distributions under D, D' and contribute no privacy loss. Conditional on the data-independent noise routing (App. C.2), the noise count in each bin is distributed as $Z \sim \pi$, so (c_A, c_B) has the same law as $(1 + Z_A, Z_B)$ under D and $(Z_A, 1 + Z_B)$ under D' with $Z_A, Z_B \sim \pi$ independent. The privacy loss is $\log \frac{\pi(c_A - 1)\pi(c_B)}{\pi(c_A)\pi(c_B - 1)} = g(c_A) - g(c_B)$, which under D equals $g(1 + Z_A) - g(Z_B)$. Condition (18) states $\Pr_D[g(1 + Z_A) - g(Z_B) > \varepsilon] \leq \delta$, so $\mathcal{P}^{H1}$ is (ε, δ)-DP by the standard tail-probability sufficient condition [14]; the reverse direction is symmetric. □

C.2 Setup and per-bin noise characterization

We analyze the low-communication histogram protocol $\mathcal{P}^{H1} = (\mathcal{R}^{H1}, \mathcal{S}, \mathcal{A}^{H1})$. Let the domain be $[d]$. Each user holds a value $x_i \in [d]$. Let $T_j := \sum_{i=1}^{n} \mathbf{1}[x_i = j]$ denote the true count of bin j, so that $\sum_{j \in [d]} T_j = n$.

Preprocessing. The preprocessing assigns each user a pair $(j, b) \in [d] \times \{0, 1\}$: a target bin j and a mode flag b. We assume the assignment is balanced: for every bin j, exactly $n/(2d)$ users receive $(j, 0)$ and $n/(2d)$ users receive $(j, 1)$.[1]

Messages and estimator. Each user always sends one raw message (their true value). In addition, each user performs k independent Bernoulli trials with parameter $p_b := p$ if $b = 0$ and $p_b := 1 - p$ if $b = 1$; for every success, the user sends one noise message equal to the assigned bin j. Let Z_j be the total number of messages equal to j received after shuffling (raw plus noise). The analyzer outputs

$$\tilde{f}_j := \frac{Z_j}{n} - \frac{k}{2d} \qquad \text{for each } j \in [d]. \tag{19}$$

Noise distribution per bin. Fix a bin j. Let $\mathcal{U}_{j,0}$ (resp. $\mathcal{U}_{j,1}$) be the users assigned $(j, 0)$ (resp. $(j, 1)$). Define the noise count in bin j:

$$N_j := \sum_{u \in \mathcal{U}_{j,0}} \sum_{\ell=1}^{k} \text{Ber}(p) \; + \sum_{u \in \mathcal{U}_{j,1}} \sum_{\ell=1}^{k} \text{Ber}(1-p).$$

Let

$$M := \frac{nk}{2d}. \tag{20}$$

Then $|\mathcal{U}_{j,0}|k = |\mathcal{U}_{j,1}|k = M$ and we can write

$$N_j = \xi_j + \eta_j, \qquad \xi_j \sim \text{Bin}(M, p), \quad \eta_j \sim \text{Bin}(M, 1-p), \tag{21}$$

with ξ_j, η_j independent. Moreover,

$$Z_j = T_j + N_j. \tag{22}$$

Finally, since $\mathbb{E}[\xi_j] = Mp$ and $\mathbb{E}[\eta_j] = M(1-p) = M - Mp$, we have

$$\mathbb{E}[N_j] = M \quad \text{and} \quad \mathbb{E}[\tilde{f}_j] = \frac{T_j}{n}. \tag{23}$$

C.3 Privacy Analysis

We prove Item 1 of Theorem 7. The preprocessing assignment $(j_i, b_i)_{i=1}^{n}$ is a uniform permutation of $n/(2d)$ copies of each $(j, b) \in [d] \times \{0, 1\}$, sampled independently of the user data $\mathbf{x}$. The proof mirrors the binary privacy analysis with an effective per-bin sample size $n_{\text{bin}} = nk/d$, and then uses the fact that neighboring datasets affect at most two bins.

[1] If the assignment is only approximately balanced, the same analysis goes through with minor slack terms.

C.3.1 Per-bin DP via a two-count release

Fix a bin j and consider an auxiliary per-bin mechanism that releases the pair of groupwise counts

$$\mathcal{C}_{M,p}^{H1,\mathrm{aux}}(t_0,t_1) := \big(t_0+\xi,\ t_1+\eta\big), \tag{24}$$

where $\xi \sim \mathrm{Bin}(M,p)$ and $\eta \sim \mathrm{Bin}(M,1-p)$ are independent. The actual released per-bin count Z_j equals $(t_0+t_1)+(\xi+\eta)$ and is a post-processing of Eq. (24).

Lemma 6 (Per-bin DP)**.** Fix $\varepsilon \in (0,2]$ and $\delta \in (0,1)$. If

$$p = \frac{96\,d}{\varepsilon^2 nk}\log\frac{8}{\delta}, \tag{25}$$

then the (actual) per-bin release Z_j is $(\varepsilon/2,\delta/2)$-DP.

Proof. It suffices to show that the auxiliary mechanism Eq. (24) is $(\varepsilon/2,\delta/2)$-DP, since Z_j is its post-processing.

We instantiate Theorem 5 (the corrected binary analysis via Lemma 3) per bin, with parameters $\varepsilon' = \varepsilon/2$, $\delta' = \delta/2$, and group size $M = \frac{nk}{2d}$ (so that $n_{\mathrm{bin}} := 2M = \frac{nk}{d}$). Theorem 5 requires $\varepsilon' \le 1$ (so $\varepsilon \le 2$) and gives the fixed parameter

$$p = \frac{24}{(\varepsilon')^2 \cdot n_{\mathrm{bin}}}\log\frac{4}{\delta'} = \frac{96\,d}{\varepsilon^2 nk}\log\frac{8}{\delta},$$

which is exactly Eq. (25). This yields $(\varepsilon/2,\delta/2)$-DP. □

C.3.2 From per-bin DP to histogram DP

Lemma 7 (Two-bin composition)**.** Let X and X' be neighboring datasets for the histogram task. Then the histogram count vectors $(T_1,\dots,T_d)$ and $(T_1',\dots,T_d')$ differ in at most two coordinates. If each affected bin mechanism is $(\varepsilon/2,\delta/2)$-DP, then the joint release of all d bin counts is (ε,δ)-DP.

Proof. A neighboring change moves one user value from some bin a to another bin b, so only T_a decreases by 1 and T_b increases by 1; all other bins are unchanged. The protocol's per-bin noise in bin j depends only on users assigned to bin j by preprocessing, so the noise for bins a and b is generated from disjoint user sets and is independent. Applying basic DP composition to the two affected bins, each with privacy limit $(\varepsilon/2,\delta/2)$, yields (ε,δ)-DP for their joint output. All other coordinates are post-processing of unaffected counts and independent noise, and thus do not consume additional privacy limit. □

Proof of Item 1 of Theorem 7. Fix ε,δ. Choose k and p so that Eq. (25) holds. Then each affected bin is $(\varepsilon/2,\delta/2)$-DP by Lemma 6. Apply Lemma 7 to conclude that the full histogram release is (ε,δ)-DP. Finally, the analyzer output Eq. (19) is post-processing, hence preserves DP. □

Choice of k. The fixed parameter from Eq. (25) is

$$p = \frac{96\,d}{\varepsilon^2 nk}\log\frac{8}{\delta}.$$

For $p \le 2/5$ (so that Lemma 3's hypothesis $p \le 2/5$ is met when the corrected Theorem 5 is instantiated per bin), it suffices to take

$$k \ \ge\ \frac{240\,d}{\varepsilon^2 n}\log\frac{8}{\delta}. \tag{26}$$

This implies the stated communication scaling $k = O\Big(\frac{d}{n\varepsilon^2}\log\frac{1}{\delta}\Big)$.

C.4 Accuracy Analysis

We prove Item 2 of Theorem 7.

Lemma 8 (Unbiasedness and variance)**.** For each bin j, the estimator $\tilde{f}_j$ defined in Eq. (19) is unbiased and satisfies

$$\mathrm{Var}(\tilde{f}_j) = \frac{2Mp(1-p)}{n^2},$$

where $M = \frac{nk}{2d}$.

Proof. Unbiasedness follows from Eq. (23). For variance, since ξ_j and η_j are independent,

$$\begin{aligned}\mathrm{Var}(N_j) =& \mathrm{Var}(\xi_j)+\mathrm{Var}(\eta_j)\\ =& Mp(1-p)+M(1-p)p = 2Mp(1-p).\end{aligned}$$

Because $\tilde{f}_j - \frac{T_j}{n} = \frac{N_j - \mathbb{E}[N_j]}{n}$, the claim follows. □

Lemma 9 (High-probability error bound)**.** Fix $\beta \in (0,1)$. For each bin j, with probability at least $1-\beta$,

$$\begin{aligned}\left|\tilde{f}_j - \frac{T_j}{n}\right| \le& \frac{1}{3n}\log\frac{2}{\beta}\\ &+\frac{1}{n}\sqrt{\Big(\tfrac{1}{3}\log\tfrac{2}{\beta}\Big)^2 + 4Mp(1-p)\log\tfrac{2}{\beta}}.\end{aligned} \tag{27}$$

Proof. $N_j - \mathbb{E}[N_j]$ is a sum of $2M$ independent, zero-mean, $[-1,1]$-bounded random variables (each Bernoulli trial shifted by its mean). Apply Bernstein's inequality (Fact 1) to obtain a tail bound for $|N_j - \mathbb{E}[N_j]|$, then divide by n to translate it to the estimator error. The resulting bound is Eq. (27). □

Proof of Item 2 of Theorem 7. Choose parameters so that privacy holds as in App. C.3, in particular take k as in Eq. (26) and any $p \le 1/2$ satisfying (25). Plugging $M = \frac{nk}{2d}$ into Lemma 9 yields the stated per-bin error $O\Big(\frac{1}{\varepsilon n}\sqrt{\log\frac{1}{\delta}\cdot\log\frac{1}{\beta}}\Big)$. The ℓ_∞ bound then follows from Lemma 10. □

LEMMA 10 (ℓ_∞ (max) error)**.** Fix $\beta \in (0,1)$. With probability at least $1-\beta$,

$$\max_{j\in[d]} \left|\tilde{f}_j - f_j\right| = O\left(\frac{1}{\varepsilon n}\sqrt{\log\frac{1}{\delta}\cdot\log\frac{d}{\beta}}\right).$$

Proof. Apply Lemma 9 to each bin j with failure probability β/d, and take a union bound over all d bins. This replaces $\log(2/\beta)$ by $\log(2d/\beta)$, yielding the stated bound. □

C.5 Robustness Analysis

Proof of Item 3 of Theorem 7. Each honest user sends at most one raw message and at most k noise messages. A corrupted user may arbitrarily choose its messages, but is still bounded by the same message limit (one raw plus at most k noise messages) under the protocol specification. Therefore, an adversary controlling m users can inject at most $(k+1)m$ messages of its choice into any bin. Since the analyzer normalizes counts by n, the estimate of any bin changes by at most $\frac{(k+1)m}{n}$, linear in the corrupted-user count m. □

ℓ_1 robustness. Let $\Delta C \in \mathbb{Z}^d$ be the change in the (expected) message-count vector $C = (C_1,\ldots,C_d)$ caused by replacing the m corrupted users' honest messages with adversarial ones, under the same per-user message cap. For a single user, both the honest and adversarial count vectors have total mass at most $k+1$. Hence their ℓ_1-distance is at most $2(k+1)$. Summing over m users gives $\|\mathbb{E}[\Delta C]\|_1 \le 2(k+1)m$. Since $\tilde{f} = \frac{C}{n} - \frac{k}{2d}\mathbf{1}$, we obtain

$$\left\|\mathbb{E}[\tilde{f}] - \mathbb{E}[\tilde{f}^{\mathsf{Adv}}]\right\|_1 = \frac{1}{n}\|\mathbb{E}[\Delta C]\|_1 \le \frac{2(k+1)m}{n}.$$

C.6 Communication Analysis

Proof of Item 4 of Theorem 7. Every user sends exactly one raw message. For noise, each user performs k Bernoulli trials with parameter $p_b \in \{p, 1-p\}$ depending on the mode flag. Because preprocessing assigns $b=0$ and $b=1$ equally often, the average success probability over users is $(p+(1-p))/2 = 1/2$, so the expected number of noise messages per user is $k/2$. Hence the average total is $k/2+1$ messages. Each message carries a bin index in $[d]$, requiring $O(\log d)$ bits, i.e., $\lceil \log_2 d \rceil$ per message. □

D Proof of Theorem 8

Notation and estimator. Fix an item $j \in [d]$. Let $f_j = 1/n\sum_{i=1}^n \mathbf{1}[x_i = j]$ and $r_j := nf_j$. For each received message $y = (u,v,t)$ define $I_y(j) := \mathbf{1}[t = h_{u,v}(j)]$. Let $C_j := \sum_{y\in\mathbf{Y}} I_y(j)$ be the total number of messages that "match" j under their own seeds. Let $p_c := 1/d_h$ and let $\mu := n\gamma k/(2d_h)$. The analyzer outputs $\tilde{f}_j := \frac{C_j/n - \gamma k/(2d_h) - p_c}{1-p_c}$.

Hashing assumption. We assume a pairwise-independent hash family $\{h_{u,v}\}_{(u,v)\in\mathcal{U}\times\mathcal{V}}$ mapping $[d]$ to $[d_h]$ such that, for all distinct $a \neq b$ and all $t \in [d_h]$, $\Pr_{(u,v)}[h_{u,v}(a) = t] = 1/d_h$ and $\Pr_{(u,v)}[h_{u,v}(a) = h_{u,v}(b)] = 1/d_h$. Every message in $\mathcal{P}^{H2}$ uses an independently sampled seed (u,v). For notational simplicity, we state the proof for an exactly balanced mode assignment with $n/2$ users in each mode; if n is odd, the same proof applies by using the exact two mode counts and subtracting the corresponding exact mean.

Remark on collision probability. If a concrete 2-universal family only guarantees $\Pr[h_{u,v}(a) = h_{u,v}(b)] \le 1/d_h$ rather than equality, the estimator should use the exact collision probability p_{col} in place of $p_c = 1/d_h$. The proof below is written for the exact-uniform case used by the stated estimator.

D.1 Privacy Analysis

We prove Item 1 of Theorem 8. Let $\Omega := \mathcal{U}\times\mathcal{V}\times[d_h]$ be the message space. For an item $x \in [d]$, define $S_x := \{(u,v,t) \in \Omega : t = h_{u,v}(x)\}$. A raw report generated from input x is uniformly supported on S_x. A successful noise message is uniform over Ω and is independent of the dataset.

Consider two neighboring datasets that differ in one user's value, changing from x to x'. The raw reports of all unchanged users are identically distributed in the two executions, and removing them only makes the privacy analysis harder. It therefore suffices to analyze the changed raw report together with the data-independent noise messages. This is a balls-into-bins mechanism over Ω, whose input set is S_x or $S_{x'}$.

Let N_0 denote the noise-only output distribution over multisets of Ω. For a multiset $W = (w_m)_{m\in\Omega}$, let $|W| = \sum_{m\in\Omega} w_m$. Since every successful noise message is uniform over Ω, the noise-only probability has the form

$$\Pr[N_0 = W] = a_{|W|}\cdot\frac{|W|!}{\prod_{m\in\Omega} w_m!}\cdot|\Omega|^{-|W|}, \tag{28}$$

where $a_{|W|}$ is the probability that the noise process produces exactly $|W|$ successful noise messages. The coefficient $a_{|W|}$ depends only on the total number of noise messages and not on their locations.

For the full mechanism on input set S_x, for any W with positive probability,

$$\begin{aligned}\Pr[M(S_x) = W] &= \frac{1}{|S_x|}\sum_{m\in S_x}\Pr[N_0 = W - \mathbf{e}_m] \\ &= \frac{a_{|W|-1}(|W|-1)!|\Omega|^{-(|W|-1)}}{|S_x|\prod_{m\in\Omega} w_m!}\sum_{m\in S_x} w_m.\end{aligned} \tag{29}$$

Because $|S_x| = |S_{x'}|$, Eq. (29) implies

$$\frac{\Pr[M(S_x) = W]}{\Pr[M(S_{x'}) = W]} = \frac{\sum_{m\in S_x} w_m}{\sum_{m\in S_{x'}} w_m}. \tag{30}$$

Thus the bad privacy event in one direction is contained in the event $\sum_{m\in S_x} w_m \geq e^{\varepsilon}\sum_{m\in S_{x'}} w_m$.

When $W \sim M(S_x)$, the changed raw report contributes one message to S_x. Let Z_x and $Z_{x'}$ be the numbers of noise messages falling in S_x and $S_{x'}$, respectively. If $S_x \cap S_{x'} \neq \emptyset$, the changed raw report may also contribute to $S_{x'}$, which only helps privacy. Therefore it suffices to consider the worst case $S_x \cap S_{x'} = \emptyset$, where the bad event is contained in $1 + Z_x \geq e^{\varepsilon} Z_{x'}$.

For every fixed x, each noise trial in mode 0 contributes to Z_x with probability $\gamma p/d_h$, and each noise trial in mode 1 contributes to Z_x with probability $\gamma(1-p)/d_h$. Hence Z_x has the same marginal distribution as

$$Z := \mathrm{Bin}(nk/2, \gamma p/d_h) + \mathrm{Bin}(nk/2, \gamma(1-p)/d_h),$$

and $\mathbb{E}[Z] = \mu$. The variables Z_x and $Z_{x'}$ need not be independent; the argument below uses only marginal concentration bounds and a union bound.

By Chernoff bounds,

$$\Pr[Z_x \geq (1+\varepsilon/3)\mu] \leq \exp(-\mu\varepsilon^2/27),$$
$$\Pr[Z_{x'} \leq (1-\varepsilon/4)\mu] \leq \exp(-\mu\varepsilon^2/32).$$

By the choices of k and γ in Theorem 8, $\mu = \frac{54\log(8/\delta)}{\varepsilon^2}$, so

$$\exp(-\mu\varepsilon^2/27) = (\delta/8)^2 \leq \delta/2,$$
$$\exp(-\mu\varepsilon^2/32) = (\delta/8)^{54/32} \leq \delta/2,$$

for $\delta \in (0,1)$.

It remains to show that the two good concentration events rule out the bad event $1 + Z_x \geq e^{\varepsilon} Z_{x'}$. Since $\delta < 1$, the above value of μ satisfies $\mu > 18/\varepsilon^2$. The upper bound $\varepsilon \leq 3$ in the theorem arises from the second inequality below; it holds on $(0,3]$ by elementary calculus (the difference $(1-\varepsilon/4) - e^{-\varepsilon/2}$ is concave with value 0 at $\varepsilon = 0$ and $1/4 - e^{-3/2} > 0$ at $\varepsilon = 3$). For $0 < \varepsilon \leq 3$, we therefore have

$$e^{\varepsilon/3} \geq 1 + \varepsilon/3 + \varepsilon^2/18,$$
$$1 - \varepsilon/4 > e^{-\varepsilon/2}.$$

Therefore

$$\frac{1+(1+\varepsilon/3)\mu}{(1-\varepsilon/4)\mu} \leq e^{\varepsilon}.$$

Thus, if $Z_x < (1+\varepsilon/3)\mu$ and $Z_{x'} > (1-\varepsilon/4)\mu$, then $1 + Z_x < e^{\varepsilon} Z_{x'}$. By a union bound,

$$\Pr[1 + Z_x \geq e^{\varepsilon} Z_{x'}] \leq \delta.$$

Combining Eq. (30) with the tail bound above gives the DP inequality from D to D'. The reverse direction follows by the same argument with x and x' interchanged, since the roles of S_x and $S_{x'}$ are symmetric and the marginal noise distribution is the same for every item. Hence $\mathcal{P}^{H2}$ satisfies (ε, δ)-DP. This direct Chernoff proof admits $\varepsilon \leq 3$ at the cost of looser per-bin constants; Theorem 7's reduction-based proof in App. C prioritizes tightness in the small-domain regime.

REMARK 3 (Numerical calibration used in experiments)**.** The closed-form parameter choice in Theorem 8 is sufficient but conservative. For experiments, one may choose the effective noise rate $\gamma k/2$ by a data-independent numerical search. A strict implementation of the DP check for $\mathcal{P}^{H2}$ evaluates the tail condition for $Z = \mathrm{Bin}(nk/2, \gamma p/d_h) + \mathrm{Bin}(nk/2, \gamma(1-p)/d_h)$. When comparing with LWY's public implementation, we may instead use LWY's binomial checker with $X \sim \mathrm{Bin}(n, \mu/n)$ as a numerical proxy with the same mean $\mu = n\gamma k/(2d_h)$. This proxy is used only for experimental calibration and should not be described as a full exact DP verifier.

D.2 Accuracy Analysis

We prove Item 2 of Theorem 8. Fix an item $j \in [d]$. We decompose C_j into three contributions: true appearances of j, collisions from raw reports of users holding values different from j, and matches contributed by noise messages.

True appearances. Each user with $x_i = j$ contributes exactly one raw message $(u, v, h_{u,v}(j))$ that always matches j, hence contributes deterministically r_j to C_j.

Collision term from other raw reports. For a user with $x_i \neq j$, its raw message uses an independently sampled seed (u, v). By the hashing assumption, it matches j with probability p_c. Hence the total collision count from the $(n - r_j)$ non-j users is $W_j \sim \mathrm{Bin}(n - r_j, p_c)$, with $\mathbb{E}[W_j] = (n - r_j)p_c$ and $\mathrm{Var}(W_j) \leq (n - r_j)p_c$.

Noise-matching term. Let U_j be the number of noise messages that match j. For any successful noise message (u, v, w) with $w \leftarrow [d_h]$ uniformly, $\Pr[w = h_{u,v}(j)] = p_c$. Therefore $U_j = U_j^{(0)} + U_j^{(1)}$ with $U_j^{(0)} \sim \mathrm{Bin}(nk/2, \gamma p\, p_c)$ and $U_j^{(1)} \sim \mathrm{Bin}(nk/2, \gamma(1-p)p_c)$. Thus $\mathbb{E}[U_j] = n\gamma k/(2d_h) = \mu$ and $\mathrm{Var}(U_j) \leq \mu$. The variables W_j and U_j are independent because raw-report seeds and noise-generation coins and seeds are sampled independently; the concentration bound below only uses separate Bernstein bounds and a union bound.

Decomposition and unbiasedness. Combining the three contributions gives $C_j = r_j + W_j + U_j$ and $\mathbb{E}[C_j] = r_j + (n - r_j)p_c + n\gamma k/(2d_h)$. Plugging this into the estimator gives $\mathbb{E}[\tilde{f}_j] = f_j$, so the estimator is unbiased.

Concentration via Bernstein. The estimator deviation decomposes as

$$\tilde{f}_j - f_j = \frac{(W_j - \mathbb{E}[W_j]) + (U_j - \mathbb{E}[U_j])}{n(1-p_c)}.$$

Both W_j and U_j are sums of independent $[0,1]$ indicators. By Bernstein's inequality and a union bound, with probability at

least $1-\beta$,

$$|W_j - \mathbb{E}[W_j]| = O\left(\sqrt{np_c \log \tfrac{1}{\beta}} + \log \tfrac{1}{\beta}\right),$$
$$|U_j - \mathbb{E}[U_j]| = O\left(\sqrt{\mu \log \tfrac{1}{\beta}} + \log \tfrac{1}{\beta}\right).$$

Dividing by $n(1-p_c)$ and using $p_c = 1/d_h$ gives

$$|\tilde{f}_j - f_j| = O\left(\sqrt{\tfrac{\log(1/\beta)}{nd_h}} + \tfrac{\sqrt{\mu \log(1/\beta)}}{n} + \tfrac{\log(1/\beta)}{n}\right).$$

By the choices of k and γ in Theorem 8, $\mu = O\left(\frac{\log(1/\delta)}{\varepsilon^2}\right)$, so the second term becomes $O\left(\frac{\sqrt{\log(1/\delta)\log(1/\beta)}}{\varepsilon n}\right)$. The additive $O(\log(1/\beta)/n)$ term is the lower-order Bernstein term and is suppressed in the theorem statement for readability.

ℓ_∞ error. Apply the per-item bound with failure probability β/d and take a union bound over all d items. This replaces $\log(1/\beta)$ by $\log(d/\beta)$ and yields the stated $\|\tilde{f} - f\|_\infty$ guarantee, again suppressing the lower-order term $O(\log(d/\beta)/n)$.

D.3 Robustness Analysis

We prove Item 3 of Theorem 8. Under the per-user message limit, each corrupted user can contribute at most $k+1$ in-domain messages. Moreover, a corrupted user can craft each message to match a fixed target item j by choosing (u,v) and setting $t := h_{u,v}(j)$. Hence the adversary can change C_j by at most $(k+1)m$. Since $\tilde{f}_j$ is an affine function of C_j with coefficient $\frac{1}{n(1-p_c)}$, the expected estimate changes by at most $\frac{(k+1)m}{n(1-1/d_h)}$, uniformly over all d items.

ℓ_1 robustness. Summing the per-item bound above over all d items, using the trivial $\ell_1 \le d \cdot \ell_\infty$ inequality, gives

$$\|\mathbb{E}[\tilde{f}] - \mathbb{E}[\tilde{f}^{\mathsf{Adv}}]\|_1 \le \sum_{j\in[d]} \left|\Delta\mathbb{E}[\tilde{f}_j]\right| \le \frac{d(k+1)m}{n(1-1/d_h)}.$$

This bound is attained (up to lower-order honest-contribution savings) by the degenerate-seed attack $(u,v) = (0,v_0)$, $t = v_0 \bmod d_h$, under which every adversarial message $y = (u,v,t)$ has $I_y(j) = 1$ for all $j \in [d]$ and therefore each C_j increases by exactly $(k+1)m$.

If a sharper uniform bound on the bucket preimage size is available, the same argument replaces the factor d by the maximum, over all seeds (u,v) and labels $t \in [d_h]$, of $|\{j \in [d] : h_{u,v}(j) = t\}|$. In particular, if every hash bucket contains at most $\lceil d/d_h \rceil$ original-domain items, then the factor d in the above ℓ_1 bound can be replaced by $\lceil d/d_h \rceil$.

D.4 Communication Analysis

We prove Item 4 of Theorem 8. Each user always sends exactly one raw message. In addition, the user performs k independent Bernoulli trials with parameter γp_b, where $p_b \in \{p, 1-p\}$ depends on its hidden mode b, and sends one noise message for each success. Thus each user sends at most $k+1$ messages. By the balanced mode assignment, $\mathbb{E}[p_b] = (p + (1-p))/2 = 1/2$, so the expected number of noise messages per user is $k\gamma/2$. By the choice of γ in Theorem 8, this equals $\frac{54 d_h}{\varepsilon^2 n} \log \frac{8}{\delta}$. Hence the expected total number of messages is $1 + \gamma k/2$ per user.

Finally, each message contains a hash seed (u,v) and a hashed label in $[d_h]$. Equivalently, each message has length $O(\lambda_{\mathcal{H}} + \log d_h)$ bits, where $\lambda_{\mathcal{H}}$ is the seed length of the hash family; the $\log d_h$ term accounts for the hashed label drawn from $[d_h]$, with a fresh seed (u,v) per message.

E Experimental Data Supplement

This appendix complements Sec. 6 with per-protocol MAE standard deviations and the complete poisoning-influence comparison across all datasets and privacy levels.

E.1 MAE Results with Standard Deviations

Across all three datasets, the SD of MAE is modest relative to the mean (about 10–15% on City [28], 4–6% on Fire [16], and 3–4% on Occupation [29]), substantially smaller than the MAE gaps that support our comparisons; reporting SD does not change the qualitative conclusions of Sec. 6.

E.2 Complete Poisoning-Influence Comparison

Fig. 3 extends Fig. 2 (Fire, $\varepsilon \in \{0.25, 0.5\}$) to all three datasets (City, Fire, Occupation) and all four privacy levels $\varepsilon \in \{0.25, 0.5, 0.75, 1\}$, with the same corrupted-user ratios m/n from 0.1% to 30% and the same baselines. The extended grid shows no qualitative departure from Fig. 2. On every panel, the unbounded-influence baselines (CSUZZ [10], BC [2], CZ [11]) scale linearly with m and exceed LWY [25] and Ours by one to two orders of magnitude, while LWY and Ours stay close to each other in the lower portion of each panel. This consistency is expected because the worst-case poisoning influence depends only on the per-user message limit $k+1$, the corrupted-user count m, and the protocol's debiasing factor, none of which depend on the data distribution at fixed $(\varepsilon, \delta, n, d_h)$, so the ranking observed across panels reflects intrinsic differences between the noise designs rather than dataset-specific artifacts.

Cross-dataset structural verification. Fixing $\varepsilon = 0.25$ and $m/n = 10\%$ and scanning across City, Fire, and Occupation

Table 4: MAE ± SD and bits per user supplementing Tab. 2. MAE is the per-bin mean absolute error in absolute counts; SD is the standard deviation of MAE across runs. Bits/user = Msgs/user (Tab. 2) × bits/msg, where bits/msg = $\lceil \log_2 d \rceil$ for scalar-message protocols (BC, GKMP, LWY, Ours), $\lceil \log_2 d \rceil + 1$ for CSUZZ, and d for CZ.

			MAE ± SD under ε						Bits/user under ε					
Protocol	Ref.	Venue	0.25	0.5	0.75	1	2	3	0.25	0.5	0.75	1	2	3
City (d=40, n≈156k)														
CSUZZ	[10]	EUROCRYPT 2019	279.7±30.5	101.0±12.2	61.8±6.9	45.7±5.6	21.8±2.7	14.3±1.8	280	280	280	280	280	280
BC	[2]	ITC 2020	148.0±16.7	84.9±10.2	58.1±6.4	43.9±5.6	21.9±2.6	14.6±1.7	171	227	238	241	245	245
GKMP	[18]	ICML 2020	7.9±1.1	4.0±0.6	2.6±0.4	1.9±0.3	0.9±0.2	0.5±0.1	2.5M	640k	290k	170k	48k	24k
CZ	[11]	IEEE S&P 2022	98.7±11.2	48.2±5.3	32.0±3.7	24.7±3.0	14.9±1.9	12.4±1.6	80	80	80	80	80	80
LWY	[25]	CCS 2022	21.6±2.3	11.1±1.3	7.7±0.8	6.0±0.7	3.8±0.4	3.1±0.3	7	6	6	6	6	6
Ours	–	–	22.0±2.5	10.9±1.4	7.3±0.8	5.4±0.7	2.6±0.3	1.6±0.2	9	9	9	9	9	9
Fire (d=272, n≈681k)														
CSUZZ	[10]	EUROCRYPT 2019	194.1±9.2	88.9±3.9	58.6±2.7	43.4±2.0	21.6±1.0	14.4±0.7	2.7k	2.7k	2.7k	2.7k	2.7k	2.7k
BC	[2]	ITC 2020	168.7±7.1	86.7±3.9	58.4±2.7	43.9±2.0	21.9±1.1	14.7±0.7	2.3k	2.4k	2.4k	2.4k	2.5k	2.5k
GKMP	[18]	ICML 2020	8.0±0.5	4.0±0.2	2.6±0.2	1.9±0.1	0.9±0.1	0.5±0.1	5.8M	1.5M	690k	400k	110k	57k
CZ	[11]	IEEE S&P 2022	92.7±4.5	46.5±2.2	31.7±1.4	24.4±1.1	14.8±0.7	12.5±0.6	544	544	544	544	544	544
LWY	[25]	CCS 2022	21.6±1.0	11.0±0.5	7.6±0.4	5.9±0.3	3.7±0.2	3.1±0.1	12	10	9	9	9	9
Ours	–	–	21.4±1.0	10.8±0.5	7.1±0.4	5.3±0.2	2.6±0.1	1.6±0.1	18	14	14	14	14	14
Occupation (d=529, n≈123k)														
CSUZZ	[10]	EUROCRYPT 2019	285.0±9.1	101.5±3.3	62.0±2.0	45.0±1.4	21.8±0.7	14.5±0.5	5.8k	5.8k	5.8k	5.8k	5.8k	5.8k
BC	[2]	ITC 2020	135.6±4.9	83.5±2.6	57.2±1.7	43.3±1.3	21.9±0.7	14.7±0.4	3.2k	4.8k	5.1k	5.2k	5.3k	5.3k
GKMP	[18]	ICML 2020	8.0±0.3	4.0±0.2	2.6±0.1	1.9±0.1	0.9±0.1	0.5±0.03	69M	18M	8.2M	4.8M	1.3M	680k
CZ	[11]	IEEE S&P 2022	102.1±3.4	47.7±1.6	32.1±1.0	24.7±0.8	14.9±0.5	12.5±0.4	1.1k	1.1k	1.1k	1.1k	1.1k	1.1k
LWY	[25]	CCS 2022	21.5±0.7	11.1±0.4	7.7±0.3	6.0±0.2	3.7±0.1	3.1±0.1	42	18	14	12	11	11
Ours	–	–	21.6±0.7	10.8±0.4	7.2±0.3	5.5±0.2	2.6±0.1	1.6±0.1	75	30	20	15	15	15

with $d = 40,272,529$ respectively confirms the structural picture of Sec. 6. BC and CZ scale approximately linearly with d: BC's influence is 6.78, 52.39, 84.42 across the three datasets and CZ's is 8.26, 54.80, 111.90, giving City-to-Occupation factors of 12.5× and 13.5× that match the d-ratio 13.2×. LWY and $\mathcal{P}^{H1}$ stay near-flat, with LWY at 0.31, 0.31, 0.82 and Ours at 0.34, 0.48, 2.05; the growth of Ours on Occupation reflects its larger per-user cap $k+1 = 14$ required by DP calibration at small n, compared with 2 on City and Fire, rather than d-scaling. CSUZZ shows additional amplification by its $1/(1-2q)$ debiasing factor at strong privacy when $q \to 1/2$ on small-n datasets, yielding values 10.03, 33.72, 152.24. On City and Occupation it inflates above the BC/CZ ceiling, whereas on Fire, where $n \approx 681$k keeps q small, CSUZZ stays close to BC/CZ at 33.72 versus 52.39 and 54.80. Overall, these observations confirm the linear-in-d scaling of the unbounded-influence baselines and the near d-independence of our bounded-influence design under the stated message-limit premise.

F Anonymous Message Control

Motivation and goal. Our robustness analysis is stated under a standard *per-user message limit*: Each user contributes at most $k+1$ protocol messages per round. Here $k+1$ is fixed by the protocol design (i.e., the protocol syntax specifies the maximum number of messages a user may emit). In our setting, the choice of $k+1$ is driven by the protocol's privacy requirement and instantiation parameters (the user number n, the target privacy level (ε, δ), and for histogram estimation, the domain size d). Concretely, $k = 1$ for the binary protocol, $k = \left\lceil \frac{240d}{\varepsilon^2 n} \log \frac{8}{\delta} \right\rceil$ for the low-communication histogram protocol, and $k = \left\lceil \frac{108 d_h}{\varepsilon^2 n} \log \frac{8}{\delta} \right\rceil$ for the compressed histogram protocol. Accordingly, $k+1$ should be viewed as a protocol parameter that caps the maximum poisoning influence.

This per-user message-limit premise is necessary for any meaningful bounded-influence notion in the shuffle model. If a corrupted user can cause the analyzer to accept an *unbounded* number of anonymous in-domain messages, then the analyzer cannot distinguish limit-respecting reports from over-limit injections from the shuffled multiset alone. In that case, the adversary can arbitrarily bias frequency or histogram estimates by repeating crafted in-domain messages, and the worst-case poisoning influence becomes unbounded (or grows linearly with the number of accepted injections). Therefore, robustness for multi-message shuffle protocols must be interpreted under a per-user message limit that is enforced on the messages the analyzer *accepts*, rather than only on those that honest users submit.

At the same time, shuffling deliberately erases attribution, so the per-user limit is not automatically enforceable from the shuffled transcript alone. This appendix provides an *anonymous message control* mechanism that can realize

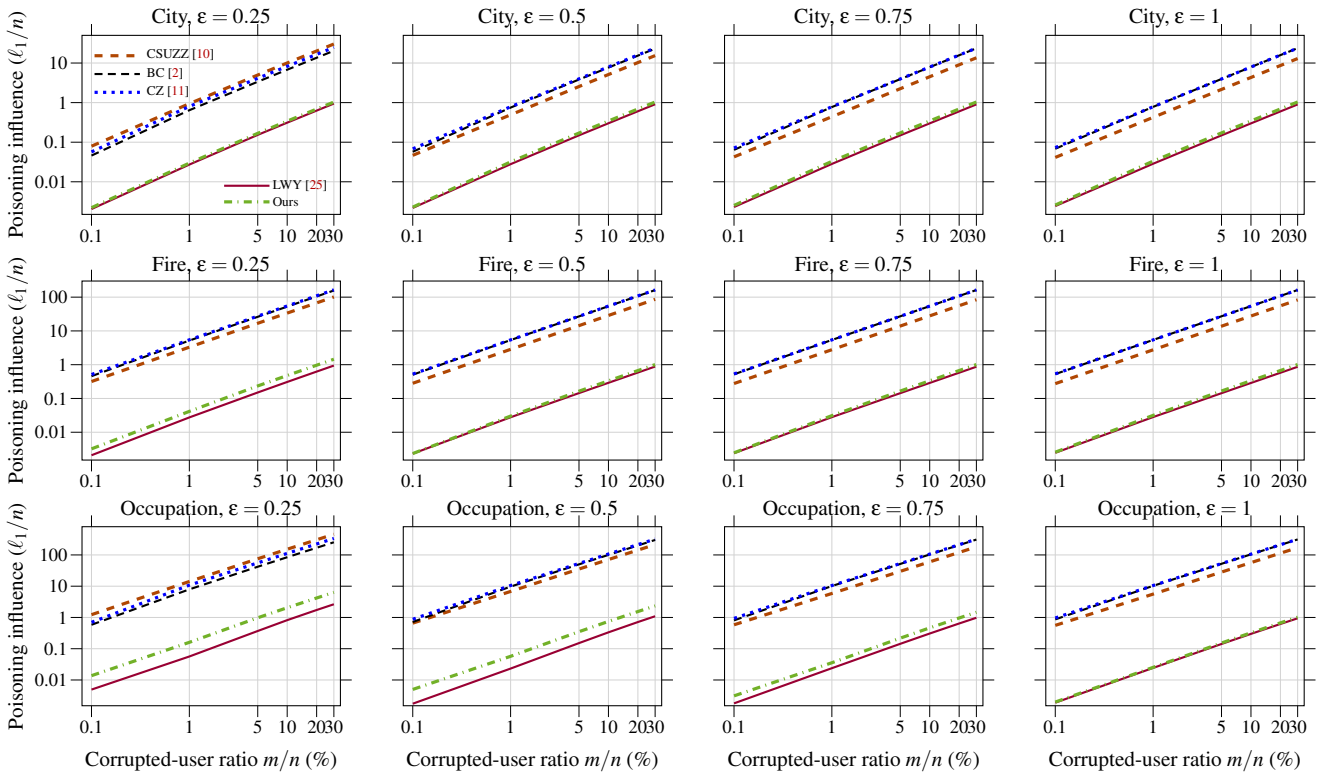

Figure 3: **Complete poisoning-influence comparison** (extends Fig. 2; metric defined in Sec. 6). Three small-domain datasets—City [28] (d=40, n≈156k), Fire [16] (d=272, n≈681k), and Occupation [29] (d=529, n≈123k)—at $\varepsilon \in \{0.25, 0.5, 0.75, 1\}$, $\delta = 10^{-6}$. GKMP [18] is omitted (no per-user message cap; unbounded influence by repetition).

the message-limit premise in a deployment, while remaining orthogonal to our protocol design and to all proofs.

We assume the standard shuffle deployment where the shuffler does not collude with the analyzer and provides anonymization by permuting messages. This anonymous message control mechanism uses the shuffler as an anonymous distribution channel in a setup phase, where the analyzer provides a pool of authenticated one-time tokens to the shuffler, the shuffler uniformly permutes them, and then delivers exactly $k+1$ token pairs to each user. The goal is that the analyzer accepts at most $k+1$ messages per user, *without* requiring sender identification from the shuffled multiset.

Anonymous message control mechanism. Fix a round identifier eid and a per-user message limit $k+1$.

1. **Signature component.** There is a standard digital signature scheme $(\mathsf{KeyGen}, \mathsf{Sign}, \mathsf{Vfy})$. The analyzer samples a key pair $(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KeyGen}(1^\lambda)$ once, keeps sk secret, and publishes pk.

2. **Token generation.** The analyzer samples $n(k+1)$ fresh one-time tokens $\tau_1, \ldots, \tau_{n(k+1)} \leftarrow \{0,1\}^\lambda$, where λ is a security parameter, and authenticates each token by signing
$$\sigma_i \leftarrow \mathsf{Sign}(\mathsf{sk}, \mathsf{eid} \,\|\, \tau_i) \qquad \text{for } i \in [n(k+1)].$$
Binding tokens to eid scopes them to the current round.

3. **Anonymous distribution.** The analyzer sends the multiset $\{(\tau_i, \sigma_i)\}_{i=1}^{n(k+1)}$ to the shuffler. The shuffler uniformly permutes the multiset and distributes exactly $k+1$ token pairs to each registered participant; let $\mathcal{T}_u$ denote the set received by participant u, with $|\mathcal{T}_u| = k+1$. Tokens carry no explicit user identifiers.

4. **Message report.** Each protocol message must include an unused token pair:
$$(m, \tau, \sigma) \quad \text{where} \quad (\tau, \sigma) \in \mathcal{T}_u.$$

5. **Validation and replay protection.** Upon receiving shuffled tuples (m, τ, σ), the analyzer maintains a per-round set $\mathsf{Spent}_{\mathsf{eid}}$ of already-used tokens and:
   (a) checks validity: $\mathsf{Vfy}(\mathsf{pk}, \mathsf{eid} \,\|\, \tau, \sigma) = 1$;
   (b) checks freshness: $\tau \notin \mathsf{Spent}_{\mathsf{eid}}$;
   (c) if both pass, accepts m and inserts τ into $\mathsf{Spent}_{\mathsf{eid}}$; otherwise, discards the tuple.

Token validation and replay checks depend only on $(\mathsf{eid}, \tau, \sigma)$ and are independent of the user data encoded in m. The analyzer's output is computed from the multiset of messages that pass validation; invalid tuples are discarded without being processed by the estimator.

We now discuss several aspects of the mechanism.

Per-user message-limit enforcement. Under signature unforgeability and correct maintenance of $\mathsf{Spent}_{\mathsf{eid}}$, each valid token can authorize at most one accepted message, and corrupted participants cannot mint additional valid tokens. Thus, an adversary corrupting m out of the n token-bearing participants can cause the analyzer to accept at most $(k+1) \cdot m$ protocol messages in a round, matching the per-user message limit assumed in our analysis.

Scope of the guarantee. The above enforcement bounds the number of accepted messages per token-bearing participant in a round. If an adversary can obtain token allocations for additional identities (Sybil participants), then our robustness bounds scale with the number of such identities; Sybil resistance and admission control are orthogonal and assumed. We further assume the setup-phase token delivery provides integrity to each participant's token bundle; compromise or theft of another participant's tokens can only increase the adversary's effective token budget and is outside our model. The analyzer maintains an independent $\mathsf{Spent}_{\mathsf{eid}}$ for each round identifier eid.

Anonymity and consistency with shuffling. Tokens carry no explicit user identifiers, and token validation occurs only after the shuffler's permutation. Under the standard non-collusion assumption, the analyzer learns no sender attribution from tokens beyond what is already abstracted away by the shuffle model. As with standard shuffle deployments, we do not address side channels (e.g., timing, network metadata, or setup-phase delivery leakage).

Deployment compatibility. The mechanism requires a one-time token bundle to reach each user before reporting. In implementations that include a preprocessing step (Sec. 7.2), the tokens can be delivered through the same setup channel as the data-independent auxiliary inputs. Their one-time use and unlinkability ensure verification adds no user-identity linkage beyond what the shuffle model permits.

Availability. This mechanism does not prevent denial-of-service attacks that drop or delay messages; such availability issues are orthogonal to our work, which conditions on the analyzer's accepted multiset.

Overhead. Per accepted message, this mechanism adds one signature verification and one membership test in $\mathsf{Spent}_{\mathsf{eid}}$. Per round, the analyzer stores up to $n \cdot (k+1)$ spent tokens. In communication, each reported message carries an additional (τ, σ) pair, and the setup phase delivers $n(k+1)$ such pairs through the shuffler. For the short scalar messages our protocols use, this authentication payload can dominate the per-message cost.