Interpolation-Based Optimization for Enforcing ℓ_p-Norm Metric Differential Privacy in Continuous and Fine-Grained Domains

Chenxi Qiu
University of North Texas

Abstract

Metric Differential Privacy (mDP) generalizes Local Differential Privacy (LDP) by adapting privacy guarantees based on pairwise distances, enabling context-aware protection and improved utility. While existing optimization-based methods reduce utility loss effectively in coarse-grained domains, optimizing mDP in fine-grained or continuous settings remains challenging due to the computational cost of constructing dense perturbation matrices and satisfying pointwise constraints.

In this paper, we propose an *interpolation-based* framework for optimizing ℓ_p-norm mDP in such domains. Our approach optimizes perturbation distributions at a sparse set of anchor points and interpolates distributions at non-anchor locations via log-convex combinations, which provably preserve mDP. To address privacy violations caused by naive interpolation in high-dimensional spaces, we decompose the interpolation process into a sequence of one-dimensional steps and derive a corrected formulation that enforces ℓ_p-norm mDP by design. We further explore joint optimization over perturbation distributions and privacy budget allocation across dimensions. Experiments on real-world location datasets demonstrate that our method offers rigorous privacy guarantees and competitive utility in fine-grained domains, outperforming baseline mechanisms.

1 Introduction

Privacy-preserving data sharing is increasingly important in applications such as location-based services (LBSs), mobility prediction, and user modeling. These applications highly depend on fine-grained data representations, which are also required to comply with stringent privacy constraints. Standard mechanisms like *Local Differential Privacy (LDP)* [15] enforce uniform privacy across all input pairs, often introducing excessive noise and degrading utility, especially in spatial, continuous, or structured domains. *Metric Differential Privacy (mDP)* [9] addresses this limitation by relaxing LDP through a distance-aware formulation: It requires stronger indistinguishability for nearby records and permits weaker guarantees for distant ones. This flexibility enables improved privacy-utility trade-offs and has been applied to protect geo-location data [3], word embeddings [20], speech [22], and image data [12, 16].

Since the introduction of mDP [9], numerous *pre-defined noise mechanisms* have been proposed to enforce distance-based privacy guarantees. Notably, the planar Laplace mechanism [3] achieves ℓ_2-based geo-indistinguishability by adding two-dimensional noise scaled to the sensitivity of location queries, while the Exponential Mechanism (EM) [11] selects outputs based on a fixed utility function that favors locations closer to the true input. These methods are efficient and well-suited for continuous and fine-grained domains. However, their fixed noise distributions often lead to suboptimal privacy-utility trade-offs, as they fail to account for direction-dependent or context-specific variations in utility.

To address the limitations of pre-defined noise mechanisms, recent work has explored *optimization-based* approaches, most notably *linear programming (LP)*, to directly optimize the perturbation distribution by minimizing expected utility loss subject to mDP constraints [6, 23, 29]. However, LP-based formulations are typically restricted to coarse-grained domains: solving them over continuous or fine-grained spaces is computationally prohibitive. A common workaround is to discretize the domain (e.g., using uniform grids or road-map features), which improves tractability but weakens formal guarantees: discretization can overestimate distances between nearby records, thereby loosening the effective mDP constraints and overlooking fine-grained privacy leakage [10].

To balance utility and efficiency, hybrid methods combine optimization with pre-defined noise. For instance, *Bayesian remapping* [10] post-processes outputs of a fixed mechanism via Bayes' rule to improve utility; *ConstOPT* [23] and *LR-Geo* [31] reduce complexity by constraining the search space (e.g., locality- or structure-aware parameterizations). Nonetheless, these approaches have notable limitations: Bayesian remapping depends on fixed noise priors and cannot guaran-

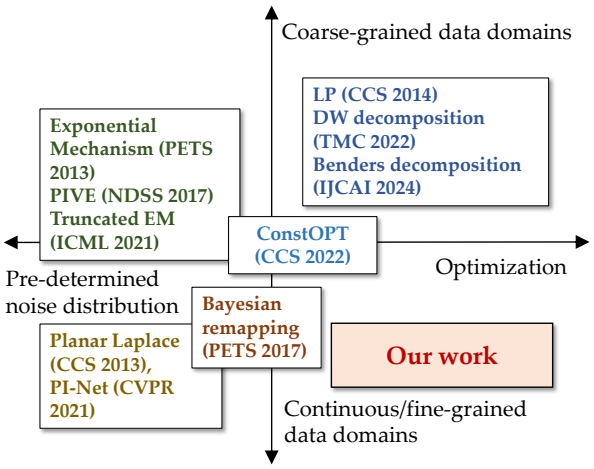


Figure 1: Related works vs. our work.
Example works in the figure: EM [9], Planar Laplace [3], LP [6], PI-Net (Laplace) [12], PIVE (EM-based) [38], Danzig-Wolfe (DW) decomposition (LP-based) [32], Benders decomposition (LP-based) [29], ConstOPT (EM+LP) [23], Bayesian Remapping [10], Truncated EM [7].

tee global optimality, while ConstOPT and LR-Geo *still incur* substantial computational overhead and remain tied to discretization, which limits scalability and can compromise strict mDP enforcement in fine-grained or continuous domains.

Fig. 1 situates prior mDP methods along two axes, domain granularity and mechanism design, and shows that most either (i) rely on pre-defined perturbation distributions suited to continuous spaces or (ii) perform optimization only after coarse discretization. This reveals a clear gap: scalable, optimization-based mechanisms that operate efficiently in continuous/fine-grained domains.

Our contributions. In this paper, we take a first step toward addressing this gap by introducing a new *interpolation-based* framework for optimizing mDP in continuous or fine-grained domains under general ℓ_p-norm distance metrics (e.g., ℓ_1, ℓ_2, ℓ_∞). The flexibility of ℓ_p-norms allows our method to accommodate diverse applications. Our framework consists of three main steps: (1) partitioning the N-dimensional secret domain into disjoint cells and selecting their corners as *anchor records* to approximate the full domain; (2) solving an *Anchor Perturbation Optimization (APO)* problem to compute the optimal perturbation probabilities at these anchor points; and (3) interpolating the perturbation probabilities for non-anchor records using a *log-convex function*, specifically a weighted geometric mean of the anchor probabilities, which ensures smooth and privacy-preserving transitions across the domain.

Naively applying log-convex interpolation in high dimensions does not by itself enforce ℓ_p-mDP: because the ℓ_p norm is convex for $p \geq 1$, combining anchor-wise bounds during interpolation can enlarge the effective distance and thus violate the *global Lipschitz condition* required by mDP [24]. We avoid this by decomposing the N-dimensional interpolation into axis-aligned, one-dimensional log-convex steps (**Definition 3**). For each axis we establish a one-dimensional Lipschitz bound (**Propositions 1–2**), then distribute a total privacy budget across coordinates using a dimension-wise composition rule (**Theorem 1**; e.g., $\sum_\ell \varepsilon_\ell^{p/(p-1)} \leq \varepsilon^{p/(p-1)}$ for $p > 1$). We combine the per-axis interpolants through a product construction (**Definition 6**) and show that the resulting mechanism satisfies a global Lipschitz guarantee with respect to the ℓ_p metric (**Theorem 2**). After normalization, this yields a valid high-dimensional perturbation mechanism that preserves the target mDP guarantee up to a small constant-factor slack quantified in our analysis.

Moreover, to minimize utility loss within this framework, we formulate the *Anchor Perturbation Optimization (APO)* problem, which jointly optimizes the perturbation probabilities at anchor points and the allocation of dimension-wise privacy budgets. Due to the non-convex nature of the APO objective, we propose a tractable linear approximation, termed *Approx-APO*, and provide theoretical bounds on its optimality gap relative to the original formulation.

We evaluate our method on real-world road network datasets from Rome, New York City, and London [1], considering both the ℓ_2 norm (Euclidean distance) and the ℓ_1 norm (Manhattan distance) for measuring spatial proximity. Experimental results show that our approach enforces strictly stronger privacy guarantees (0% mDP violations) than coarse-grained LP-based mechanisms [6, 23], while consistently achieving lower utility loss compared to pre-defined noise mechanisms (e.g., EM [7, 11], Laplace [3], and *Truncated EM* [7]), and hybrid methods (e.g., ConstOPT [23] and Bayesian remapping [10]).

Our main contribution can be summarized as follows:

- ▷ We propose a novel interpolation-based framework for enforcing ℓ_p-norm mDP in continuous and fine-grained domains, bridging the gap between rigorous privacy guarantees and practical utility.
- ▷ We introduce a dimension-wise composition strategy for ℓ_p-mDP and design a log-convex interpolation mechanism, proving its theoretical validity under (ε, d_p)-mDP (**Theorems 1–2** and **Proposition 3**).
- ▷ We formulate the *Anchor Perturbation Optimization (APO)* problem, which jointly optimizes anchor perturbations and privacy budget allocation, and propose a linear approximation with provable optimality bounds.
- ▷ Across mobility datasets from Rome, New York City, and London, our method shows zero observed mDP violations in fine-grained settings and consistently achieves lower utility loss than competing approaches.

The remainder of the paper is organized as follows. Section 2 introduces the preliminaries of mDP optimization. Section 3 outlines the overall framework, while Sections 4 and 5 present the algorithmic design for one-dimensional and multi-dimensional interpolation, respectively. Section 6 evaluates the performance of the proposed algorithm. Section 7 discusses related work, and Section 8 concludes the paper.

2 Preliminaries

In this section, we introduce the *formal definition* of mDP (**Section 2.2**), describe the *perturbation optimization framework* (**Section 2.2**), and highlight the *limitations* of existing optimization-based approaches (**Section 2.3**).

2.1 Metric Differential Privacy

Local Differential Privacy (LDP) enforces the same level of *indistinguishability* for every pair of inputs, regardless of how similar they are [15]. *Metric differential privacy* (mDP), also called *Lipschitz privacy* [24], $d_{\mathcal{X}}$-privacy [18], or *smooth DP* [14], generalizes this idea by tying the privacy guarantee to a distance over the input space: nearby inputs must be harder to distinguish than far-apart ones. Originally proposed for location privacy [3], mDP naturally extends to high-dimensional continuous domains via distance-aware privacy control.

We consider a continuous (or fine-grained) secret domain $\mathcal{X} \subseteq \mathbb{R}^N$, where each record $\mathbf{x}_a \in \mathcal{X}$ is an N-dimensional vector $\mathbf{x}_a = [x_{a,1}, \ldots, x_{a,N}]$. We measure similarity between any $\mathbf{x}_a, \mathbf{x}_b \in \mathcal{X}$ using the ℓ_p distance

$$d_p(\mathbf{x}_a, \mathbf{x}_b) \coloneqq \Big(\sum_{\ell=1}^{N} |x_{a,\ell} - x_{b,\ell}|^p \Big)^{1/p}, \tag{1}$$

which provides a flexible family of metrics (e.g., ℓ_1, ℓ_2, ℓ_∞) for capturing task-specific sensitivity. By tuning p, the metric captures not only the *overall magnitude* of deviations, but also how deviations are *distributed across dimensions*: values $1 < p < 2$ provide a compromise between the robustness of ℓ_1 and the smoothness of ℓ_2, whereas $p > 2$ places increasing weight on the largest coordinate differences, making it well suited for peak-driven signals such as electricity loads. For example, [13] adopts $p = 5$ to compare residential load profiles, treating two profiles as similar when their dominant deviations align. Unless otherwise stated, we focus on ℓ_p-based metrics in what follows.

Definition 1 (Lipschitz bound and continuity w.r.t. d_p)**.** *Let $f : \mathcal{X} \to \mathbb{R}$ and let d_p be the ℓ_p distance on $\mathbb{R}^N$ with $p \in [1, \infty]$.*

- ▷ ***Pairwise Lipschitz bound.*** *We say f satisfies an* (ε, d_p)-Lipschitz bound *between $\mathbf{x}_a, \mathbf{x}_b \in \mathcal{X}$ if*

$$|f(\mathbf{x}_a) - f(\mathbf{x}_b)| \leq \varepsilon d_p(\mathbf{x}_a, \mathbf{x}_b). \tag{2}$$

- ▷ ***Lipschitz continuity.*** *We say f is* (ε, d_p)-Lipschitz continuous *if the bound in (2) holds for all $\mathbf{x}_a, \mathbf{x}_b \in \mathcal{X}$.*

Definition 2 ((ε, d_p)-metric differential privacy (mDP))**.** *Let $\mathcal{X} \subseteq \mathbb{R}^N$ be the secret (input) domain and let $\mathcal{Y}$ denote the perturbation (output) domain. A randomized mechanism $\mathcal{M} : \mathcal{X} \to \mathcal{Y}$ is said to satisfy* (ε, d_p)-mDP *(or ℓ_p-*norm mDP*) if,*

- ▷ *for each element $\mathbf{y} \in \mathcal{Y}$, the log-probability function $f_{\mathbf{y}}(\mathbf{x}) \coloneqq \ln \Pr[\mathcal{M}(\mathbf{x}) = \mathbf{y}]$ of the output is* (ε, d_p)-Lipschitz continuous*, i.e., $\forall \mathbf{x}_a, \mathbf{x}_b \in \mathcal{X}$*

$$\big|\ln \Pr[\mathcal{M}(\mathbf{x}_a) = \mathbf{y}] - \ln \Pr[\mathcal{M}(\mathbf{x}_b) = \mathbf{y}]\big| \leq \varepsilon d_p(\mathbf{x}_a, \mathbf{x}_b), \tag{3}$$

- ▷ *and $\forall \mathbf{x} \in \mathcal{X}$, the* normalization constraint *of its perturbation probability is satisfied, i.e.,*

$$\sum_{\mathbf{y} \in \mathcal{Y}} \Pr[\mathcal{M}(\mathbf{x}) = \mathbf{y}] = 1. \tag{4}$$

Threat Model. We assume an adversary that observes the obfuscated output $\mathbf{y} \in \mathcal{Y}$ released by the mechanism $\mathcal{M}$ and attempts to infer the user's true input $\mathbf{x} \in \mathcal{X}$. The adversary may have arbitrary auxiliary knowledge about the domain $\mathcal{X}$ (e.g., road networks, prior distributions) and the mechanism $\mathcal{M}$, but does not control $\mathcal{M}$. The (ε, d_p)-mDP guarantee ensures that the likelihood ratio between any two possible inputs $\mathbf{x}_a, \mathbf{x}_b$ is bounded in proportion to their distance $d_p(\mathbf{x}_a, \mathbf{x}_b)$, thereby limiting the adversary's ability to distinguish between nearby inputs.

2.2 Perturbation Discretization and Optimization

While pre-defined noise mechanisms can enforce mDP over continuous domains, they do not explicitly optimize for utility. Optimization-based methods address this by minimizing utility loss, but exact solutions in continuous or fine-grained spaces are computationally intractable. To enable tractable optimization, prior work discretizes the input domain into a finite set of representative points. For instance, [6, 34] partition geographic regions into uniform grids and represent each cell by its centroid. Similarly, [29, 32] project raw locations onto road network features (e.g., intersections and junctions) to define a discrete optimization space.

Formally, let $\hat{\mathcal{X}} \subset \mathcal{X}$ denote the finite set of representative records obtained by discretizing the continuous domain $\mathcal{X}$. Let $p(\mathbf{x})$ be the prior distribution over $\mathcal{X}$, representing the probability that the secret record is located at $\mathbf{x} \in \mathcal{X}$. We consider the case that perturbation domain $\mathcal{Y}$ is discrete, and model the randomized mechanism $\mathcal{M}$ as a stochastic *perturbation matrix* $\mathbf{Z} = \{z(\mathbf{y}_k \mid \hat{\mathbf{x}}_i)\}_{(\hat{\mathbf{x}}_i, \mathbf{y}_k) \in \hat{\mathcal{X}} \times \mathcal{Y}}$, where each entry $z(\mathbf{y}_k \mid \hat{\mathbf{x}}_i)$ denotes the probability of reporting output $\mathbf{y}_k \in \mathcal{Y}$ given input $\hat{\mathbf{x}}_i \in \hat{\mathcal{X}}$, i.e., $z(\mathbf{y}_k \mid \hat{\mathbf{x}}_i) = \Pr[\mathcal{M}(\hat{\mathbf{x}}_i) = \mathbf{y}_k]$. The Lipschitz bound constraint in Eq. (3) is then enforced over all

pairs of records in the discretized domain $\hat{\mathcal{X}}$, and corresponds to the following set of linear constraints:

$$z(\mathbf{y}_k \mid \hat{\mathbf{x}}_i) - e^{\varepsilon d_p(\hat{\mathbf{x}}_i, \hat{\mathbf{x}}_j)} z(\mathbf{y}_k \mid \hat{\mathbf{x}}_j) \leq 0,\ \forall \hat{\mathbf{x}}_i, \hat{\mathbf{x}}_j \in \hat{\mathcal{X}},\ \forall \mathbf{y}_k \in \mathcal{Y}. \tag{5}$$

We define $\mathcal{L}(\hat{\mathbf{x}}_i, \mathbf{y}_k)$ as the utility loss incurred when the mechanism reports $\mathbf{y}_k$ while the true input lies in the region of $\mathcal{X}$ represented by the discretized point $\hat{\mathbf{x}}_i$; let $p(\hat{\mathbf{x}}_i)$ denote the prior probability that the true input lies in that region (i.e., the probability mass of the cell corresponding to $\hat{\mathbf{x}}_i$, with $\sum_i p(\hat{\mathbf{x}}_i) = 1$). Then, the expected utility loss caused by the perturbation matrix $\mathbf{Z}$ can be represented by

$$\mathcal{L}(\mathbf{Z}) = \sum_{\hat{\mathbf{x}}_i \in \hat{\mathcal{X}}} \sum_{\mathbf{y}_k \in \mathcal{Y}} p(\hat{\mathbf{x}}_i) z(\mathbf{y}_k | \hat{\mathbf{x}}_i) \mathcal{L}(\hat{\mathbf{x}}_i, \mathbf{y}_k). \tag{6}$$

Consequently, the goal of the *mDP optimization* problem is to minimize the expected utility loss $\mathcal{L}(\mathbf{Z})$ subject to the constraints imposed by mDP (*Lipschitz bound* and *normalization* constraints). This can be formulated as the following *linear programming (LP)* problem:

$$\begin{aligned}
\min \quad & \mathcal{L}(\mathbf{Z}) && (7)\\
\text{s.t.} \quad & z(\mathbf{y}_k \mid \hat{\mathbf{x}}_i) - e^{\varepsilon d_p(\hat{\mathbf{x}}_i, \hat{\mathbf{x}}_j)} z(\mathbf{y}_k \mid \hat{\mathbf{x}}_j) \leq 0, \\
& \forall \hat{\mathbf{x}}_i, \hat{\mathbf{x}}_j \in \hat{\mathcal{X}}, \forall \mathbf{y}_k \in \mathcal{Y} \text{ (Lipschitz bound)} && (8)\\
& \sum_{\mathbf{y}_k \in \mathcal{Y}} z(\mathbf{y}_k \mid \hat{\mathbf{x}}_i) = 1, \forall \hat{\mathbf{x}}_i \in \hat{\mathcal{X}} \text{ (Normalization)} && (9)\\
& z(\mathbf{y}_k \mid \hat{\mathbf{x}}_i) \geq 0,\ \forall \hat{\mathbf{x}}_i \in \hat{\mathcal{X}},\ \forall \mathbf{y}_k \in \mathcal{Y} && (10)
\end{aligned}$$

where the *non-negativity constraint* in Eq. (10) enforces that each individual probability is non-negative [33].

2.3 Limitations of Discretization-Based mDP Enforcement

While discretization-based methods significantly reduce the computational cost of solving LPs, they do not ensure strict compliance with mDP in continuous domains or high-resolution settings. The key limitation arises from distance overestimation: discretized representative points (e.g., grid cell centers) may misrepresent the true distance between original inputs, especially when records lie near cell boundaries. This can lead to relaxed mDP constraints that are satisfied by the mechanism, but fail to hold for the underlying continuous/fine-grained domain.

Fig. 2 illustrates this limitation. Suppose that a user moves from location $\mathbf{x}_a$ to a nearby location $\mathbf{x}_b$ in an adjacent grid cell, and let $\hat{\mathbf{x}}_i$ and $\hat{\mathbf{x}}_j$ denote the corresponding cell centers. Prior methods [6, 28, 34] discretize the domain by approximating each location with its cell center, so that $\Pr[\mathcal{M}(\mathbf{x}) = \mathbf{y}] = \Pr[\mathcal{M}(\hat{\mathbf{x}}) = \mathbf{y}]$, and they enforce $\frac{\Pr[\mathcal{M}(\hat{\mathbf{x}}_i)=\mathbf{y}]}{\Pr[\mathcal{M}(\hat{\mathbf{x}}_j)=\mathbf{y}]} \leq e^{\varepsilon d_p(\hat{\mathbf{x}}_i, \hat{\mathbf{x}}_j)}$, $\forall \mathbf{y} \in \mathcal{Y}$. However, near cell boundaries one typically has $d_p(\hat{\mathbf{x}}_i, \hat{\mathbf{x}}_j) > d_p(\mathbf{x}_a, \mathbf{x}_b)$. Thus, the center-based enforcement is *weaker* than the true constraint:

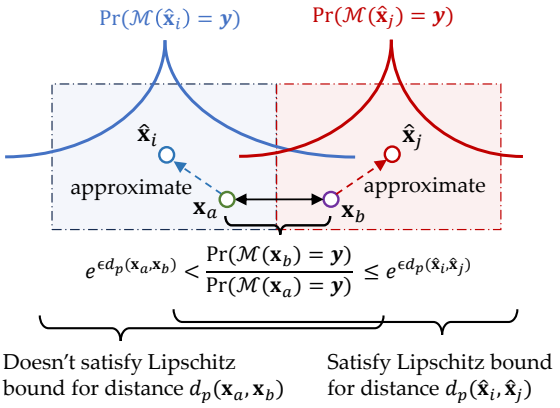


Figure 2: mDP based on approximated distance.

$\frac{\Pr[\mathcal{M}(\mathbf{x}_a)=\mathbf{y}]}{\Pr[\mathcal{M}(\mathbf{x}_b)=\mathbf{y}]} \leq e^{\varepsilon d_p(\mathbf{x}_a, \mathbf{x}_b)}$ in Definition 2. Consequently, there may exist some $\mathbf{y}$ such that $\frac{\Pr[\mathcal{M}(\mathbf{x}_a)=\mathbf{y}]}{\Pr[\mathcal{M}(\mathbf{x}_b)=\mathbf{y}]} \leq e^{\varepsilon d_p(\hat{\mathbf{x}}_i, \hat{\mathbf{x}}_j)}$ but $> e^{\varepsilon d_p(\mathbf{x}_a, \mathbf{x}_b)}$, which violates the Lipschitz bound between the true locations $\mathbf{x}_a$ and $\mathbf{x}_b$. In essence, mDP requires that small changes in the input domain induce only $e^{\varepsilon d_p}$-bounded changes in output probabilities, an invariant that coarse discretization may fail to preserve.

3 Framework

In this section, we introduce our interpolation-based framework for optimizing perturbation mechanisms under ℓ_p-norm mDP in continuous or fine-grained secret domains. As illustrated in Fig. 3, the framework consists of three main steps:

① **Domain partitioning**: The secret domain $\mathcal{X}$ is partitioned into cells, with anchor records placed at each cell corner to serve as representative inputs for optimization.

② **Anchor perturbation optimization**: Perturbation distributions are optimized at anchor locations to minimize expected utility loss, subject to mDP constraints over the anchor set.

③ **Perturbation interpolation**: For non-anchor records, perturbation probabilities are inferred via interpolation from anchor distributions, while preserving mDP guarantees.

Intuition behind interpolation design. In Step ③, let $\mathcal{C}_m$ denote a cell in the partitioned domain, and let $\hat{\mathcal{X}}_m$ denote the set of anchor points within $\mathcal{C}_m$. Suppose the perturbation distributions at these anchor points, denoted by $\mathbf{Z}_{\hat{\mathcal{X}}_m} = \{z(\mathbf{y}_k \mid \hat{\mathbf{x}}_j)\}_{(\hat{\mathbf{x}}_j, \mathbf{y}_k) \in \hat{\mathcal{X}}_m \times \mathcal{Y}}$, have already been optimized. The goal of the interpolation function f_{int} is to assign perturbation probabilities to each non-anchor record $\mathbf{x}_a \in \mathcal{C}_m$ such that mDP is satisfied between $\mathbf{x}_a$ and any other record in the domain $\mathcal{X}$.

As defined by the Lipschitz bound (Eq. (3) in Definition 2), (ε, d_p)-mDP requires that the *log-probability* of outputs varies

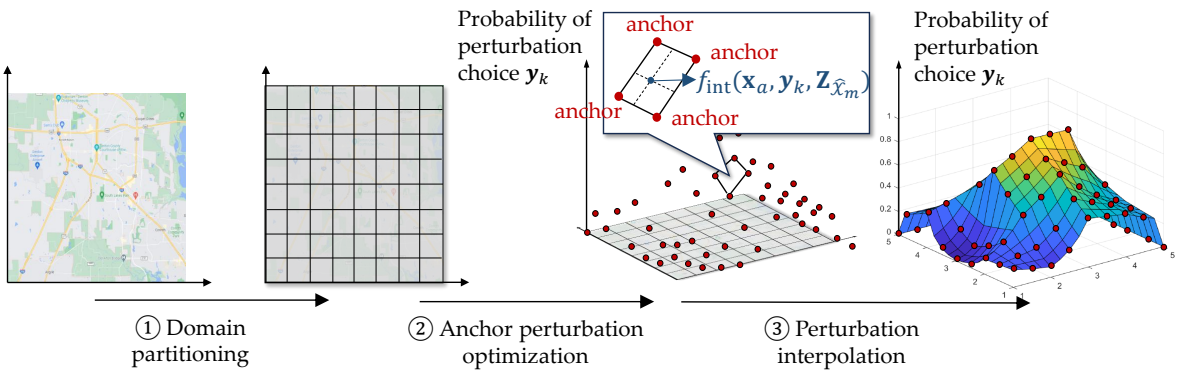


Figure 3: Framework of our method

at most linearly with the ℓ_p-distance between inputs. This motivates the use of *log-convex interpolation*, which linearly interpolates log-probabilities to ensure smooth transitions in the output distribution. As a natural first attempt, we define the log-probability at a non-anchor point $\mathbf{x}_a \in \mathcal{C}_m$ as a convex combination of those at the anchor points:

$$\ln z(\mathbf{y}_k \mid \mathbf{x}_a) = \sum_{\hat{\mathbf{x}} \in \hat{\mathcal{X}}_m} \lambda_{\hat{\mathbf{x}},\mathbf{x}_a} \ln z(\mathbf{y}_k \mid \hat{\mathbf{x}}), \tag{11}$$

where the convex coefficients $\lambda_{\hat{\mathbf{x}},\mathbf{x}_a} \geq 0$ sum to one and reflect the relative position of $\mathbf{x}_a$ within $\mathcal{C}_m$. Ideally, such interpolation would preserve mDP, i.e., if a reference point $\mathbf{x}_b$ satisfies mDP with respect to each anchor in $\hat{\mathcal{X}}_m$, then it should also satisfy mDP with respect to the interpolated point $\mathbf{x}_a$. Formally, we would desire:

$$\Big|\underbrace{\textstyle\sum_{\hat{\mathbf{x}} \in \hat{\mathcal{X}}_m} \lambda_{\hat{\mathbf{x}},\mathbf{x}_a} \ln z(\mathbf{y}_k \mid \hat{\mathbf{x}})}_{\text{Interpolated probability for } \mathbf{x}_a} - \ln z(\mathbf{y}_k \mid \mathbf{x}_b)\Big| \leq \varepsilon \cdot d_p(\mathbf{x}_a, \mathbf{x}_b). \tag{12}$$

However, this inequality does not generally hold in high dimensions due to the convexity of the ℓ_p-norm for $p \geq 1$. Specifically, the convex combination of anchor distances $\sum \lambda_{\hat{\mathbf{x}},\mathbf{x}_a} d_p(\hat{\mathbf{x}}, \mathbf{x}_b)$ may exceed $d_p(\mathbf{x}_a, \mathbf{x}_b)$, leading to interpolated log-probabilities that violate the Lipschitz bound required by mDP.

To overcome this issue, we factor the N-dimensional interpolation into a sequence of *one-dimensional, log-convex interpolations*, applied independently along each coordinate axis. In one dimension, the Lipschitz bound in Eq. (12) holds exactly (i.e., for $N = 1$), as formally established in **Propositions 1** and **2**. We then construct the N-dimensional mechanism by multiplicatively composing the per-axis interpolators in **Definition 5** and normalizing to obtain a valid joint perturbation distribution in **Definition 6**. The correctness of this composition is established by **Theorem 2** and **Proposition 3**, which respectively show that the resulting mechanism is (ε, d_p)-*Lipschitz continuous* and satisfies (ε, d_p)-*mDP* over the entire N-dimensional domain $\mathcal{X}$.

Next, we first introduce the one-dimensional interpolation primitive in **Section 4**, and then extend this construction to the multi-dimensional setting in **Section 5**, following the three-step procedure outlined in Fig. 3 (Steps ①–③).

4 One-Dimensional Interpolation and Privacy Composition

Definition 3 (One-Dimensional Log-Convex Interpolation). *Let $\mathbf{x}_i, \mathbf{x}_{i'} \in \mathcal{X}$ be two records that differ only in their ℓth coordinate, with $x_{i',\ell} = x_{i,\ell} + \Delta_\ell$ for some $\Delta_\ell > 0$. For any intermediate point $\mathbf{x}_a$ such that $x_{a,\ell} \in [x_{i,\ell}, x_{i',\ell}]$ and all other coordinates match $\mathbf{x}_i$ and $\mathbf{x}_{i'}$, define the convex coefficient $\lambda^{\ell}_{\mathbf{x}_i,\mathbf{x}_a} = \frac{x_{i',\ell} - x_{a,\ell}}{x_{i',\ell} - x_{i,\ell}}$. Then, the log-probability at $\mathbf{x}_a$ is given by the log-convex interpolation:*

$$\ln z(\mathbf{y}_k \mid \mathbf{x}_a) = \lambda^{\ell}_{\mathbf{x}_i,\mathbf{x}_a} \ln z(\mathbf{y}_k \mid \mathbf{x}_i) + (1 - \lambda^{\ell}_{\mathbf{x}_i,\mathbf{x}_a}) \ln z(\mathbf{y}_k \mid \mathbf{x}_{i'}), \tag{13}$$

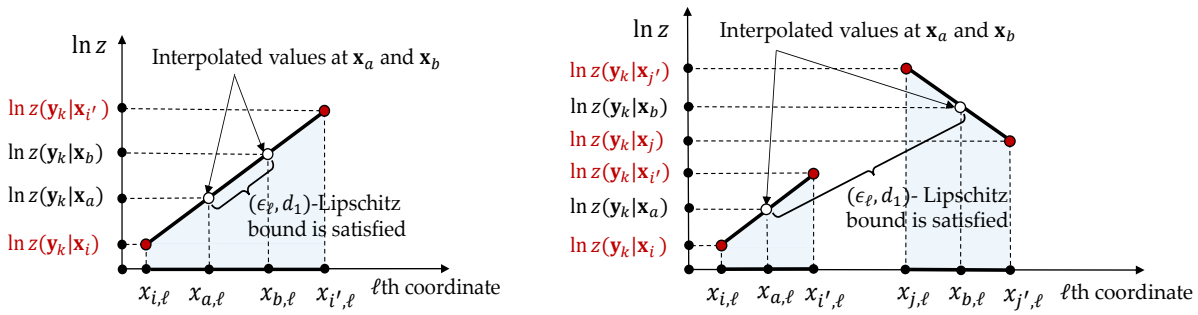
which is written as $z(\mathbf{y}_k \mid \mathbf{x}_a) \overset{\text{logcvx}}{\sim} (z(\mathbf{y}_k \mid \mathbf{x}_i), z(\mathbf{y}_k \mid \mathbf{x}_{i'}))$.

In **Propositions 1** and **2**, we prove that the interpolation mechanism preserves (ε_ℓ, d_1)-Lipschitz continuity; that is, it is (ε_ℓ, d_1)-Lipschitz within each one-dimensional interval and across adjacent intervals. The detailed proofs of Propositions 1 and 2 can be found in [30].

Proposition 1 (Intra-Interval Validity). *Let $\mathbf{x}_i$ and $\mathbf{x}_{i'}$ be two records that differ only in their ℓth coordinate, with $x_{i,\ell} < x_{i',\ell}$, and suppose their corresponding log-perturbation probabilities $\ln z(\mathbf{y}_k \mid \mathbf{x}_i)$ and $\ln z(\mathbf{y}_k \mid \mathbf{x}_{i'})$ satisfy the (ε, d_1)-Lipschitz bound. Then, for any $\mathbf{x}_a, \mathbf{x}_b \in \mathcal{X}$ such that $x_{a,\ell}, x_{b,\ell} \in [x_{i,\ell}, x_{i',\ell}]$ and all other coordinates are identical to those of $\mathbf{x}_i$, if the interpolated values $\hat{z}(\mathbf{y}_k \mid \mathbf{x}_a)$ and $\hat{z}(\mathbf{y}_k \mid \mathbf{x}_b)$ are calculated by*

$$\hat{z}(\mathbf{y}_k \mid \mathbf{x}_a) \overset{\text{logcvx}}{\sim} (z(\mathbf{y}_k \mid \mathbf{x}_i), z(\mathbf{y}_k \mid \mathbf{x}_{i'})), \tag{14}$$

$$\hat{z}(\mathbf{y}_k \mid \mathbf{x}_b) \overset{\text{logcvx}}{\sim} (z(\mathbf{y}_k \mid \mathbf{x}_i), z(\mathbf{y}_k \mid \mathbf{x}_{i'})), \tag{15}$$



(a) Intra-interval validity (b) Across-interval validity

Figure 4: Illustration of Proposition 1 and Proposition 2.

then $\ln \hat{z}(\mathbf{y}_k \mid \mathbf{x}_a)$ *and* $\ln \hat{z}(\mathbf{y}_k \mid \mathbf{x}_b)$ *also satisfy the* (ε, d_1)*-Lipschitz bound between* $\mathbf{x}_a$ *and* $\mathbf{x}_b$*. This property is illustrated in Fig. 4(a).*

Proposition 2 (Across-Interval Validity). *Let* $\mathbf{x}_i, \mathbf{x}_{i'}, \mathbf{x}_j, \mathbf{x}_{j'}$ *be four records that differ only in their* ℓ*th coordinate, with* $x_{i,\ell} < x_{i',\ell} \le x_{j,\ell} < x_{j',\ell}$*. Suppose that each pair of log-perturbation probabilities* $\ln z(\mathbf{y}_k \mid \mathbf{x}_i)$*,* $\ln z(\mathbf{y}_k \mid \mathbf{x}_{i'})$*,* $\ln z(\mathbf{y}_k \mid \mathbf{x}_j)$*, and* $\ln z(\mathbf{y}_k \mid \mathbf{x}_{j'})$ *satisfy* (ε, d_1)*-Lipschitz bound.*

Let $\mathbf{x}_a$ *and* $\mathbf{x}_b$ *be two additional records that differ from the above four points only in the* ℓ*th coordinate, with* $x_{a,\ell} \in [x_{i,\ell}, x_{i',\ell}]$ *and* $x_{b,\ell} \in [x_{j,\ell}, x_{j',\ell}]$*. If the corresponding interpolated values* $\hat{z}(\mathbf{y}_k \mid \mathbf{x}_a)$ *and* $\hat{z}(\mathbf{y}_k \mid \mathbf{x}_b)$ *are defined via log-convex interpolation as:*

$$\hat{z}(\mathbf{y}_k \mid \mathbf{x}_a) \overset{\text{logcvx}}{\sim} (z(\mathbf{y}_k \mid \mathbf{x}_i), z(\mathbf{y}_k \mid \mathbf{x}_{i'})) \tag{16}$$

$$\hat{z}(\mathbf{y}_k \mid \mathbf{x}_b) \overset{\text{logcvx}}{\sim} \left(z(\mathbf{y}_k \mid \mathbf{x}_j), z(\mathbf{y}_k \mid \mathbf{x}_{j'})\right), \tag{17}$$

then the pair $(\hat{z}(\mathbf{y}_k \mid \mathbf{x}_a), \hat{z}(\mathbf{y}_k \mid \mathbf{x}_b))$ *satisfies* (ε, d_1)*-Lipschitz bound between* $\mathbf{x}_a$ *and* $\mathbf{x}_b$*. This property is illustrated in Fig. 4(b).*

Propositions 1 and **2** establish that the one-dimensional log-convex interpolation is (ε_ℓ, d_1)-Lipschitz along coordinate ℓ. In multi-dimensional domains, however, inputs typically differ along multiple coordinates simultaneously. To extend our interpolation framework while preserving an overall (ε, d_p)-Lipschitz guarantee, we must coordinate the privacy leakage across dimensions via per-coordinate budgets $\{\varepsilon_\ell\}_{\ell=1}^N$. **Theorem 1** formalizes this by specifying how the $\{\varepsilon_\ell\}$ compose to yield a mechanism that satisfies the (ε, d_p)-Lipschitz condition on $\mathcal{X}$, thereby providing the theoretical foundation for our high-dimensional interpolation scheme.

Theorem 1 (Dimension-wise Composition for Lipschitz Bound Condition). *Let* $f : \mathcal{X} \to \mathbb{R}$ *be a mechanism that interpolates values in an* N*-dimensional space. Suppose that for each* $\ell \in \{1, \ldots, N\}$*,* f *satisfies* (ε_ℓ, d_1)*-Lipschitz bound when the input records differ only in the* ℓ*th coordinate. If the parameters* $\varepsilon_1, \ldots, \varepsilon_N$ *satisfy the following budget composition condition:*

$$\sum_{\ell=1}^{N} \varepsilon_\ell^{\frac{p}{p-1}} \le \varepsilon^{\frac{p}{p-1}}, \quad \text{for } p > 1, \tag{18}$$

$$\text{and} \qquad \max_\ell \varepsilon_\ell \le \varepsilon, \quad \text{for } p = 1, \tag{19}$$

then f *is* (ε, d_p)*-Lipschitz continuous.*

Proof Sketch. Fix $\mathbf{x}_a, \mathbf{x}_b \in \mathcal{X}$ and let $\Delta = \mathbf{x}_b - \mathbf{x}_a$, where $\Delta = [\Delta_1, \ldots, \Delta_N]$. Construct an axis-aligned path that updates one coordinate at a time: $\mathbf{x}^{(0)} = \mathbf{x}_a$ and $\mathbf{x}^{(\ell)} = \mathbf{x}^{(\ell-1)} + \Delta_\ell \mathbf{e}_\ell$, where $\mathbf{e}_\ell$ is the ℓth basis vector. Given that each one-dimensional step satisfies, for all $\mathbf{y}_k \subseteq \mathcal{Y}$,

$$\left| f(\mathbf{x}^{(\ell)}) - f(\mathbf{x}^{(\ell-1)}) \right| \le \varepsilon_\ell |\Delta_\ell|. \tag{20}$$

Summing these bounds along the path gives

$$|f(\mathbf{x}_b) - f(\mathbf{x}_a)| \le \textstyle\sum_{\ell=1}^N \varepsilon_\ell |\Delta_\ell|. \tag{21}$$

Let $p \in [1, \infty]$ and q be its dual exponent $(1/p + 1/q = 1)$. By Hölder's inequality, $\sum_{\ell=1}^N \varepsilon_\ell |\Delta_\ell| \le \|\varepsilon\|_q \|\Delta\|_p$. Setting $\varepsilon := \|\varepsilon\|_q$ and $d_p(\mathbf{x}_a, \mathbf{x}_b) := \|\mathbf{x}_b - \mathbf{x}_a\|_p$ yields $\sum_{\ell=1}^N \varepsilon_\ell |\Delta_\ell| \le \varepsilon d_p(\mathbf{x}_a, \mathbf{x}_b)$, and hence $|f(\mathbf{x}_b) - f(\mathbf{x}_a)| \le \varepsilon d_p(\mathbf{x}_a, \mathbf{x}_b)$, which is the desired (ε, d_p)-Lipschitz bound. A complete proof is provided in **Appendix A.1**. □

Remark. Theorem 1 ensures preservation of coordinate-wise Lipschitz bounds but does not guarantee that the interpolated values form a valid probability distribution (i.e., summing to one over $\mathcal{Y}$). In applications where $\mathcal{M}$ represents a data perturbation mechanism (where the normalization constraint is required), we therefore apply a normalization step after interpolation to restore validity. In our construction, normalization can increase the effective Lipschitz bound (and thus the mDP budget) by up to a factor of 2, which will be further discussed in **Section 5.2**.

In **Corollary 1**, we instantiate **Theorem 1** for a normalized data-perturbation mechanism $\mathcal{M} : \mathcal{X} \to \mathcal{Y}$.

Corollary 1 (**Dimension-wise Composition for mDP**). *Let $\mathcal{M}$ be a mechanism that perturbs data in an N-dimensional space. Suppose that for each $\ell \in \{1,\dots,N\}$, $\mathcal{M}$ satisfies (ε_ℓ, d_1)-mDP when the input records differ only in the ℓth coordinate. If the parameters $\varepsilon_1,\dots,\varepsilon_N$ satisfy the budget composition condition defined in Eqs. (18)(19), then $\mathcal{M}$ is (ε, d_p)-mDP.*

Discussion: Multi-attribute LDP as a special case of ℓ_p-norm mDP. Consider $\mathcal{X} \subseteq \mathcal{A}^N$ with Hamming distance $d_H(\mathbf{x}_a,\mathbf{x}_b) = \sum_{\ell=1}^N \mathbf{1}[x_{a,\ell} \neq x_{b,\ell}]$. A mechanism $\mathcal{M}$ satisfies *multi-attribute* ε-*LDP* [4] if, for all $\mathbf{x}_a,\mathbf{x}_b \in \mathcal{X}$ and $\mathbf{y} \in \mathcal{Y}$,

$$\Pr[\mathcal{M}(\mathbf{x}_a) = \mathbf{y}] \le e^{\varepsilon d_H(\mathbf{x}_a,\mathbf{x}_b)} \Pr[\mathcal{M}(\mathbf{x}_b) = \mathbf{y}], \tag{22}$$

which is exactly (ε, d_H)-mDP.

Moreover, multi-attribute LDP can be seen as a limiting case of ℓ_p-mDP when $p \to \infty$ on binary domains. Assume $\mathcal{X} \subseteq \{0,1\}^N$ (or more generally, that each coordinate difference is at most 1 after rescaling). Then

$$d_p(\mathbf{x}_a,\mathbf{x}_b) = \|\mathbf{x}_a - \mathbf{x}_b\|_\infty = \max_\ell |x_{a,\ell} - x_{b,\ell}| = \begin{cases} 1, & \mathbf{x}_a \neq \mathbf{x}_b, \\ 0, & \mathbf{x}_a = \mathbf{x}_b, \end{cases} \tag{23}$$

and per-coordinate distances reduce to

$$d_1(x_{a,\ell}, x_{b,\ell}) = |x_{a,\ell} - x_{b,\ell}| = \begin{cases} 1, & x_{a,\ell} \neq x_{b,\ell}, \\ 0, & x_{a,\ell} = x_{b,\ell}. \end{cases} \tag{24}$$

Under this metric, the pointwise mDP guarantee

$$\Pr[\mathcal{M}(\mathbf{x}_a) = \mathbf{y}] \le e^{\varepsilon d_p(\mathbf{x}_a,\mathbf{x}_b)} \Pr[\mathcal{M}(\mathbf{x}_b) = \mathbf{y}] \tag{25}$$

simplifies to the standard LDP bound

$$\Pr[\mathcal{M}(\mathbf{x}_a) = \mathbf{y}] \le e^{\varepsilon} \Pr[\mathcal{M}(\mathbf{x}_b) = \mathbf{y}] \tag{26}$$

whenever $\mathbf{x}_a \neq \mathbf{x}_b$ (and equals 1 when $\mathbf{x}_a = \mathbf{x}_b$).

Finally, as $p \to \infty$, the exponent $p/(p-1)$ in the budget composition rule in Eqs. (18)(19) tends to 1, and hence

$$\sum_{\ell=1}^N \varepsilon_\ell^{\frac{p}{p-1}} \le \varepsilon^{\frac{p}{p-1}} \quad \Longrightarrow \quad \sum_{\ell=1}^N \varepsilon_\ell \le \varepsilon, \tag{27}$$

which is exactly the familiar sequential composition condition. Thus, ℓ_p-norm mDP recovers multi-attribute LDP in the $p \to \infty$ limit on binary domains (equivalently, under the Hamming metric).

5 Multi-Dimensional Interpolation

Having established one-dimensional interpolation and the corresponding privacy composition property in Section 4, we now turn to extending our framework to multi-dimensional domains. Directly applying log-convex interpolation in higher dimensions can violate (ε, d_p)-mDP due to the geometric properties of ℓ_p-norms. Based on **Theorem 1**, we adopt a coordinate-wise approach: we interpolate along each dimension and then combine the results using a carefully designed composition rule to ensure global privacy guarantees. For clarity, we present the anchor perturbation optimization (Step ②) before introducing the interpolation function (Step ③), as the optimization depends on the specific interpolation structure.

5.1 Step ① - Domain Partitioning

According to **Theorem 1**, ℓ_p-norm mDP can be enforced by bounding privacy leakage separately along each coordinate. To support this dimension-wise composition, we partition the secret domain $\mathcal{X}$ into M non-overlapping N-orthotopes $\mathcal{C}_1,\dots,\mathcal{C}_M$, axis-aligned hyperrectangles that generalize rectangles to N dimensions. This coordinate-aligned structure ensures that neighboring anchors differ in only one dimension, enabling efficient log-convex interpolation while preserving (ε, d_p)-mDP.

More precisely, for each N-orthotope $\mathcal{C}_m$, we let $\hat{\mathbf{x}}_{i_m} = [\hat{x}_{i_m,1},\dots,\hat{x}_{i_m,N}]$ denote its base (minimum) corner, and let $\Delta = [\Delta_1,\dots,\Delta_N]$ represent the side lengths along each dimension. The full set of 2^N corner points of $\mathcal{C}_m$ is given by: $\hat{\mathcal{X}}_m = \left\{\hat{\mathbf{x}}_{i_m} + \gamma \odot \Delta \;\middle|\; \gamma \in \{0,1\}^N\right\}$, where γ is a binary indicator vector and $\odot$ denotes element-wise multiplication. Each corner point thus has coordinate $\hat{x}_{i_m,\ell} + \gamma_\ell \Delta_\ell$ along dimension ℓ. We refer to $\hat{\mathcal{X}}_m$ as the *anchor set* of $\mathcal{C}_m$, and define the full anchor set as $\hat{\mathcal{X}} = \bigcup_m \hat{\mathcal{X}}_m$, which serves as the support for interpolation and optimization.

Definition 4 (Axis-Aligned Anchor Neighbors). *Let $\hat{\mathbf{x}}_i, \hat{\mathbf{x}}_j \in \hat{\mathcal{X}}_m$ be two anchor points within the same N-orthotope $\mathcal{C}_m$. We say that $\hat{\mathbf{x}}_i$ and $\hat{\mathbf{x}}_j$ are ℓ-**axis neighbors** (or simply ℓ-**neighbors**) if they differ only along the ℓth coordinate; that is, $\hat{x}_{i,\ell} \neq \hat{x}_{j,\ell}$ and $\hat{x}_{i,\ell'} = \hat{x}_{j,\ell'}$ for all $\ell' \neq \ell$.*

5.2 Step ③ - Perturbation Interpolation

With the domain partitioned into N-orthotopes (Step ①), we interpolate perturbation distributions for non-anchor records using their corner anchors. For a record inside a cell, we first construct an *unnormalized* multi-dimensional log-convex interpolant f_{int} by applying log-convex interpolation *separately* along each coordinate and composing the results multiplicatively (**Definition 5**). Because these interpolated values might not sum to one over $\mathcal{Y}$, we then obtain a valid probability distribution by *normalizing* them, yielding the *normalized* multi-dimensional log-convex interpolation $\overline{f}_{\text{int}}$ (**Definition 6**). The *Lipschitz continuity* of the interpolated (log-)values is established by **Theorem 2**, and the (ε, d_p)-mDP guarantee for the normalized mechanism follows from **Proposition 3**.

Definition 5 (**Unnormalized Multi-Dimensional Log-Convex Interpolation** f_{int}). *Let $\hat{\mathcal{X}}_m$ denote the set of 2^N anchor*

points at the corners of an N-dimensional orthotope $\mathcal{C}_m$, and let $\mathbf{Z}_{\hat{\mathcal{X}}_m}$ represent their corresponding perturbation distributions. For any point $\mathbf{x}_a \in \mathcal{C}_m$ and output $\mathbf{y}_k \in \mathcal{Y}$, the interpolated value is defined as:

$$\ln f_{\text{int}}(\mathbf{x}_a, \mathbf{y}_k, \mathbf{Z}_{\hat{\mathcal{X}}_m}) = \sum_{\gamma \in \{0,1\}^N} w(\gamma) \ln z\big(\mathbf{y}_k \mid \hat{\mathbf{x}}_{i_m} + \gamma \odot \Delta\big), \quad (28)$$

or equivalently:

$$f_{\text{int}}(\mathbf{x}_a, \mathbf{y}_k, \mathbf{Z}_{\hat{\mathcal{X}}_m}) = \prod_{\gamma \in \{0,1\}^N} z\big(\mathbf{y}_k \mid \hat{\mathbf{x}}_{i_m} + \gamma \odot \Delta\big)^{w(\gamma)}. \quad (29)$$

Here, the weight function $w(\gamma)$, defined as

$$w(\gamma) = \prod_{\ell=1}^{N} \left[(1-\gamma_\ell) \lambda^{\ell}_{\hat{\mathbf{x}}_{i_m}, \mathbf{x}_a} + \gamma_\ell \left(1 - \lambda^{\ell}_{\hat{\mathbf{x}}_{i_m}, \mathbf{x}_a}\right) \right], \quad (30)$$

represents how each anchor point's distribution contributes to the interpolated distribution at the non-anchor point $\mathbf{x}_a$, with the weights reflecting the relative position of the point within its cell.

Theorem 2 (Correctness of Log-Convex Interpolation f_{int})**.** *Given that (ε_ℓ, d_1)-Lipschitz bound holds between each pair of ℓ-neighbors in $\hat{\mathcal{X}}$ and $\{\varepsilon_\ell\}_{\ell=1}^N$ satisfy the privacy budget composition condition formalized in Eq. (18)(19), the use of the interpolation function f_{int} (defined by Eq. (28)) guarantees that any two interpolated values within the entire secret data domain $\mathcal{X}$ satisfy (ε, d_p)-Lipschitz bound.*

Proof Sketch. We begin by proving that if two records $\mathbf{x}_a, \mathbf{x}_b \in \mathcal{X}$ differ only in a single coordinate ℓ, then their interpolated perturbation probabilities under f_{int} satisfy (ε_ℓ, d_1)-mDP along that dimension. Then by applying the dimension-wise composition theorem (**Theorem 1**), we establish that the composed mechanism satisfies (ε, d_p)-mDP globally. The full proof is provided in **Appendix A.2**. □

While the unnormalized log-convex interpolant preserves coordinate-wise Lipschitz continuity, its outputs do not necessarily lie on the probability simplex. To obtain a valid perturbation mechanism, we normalize these values (**Definition 6**), which restores a proper probability distribution at the cost of increasing the effective Lipschitz bound by at most a factor of 2 (**Proposition 3**).

Definition 6 (**Normalized Multi-Dimensional Log-Convex Interpolation** $\overline{f}_{\text{int}}$)**.** *Given a point $\mathbf{x}_a$ within cell $\mathcal{C}_m$ and a perturbation candidate $\mathbf{y}_k \in \mathcal{Y}$, the normalized interpolated probability is defined as:*

$$\overline{f}_{\text{int}}(\mathbf{x}_a, \mathbf{y}_k, \mathbf{Z}_{\hat{\mathcal{X}}_m}) = \frac{f_{\text{int}}(\mathbf{x}_a, \mathbf{y}_k, \mathbf{Z}_{\hat{\mathcal{X}}_m})}{\sum_{\mathbf{y}_j \in \mathcal{Y}} f_{\text{int}}(\mathbf{x}_a, \mathbf{y}_j, \mathbf{Z}_{\hat{\mathcal{X}}_m})}, \quad (31)$$

where the denominator normalizes the interpolated scores over all possible outputs, ensuring that the resulting distribution is valid: $\sum_{\mathbf{y}_k \in \mathcal{Y}} \overline{f}_{\text{int}}(\mathbf{x}_a, \mathbf{y}_k, \mathbf{Z}_{\hat{\mathcal{X}}_m}) = 1$.

Proposition 3. *Given that any pair of real records $\mathbf{x}_a \in \mathcal{X}_m$ and $\mathbf{x}_b \in \mathcal{X}_{m'}$ with their perturbation probabilities interpolated by $z(\mathbf{y}_k \mid \mathbf{x}_a) = \overline{f}_{\text{int}}(\mathbf{x}_a, \mathbf{y}_k, \mathbf{Z}_{\hat{\mathcal{X}}_m})$ and $z(\mathbf{y}_k \mid \mathbf{x}_b) = \overline{f}_{\text{int}}(\mathbf{x}_b, \mathbf{y}_k, \mathbf{Z}_{\hat{\mathcal{X}}_{m'}})$, then their perturbation probabilities satisfy $(2\varepsilon, d_p)$-mDP.*

Proof Sketch. The unnormalized interpolant f_{int} already satisfies pointwise (ε, d_p)-mDP, i.e., $\frac{f_{\text{int}}(\mathbf{x}_a, \mathbf{y}_k, \mathbf{Z}_{\hat{\mathcal{X}}_m})}{f_{\text{int}}(\mathbf{x}_b, \mathbf{y}_k, \mathbf{Z}_{\hat{\mathcal{X}}_m})} \le e^{\varepsilon d_p(\mathbf{x}_a, \mathbf{x}_b)}$. Summing over $\mathbf{y}_j$ shows that the partition function $Z(\mathbf{x}) = \sum_{\mathbf{y}_j \in \mathcal{Y}} f_{\text{int}}(\mathbf{x}_a, \mathbf{y}_j, \mathbf{Z}_{\hat{\mathcal{X}}_m})$ obeys the same multiplicative bounds. The normalized mechanism is $z(\mathbf{y}_k|\mathbf{x}_a) = \frac{f_{\text{int}}(\mathbf{x}_a, \mathbf{y}_k)}{Z(\mathbf{x}_a)}$. Thus, the ratio of normalized probabilities is the product of the per-class ratio and the inverse normalizer ratio, giving $\frac{z(\mathbf{y}_k|\mathbf{x}_a)}{z(\mathbf{y}_k|\mathbf{x}_b)} \le e^{2\varepsilon d_p(\mathbf{x}_a, \mathbf{x}_b)}$. Hence, the normalized mechanism is $(2\varepsilon, d_p)$-mDP. The detailed proof can be found in [30]. □

5.3 Step ② - Anchor Perturbation Optimization (APO)

The goal of APO is to jointly optimize (1) the per-dimension privacy budgets $\{\varepsilon_\ell\}_{\ell=1}^N$ under a global privacy budget composition constraint (formalized in Eq. (18)(19) in **Theorem 1**), and (2) the perturbation distributions of each pair of ℓ-neighbor in the anchor set $\hat{\mathcal{X}}$ satisfy (ε_ℓ, d_1)-mDP constraints along each dimension ℓ, so as to minimize the expected utility loss over the secret domain. Here, we define the expected utility loss of the secret data within each N-orthotope $\mathcal{C}_m$ by

$$\mathcal{L}(\mathbf{Z}_{\hat{\mathcal{X}}_m}) = \sum_{\mathbf{y}_k \in \mathcal{Y}} \int_{\mathcal{C}_m} \overline{f}_{\text{int}}(\mathbf{x}, \mathbf{y}_k, \mathbf{Z}_{\hat{\mathcal{X}}_m}) p(\mathbf{x}) \mathcal{L}(\mathbf{x}, \mathbf{y}_k) \mathrm{d}\mathbf{x}. \quad (32)$$

Definition 7 (**Anchor Perturbation Optimization (APO)**)**.** *The APO problem jointly optimizes the anchor perturbation distributions and the per-dimension privacy budgets to minimize the total expected utility loss $\sum_{m=1}^M \mathcal{L}(\mathbf{Z}_{\hat{\mathcal{X}}_m})$, subject to privacy and probability constraints. Formally:*

$$\min \quad \sum_{m=1}^{M} \mathcal{L}(\mathbf{Z}_{\hat{\mathcal{X}}_m}) \quad (33)$$

$$s.t. \quad \sum_{\ell=1}^{N} \varepsilon_\ell^{\frac{p}{p-1}} \le \left(\frac{\varepsilon}{2}\right)^{\frac{p}{p-1}}, \quad when\ p > 1, \quad (34)$$

$$\max_\ell \varepsilon_\ell \le \frac{\varepsilon}{2}, \quad when\ p = 1 \quad (35)$$

$$z(\mathbf{y}_k \mid \hat{\mathbf{x}}_i) - e^{\varepsilon_\ell \Delta_\ell} z(\mathbf{y}_k \mid \hat{\mathbf{x}}_j) \le 0,$$
$$for\ each\ pair\ \ell\text{-}neighbor\ \hat{\mathbf{x}}_i, \hat{\mathbf{x}}_j \in \hat{\mathcal{X}}, \forall \ell \quad (36)$$

$$\sum_{\mathbf{y}_k \in \mathcal{Y}} z(\mathbf{y}_k \mid \hat{\mathbf{x}}_i) = 1, \ \forall \hat{\mathbf{x}}_i \in \hat{\mathcal{X}}, \quad (37)$$

$$z(\mathbf{y}_k \mid \hat{\mathbf{x}}_i) \ge 0, \forall \hat{\mathbf{x}}_i \in \hat{\mathcal{X}}, \ \forall \mathbf{y}_k \in \mathcal{Y}. \quad (38)$$

Here, Eq. (34) enforces the privacy budget composition constraint*, Eq. (36) imposes (ε_ℓ, d_1)-mDP constraints between each pair of ℓ-neighbor anchors, and Eqs. (37)–(38) define the* normalization and non-negativity constraints*.*

The APO formulation introduces a nontrivial coupling between two sets of decision variables: the per-dimension privacy budgets $\{\varepsilon_\ell\}_{\ell=1}^N$ and the perturbation matrix $\mathbf{Z}_{\hat{\mathcal{X}}} = \{z(\mathbf{y}_k \mid \hat{\mathbf{x}}_i)\}_{(\hat{\mathbf{x}}_i,\mathbf{y}_k)\in\hat{\mathcal{X}}\times\mathcal{Y}}$. Since the feasible space of $\mathbf{Z}_{\hat{\mathcal{X}}}$ is constrained by the choice of $\{\varepsilon_\ell\}$, this coupling complicates joint optimization. In our experiments (**Section 6**), we focus on a 2D ℓ_2-norm setting, where the set of feasible privacy budget allocations $\{(\varepsilon_1,\varepsilon_2)\}$ must satisfy the constraint $\varepsilon_1^2+\varepsilon_2^2 \le \varepsilon^2/4$. This defines a quarter-circle region in the first quadrant. To search for the optimal allocation, we discretize this curve by sampling values of ε_1, compute the corresponding $\varepsilon_2 = \sqrt{\varepsilon^2/4-\varepsilon_1^2}$, and evaluate the resulting utility loss. This grid search remains tractable in low dimensions.

Another practical obstacle in solving the APO problem is the form of the expected-loss objective $\mathcal{L}(\mathbf{Z}_{\mathcal{X}}) = \sum_m \mathcal{L}(\mathbf{Z}_{\hat{\mathcal{X}}_m})$, where for any fixed cell $\mathcal{C}_m$ the integrand $\overline{f}_{\text{int}}(\mathbf{x}_a,\mathbf{y}_k,\mathbf{Z}_{\hat{\mathcal{X}}_m})$ is a *ratio of products of exponentials*; hence the resulting integral in Eq. (32) is analytically intractable and costly to evaluate numerically. Therefore, we replace this exact probability by a weighted geometric mean of the anchor probabilities and thereby convert the cell-level loss into a linear form.

Proposition 4 (Linear Surrogate for Utility Loss). *Let $\mathbf{x} = \hat{\mathbf{x}}_{i_m} + \lambda \odot \Delta \in \mathcal{C}_m$ be a non-anchor point, where $\lambda = [\lambda^1_{\hat{\mathbf{x}}_{i_m},\mathbf{x}},\ldots,\lambda^N_{\hat{\mathbf{x}}_{i_m},\mathbf{x}}] \in [0,1]^N$ are the convex interpolation weights. Approximating the perturbation probability in the objective function via weighted geometric interpolation yields:*

$$\Pr\left[\mathcal{M}(\mathbf{x};\mathbf{Z}_{\hat{\mathcal{X}}_m}) = \mathbf{y}_k\right] \tag{39}$$

$$\approx \sum_{\gamma\in\{0,1\}^N}\left(\prod_{\ell=1}^N\left((1-\gamma_\ell)\lambda^\ell_{\hat{\mathbf{x}}_{i_m},\mathbf{x}} + \gamma_\ell(1-\lambda^\ell_{\hat{\mathbf{x}}_{i_m},\mathbf{x}})\right)\right) \times z\left(\mathbf{y}_k \mid \hat{\mathbf{x}}_{i_m} + \gamma\odot\Delta\right). \tag{40}$$

*Under this approximation, the expected utility loss over cell $\mathcal{C}_m$ admits a linear surrogate: $\tilde{\mathcal{L}}(\mathbf{Z}_{\hat{\mathcal{X}}_m}) = \left\langle \tilde{\mathbf{C}}_{\hat{\mathcal{X}}_m}, \mathbf{Z}_{\hat{\mathcal{X}}_m}\right\rangle$, where $\tilde{\mathbf{C}}_{\hat{\mathcal{X}}_m} = \{\tilde{c}(\hat{\mathbf{x}}_i,\mathbf{y}_k)\}_{(\hat{\mathbf{x}}_i,\mathbf{y}_k)\in\hat{\mathcal{X}}_m\times\mathcal{Y}}$ is a constant coefficient matrix that depends only on the prior distribution $p(\mathbf{x})$ and the pointwise utility loss $\mathcal{L}(\mathbf{x},\mathbf{y}_k)$. Here, $\langle A,B\rangle$ denotes the standard Frobenius inner product between two matrices of the same shape: $\langle A,B\rangle = \sum_{i,k} A_{i,k}\cdot B_{i,k}$. A closed-form derivation of $\tilde{\mathbf{C}}_{\hat{\mathcal{X}}_m}$ is provided in **Appendix A.3**.*

An alternative approximation approach derives a per-cell upper/lower bound on the expected utility loss by introducing auxiliary decision variables that capture the minimum and maximum anchor values within each cell. When the secret domain $\mathcal{X}$ is partitioned into sufficiently small cells, the in-cell expectation can be tightly upper- and lower-bounded by the extreme anchor values. Optimizing this bound yields a linear–fractional objective, which can be efficiently converted into a linear program via the Charnes-Cooper transformation [8]. A detailed discussion is provided in [30].

Definition 8 (Approximate Anchor Perturbation Optimization (Approx-APO)). *Approx-APO retains all linear (ε_ℓ,d_1)-mDP and probability constraints from the original APO formulation, but replaces the expected utility loss objective $\sum_{m=1}^M \mathcal{L}(\mathbf{Z}_{\hat{\mathcal{X}}_m})$ with its linear surrogate:*

$$\sum_{m=1}^M \tilde{\mathcal{L}}(\mathbf{Z}_{\hat{\mathcal{X}}_m}) = \sum_{m=1}^M \left\langle \tilde{\mathbf{C}}_{\hat{\mathcal{X}}_m}, \mathbf{Z}_{\hat{\mathcal{X}}_m}\right\rangle, \tag{41}$$

*as defined in **Proposition 4**.*

Approx-APO can be solved efficiently using standard LP solvers [2], making it scalable to larger domains. However, since the surrogate objective is an approximation of the original expected utility loss, the resulting solution may not be optimal for the original APO. To assess its quality, we derive a universal lower bound on the optimal utility loss of the full APO formulation (see **Appendix B**). This bound serves as a benchmark for evaluating how closely the Approx-APO solution approaches the true optimum.

6 Case Study: Location Privacy in Navigation and Spatial Crowdsourcing

This section presents a case study of mDP in the context of location privacy protection. Such problems commonly arise when users must approach a target location to obtain a service or complete a task. Representative examples include (i) navigation services [32], where users query for routes while concealing their true location, and (ii) spatial crowdsourcing [34], where participants contribute geo-tagged data under privacy constraints.

To investigate this setting, we evaluate the proposed *Anchor-Interpolated Privacy Optimization (AIPO)* algorithm on real-world road-network datasets and compare it against representative baselines. The evaluation covers three dimensions: (i) *privacy* (**Section 6.2**), measured by violations of the (ε,d_p)-mDP constraints; (ii) *utility loss* (**Section 6.3**), quantified by expected service loss under different distance metrics; and (iii) *computational efficiency* (**Section 6.4**), assessed via runtime performance. The results show that AIPO enforces strict privacy with zero observed mDP violations, consistently reduces utility loss relative to existing methods, and achieves lower runtime compared to other optimization-based approaches. We begin by describing the experimental setup in **Section 6.1**, including details on the *datasets*, *computational resources*, and *baseline methods*.

6.1 Experiment Settings

Datasets. We conduct experiments on road network datasets from three major cities: *Rome, Italy*, *New York City (NYC), USA*, and *London, UK*. Each dataset consists of nodes representing intersections, junctions, and other key points in the

Rome road map									
Method		$\varepsilon = 0.2$	$\varepsilon = 0.4$	$\varepsilon = 0.6$	$\varepsilon = 0.8$	$\varepsilon = 1.0$	$\varepsilon = 1.2$	$\varepsilon = 1.4$	$\varepsilon = 1.6$
Pre-defined Noise Distribution	EM	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00
	Laplace	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00
	TEM	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00
Hybrid Method	RMP	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00
	COPT	1.17±0.72	3.14±2.04	1.12±0.54	0.99±0.39	0.90±0.32	0.82±0.27	0.76±0.23	0.70±0.20
LP		1.76±0.14	1.55±0.13	1.32±0.11	1.06±0.10	1.21±0.05	1.11±0.04	0.95±0.03	0.88±0.03
AIPO-R		7.24±0.51	6.91±0.38	4.80±0.30	3.84±0.17	2.70±0.04	0.00±0.00	0.00±0.00	0.00±0.00
AIPO†		0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00
London road map									
Method		$\varepsilon = 0.2$	$\varepsilon = 0.4$	$\varepsilon = 0.6$	$\varepsilon = 0.8$	$\varepsilon = 1.0$	$\varepsilon = 1.2$	$\varepsilon = 1.4$	$\varepsilon = 1.6$
Pre-defined Noise Distribution	EM	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00
	Laplace	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00
	TEM	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00
Hybrid Method	RMP	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00
	COPT	0.72±0.69	2.43±1.10	0.64±0.29	0.63±0.26	0.59±0.24	0.55±0.21	0.51±0.19	0.48±0.17
LP		1.35±0.10	1.35±0.09	1.10±0.06	0.82±0.05	0.72±0.04	0.58±0.03	0.65±0.04	1.86±0.17
AIPO-R		7.82±0.25	9.26±0.32	5.77±0.30	4.21±0.14	1.75±0.02	1.40±0.05	1.08±0.06	0.89±0.08
AIPO†		0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00
New York City road map									
Method		$\varepsilon = 0.2$	$\varepsilon = 0.4$	$\varepsilon = 0.6$	$\varepsilon = 0.8$	$\varepsilon = 1.0$	$\varepsilon = 1.2$	$\varepsilon = 1.4$	$\varepsilon = 1.6$
Pre-defined Noise Distribution	EM	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00
	Laplace	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00
	TEM	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00
Hybrid Method	RMP	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00
	COPT	0.97±0.48	4.10±0.84	0.74±0.39	0.72±0.41	0.68±0.41	0.65±0.40	0.61±0.38	0.57±0.37
LP		1.28±0.08	1.27±0.07	1.00±0.05	1.12±0.08	0.88±0.05	0.77±0.06	0.64±0.05	1.76±0.12
AIPO-R		10.99±0.52	6.84±0.36	6.01±0.37	4.22±0.29	2.07±0.11	1.35±0.06	1.06±0.07	0.90±0.08
AIPO†		0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00	0.00±0.00

Table 1: mDP violation ratio (Mean±1.96×standard deviation).

urban road network, with edges corresponding to road segments. The data are obtained from OpenStreetMap [1]. To support our interpolation-based method, we discretize each city's geographic area into a uniform grid of size 15×20, 10×20, and 10×20 for Rome, NYC, and London, respectively. Unless otherwise stated, all main results are reported under the Euclidean distance d_2 (i.e., the ℓ_2 norm) when evaluating (ε, d_p)-mDP and utility.

Experiments Compute Resources. Our experiments were conducted on a workstation equipped with an Intel Core i9-13900F processor (24 cores, 2.00–5.60 GHz), 32 GB of DDR5 memory (4800 MHz), and an NVIDIA GeForce RTX 4090 GPU with 24 GB of GDDR6X VRAM. Linear programs were solved using the MATLAB Optimization Toolbox function `linearprog` [2].

Representative baselines. We list several representative baselines that achieve ε-mDP:

(1) *Pre-defined Noise Distribution Methods*, including *Exponential Mechanism (EM)* [11], *Planar Laplace Mechanism (Laplace)* [3], and *Truncated Exponential Mechanism (TEM)* [7].

(2) *Linear programming (LP) [6]*, which minimizes expected utility under mDP constraints via a LP on a *discretized* domain; this approximation can misestimate pairwise distances and weaken mDP enforcement in fine-grained settings.

(3) *Hybrid Methods*, including *ConstOPTMech (COPT) [23]*, which combines LP with EM to balance utility and scalability under mDP, and *Bayesian Remapping (RMP) [10]*, which is a post-processing technique that enhances utility without compromising mDP.

(4) *AIPO-Relaxed (AIPO-R)*. AIPO-R is a relaxed variant of the proposed method that directly enforces the (ε, d_p)-mDP constraint using pairwise distances between anchors, without decomposing the privacy budget across dimensions.

Rome road map									
Method		$\varepsilon = 0.2$	$\varepsilon = 0.4$	$\varepsilon = 0.6$	$\varepsilon = 0.8$	$\varepsilon = 1.0$	$\varepsilon = 1.2$	$\varepsilon = 1.4$	$\varepsilon = 1.6$
Pre-defined	EM	8.71±0.78	8.70±1.13	8.65±1.28	8.62±1.38	8.58±1.45	8.56±1.51	8.54±1.55	8.52±1.58
Noise	Laplace	8.71±0.71	8.48±1.00	8.46±1.40	8.45±1.69	8.44±1.90	8.43±2.05	8.42±2.16	8.40±2.24
Distribution	TEM	8.85±2.71	8.95±3.19	8.66±2.44	8.64±1.83	8.66±1.11	8.66±0.69	8.66±0.27	8.62±0.22
Hybrid	RMP	5.94±0.25	4.96±0.45	4.28±0.36	3.85±0.26	3.58±0.21	3.40±0.19	3.28±0.18	3.19±0.18
Method	COPT	7.99±1.53	7.95±1.04	8.33±1.50	8.29±1.57	8.27±1.59	8.25±1.61	8.24±1.60	8.24±1.60
LP		4.25±0.41	2.97±0.11	2.56±0.03	2.45±0.07	2.43±0.03	2.42±0.02	2.42±0.01	2.42±0.01
AIPO-R		5.19±0.23	3.97±0.21	3.34±0.17	3.01±0.12	2.81±0.07	2.66±0.03	2.56±0.05	2.50±0.01
LB		1.82±0.01	1.73±0.01	1.73±0.00	1.73±0.01	1.73±0.00	1.73±0.01	1.73±0.00	1.73±0.00
AIPO†		**5.68±0.34**	**4.65±0.45**	**4.02±0.22**	**3.63±0.08**	**3.38±0.37**	**3.14±0.25**	**2.99±0.11**	**2.88±0.19**
London road map									
Method		$\varepsilon = 0.2$	$\varepsilon = 0.4$	$\varepsilon = 0.6$	$\varepsilon = 0.8$	$\varepsilon = 1.0$	$\varepsilon = 1.2$	$\varepsilon = 1.4$	$\varepsilon = 1.6$
Pre-defined	EM	7.69±0.74	7.44±1.30	7.30±1.58	7.20±1.73	7.14±1.82	7.09±1.88	7.05±1.93	7.02±1.97
Noise	Laplace	8.65±0.93	8.63±0.92	8.60±0.91	8.56±0.90	8.51±0.88	8.42±0.83	8.29±0.71	8.11±0.43
Distribution	TEM	8.01±2.22	7.72±2.27	7.87±1.94	7.96±1.11	7.98±0.62	7.99±0.35	7.99±0.12	7.98±0.13
Hybrid	RMP	5.86±0.21	5.07±0.39	4.49±0.41	4.09±0.37	3.83±0.32	3.65±0.29	3.53±0.26	3.44±0.24
Method	COPT	8.06±2.24	8.06±2.22	8.06±2.18	8.07±2.15	8.07±2.10	8.07±2.04	8.11±1.19	7.35±0.93
LP		4.19±0.24	2.92±0.13	2.56±0.14	2.47±0.11	2.45±0.07	2.44±0.09	2.44±0.03	2.44±0.05
AIPO-R		4.97±0.21	3.94±0.13	3.22±0.08	2.83±0.03	2.59±0.01	2.43±0.04	2.31±0.03	2.24±0.05
LB		1.51±0.05	1.44±0.02	1.44±0.03	1.44±0.01	1.44±0.00	1.44±0.01	1.44±0.00	1.44±0.00
AIPO†		**5.42±0.76**	**4.50±0.26**	**3.90±0.17**	**3.43±0.17**	**3.11±0.11**	**2.88±0.09**	**2.71±0.21**	**2.58±0.13**
New York City road map									
Method		$\varepsilon = 0.2$	$\varepsilon = 0.4$	$\varepsilon = 0.6$	$\varepsilon = 0.8$	$\varepsilon = 1.0$	$\varepsilon = 1.2$	$\varepsilon = 1.4$	$\varepsilon = 1.6$
Pre-defined	EM	13.96±1.59	13.95±2.38	13.88±2.78	13.80±3.06	13.73±3.27	13.69±3.44	13.65±3.57	13.64±3.66
Noise	Laplace	13.75±1.95	13.62±2.52	13.48±2.58	13.41±2.63	13.37±2.72	13.36±2.82	13.35±2.92	13.35±3.00
Distribution	TEM	13.62±3.79	13.53±4.02	13.77±3.00	13.98±1.96	13.95±1.11	13.92±0.64	13.83±0.23	13.72±0.17
Hybrid	RMP	7.69±0.37	5.58±0.37	4.55±0.24	4.00±0.24	3.69±0.28	3.50±0.31	3.38±0.32	3.29±0.32
Method	COPT	8.19±1.63	13.39±2.91	13.72±3.30	13.64±3.55	13.62±3.64	13.61±3.72	13.63±3.75	13.64±3.77
LP		4.80±0.12	3.14±0.07	2.67±0.11	2.56±0.21	2.53±0.09	2.52±0.04	2.52±0.07	2.52±0.01
AIPO-R		6.10±0.21	4.24±0.05	3.42±0.08	2.97±0.12	2.73±0.03	2.59±0.01	2.49±0.01	2.37±0.09
LB		2.18±0.11	2.04±0.02	2.03±0.01	2.03±0.01	2.03±0.00	2.03±0.00	2.03±0.01	2.03±0.00
AIPO†		**7.14±0.32**	**5.26±0.21**	**4.32±0.11**	**3.73±0.09**	**3.36±0.12**	**3.10±0.17**	**2.91±0.36**	**2.77±0.05**

Table 2: Utility loss across different perturbation methods (Mean±1.96×standard deviation).

6.2 Privacy Evaluation

To evaluate the empirical compliance of each mechanism with (ε, d_p)-mDP, we introduce the *perturbation probability ratio (PPR)* as a diagnostic metric. We randomly sample 1,000 locations from the input domain, denoted by $\mathcal{S}$, and for each pair $\mathbf{x}, \mathbf{x}' \in \mathcal{S}$ and output $\mathbf{y}_k$, compute

$$\mathrm{PPR}(\mathbf{x}, \mathbf{x}', \mathbf{y}_k) = \frac{|\ln z(\mathbf{y}_k \mid \mathbf{x}) - \ln z(\mathbf{y}_k \mid \mathbf{x}')|}{d_p(\mathbf{x}, \mathbf{x}')}. \tag{42}$$

A violation of the (ε, d_p)-mDP constraint is recorded whenever $\mathrm{PPR}(\mathbf{x}, \mathbf{x}', \mathbf{y}_k) > \varepsilon$ for any output $\mathbf{y}_k$.

Table 1 reports empirical violation ratios under privacy budgets $\varepsilon \in \{0.2, 0.4, \ldots, 1.6\}\,\mathrm{km}^{-1}$. *AIPO* attains 0% violations across all datasets and budgets, corroborating the correctness of its dimension-wise composition and log-convex interpolation. In contrast, *LP* and *COPT* exhibit nonzero violation ratios because they enforce constraints over discretized representatives, thereby *approximating* pairwise distances; such approximations can overestimate true continuous distances and relax the effective mDP constraints, missing privacy leakage at finer granularity. Pre-defined Noise Distribution mechanisms (e.g., Laplace, EM, TEM) do not incur violations but achieve this via heavier randomization, as reflected in their utility (detailed in Section 6.3).

The relaxed variant, *AIPO-R*, exhibits higher violation ratios. Unlike AIPO, it enforces the (ε, d_p)-mDP constraint only between anchor points and provides no guarantee in interpolated regions; as a result, privacy violations persist between anchors, particularly when the anchoring is sparse.

Rome road map								
Method	$\varepsilon = 0.2$	$\varepsilon = 0.4$	$\varepsilon = 0.6$	$\varepsilon = 0.8$	$\varepsilon = 1.0$	$\varepsilon = 1.2$	$\varepsilon = 1.4$	$\varepsilon = 1.6$
COPT	147.4±18.2	138.7±10.3	140.4±9.2	135.8±6.2	137.2±6.6	136.7±8.2	136.2±4.2	137.6±5.0
LP	210.8±296.7	122.1±19.3	345.5±323.4	342.1±242.8	399.0±158.7	318.1±138.7	428.2±223.9	366.3±249.4
AIPO†	29.3±10.9	20.0±5.8	28.6±5.6	64.7±159.9	10.0±2.4	8.3±2.6	5.1±0.9	8.0±19.3
London road map								
Method	$\varepsilon = 0.2$	$\varepsilon = 0.4$	$\varepsilon = 0.6$	$\varepsilon = 0.8$	$\varepsilon = 1.0$	$\varepsilon = 1.2$	$\varepsilon = 1.4$	$\varepsilon = 1.6$
COPT	285.1±115.0	274.6±110.9	272.9±114.4	272.2±113.2	271.5±118.1	269.7±117.8	266.4±109.7	269.4±103.2
LP	159.3±48.7	106.9±32.2	125.7±100.3	99.9±41.3	111.8±57.6	104.7±51.3	101.5±34.3	180.7±208.6
AIPO†	78.2±13.9	77.0±22.4	78.5±24.2	77.4±13.8	80.7±16.1	75.0±26.4	62.3±8.7	63.9±10.9
New York City road map								
Method	$\varepsilon = 0.2$	$\varepsilon = 0.4$	$\varepsilon = 0.6$	$\varepsilon = 0.8$	$\varepsilon = 1.0$	$\varepsilon = 1.2$	$\varepsilon = 1.4$	$\varepsilon = 1.6$
COPT	157.3±12.3	159.4±16.2	154.6±14.5	153.1±15.6	155.2±5.1	151.3±11.9	152.5±12.6	153.5±9.6
LP	303.8±140.8	324.1±218.8	263.0±141.9	418.6±202.1	265.6±178.5	284.8±203.5	393.7±304.8	403.2±174.9
AIPO†	75.1±25.9	56.3±12.9	39.9±14.9	29.3±7.5	24.3±7.7	21.3±3.8	27.6±44.7	17.3±3.9

Table 3: Computation time of different perturbation methods (Mean±1.96×standard deviation).

6.3 Utility Loss Comparison

Given a true vehicle position $\mathbf{x}_i$ and a task destination $\mathbf{x}_{\text{task}}$, the loss incurred by releasing the perturbed location $\mathbf{y}_k$ is defined as the absolute difference between the corresponding shortest-path lengths, $\left|\text{path}(\mathbf{x}_i, \mathbf{x}_{\text{task}}) - \text{path}(\mathbf{y}_k, \mathbf{x}_{\text{task}})\right|$. Aggregating over the prior distribution $p(\mathbf{x}_{\text{task}})$ of task locations Q yields the expected utility loss:

$$\mathcal{L}(\mathbf{x}_i, \mathbf{y}_k) = \sum_{\mathbf{x}_{\text{task}} \in Q} p(\mathbf{x}_{\text{task}}) \left|\text{path}(\mathbf{x}_i, \mathbf{x}_{\text{task}}) - \text{path}(\mathbf{y}_k, \mathbf{x}_{\text{task}})\right|. \tag{43}$$

To compute $\text{path}(\cdot,\cdot)$, we apply Dijkstra's algorithm [5] to determine the shortest-path distance between the origin and destination nodes. Since the three roadmap datasets are discrete location sets, the true objective can be evaluated exactly (the integral in Eq. (32) reduces to a summation). This allows us to quantify the gap between the true objective $\mathcal{L}$ and the approximated objective $\tilde{\mathcal{L}}$ (by **Definition 8**), measured as $\frac{\mathcal{L}-\tilde{\mathcal{L}}}{\tilde{\mathcal{L}}}$. The average gap across three datasets is 0.1779.

Table 2 reports the utility loss of our interpolation mechanism AIPO versus baseline perturbation methods on three road-network datasets:

AIPO vs. pre-defined Noise Distribution Methods (EM, TEM, Laplace). Across all three datasets, *AIPO* consistently outperforms pre-defined noise distribution methods in terms of utility. *On average, AIPO reduces utility loss by 59.03%, 60.94%, and 60.65% compared to EM, Laplace, and TEM, respectively.* EM and TEM define perturbation probabilities using (truncated) exponential scoring functions, while Laplace adds noise sampled independently from a fixed distribution. These methods rely on global or isotropic perturbation rules that ignore the underlying geometry and the direction-dependent sensitivity of task-specific utility. Consequently, they tend to over-perturb in dense regions and under-protect in sparse ones. In contrast, AIPO optimizes perturbation distributions at a sparse set of anchor points and interpolates them across the domain via log-convex combinations, yielding smooth transitions aligned with the metric structure.

AIPO vs. Hybrid Methods (COPT and RMP). *Across the three datasets, AIPO achieves an average of 59.16% lower utility loss compared to COPT.* The limited scalability of COPT stems from its rigid LP formulation, which becomes impractical in high-resolution domains. In contrast, AIPO employs an anchor-based framework with log-convex interpolation, enabling flexible adaptation to fine-grained input spaces. By explicitly optimizing perturbation probabilities at a sparse set of anchor points, AIPO preserves the mDP guarantee while achieving improved utility.

AIPO also outperforms RMP, achieving 10.36% lower utility loss on average across the evaluated datasets. While RMP improves pre-defined noise mechanisms by reshaping posterior distributions to approximate mDP, its effectiveness is fundamentally limited by the quality of the initial noise distribution, which often falls short of global optimality. In contrast, AIPO formulates and solves an optimization problem directly under mDP constraints, without relying on any pre-defined perturbation mechanism, leading to a lower utility loss.

AIPO vs. LP. The LP method achieves lower utility loss compared to AIPO, since LP directly optimizes the perturbation distributions to minimize expected loss under distance-based constraints. However, this comes at the expense of privacy: the LP method only enforces the mDP constraints on a discrete grid of locations, without guaranteeing that the constraints hold for all possible input pairs in the continuous domain. As a result, while LP appears effective in terms of utility, it

may produce perturbation probabilities that violate the mDP guarantee in off-grid regions, which has been demonstrated in **Table 1**.

AIPO vs. Lower Bound (LB). For theoretical reference, we also compare *AIPO* with a universal lower bound (LB) from Proposition 5 (Appendix B), which lower-bounds the minimum utility loss attainable by *any* mechanism satisfying (ε, d_p)-mDP. Empirically, *AIPO* lies within 1.36×–3.13× of this bound across datasets and budgets, with larger gaps at tighter privacy (e.g., $\varepsilon = 0.4$) due to stricter constraints; the ratio narrows as ε increases.

AIPO vs. AIPO-R. Empirically, AIPO achieves higher utility loss compared to AIPO-R as AIPO-R relaxes the mDP constraint. In return, AIPO offers stronger theoretical guarantees, ensuring full compliance with (ε, d_p)-mDP across the continuous space.

6.4 Computation Efficiency

Table 3 compares runtimes for the optimization-based methods (AIPO, LP) and the hybrid method COPT, each leveraging LP at some stage to construct perturbation mechanisms. AIPO attains a favorable utility–efficiency trade-off: it solves LPs only at a sparse set of anchors, and then uses log-convex interpolation to cover the continuous domain without a dense, full-scale program. This anchor-based design reduces both the number of decision variables and the effective constraint set, yielding stable performance across cities. By contrast, the LP baseline incurs substantially higher cost in fine-grained settings because it solves a large program with mDP constraints over (nearly) all pairs of grid cells: the constraint count grows quadratically with grid size, and barrier/simplex iterations amplify this growth, making LP increasingly prohibitive as resolution increases.

7 Related Works

The earliest and most extensively studied applications of mDP focus on *location privacy over grid maps*. Foundational works such as Andrés et al. [3] and Bordenabe et al. [6] introduced geo-indistinguishability and linear programming (LP)-based mechanisms to enforce spatial privacy while preserving utility. These approaches rely on *pre-defined noise distributions* or *grid-based optimization* over discretized spatial domains. Subsequent works improve utility or scalability by incorporating personalized or group-based noise [25, 39], truncated noise distributions [36], and adaptations to federated or blockchain settings [21]. Hybrid mechanisms such as Bayesian remapping [10] and utility-aware post-processing [27] refine predefined perturbations to improve performance. Yu et al. [38] exemplify a hybrid design that combines mDP with an alternative privacy metric, "expected inference error", using context-aware noise adaptation.

A parallel line of research applies mDP to *high-dimensional embeddings* in domains such as natural language processing and multimedia. In these settings, input records reside in continuous semantic vector spaces. Fernandes et al. [17] and Feyisetan et al. [19] added Laplace noise directly to text embeddings or used nearest-neighbor remapping. Carvalho et al. [7] proposed a truncated exponential mechanism constrained to semantically similar candidates. For non-text data, Han et al. [22] and Chen et al. [12] adapted Laplace-based perturbations to voice and facial embeddings. These approaches often fall under *pre-defined* or *hybrid* categories, tailored for continuous or structured vector domains.

Recent work has shifted toward *optimization-based mDP mechanisms* designed for *fine-grained or continuous domains*, where more precise control over privacy-utility trade-offs is critical. Imola et al. [23] introduced ConstOPT, which combines LP with the exponential mechanism to improve scalability and utility. Qiu et al. [29, 31] proposed decomposition-based perturbation frameworks, including partitioning and Benders decomposition. Other studies explore reinforcement learning [26] and bilevel optimization [37] to enforce mDP under task-specific utility objectives such as mobility prediction. These results demonstrate the versatility of *customized optimization* strategies for enforcing mDP in high-resolution domains. Additionally, trajectory privacy methods such as sequential perturbation [35] showcase the broader applicability of mDP frameworks in complex spatiotemporal contexts.

8 Conclusion

We propose an interpolation-based framework for optimizing (ε, d_p)-mDP in continuous domains. By learning perturbation distributions at anchor points and applying dimension-wise log-convex interpolation, our method achieves both *strict privacy guarantees* (zero mDP violations) and *low utility loss*, supported by rigorous theoretical analysis and empirical evaluation on real-world datasets. This approach enables practical and theoretically grounded privacy mechanisms for fine-grained spatial and continuous data. While our method is highly effective in low-dimensional settings, its scalability is challenged in high-dimensional domains due to the exponential growth in anchor points, increased interpolation complexity, and fragmented privacy budget allocation. These limitations, common across mDP mechanisms under ℓ_p-norms, motivate future work on bilevel programming, optimization decomposition, or adaptive partitioning.

9 Acknowledgements

The authors would like to thank the anonymous reviewers and the shepherd for their valuable comments and suggestions. This research was partially supported by U.S. NSF grants CNS-2136948 and CNS-2313866.

Ethical Considerations

We reviewed the venue's ethics guidance, submission instructions, and ethics-document requirements. The study was conducted responsibly, and our planned next steps adhere to those principles.

Stakeholders. The primary stakeholders of this work include: (i) the research team and the broader community of researchers working on mDP; (ii) practitioners and platform operators who may deploy mDP mechanisms in navigation, mobility analytics, spatial crowdsourcing, or related services, and the end-users of such systems; (iii) data subjects whose locations or other sensitive attributes might be protected by such mechanisms in future deployments; (iv) data and infrastructure providers such as OpenStreetMap contributors; and (v) society at large, including companies and institutions that rely on privacy-preserving data analytics, as well as marginalized or highly surveilled groups who may face disproportionate harms if location privacy is weak. For the identified stakeholders, the research process has negligible direct impact beyond advancing mDP methodology, whereas publication and deployment primarily affect practitioners, data subjects, and society by shaping how privacy mechanisms are chosen and used in practice.

Impacts of the research process and publication. Regarding the *research process*, our evaluation relies solely on publicly available OpenStreetMap road-network graphs and synthetic perturbations. We do not collect, process, or attempt to infer personal or identifiable user-level information, and we do not interact with live systems. As a result, we believe that tangible harms (e.g., financial loss, exposure to disturbing content) and rights-based harms (e.g., violations of informed consent or expectations of privacy) to individuals during the research process are minimal.

Regarding the *publication and potential deployment* of our results, the intended impact is positive: our interpolation-based optimization framework aims to make ℓ_p-mDP mechanisms for continuous and fine-grained domains more scalable, analyzable, and usable, thereby strengthening protections for sensitive data. This can benefit researchers (clearer theory and reproducible baselines), practitioners (deployable and auditable mechanisms with guidance on composition and parameter selection), and data subjects (stronger protection for location and similar data in downstream applications). At the same time, there is a dual-use risk: our framework could be misused to justify overly weak privacy configurations, such as very large ε or poorly chosen metrics, that technically satisfy formal definitions while providing limited real-world protection, particularly for vulnerable populations. There is also a risk that deployers might over-interpret our results outside the assumptions we state (e.g., different threat models), leading to a mismatch between perceived and actual privacy.

Mitigations and residual risks. To mitigate these risks, we (i) clearly document the assumptions and limitations of our approach, including its sensitivity to ε, metric choice, and modeling assumptions; (ii) explicitly warn against using extreme parameter settings or metrics that do not reflect meaningful notions of distance or harm for the affected population; (iii) recommend that practitioners adopt conservative privacy budgets and carefully evaluate deployments in light of relevant ethical, legal, and fairness considerations, especially in high-risk or high-stakes domains; and (iv) restrict our artifacts to code and derived or synthetic data that support defensive, privacy-preserving mechanisms, without releasing additional sensitive raw data. Despite these steps, some residual risk remains that third parties may deploy mechanisms irresponsibly or ignore our guidance; this residual risk cannot be fully eliminated.

Decision to conduct and publish the research. In deciding to conduct this research, we judged, under the lens of Beneficence and Respect for Persons, that simulation-based analysis on public infrastructure-level data, with no new collection or linkage of individual records, posed low risk relative to the potential benefit of improving privacy protection tools. Justice and Respect for Law and Public Interest are reflected in our use of publicly available, properly licensed datasets and in our aim to strengthen protections for users, including those at higher risk of surveillance or discrimination. In deciding to publish, we considered whether withholding the methods would better protect stakeholders; however, we concluded that making the techniques public, together with explicit discussion of their assumptions, limitations, potential dual use, and recommended safeguards, provides a net ethical benefit. Publication enables community scrutiny, supports more robust and transparent privacy mechanisms, and offers practitioners concrete guidance on responsible deployment.

Open Science

We provide a complete MATLAB implementation of the proposed AIPO framework to enable full reproducibility of all experimental results reported in the paper. The artifact is publicly available at https://doi.org/10.5281/zenodo.17851733. It includes a README that documents the code structure, key components, and all relevant configuration parameters.

References

[1] 2020. openstreetmap. https://www.openstreetmap.org/. Accessed: 2020-04-07.

[2] 2024. linprog: Solve linear programming problems. https://www.mathworks.com/help/optim/ug/linprog.html. Accessed in January 2024.

[3] Miguel E. Andrés, Nicolás E. Bordenabe, Konstantinos Chatzikokolakis, and Catuscia Palamidessi. 2013. Geo-indistinguishability: differential privacy for location-based systems. In *Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security* (Berlin, Germany) *(CCS '13)*. Association for Computing Machinery, New York, NY, USA, 901–914. doi:10.1145/2508859.2516735

[4] Héber H. Arcolezi, Sébastien Gambs, Jean-François Couchot, and Catuscia Palamidessi. 2023. On the Risks of Collecting Multidimensional Data Under Local Differential Privacy. *Proc. VLDB Endow.* 16, 5 (Jan. 2023), 1126–1139. doi:10.14778/3579075.3579086

[5] Harsh Bhasin. 2015. *Algorithms: Design and Analysis*. Oxford Univ Press.

[6] N. E. Bordenabe, K. Chatzikokolakis, and C. Palamidessi. 2014. Optimal Geo-Indistinguishable Mechanisms for Location Privacy. In *Proc. of ACM CCS*. 251–262.

[7] Ricardo Silva Carvalho, Theodore Vasiloudis, and Oluwaseyi Feyisetan. 2021. TEM: High Utility Metric Differential Privacy on Text. *ArXiv* abs/2107.07928 (2021). https://api.semanticscholar.org/CorpusID:236034456

[8] A. Charnes and W. W. Cooper. 1962. Programming with linear fractional functionals. *Naval Research Logistics Quarterly* 9 (1962), 181–186. doi:10.1002/nav.3800090303

[9] Konstantinos Chatzikokolakis, Miguel E. Andrés, Nicolás Emilio Bordenabe, and Catuscia Palamidessi. 2013. Broadening the Scope of Differential Privacy Using Metrics. In *Proc. of Privacy Enhancing Technologies*, Emiliano De Cristofaro and Matthew Wright (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 82–102.

[10] Kostas Chatzikokolakis, Ehab Elsalamouny, and Catuscia Palamidessi. 2017. Efficient Utility Improvement for Location Privacy. *Proceedings on Privacy Enhancing Technologies* 2017 (10 2017). doi:10.1515/popets-2017-0051

[11] Konstantinos Chatzikokolakis, Catuscia Palamidessi, and Marco Stronati. 2015. Constructing elastic distinguishability metrics for location privacy. *Privacy Enhancing Technologies (PoPETs)* 2015 (2015), 156–170. http://www.degruyter.com/view/j/popets.2015.2015.issue-2/popets-2015-0023/popets-2015-0023.xml

[12] Jia-Wei Chen, Li-Ju Chen, Chia-Mu Yu, and Chun-Shien Lu. 2021. Perceptual Indistinguishability-Net (PI-Net): Facial Image Obfuscation with Manipulable Semantics . In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE Computer Society, Los Alamitos, CA, USA, 6474–6483. doi:10.1109/CVPR46437.2021.00641

[13] The-Hien Dang-Ha, Roland Olsson, and Hao Wang. 2017. Clustering Methods for Electricity Consumers: An Empirical Study in Hvaler-Norway. arXiv:1703.02502 [stat.AP] https://arxiv.org/abs/1703.02502

[14] Prathamesh Dharangutte, Jie Gao, Ruobin Gong, and Fang-Yi Yu. 2023. Integer subspace differential privacy. In *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence and Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence and Thirteenth Symposium on Educational Advances in Artificial Intelligence (AAAI'23/IAAI'23/EAAI'23)*. AAAI Press, Article 826, 9 pages. doi:10.1609/aaai.v37i6.25895

[15] John C. Duchi, Michael I. Jordan, and Martin J. Wainwright. 2013. Local Privacy and Statistical Minimax Rates. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*. 429–438. doi:10.1109/FOCS.2013.53

[16] Liyue Fan. 2019. Practical Image Obfuscation with Provable Privacy. In *2019 IEEE International Conference on Multimedia and Expo (ICME)*. 784–789. doi:10.1109/ICME.2019.00140

[17] Natasha Fernandes, Mark Dras, and Annabelle McIver. 2018. Author Obfuscation Using Generalised Differential Privacy. *ArXiv* abs/1805.08866 (2018). https://api.semanticscholar.org/CorpusID:43942677

[18] Oluwaseyi Feyisetan, Borja Balle, Thomas Drake, and Tom Diethe. 2020. Privacy- and Utility-Preserving Textual Analysis via Calibrated Multivariate Perturbations. In *Proceedings of the 13th International Conference on Web Search and Data Mining* (Houston, TX, USA) *(WSDM '20)*. Association for Computing Machinery, New York, NY, USA, 178–186. doi:10.1145/3336191.3371856

[19] O. Feyisetan, T. Diethe, and T. Drake. 2019. Leveraging Hierarchical Representations for Preserving Privacy and Utility in Text. In *2019 IEEE International Conference on Data Mining (ICDM)*. IEEE Computer Society, Los Alamitos, CA, USA, 210–219. doi:10.1109/ICDM.2019.00031

[20] Oluwaseyi Feyisetan and Shiva Kasiviswanathan. 2021. Private Release of Text Embedding Vectors. In *Proceedings of the First Workshop on Trustworthy Natural Language Processing*, Yada Pruksachatkun, Anil Ramakrishna, Kai-Wei Chang, Satyapriya Krishna, Jwala Dhamala, Tanaya Guha, and Xiang Ren (Eds.). Association for Computational Linguistics, Online, 15–27. doi:10.18653/v1/2021.trustnlp-1.3

[21] Filippo Galli, Sayan Biswas, Kangsoo Jung, Tommaso Cucinotta, and Catuscia Palamidessi. 2022. Group privacy for personalized federated learning. In *Workshop on Federated Learning: Recent Advances and New Challenges (in Conjunction with NeurIPS 2022)*. https://openreview.net/forum?id=R45g30SnwsR

[22] Yaowei Han, Sheng Li, Yang Cao, Qiang Ma, and Masatoshi Yoshikawa. 2020. Voice-Indistinguishability: Protecting Voiceprint In Privacy-Preserving Speech Data Release . In *2020 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE Computer Society, Los Alamitos, CA, USA, 1–6. doi:10.1109/ICME46284.2020.9102875

[23] Jacob Imola, Shiva Kasiviswanathan, Stephen White, Abhinav Aggarwal, and Nathanael Teissier. 2022. Balancing utility and scalability in metric differential privacy. In *Proc. of UAI 2022*.

[24] Fragkiskos Koufogiannis, Shuo Han, and George J. Pappas. 2015. Optimality of the Laplace Mechanism in Differential Privacy. arXiv:1504.00065 [cs.CR] https://arxiv.org/abs/1504.00065

[25] Baihe Ma, Xu Wang, Wei Ni, and Ren Ping Liu. 2022. Personalized Location Privacy With Road Network-Indistinguishability. *IEEE Transactions on Intelligent Transportation Systems* 23, 11 (2022), 20860–20872. doi:10.1109/TITS.2022.3179501

[26] Minghui Min, Haopeng Zhu, Jiahao Ding, Shiyin Li, Liang Xiao, Miao Pan, and Zhu Han. 2024. Personalized 3D Location Privacy Protection With Differential and Distortion Geo-Perturbation. *IEEE Trans. Dependable Secur. Comput.* 21, 4 (July - August 2024), 3629–3643. https://doi.org/10.1109/TDSC.2023.3335374

[27] Simon Oya, Carmela Troncoso, and Fernando Pérez-González. 2017. Is Geo-Indistinguishability What You Are Looking For?. In *Proc. of the 2017 on Workshop on Privacy in the Electronic Society* (Dallas, Texas, USA) *(WPES '17)*. Association for Computing Machinery, New York, NY, USA, 137–140. doi:10.1145/3139550.3139555

[28] P. Pappachan, C. Qiu, A. Squicciarini, and V. Manjunath. 2023. User Customizable and Robust Geo-Indistinguishability for Location Privacy. In *Proc. of International Conference on Extending Database Technology (EDBT)*.

[29] Chenxi Qiu. 2024. Enhancing Scalability of Metric Differential Privacy via Secret Dataset Partitioning and Benders Decomposition. In *Proc. of 33rd International Joint Conference on Artificial Intelligence (IJCAI)*.

[30] Chenxi Qiu. 2026. Interpolation-Based Optimization for Enforcing lp-Norm Metric Differential Privacy in Continuous and Fine-Grained Domains. arXiv:2601.09946 [cs.LG] https://arxiv.org/abs/2601.09946

[31] Chenxi Qiu, Ruiyao Liu, Primal Pappachan, Anna Squicciarini, and Xinpeng Xie. 2025. Time-Efficient Locally Relevant Geo-Location Privacy Protection. In *Proc. on Privacy Enhancing Technologies*.

[32] C. Qiu, A. C. Squicciarini, C. Pang, N. Wang, and B. Wu. 2022. Location Privacy Protection in Vehicle-Based Spatial Crowdsourcing via Geo-Indistinguishability. *IEEE Transactions on Mobile Computing* (2022), 1–1. doi:10.1109/TMC.2020.3037911

[33] Daniel W. Stroock. 2010. *Probability Theory: An Analytic View* (2nd ed.). Cambridge University Press.

[34] L. Wang, D. Yang, X. Han, T. Wang, D. Zhang, and X. Ma. 2017. Location Privacy-Preserving Task Allocation for Mobile Crowdsensing with Differential Geo-Obfuscation. In *Proc. of ACM WWW*. 627–636.

[35] Yonghui Xiao and Li Xiong. 2015. Protecting Locations with Differential Privacy under Temporal Correlations *(CCS '15)*. Association for Computing Machinery, New York, NY, USA, 1298–1309. doi:10.1145/2810103.2813640

[36] Changxin et al. Yang. 2021. Blockchain-based indoor location paging and answering service with truncated-geo-indistinguishability. *IET Blockchain* (2021).

[37] Dan Yu, Xiufang Shi, Li Chai, Wen-An Zhang, and Jiming Chen. 2023. Balancing Localization Accuracy and Location Privacy in Mobile Cooperative Localization. *IEEE Transactions on Signal Processing* PP (01 2023), 1–14. doi:10.1109/TSP.2023.3292505

[38] L. Yu, L. Liu, and C. Pu. 2017. Dynamic Differential Location Privacy with Personalized Error Bounds. In *Proc. of IEEE NDSS*.

[39] Pengfei Zhang, Xiang Cheng, Sen Su, and Ning Wang. 2023. Task Allocation Under Geo-Indistinguishability via Group-Based Noise Addition. *IEEE Transactions on Big Data* 9, 3 (2023), 860–877. doi:10.1109/TBDATA.2022.3215467

A Omitted Proofs

A.1 Proof of Theorem 1: Dimension-wise Composition for Lipschitz Bound Condition

Theorem 1 (Dimension-wise Composition for Lipschitz Bound Condition). *Let $f : \mathcal{X} \to \mathbb{R}$ be a mechanism that interpolates values in an N-dimensional space. Suppose that for each $\ell \in \{1,\dots,N\}$, f satisfies (ε_ℓ, d_1)-Lipschitz bound when the input records differ only in the ℓth coordinate. If the parameters $\varepsilon_1,\dots,\varepsilon_N$ satisfy the following budget composition condition:*

$$\sum_{\ell=1}^{N} \varepsilon_\ell^{\frac{p}{p-1}} \le \varepsilon^{\frac{p}{p-1}}, \quad \text{for } p > 1, \tag{44}$$

$$\text{and} \qquad \max_\ell \varepsilon_\ell \le \varepsilon, \quad \text{for } p = 1, \tag{45}$$

then f is (ε, d_p)-Lipschitz continuous.

Proof. For any pair $\mathbf{x}_a, \mathbf{x}_b \in \mathcal{X}$, we write the coordinate difference as $\Delta = \mathbf{x}' - \mathbf{x}$ with components $\Delta_\ell = x'_\ell - x_\ell$ for $\ell \in \{1,\dots,N\}$, and let $\mathbf{e}_\ell$ denote the ℓth standard basis vector. Consider the axis-aligned path that changes one coordinate at a time: $\mathbf{x}^{(0)} := \mathbf{x}$, $\mathbf{x}^{(\ell)} := \mathbf{x}^{(\ell-1)} + \Delta_\ell \mathbf{e}_\ell$ for $\ell = 1,\dots,N$, i.e., at the ℓth step only coordinate ℓ changes while all other coordinates are held fixed at their current values.

Given that the mechanism satisfies the (ε_ℓ, d_1)-Lipschitz bound along each coordinate, for each ℓ and for any $\mathbf{y}_k \in \mathcal{Y}$,

$$|\ln z(\mathbf{y}_k|\mathbf{x}^{(\ell)}) - \ln z(\mathbf{y}_k|\mathbf{x}^{(\ell-1)})| \le \varepsilon_\ell |\Delta_\ell|. \tag{46}$$

Iterating (46) along the N steps and telescoping yields

$$\begin{aligned} &\left|\ln z(\mathbf{y}_k|\mathbf{x}') - \ln z(\mathbf{y}_k|\mathbf{x})\right| \\ \le\ & \sum_{\ell=1}^{N} \left|\ln z(\mathbf{y}_k|\mathbf{x}^{(\ell)}) - \ln z(\mathbf{y}_k|\mathbf{x}^{(\ell-1)})\right| \le \sum_{\ell=1}^{N} \varepsilon_\ell |\Delta_\ell| \end{aligned} \tag{47}$$

Given $p \in [1,\infty]$, we let q be its Hölder dual ($1/p + 1/q = 1$). Define the vectors $\varepsilon = (\varepsilon_1,\dots,\varepsilon_N)$ and $\Delta = (\Delta_1,\dots,\Delta_N)$. By Hölder's inequality,

$$\sum_{\ell=1}^{N} \varepsilon_\ell |\Delta_\ell| \le \|\varepsilon\|_q \, \|\Delta\|_p. \tag{48}$$

Note that according to Eq. (18), and $q = \frac{p}{p-1}$

$$\sum_{\ell=1}^{N} \varepsilon_\ell^{\frac{p}{p-1}} \le \varepsilon^{\frac{p}{p-1}} \Rightarrow \sum_{\ell=1}^{N} \varepsilon_\ell^{q} \le \varepsilon^{q} \tag{49}$$

$$\Rightarrow \|\varepsilon\|_q \le \varepsilon, \tag{50}$$

and by definition $d_p(\mathbf{x}_a, \mathbf{x}_b) := \|\mathbf{x}_b - \mathbf{x}_a\|_p = \|\Delta\|_p$. Therefore, we can obtain that

$$\left|\ln z(\mathbf{y}_k|\mathbf{x}') - \ln z(\mathbf{y}_k|\mathbf{x})\right| \tag{51}$$

$$\le \sum_{\ell=1}^{N} \varepsilon_\ell |\Delta_\ell| \quad \text{(according to Eq. (47))} \tag{52}$$

$$\le \|\varepsilon\|_q \, \|\Delta\|_p \quad \text{(according to Eq. (48))} \tag{53}$$

$$\le \varepsilon d_p(\mathbf{x}_a, \mathbf{x}_b) \tag{54}$$

which is precisely the (ε, d_p)-Lipschitz bound between $\mathbf{x}$ and $\mathbf{x}'$.

Remarks on edge cases. For $p = 1$ (so $q = \infty$), the bound reduces to $\sum_\ell \varepsilon_\ell |\Delta_\ell| \le \|\varepsilon\|_\infty \|\Delta\|_1$. For $p = \infty$ (so $q = 1$), it becomes $\sum_\ell \varepsilon_\ell |\Delta_\ell| \le \|\varepsilon\|_1 \|\Delta\|_\infty$. Both cases are encompassed by (48) and thus by the argument above.

This establishes that coordinating the per-dimension budgets $\{\varepsilon_\ell\}_{\ell=1}^N$ via $\|\varepsilon\|_q \le \varepsilon$ yields a mechanism that satisfies the global (ε, d_p)-Lipschitz bound on $\mathcal{X}$. □

A.2 Proof of Theorem 2: Correctness of Interpolation Function

Theorem 2 (Correctness of Log-Convex Interpolation f_{int})**.** *Given that (ε_ℓ, d_1)-Lipschitz bound holds between each pair of ℓ-neighbors in $\hat{\mathcal{X}}$ and $\{\varepsilon_\ell\}_{\ell=1}^N$ satisfy the privacy budget composition condition formalized in Eq. (18)(19), the use of the interpolation function f_{int} (defined by Eq. (28)) guarantees that any two interpolated values within the entire secret data domain $\mathcal{X}$ satisfy (ε, d_p)-Lipschitz bound.*

Before proving Theorem 2, we introduce the following *Dimension-Wise Lipschitz bound*, which provides a sufficient condition for ensuring (ε, d_p)-mDP.

Definition 9 (Dimension-Wise Lipschitz bound (DW-Lipschitz bound))**.** *Given coordinate-specific privacy budgets $\varepsilon = \{\varepsilon_1,\dots,\varepsilon_N\}$, a mechanism f satisfies (ε, d_p)-*DW-Lipschitz bound *if for any two records $\mathbf{x}_a \in \mathcal{X}_m, \mathbf{x}_b \in \mathcal{X}_{m'}$ and perturbed value $\mathbf{y}_k \in \mathcal{Y}$, it holds that*

$$\frac{f(\mathbf{x}_a, \mathbf{y}_k, \mathbf{Z}_{\hat{\mathcal{X}}_m})}{f(\mathbf{x}_b, \mathbf{y}_k, \mathbf{Z}_{\hat{\mathcal{X}}_{m'}})} \le e^{\sum_{\ell=1}^{N} \varepsilon_\ell |x_{a,\ell} - x_{b,\ell}|}. \tag{55}$$

Satisfying the DW-Lipschitz bound ensures that the overall privacy leakage between any two records is controlled by their ℓ_p-norm distance. We formalize this connection in the following **Lemma 1**.

Lemma 1. *Suppose that an interpolation mechanism f satisfies (ε, d_p)-*DW-Lipschitz bound*, where $\varepsilon = (\varepsilon_1,\dots,\varepsilon_N)$ are coordinate-specific privacy parameters. Then, if the dimension-wise composition condition holds:*

$$\sum_{\ell=1}^{N} \varepsilon_\ell^{\frac{p}{p-1}} \le \varepsilon^{\frac{p}{p-1}} \quad \text{for} \quad p > 1, \tag{56}$$

$$\max_\ell \varepsilon_\ell \le \varepsilon \quad \text{for} \quad p = 1, \tag{57}$$

the mechanism $\mathcal{M}$ satisfies (ε, d_p)-Lipschitz bound.

Next, we introduce **Lemma 2–Lemma 4** as a preparation of the proof for **Theorem 2**. The detailed proofs of **Lemma 2–Lemma 4** can be found in [30].

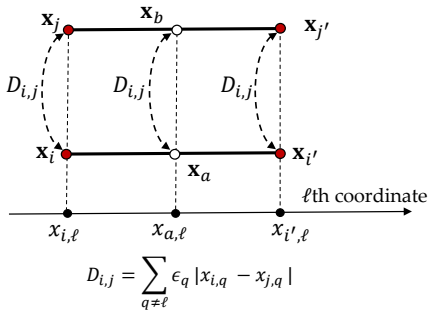


Figure 5: Proof of Lemma 3.

Lemma 2 (Chain Rule for (ε, d_p)-DW-Lipschitz bound)**.** *Let $\hat{\mathcal{X}}$ denote the set of anchor records obtained from the N-orthotope partitioning of the secret domain. If an interpolation mechanism f ensures that every pair of ℓ-neighbor anchors satisfies (ε, d_p)-DW-Lipschitz bound, then f also guarantees that any pair of anchors $\hat{\mathbf{x}}_i, \hat{\mathbf{x}}_j \in \hat{\mathcal{X}}$ satisfies (ε, d_p)-DW-Lipschitz bound.*

Lemma 3. *As shown in Fig. 5, let $\mathbf{x}_a$, $\mathbf{x}_b$, $\mathbf{x}_i$, $\mathbf{x}_{i'}$, $\mathbf{x}_j$, and $\mathbf{x}_{j'}$ be six points in $\mathbb{R}^N$ such that: (1) $\mathbf{x}_a$ shares the same coordinates with $\mathbf{x}_i$ and $\mathbf{x}_{i'}$ at all dimensions except ℓ; (2) $\mathbf{x}_b$ shares the same coordinates with $\mathbf{x}_j$ and $\mathbf{x}_{j'}$ at all dimensions except ℓ; (3) at dimension ℓ, $x_{a,\ell} = x_{b,\ell}$, and $x_{i,\ell} = x_{j,\ell}$, $x_{i',\ell} = x_{j',\ell}$.*

Suppose further that: both pairs $(\mathbf{x}_i, \mathbf{x}_j)$ and $(\mathbf{x}_{i'}, \mathbf{x}_{j'})$ satisfy (ε, d_p)-DW-Lipschitz bound; and the perturbation distributions satisfy log-convex interpolation:

$$z(\mathbf{y}_k \mid \mathbf{x}_a) \overset{\text{logcvx}}{\sim} (z(\mathbf{y}_k \mid \mathbf{x}_i), z(\mathbf{y}_k \mid \mathbf{x}_{i'})) \tag{58}$$

$$z(\mathbf{y}_k \mid \mathbf{x}_b) \overset{\text{logcvx}}{\sim} \left(z(\mathbf{y}_k \mid \mathbf{x}_j), z(\mathbf{y}_k \mid \mathbf{x}_{j'})\right). \tag{59}$$

Then, the pair $(\mathbf{x}_a, \mathbf{x}_b)$ satisfies (ε, d_p)-DW-Lipschitz bound, and hence (ε, d_p)-Lipschitz bound by Lemma 1 (under the composition condition).

Lemma 4. *Let $\mathbf{x}_i, \mathbf{x}_{i'} \in \mathcal{X}_m$ be any two points within an N-orthotope that differ only in the ℓth coordinate, and let $\mathbf{x}_a \in \mathcal{X}_m$ be any point such that $x_{i,\ell} \leq x_{a,\ell} \leq x_{i',\ell}$ and $\mathbf{x}_a$ also differs from $\mathbf{x}_i$ and $\mathbf{x}_{i'}$ only in the ℓth coordinate. Then, the interpolation function defined in Definition 5 satisfies the following log-convexity condition: $z(\mathbf{y}_k \mid \mathbf{x}_a) \overset{\text{logcvx}}{\sim} (z(\mathbf{y}_k \mid \mathbf{x}_i), z(\mathbf{y}_k \mid \mathbf{x}_{i'}))$.*

Next, we provide the detailed proof of **Theorem 2**:

Proof of Theorem 2. Without loss of generality, assume that $\mathbf{x}_a$ and $\mathbf{x}_b$ differ only in dimension 1, with $x_{a,1} < x_{b,1}$. Let $\mathbf{x}_a \in \mathcal{C}_i$ and $\mathbf{x}_b \in \mathcal{C}_j$, where the ranges of $\mathcal{C}_i$ and $\mathcal{C}_j$ along dimension ℓ are $[\hat{x}_{i,\ell}, \hat{x}_{i',\ell}]$ and $[\hat{x}_{j,\ell}, \hat{x}_{j',\ell}]$, respectively.

We construct two trees rooted at $\mathbf{x}_a$ and $\mathbf{x}_b$ as follows:

Intuitively, the two trees systematically decompose the difference between $\mathbf{x}_a$ and $\mathbf{x}_b$ along one dimension at a time. At each level ℓ, nodes extend along the ℓ-th coordinate toward their cell boundaries, progressively aligning the records with anchor points. This recursive structure allows us to apply a dimension-wise induction, ultimately connecting the original records to anchor records through a sequence of intermediate steps that satisfy local Lipschitz bound.

(a) Initialization (Level 0): Set $\mathbf{x}_a$ and $\mathbf{x}_b$ as the roots.

(b) Level 1 Construction: Extend both $\mathbf{x}_a$ and $\mathbf{x}_b$ along dimension 1 toward the boundaries of their respective cells, producing four points $\mathbf{x}_c, \mathbf{x}_d, \mathbf{x}_e, \mathbf{x}_f$, where:

$$x_{c,1} = \hat{x}_{i,1}, \quad x_{d,1} = \hat{x}_{i',1}, \quad x_{e,1} = \hat{x}_{j,1}, \quad x_{f,1} = \hat{x}_{j',1}. \tag{60}$$

Add $\mathbf{x}_c, \mathbf{x}_d$ as children of $\mathbf{x}_a$, and $\mathbf{x}_e, \mathbf{x}_f$ as children of $\mathbf{x}_b$ (see Fig. 6).

(c) Level ℓ Construction ($\ell = 2, \ldots, N$): For each node $\mathbf{x}_v$ at level $\ell - 1$, extend along dimension ℓ toward the cell boundaries, producing children $\mathbf{x}_{v'}$ and $\mathbf{x}_{v''}$, where $x_{v',\ell} = \hat{x}_{i,\ell}$ and $x_{v'',\ell} = \hat{x}_{i',\ell}$. All other coordinates are inherited from $\mathbf{x}_v$.

By construction, every node at level ℓ lies on the boundary along the first ℓ dimensions.

We now prove the theorem via **induction** on the tree levels:

(i) Base case (Level N): Each leaf node lies on the boundaries in all N dimensions, thus is an anchor record. Since every pair of ℓ-neighbor anchors satisfies (ε, d_p)-DW-Lipschitz bound by assumption, any pair of leaves satisfies (ε, d_p)-DW-Lipschitz bound (**Lemma 2**).

(ii) Inductive step: Suppose at level ℓ ($2 \leq \ell \leq N$), any pair of nodes satisfies (ε, d_p)-DW-Lipschitz bound. We show that at level $\ell - 1$, their parents also satisfy (ε, d_p)-DW-Lipschitz bound.

Consider two parent nodes $\mathbf{x}_v$ and $\mathbf{x}_u$ at level $\ell - 1$ with children $\{\mathbf{x}_{v'}, \mathbf{x}_{v''}\}$ and $\{\mathbf{x}_{u'}, \mathbf{x}_{u''}\}$, respectively. As shown in Fig. 7: (1) $\mathbf{x}_v$ shares all coordinates with $\mathbf{x}_{v'}$ and $\mathbf{x}_{v''}$ except along dimension ℓ; (2) $\mathbf{x}_u$ shares all coordinates with $\mathbf{x}_{u'}$ and $\mathbf{x}_{u''}$ except along dimension ℓ; and (3) $x_{v,\ell} = x_{u,\ell} = x_{a,\ell}$, $x_{v',\ell} = x_{u',\ell} = \hat{x}_{i,\ell}$, and $x_{v'',\ell} = x_{u'',\ell} = \hat{x}_{i',\ell}$.

Then, by applying **Lemma 4**, we can obtain

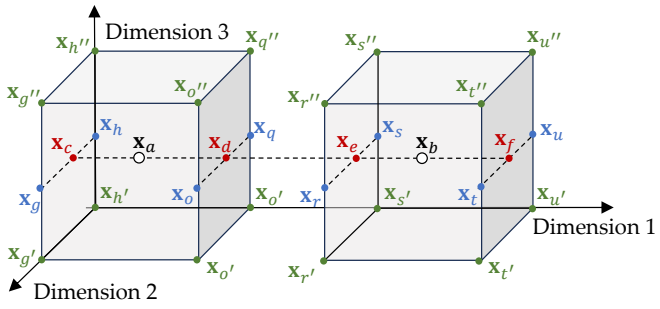
$$z(\mathbf{y}_k \mid \mathbf{x}_v) \overset{\text{logcvx}}{\sim} (z(\mathbf{y}_k \mid \mathbf{x}_{v'}), z(\mathbf{y}_k \mid \mathbf{x}_{v''})) \tag{61}$$

$$z(\mathbf{y}_k \mid \mathbf{x}_u) \overset{\text{logcvx}}{\sim} (z(\mathbf{y}_k \mid \mathbf{x}_{u'}), z(\mathbf{y}_k \mid \mathbf{x}_{u''})). \tag{62}$$

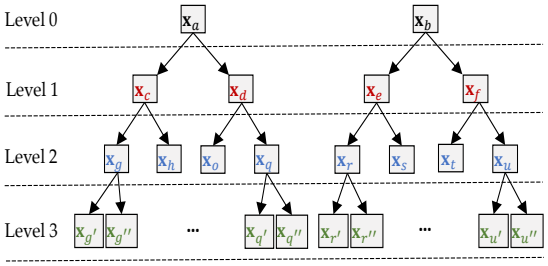
and then by **Lemma 3**, we can obtain the parent nodes $\mathbf{x}_v$ and $\mathbf{x}_u$ satisfy (ε, d_p)-DW-Lipschitz bound.

(iii) Conclusion (Level 0): At level 1, by induction, we have deduced that the four nodes $\mathbf{x}_c, \mathbf{x}_d, \mathbf{x}_e, \mathbf{x}_f$ satisfy (ε, d_p)-DW-Lipschitz bound. By applying **Proposition 2** (or **Proposition 1** if $\mathbf{x}_a$ and $\mathbf{x}_b$ lie in the same N-orthotope), we conclude that the original pair $(\mathbf{x}_a, \mathbf{x}_b)$ satisfies (ε, d_p)-DW-Lipschitz bound, and hence $\varepsilon, d_p)$-mDP (according to **Lemma 1**).

Finally, applying **Theorem 1** under Eqs. (18)-(19) upgrades the per-coordinate (ε_ℓ, d_1) bounds to the desired global (ε, d_p)-Lipschitz guarantee over $\mathcal{X}$. This completes the proof. □



(a) Node creation



(b) Nodes organized by the two trees

Figure 6: Illustration of node creation in Proof of Theorem 2 (Nodes are recursively extended along dimensions 1, 2, and 3 toward cell boundaries, reaching anchor points at the leaves).

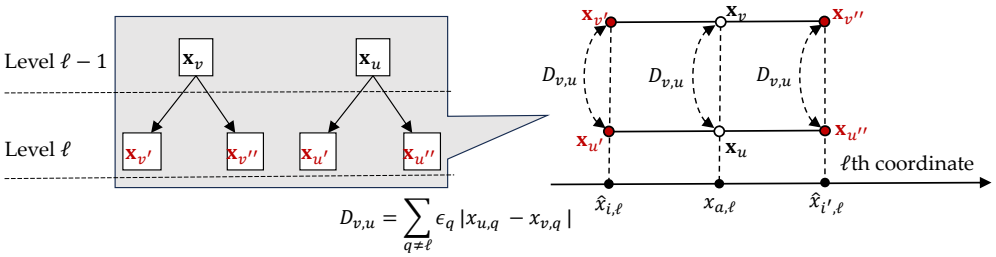


Figure 7: Illustration of Property **P-B**.

A.3 Proof of Proposition 4: Linear Surrogate For Utility Loss

Proposition 1 (Linear surrogate for utility loss)**.** *For any point* $\mathbf{x} = \hat{\mathbf{x}}_{i_m} + \lambda_a \odot \Delta \in \mathcal{C}_m$*, with the convex coefficients* $\lambda = [\lambda^1_{\hat{\mathbf{x}}_{i_m},\mathbf{x}}, \ldots, \lambda^N_{\hat{\mathbf{x}}_{i_m},\mathbf{x}}] \in [0,1]^N$*, approximate the perturbation probability by*

$$\Pr\big[\mathcal{M}(\mathbf{x}; \mathbf{Z}_{\hat{\mathcal{X}}_m}) = \mathbf{y}_k\big] \tag{63}$$

$$\approx \sum_{\gamma\in\{0,1\}^N} \prod_{\ell=1}^{N} \big((1-\gamma_\ell)\lambda^\ell_{\hat{\mathbf{x}}_{i_m},\mathbf{x}} + \gamma_\ell(1-\lambda^\ell_{\hat{\mathbf{x}}_{i_m},\mathbf{x}})\big) \tag{64}$$

$$\times z\big(\mathbf{y}_k \mid \hat{\mathbf{x}}_{i_m} + \gamma \odot \Delta\big) \tag{65}$$

the cell loss $\mathcal{L}(\mathbf{Z}_{\hat{\mathcal{X}}_m})$ *admits the linear surrogate* $\tilde{\mathcal{L}}(\mathbf{Z}_{\hat{\mathcal{X}}_m}) = \langle \tilde{\mathbf{C}}_{\hat{\mathcal{X}}_m}, \mathbf{Z}_{\hat{\mathcal{X}}_m} \rangle$*, where* $\tilde{\mathbf{C}}_{\hat{\mathcal{X}}_m} = \big\{\tilde{c}(\hat{\mathbf{x}}_i, \mathbf{y}_k)\big\}_{(\hat{\mathbf{x}}_i,\mathbf{y}_k)\in\hat{\mathcal{X}}_m\times\mathcal{Y}}$ *is a constant coefficient matrix depending only on the prior* $p(\mathbf{x})$ *and the utility loss* $\mathcal{L}(\mathbf{x},\mathbf{y}_k)$*. For each* $\hat{\mathbf{x}}_j = \hat{\mathbf{x}}_{i_m} + \gamma \odot \Delta \in \hat{\mathcal{X}}_m$

$$\tilde{c}(\hat{\mathbf{x}}_j, \mathbf{y}_k) \tag{66}$$

$$= \int_{\mathcal{C}_m} \prod_{\ell=1}^{N} \big((1-\gamma_\ell)\lambda^\ell_{\hat{\mathbf{x}}_{i_m},\mathbf{x}} + \gamma_\ell(1-\lambda^\ell_{\hat{\mathbf{x}}_{i_m},\mathbf{x}})\big) p(\mathbf{x})\mathcal{L}(\mathbf{x},\mathbf{y}_k)\mathrm{d}\mathbf{x}.$$

If $\mathcal{X}$ *is discretized,*

$$\tilde{c}(\hat{\mathbf{x}}_j, \mathbf{y}_k) \tag{67}$$

$$= \sum_{\mathbf{x}\in\mathcal{X}_m} \prod_{\ell=1}^{N} \big((1-\gamma_\ell)\lambda^\ell_{\hat{\mathbf{x}}_{i_m},\mathbf{x}} + \gamma_\ell(1-\lambda^\ell_{\hat{\mathbf{x}}_{i_m},\mathbf{x}})\big) p(\mathbf{x})\mathcal{L}(\mathbf{x},\mathbf{y}_k).$$

Proof. Let $\mathbf{x} = \hat{\mathbf{x}}_{i_m} + \lambda \odot \Delta$ where $\lambda = [\lambda^1_{\hat{\mathbf{x}}_{i_m},\mathbf{x}}, \ldots, \lambda^N_{\hat{\mathbf{x}}_{i_m},\mathbf{x}}] \in [0,1]^N$. By definition of the approximated mechanism, we have:

$$\begin{aligned}
&\tilde{\mathcal{L}}(\mathbf{Z}_{\hat{\mathcal{X}}_m}) \\
&= \sum_{\mathbf{y}_k\in\mathcal{Y}} \int_{\mathcal{C}_m} \Pr\Big[\mathcal{M}(\mathbf{x}, \mathbf{Z}_{\hat{\mathcal{X}}_m}) = \mathbf{y}_k\Big]\, p(\mathbf{x})\mathcal{L}(\mathbf{x},\mathbf{y}_k)\mathrm{d}\mathbf{x} \\
&\approx \sum_{\mathbf{y}_k\in\mathcal{Y}} \int_{\mathcal{C}_m} \Bigg[\sum_{\gamma\in\{0,1\}^N} \prod_{\ell=1}^{N} \big((1-\gamma_\ell)\lambda^\ell_{\hat{\mathbf{x}}_{i_m},\mathbf{x}} \\
&\quad + \gamma_\ell(1-\lambda^\ell_{\hat{\mathbf{x}}_{i_m},\mathbf{x}})\big) z(\mathbf{y}_k \mid \hat{\mathbf{x}}_{i_m} + \gamma\odot\Delta)\Bigg]\, p(\mathbf{x})\mathcal{L}(\mathbf{x},\mathbf{y}_k)\mathrm{d}\mathbf{x} \\
&= \sum_{\mathbf{y}_k\in\mathcal{Y}} \sum_{\gamma\in\{0,1\}^N} z(\mathbf{y}_k \mid \hat{\mathbf{x}}_{i_m} + \gamma\odot\Delta) \\
&\quad \times \int_{\mathcal{C}_m} \prod_{\ell=1}^{N} \big((1-\gamma_\ell)\lambda^\ell_{\hat{\mathbf{x}}_{i_m},\mathbf{x}} + \gamma_\ell(1-\lambda^\ell_{\hat{\mathbf{x}}_{i_m},\mathbf{x}})\big) p(\mathbf{x})\mathcal{L}(\mathbf{x},\mathbf{y}_k)\mathrm{d}\mathbf{x} \\
&= \sum_{(\hat{\mathbf{x}}_j,\mathbf{y}_k)\in\hat{\mathcal{X}}_m\times\mathcal{Y}} \tilde{c}(\hat{\mathbf{x}}_j,\mathbf{y}_k) z(\mathbf{y}_k \mid \hat{\mathbf{x}}_j),
\end{aligned} \tag{68}$$

where $\hat{\mathbf{x}}_j = \hat{\mathbf{x}}_{i_m} + \boldsymbol{\gamma} \odot \Delta$ and the coefficient

$$\begin{aligned} & \tilde{c}(\hat{\mathbf{x}}_j, \mathbf{y}_k) && (69)\\ = & \int_{\mathcal{C}_m} \prod_{\ell=1}^{N} ((1-\gamma_\ell)\lambda^{\ell}_{\hat{\mathbf{x}}_{i_m},\mathbf{x}} + \gamma_\ell(1-\lambda^{\ell}_{\hat{\mathbf{x}}_{i_m},\mathbf{x}})) && (70)\\ & \times p(\mathbf{x})\mathcal{L}(\mathbf{x},\mathbf{y}_k)\mathrm{d}\mathbf{x} && (71)\end{aligned}$$

is a constant with respect to $\mathbf{Z}$.

If $\mathcal{X}$ is discretized, then the integral becomes a summation:

$$\tilde{c}(\hat{\mathbf{x}}_j, \mathbf{y}_k) = \sum_{\mathbf{x}\in\mathcal{X}_m} \prod_{\ell=1}^{N} ((1-\gamma_\ell)\lambda^{\ell}_{\hat{\mathbf{x}}_{i_m},\mathbf{x}} + \gamma_\ell(1-\lambda^{\ell}_{\hat{\mathbf{x}}_{i_m},\mathbf{x}}))p(\mathbf{x})\mathcal{L}(\mathbf{x},\mathbf{y}_k). \tag{72}$$

In both cases, the resulting loss $\tilde{\mathcal{L}}(\mathbf{Z}_{\hat{\mathcal{X}}_m})$ is a linear function of the perturbation matrix $\mathbf{Z}_{\hat{\mathcal{X}}_m}$. □

B Optimal Gap Analysis

To evaluate how closely our solution can approach the true optimum, we derive a universal lower bound on the utility loss, labeled as **"LB"** in the following experiment part.

For any two cells $\mathcal{C}_m$ and $\mathcal{C}_{m'}$, we first define the following inter-cell distance: $\hat{d}_p(\mathcal{C}_m, \mathcal{C}_{m'}) := \max_{\mathbf{x}\in\mathcal{C}_m,\, \mathbf{x}'\in\mathcal{C}_{m'}} d_p(\mathbf{x},\mathbf{x}')$. Since $d_p(\mathbf{x},\mathbf{x}') \le \hat{d}_p(\mathcal{C}_m,\mathcal{C}_{m'})$ for all $\mathbf{x}\in\mathcal{C}_m, \mathbf{x}'\in\mathcal{C}_{m'}$, replacing d_p with $\hat{d}_p$ yields a relaxation of the original mDP constraints. We define the aggregated decision variables:

$$z(\mathbf{y}_k \mid \mathcal{C}_m) := \int_{\mathcal{C}_m} \Pr(\mathcal{M}(\mathbf{x}) = \mathbf{y}_k)\, \mathrm{d}\mathbf{x}, \tag{73}$$

and impose the relaxed mDP constraints:

$$z(\mathbf{y}_k \mid \mathcal{C}_m) - e^{\varepsilon \hat{d}_p(\mathcal{C}_m,\mathcal{C}_{m'})} z(\mathbf{y}_k \mid \mathcal{C}_{m'}) \le 0, \quad \forall \mathcal{C}_m, \mathcal{C}_{m'}. \tag{74}$$

and normalization constraint:

$$\sum_{\mathbf{y}_k\in\mathcal{Y}} z(\mathbf{y}_k \mid \mathcal{C}_m) = \int_{\mathcal{C}_m} \mathrm{d}\mathbf{x} = \prod_{\ell=1}^{N} \Delta_\ell. \tag{75}$$

Finally, define the minimal utility loss over each cell:

$$\bar{\mathcal{L}}_m(\mathbf{y}_k) := \min_{\mathbf{x}\in\mathcal{C}_m} p(\mathbf{x})\mathcal{L}(\mathbf{x},\mathbf{y}_k). \tag{76}$$

We now formulate the relaxed problem $\mathrm{APO}_{\mathrm{lb}}$ that minimizes a lower bound of the cell-aggregated utility loss:

$$\begin{aligned} \min \quad & \hat{\mathcal{L}}(\mathbf{Z}) := \sum_{m=1}^{M} \sum_{\mathbf{y}_k\in\mathcal{Y}} \bar{\mathcal{L}}_m(\mathbf{y}_k) \cdot z(\mathbf{y}_k \mid \mathcal{C}_m), && (77)\\ \text{s.t.} \quad & z(\mathbf{y}_k \mid \mathcal{C}_m) - e^{\varepsilon \hat{d}_p(\mathcal{C}_m,\mathcal{C}_{m'})} z(\mathbf{y}_k \mid \mathcal{C}_{m'}) \le 0,\ \forall \mathcal{C}_m, \mathcal{C}_{m'} && (78)\\ & \sum_{\mathbf{y}_k\in\mathcal{Y}} z(\mathbf{y}_k \mid \mathcal{C}_m) = \prod_{\ell=1}^{N} \Delta_\ell, \forall \mathcal{C}_m && (79)\end{aligned}$$

and denote its optimal value by $\hat{\mathcal{L}}^*$.

Proposition 5 (Universal Lower Bound). *Let $\mathcal{M} : \mathcal{X} \to \mathcal{Y}$ be any mechanism satisfying (ε, d_p)-mDP. Let*

$$\mathcal{L}(\mathcal{M}) := \int_{\mathcal{X}} \sum_{\mathbf{y}_k\in\mathcal{Y}} p(\mathbf{x})\mathcal{L}(\mathbf{x},\mathbf{y}_k) \Pr(\mathcal{M}(\mathbf{x}) = \mathbf{y}_k)\, \mathrm{d}\mathbf{x} \tag{80}$$

denote its expected utility loss. Then, $\mathcal{L}(\mathcal{M}) \ge \hat{\mathcal{L}}^*$.

Proof of Proposition 5. Fix any data perturbation mechanism $\mathcal{M}$ that satisfies (ε, d_p)-mDP. For a fixed output symbol $\mathbf{y}_k$ let

$$f(\mathbf{x}) := \Pr\big(\mathcal{M}(\mathbf{x}) = \mathbf{y}_k\big), \qquad \mathbf{x}\in\mathcal{X}. \tag{81}$$

For any two cells $\mathcal{C}_m, \mathcal{C}_{m'}$ and for every $\mathbf{x}\in\mathcal{C}_m$, $\mathbf{x}'\in\mathcal{C}_{m'}$ the mDP guarantee gives

$$\begin{aligned} & \frac{f(\mathbf{x})}{f(\mathbf{x}')} \le e^{\varepsilon d_p(\mathbf{x}_a,\mathbf{x}_b)} \le e^{\varepsilon \hat{d}_p(\mathcal{C}_m,\mathcal{C}_{m'})}, && (82)\\ & \hat{d}_p(\mathcal{C}_m,\mathcal{C}_{m'}) = \max_{\mathbf{u}\in\mathcal{C}_m,\, \mathbf{v}\in\mathcal{C}_{m'}} d_p(\mathbf{u},\mathbf{v}). && (83)\end{aligned}$$

Set $M_m := \sup_{\mathbf{x}\in\mathcal{C}_m} f(\mathbf{x})$, $m_{m'} := \inf_{\mathbf{x}'\in\mathcal{C}_{m'}} f(\mathbf{x}')$. By Eq. (82) we have

$$M_m \le e^{\varepsilon \hat{d}_p(\mathcal{C}_m,\mathcal{C}_{m'})} m_{m'}. \tag{84}$$

Since every grid cell has the same volume $V := \mu(\mathcal{C}_m) = \prod_{\ell=1}^{N} \Delta_\ell$,

$$z(\mathbf{y}_k \mid \mathcal{C}_m) := \int_{\mathcal{C}_m} f(\mathbf{x})\mathrm{d}\mathbf{x} \le V M_m,\ z(\mathbf{y}_k \mid \mathcal{C}_{m'}) \ge V m_{m'}. \tag{85}$$

Combining with Eq. (84) yields

$$z(\mathbf{y}_k \mid \mathcal{C}_m) \le e^{\varepsilon \hat{d}_p(\mathcal{C}_m,\mathcal{C}_{m'})} z(\mathbf{y}_k \mid \mathcal{C}_{m'}), \tag{86}$$

i.e. the vector $\{z(\mathbf{y}_k \mid \mathcal{C}_m)\}$ satisfies the relaxed mDP constraints (74). Normalisation also holds because $\sum_{\mathbf{y}_k} f(\mathbf{x}) = 1$ for every $\mathbf{x}$. Hence, *the aggregated variables induced by $\mathcal{M}$ form a feasible solution of* $\mathrm{APO}_{\mathrm{lb}}$.

Now, we can derive the expected loss of $\mathcal{M}$ as

$$\begin{aligned} & \mathcal{L}(\mathcal{M}) && (87)\\ = & \sum_{\mathbf{y}_k\in\mathcal{Y}} \int_{\mathcal{X}} p(\mathbf{x})\mathcal{L}\big(\mathbf{x},\mathbf{y}_k\big) f(\mathbf{x})\mathrm{d}\mathbf{x} && (88)\\ = & \sum_{m=1}^{M} \sum_{\mathbf{y}_k\in\mathcal{Y}} \int_{\mathcal{C}_m} p(\mathbf{x})\mathcal{L}\big(\mathbf{x},\mathbf{y}_k\big) f(\mathbf{x})\mathrm{d}\mathbf{x} && (89)\\ \ge & \sum_{m=1}^{M} \sum_{\mathbf{y}_k\in\mathcal{Y}} \underbrace{\min_{\mathbf{x}\in\mathcal{C}_m} p(\mathbf{x})\mathcal{L}\big(\mathbf{x},\mathbf{y}_k\big)}_{=\bar{\mathcal{L}}_m(\mathbf{y}_k)} z(\mathbf{y}_k \mid \mathcal{C}_m) && (90)\\ = & \hat{\mathcal{L}}\big(\mathbf{Z}\big). && (91)\end{aligned}$$

Because $\mathbf{Z}$ is feasible for $\mathrm{APO}_{\mathrm{lb}}$, $\hat{\mathcal{L}}(\mathbf{Z}) \ge \hat{\mathcal{L}}^*$. Therefore $\mathcal{L}(\mathcal{M}) \ge \hat{\mathcal{L}}^*$, establishing that $\hat{\mathcal{L}}^*$ is a universal lower bound on the utility loss of any (ε, d_p)-mDP mechanism. □