Operations Research

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Differential Privacy in Personalized Pricing with Nonparametric Demand Models

Xi Chen, Sentao Miao, Yining Wang

To cite this article:
Xi Chen, Sentao Miao, Yining Wang (2022) Differential Privacy in Personalized Pricing with Nonparametric Demand Models. Operations Research

Published online in Articles in Advance 31 Aug 2022

. https://doi.org/10.1287/opre.2022.2347

informs.
http://pubsonline.informs.org/journal/opre

OPERATIONS RESEARCH
Articles in Advance, pp. 1–22
ISSN 0030-364X (print), ISSN 1526-5463 (online)

Crosscutting Areas

Differential Privacy in Personalized Pricing with Nonparametric Demand Models

Xi Chen,[a] Sentao Miao,[b,*] Yining Wang[c]

[a] Leonard N. Stern School of Business, New York University, New York, New York 10012; [b] Desautels Faculty of Management, McGill University, Montreal, Quebec H3A 1G5, Canada; [c] Naveen Jindal School of Management, University of Texas at Dallas, Richardson, Texas 75080
*Corresponding author

Contact: xc13@stern.nyu.edu, https://orcid.org/0000-0002-9049-9452 (XC); sentao.miao@mcgill.ca, https://orcid.org/0000-0002-0380-0797 (SM); yining.wang@utdallas.edu (YW)

Received: September 15, 2021
Revised: April 9, 2022
Accepted: July 11, 2022
Published Online in Articles in Advance: August 31, 2022

Area of Review: Revenue Management and Market Analytics

https://doi.org/10.1287/opre.2022.2347

Abstract. In recent decades, the advance of information technology and abundant personal data facilitate the application of algorithmic personalized pricing. However, this leads to the growing concern of potential violation of privacy because of adversarial attack. To address the privacy issue, this paper studies a dynamic personalized pricing problem with *unknown* nonparametric demand models under data privacy protection. Two concepts of data privacy, which have been widely applied in practices, are introduced: *central differential privacy (CDP)* and *local differential privacy (LDP)*, which is proved to be stronger than CDP in many cases. We develop two algorithms that make pricing decisions and learn the unknown demand on the fly while satisfying the CDP and LDP guarantee, respectively. In particular, for the algorithm with CDP guarantee, the regret is proved to be at most $\widetilde{O}(T^{(d+2)/(d+4)} + \varepsilon^{-1}T^{d/(d+4)})$. Here, the parameter T denotes the length of the time horizon, d is the dimension of the personalized information vector, and the key parameter $\varepsilon > 0$ measures the strength of privacy (smaller ε indicates a stronger privacy protection). Conversely, for the algorithm with LDP guarantee, its regret is proved to be at most $\widetilde{O}(\varepsilon^{-2/(d+2)}T^{(d+1)/(d+2)})$, which is near optimal as we prove a lower bound of $\Omega(\varepsilon^{-2/(d+2)}T^{(d+1)/(d+2)}/d^{7/3})$ for any algorithm with LDP guarantee.

Funding: X. Chen is supported by the National Science Foundation [Grant IIS-1845444].
Supplemental Material: The online appendix is available at https://doi.org/10.1287/opre.2022.2347.

Keywords: differential privacy • dynamic pricing • local privacy • regret

1. Introduction

From the early day bargaining to customized prices based on such as customer groups (e.g., student versus nonstudent), genders (e.g., personal care products; De Blasio and Menin 2015), and regions (e.g., regional prices of AIDS drug Combivir; Cowen and Tabarrok 2015), personalized pricing has long been implemented in many commercial activities. With the recent advance of information technology, the pricing platform could use customer data more efficiently, and personalized prices can be set algorithmically. For example, insurance companies quote the premium based on customers' demographic and behavioral data (Arumugam and Bhargavi 2019); hoteling websites charge different prices based on customers' locations and devices (Vissers et al. 2014). Besides industry practices, there is also a growing body of academic research on algorithmic personalized pricing (see related literature in Section 1.1).

With this surge of algorithmic personalized pricing, there is growing concern of privacy issue because of potential leakage of customers' personal information. As quoted in a report by the Organisation for Economic Co-operation Directorate for Financial and Enterprise Affairs,[1] "Similarly, the collection and use of personal data used in personalized pricing could implicate privacy concerns." However, most of the practices in personalized pricing to protect data privacy are quite ad hoc such as anonymizing personal information, which cannot guarantee the data security (Federal Trade Commission 2012, Kolata 2019). Furthermore, even if the adversary does not have direct access to the data set, they are still able to reconstruct customers' personal information by interacting with and observing the decisions made by the pricing platform (Fredrikson et al. 2014, Hidano et al. 2017). To address these malicious attack to personal data, Dwork et al. (2006a, b) proposed an important concept of the so-called *differential privacy*, which is the de facto privacy standard in practice. More specifically, there are mainly two types of differential privacy that are widely used in practice: *central differential privacy*

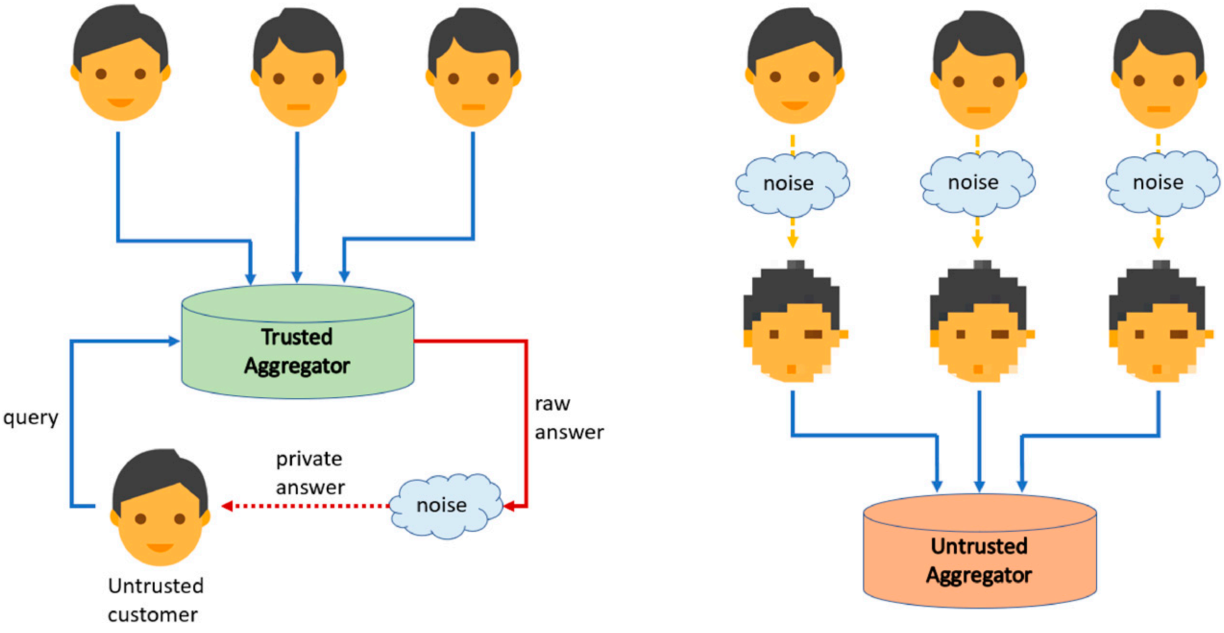
(CDP) (Dwork and Roth 2014) and *local differential privacy (LDP)* (Evfimievski et al. 2003; Duchi et al. 2013, 2018). Intuitively, CDP guarantees that for any time t, the adversary is unlikely, depending on a privacy parameter $\varepsilon > 0$ (smaller ε leads to higher security; hence, it is also called ε-CDP) to infer the data of any customer who has arrived before t. For example, the U.S. Census Bureau (2020) applied the techniques of CDP for the privacy of census data. Although for LDP, the customer does not even trust the platform, so that the platform can *only* use privatized historical data to make decisions. As a result, an adversary is unlikely (again, depending on ε, also known as ε-LDP) to obtain other customers' data even if it has direct access to the platform's data set. Examples of practices of LDP include Google (2014) and Apple (2019). We refer to Figure 1 for a graphical representation of CDP and LDP. As shown in the left panel of Figure 1, for CDP, the *trusted* aggregator (i.e., the platform) collects historical data and the query from the arriving customer, and it outputs a *privatized* answer (e.g., price) such that an adversary cannot infer sensitive information from this answer. The LDP is illustrated in the right panel of Figure 1, where the aggregator (e.g., the platform) is *untrusted* so that it can only collect privatized/perturbed data from customers. Please refer to Section 3 for the detailed formulation and comparisons between CDP and LDP in our setting.

In this paper, we address the concern of protecting customers' data in a dynamic personalized pricing problem with demand learning. Briefly, there is a finite selling horizon with length T, and in each time period t, there is one customer with a d-dimensional feature/data vector $\boldsymbol{x}_t \in \mathbb{R}^d$ arriving at the platform for purchasing a single product. To maximize the cumulative revenue, the platform needs to decide a personalized price p_t based on the knowledge of the unknown demand model (from historical data) and $\boldsymbol{x}_t$ while at the same time protecting customer's data. Assuming the demand model to be *nonparametric*, we propose two algorithms which protect data privacy by satisfying ε-CDP and ε-LDP, respectively. The main contributions of this paper are summarized as follows.

Protecting data privacy with nonparametric demand model. As mentioned, the demand model in this paper is assumed to be nonparametric (see Chen and Gallego 2021 for preceding work without data privacy), as opposed to the existing literature on data privacy in pricing which assumes parametric demand (Lei et al. 2020, Han et al. 2021, Chen et al. 2022). The preservation of data privacy in such nonparametric settings gives rises to several technical challenges, which we describe in more details later.

- In the work of Chen and Gallego (2021), the authors divide the space of contextual vectors into local *hypercubes* (i.e., each arriving customer belongs to a certain hypercube depending on his or her data $\boldsymbol{x}_t$), and a pricing algorithm is proposed that runs some multi-armed bandit algorithm for each hypercube. However, the dynamic pricing method is different: in the work of Chen and Gallego (2021), each hyper-cube is discretized into $\lceil \ln T \rceil$ price candidates in parallel, and a successive elimination type algorithm is used. In contrast,

Figure 1. (Color online) Illustration of CDP and LDP



Note. CDP (left) and LDP (right) in a general setting.

our paper uses a local quadrisection search method with a fixed update schedule within each context hypercube. The motivation for this approach is to minimize the number of statistics maintained by the algorithm at any time (four instead of $\lceil \ln T \rceil$), so that statistically efficient de-anonymization could be effectively carried out.

- Many existing works on pricing with nonparametric demands (either contextual or not) use the *trisection search* approach to search for the revenue-optimal prices by exploiting the concavity structures of the revenue function (with respect to price) without other prior knowledge (Lei et al. 2014, Wang et al. 2014). Although the trisection search approach has the advantage of maintaining a constant number of cumulative revenue statistics (in contrast to the previously mentioned multiarmed bandit approach), there is another subtle technical difficulty of directly applying it in the privacy preserving setting: when working with trisection search, we need a preallocated sample budget for each iteration to decide when to stop comparing the two midpoints. With data privacy (especially local privacy) constraints, it is difficult for the algorithm to maintain the number of samples or customers already arrived who belong to a certain hypercube. Therefore, the algorithm has difficulty knowing when to stop comparing midpoints in a trisection search procedure.

To address the previously mentioned technical challenges, in this paper, we use the idea of *quadrisection search* within each hypercube of customers' contextual vectors. In quadrisection search, the price interval is divided equally into four pieces with three midpoints. This gives the algorithm more information to decide the direction of price interval shrinkage, without the need to maintain an accurate counting of customers arriving in each hypercube during a certain time range. It also avoids the regret inflation problem from multiarmed bandit approaches because at each time the quadrisection search method maintains only 5 (or 10) anonymized statistics for each hypercube, which will not add unreasonable cost due to privacy preservation.

Near-optimal pricing algorithms preserving data privacy. We present two algorithms named *CPPQ* and *LPPQ* for nonparametric personalized pricing with privacy guarantees. Both algorithms satisfy ε-CDP (for *CPPQ*) or ε-LDP (for *LPPQ*) constraints regardless of choices of algorithm input parameters (except ε); thus, in practice, we can tune the input parameters for better performance without worrying about privacy preservation. In addition to the local quadrisection search method mentioned in the previous paragraph, the proposed privacy-preserving algorithms also use several advanced techniques such as tree-based aggregation and noisy statistical counts to ensure customers' data privacy, which we discuss and explain in more details later in the paper when we describe the proposed algorithms.

In addition to rigorous privacy guarantees, we also demonstrate that when certain algorithm parameters are carefully chosen, the proposed algorithms have near-optimal regrets (up to logarithmic factors in T). More specifically, for the CDP setting, the proposed algorithm enjoys a regret upper bound of $\widetilde{O}(T^{(d+2)/(d+4)} + \varepsilon^{-1}T^{d/(d+4)})$, which is optimal when ε is not too small because $\Omega(T^{(d+2)/(d+4)})$ is a known *lower bound* for personalized pricing with nonparametric demands even when data privacy is not of concern (Chen and Gallego 2021). The notation $\widetilde{O}(\cdot)$ hides some logarithmic factors in T (see Theorem 1 for more details). For the more challenging LDP setting,[2] our proposed LPPQ algorithm achieves a regret upper bound of $\widetilde{O}(\varepsilon^{-2/(d+2)}T^{(d+1)/(d+2)})$ (see Theorem 2). Although this regret upper bound is considerably worse than the $\widetilde{O}(T^{(d+2)/(d+4)})$ scaling even if $\varepsilon = \Omega(1)$, we show that it is indeed *rate-optimal*, as explained in the next paragraph.

Minimax lower bound for locally private personalized pricing. In this paper, we prove a minimax lower bound of $\Omega(\varepsilon^{-2/(d+2)}T^{(d+1)/(d+2)}/d^{7/3})$ on the regret of *any* possible personalized pricing policy for T sequentially arriving customers subject to the ε-LDP constraint (see Theorem 3). Although minimax lower bounds of locally private estimators have been previously studied (Duchi et al. 2018), the lower bound in our problem setting is more complicated because the prices offered to sequentially arriving customers are *adaptive* and not independently distributed with respect to any underlying distribution, which is the case in the work of Duchi et al. (2018). To establish a lower bound for *any adaptive* personalized pricing strategy subject to local privacy constraints, we carefully generalize the information theoretical arguments in Duchi et al. (2018) to adaptively collected data and obtain a tight minimax lower bound. More details and discussion is presented in Section 6.

1.1. Related Literature

This section reviews some related literature from two streams: the theory and application of data privacy, and the related paper in personalized pricing with demand learning.

Literature in Data Privacy. The concept of data privacy was first rigorously quantified by an important framework called differential privacy (DP), which was first introduced in Dwork et al. (2006a, b). The definition of DP has become a de facto standard for data privacy in both academic and industrial practice. We refer the interested readers to Dwork and Roth (2014) and Acquisti et al. (2016) for comprehensive reviews. Based on this concept, different theories and techniques have been developed, such as the technique of tree-based aggregation (Chan et al. 2011), which is used in the design of our algorithm CPPQ, and the mechanism of random perturbation (Dwork and Roth 2014), which is an important component in the design of both CPPQ and LPPQ. For LDP, as discussed earlier this notion is mentioned in Duchi et al. (2018). Besides the difference of the problem and model (as ours is a pricing problem which has specific structures), our paper considers an online decision-making problem instead of a static statistical problem where the data are collected passively as in Duchi et al. (2018). As a result, both the notion of DP (CDP and LDP) and the techniques used in our paper are significantly different, which will be elaborated in our main context.

A more related stream of literature to ours is the so-called online learning with differential privacy. That is, the decision maker has to learn the environment and make decisions on the fly while preserving the data privacy. For example, Mishra and Thakurta (2015) proposed (central) differentially private upper-confidence bound (UCB) and Thompson sampling algorithms for multiarmed bandit (MAB) problems. Shariff and Sheffet (2018) studied linear contextual bandit under the CDP constraint, and Ren et al. (2020) and Zheng et al. (2020) studied the (contextual) bandit problem with ε-LDP guarantee. Later, Han et al. (2021) extended the results to generalized linear bandits with stochastic contexts. In particular, the authors leveraged the idea of stochastic gradient descent and proposed a novel LDP strategy so that their algorithms are proved to have regret $O((\ln(T)/\varepsilon)^2)$ or $\widetilde{O}(T^{(1-\beta)/2}/\varepsilon^{1+\beta})$, depending on whether some "optimality margin" parameter $\beta = 1$ or $\beta \in (0, 1)$. Our work differ from the private MAB literature in that our model is nonparametric as opposed to the (generalized) linear model in MAB literature. Therefore, we cannot preserve the privacy (either centrally or locally) through protecting the demand parameters as in parametric models. Besides MAB problems, there are some other differentially private online learning problems such as private sequential learning (Xu 2018, Tsitsiklis et al. 2021, Xu et al. 2021), and dynamic pricing (Chen et al. 2022), which will be discussed later in literature review in personalized pricing.

We also note that there is a growing body of literature on data privacy in service systems. For instance, Hu et al. (2022) studied customer-centric privacy management under queueing models, where customers are strategic in deciding whether to disclose private personal information to the service provider.

Literature in Personalized Pricing with Demand Learning. As we discussed in the Introduction, algorithmic dynamic pricing with demand learning has been increasingly popular especially in recent years (for an incomplete list of literature, see Araman and Caldentey 2009, Besbes and Zeevi 2009, Farias and Van Roy 2010, Harrison et al. 2012, Broder and Rusmevichientong 2012, den Boer and Zwart 2013, Wang et al. 2014, Besbes and Zeevi 2015, Chen et al. 2015, Cheung et al. 2017, Ferreira et al. 2018, Chen and Gallego 2021, Miao et al. 2021, Wang et al. 2021). With the abundance of customer's personal data, there is also a growing trend of implementing personalized prices based on customers' contextual information. For instance, Qiang and Bayati (2016) applied a greedy iterated least squares method on linear demand function. Ban and Keskin (2021) and Javanmard and Nazerzadeh (2019) studied the high-dimensional personalized pricing with parametric demand model and sparse parameters. Keskin et al. (2020) leveraged data clustering for customized electricity pricing. The most related work to ours is Chen and Gallego (2021), who consider the same problem of nonparametric personalized pricing but without privacy guarantee. However, because of the privacy guarantee, our paper has to use a different pricing technique that is based on a "local quadrisection search" (to be specified later in the main context) instead of some successive elimination type of algorithm as in Chen and Gallego (2021). Moreover, we further show that with ε-LDP, the lower bound of the regret is significantly different from the one in Chen and Gallego (2021), showing that ensuring (local) privacy sets a fundamental limit on what we can achieve for any algorithm's performance.

With widespread public concern of personal data security, several recent works in personalized pricing start to take the data privacy into consideration (Lei et al. 2020, Tang et al. 2020, Bimpikis et al. 2021, Chen et al. 2022). Among this literature, the most related work to this paper is Chen et al. (2022), which also consider a dynamic personalized pricing problem with differential privacy guarantee. Compared with this paper, our work has the following differences. First, Chen et al. (2022) studied a parametric demand model, whereas this work considers a nonparametric model, making the demand learning and privacy protection completely different. Second, the differential privacy studied in Chen et al. (2022) is the ε-CDP, whereas our paper not only proposes an algorithm for ε-CDP but also develops an algorithm for ε-LDP—a stronger notation of privacy than ε-CDP in many cases. Because of these differences, especially for ε-LDP, none of the techniques (e.g., UCB algorithm, differentially private maximum likelihood estimation) in Chen et al. (2022) can be applied to our problem. Another related paper to ours is Lei et al. (2020), who also studied pricing algorithms satisfying ε-CDP and ε-LDP. The difference in their work is that their setting is an offline pricing problem, whereas ours is online dynamic pricing with demand learning, and the demand model considered in Lei et al. (2020) is parametric (in contrast to the nonparametric model in this paper). To the best of our knowledge, this paper is the first to consider a dynamic personalized pricing problem with nonparametric demand under both CDP and LDP guarantees.

Other Related Literature on Dynamic Pricing with Demand Learning. The work of den Boer and Keskin (2020) studied the revenue management and pricing question when the underlying demand/revenue curve could have multiple discontinuity points. Smoothness (twice continuous differentiability with bounded derivatives) of the demand function is essential to the algorithm and analysis in this paper because of the hyper-cube discretization technique adopted. The works of Birge et al. (2021a, b) studied the dynamic pricing with demand learning question when incoming customers exhibit certain strategic purchase behaviors, such as evaluating price offerings over an extended "patience" window (Birge et al. 2021a) or being involved in strategic betting behaviors ("bluffing") to influence sellers' revenue decisions (Birge et al. 2021b). It is an interesting question to extend privacy considerations to pricing problems with strategic customers. The work of Keskin and Zeevi (2018) studies the "myopic" policy (also known as greedy or follow-the-leader policy) under stationary or changing environments. It is shown that for many parametric models, the myopic policy diverges and is suboptimal with positive probability. It is an interesting research question to study, at least under a parametric model, whether artificial noise calibrated because of privacy constraints could lead to "inherent exploration" on top of a myopic policy, eventually leading to optimal asymptotic regret (Keskin and Zeevi 2014).

1.2. Paper Organization

The rest of this paper is organized as follows. In Section 2, the model and some technical assumptions are introduced. Section 3 introduces the important definitions of ε-CDP and ε-LDP in our problem setting, and their relationship. After that, the algorithms satisfying ε-CDP and ε-LDP are introduced in Section 4 and Section 5, respectively. Furthermore, we develop a lower bound for any algorithm with ε-LDP in Section 6. In Section 7, some numerical experiments are used to illustrate the performance of the proposed algorithms. In the end, the paper is concluded in Section 8. Some technical proofs can be found in the online appendix.

2. Model and Assumptions

We study a stylized dynamic personalized pricing problem of a single type of product for T consecutive selling periods. At the beginning of selling period t, the pricing platform observes a contextual vector $\boldsymbol{x}_t \in \mathcal{X} \subseteq \mathbb{R}^d$ of the incoming customer. The platform then offers a price $p_t \in [\underline{p}, \overline{p}]$, and the stochastically realized demand $y_t \in \mathcal{Y} \subseteq \mathbb{R}^+$ is being modeled by an unknown nonparametric model $\lambda : [\underline{p}, \overline{p}] \times \mathcal{X} \to \mathbb{R}^+$ as

$$\mathbb{E}[y_t | \boldsymbol{x}_t, p_t] = \lambda(p_t, \boldsymbol{x}), \quad y_t \in \mathcal{Y}. \tag{1}$$

Throughout the paper, we also define $f(p, \boldsymbol{x}) := p\lambda(p, \boldsymbol{x})$ as the function that gives the expected revenue of price p conditioned on customer context $\boldsymbol{x}$.

Clearly, when λ (and subsequently f) is known a priori to the platform, the optimal pricing strategy (without considerations of privacy concerns) would be to simply set $p_t = p^*(\boldsymbol{x}_t) = \arg\max_{p \in [\underline{p}, \overline{p}]} f(p, \boldsymbol{x}_t)$. Without knowing λ or f, on the other hand, requires the platform to learn the unknown demand model and offer near-optimal personalized prices simultaneously, commonly known in the literature as the *exploration-exploitation tradeoff*. We adopt the classical measure of *cumulative regret* (we also call it *regret* for brevity) to measure the performance of a pricing policy π over T time periods. (See the next section for a rigorous definition of an admissible pricing policy and when it satisfies privacy guarantees.) More specifically, the regret of a policy π under model f is defined as

$$\mathfrak{R}_T(f, \pi) := \mathbb{E}\left[\sum_{t=1}^{T} (f(p^*(\boldsymbol{x}_t), \boldsymbol{x}_t) - f(p_t, \boldsymbol{x}_t))\right],$$
$$\text{where } \; p^*(\boldsymbol{x}_t) = \arg\max_{p \in [\underline{p}, \overline{p}]} f(p, \boldsymbol{x}_t). \tag{2}$$

We make the following assumptions throughout this paper:

Assumption 1. *The domains of* (x_t, y_t, p_t) *satisfy* $\mathcal{X} \subseteq [0,1]^d$, $\mathcal{Y} \subseteq [0, 1]$ *and* $[\underline{p}, \overline{p}] \subset [0, 1]$. *Furthermore,* $\boldsymbol{x}_t$ *are independent and identically distributed (i.i.d.) over* $t \in [T]$ *thatwhich follows an unknown underlying distribution* P_X *supported on* $\mathcal{X}$ *with probability density function* $\chi : \mathcal{X} \to \mathbb{R}^+$ *that satisfies* $\sup_{\boldsymbol{x} \in \mathcal{X}} \chi(\boldsymbol{x}) \le C_X$ *almost surely.*

Assumption 2. *There exists a finite constant* $C_L < \infty$ *such that* $|f(p, \boldsymbol{x}) - f(p', \boldsymbol{x}')| \le C_L(|p - p'| + \|\boldsymbol{x} - \boldsymbol{x}'\|_2)$ *for all* $p, p' \in [\underline{p}, \overline{p}]$ *and* $\boldsymbol{x}, \boldsymbol{x}' \in \mathcal{X}$.

Assumption 3. *For any hypercube* $B \subseteq \mathcal{X}$ *and* $p \in [\underline{p}, \overline{p}]$, *define the expected revenue and the optimal price on the hypercube* B,

$$f_B(p) := \mathbb{E}_{P_X}[f(p, \boldsymbol{x}) | \boldsymbol{x} \in B], \qquad p^*(B) = \arg\max_{p \in [\underline{p}, \overline{p}]} f_B(p). \tag{3}$$

Then there exist uniform constants $0 < \sigma_H \le C_H < \infty$ *and* $C_p < \infty$ *such that*

(a) $p^*(B) \in (\underline{p}, \overline{p})$; *furthermore,* $f_B(\cdot)$ *is twice continuously differentiable in* p *and satisfies* $\sigma_H^2 \le -f_B''(p) \le C_H^2$ *for all* $p \in (\underline{p}, \overline{p})$;

(b) $\inf_{\boldsymbol{x} \in B} p^*(\boldsymbol{x}) \le p^*(B) \le \sup_{\boldsymbol{x} \in B} p^*(\boldsymbol{x})$;

(c) $\sup_{\boldsymbol{x}\in B} p^*(\boldsymbol{x}) - \inf_{\boldsymbol{x}\in B} p^*(\boldsymbol{x}) \le C_p \sup_{\boldsymbol{x},\boldsymbol{x}'\in B}\|\boldsymbol{x}-\boldsymbol{x}'\|_2$.

Assumptions 1 and 2 are standard regularity assumptions in the literature (Qiang and Bayati 2016, Javanmard and Nazerzadeh 2019, Ban and Keskin 2021 for parametric demand models satisfying these two assumptions). Assumption 3 follows the model setup in the work of Chen and Gallego (2021). First, the intuition of $f_B(p)$ is the average revenue of price p when the context $\boldsymbol{x}$ is in the hypercube B. One can think of a case where the retailer can only observe $\mathbf{1}\{\boldsymbol{x}\in B\}$ instead of the value of $\boldsymbol{x}$, so that the best pricing strategy is obviously to charge $p^*(B)$. For the three technical conditions of Assumption 3, (b) and (c) are exactly the same as in Chen and Gallego (2021) (in particular, parts 2 and 3 of assumption 3). The only slightly different assumption compared with (part 1 of) assumption 3 in Chen and Gallego (2021) (which basically assumes that $|f_B(p^*(B)) - f_B(p)| = \Omega((p^*(B) - p)^2)$) is our (a). On one hand, as shown in proposition 1 of Chen and Gallego (2021), our Assumption 3(a) implies its counterpart in the first part of assumption 3 in Chen and Gallego (2021). On the other hand, for all the motivating examples studied in remark 1 in Chen and Gallego (2021) (e.g., the linear covariate case, separable demand functions, and localized functions), our Assumption 3(a) is satisfied as well.

3. Central and Local Differential Privacy

This section introduces two important concepts of differential privacy: the CDP and LDP. In the following two sections, we will discuss each of these two concepts adapted to our problem setting and illustrate their relationship, especially their differences.

3.1. Central Differential Privacy

We first introduce the standard definition of (central) differential privacy (Dwork et al. 2006a, b; Dwork and Roth 2014) for offline problems.

Definition 1 (Differential Privacy). Let $s_t := (\boldsymbol{x}_t, y_t, p_t) \in \mathcal{S} := \mathcal{X}\times\mathcal{Y}\times[\underline{p},\overline{p}]$. Let $\pi : (s_1, \cdots, s_T) \longmapsto o$ be an offline randomized algorithm that takes customers' sensitive information as input and outputs statistics or decision $o \in \mathcal{O}$. For any $\varepsilon > 0$, π satisfies ε-differential privacy if for all $s_1, \cdots, s_t, \cdots, s_T$ and $s_t' \ne s_t$, and any measurable $O \subseteq \mathcal{O}$, it holds that

$$\Pr[\pi(s_1,\cdots,s_t,\cdots,s_T)\in O] \le e^{\varepsilon}\cdot \Pr[\pi(s_1,\cdots,s_t',\cdots,s_T)\in O].$$

In Definition 1, the output domain $\mathcal{O}$ captures all information released by the randomized algorithm π to the general public, which would be the offered prices $\{p_t\}_{t=1}^T$ in our setting. Such a definition, however, poses technical challenges in the context of dynamic personalized pricing as the price p_t for customer arriving at time t is highly indicative of the customer's personal information (section 4.2 in Chen et al. 2022). More specifically, proposition 1 in Chen et al. (2022) proves that any pricing policy satisfying Definition 1 will have worst-case regret at least $\Omega(T)$, suggesting that this definition is too strong for our setting. As a result, in the works of Shariff and Sheffet (2018) and Chen et al. (2022), the notion of *anticipating differential privacy* (for brevity of notation, we just call it central differential privacy, or CDP) is introduced to focus on the impact of sensitive customers' information on *future* prices as formalized here.[3]

Definition 2 (Central Differential Privacy for Personalized Pricing). Let data of customer t be $s_t := (\boldsymbol{x}_t, y_t, p_t) \in \mathcal{S} := \mathcal{X}\times\mathcal{Y}\times[\underline{p},\overline{p}]$. Let $\pi = (A_1, A_2, \cdots, A_T)$ be an admissible personalized pricing policy, where $A_t(\cdot|s_1, \cdots, s_t)$ is a distribution of $p_t \in [\underline{p},\overline{p}]$ measurable conditioned on $\{s_1, \cdots, s_t\}$. We say π satisfies ε-central differential privacy (ε-CDP) if for all $j < t$, $s_1, \cdots, s_j, \cdots, s_{t-1}$ and $s_j' \ne s_j$ with $y_t, y_t' \in \mathcal{Y}$, and measurable $U \subseteq [\underline{p},\overline{p}]$, it holds that

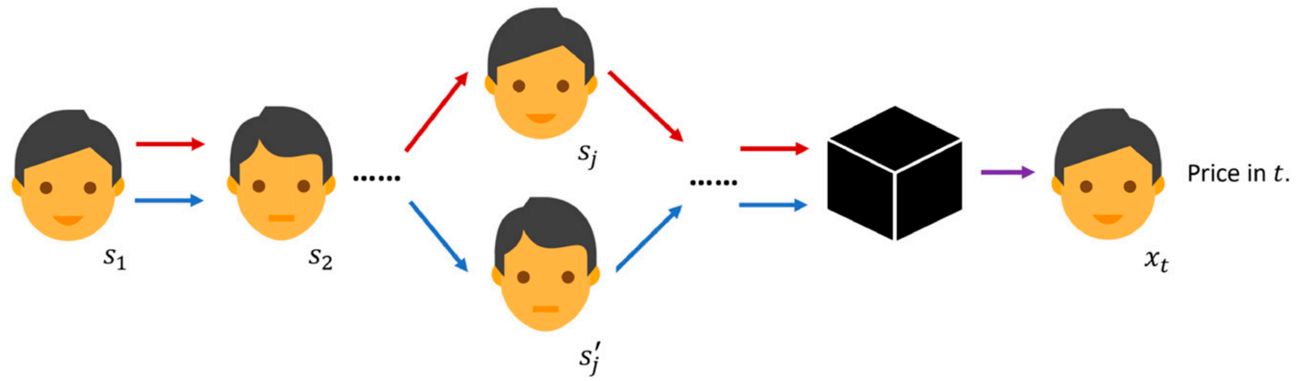
$$A_t(U|\boldsymbol{x}_t, s_1, \cdots, s_j, \cdots, s_{t-1}) \le e^{\varepsilon} A_t(U|\boldsymbol{x}_t, s_1, \cdots, s_j', \cdots, s_{t-1}).$$

Intuitively, Definition 2 requires the pricing policy π to be stable with respect to the sensitive information of a customer *prior to* the current time period, so that a malicious third party arriving at time t cannot reliably infer protected information of previously arrived customers. One key difference between Definitions 2 and 1 is the notable exclusion of s_t from the conditional set (i.e., we do not require the pricing policy to be stable with respect to $\boldsymbol{x}_t$), meaning that we assume the customer arriving at time t is either (1) a malicious third party who is therefore not interested in its own protected information or (2) a customer trusting that her sensitive information will *not* be released to later customers directly or indirectly through prices. We refer to Figure 2 as a graphic illustration of ε-CDP.

3.2. Local Differential Privacy

One fundamental assumption being made in the central differential privacy notions (including both Definitions 1 and 2) is that the customers trust the *platform* to protect their sensitive information and are only worried about malicious third parties pretending to be customers illegally extracting data indirectly through offered prices. This is clear from the fact that the pricing policy A_t is conditioned on the true data $\{s_j = (\boldsymbol{x}_j, y_j, p_j) : j < t\}$ for all customers arriving before t. That is, the platform can use the actual data of its customers to make pricing decision in each time period. On the other hand, in local differential privacy formulations, it is essential to constrain the platform (i.e., its pricing algorithm/policy) so that the platform never stores the actual sensitive data of customers and

Figure 2. (Color online) ε-CDP



Note. The arrows above and below mean that two neighboring sequences of data (with the only difference in s_j and s_j') are not likely to be distinguished after privatization.

instead make pricing decisions based on anonymized information. More specifically, let $z_t \in \mathcal{Z}$ be the privatized/anonymized statistics the platform records at time t and abbreviate $\boldsymbol{z}_{<t} = (z_1, \cdots, z_{t-1})$ as the anonymized statistics of all customers arriving prior to time t. The firm's (locally private) dynamic personalized pricing strategy can be parameterized by two (sequences of) conditional distributions: the *statistics recorder* Q_t, and the pricing strategy A_t, as follows:

$$z_t \sim Q_t(\cdot | s_t, \boldsymbol{z}_{<t}); \tag{4}$$

$$p_t \sim A_t(\cdot | \boldsymbol{x}_t, \boldsymbol{z}_{<t}). \tag{5}$$

For notational convenience, in the rest of the paper we write $z_t = Q_t(s_t, \boldsymbol{z}_{<t})$ and $p_t = A_t(\boldsymbol{x}_t, \boldsymbol{z}_{<t})$ with the understanding that both Q_t and A_t are randomized functions/procedures. In nonprivate settings, one can simply let $z_t = s_t$ and then $p_t = A_t(\boldsymbol{x}_t, s_1, \cdots, s_{t-1})$ reduces to the standard personalized pricing policy. With (local) privacy constraints, however, the statistics $\{z_t\}$ being recorded are constrained to be privatized statistics (i.e., do not leak too much protected information $\{s_t\}$) and the pricing decisions p_t are forced to be made on the anonymized statistics $\{z_t\}$ instead of the sensitive data $\{s_t\}$. It is also clear from Equation (4) that z_t and s_j are independent conditioned on $s_t, \boldsymbol{z}_{<t}$, for all $j < t$. We refer to Figures 3 and 4 for graphic representations of Q_t and A_t, respectively.

The formal definition of local differential privacy, following the seminal works of Evfimievski et al. (2003) and Duchi et al. (2018) in the literature, is given here.

Definition 3 (Local Differential Privacy for Personalized Pricing). For any $\varepsilon > 0$, a personalized pricing policy $\pi = \{Q_t, A_t\}_{t=1}^T$ satisfies ε-local differential privacy (ε-LDP) if for every t, $\boldsymbol{z}_{<t}$ and $s_t \neq s_t'$, it holds for every measurable $Z_t \subseteq \mathcal{Z}$ that

$$Q_t(Z_t | s_t, \boldsymbol{z}_{<t}) \le e^{\varepsilon} \cdot Q_t(Z_t | s_t', \boldsymbol{z}_{<t}).$$

We provide some additional notes for the definition of LDP. First, LDP guarantees the privacy of customer's data by the statistics recorder Q_t instead of directly by A_t as in CDP. More specifically, by privatizing s_t through Q_t, the pricing policy A_t can only use privatized data for decision making. As a result, even if the adversary in time t is able to infer any z_j where $j < t$ or hack the whole data set of the platform, the jth customer's personal data are still protected as none of the true data s_j are stored in the system. Second, in LDP the platform also makes pricing decision A_t conditioned on customer's raw data $\boldsymbol{x}_t$ as in CDP. Again, this can happen when customer in t is the adversary, who will not hack his/her own data. In the case of a normal customer, statistics recorder Q_t guarantees that $\boldsymbol{x}_t$ is unlikely to be leaked to others; thus, customer t can still trust the platform to use his/her personal data.

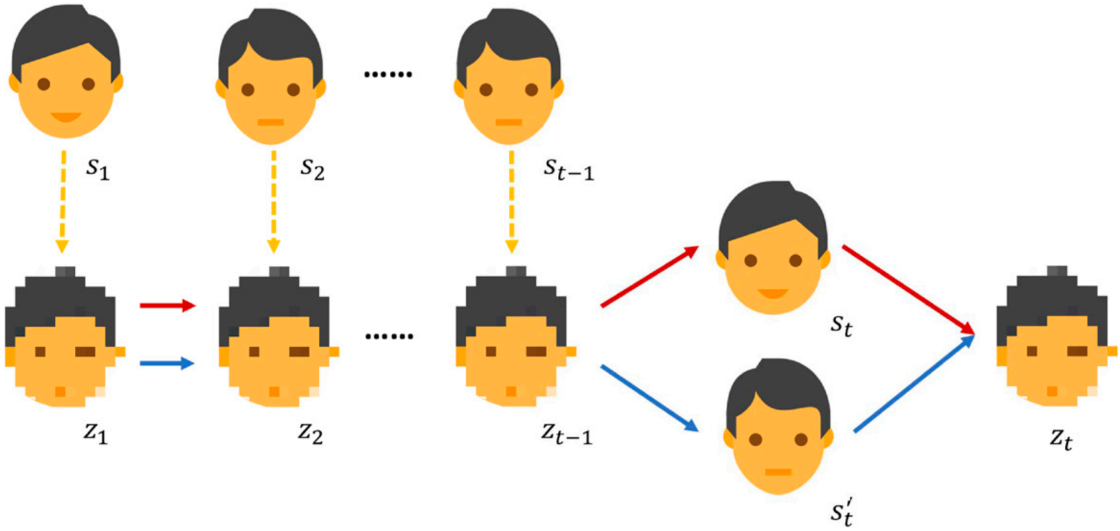
Although local and central differential privacy are not necessarily comparable, in the special case of the "noninteractive regime" where $\{Q_t\}$ are independent distributions, the following proposition shows that local differential privacy is stronger than its centralized counterpart.

Proposition 1. *Let $\pi = \{Q_t, A_t\}_{t=1}^T$ be a personalized pricing policy satisfying ε-LDP. Suppose also that $Q_t(\cdot)$ is independent of $\boldsymbol{z}_{<t}$ for all t. Then π satisfies ε-CDP.*

Proof of Proposition 1. This is true because

$$\begin{aligned}
&A_t(U | \boldsymbol{x}_t, s_1, \ldots, s_j, \ldots, s_{t-1}) \\
&= A_t(U | \boldsymbol{x}_t, z_1, \ldots, z_j, \ldots, z_{t-1}) \mathbb{P}(z_1, \ldots, z_j, \ldots, z_{t-1} | s_1, \ldots, \\
&\quad s_j, \ldots, s_{t-1}) \\
&= A_t(U | \boldsymbol{x}_t, z_1, \ldots, z_j, \ldots, z_{t-1}) \prod_{s \in [t] \setminus \{j\}} Q_s(z_s | s_s) Q_j(z_j | s_j) \\
&\le e^{\varepsilon} A_t(U | \boldsymbol{x}_t, z_1, \ldots, z_j, \ldots, z_{t-1}) \prod_{s \in [t] \setminus \{j\}} Q_s(z_s | s_s) Q_j(z_j | s_j') \\
&= e^{\varepsilon} A_t(U | \boldsymbol{x}_t, s_1, \ldots, s_j', \ldots, s_{t-1}),
\end{aligned}$$

Figure 3. (Color online) Statistics Recorder Q_t of ε-LDP



Note. The arrows above and below mean that privatizing different s_t and s'_t with the sequence of historical privatized data are not likely to be distinguished.

where the second equality is because $Q_t(\cdot)$ is independent of $z_{<t}$ for all t, and the inequality is from the definition of LDP, which states that $Q_j(z_j|s_j) \le e^{\varepsilon} Q_j(z_j|s'_j)$ for any $j < t$ and $s_j, s'_j \in \mathcal{S}$. □

4. Centralized-Private-Parallel-Quadrisection (CPPQ) Algorithm

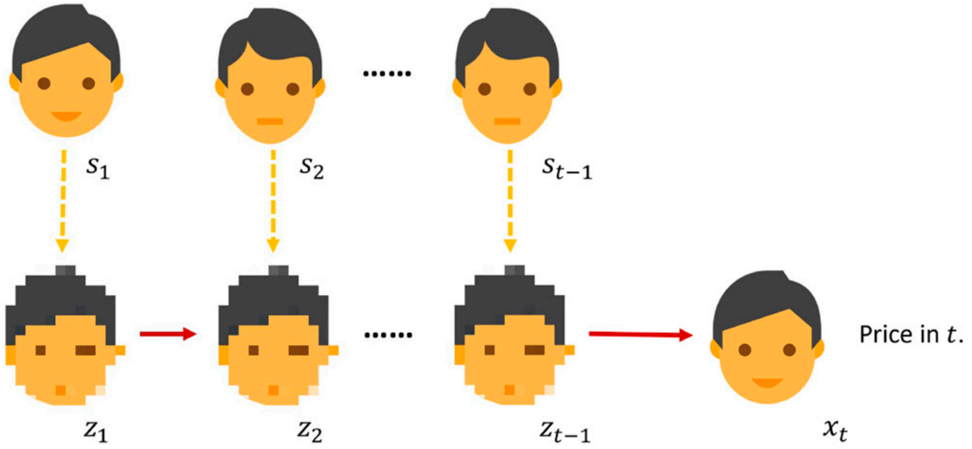
In this section, we describe a personalized pricing algorithm that satisfies the ε-CDP as defined in Definition 2. The proposed algorithm is named Centralized-Private-Parallel-Quadrisection (CPPQ) and its pseudocode description is given in Algorithm 1. There are several important techniques in CPPQ that will be used as building blocks for algorithms with ε-LDP (see Section 5).

Algorithm 1 (The CPPQ Algorithm)

```
**Input**: time horizon $T$, privacy parameter $\varepsilon$, algorithm parameters $J, c_1, c'_1, c_2$.
Initialize: partition $[0,1]^d$ into $J$ equally-sized hypercubes (each side's length being $h = J^{-1/d}$); for each hypercube $B_j$, $j \in [J]$, let $\boldsymbol{\rho}_j = (\underline{p}, \frac{1}{4}\underline{p} + \frac{3}{4}\overline{p}, \frac{1}{2}\underline{p} + \frac{1}{2}\overline{p}, \frac{3}{4}\underline{p} + \frac{1}{4}\overline{p}, \overline{p}) \in [\underline{p}, \overline{p}]^5$; $r_{j,k}(0) = \mu_{j,k}(0) = 0$ for $k \in \{1,2,3,4,5\}$; $\varsigma_j \leftarrow 0$;
**for** $t = 1, 2, \cdots, T$ **do**
  Receive $\boldsymbol{x}_t \in [0,1]^d$ and let $j_t \in [J]$ be the index such that $\boldsymbol{x}_t \in B_{j_t}$;
  Offer price $p_t = \rho_{j_t,k_t}$ where $k_t \equiv t \mod 5$, and receive $y_t \in [0,1]$;
  **for** $j = 1, 2, \cdots, J$ **do**
    Let $u_{t,j,k} = \mathbf{1}\{j_t = j \wedge k_t = k\} y_t p_t$ and $v_{t,j,k} = \mathbf{1}\{j_t = j \wedge k_t = k\}$ for $j \in [J]$ and $k \in [5]$;
    Update $r_{j,k_t}(t) \leftarrow$ TreeBasedAggregation $(\{u_{\tau,j,k_t}\}_{\tau<t}, t, u_{t,j,k_t}, \varepsilon/2, T)$;
    Update $\mu_{j,k_t}(t) \leftarrow$ TreeBasedAggregation $(\{v_{\tau,j,k_t}\}_{\tau<t}, t, v_{t,j,k_t}, \varepsilon/2, T)$;
    Let $r_{j,k}(t) \leftarrow r_{j,k}(t-1)$ and $\mu_{j,k}(t) \leftarrow \mu_{j,k}(t-1)$ for $k \ne k_t$;
    For $k \in [5]$ compute $\widehat{r}_{jk} = r_{j,k}(t) - r_{j,k}(\varsigma_j)$ and $\widehat{\mu}_{jk} = \mu_{j,k}(t) - \mu_{j,k}(\varsigma_j)$;
    Let $\underline{\mu}_{1\to3} = \min\{\widehat{\mu}_{j1}, \widehat{\mu}_{j2}, \widehat{\mu}_{j3}\}$ and $\underline{\mu}_{3\to5} = \min\{\widehat{\mu}_{j3}, \widehat{\mu}_{j4}, \widehat{\mu}_{j5}\}$;
    **if** $\underline{\mu}_{1\to3} \ge c_2$ and $\min\left\{\frac{\widehat{r}_{j3}}{\widehat{\mu}_{j3}} - \frac{\widehat{r}_{j2}}{\widehat{\mu}_{j2}}, \frac{\widehat{r}_{j2}}{\widehat{\mu}_{j2}} - \frac{\widehat{r}_{j1}}{\widehat{\mu}_{j1}}\right\} > \frac{3c_1}{\sqrt{\underline{\mu}_{1\to3}}} + \frac{3c'_1}{\underline{\mu}_{1\to3}}$ **then**
      $\boldsymbol{\rho}_j \leftarrow (\rho_{j2}, \frac{1}{4}\rho_{j2} + \frac{3}{4}\rho_{j5}, \frac{1}{2}\rho_{j2} + \frac{1}{2}\rho_{j5}, \frac{3}{4}\rho_{j2} + \frac{1}{4}\rho_{j5}, \rho_{j5})$, $\varsigma_j \leftarrow t$;
    **else if** $\underline{\mu}_{3\to5} \ge c_2$ and $\min\left\{\frac{\widehat{r}_{j3}}{\widehat{\mu}_{j3}} - \frac{\widehat{r}_{j4}}{\widehat{\mu}_{j4}}, \frac{\widehat{r}_{j4}}{\widehat{\mu}_{j4}} - \frac{\widehat{r}_{j5}}{\widehat{\mu}_{j5}}\right\} > \frac{3c_1}{\sqrt{\underline{\mu}_{3\to5}}} + \frac{3c'_1}{\underline{\mu}_{3\to5}}$ **then**
      $\boldsymbol{\rho}_j \leftarrow (\rho_{j1}, \frac{1}{4}\rho_{j1} + \frac{3}{4}\rho_{j4}, \frac{1}{2}\rho_{j1} + \frac{1}{2}\rho_{j4}, \frac{3}{4}\rho_{j1} + \frac{1}{4}\rho_{j4}, \rho_{j4})$, $\varsigma_j \leftarrow t$;
    **end if**
  **end for**
**end for**
```

We first explain the intuitions and design principles behind Algorithm 1 without data privacy considerations. Algorithm 1 uses two main ideas to carry out nonparametric personalized pricing with demand learning. The first idea is to partition the space of contextual vectors $\{\boldsymbol{x}_t\}_{t=1}^T \subseteq \mathcal{X} = [0,1]^d$ into J small hypercubes (denoted by B_j for $j \in [J]$) with equal volume. The algorithm then treats customers whose context vectors belonging to the same hypercube B_j the same. The effectiveness of this "localized" strategy is justified by the Lipschitz continuity of the expected reward function f (Assumption 2) and similar conditions in

Figure 4. (Color online) Pricing Strategy A_t of ε-LDP



Note. This graph shows that the platform only uses the privatized historical data (instead of the real historical data) for pricing.

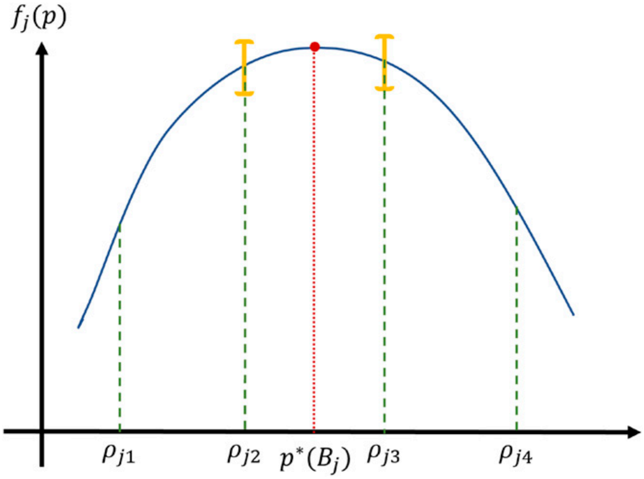
Assumption 3, implying that customers with similar context vectors have similar demand. Clearly, with more hypercubes (i.e., larger J) we are able to approximate customers' demands more accurately as hypercubes become smaller. On the other hand, we would suffer from the insufficiency of sample size for each hypercube as we are learning the localized demands of more hypercubes. An appropriate selection of the number of hypercubes (J) is vital for the good performance of the CPPQ policy, which we give more details in Theorem 1, where J is set to be $\lceil T^{d/(d+4)} \rceil$.

The second idea in Algorithm 1 is to use *quadrisection search* to find the optimal price $p^*(B_j)$ for all customers belonging to the same hypercube B_j. The quadrisection search procedure is justified by Assumption 3(a), which asserts that the expected reward of all customers belonging to hypercube B_j is *strongly concave* with respect to the offered price p. Although similar bisection or trisection methods have been adopted in the dynamic pricing literature for concave objectives (Lei et al. 2014, Wang et al. 2014), the previous work does not require dividing price intervals into four equally sized pieces. The reason we need to use a quadrisection search approach is because of the following: when dividing the price interval into three equally sized pieces, the (noisy) comparison of objective values between the two midpoints in this trisection search approach may be nonconclusive (Figure 5). Under normal circumstances, the price interval could still be updated and shrunk in case of nonconclusive comparison once a preallocated sample budget is consumed. However, when the pricing platform is subject to privacy constraints (especially the local privacy constraints), it is difficult for the algorithm to maintain accurate sample counts for each hypercube. On the other hand, by increasing the number of midpoints to explore, we can ensure that at least one sets of conditions in lines 13 and 15 of Algorithm 1 are automatically satisfied when a hypercube receives enough samples.

In the left panel of Figure 6, without noise, if we have $f_j(\rho_{j1}) \le f_j(\rho_{j2}) \le f_j(\rho_{j3})$, this must imply that $p^*(B_j) \ge \rho_{j2}$ by concavity of $f_j(\cdot)$ (i.e., Assumption 3(a)). Therefore, by shrinking the price range from $[\rho_{j1}, \rho_{j5}]$ to $[\rho_{j2}, \rho_{j5}]$, we still have $p^*(B_j)$ contained in the new price range. Similarly, the right panel of Figure 6 shows the case that $f_j(\rho_{j5}) \le f_j(\rho_{j4}) \le f_j(\rho_{j3})$ implies $p^*(B_j) \le \rho_{j4}$; thus, we can shrink $[\rho_{j1}, \rho_{j5}]$ to $[\rho_{j1}, \rho_{j4}]$ without losing $p^*(B_j)$. When there is noise (from both demand realization and privacy), we still have the same conclusion (with high probability) given the data samples of all prices are large enough.

In the work of Chen and Gallego (2021), an alternative multiarmed bandit formulation based on successive elimination strategies was adopted, which is asymptotically optimal under nonprivate settings. Because the

Figure 5. (Color online) Trisection Search



Note. In this case, we observe similar revenue of two midpoints; hence, it is difficult to decide whether it is better to shrink from left or right.

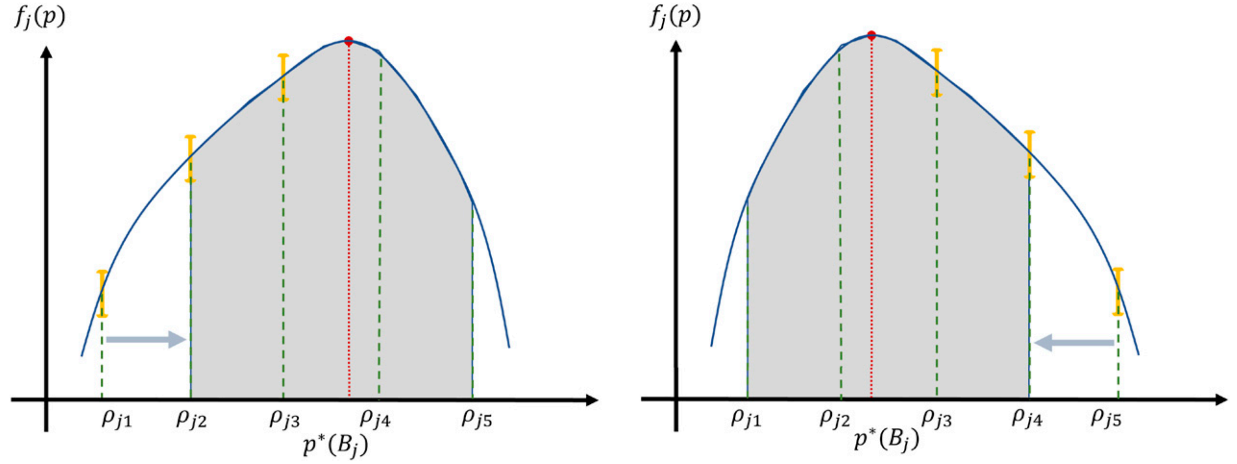
number of arms in the multiarmed bandit formulation scales with $\log T$, privatizing all arms will lead to considerably larger noise variation and subsequently higher regret.

It is worth explaining in more details of the construction of the $r_{j,k}(t)$ and $\mu_{j,k}(t)$ values in Algorithm 1 and how they can be used to guide the parallel quadrisection updates of $\boldsymbol{\rho}_j$ in different hypercubes B_j. To gain more intuitions, imagine for now that Algorithm 1 is no longer subject to any privacy constraints. In this ideal scenario, we have $\varepsilon \to \infty$ and as a result $r_{j,k}(t) = \sum_{\tau\le t} u_{\tau,j,k}$ and $\mu_{j,k}(t) = \sum_{\tau\le t} v_{\tau,j,k_t}$. Hence, $r_{j,k}(t)$ corresponds to the cumulative reward received for customers in hypercube B_j to whom price ρ_{jk} is offered, and $\mu_{j,k}(t)$ is the total number of customers in hypercube B_j to whom price ρ_{jk} is offered. Because $\boldsymbol{\rho}_j = (\rho_{jk})_{k=1}^r$ is constantly updated in Algorithm 1, the ς_j serves as a "pointer" for hypercube B_j, meaning that the time periods after ς_j correspond to rewards and customer counts for the current version of the $\boldsymbol{\rho}_j$ parameter. With $\widehat{r}_{jk} = r_{jk}(t) - r_{jk}(\varsigma_j)$ and $\widehat{\mu}_{jk} = \mu_{jk}(t) - \mu_{jk}(\varsigma_j)$, the ratio $\widehat{r}_{jk}/\widehat{\mu}_{jk}$ would then approximate $f_{B_j}(\rho_{jk})$ by the law of large numbers.

Finally, we explain how Algorithm 1 protects customers' data privacy in a centralized manner. The main idea is to calibrate artificial Laplace noise into the cumulative reward $r_{j,k}(t)$ and customer counts $\mu_{j,k}(t)$, so that the price and realized demand of a single customer will not affect much of the final reward/customer counts. It is worth noting that the hypercube *index* j_t also reveals sensitive information of $\boldsymbol{x}_t$. To prevent an adversary from identifying the hypercube j_t that $\boldsymbol{x}_t$ belongs to, we need to calibrate artificial noise into *all* hypercubes $j = 1,\ldots,J$ at each time period t, regardless of whether $\boldsymbol{x}_t$ belongs to a particular hypercube or not. In addition, to alleviate composition of central differential privacy, our proposed CPPQ algorithm uses a *tree-based aggregation* framework (Chan et al. 2011, Dwork et al. 2014). For completeness purposes, we provide the pseudo-code description of this aggregation framework in Algorithm 2. The main objective of this procedure is to release sequential private data with provable central privacy guarantees. More specifically, the $r_{j,k_t}(t)$ statistics constructed in Algorithm 2 is a (centralized) privatized version of the cumulative statistic $\sum_{\tau\le t} u_{\tau,j,k_t}$, and similarly $\mu_{j,k_t}(t)$ is a privatized version of the cumulative statistic $\sum_{\tau\le t} v_{\tau,j,k_t}$. We refer the readers to the works of Chan et al. (2011) and Dwork et al. (2014), as well as the recent work of Chen et al. (2022), for details and motivations of this procedure and its analysis. Because of the space limit, in this paper, we only give some high-level idea of this tree-based method for illustration.

Let us consider how we can privatize the sequence of the (true) customer counts $\{\overline{\mu}_{j,k}(t) := \sum_{s=1}^{t} v_{s,j,k} : t \in [T]\}$ with j, k fixed as an illustrative example. A general idea is based on the so-called *partial sums* (p-sums). In particular, define a generic p-sum as $\overline{\mu}_{j,k}([t_1, t_2]) := \sum_{s=t_1+1}^{t_2} v_{s,j,k}$; then each $\overline{\mu}_{j,k}(t)$ essentially can be decomposed into multiple p-sums, depending on how we define each of them. The way we privatize all $\overline{\mu}_{j,k}(t)$ is simply to add a Laplace noise $w_t \sim \mathrm{Lap}(O(1)/\varepsilon)$ to each p-sum. Based on this concept, there are two simple (yet ineffective) methods. The first method is to release each privatized $\overline{\mu}_{j,k}(t)$ by a single t-sum $\overline{\mu}_{j,k}([0,t]) + w_t$ (i.e., directly adding a noise to itself). Unfortunately, this method leads to $O(T\varepsilon)$-CDP because, for instance, a flip of the value $v_{1,j,k}$ will affect all t-sums $\overline{\mu}_{j,k}[0,t]$ with $t = 1,\ldots,T$. On the other hand, the second method represents the privatized $\overline{\mu}_{j,k}(t)$ by a summation of t 1-sums as $\sum_{s=1}^{t}(\overline{\mu}_{j,k}([s-1,s]) + w_s)$. Although it guarantees ε-CDP, this method accumulates too much noise from all w_s, which is roughly $O(\sqrt{t}/\varepsilon)$.

Figure 6. (Color online) Quadrisection Search



Note. The left/right panel satisfies condition in line 13/line 15 of Algorithm 1.

As a result, we need to find a method which is in between these two extreme cases, and this can be achieved by the tree-based aggregation. More specifically, tree-based aggregation represents each privatized $\overline{\mu}_{j,k}(t)$ by $O(\log_2 t)$ number of p-sums. To do that, we represent t by its binary expression as $t = \sum_{\ell=0}^{\lfloor \log_2 T \rfloor} b_\ell(t)2^\ell$ with $b_\ell(t) \in \{0, 1\}$. For instance, when $t = 10$, its binary form is $1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0$. Then the privatized $\overline{\mu}_{j,k}(10)$ is equal to $(\overline{\mu}_{j,k}([0, 2^3]) + w_1) + (\overline{\mu}_{j,k}([2^3, 2^3 + 2^1]) + w_2)$. By this tree-based aggregation method, we achieve $O(\log T\varepsilon)$-CDP and the cumulative error is at most $O((\log T)^{1.5}/\varepsilon)$. Because the factor on both privacy and error is only logarithmic in T, we are able to achieve the optimal rate by adjusting ε accordingly.

Algorithm 2 (Tree-Based Aggregation Procedure; Chan et al. 2011, Dwork et al. 2014)

```
function TreeBasedAggregation($\{u_\tau\}_{\tau<t}, t, s_t, \varepsilon, T$)
    (Initialization: $\alpha_\ell = \widehat{\alpha}_\ell = 0$ for $\ell = 0, 1, \cdots, \lfloor \log_2 T \rfloor$ when initialized;)
    Let $\{\alpha_\ell, \widehat{\alpha}_\ell\}_{\ell=0}^{L}$ be associated with $\{u_\tau\}_{\tau<t}$, where $L = \lfloor \log_2 T \rfloor$ and $\varepsilon' = \varepsilon/(L+1)$;
    Let $t = \sum_{\ell=0}^{L} b_\ell(t)2^\ell$ with $b_\ell(t) \in \{0, 1\}$ be the binary expression of $t$;
    $\ell_{\min}(t) := \min\{\ell : b_\ell(t) = 1\}$;
    Update $\alpha_{\ell_{\min}(t)} \leftarrow \sum_{\ell<\ell_{\min}(t)} \alpha_\ell + u_t$ and $\alpha_\ell = \widehat{\alpha}_\ell = 0$ for all $\ell < \ell_{\min}(t)$;
    Calibrate noise $\widehat{\alpha}_{\ell_{\min}}(t) \leftarrow \widehat{\alpha}_{\ell_{\min}}(t) + w_t$, where $w_t \sim \mathrm{Lap}(2/\varepsilon')$;
    return $\widehat{U}(t) = \sum_{\ell=0}^{L} b_\ell(t)\widehat{\alpha}_\ell(t)$;
end function
```

Our next theorem proves that the proposed CPPQ policy satisfies ε-CDP and establishes an upper bound on the regret incurred by the algorithm.

Theorem 1. *The CPPQ policy described in Algorithm 1 satisfies ε-CDP as defined in Definition 2. Furthermore, if Algorithm 1 is executed with $J = \lceil T^{d/(d+4)} \rceil$, $c_1 = \sqrt{\ln(2T^3)}$, $c_1' = 4c_2$ and $c_2 = 76\varepsilon^{-1}\ln^2(2T^3)$, then the regret of Algorithm 1 can be upper bounded by*

$$\mathbb{E}\left[\sum_{t=1}^{T} f(p^*(\boldsymbol{x}_t), \boldsymbol{x}_t) - f(p_t, \boldsymbol{x}_t)\right] \le \overline{C}_1 \times T^{(d+2)/(d+4)} + \overline{C}_1' \times \varepsilon^{-1} T^{d/(d+4)} + O(1),$$

where $\overline{C}_1, \overline{C}_1'$ are constants satisfying $\overline{C}_1 \le \mathrm{const.} \times C_H^2 (\sigma_H^{-4} + C_H\sqrt{C_X})\ln^2(2C_H^2T^3) + C_H^2C_p^2 d/2$ and $\overline{C}_1' \le \mathrm{const.} \times C_H^2\sigma_H^{-2}\ln^3(2C_H^2T^3)$, where const. *are numerical constants that do not depend on any problem parameters.*

Remark 1. The ε-CDP privacy guarantee of Algorithm 1 holds for *any* values of algorithm parameters J, c_1, c_1', and c_2. This gives practitioners more flexibility in tuning numerical constants in these algorithm parameters for better empirical performances.

Remark 2. In this remark, we explain how to convert the CPPQ policy in Algorithm 1 into an anytime policy (i.e., without prior knowledge of time horizon T) with simple changes. Consider an infinite geometric sequence $\{T_\zeta = 2^\zeta\}$ with $\zeta = 1, 2, 3, \cdots$ and run Algorithm 1 repeatedly with $T = T_\zeta$, $\varepsilon_\zeta = 6\varepsilon/(\pi^2\zeta^2)$ and other problem parameters (J, c_1, c_1', c_2) set accordingly using T_ζ and ε_ζ. This revised any-time policy satisfies ε-CDP with $\varepsilon = \sum_{\zeta=1}^{\infty} \varepsilon_\zeta = \sum_{\zeta=1}^{\infty} 6\varepsilon/(\pi^2\zeta^2) \le \varepsilon$ because of single composition of differentially private algorithms. To upper bound the cumulative regret of such an anytime algorithm, for a total of T time periods elapsed and $\zeta_0 = \lceil \log_2 T \rceil$ being the last "epoch," Theorem 1 implies that the total regret is upper bounded by $\sum_{\zeta=1}^{\zeta_0} \overline{C}_1 \times T_\zeta^{(d+2)/(d+4)} + \overline{C}_1' \times \varepsilon_\zeta^{-1}\ T_\zeta^{d/(d+4)} + O(1) \le \overline{C}_1 \times T^{(d+2)/(d+4)} \log_2 T + \overline{C}_1' \times 2\varepsilon^{-1} T^{d/(d+4)} \log_2^2 T + O(\log_2 T)$.

Because the primary focus of the paper lies on the more practical local privacy setting, we relegate the proof of Theorem 1 to the online appendix. It is, however, interesting to discuss the regret upper bound obtained in Theorem 1 and contrast it with existing results of Chen and Gallego (2021) under nonprivacy settings. The dominating term (as $T \to \infty$) in Theorem 1 is $\widetilde{O}(T^{(d+2)/(d+4)})$ with the coefficient $\overline{C}_1$ being independent from the privacy parameter ε. This term matches the upper and lower bound in Chen and Gallego (2021). The cost of performance arising from protecting customers' data privacy is reflected in the $\widetilde{O}(\varepsilon^{-1}T^{d/(d+4)})$ term, with smaller ε corresponding to stronger privacy guarantees and therefore larger regret. However, for this term, the $T^{d/(d+4)}$ regret will be asymptotically dominated by the $T^{(d+2)/(d+4)}$ regret in the other term, showing that the impact of privacy constraints will diminish as more customers and their data are available to the platform. This is a unique feature of the central privacy regime for which the platform has centralized control over the release of sensitive information, which is also observed in the work of Chen et al. (2022) for parametric demand models (we will do a comparison with their algorithm in numerical experiments in Section 7). The situation will be completely different for locally private settings, as we shall see in the next section and Theorem 2.

5. Locally-Private-Parallel-Quadrisection (LPPQ) Algorithm

In this section, we describe a personalized pricing algorithm that satisfies ε-LDP. The proposed algorithm is named Locally-Private-Parallel-Quadrisection (LPPQ) and its pseudocode description is given in Algorithm 3.

Algorithm 3 (The LPPQ Algorithm)

```
Input: time horizon $T$, privacy parameter $\varepsilon$, algorithm parameters $J, \kappa_1, \kappa_2$.
```

```
2: Initialize: partition $[0,1]^d$ into $J$ equally sized hypercubes (each side's length being $h = J^{-1/d}$); for each hypercube $B_j$, $j \in [J]$, let $\boldsymbol{\rho}_j = (\underline{p}, \frac{1}{4}\underline{p} + \frac{3}{4}\overline{p}, \frac{1}{2}\underline{p} + \frac{1}{2}\overline{p}, \frac{3}{4}\underline{p} + \frac{1}{4}\overline{p}, \overline{p}) \in [\underline{p}, \overline{p}]^5$, $\boldsymbol{r}_j(0) = (0,0,0,0,0)$, $\varsigma_j = 0$;
3: for $t = 1, 2, \cdots, T$ do
4:     Receive $\boldsymbol{x}_t \in [0,1]^d$ and let $j_t \in [J]$ be an integer such that $\boldsymbol{x}_t \in B_{j_t}$;
5:     Offer price $p_t = \rho_{j_t k_t}$ where $k_t \equiv t \mod 5$, and receive $y_t \in [0, 1]$;
6:     for $j = 1, 2, \cdots, J$ do
7:         $r_{j,k_t}(t) \leftarrow r_{j,k_t}(t-1) + \mathbf{1}\{j = j_t\} p_t y_t + w_{j,t}$, $w_{j,t} \overset{i.i.d}{\sim} \mathrm{Lap}(2/\varepsilon)$;
8:         $r_{j,k}(t) \leftarrow r_{j,k}(t-1)$ for $k \neq k_t$;
9:         For $k \in [5]$ compute $\widehat{r}_{jk} = r_{j,k}(t) - r_{j,k}(\varsigma_j)$; let $n_j \leftarrow t - \varsigma_j$;
10:        if $n_j \geq \kappa_2$ and $\min\{\widehat{r}_{j2} - \widehat{r}_{j1}, \widehat{r}_{j3} - \widehat{r}_{j2}\}/(5h^d n_j) > 3\kappa_1/(\varepsilon h^d \sqrt{n_j})$ then
11:            $\boldsymbol{\rho}_j \leftarrow (\rho_{j2}, \frac{1}{4}\rho_{j2} + \frac{3}{4}\rho_{j5}, \frac{1}{2}\rho_{j2} + \frac{1}{2}\rho_{j5}, \frac{3}{4}\rho_{j2} + \frac{1}{4}\rho_{j5}, \rho_{j5})$, $\varsigma_j \leftarrow t$;
12:        else if $n_j \geq \kappa_2$ and $\min\{\widehat{r}_{j3} - \widehat{r}_{j4}, \widehat{r}_{j4} - \widehat{r}_{j5}\}/(5h^d n_j) > 3\kappa_1/(\varepsilon h^d \sqrt{n_j})$ then
13:            $\boldsymbol{\rho}_j \leftarrow (\rho_{j1}, \frac{1}{4}\rho_{j1} + \frac{3}{4}\rho_{j4}, \frac{1}{2}\rho_{j1} + \frac{1}{2}\rho_{j4}, \frac{3}{4}\rho_{j1} + \frac{1}{4}\rho_{j4}, \rho_{j4})$, $\varsigma_j \leftarrow t$;
14:        end if
15:    end for
16: end for
```

The main ideas and principles of the LPPQ algorithm are the same as the CPPQ algorithm presented in the previous section: the algorithm first partitions the contextual vector space $\mathcal{X} = [0,1]^d$ into J equally sized small hypercubes[4] and then uses quadrisection search to localize the optimal price $p^*(B_j)$ (see (3)) for all customers whose contextual vectors belong to hypercube B_j. The major difference between CPPQ and LPPQ lies in how the privatized statistics $\widehat{r}_{jk}, \widehat{\mu}_{jk}$ are constructed, as central and local privacy impose different constraints on the platform's side. Here we explain the difference in details.

1. In the central privacy setting, the per-period instantaneous statistics $u_{j,t,k} = \mathbf{1}\{\boldsymbol{x}_t \in B_j \wedge k_t = k\} p_t y_t$ and $v_{j,t,k} = \mathbf{1}\{\boldsymbol{x}_t \in B_j \wedge k_t = k\}$ involving sensitive data are in complete possession of the pricing platform. The platform calibrates noise whenever it needs to release statistics $r_{j,k}(t)$ or $\mu_{j,k}(t)$ for pricing or model update purposes. In this way, the pricing platform has full knowledge of each customer's sensitive data, but third-party malicious agents would not be able to recover these sensitive data through interactions with the platform. It can also be shown (see Lemma EC.2 in the online appendix) that the magnitude of noise calibrated into $r_{j,k}(t)$ and $\mu_{j,k}(t)$ is on the order of $O(\varepsilon^{-1}\ln^2 T)$, a relatively small level because the cumulative statistics $r_{j,k}(t)$ and $\mu_{j,k}(t)$ could be as large as n_j.

On the other hand, in the local privacy setting, the per-period instantaneous statistics $\mathbf{1}\{\boldsymbol{x}_t \in B_j \wedge k_t = k\} p_t y_t$ is directly privatized by a Laplace noise and then communicated to the pricing platform. In this way, the platform keeps no copy of any customer's sensitive data, and the system/protocol is therefore more secure compared with CPPQ. The downside is that the noise calibrated to the privatized statistics is on the order of $\widetilde{O}(\varepsilon^{-1}\sqrt{n_j})$ (see Lemma 1), significantly larger than $O(\varepsilon^{-1}\ln^2 T)$ achievable for central privatizing mechanisms.

2. Because the calibrated noise magnitude is too large in local privacy settings, it is no longer feasible to keep track of the customer count statistics $\widehat{\mu}_{jk}$ like CPPQ does as the signals in the customer counts are too weak. Instead, in the LPPQ algorithm, we directly use $n_j = t - \varsigma_j$ (i.e., the total number of time periods after the previous pointer ς_j) to construct confidence intervals. This difference is more explicit if one compares the conditions of lines 13 and 15 in Algorithm 1 with those in Algorithm 3, as the confidence intervals in the former conditions are constructed using privatized customer counts $\widehat{\mu}_{jk}$, whereas the confidence intervals in the conditions of lines 10 and 12 in Algorithm 3 are built directly using the total time period counts n_j.

As local privacy is the main focus of this paper, we present detailed analysis of the privacy and regret performance guarantees of the proposed LPPQ policy in the next two sections. We also present an information theoretical lower bound in Section 6 that nicely complements our regret upper bound in Theorem 2.

5.1. Privacy Analysis

To see the LPPQ policy satisfies ε-LDP, we first explain how the policy can be parameterized as $\{Q_t, A_t\}_{t=1}^T$ as defined in Equations (4) and (5)). For each time period t, the intermediate variable $z_t \in \mathbb{R}^J$ is produced as $z_{tj} = \mathbf{1}\{j = j_t\} p_t y_t + w_{j,t}$, where $w_{j,t} \overset{i.i.d}{\sim} \mathrm{Lap}(2/\varepsilon)$. Clearly, the distribution of z_t is measurable conditioned on $s_t = (\boldsymbol{x}_t, y_t, p_t)$, satisfying Equation (4). With $\{\boldsymbol{z}_1, \cdots, \boldsymbol{z}_{t-1}\}$, the algorithm can compute the values of $\{\boldsymbol{\rho}_j, \boldsymbol{r}_j, \varsigma_j, n_j\}_{j=1}^J$ without accessing any of $s_1, \cdots, s_{t-1}$. The offered price p_t only depends on k_t, $\boldsymbol{x}_t$, and $\{\boldsymbol{\rho}_j\}_{j=1}^J$. Hence, A_t is measurable conditioned on $\boldsymbol{x}_t$ and $\boldsymbol{z}_{<t}$, satisfying Equation (5). The following proposition then follows immediately.

Proposition 2. *The LPPQ policy described in Algorithm 3 satisfies ε-LDP.*

Proof of Proposition 2. Fix arbitrary t. Because $y_t \in \mathcal{Y} \subseteq [0, 1]$ and $p_t \in [\underline{p}, \overline{p}] \subseteq [0, 1]$, the ℓ_1-sensitivity of $y_t p_t \boldsymbol{e}_{j_t} \in \mathbb{R}^J$ is upper bounded by two (i.e., $\sup_{s_t, s_t'} \|y_t p_t \boldsymbol{e}_{j_t} - y_t' p_t' \boldsymbol{e}_{j_t'}\|_1 \leq 2$). Hence, the ε-LDP of $z_{tj} = \mathbf{1}\{j = j_t\} p_t y_t + w_{j,t}$, $j \in [J]$, $w_{j,t} \overset{i.i.d.}{\sim} \mathrm{Lap}(2/\varepsilon)$ is guaranteed by the Laplace mechanism (Dwork and Roth 2014). □

5.2. Regret Analysis

The main objective of this section is to establish the following theorem upper bounding the cumulative regret of LPPQ when algorithm parameters are carefully chosen.

Theorem 2. *Suppose Assumptions 1–3 hold, and Algorithm 3 is executed with* $J = \lceil (\varepsilon\sqrt{T})^{d/(d+2)} \rceil$, $\kappa_1 = 1.7\sqrt{\ln(2T)}$, *and* $\kappa_2 = 31\ln T$. *Then the regret of Algorithm 3 can be upper bounded by*

$$\mathbb{E}\left[\sum_{t=1}^{T} f(p^*(x_t), x_t) - f(p_t, x_t)\right] \le \overline{C}_2 \times \varepsilon^{-2/(d+2)} T^{(d+1)/(d+2)},$$

where $\overline{C}_2 := \sigma_H^{-1}(160\kappa_1 + C_X\sqrt{\kappa_2})\sqrt{2\ln(2C_XC_LT)} + 0.5C_H^2 C_p^2 C_X d$.

Remark 3. The ε-LDP privacy guarantee of Algorithm 3 holds for *any* values of algorithm parameters J, κ_1, and κ_2. This gives practitioners more flexibility in tuning numerical constants in these algorithm parameters for better empirical performances.

Remark 4. In the setting where customers' personal data privacy is not of concerns, the work of Chen and Gallego (2021) achieves a cumulative regret upper bound of $\widetilde{O}(T^{(d+2)/(d+4)})$. In contrast, the regret upper bound in Theorem 2 is on the order of $\widetilde{O}(\varepsilon^{-2/(d+2)} T^{(d+1)/(d+2)})$, which is a polynomial factor worse even if the privacy parameter $1/\varepsilon$ is on a constant level. Such a performance guarantee, although seemingly undesirable, is the best one can achieve, as we show in Theorem 3 in the next section that no locally differentially private policy can achieve regret significantly smaller than $\widetilde{O}(\varepsilon^{-2/(d+2)} T^{(d+1)/(d+2)})$.

Remark 5. In this remark we explain how to convert the LPPQ policy in Algorithm 3 into an anytime policy (i.e., without prior knowledge of time horizon T) with simple changes. Consider an infinite geometric sequence $\{T_\zeta = 2^\zeta\}$ with $\zeta = 1, 2, 3, \cdots$ and run Algorithm 3 repeatedly with $T = T_\zeta$, ε, and other problem parameters (J, κ_1, κ_2) set accordingly using T_ζ and ε. This revised any-time policy satisfies ε-LDP because of the properties of the LDP definition. To upper bound the cumulative regret of such an any-time algorithm, for a total of T time periods elapsed and $\zeta_0 = \lceil \log_2 T \rceil$ being the last "epoch," Theorem 2 implies that the total regret is upper bounded by $\sum_{\zeta=1}^{\zeta_0} \overline{C}_2 \times \varepsilon^{-2/(d+2)} T_\zeta^{(d+1)/(d+2)} \le \overline{C}_2 \times 2\varepsilon^{-2/(d+2)} T^{(d+1)/(d+2)}$.

In the rest of this section, we prove Theorem 2. We first establish a technical lemma showing that the $\widehat{r}_{jk}$ values are faithful estimates of $f_{B_j}(\cdot)$ evaluated on price vectors $\boldsymbol{\rho}_j$.

Lemma 1. *For* $j \in [J]$, *define* $\overline{\chi}(B_j) := J \times \Pr_{x \sim P_X}[x \in B_j]$. *With probability* $1 - O(T^{-1})$ *the following holds uniformly for all* $t, j \in [J]$, *and* $k \in [5]$ *that satisfies* $n_j \ge \kappa_2$:

$$\left| \frac{\widehat{r}_{jk}}{5h^d n_j} - \overline{\chi}(B_j) f_{B_j}(\rho_{jk}) \right| \le \frac{\kappa_1}{\varepsilon h^d \sqrt{n_j}}.$$

Proof of Lemma 1. Fix j, k and a particular time period t. Note that $J = h^{-d}$. Without loss of generality assume the time periods are $t = k, 5+k, \cdots, t_j$ where t_j is the largest integer not exceeding n_j that is equivalent to k modulo 5, and $\widehat{r}_{jk} = \sum_t u_{j,t}$, where $u_{j,t} = \mathbf{1}\{j_t = j\} y_t p_t + w_{j,t}$, $w_{j,t} \overset{i.i.d.}{\sim} \mathrm{Lap}(2/\varepsilon)$. It then holds that $u_{j,t} = \mu_j + \xi_{j,t} + w_{j,t}$, where $\mu_j = \overline{\chi}(B_j) h^d f_{B_j}(\rho_{jk})$, $\mathbb{E}[\xi_{j,t}] = \mathbb{E}[w_{j,t}] = 0$ and $|\xi_{j,t}| \le 1$ almost surely. Furthermore, $\{\xi_{j,t}, w_{j,t}\}_t$ are independent random variables. By Hoeffding's inequality (Hoeffding 1963) and the fact that $\sum_t \xi_{j,t}$ is the sum of $n_j/5$ independently distributed centered random variables, we have with probability $1 - O(T^{-3})$ that

$$\left| \sum_t \xi_{j,t} \right| \le 1.2\sqrt{n_j \ln(2T)}. \tag{6}$$

For the $\sum_t w_{j,t}$ term, invoke Chan et al. (2011, lemma 2.8) for concentration inequalities of independently distributed Laplace random variables. We have that for all $n_j \ge 31\ln T$, with probability $1 - O(T^{-3})$ it holds that

$$\left| \sum_t w_{j,t} \right| \le 7\varepsilon^{-1}\sqrt{n_j \ln(2T)}. \tag{7}$$

Combining Equations (6) and (7), we have with probability $1 - O(T^{-3})$ that

$$\left| \frac{\widehat{r}_{jk}}{5h^d n_j} - \overline{\chi}(B_j) f_{B_j}(\rho_{jk}) \right| \le \frac{1}{5h^d n_j}\left[1.2\sqrt{n_j \ln(2T)} + \frac{7\sqrt{n_j \ln(2T)}}{\varepsilon} \right] \le \frac{1.7\sqrt{\ln(2T)}}{\varepsilon h^d \sqrt{n_j}}.$$

Lemma 1 is then proved by applying the union bound over all t and $j \in [J]$. □

With Lemma 1, the concavity of $f_{B_j}(\cdot)$ immediately yields the following corollary.

Corollary 1. *With probability* $1 - O(T^{-1})$ *it holds for all* $j \in [J]$ *and* t *that* $p^*(B_j) \in [\rho_{j1}, \rho_{j5}]$.

To introduce the next key technical lemma, we need to define some notations. For a hypercube B_j, $j \in [J]$, we partition the entire T selling periods into *epochs* denoted as $\tau = 1, 2, 3, \cdots$, with each epoch starting with a time period at which ς_j is reset (at the start of T time periods or as a result of the execution of line 10 or 12 in Algorithm 3), and ending when either line 10 or line 12 is executed again to reset the ς_j pointer. Let $\mathcal{T}_j(\tau)$ be the collection of time periods during epoch τ for hypercube j, and define $n_j(\tau) := |\mathcal{T}_j(\tau)|$. Note that

$\mathcal{T}_j(\tau)$ includes selling periods during which customers with $x_t \notin B_j$ arrive as well. The following technical lemma upper bounds the cumulative regret incurred by customers with $x_t \in B_j$ during epoch τ.

Lemma 2. *Fix hypercube B_j, $j \in [J]$, and let τ be an epoch. Then conditioned on the success event in Lemma 1, it holds that*

$$n_j(\tau) \geq \frac{\kappa_1^2}{\varepsilon^2 h^{2d} C_X^2 C_L^2 \delta_\tau^2}, \tag{8}$$

where $\delta_\tau = (\overline{p} - \underline{p}) \cdot (3/4)^{\tau-1}$. Furthermore,

$$\mathbb{E}\left[\sum_{t \in \mathcal{T}_j(\tau)} \mathbf{1}\{x_t \in B_j\}(f(p^*(x_t), x_t) - f(p_t, x_t))\right]$$
$$\leq \frac{64\kappa_1 + C_X\sqrt{\kappa_2}}{\sigma_H \varepsilon}\mathbb{E}\left[\sqrt{n_j(\tau)}\right] + \frac{1}{2}C_H^2 C_p^2 C_X d h^{d+2}\mathbb{E}[n_j(\tau)]. \tag{9}$$

Proof of Lemma 2. Decompose the difference $f(p^*(x_t), x_t) - f(p_t, x_t)$ as

$$\begin{aligned} f(p^*(x_t), x_t) - f(p_t, x_t) = {} & [f(p^*(x_t), x_t) - f(p^*(B_j), x_t)] \\ & + [f(p^*(B_j), x_t) - f(p_t, x_t)]. \end{aligned}$$

To upper bound the first term, invoke Assumption 3(a) with $B = \{x_t\}$ and Assumption 3, (b) and (c), with $B = B_j$. We have

$$\begin{aligned} f(p^*(x_t), x_t) - f(p^*(B_j), x_t) &\leq \frac{C_H^2}{2}|p^*(x_t) - p^*(B_j)|^2 \\ &\leq \frac{C_H^2 C_p^2}{2}\sup_{x, x' \in B_j}\|x - x'\|_2^2 \\ &\leq \frac{C_H^2 C_p^2 d}{2}h^2. \end{aligned}$$

Subsequently, the left-hand side of Equation (9) can be upper bounded by

$$\mathbb{E}\left[\sum_{t \in \mathcal{T}_j(\tau)} \mathbf{1}\{x_t \in B_j\}(f_{B_j}(p^*(B_j)) - f_{B_j}(p_t))\right] + \frac{C_H^2 C_p^2 d}{2}h^2 n_j(\tau)\Pr[x \in B_j]. \tag{10}$$

In epoch τ, define $\delta_\tau := \rho_{j5} - \rho_{j1} = (\overline{p} - \underline{p}) \cdot (3/4)^{\tau-1}$, where the equality is by update rule of price range. Because $p^*(B_j) \in [\rho_{j1}, \rho_{j5}]$ thanks to Corollary 1, and $f_{B_j}(\cdot)$ is σ_H-strongly concave because of Assumption 3(a) with $B = B_j$, we have that either $\min\{f_{B_j}(\rho_{j2}) - f_{B_j}(\rho_{j1}), f_{B_j}(\rho_{j3}) - f_{B_j}(\rho_{j2})\} \geq \frac{\sigma_H^2}{32}\delta_\tau^2$ (if $p^*(B_j) \geq \rho_{j3}$), or $\min\{f_{B_j}(\rho_{j3}) - f_{B_j}(\rho_{j4}), f_{B_j}(\rho_{j4}) - f_{B_j}(\rho_{j5})\} \geq \frac{\sigma_H^2}{32}\delta_\tau^2$ (if $p^*(B_j) \leq \rho_{j3}$). Without loss of generality assume $p^*(B_j) \geq \rho_{j3}$ and $\min\{f_{B_j}(\rho_{j2}) - f_{B_j}(\rho_{j1}), f_{B_j}(\rho_{j3}) - f_{B_j}(\rho_{j2})\} \geq \frac{\sigma_H^2}{32}\delta_\tau^2$. By Lemma 1, this implies that throughout the epoch τ,

$$\begin{aligned} 5 \times \frac{\kappa_1}{\varepsilon h^d \sqrt{n_j}} \geq \overline{\chi}(B_j)(f_{B_j}\min\{f_{B_j}(\rho_{j2}) - f_{B_j}(\rho_{j1}), \\ f_{B_j}(\rho_{j3}) - f_{B_j}(\rho_{j2})\} \geq \tfrac{\sigma_H^2}{32}\overline{\chi}(B_j)\delta_\tau^2, \end{aligned}$$

where $\overline{\chi}(B_j) = J \times \Pr[x \in B_j] \in [0, C_X]$, because of Assumption 1, and the first inequality is by Lemma 1 and the fact that in any time period of τ before its end, line 10 is not executed. Inverting the previous inequality and noting that $n_j(\tau) \geq \kappa_2$ almost surely, we have

$$n_j(\tau) \leq \max\left\{\kappa_2, \frac{25{,}600\kappa_1^2}{\sigma_H^2 \varepsilon^2 h^{2d}\overline{\chi}(B_j)^2 \delta_\tau^4}\right\}. \tag{11}$$

Again within epoch τ and recall the definition that $\delta_\tau = \rho_{j5} - \rho_{j1}$. Because $f(\cdot)$ is C_L-Lipschitz continuous thanks to Assumption 2, we have that

$$\max_k |f_{B_j}(\rho_{j,k+1}) - f_{B_j}(\rho_{jk})| \leq C_L \max_k |\rho_{j,k+1} - \rho_{jk}| \leq C_L \delta_\tau. \tag{12}$$

This implies that $n_j(\tau)$ must satisfy

$$\frac{\kappa_1}{\varepsilon h^d \sqrt{n_j(\tau)}} \leq \overline{\chi}(B_j) C_L \delta_\tau,$$

which yields $n_j(\tau) \geq \frac{\kappa_1^2}{\varepsilon^2 h^{2d}\overline{\chi}(B_j)^2 C_L^2 \delta_\tau^2} \geq \frac{\kappa_1^2}{\varepsilon^2 h^{2d} C_X^2 C_L^2 \delta_\tau^2}$. This completes the proof of Equation (8).

Additionally, because $p^*(B_j) \in [\rho_{j1}, \rho_{j5}]$, and $f_{B_j}(\cdot)$ is twice continuously differentiable with its second derivative bounded by C_H^2 and $f_{B_j}'(p^*(B_j)) = 0$ because $p^*(B_j)$ is an interior maximizer of $f_{B_j}(\cdot)$, we have that

$$\begin{aligned} f_{B_j}(p^*(B_j)) - f_{B_j}(p_t) &\leq \frac{C_H^2}{2}\delta_\tau^2 \\ &\leq \frac{160\kappa_1}{\sigma_H \varepsilon h^d \overline{\chi}(B_j)}\frac{1}{\sqrt{n_j(\tau)}} + \frac{C_H^2\sqrt{\kappa_2}}{2\sqrt{n_j(\tau)}}, \end{aligned} \tag{13}$$

where the last inequality holds by inverting Equation (11). Subsequently, Equation (10) can be upper bounded by the expectations of

$$\begin{aligned} &\Pr[x \in B_j] \times \left[\frac{160\kappa_1\sqrt{n_j(\tau)}}{\sigma_H \varepsilon h^d \overline{\chi}(B_j)} + \frac{1}{2}\sqrt{\kappa_2 n_j(\tau)}\right] \\ &\quad + \frac{C_H^2 C_p^2 d}{2}h^2 n_j(\tau)\Pr[x \in B_j] \\ &= h^d \overline{\chi}(B_j) \times \left[\frac{160\kappa_1\sqrt{n_j(\tau)}}{\sigma_H \varepsilon h^d \overline{\chi}(B_j)} + \frac{1}{2}\sqrt{\kappa_2 n_j(\tau)}\right] \\ &\quad + \frac{C_H^2 C_p^2 d}{2}h^2 n_j(\tau) \times h^d \overline{\chi}(B_j) \\ &\leq \frac{160\kappa_1 + C_X\sqrt{\kappa_2}}{\sigma_H \varepsilon}\sqrt{n_j(\tau)} + \frac{1}{2}C_H^2 C_p^2 C_X d h^{d+2} n_j(\tau). \end{aligned}$$

This completes the proof of Lemma 2. □

We are now ready to prove Theorem 2.

Proof of Theorem 2. The entire proof is conditioned on the success event of Lemma 1, with an extra $O(1)$ term in the upper bound of the regret because the failure probability is $O(T^{-1})$ and in the event of failure the cumulative regret of Algorithm 3 is at most $O(T)$.

For each $j\in[J]$, the result in Lemma 2 establishes that

$$\begin{aligned}\mathbb{E}&\left[\sum_{t=1}^{T}\mathbf{1}\{\boldsymbol{x}_t\in B_j\}(f(p^*(\boldsymbol{x}_t),\boldsymbol{x}_t)-f(p_t,\boldsymbol{x}_t))\right]\\ &\le\frac{160\kappa_1+C_X\sqrt{\kappa_2}}{\sigma_H\varepsilon}\mathbb{E}\left[\sum_{\tau}\sqrt{n_j(\tau)}\right]\\ &\quad+\frac{1}{2}C_H^2C_p^2C_Xdh^{d+2}\mathbb{E}\left[\sum_{\tau}n_j(\tau)\right]\\ &\le\frac{(160\kappa_1+C_X\sqrt{\kappa_2})\sqrt{2\ln(2C_XC_LT)}}{\sigma_H}\frac{\sqrt{T}}{\varepsilon}\\ &\quad+\frac{1}{2}C_H^2C_p^2C_Xd\times h^{d+2}T,\end{aligned}\qquad(14)$$

where the last inequality holds by invoking the Cauchy-Schwarz inequality and noting that $\sum_\tau n_j(\tau)\le T$ and that the total number of epochs for each B_j is upper bounded by $2\ln(2C_XC_LT)$, thanks to Equation (8) in Lemma 2. Define $C_1':=\sigma_H^{-1}(160\kappa_1+C_X\sqrt{\kappa_2})\sqrt{2\ln(2C_XC_LT)}$ and $C_2':=0.5C_H^2C_p^2C_Xd$. Summing both sides of Equation (14) over all hypercubes $j\in[J]$, and noting that $J=h^{-d}$, we have

$$\mathbb{E}\left[\sum_{t=1}^{T}f(p^*(\boldsymbol{x}_t),\boldsymbol{x}_t)-f(p_t,\boldsymbol{x}_t)\right]\le C_1'\frac{\sqrt{T}}{h^d\varepsilon}+C_2'h^2T.\qquad(15)$$

With $J=\lceil(\varepsilon\sqrt{T})^{d/(d+2)}\rceil$ and $h=J^{-1/d}\approx(\varepsilon\sqrt{T})^{-1/(d+2)}$, Equation (15) yields the results in Theorem 2, with $\overline{C}_2=C_1'+C_2'$. □

6. Lower Bound of Algorithms with ε-LDP

In this section, we establish the following lower bound, showing that the $\widetilde{O}(T^{(d+1)/(d+2)})$ regret obtained in Theorem 2 is minimax optimal when the LDP parameter ε is finite. More specifically, we will prove the following result.

Theorem 3. *Let $\pi=\{Q_t,A_t\}_{t=1}^T$ be any personalized pricing policy that satisfies ε-LDP for some $\varepsilon\in(0,1]$. Let P_X be the uniform distribution on $[0,1]^d$. Then there exists $\lambda(\cdot,\cdot)$ and its associated revenue function $f(\cdot,\cdot)$ satisfying* Assumptions 1–3 *with* $C_X=1$, $C_L=4$, $\sigma_H=\sqrt{2}$, $C_H=2$, and $C_p=1$, *such that*

$$\mathbb{E}^{\pi}\left[\sum_{t=1}^{T}f(p^*(\boldsymbol{x}_t),\boldsymbol{x}_t)-f(p_t,\boldsymbol{x}_t)\right]\ge\underline{C}\times\frac{\varepsilon^{-2/(d+2)}T^{(d+1)/(d+2)}}{d^{7/3}},$$

where $\underline{C}>0$ is a universal numerical constant.

To prove Theorem 3, we use similar construction of adversarial instances as in the work of Chen and Gallego (2021), but with different analytical tools such as the Assouad's method (Assouad 1983, Yu 1997) and strong data processing inequalities as consequences of local privacy constraints, as developed in the work of Duchi et al. (2018).

6.1. Construction of Adversarial Problem Instances

We adopt the same construction of adversarial problem instances as in the work of Chen and Gallego (2021). For readers who are not familiar with the construction, we recapture it here in this section for completeness purposes. Suppose $[0,1]^d$ is being partitioned into J equally sized hypercubes, each of length $h=J^{-1/d}$, with J being specified later in the proof. Let $\{B_j\}_{j=1}^J$ be the J hypercubes that partition $[0,1]^d$. For each vector $\boldsymbol{\nu}\in\{0,1\}^J$, define problem instance $P_{\boldsymbol{\nu}}$ associated with demand model $\lambda_{\boldsymbol{\nu}}$ as

$$\lambda_{\boldsymbol{\nu}}(p,\boldsymbol{x}):=\frac{2}{3}-\frac{p}{2}+\sum_{j=1}^{J}\nu_j\left(\frac{1}{3}-\frac{p}{2}\right)\mathfrak{d}(\boldsymbol{x},\partial B_j),\qquad(16)$$

where $\mathfrak{d}(\boldsymbol{x},\partial B_j):=\inf_{\boldsymbol{y}\in\partial B_j}\|\boldsymbol{x}-\boldsymbol{y}\|_2$. It is proved in Chen and Gallego (2021, proposition 2) that all $\lambda_{\boldsymbol{\nu}}$ and their associated revenue functions $f_{\boldsymbol{\nu}}(p,\boldsymbol{x})=p\lambda_{\boldsymbol{\nu}}(p,\boldsymbol{x})$ satisfy Assumptions 2 and 3 with $C_L=4$, $\sigma_H=\sqrt{2}$, $C_H=2$, and $C_p=1$.

The demands $\{y_t\}_{t=1}^T$ are stochastically realized as $\Pr_{\boldsymbol{\nu}}[y_t=1|p_t,\boldsymbol{x}_t]=\lambda_{\boldsymbol{\nu}}(p_t,\boldsymbol{x}_t)$ and $\Pr_{\boldsymbol{\nu}}[y_t=0|p_t,\boldsymbol{x}_t]=1-\lambda_{\boldsymbol{\nu}}(p_t,\boldsymbol{x}_t)$. It is easy to verify that $\mathbb{E}_{\boldsymbol{\nu}_t}[y_t|p_t,\boldsymbol{x}_t]=\lambda_{\boldsymbol{\nu}}(p_t,\boldsymbol{x}_t)$. With P_X being the uniform distribution on $[0,1]^d$ and $[\underline{p},\overline{p}]=1$, all constructed problem instances satisfy Assumption 1 with $C_X=1$, $[\underline{p},\overline{p}]=[0,1]$, and $\mathcal{Y}=\{0,1\}\subseteq[0,1]$.

Although the construction of the adversarial problem instances are the same with the work of Chen and Gallego (2021), the analysis of "distinguishability" between problem instances are significantly different from Chen and Gallego (2021). More specifically, when the personalized pricing policy is subject to ε-LDP constraints, we need much sharper upper bounds on the distinguishability between problem instances to derive the $\Omega(T^{(d+1)/(d+2)})$ minimax lower bound, which is larger by polynomial factors of T compared with the $\Omega(T^{(d+2)/(d+4)})$ regret lower bound proved in Chen and Gallego (2021). The sharper lower bound also requires us to use different choices of J compared with the arguments in Chen and Gallego (2021). More details of the analysis is given in subsequent sections.

6.2. Reduction to Classification and Assouad's Lemma

Recall the definitions that $s_t=(\boldsymbol{x}_t,y_t,p_t)\in\mathcal{S}=\mathcal{X}\times\mathcal{Y}\times[\underline{p},\overline{p}]$ and $z_1,z_2,\cdots,z_T\in\mathcal{Z}$ are intermediate quantities satisfying ε-LDP, as defined in Equations (4) and (5) and Definition 3. Our first technical lemma shows that, for the problem instances $\{P_{\boldsymbol{\nu}}\}_{\boldsymbol{\nu}\in\{0,1\}^J}$ constructed in the previous section, a personalized pricing policy with low worst-case regret over $\{P_{\boldsymbol{\nu}}\}_{\boldsymbol{\nu}\in\{0,1\}^J}$ must have prices hitting the right subset in different cubes.

Lemma 3. *Let $\pi = \{Q_t, A_t\}_{t=1}^T$ be a personalized pricing policy that satisfies ε-LDP. Define $\mathfrak{R}(\pi) := \sup_{\boldsymbol{\nu}\in\{0,1\}^J} \mathbb{E}^\pi_{\boldsymbol{\nu}}\left[\sum_{t=1}^T f_{\boldsymbol{\nu}}(p^*(\boldsymbol{x}_t),\boldsymbol{x}_t) - f_{\boldsymbol{\nu}}(p_t,\boldsymbol{x}_t)\right]$. Let also $\mathcal{S}_0 := \{p : p < \frac{2}{3} - \frac{\eta h}{12}\}$ and $\mathcal{S}_1 := \left\{p : p > \frac{2}{3} - \frac{\eta h}{12}\right\}$, where $\eta = (1-2^{-1/d})/2$. Then*

$$\mathfrak{R}(\pi) \geq \frac{\eta^2 h^2}{\underline{K}_1} \sum_{t=1}^T \frac{1}{2^d} \sum_{\boldsymbol{\nu}\in\{0,1\}^d} \frac{1}{2J} \sum_{j=1}^J \Pr\left[p_t \in \mathcal{S}_{\nu_j} | x_t \in B_j, \mathfrak{d}(x_t, B_j) \geq \eta h; P^\pi_{\boldsymbol{\nu}}\right], \tag{17}$$

where $h = J^{-1/d}$ and $\underline{K}_1 > 0$ is a universal numerical constant.

To derive a lower bound on the classification error of ψ, we shall use the celebrated *Assuard's method* in the mathematical statistics and information theory literature (Assouad 1983, Yu 1997). To state the method, we need some additional notations. For each $\boldsymbol{\nu} \in \{0,1\}^J$, let $M^\pi_{\boldsymbol{\nu}}$ be the distribution of the intermediate quantities $\{z_1, \cdots, z_T\}$ under model $f_{\boldsymbol{\nu}}$ and personalized pricing policy π. For each $j \in [J]$, define

$$M^\pi_{+j} := \frac{1}{2^{J-1}} \sum_{\boldsymbol{\nu}:\boldsymbol{\nu}_j=1} M^\pi_{\boldsymbol{\nu}}, \quad M^\pi_{-j} := \frac{1}{2^{J-1}} \sum_{\boldsymbol{\nu}:\boldsymbol{\nu}_j=0} M^\pi_{\boldsymbol{\nu}} \tag{18}$$

as the mixture distribution by fixing the jth bit of $\boldsymbol{\nu}$ to either one or zero. We also define, for any $t \in [T]$ and $\boldsymbol{x}_t \in [0,1]^d$, $W^\pi_{\boldsymbol{\nu},t}(\boldsymbol{x}_t)$, a conditional distribution over $p \in [0, 1]$ such that

$$W^\pi_{\boldsymbol{\nu},t}(\boldsymbol{x}_t)(p \in \mathcal{S}) := \Pr\left[p_t \in \mathcal{S} | \pi, \boldsymbol{x}_t; z_1, \cdots, z_{t-1} \sim M^\pi_{\boldsymbol{\nu}}\right]. \tag{19}$$

For any measurable $\mathcal{X} \subseteq [0,1]^d$, define

$$\overline{W}^\pi_{\boldsymbol{\nu},t}(\mathcal{X}) := \left[\int_{x\in\mathcal{X}} W^\pi_{\boldsymbol{\nu},t}(x)\mathrm{d}x\right] \Big/ \left[\int_{x\in\mathcal{X}} 1\mathrm{d}x\right]. \tag{20}$$

Notations similar to Equation (18) are also defined as

$$\overline{W}^\pi_{+j,t}(\mathcal{X})(p\in\mathcal{S}) := \Pr\left[p_t\in\mathcal{S}|\pi,\boldsymbol{x}_t\in\mathcal{X};z_1,\cdots,z_{t-1}\sim M^\pi_{+j}\right] = \frac{1}{2^{J-1}} \sum_{\boldsymbol{\nu}:\boldsymbol{\nu}_j=1} \overline{W}^\pi_{\boldsymbol{\nu},t}(\mathcal{X})(p\in\mathcal{S});$$

$$\overline{W}^\pi_{-j,t}(\mathcal{X})(p\in\mathcal{S}) := \Pr\left[p_t\in\mathcal{S}|\pi,\boldsymbol{x}_t\in\mathcal{X};z_1,\cdots,z_{t-1}\sim M^\pi_{-j}\right] = \frac{1}{2^{J-1}} \sum_{\boldsymbol{\nu}:\boldsymbol{\nu}_j=0} \overline{W}^\pi_{\boldsymbol{\nu},t}(\mathcal{X})(p\in\mathcal{S}).$$

For two distributions P, Q let $\|P-Q\|_{\mathsf{TV}} = \frac{1}{2}\int|\mathrm{d}P - \mathrm{d}Q|$ be the total variation distance between P and Q, and $D^{\mathsf{sy}}_{\mathsf{KL}}(P,Q) = D_{\mathsf{KL}}(P\|Q) + D_{\mathsf{KL}}(Q\|P) = \int(\mathrm{d}P - \mathrm{d}Q)\ln(\mathrm{d}P/\mathrm{d}Q)$ be the symmetric Kullback-Leibler (KL) divergence between P and Q. Let $H_j := \{\boldsymbol{x} \in B_j : \mathfrak{d}(\boldsymbol{x}, \partial B_j) \geq \eta h\}$, where $\eta = (1-2^{-1/d})/2$. Using a derivation that is similar to Assouad's lemma (lemma 1 and equation (29) of Duchi et al. 2018), the right-hand side of Equation (17) can be lower bounded as

$$\begin{aligned}
&\sum_{t=1}^T \frac{1}{2J} \sum_{j=1}^J \frac{1}{2^d} \sum_{\boldsymbol{\nu}\in\{0,1\}^d} \Pr\left[p_t \in \mathcal{S}_{\nu_j} | x_t \in B_j, \mathfrak{d}(x_t, B_j) \geq \eta h; P^\pi_{\boldsymbol{\nu}}\right] \\
&= \sum_{t=1}^T \frac{1}{2J} \sum_{j=1}^J \frac{1}{2^d} \sum_{\boldsymbol{\nu}\in\{0,1\}^d} \overline{W}^\pi_{\boldsymbol{\nu},t}(H_t)(p \in \mathcal{S}_{\nu_j}) \\
&= \sum_{t=1}^T \frac{1}{2J} \sum_{j=1}^J \left[\frac{1}{2}\overline{W}^\pi_{+j,t}(H_t)(p\in\mathcal{S}_1) + \frac{1}{2}\overline{W}^\pi_{-j,t}(H_t)(p\in\mathcal{S}_0)\right] \\
&\geq \sum_{t=1}^T \frac{1}{4J} \sum_{j=1}^J (1 - |\overline{W}^\pi_{+j,t}(H_t)(p\in\mathcal{S}_1) - \overline{W}^\pi_{-j,t}(H_t)(p\in\mathcal{S}_1)|) \\
&\geq \sum_{t=1}^T \frac{1}{4J} \sum_{j=1}^J (1 - \|\overline{W}^\pi_{+j,t}(H_t) - \overline{W}^\pi_{-j,t}(H_t)\|_{\mathsf{TV}}) \\
&\geq \sum_{t=1}^T \frac{1}{4J} \sum_{j=1}^J (1 - \|M^\pi_{+j} - M^\pi_{-j}\|_{\mathsf{TV}}), \qquad (21) \\
&\geq \frac{T}{4J} \sum_{j=1}^J \left(1 - \sqrt{\frac{1}{4} D^{\mathsf{sy}}_{\mathsf{KL}}(M^\pi_{+j}, M^\pi_{-j})}\right) \\
&\geq \frac{T}{4}\left(1 - \sqrt{\frac{1}{4J}\sum_{j=1}^J D^{\mathsf{sy}}_{\mathsf{KL}}(M^\pi_{+j}, M^\pi_{-j})}\right). \qquad (22)
\end{aligned}$$

Here, Equation (21) holds by the standard data processing inequality (the $\overline{W}^\pi_{\pm j,t}$ distributions are derived from $M^\pi_{\pm j}$); Equation (22) holds because of Pinsker's inequality (the first inequality) and Jensen's inequality applied to the concavity of the $f(x) = \sqrt{x}$ function (the second inequality).

The question of upper bounding the symmetric KL divergence between M^π_{+j} and M^π_{-j}, crucial to lower bounding the right-hand side of Equation (22), is addressed in the next section.

6.3. Strong Data Processing Inequality

The main objective of this section is to provide technical tools to upper bound $D^{\mathsf{sy}}_{\mathsf{KL}}(M^\pi_{+j}, M^\pi_{-j})$. We first define some notations. Let $\boldsymbol{z}_{<t} = (z_1, \cdots, z_{<t})$, and $M^\pi_{\pm j,<t}$ be the marginal distributions of $\boldsymbol{z}_{<t}$ under $P^\pi_{\pm j}$, where $P^\pi_{+j} = \frac{1}{2^{J-1}}\sum_{\boldsymbol{\nu}:\boldsymbol{\nu}_j=1} P^\pi_{\boldsymbol{\nu}}$ and $P^\pi_{-j} = \frac{1}{2^{J-1}}\sum_{\boldsymbol{\nu}:\boldsymbol{\nu}_j=0} P^\pi_{\boldsymbol{\nu}}$. Let $P^\pi_{\pm j,t}$ be the distribution of s_t, which is measurable conditioned on $\boldsymbol{z}_{<t}$ because the price p_t being offered by

A_t can only depend on $\boldsymbol{x}_t$ and $\boldsymbol{z}_{<t}$ (see Equation (5)). We establish the following lemma.

Lemma 4. *Let π be a personalized pricing policy that satisfies 2ε-LDP. Then*

$$\sum_{j=1}^{J} D_{\mathsf{KL}}^{\mathsf{sy}}(M_{+j}^{\pi}, M_{-j}^{\pi})$$
$$\le 2(e^{2\varepsilon}-1)^2 \sum_{t=1}^{T} \sup_{\|\gamma\|_\infty \le 1} \sum_{j=1}^{J} \int_{\mathcal{Z}^{t-1}} \left| \int_{\mathcal{S}} \gamma(s_t, \boldsymbol{z}_{<t})[\mathrm{d}P_{+j,t}^{\pi}(s_t|\boldsymbol{z}_{<t}) - \mathrm{d}P_{-j,t}^{\pi}(s_t|\boldsymbol{z}_{<t})] \right|^2 \mathrm{d}\overline{M}_{<t}^{\pi}(\boldsymbol{z}_{<t})$$
$$\le 8(e^{2\varepsilon}-1)^2 \sum_{t=1}^{T}\sum_{j=1}^{J} \mathbb{E}_{\boldsymbol{z}_{<t}\sim \overline{M}_{<t}^{\pi}(\boldsymbol{z}_{<t})}\left[\|P_{+j,t}^{\pi}(\cdot|\boldsymbol{z}_{<t}) - P_{-j,t}^{\pi}(\cdot|\boldsymbol{z}_{<t})\|_{\mathsf{TV}}^2 \right],$$

where $\overline{M}_{<t}^{\pi} = \frac{1}{2^J}\sum_{\boldsymbol{\nu}\in\{0,1\}^J} M_{\boldsymbol{\nu},<t}^{\pi}$, and $\gamma(s_t, \boldsymbol{z}_{<t})$ is any arbitrary function of $s_t, \boldsymbol{z}_{<t}$ with $|\gamma(s_t,\boldsymbol{z}_{<t})| \le 1$ for any $s_t, \boldsymbol{z}_{<t}$.

Lemma 4 is a version of "strong data processing inequality" (Anantharam et al. 2013) that is in principle similar to theorem 3 of Duchi et al. (2018), with one significant difference: for all the results in the work of Duchi et al. (2018), the sensitive data $\{s_t\}_{t=1}^{T}$ are distributed independently and identically with respect to an unknown distribution that does not depend on the estimator or algorithm used (i.e., the classical statistical estimation setting). In contrast, for our personalized pricing problem, the distribution of the sensitive information $\{s_t\}_{t=1}^{T}$ depends on both the underlying demand model $f_{\boldsymbol{\nu}}$ and the (locally private) pricing policy π, as shown in the $P_{+j,t}^{\pi}$ and $P_{-j,t}^{\pi}$ measures that are measurable conditioned on $\boldsymbol{z}_{<t}$. This leads to a more sophisticated upper bound on the symmetric KL divergence between M_{+j}^{π} and M_{-j}^{π} measures as shown in Lemma 4.

Note that $P_{\pm j,t}^{\pi}(\boldsymbol{x}_t, y_t, p_t|\boldsymbol{z}_{<t}) = \chi(\boldsymbol{x}_t)A_t(p_t|\ \boldsymbol{x}_t, \boldsymbol{z}_{<t})\Pr_{\pm j}(y_t|p_t, \boldsymbol{x}_t)$, where χ is the PDF of P_X being the uniform distribution on $[0,1]^d$, $A_t(\cdot|\boldsymbol{x}_t, \boldsymbol{z}_{<t})$ is the pricing distribution of π that is measurable conditioned on $\boldsymbol{x}_t$ and $\boldsymbol{z}_{<t}$ by its definition in Equation (5), and $\Pr_{+j}(y_t|p_t,\boldsymbol{x}_t) = \frac{1}{2^{J-1}}\sum_{\boldsymbol{\nu}:\boldsymbol{\nu}_j=1}\Pr(y_t|f_{\boldsymbol{\nu}}(p_t,\boldsymbol{x}_t))$ and $\Pr_{-j}(y_t|p_t,\boldsymbol{x}_t) = \frac{1}{2^{J-1}}\sum_{\boldsymbol{\nu}:\boldsymbol{\nu}_j=0}\Pr(y_t|f_{\boldsymbol{\nu}}(p_t,\boldsymbol{x}_t))$. We then have the following lemma upper bounding the total variation between $P_{\pm j,t}^{\pi}(\cdot|\boldsymbol{z}_{<t})$.

Lemma 5. *For every $j\in[J], t\in[T]$ and $\boldsymbol{z}_{<t}\in\mathcal{Z}^{t-1}$, it holds that*

$$\|P_{+j,t}^{\pi}(\cdot|\boldsymbol{z}_{<t}) - P_{-j,t}^{\pi}(\cdot|\boldsymbol{z}_{<t})\|_{\mathsf{TV}}^2 \le \frac{dh^2}{16J^2}\mathbb{E}_{\boldsymbol{x}\sim U(B_j)}\mathbb{E}_{p\sim A_t(\cdot|\boldsymbol{x},\boldsymbol{z}_{<t})}[(p-p_0)^2],$$

where $p_0 = 2/3$ and $U(B_j)$ is the uniform distribution on B_j.

Combining Lemmas 4 and 5, and noting that $\Pr[\boldsymbol{x}\in B_j] = \int_{B_j}\mathrm{d}P_X(\boldsymbol{x}) = 1/J$ for all $j\in[J]$, we arrive at the following corollary.

Corollary 2. *Let π be a personalized pricing policy that satisfies 2ε-LDP, and $p_0 = 2/3$. Then*

$$\sum_{j=1}^{J} D_{\mathsf{KL}}^{\mathsf{sy}}(M_{+j}^{\pi}, M_{-j}^{\pi}) \le \frac{dh^2(e^{2\varepsilon}-1)^2}{2J} \times \sum_{t=1}^{T}\mathbb{E}[(p-p_0)^2|p\sim A_t(\cdot|\boldsymbol{x},\boldsymbol{z}_{<t}), \boldsymbol{x}\sim P_X, \boldsymbol{z}_{<t}\sim\overline{M}_{<t}^{\pi}].$$

6.4. Completing the Proof of Theorem 3

We first establish the following technical lemma showing that, if a personalized pricing policy π has small regret then it must produce prices that are close to $p_0 = 2/3$.

Lemma 6. *Let π be a personalized pricing policy, $p_0 = 2/3$, and $\phi(x,\delta) := x\mathbf{1}\{|x|\ge\delta\}$ be the hard-thresholding operator. Then*

$$\mathfrak{R}(\pi) \ge \frac{1}{4}\sum_{t=1}^{T}\mathbb{E}[\phi(|p_t-p_0|^2, h^2)|p_t\sim A_t(\cdot|\boldsymbol{x}_t,\boldsymbol{z}_{<t}), \boldsymbol{x}_t\sim P_X, \boldsymbol{z}_{<t}\sim\overline{M}_{<t}^{\pi}].$$

We are now ready to prove Theorem 3.

Proof of Theorem 3. First, if $\mathfrak{R}(\pi)\ge\varepsilon^{-2/(d+2)}T^{(d+1)/(d+2)}$, we have already proved Theorem 3. Hence, in the rest of the proof, we assume that $\mathfrak{R}(\pi)\le\varepsilon^{-2/(d+2)}T^{(d+1)/(d+2)} =: R$. Note that $\phi(|p-p_0|^2, h^2)\ge(p-p_0)^2 - h^2$, and $e^{2\varepsilon}-1\le 8\varepsilon$ for all $\varepsilon\in(0,1]$. Also, $J = h^{-d}$ by definition. Corollary 2 and Lemma 6 together yield

$$\sum_{j=1}^{J} D_{\mathsf{KL}}^{\mathsf{sy}}(M_{+j}^{\pi}, M_{-j}^{\pi}) \le 128dh^{2+d}\varepsilon^2(h^2T + R). \tag{23}$$

Set hypercube size h as

$$h := (8^{-1}\varepsilon\sqrt{dT})^{-1/(d+2)}. \tag{24}$$

It is then easy to verify that $\frac{1}{4J}\sum_{j=1}^{J} D_{\mathsf{KL}}^{\mathsf{sy}}(M_{+j}^{\pi}, M_{-j}^{\pi})\le\frac{1}{2}$. Subsequently, Equation (22) and Lemma 3 yield

$$\mathfrak{R}(\pi)\ge\frac{\eta^2h^2T}{8\underline{K}_1} = \Omega\left(\frac{h^2T}{d^2}\right) = \Omega\left(\frac{\varepsilon^{-2/(d+2)}T^{(d+1)/(d+2)}}{d^{2+1/(d+2)}}\right) = \Omega\left(\frac{\varepsilon^{-2/(d+2)}T^{(d+1)/(d+2)}}{d^{7/3}}\right),$$

which proves the regret lower bound in Theorem 3. □

7. Numerical Experiments

In this section, we conduct some illustrative numerical experiments to show the effect of central and local differential privacy on the regret. Moreover, we will compare the performance of our CPPQ with a benchmark algorithm (for parametric demand function) introduced in Chen et al. (2022) to illustrate the impact of nonparametric demand.

7.1. Effect of Central and Local Differential Privacy

To do that, we assume the dimension $d = 2$ and the demand $y_t(p)$ is a linear demand, that is, $y_t(p) = \theta_0 + \theta_1 x_{t,1} + \theta_2 x_{t,2} + \theta_3 p + \nu_t$, where ν_t is an independent zero-mean noise. Because this example is for illustrative purpose, the value of $\boldsymbol{\theta}$ is taken as $(0.4, 0.6, 0.6, -0.2)$, $\nu_t \in [-0.1, 0.1]$ is a uniform distribution, and $p \in [0.5, 4.5]$. Customer's data $\boldsymbol{x}_t$ is taken uniformly from $[0, 1]^2$. For the input parameters of the two algorithms (CPPQ and LPPQ), we simply fix $c_1 = 0.001\sqrt{\ln(T)}$, $c_1' = 0.01c_2, c_2 = \varepsilon^{-1}\ln^2(T)$, and $\kappa_1 = 0.001\sqrt{\ln(T)}, \kappa_2 = 0.1\ln(T)$ for LPPQ. Both algorithms have 30 independent runs with $T \in \{500, 2,500, 12,500, 62,500\}$ and $\varepsilon \in \{0.01, 0.1, 1, 10\}$, and for each T and ε, the average is taken as the output.

Results of the two algorithms are summarized in Table 1 (for CPPQ), Table 2 (for LPPQ), and Figure 7. To compare the performance of the algorithms across different T and ε, we compute the percentage regret, which is defined as $\mathfrak{R}_T(f, \pi)/\sum_{t=1}^T f(p^*(\boldsymbol{x}_t), \boldsymbol{x}_t)$. For the nonprivate benchmark, we adopt the nonprivate version of CPPQ (i.e., without adding noise) because it estimates the revenue more directly. According to these results, we can see that larger ε in general leads to better regrets, which are closer to the one with no privacy. Moreover, it can be observed that the performance of CPPQ is quite sensitive to ε (especially when ε is relatively large). This is in line with Theorem 1 as the impact of ε on the regret is by the factor of ε^{-1}. For LPPQ, our observation is similar. That is, higher privacy leads to higher percentage regret. However, it shall be noted that the difference of percentage regret with respect to privacy parameter ε is not very significant. This observation is consistent with our theoretical results in Theorem 2, which shows that the dependency of regret on ε is $\varepsilon^{-2/(d+2)}$. Moreover, in Figure 8, we plot the log-log scale of cumulative regret $\mathfrak{R}_T(f, \pi)/\ln(T)$ (it is divided by $\ln(T)$ because the regret upper bound has $\ln(T)$ factor as shown in Theorem 2) of LPPQ with all values of ε. The slopes of the fitted lines are $\{0.79, 0.77, 0.77, 0.75\}$ with respect to $\varepsilon \in \{0.01, 0.1, 1, 10\}$, which are quite close to $(d+1)/(d+2) = 0.75$ when $d = 2$. Thus, our numerical results verify the regret upper bound $\widetilde{O}(\varepsilon^{-2/(d+2)}T^{(d+1)/(d+2)})$ (with respect to T) of LPPQ.

Table 1. Percentage Regret (%) for CPPQ

	T = 500	T = 2,500	T = 12,500	T = 62,500
Nonprivate	15.79	7.40	3.33	1.76
$\varepsilon = 10$	26.77	20.68	12.65	8.68
$\varepsilon = 1$	34.61	31.48	25.89	21.04
$\varepsilon = 0.1$	34.81	33.06	29.89	26.72
$\varepsilon = 0.01$	34.70	33.63	30.51	27.21

Table 2. Percentage Regret (%) for LPPQ

	T = 500	T = 2,500	T = 12,500	T = 62,500
Nonprivate	15.79	7.40	3.33	1.76
$\varepsilon = 10$	21.82	17.53	15.50	13.27
$\varepsilon = 1$	20.81	17.40	15.73	14.29
$\varepsilon = 0.1$	22.89	17.66	15.95	14.80
$\varepsilon = 0.01$	22.53	20.70	17.20	16.74

In the end, comparing the result of CPPQ and LPPQ, we see that CPPQ does not necessarily have better performance than LPPQ, even though ε-CDP is weaker than ε-LDP in many cases. One reason is that in CPPQ, the Laplace noise has a scale of $2(L+1)/\varepsilon$ as opposed to $2/\varepsilon$ in LPPQ. As a result, for small ε, $\text{Lap}(2(L+1)/\varepsilon)$ can be quite significant especially in a relatively short horizon. This result is actually not surprising from the theoretical performance of CPPQ (i.e., $\widetilde{O}(\varepsilon^{-1}T^{d/(d+4)})$ for the part with ε) versus LPPQ (i.e., $\widetilde{O}(\varepsilon^{-2/(d+2)}T^{(d+1)/(d+2)})$). That is, when ε is small and T is not very large, it is very likely $\varepsilon^{-1}T^{d/(d+4)} > \varepsilon^{-2/(d+2)}T^{(d+1)/(d+2)}$ (e.g., $\varepsilon < o(T^{-5/12})$ when $d = 2$).

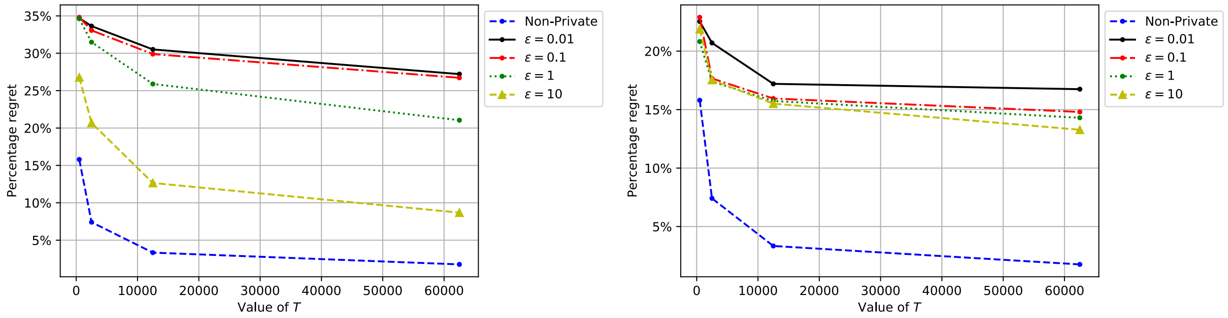
7.2. Comparison Between CPPQ and the Benchmark

As we mentioned in the Introduction, to the best of our knowledge, this paper is the first to consider both central and local differential privacies in dynamic pricing with *nonparametric* demand; thus, there is no direct benchmark in the literature to compare with. In this section, we select a benchmark algorithm (for convenience, named Benchmark) introduced in Chen et al. (2022), which is most related to ours in the literature. In particular, this benchmark solves a dynamic pricing problem with *parametric* demand (in particular, generalized linear model) under *central differential privacy*. Therefore, for fair comparison, we compare its performance with our algorithm CPPQ in two scenarios. The first scenario is the same as in our previous section, where the demand is a linear function and Benchmark correctly specifies its function format. In the second scenario, we assume the demand is a more sophisticated nonlinear function, but Benchmark misspecifies it as a linear function because model misspecification is quite prevalent in real application.

7.2.1. First Scenario with Linear Demand.

This scenario is exactly the same as in our previous experiment, and Benchmark correctly specifies the demand function as a linear function. For illustration, we demonstrate the comparisons between CPPQ and

Figure 7. (Color online) Performance of CPPQ and LPPQ

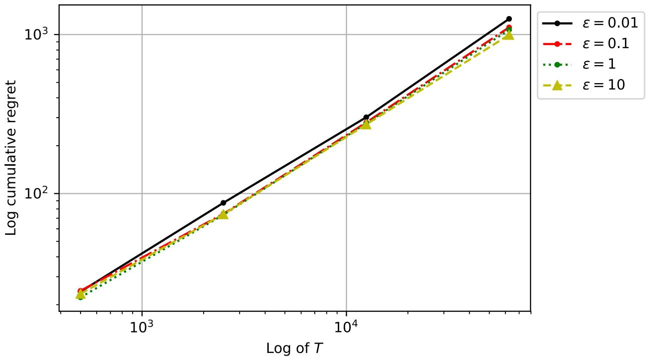


Notes. Percentage regret with respect to T of CPPQ (left) and LPPQ (right). In both cases, the percentage regrets converge to zero as T grows, and a larger ε (i.e., less privacy protection) leads to a smaller percentage regret.

Benchmark with $T \in \{500, 2, 500, 12, 500, 62, 500\}$ and $\varepsilon \in \{1, 10\}$ (and for Benchmark, achieving ε-CDP is the same as setting $\varepsilon_1 = \varepsilon_2 = \varepsilon$) and no privacy. Because Benchmark requires several learning parameters as input, we choose all of them the same as in numerical experiments in Chen et al. (2022) except $\rho = 1, \gamma = 3$, as we found they achieve better empirical performance for Benchmark.

Results are summarized in Figure 9. We can see that when there is no privacy, Benchmark significantly outperforms CPPQ. This is not surprising because Benchmark correctly specifies the linear demand and makes pricing decision based on this information, whereas CPPQ still treats the demand as a generic nonparametric function. However, when we have privacy guarantee, for example $\varepsilon = 10$, Benchmark only outperforms CPPQ when T is large; on the other hand, when $\varepsilon = 1$, CPPQ has better performance than Benchmark overall, although its advantage becomes smaller as T grows. This result is indeed consistent with the theoretical performance of both CPPQ and Benchmark. In particular, the regret upper bounds related to ε for CPPQ and Benchmark are $\widetilde{O}(\varepsilon^{-1}\ T^{d/(d+4)})$ and $\widetilde{O}(\varepsilon^{-2}d^2)$ (see theorem 2 in Chen et al. 2022 as our method of generating x_t satisfies their assumption 1), respectively. Therefore, CPPQ is less sensitive to ε, whereas Benchmark will have better performance when T is relatively large compared with ε^{-1}.

Figure 8. (Color online) Log-log Plot of $\mathfrak{R}_T(f, \pi)/\ln(T)$ for LPPQ for All ε and T. The Slopes of Fitted Lines with $\varepsilon \in \{0.01, 0.1, 1, 1\}$ are $\{0.79, 0.77, 0.77, 0.75\}$ Respectively



7.2.2. Second Scenario with Nonlinear Demand.

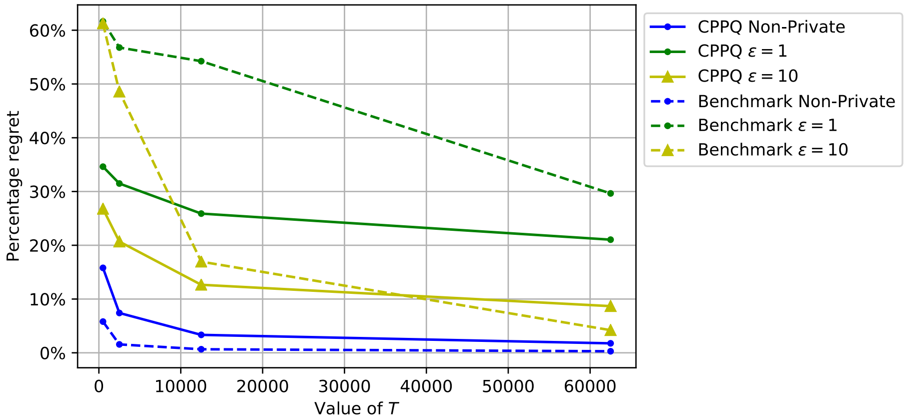
In this experiment, we assume a separable demand function (as described in Chen and Gallego 2021) defined as $y_t(p) = g_1(p)h_1(x_t) + g_2(p)h_2(x_t)$, where $g_1(p) := 1 - p^2$, $g_2(p) := 2.5 - 0.9\exp(p)$, $h_1(x_t) = \|x_t\|_2$, $h_2(x_t) = \|x_t\|_2^2$. With this nonlinear function, our Assumptions 1–3 are satisfied. For Benchmark, it still assumes the demand is linear like in the first scenario; hence, we want to test the effect of model misspecification of Benchmark (whereas our algorithm CPPQ does not assume any function format of the demand). For learning parameters of both algorithms, we still use the same ones in the previous experiment, but this time, we let the range of price be $[0, 1]$.

Figure 10 demonstrates the performance of CPPQ and Benchmark. We can see that CPPQ outperforms Benchmark in all scenarios as expected. Because of the model misspecification, even without privacy guarantee, the percentage regret of Benchmark is constantly greater than 8%.

8. Conclusion

This paper studies the online personalized pricing problem with nonparametric demand and data privacy (to the best of our knowledge, our paper is the first result on this problem). That is, over a finite time horizon, the platform decides a price for each arriving customer based on her personal data to maximize the cumulative revenue and protect customer's privacy. Two definitions of data privacy have been investigated: ε-CDP and ε-LDP, each of which depends on a parameter $\varepsilon > 0$ (i.e., smaller ε means higher security). Two algorithms are developed in this paper: CPPQ for ε-CDP and LPPQ for ε-LDP. Both algorithms are based on the idea of splitting the domain of customer's data into hypercubes

Figure 9. (Color online) Percentage Regret of CPPQ (Solid Lines) vs. Benchmark (Dashed Lines) under Different CDP Levels with Linear Demand

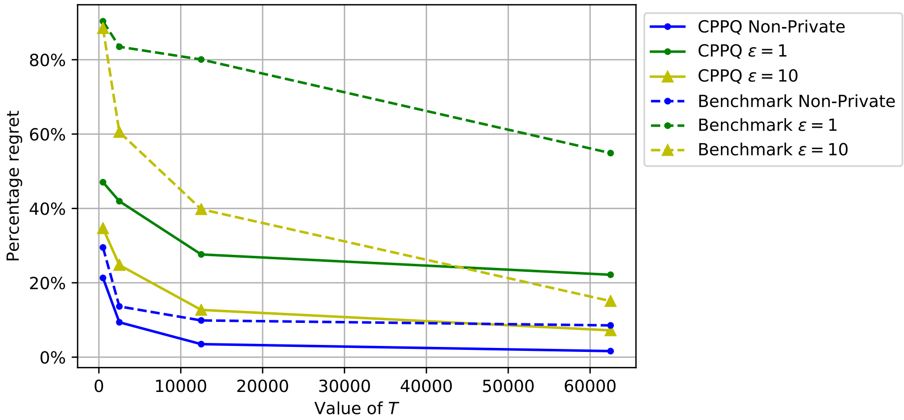


and applying a novel quadrisection search of optimal price in each hypercube. To satisfy ε-CDP, CPPQ uses a tree-based aggregation method to estimate the reward of the tested prices in each hypercube. Results show that this algorithm has regret at least $\widetilde{O}(T^{(d+2)/(d+4)} + \varepsilon^{-1}T^{d/(d+4)})$ (Theorem 1), where the first term matches the near-optimal regret of $\widetilde{O}(T^{(d+2)/(d+4)})$ without ε-CDP (Chen and Gallego 2021). On the other hand, LPPQ protects the privacy by adding noise to each data sample, and it achieves a regret of $\widetilde{O}(\varepsilon^{-2/(d+2)}\ T^{(d+1)/(d+2)})$ (Theorem 2). Moreover, this regret is proved to be near-optimal in Theorem 3, which shows that any algorithm satisfying ε-LDP has the regret at least $\Omega(\varepsilon^{-2/(d+2)}\ T^{(d+1)/(d+2)})$.

For potential future research, one direction is to consider data privacy protection with nonparametric model in other operations management problems (e.g., healthcare, assortment selection, ranking). Second, one may consider other techniques of protecting differential privacy in personalized pricing problem. For instance, the platform may add noise directly to historical customers' data $\boldsymbol{x}_t$, and a potential technique of demand learning in this scenario is the so-called nonparametric regression with error in variables (Fan and Truong 1993). Developing near-optimal online learning algorithms using this technique is an interesting open problem.

Another interesting direction for future research is to establish rigorous lower bounds of cumulative regret for dynamic personalized pricing policies satisfying ε-CDP constraints. Showing lower bound for CDP requirement is significantly more difficult compared with lower bounds for LDP algorithms. As far as we know, information-theoretical tools such as the ones established in (Duchi et al. 2013, 2018) do not exist for CDP, and virtually all existing lower bounds on CDP estimators rely on ad hoc techniques such as the construction of "tracing attack" mechanisms that reliably identify certain datum with the help of a very accurate summary statistics (Dwork et al. 2015; Cai et al. 2020, 2021). As a result, existing lower bounds on CDP algorithms are only established for *parametric*

Figure 10. (Color online) Percentage Regret of CPPQ (Solid Lines) vs. Benchmark (Dashed Lines) Under Different CDP Levels with Nonlinear Demand



models with *independently and identically distributed* samples, and relaxation of either parametric or i.i.d. constraints is significantly more challenging because of the delicate nature of the lower bound proof techniques.

Acknowledgments

The authors thank the area editor, Prof. Gustavo Vulcano, for handling the submission and the associate editor and three anonymous referees for valuable suggestions to improve our paper.

Endnotes

[1] See https://www.oecd.org/competition/personalised-pricing-in-the-digital-era.htm.

[2] Proposition 1 shows that LDP is a stronger data privacy notion compared with CDP under certain assumptions.

[3] The anticipating privacy notion defined here is slightly weaker than the definitions in Shariff and Sheffet (2018) and Chen et al. (2022), which considered joint distributions of future prices. Nevertheless, we adopt this definition here to be more compatible with existing definitions of local privacy notations, which is also appropriate for practical usage.

[4] The choice of J is, however, different from the CPPQ algorithm. See Theorems 2 and 1 for more details.

References

Acquisti A, Taylor C, Wagman L (2016) The economics of privacy. *J. Econom. Literature* 54(2):442–492.

Anantharam V, Gohari A, Kamath S, Nair C (2013). On maximal correlation, hypercontractivity, and the data processing inequality studied by Erkip and Cover. Preprint, submitted April 22, https://arxiv.org/abs/1304.6133.

Apple (2019) Improving Siri's privacy protections. Accessed April 9, 2022, https://www.apple.com/ca/newsroom/2019/08/improving-siris-privacy-protections/.

Araman VF, Caldentey R (2009) Dynamic pricing for nonperishable products with demand learning. *Oper. Res.* 57(5):1169–1188.

Arumugam S, Bhargavi R (2019) A survey on driving behavior analysis in usage based insurance using big data. *J. Big Data* 6(1):1–21.

Assouad P (1983) Deux remarques sur l'estimation. Comptes rendus des séances de l'Académie des sciences. Série 1. *Mathématique* 296(23):1021–1024.

Ban G-Y, Keskin NB (2021) Personalized dynamic pricing with machine learning: High-dimensional features and heterogeneous elasticity. *Management Sci.* 67(9):5549–5568.

Besbes O, Zeevi A (2009) Dynamic pricing without knowing the demand function: Risk bounds and near-optimal algorithms. *Oper. Res.* 57(6):1407–1420.

Besbes O, Zeevi A (2015) On the (surprising) sufficiency of linear models for dynamic pricing with demand learning. *Management Sci.* 61(4):723–739.

Bimpikis K, Morgenstern I, Saban D (2021). Data tracking under competition. Preprint, submitted February 10, https://dx.doi.org/10.2139/ssrn.3808228.

Birge JR, Chen H, Keskin NB (2021a). Markdown policies for demand learning with forward-looking customers. Preprint, submitted December 6, https://dx.doi.org/10.2139/ssrn.3299819.

Birge JR, Feng Y, Keskin NB, Schultz A (2021b) Dynamic learning and market making in spread betting markets with informed bettors. *Oper. Res.* 69(6):1746–1766.

Broder J, Rusmevichientong P (2012) Dynamic pricing under a general parametric choice model. *Oper. Res.* 60(4):965–980.

Cai TT, Wang Y, Zhang L (2020). The cost of privacy in generalized linear models: Algorithms and minimax lower bounds. Preprint, submitted December 6, https://arxiv.org/abs/2011.03900.

Cai TT, Wang Y, Zhang L (2021) The cost of privacy: Optimal rates of convergence for parameter estimation with differential privacy. *Ann. Statist.* 49(5):2825–2850.

Chan T-HH, Shi E, Song D (2011) Private and continual release of statistics. *ACM Trans. Inform. Systems Security* 14(3):1–24.

Chen N, Gallego G (2021) Nonparametric pricing analytics with customer covariates. *Oper. Res.* 69(3):974–984.

Chen Q, Jasin S, Duenyas I (2015) Real-time dynamic pricing with minimal and flexible price adjustment. *Management Sci.* 62(8): 2437–2455.

Chen X, Simchi-Levi D, Wang Y (2022) Privacy-preserving dynamic personalized pricing with demand learning. *Management Sci.* 68(7):4878–4898.

Cheung WC, Simchi-Levi D, Wang H (2017) Dynamic pricing and demand learning with limited price experimentation. *Oper. Res.* 65(6):1722–1731.

Cowen T, Tabarrok A (2015) *Modern Principles of Economics* (Macmillan International Higher Education).

De Blasio B, Menin J (2015) *From Cradle to Cane: The Cost of Being a Female Consumer. A Study of Gender Pricing in New York City* (The New York Department of Consumer Affairs, New York).

den Boer AV, Keskin NB (2020) Discontinuous demand functions: Estimation and pricing. *Management Sci.* 66(10):4516–4534.

den Boer AV, Zwart B (2013) Simultaneously learning and optimizing using controlled variance pricing. *Management Sci.* 60(3): 770–783.

Duchi JC, Jordan MI, Wainwright MJ (2013) Local privacy and statistical minimax rates. *Proc. IEEE Annual Sympos. on Foundations of Comput. Sci.* (IEEE, New York), 429–438.

Duchi JC, Jordan MI, Wainwright MJ (2018) Minimax optimal procedures for locally private estimation. *J. Amer. Statist. Assoc.* 113(521):182–201.

Dwork C, Roth A (2014) The algorithmic foundations of differential privacy. *Foundations Trends Theoretical Comput. Sci.* 9(3-4):211–407.

Dwork C, McSherry F, Nissim K, Smith A (2006b) Calibrating noise to sensitivity in private data analysis. *Proc. Theory of Cryptography Conf.* (Springer, Berlin), 265–284.

Dwork C, Talwar K, Thakurta A, Zhang L (2014). Analyze Gauss: Optimal bounds for privacy-preserving principal component analysis. *Proc. Annual ACM Sympos. on Theory of Comput.* (ACM, New York), 11–20).

Dwork C, Kenthapadi K, McSherry F, Mironov I, Naor M (2006a) Our data, ourselves: Privacy via distributed noise generation. *Proc. Annual Internat. Conf. on the Theory and Applications of Cryptographic Techniques* (Springer, Berlin), 486–503.

Dwork C, Smith A, Steinke T, Ullman J, Vadhan S (2015) Robust traceability from trace amounts. *Proc. IEEE 56th Annual Sympos. on Foundations of Comput. Sci.* (IEEE, New York), 650–669.

Evfimievski A, Gehrke J, Srikant R (2003) Limiting privacy breaches in privacy preserving data mining. *Proc. ACM SIGMOD-SIGACT-SIGART Sympos. on Principles of Database Systems* (ACM, New York), 211–222.

Fan J, Truong YK (1993) Nonparametric regression with errors in variables. *Ann. Statist.* 21(1):1900–1925.

Farias VF, Van Roy B (2010) Dynamic pricing with a prior on market response. *Oper. Res.* 58(1):16–29.

Federal Trade Commission (2012) Protecting consumer privacy in an era of rapid change. https://bit.ly/3k2AUhN.

Ferreira KJ, Simchi-Levi D, Wang H (2018) Online network revenue management using thompson sampling. *Oper. Res.* 66(6):1586–1602.

Fredrikson M, Lantz E, Jha S, Lin S, Page D, Ristenpart T (2014). Privacy in pharmacogenetics: An end-to-end case study of personalized warfarin dosing. *Proc. 23rd USENIX Security Sympos.* (USENIX Association, San Diego), 17–32.

Google (2014) Learning statistics with privacy, aided by the flip of a coin. Accessed April 9, 2022, https://ai.googleblog.com/2014/10/learning-statistics-with-privacy-aided.html.

Han Y, Liang Z, Wang Y, Zhang J (2021). Generalized linear bandits with local differential privacy. Ranzato M, Beygelzimer A, Dauphin Y, Liang PS, Wortman Vaughan J, eds. *Advances in Neural Information Processing Systems*, vol. 34 (Curran Associates, Inc., Red Hook, NY), 26511–26522.

Harrison JM, Keskin NB, Zeevi A (2012) Bayesian dynamic pricing policies: Learning and earning under a binary prior distribution. *Management Sci.* 58(3):570–586.

Hidano S, Murakami T, Katsumata S, Kiyomoto S, Hanaoka G (2017) Model inversion attacks for prediction systems: Without knowledge of non-sensitive attributes. *Proc. 15th Annual Conf. on Privacy, Security and Trust* (IEEE, New York), 115–11509.

Hoeffding W (1963) Probability inequalities for sums of bounded random variables. *J. Amer. Statist. Assoc.* 58(301):13–30.

Hu M, Momot R, Wang J (2022) Privacy management in service systems. *Manufacturing Service Oper. Management*, ePub ahead of print July 22, https://doi.org/10.1287/msom.2022.1130.

Javanmard A, Nazerzadeh H (2019) Dynamic pricing in high-dimensions. *J. Machine Learn. Res.* 20(9):1–49.

Keskin NB, Zeevi A (2014) Dynamic pricing with an unknown demand model: Asymptotically optimal semi-myopic policies. *Oper. Res.* 62(5):1142–1167.

Keskin NB, Zeevi A (2018) On incomplete learning and certainty-equivalence control. *Oper. Res.* 66(4):1136–1167.

Keskin NB, Li Y, Sunar N (2020) Data-driven clustering and feature-based retail electricity pricing with smart meters. Preprint, submitted March 7, https://dx.doi.org/10.2139/ssrn.3686518.

Kolata G (2019) Your data were 'anonymized'? These scientists can still identify you. *New York Times*.

Lei YM, Jasin S, Sinha A (2014) Near-optimal bisection search for nonparametric dynamic pricing with inventory constraint. Preprint, submitted September 20, https://dx.doi.org/10.2139/ssrn.2509425.

Lei YM, Miao S, Momot R (2020) Privacy-preserving personalized revenue management. Preprint, submitted May 9, https://dx.doi.org/10.2139/ssrn.3704446.

Miao S, Wang Y, Zhang J (2021) A general framework for resource constrained revenue management with demand learning and large action space. Preprint, submitted September 16, https://dx.doi.org/10.2139/ssrn.3841273.

Mishra N, Thakurta A (2015) (Nearly) optimal differentially private stochastic multi-arm bandits. *Proc. Conf. on Uncertainty in Artificial Intelligence* (AUAI Press, Arlington, VA), 592–601.

Qiang S, Bayati M (2016) Dynamic pricing with demand covariates. Preprint, submitted April 25, https://dx.doi.org/10.2139/ssrn.2765257.

Ren W, Zhou X, Liu J, Shroff NB (2020) Multi-armed bandits with local differential privacy. Preprint, submitted July 6, https://arxiv.org/abs/2007.03121.

Shariff R, Sheffet O (2018) Differentially private contextual linear bandits. *Advances in Neural Information Processing Systems (NeurIPS)* (Curran Associates, Inc., Red Hook, NY).

Tang W, Ho C-J, Liu Y (2020). Differentially private contextual dynamic pricing. In *Proceedings of the International Conference on Autonomous Agents and MultiAgent Systems* (International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC), 1368–1376.

Tsitsiklis JN, Xu K, Xu Z (2021) Private sequential learning. *Oper. Res.* 69(5):1575–1590.

U.S. Census Bureau (2020) 2020 census data products: Disclosure avoidance modernization. Accessed April 9, 2022, https://www.census.gov/programs-surveys/decennial-census/decade/2020/planning-management/process/disclosure-avoidance.html.

Vissers T, Nikiforakis N, Bielova N, Joosen W (2014) Crying wolf? On the price discrimination of online airline tickets. *Proc. 7th Workshop on Hot Topics in Privacy Enhancing Technologies* (Amsterdam).

Wang Z, Deng S, Ye Y (2014) Close the gaps: A learning-while-doing algorithm for single-product revenue management problems. *Oper. Res.* 62(2):219–482.

Wang Y, Chen X, Chang X, Ge D (2021) Uncertainty quantification for demand prediction in contextual dynamic pricing. *Production Oper. Management* 30(6):1703–1717.

Xu K (2018) Query complexity of Bayesian private learning. *Proc. Adv. in Neural Inform. Processing Systems* (Curran Associates, Inc., Red Hook, NY).

Xu J, Xu K, Yang D (2021) Optimal query complexity for private sequential learning against eavesdropping. Banerjee A, Fukumizu K, eds. *Proc. 24th Internat. Conf. Artificial Intelligence Statist.*, Proceedings of Machine Learning Research Series (PMLR), 2296–2304.

Yu B (1997) Assouad, Fano, and le Cam. *Festschrift for Lucien Le Cam* (Springer, Berlin), 423–435.

Zheng K, Cai T, Huang W, Li Z, Wang L (2020) Locally differentially private (contextual) bandits learning. Larochelle H, Ranzato M, Hadsell R, Balcan MF, Lin H, eds. *Advances in Neural Information Processing Systems* (Curran Associates, Inc., Red Hook, NY), 12300–12310.

Xi Chen is an associate professor at the Department of Technology, Operations, and Statistics at Stern School of Business, New York University. His research interests include statistical machine learning, stochastic optimization, and data-driven operations management.

Sentao Miao is an assistant professor in Bensadoun School of Retail Management and Desautels Faculty of Management at McGill University. His research interests are mainly in developing efficient learning and optimization algorithms with various applications in operations management.

Yining Wang is currently an associate professor at the Naveen Jindal School of Management, University of Texas at Dallas. His main research focus is on the development and analysis of sequential decision making methods under uncertainty, with emphasis to revenue management applications such as assortment optimization and dynamic pricing. His research is also connected with bandit online learning and reinforcement learning in machine learning research.