# Grace Period is All You Need: Individual Fairness without Revenue Loss in Revenue Management

Patrick Jaillet
Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, jaillet@mit.edu

Chara Podimata
Sloan School of Management, Massachusetts Institute of Technology, Cambridge, MA, 02139, podimata@mit.edu

Zijie Zhou
Operations Research Center, Massachusetts Institute of Technology, Cambridge, MA, 02139, zhou98@mit.edu

Imagine you and a friend purchase identical items at a store, yet only your friend received a discount. Would your friend's discount make you feel unfairly treated by the store? And would you be less willing to purchase from that store again in the future? Based on a large-scale online survey that we ran on Prolific, it turns out that the answers to the above questions are *positive*. Motivated by these findings, in this work we propose a notion of *individual fairness* in online revenue management and an algorithmic module (called *"Grace Period"*) that can be embedded in traditional revenue management algorithms and guarantee individual fairness. Specifically, we show how to embed the Grace Period in five common revenue management algorithms including Deterministic Linear Programming with Probabilistic Assignment (DLP-PA), Resolving Deterministic Linear Programming with Probabilistic Assignment (RDLP-PA), Static Bid Price Control (S-BPC), Booking Limit (BL), and Nesting (NESTING), thus covering both *stochastic* and *adversarial* customer arrival settings. Embedding the Grace Period does *not* incur additional regret for any of these algorithms. This finding indicates that there is *no* tradeoff between a seller maximizing their revenue and guaranteeing that each customer feels fairly treated.

## 1. Introduction

Revenue management plays a pivotal role in various sectors including airline, retail, advertising, and hospitality, as evidenced by previous research Williamson (1992), Talluri and Van Ryzin (1998, 2004), Talluri et al. (2004). In the most standard version, sequential decision-making for revenue management encompasses the following seller-customer interaction: The seller has access to limited-capacity *resources* of potentially more than one *types*. Customers of different *types* arrive sequentially (the order of arrival can be either *stochastic* or *adversarial*) over a finite period of time. Upon arrival, the seller can choose either to *accept* (i.e., sell the item to them) or *reject* (i.e., not sell) the customer. Acceptance leads to revenue generation equal to the payment made by the customer, alongside the consumption of specific units of resources based on the customer's type. On the other hand, rejection of a customer does not lead to any revenue or resource usage. The goal of the seller is to maximize the total anticipated revenue throughout the given period, while adhering to the capacity constraints of all resources.

This work emphasizes the equitable treatment of customers through the requirement of *individual fairness*. The concept of fairness is crucial not only from an ethical standpoint but also for its impact on customer behavior and potential to increase revenue. For example, think about the following (hypothetical, for now) scenario. Two stores, A and B, offer an identical product at a same price. However, Store A selectively offers discounts leading to perceived unfairness among customers. One may think that such perceived unfair treatment may deter customers from buying at Store A, possibly benefiting Store B. To confirm if customer reactions align with our aforementioned intuition, we conducted a large-scale online survey on Prolific, detailed in Section 6. Our survey's findings validate our priors; customers do indeed perceive various forms of unfair treatment from sellers and this in turn affects the probability of them buying again from certain stores.

We next define individual fairness in the context of revenue management.

Definition 1 (Individual Fairness in Revenue Management). Let $n$ be the number of customer types, with each type $i \in [n]$ having $n_i$ customers arriving. An algorithm $\mathcal{A}$ is $(\alpha, \delta)$-fair for any choice of $\alpha, \delta \in [0,1]$ if for each customer of type $i$, with a probability at least $1-\delta$, for any $u, v \in [n_i]$,

$$\mathbb{P}\left[i(v) \succ_{\mathcal{A}} i(u)\right] \leq \alpha |u-v|,$$

where $i(u), i(v)$ denote the $u$-th and $v$-th customer's of type $i$, and $i(v) \succ_{\mathcal{A}} i(u)$ denotes that algorithm $\mathcal{A}$ accepts customer $i(v)$ but not $i(u)$.

Intuitively speaking, an algorithm satisfying individual fairness ensures that the probability of two customers of the same type that arrive closely in time but receive disparate treatments is very small. Our definition of individual fairness (both the time- and the type-dependent components) is driven by our survey findings (cf. Section 6), which show that people often perceive unfairness when they do not obtain a resource that other people that are similar to them receive. As a result, we aim to upper bound the probability of customers of the same type ($i(v)$ and $i(u)$) receiving disparate treatment. As for the time component, our survey findings suggest that individuals perceive different treatments occurring in the *near* term as more unfair compared to those in the distant past. This insight suggests that the probability of two similar customers being treated differently, $\mathbb{P}\left[i(v) \succ_{\mathcal{A}} i(u)\right]$, should be moderated according to the *time gap* between their arrivals, specifically using the L1 norm, $\alpha|u-v|$, as a measure[1].

In this paper, our primary objective is to incorporate the principle of individual fairness into the most commonly used algorithms solving the quantity-based revenue management problem, including Deterministic Linear Programming with Probabilistic Assignment (DLP-PA), Resolving Deterministic Linear Programming with Probabilistic Assignment (RDLP-PA), Static Bid Price Control (S-BPC), Booking Limit (BL), and Nesting (NESTING). We modify each algorithm one by one to meet individual fairness criteria as defined in Definition 1. In addition to implementing individual fairness, we also strive to maintain revenue optimization, ensuring that incorporating fairness does not lead to any loss compared to the original algorithm's performance[2]. By adapting these well-known algorithms, we aim to ensure that any revenue management problems addressed by them also upholds the principles of individual fairness.

### 1.1. Main Contributions

**"Grace Period" Design.** In Section 3, we introduce the *Grace Period* which we find to be the workhorse concept for achieving individual fairness in revenue management. At a simplified level, the Grace Period works as follows: As the seller's available resources approach depletion, an arriving customer is either accepted or rejected based on the treatment of the preceding customer. If the previous customer was accepted, the succeeding customer is likewise accepted with a probability of $(1-\alpha)$. However, if the previous customer was rejected, the incoming customer is also declined. This design (which gets utilized only when the resources are almost depleted) forms a grace period during which the probability of unequal treatment is confined. We remark that the Grace Period for customers of type $i$ can (and will in general) be different from the Grace Period for customers of type $j$ (for $i \neq j$). The tricky part is to guarantee that despite these distinct grace periods, the seller does not sell resources beyond their allowable capacity. We showcase how the Grace Period can be used via an example analysis of the First-Come-First-Serve (FCFS) algorithm.

The rationale for the Grace Period Design lies in its mitigation of unfairness. If we consider a scenario where the Grace Period Design isn't applied and we adhere to a first-come-first-serve model, the pattern of acceptances and rejections is distinctly binary; the early arrivals are accepted,

[1] All of our results still hold even if we take another metrics, such as L2 norm: $\alpha\|u-v\|^2$.

[2] In Appendix EC.5, we show how our approach can also produce an individually-fair variant of Bid Price Control with Online Gradient Descent (BPC-OGD), albeit with a small asymptotic loss in revenue.

followed by a series of rejections as resources are exhausted. In such a scenario, the initial group of customers who are declined would be unfairly treated (per Definition 1). However, the application of the Grace Period Design spreads this perception of unfairness more equally among earlier arrivals. Hence, this design maintains individual fairness among every customer with high probability.

**Grace-Period-Enhanced Algorithms for Revenue Management.** In Sections 4 and 5, we integrate the Grace Period design into five well-known algorithms frequently used in revenue management (RM): DLP-PA, RDLP-PA, S-BPC, BL and NESTING. The first three DLP-PA, RDLP-PA, S-BPC are used for stochastic customer arrivals, while BL and NESTING are used for adversarial arrivals. For completeness, we outline each of these algorithms together with the explanation of why they are not individually fair in Section 2. Based on the different structure of these algorithms, we employ different methods to incorporate the grace period design into them. For example, in DLP-PA, each type $i$ customer is accepted with certain probability $p_i$, leading to unfairness because consecutive type $i$ customers are treated differently with a probability $p_i(1-p_i)$, violating our fairness definition (1) for $\alpha < p_i(1-p_i)$. To address this, we avoid random decisions for individual arrivals. Instead, we divide the timeline into segments, accepting a random number of customers in a first-come-first-serve manner per segment. Grace periods *inside* each segment ensure fairness when decisions change from acceptance to rejection within each segment; grace periods *between each segment* guarantee fairness when decisions change from rejection to acceptance between the segments. For clarity, we refer to the new algorithms as GP-Enhanced.

***No*** **Trade-off between Revenue Maximization and Individual Fairness.** Contrary to the common belief that there exists a trade-off between maximizing revenue and achieving individual fairness, we demonstrate that by augmenting standard RM algorithms with the grace period design, it is possible to attain optimal revenue (asymptotically) while simultaneously ensuring individual fairness for most of the algorithms.

For settings where the customer arrival process is stochastic, the performance of RM algorithms is measured in terms of *regret* with regards to the benchmark of the optimal revenue achievable (which assumes known realized demand). Table 1 contains a summary of the regret for GP-Enhanced algorithms for stochastic arrivals versus the traditional non-individually fair counterparts. For settings where the customer arrival process is adversarial, the performance of RM algorithms is measured in terms of the *competitive ratio*; that is, the worst-case ratio (across all possible arrival instances) between the revenue generated by the algorithm and the hindsight optimal (i.e., the maximum revenue assuming full knowledge of the realized demand). Table 2 contains a summary of the competitive ratio for GP-Enhanced algorithms for adversarial arrivals versus the traditional non-individually fair counterparts.

**Table 1 Regret of Original Algorithms and Grace Period Enhanced Algorithms under Stochastic Arrival**

| Original Algorithm | Regret of Original Algorithm | Regret of Grace Period Enhanced Algorithm |
|---|---:|---:|
| DLP-PA | $O(\sqrt{T})$ | $O(\sqrt{T})$ (Theorem 2) |
| RDLP-PA | $O(T^{1/3})$ | $O(T^{1/3})$ (Theorem 2) |
| S-BPC | $O(\sqrt{T})$ | $O(\sqrt{T})$ (Theorem 3) |

**Table 2 Competitive Ratio of Original Algorithms and Grace Period Enhanced Algorithms under Adversarial Arrival.** $C \in [0,1]$ **is a setting-specific value that is specified in the analysis.**

| Original Algorithm | Competitive Ratio of Original Algorithm | Competitive Ratio of Grace Period Enhanced Algorithm |
|---|---:|---:|
| BL | $C$ | $C - O(\log m/m)$ (Theorem 4) |
| NESTING | $C$ | $C - O(\log m/m)$ (Theorem 5) |

***No*** **Additional** ***Computational*** **Cost for Implementing Individual Fairness.** One might conjecture that simultaneously achieving optimal revenue and being individually fair might necessitate additional computational cost for the algorithm. However, this is *not* the case. In all GP-Enhanced algorithms, we have preserved the algorithms' original structure to the maximum extent possible, maintaining the order of computational complexity equivalent to that of the original algorithm.

**Survey Highlighting the Significance of Individual Fairness in RM.** We conducted an online survey between October and November 2023 on Prolific about people's perception of unfair treatment in RM. Our findings suggest: (i) When customers witness behavior in a store that is considered "unfair" (as described in Definition 1) before they shop, their inclination to buy decreases. (ii) A customer who has bought from a store in the past and then experiences what they perceive as unfair treatment is likely to decrease their purchasing from that store in the future. (iii) Individuals perceive different treatment happening in the near term as more unfair compared to those in the distant past. With these observations, we can evaluate the importance of individual fairness to business and validate the mathematical definition of individual fairness metrics in Definition 1.

### 1.2. Other Related Work

**Group Fairness.** While a considerable volume of literature exists concerning fairness in RM algorithms, much of it primarily aims to achieve group fairness (Aminian et al. 2023, Balseiro et al. 2021, Cayci et al. 2020, Donahue and Kleinberg 2020, Ma et al. 2020, Manshadi et al. 2021). Group fairness in decision-making entails ensuring that every group (or type) of customer is treated equitably. However, group fairness does not guarantee individual fairness. This is because, firstly, individuals might not be aware of the groups to which they belong. Secondly, individuals within each group can be treated inequitably based on their specific contexts and features.

**Individual Fairness from the Regulator's Perspective.** Individual fairness can be categorized into two types, each viewed from distinct perspectives. The first one considers the regulator's perspective (Arsenis and Kleinberg 2022), suggesting that the acceptance probability for each arriving customer should be equal. The second type looks at fairness from the customer's perspective, seeking to reduce the likelihood of individuals experiencing envy and subsequently minimize the loss associated with such envy. Our research, along with Sinclair et al. (2022), Gupta and Kamble (2021), Guruswami et al. (2005), Bérczi et al. (2023), falls under this latter category of metrics.

These two dimensions of individual fairness differ due to their distinctive focuses. Consider an example: two customers arrive, and the first algorithm accepts both customers A and B with a probability of 0.5. The second algorithm accepts both customers A and B with a probability of 0.1. It appears that both algorithms uphold individual fairness from the regulator's perspective. However, the probability that customer 1 is rejected and customer 2 is accepted under the first algorithm is $(1-0.5)\cdot 0.5 = 0.25$, whereas for the second algorithm, it is $(1-0.1)\cdot 0.1 = 0.09$. This implies that the second algorithm would potentially lead to lower feelings of envy and unfairness among customers, thus demonstrating the difference in the two metrics of individual fairness.

**Modern Revenue Management Models.** Due to the limitations of the traditional network revenue management model, researchers have diversified its approach for greater relevance and depth. Despite these advancements, the resolution of these contemporary models often relies on the foundational algorithmic frameworks discussed in Section 2. Our contributions aim to refine these foundational structures, thereby promoting individual fairness across all advanced revenue management models that employ decision-making algorithms outlined in Section 2. For instance, to ensure group fairness, a DLP-PA algorithm is proposed by Ma et al. (2020), while a BPC-OGD algorithm is suggested by Balseiro et al. (2021). In the context of choice-based revenue management, a choice-based DLP-PA algorithm is recommended by Liu and Van Ryzin (2008), Gallego et al. (2015). To tackle revenue management issues involving offline prediction in demand, Balseiro et al.

(2022), Golrezaei et al. (2023) advocate for a BL algorithm. Revenue management with correlated arrivals is addressed by Jiang (2023) with an S-BPC method, and Jiang et al. (2020) proposes a BPC-OGD algorithm for non-stationary arrivals. In scenarios involving revenue management with reusable resources, Huo and Cheung (2022) employs a NESTING algorithm. Finally, for revenue management with horizontal uncertainty, a BPC-OGD algorithm is suggested by Balseiro et al. (2023a).

**Price-based Revenue Management.** Contrasting with the quantity-based model in this paper, price-based revenue management represents a distinct category. Here, pricing decisions inversely affect demand, with higher prices leading to lower demand and vice versa. Maglaras and Meissner (2006) suggests that, under certain conditions, price-based and quantity-based models align in function. Furthermore, price-based strategies also utilize the algorithmic frameworks mentioned in Section 2. Works by Gallego and Van Ryzin (1994, 1997) implement the DLP-PA algorithm for decision-making, whereas Jasin and Kumar (2012), Jasin (2014), Wang and Wang (2022) employ the RDLP-PA algorithm. In Section 7, we explore the concept of individual fairness within the price-based framework and discuss the integration of a grace period design in this context.

## 2. Model and Preliminaries

Although our work does not require adherence to a specific model, for ease of exposition, we present the classical *network revenue management model*[3]. In the main text, we will operate under the assumption that all algorithms are applied to this classical model.

In classical network revenue management, there are $L$ types of indivisible resources, each with a capacity of $m_j, j \in [L]$. All capacities $m_j, \forall j \in [L]$, scale consistently, i.e., there exist constants $q_j$ for every $j \in [L]$ such that $m_j = q_j m$. There is a total of $T$ —where $T$ is also scaling with $m$— customers, arriving one by one. There are $n$ customer types. When a customer of type $i \in [n]$ arrives, they may request multiple units from several resource types $j$; we denote this as $A_{ij}$. Matrix $\mathbf{A} \in \mathbb{R}_+^{n\times L}$ is called the *demand* matrix. Without loss of generality, we assume that $A_{ij} \le \bar{a}$. When a type $i$ customer arrives in the system, the decision-maker has two choices: either accept the customer, allocate $A_{ij}$ units of resource $j$, and earn $r_i$ in revenue, where $r_i$ are known constants; or decline the customer's request. The primary objective is to *maximize cumulative revenue.*

We study two variants of the above model based on the *customers' arrival process*. For the first variant (Section 4), we assume that customers arrive sequentially at random; i.e., each customer of type $i$ has an associated arrival rate $\lambda_i$. In every round, at most one customer will arrive, with the probability of a type $i$ customer arriving being $\frac{\lambda_i}{\sum_{i'=0}^n \lambda_{i'}}$[4]. Here, $\lambda_0$ represents the rate for no arrivals. In Sections 4.1, and 4.2, we study the case where the arrival rates $\lambda_i$ are known; Appendix EC.5.1 studies the case where the rates $\lambda_i$ are unknown. Under stochastic arrivals, *regret* is commonly used for measuring the performance of an online algorithm. For the second variant (discussed in Section 5), we study the case where an adversary determines both the quantity and sequence of each type of customer, commonly referred to as the worst-case/adversarial scenario. Under adversarial arrivals, the *competitive ratio* is the metric to study the performance of an online algorithm.

### 2.1. Regret and Competitive Ratio Analysis

We first focus on the *stochastic* arrival setting. Let $\Lambda_i(T)$ be the random variable corresponding to the total number of type $i$ customers arriving between $[0,T]$. The *hindsight optimum* denotes the optimal revenue achievable if the exact number of arrivals for each customer type is known in advance. For the classic network RM, the hindsight optimum is defined as

$$\text{Rev}^\star = \max_{\mathbf{x}} \Big\{ \sum_{i\in[n]} r_i x_i \quad \text{s.t.} \sum_{i\in[n]} \mathbf{A}_i x_i \le \mathbf{m},\ 0 \le x_i \le \Lambda_i(T), \forall i \in [n] \Big\}, \tag{1}$$

[3] There is a special case where the decision-maker only has one type of resource, and the special model is called as *single-leg revenue management.*

[4] Typically, we normalize all arrival rates so that $\sum_{i'=0}^n \lambda_{i'} = 1$.

where in Eq. (1), $\mathbf{x}$ is the decision variable vector representing the quantity of each type of customer to accept, $\mathbf{A}_i$ is the $i^{\text{th}}$ column of preference matrix $\mathbf{A}$, $\mathbf{m}$ is the vector containing the capacity of each resource $m_j$, and $\Lambda_i(T)$ is a known value representing the exact number of type $i$ arrivals.

Given that the hindsight optimum is privy to the precise count of each customer type's arrivals, it is typically utilized as a benchmark to evaluate the online algorithms. The following definition introduces the concept of *regret*.

DEFINITION 2. Given $\text{Rev}(\pi)$ as the expected revenue generated by the online algorithm $\pi$, the regret of algorithm $\pi$ is defined as $\text{Regret}(\pi) := \mathbb{E}[\text{Rev}^\star] - \text{Rev}(\pi)$.

Note that since the hindsight optimum is informed of the precise number of each type's arrivals, the resulting revenue is an upper bound for the revenue of any online algorithm. Hence, Regret $\geq 0$, and the smaller the regret, the better.

Next, we consider the *adversarial* arrival setting. Let $\mathcal{I}$ represent the set of all potential arrival instances. For any $I \in \mathcal{I}$, let $\text{OPT}(I)$ denote the maximum revenue achievable for arrival instance $I$. We denote $\text{Rev}(\pi(I))$ as the expected revenue elicited by an online algorithm $\pi$ under the specific arrival instance $I$.

DEFINITION 3. The *competitive ratio* of an algorithm $\pi$ is defined as $\text{CR}(\pi) = \inf_{I \in \mathcal{I}} \frac{\text{Rev}(\pi(I))}{\text{OPT}(I)}$.

Note that competitive ratio is *not* equivalent to regret. If the competitive ratio of an online algorithm $\pi$ equals 0, this implies that a sequence exists in which the algorithm $\pi$ fails to generate *any* revenue. However, generally speaking, it is possible that $\pi$ could perform efficiently on the majority of sequences, and if the arrival process is stochastic, $\pi$ could also incur diminishing regret.

In the next two subsections, we outline the most popular algorithmic structures for RM with *stochastic* and *adversarial* arrivals. The pseudo-code for all the algorithms introduced below can be found in the Appendix.

### 2.2. Popular Algorithm Structures under Stochastic Arrivals

**Algorithm DLP-PA.** The Deterministic Linear Programming with Probabilistic Assignment algorithm (DLP-PA) solves the deterministic linear programming at the beginning and uses the optimal primal solution to make probabilistic decisions, where the DLP formulation is derived by replacing all random variables with their expected values in the hindsight formulation Eq. (1). In the network RM model, since $\lambda_i$ is the likelihood of type $i$ arrival in each time epoch, the total number of type $i$ customers arriving between $[0,T]$ is $\Lambda_i(T) \sim \text{Bin}(T, \lambda_i)$. The DLP is defined as:

$$\text{DLP}^\star = \max_{\mathbf{x}} \Big\{ T \sum_{i \in [n]} r_i x_i \quad \text{s.t.} \sum_{i \in [n]} \mathbf{A}_i x_i \leq \frac{\mathbf{m}}{T},\ 0 \leq x_i \leq \lambda_i, \forall i \in [n] \Big\}, \tag{2}$$

We denote the optimal solution of Eq. (2) as $\mathbf{x}^*$ (assuming it is unique). Algorithm DLP-PA then accepts each type $i$ customer with a probability of $\frac{x_i^*}{\lambda_i}$, given sufficient capacity. The details can be found in Algorithm 5 in Appendix EC.2. The asymptotic regret of Algorithm DLP-PA for classical network RM is $O(\sqrt{T})$ (Gallego and Van Ryzin 1994, 1997).

Note that for $\alpha < \frac{x_i^*}{\lambda_i}(1 - \frac{x_i^*}{\lambda_i})$ *Algorithm DLP-PA is unfair to all type $i$ customers.* This is because if each type $i$ customer is accepted with a probability of $\frac{x_i^*}{\lambda_i}$, it results in a situation where each type $i$ customer is rejected and the prior customer is accepted with probability $\frac{x_i^*}{\lambda_i}(1 - \frac{x_i^*}{\lambda_i}) > \alpha$.

**Algorithm Resolving DLP (RDLP-PA).** The RDLP-PA algorithm addresses DLP's limitation of not accounting for the demand's randomness after $t = 0$. At a high level, RDLP-PA solves the DLP and employs its optimal primal solution, denoted as $\mathbf{x}^*$, to make probabilistic assignments. However, at time point $t^*$, the applicability of the initial optimal solution $\mathbf{x}^*$ for making decisions may be compromised due to the random noise of the stochastic arrival process and probabilistic decision behavior. Hence, we re-solve the DLP with the remaining capacity and remaining time horizon, and continue making probabilistic assignments by the new optimal solution $\tilde{\mathbf{x}}^*$ until the end. For network RM, Reiman and Wang (2008) show that the regret incurred by RDLP-PA is $O(T^{1/3})$.

Similarly to the DLP-PA algorithm, the *RDLP-PA algorithm is unfair to all type $i$ customers* for $\alpha < \min\{\frac{x_i^*}{\lambda_i}(1-\frac{x_i^*}{\lambda_i}), \frac{\tilde{x}_i^*}{\lambda_i}(1-\frac{\tilde{x}_i^*}{\lambda_i})\}$.

**Algorithm Static Bid Price Control (S-BPC).** In contrast to the DLP-PA and RDLP-PA algorithms, bid price control is a threshold-based policy that makes deterministic decisions for every arriving customer. A Static Bid Price Control (S-BPC) uses a consistent threshold price for each leg, which remains unchanged over time. A request is accepted if the revenue it will provide the supplier exceeds the threshold price and is otherwise rejected.

To obtain the optimal fixed threshold price, Williamson (1992), Talluri and Van Ryzin (1998), Talluri et al. (2004) propose solving the dual problem of the DLP at the beginning, denoting the optimal dual variable as $\theta^* \in \mathbb{R}^L$ (assuming uniqueness). When each customer of type $i$ arrives, if the revenue $r_i$ exceeds the aggregated bid price $\sum_{j=1}^L \theta_j^* A_{ij}$, the customer is accepted; if not, the customer is rejected. The regret of S-BPC for network RM is $O(\sqrt{T})$.

We note that, once the bid price for a type $i$ is established, every customer of type $i$ is either accepted (if there is sufficient capacity) or rejected. Therefore, if resource capacity exceeds demand, Algorithm S-BPC is inherently fair because it either accepts or rejects all customers of type $i$. However, if the demand exceeds the resource capacity, Algorithm S-BPC's first-come-first-serve nature leads to unequal treatment between earlier and later customers.

### 2.3. Popular Algorithm Structures under Adversarial Arrivals

**Booking Limit (BL).** In the BL algorithm (Williamson 1992), the supplier assigns a fixed quota/booking limit $\mathbf{b} = \{b_1, b_2, \ldots, b_n\}$ for each customer type. In the asymptotic setting, all booking limits scale with the resource quantity; hence, $b_i = \Theta(m)$ for $i \in [n]$. The algorithm accepts any type $i$ customers provided that there is sufficient resource capacity and the number of accepted type $i$ customers has not yet reached $b_i$. Beyond this point, all type $i$ customers are rejected. The BL algorithm is not individually fair; for any type $i$, the $b_i^{\text{th}}$ type $i$ customer is accepted while the $(b_i+1)^{\text{th}}$ type $i$ customer is rejected.

**NESTING.** The NESTING algorithm (Williamson 1992) is applicable for settings with a single type of resource and where customers can be categorized and ranked based on the revenue $r_i$ that they add to the supplier. Unlike the approach of assigning individual booking limits to each customer type (which is suboptimal) the NESTING algorithm assigns booking limits to aggregated categories of customer types. For instance, in a situation with three customer types ranked as $r_1 > r_2 > r_3$, the algorithm sets a booking limit $b_1$ for type 1 customers and $b_2$ for the combined group of type 1 and type 2 customers. This method demonstrates superior performance compared to traditional booking limits and is considered optimal in the context of single-resource scenarios. However, a significant limitation of the NESTING algorithm is its inapplicability in multi-resource environments, where ranking customers becomes impossible.

The NESTING framework shares some similarities with the BL algorithm. Initially, $n$ quotas $\mathbf{b} = \{b_1, b_2, \ldots, b_n\}$ need to be established. Following this, $b_i$ is assigned as a quota for the total number of type $i$, $i+1$, …, $n$ customers. Much like the BL, the NESTING algorithm also lacks individual fairness owing to its quota-based FCFS structure.

## 3. The "Grace Period"

In this section, we present the concept of the "grace period" and present the analysis of its incorporation into the First-Come-First-Serve algorithm (FCFS). This will serve as a building block for the GP-Enhanced algorithms that we will follow. FCFS works both in the case of *stochastic* and *adversarial* arrival sequences in network revenue management problems and is rather simple: it keeps accepting every arriving customer until the capacity of the requested resources is not enough. The primary sources of unfairness in RM settings (and hence also in the case of FCFS) are: (i) limited resources, and (ii) the inherent structure of the algorithm. Note that when the available resources exceed the total demand, then FCFS is always *fair* since it accepts all customers; the problem arises when there are limited resources compared to the demand faced.

At a high level, the *grace period* is a time interval during which the algorithm (i.e., FCFS in this case) decides when to stop accepting customers gracefully using randomness.

DEFINITION 4. (Decreasing Grace Period) The *decreasing grace period* $[t_1(i), t_2(i)]$ for type $i$ customers is defined as: for *any* $i(k)$ customer that arrives in between round $t_1(i)$ and $t_2(i)$, $i(k)$ is accepted with a probability of $1-\alpha$ if customer $i(k-1)$ was accepted, and $i(k)$ is rejected if $i(k-1)$ was rejected.

Note that the grace period is defined *per customer type*, i.e., in general, each customer type will have a different grace period. The exact mathematical instantiation of the parameters of the grace period (i.e., the interval $[t_1(i), t_2(i)]$) as well as the fairness parameters $\alpha, \delta$ are *algorithm* specific. To illustrate this, we mathematically instantiate the grace period for the FCFS algorithm below.

### 3.1. Decreasing Grace Period in Network Revenue Management under FCFS

To achieve $(\alpha, \delta)$-fairness in network revenue management under FCFS, we can utilize the decreasing grace period as follows: for any $\alpha, \delta$, let $\gamma = \log_{1-\alpha}\delta$, and assign a decreasing grace period $[t_1(i), t_2(i)]$, where $t_1(i) = \inf\{t : \min_{j\in[L]} m_j(t) \le \bar{a}n\gamma\}$ and $t_2(i) = T$ to each type $i$ customer within the range of $[n]$. Here, $m_j(t)$ denotes the remaining capacity of resource $j$ at time $t$. In other words, in order to address the unfairness arising from resource scarcity, it is sufficient for FCFS to have a common decreasing grace period for all customer types. As a result, for FCFS we will drop the dependence on the customer type and denote the grace period as $[t_1, t_2]$. The following theorem states that, by implementing the decreasing grace period as above, we satisfy $(\alpha, \delta)$-fairness, and the revenue loss in comparison to the FCFS algorithm is bounded.

THEOREM 1. *Assigning a decreasing grace period* $[t_1, t_2]$*, where* $t_1 = \inf\{t : \min_{j\in[L]} m_j(t) \le \bar{a}n\gamma\}$ *and* $t_2 = T$ *to each type* $i$ *customer, where* $\gamma = \log_{1-\alpha}\delta$*, FCFS is* $(\alpha, \delta)$*-fair for network revenue management. Moreover, compared to the FCFS algorithm without grace period, the revenue loss is bounded by* $\frac{\bar{a}}{\underline{a}} n\gamma\bar{r}$*, where* $\bar{r} = \sup_{i\in[n]} r_i$*,* $\bar{a} = \sup_{i,j} A_{ij}$*, and* $\underline{a} = \inf_{i,j} A_{ij}$*.*

To prove Theorem 1, we first introduce a lemma below (see Appendix EC.3 for proof).

LEMMA 1. *Given an online algorithm* $\mathcal{A}$*, for any customer type* $i \in [n]$*, and for any* $u$*, if* $\mathbb{P}(i(u+1) \succ_{\mathcal{A}} i(u)) \le \alpha$ *and* $\mathbb{P}(i(u-1) \succ_{\mathcal{A}} i(u)) \le \alpha$*, then* $\mathcal{A}$ *satisfies individual fairness (Definition 1); namely, for any* $u, v \in [n_i]$*:* $\mathbb{P}(i(v) \succ_{\mathcal{A}} i(u)) \le \alpha|u-v|$*, with probability at least* $1-\delta$*.*

At a high level, the lemma states that for any algorithm $\mathcal{A}$, if for a customer type $i$ the probability that the $u$-th customer gets rejected while the $(u+1)$-th is accepted is upper bounded by $\alpha$ and the same thing holds for customers $u$ and $u-1$, then the algorithm is $(\alpha, \delta)$-fair.

We highlight the intuition of why the revised FCFS is $(\alpha, \delta)$-fair. In the revised algorithm, every customer arriving during $[1, t_1 - 1]$ is accepted, but starting from the period $t_1$, the acceptance probability for each customer type progressively decreases. This decrease is conditioned on the preceding customer's realization. This implies that if the resources are not depleted, the probability that customers of the same type that arrive one after the other get different treatment is bounded by $\alpha$. At round $t_1$, the remaining capacity of each resource $j \in [L]$ is at least $\bar{a}n\gamma$. Given that the stopping time (i.e., the time that the algorithm stops accepting them) for each customer type is a geometric random variable, we can apply standard concentration bounds to demonstrate that, with a high probability, all geometric random variables will stop before $\bar{a}n\gamma$ resource $j$ are exhausted for each $j \in [L]$. This concludes the proof that the revised algorithm is $(\alpha, \delta)$-fair.

To obtain a better understanding about the various parameters describing the revenue loss and the efficiency of our proposed algorithm, note that if $\delta = 1/m$, and given that $\underline{a}$, $\bar{a}$, $n$ and $\bar{r}$ are constants, Theorem 1 implies that the loss in revenue relative to the FCFS algorithm is bounded by $O(\log m)$. Next, we provide the full proof of Theorem 1.

*Proof of Theorem 1* First, we show that by assigning a decreasing grace period $[t_1(i), t_2(i)]$, where $t_1(i) = \inf\{t : \min_{j\in[L]} m_j(t) \le \bar{a}n\gamma\}$ and $t_2(i) = T$ to each type $i$ customer, Algorithm FCFS is $(\alpha,\delta)$-fair. By Lemma 1, we only need to show that for any customer type $i$, with probability at least $1-\delta$, for any $u \in [n_i]$, $\mathbb{P}(i(u+1) \succ_{\mathcal{A}} i(u)) \le \alpha$ and $\mathbb{P}(i(u-1) \succ_{\mathcal{A}} i(u)) \le \alpha$.

Define event $E$ as $E := \{$no resource is depleted in the time interval $[0,T]\}$. We show that $E$ happens with probability at least $1-\delta$ by showing that the complement of $E$ happens with probability at most $\delta$.

$$
\begin{aligned}
\mathbb{P}(E^{\mathsf{C}}) &\le \mathbb{P}(\text{more than } \gamma n \text{ customers are accepted in the grace period}) \\
&\le \mathbb{P}(\exists i \in [n], \text{ s.t. the number of type } i \text{ customers accepted in the grace period } \ge \gamma) \quad (3)\\
&= 1 - \mathbb{P}(\exists i \in [n], \text{ s.t. the number of type } i \text{ customers accepted in the grace period } < \gamma) \\
&= (1-\alpha)^{\gamma} \quad (4)\\
&= (1-\alpha)^{\log_{1-\alpha}\delta} = \delta,
\end{aligned}
$$

where (3) is due to the pigeonhole principle, and (4) is because for each type $i$, the number of type $i$ customers accepted after the grace period starts is a geometric random variable with success probability $\alpha$. Therefore, the cdf is $1-(1-\alpha)^{\gamma}$.

Conditional on event $E$ happening, for any customer $i(u)$ arriving within $[1, t_1]$, the algorithm accepts $i(u)$ with probability 1. For any customer $i(u)$ arriving within $[t_1, t_2]$, by the definition of the grace period, we obtain that $\mathbb{P}(i(u+1) \succ_{\mathcal{A}} i(u)) \le \alpha$ and $\mathbb{P}(i(u-1) \succ_{\mathcal{A}} i(u)) \le \alpha$.

To complete the proof of the theorem, we need to show that the revenue loss is bounded by $\frac{\bar{a}}{\underline{a}}n\gamma\bar{r}$. In the worst case for the revenue, the algorithm rejects all customers after $t_1$. By the definition of $t_1$, the remaining units for each $j$ are $\bar{a}n\gamma$. Since $\bar{a}n\gamma$ units of resource can serve at most $\frac{\bar{a}}{\underline{a}}n\gamma$ customers, we have that the revenue loss is at most $\frac{\bar{a}}{\underline{a}}n\gamma\bar{r}$.

### 3.2. Increasing Grace Period

We conclude this section with a related concept, the *increasing grace period*.

DEFINITION 5. The *increasing grace period* $[t_1(i), t_2(i)]$ for type $i$ customers is defined as: for any customer $i(k)$ that arrives within the interval from $t_1(i)$ to $t_2(i)$, customer $i(k)$ is *accepted* with probability $\alpha$ if customer $i(k-1)$ was *rejected*, and customer $i(k)$ is *accepted* if customer $i(k-1)$ was *accepted*.

The intuition behind the increasing grace period design is that, after a sequence of customer rejections, when the algorithm is ready to start accepting customers again in order to maintain individual fairness, it should not do so abruptly. Instead, the acceptance probability should incrementally increase, with the degree of increase conditioned on the previous customer's outcome.

## 4. Individually Fair RM Algorithms under Stochastic Arrivals

In this section, we focus on making three algorithm structures mentioned in Section 2.2 – DLP-PA, RDLP-PA, and S-BPC– individually fair without incurring significant additional regret. Based on the techniques we use for these three algorithms, we also enhance a more advanced algorithm – Bid Price Control with Online Gradient Descent (BPC-OGD) to satisfy the fairness metrics in Appendix EC.5. Further, as Algorithm DLP-PA is a variant for RDLP-PA (resolve 0 times), we analyze DLP-PA and RDLP-PA together.

### 4.1. Algorithm DLP-PA and RDLP-PA

Here, we introduce the Grace Period Enhanced RDLP-PA algorithm (GP-Enhanced-RDLP). The regret of the original RDLP-PA algorithm is $O(T^{\beta})$, where $\beta$ is a parameter that depends on the frequency of resolving the DLP. For example, $\beta = 1/2$ corresponds to resolving zero times (i.e., Algorithm DLP-PA), while $\beta = 1/3$ corresponds to resolving one or more times.

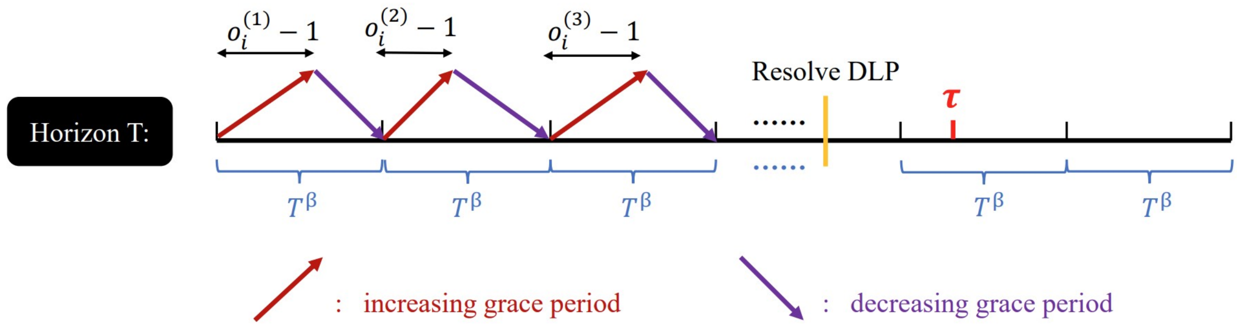


**Figure 1 Decisions from Algorithm 1 on type $i$ customers**

We begin by uniformly partitioning the time horizon $[0,T]$ into $K = T^{1-\beta}$ segments of size $T^\beta$. We solve the DLP with the initial capacity vector at $t=0$, and solve the DLP with the remaining capacity vector at $t=t^*$, where $t^*$ is the re-solving time point of the original RDLP-PA algorithm. In the first time segment ($k=1$), for each type $i$ customer, we sample a random variable $y_i \sim \text{Bin}(\lambda_i T^\beta, \frac{x_i^*}{\lambda_i})$. Our objective is to accept approximately $y_i$ customers of type $i$ in this interval, akin to the original RDLP-PA algorithm's acceptance of $\text{Bin}(\lambda_i T^\beta, \frac{x_i^*}{\lambda_i})$ customers of type $i$. However, different from RDLP-PA's probabilistic allocation, we initially accept $y_i - \bar{a}n\gamma - 1$ customers of type $i$, where $\gamma = \log_{1-\alpha}\delta$. Subsequently, we implement a decreasing grace period for type $i$ customers until the end of the first interval. According to Theorem 1, this ensures the individual fairness metrics while accepting $z_i^{(1)} = \Theta(\log T)$ fewer customers of type $i$ compared to Algorithm RDLP-PA.

Starting from the second time segment ($k \geq 2$), due to the probable rejection of the last few customers in segment $k-1$, an immediate acceptance of customers in segment $k$ is unfair. Thus, we introduce an increasing grace period, progressively shifting from rejection to acceptance. Notably, in segment $k-1$, as $z_i^{(k-1)}$ fewer customers of type $i$ are accepted compared to Algorithm RDLP-PA, we compensate by accepting $z_i^{(k-1)}$ additional type $i$ customers in segment $k$. This is achieved by drawing $y_i \sim z_i^{(k-1)} + \text{Bin}(\lambda_i T^\beta, \frac{x_i^*}{\lambda_i})$ and implementing an increasing grace period until the arrival of the $(y_i - \bar{a}n\gamma)^{\text{th}}$ customer of type $i$. Subsequently, a decreasing grace period is applied until the conclusion of the time segment.

Lastly, following the methodology introduced in Section 3.1, as a resource approaches depletion, an additional decreasing grace period is introduced to conclude the algorithm. The details can be found in Algorithm 1, and the intuition is illustrated in Figure 1.

---

**Algorithm 1:** GP-Enhanced-RDLP

**Input:** Preference matrix $\mathbf{A}$. Arriving rate vector $\lambda$. Capacity vector $\mathbf{m}$. Resolving time point $t^*$. Regret bound of input R-DLP algorithm $O(T^\beta)$. Fairness parameters $(\alpha, \delta)$. $\gamma = \log_{1-\alpha}\delta$.

**Output:** Optimal solution vectors $\mathbf{x}^*$ and $\tilde{\mathbf{x}}^*$.

**Initialize:** Vector $\mathbf{z}^{(0)} = 0$. $K = T^{1-\beta}$.

Solve DLP based on the initial capacity, and denote the optimal primal solution as $\mathbf{x}^*$.

**for** $k \in \{1, 2, ..., \lfloor t^* T^{-\beta} \rfloor\}$ **do**

  Run Algorithm 2 with $t_1 = (k-1)T^\beta$, $t_2 = kT^\beta$, $\mathbf{z} = \mathbf{z}^{(k-1)}$, $\mathbf{x}^*$. Get the output $\mathbf{z}^{(k)} \leftarrow \mathbf{w}$.

Solve DLP based on the remaining capacity and time, and denote the optimal solution as $\tilde{\mathbf{x}}^*$.

**for** $k \in \{\lceil t^* T^{-\beta} \rceil, \lceil t^* T^{-\beta} \rceil + 1, \ldots, K\}$ **do**

  Run Algorithm 2 with $t_1 = (k-1)T^\beta$, $t_2 = kT^\beta$, $\mathbf{z} = \mathbf{z}^{(k-1)}$, $\tilde{\mathbf{x}}^*$. Get the output $\mathbf{z}^{(k)} \leftarrow \mathbf{w}$.

---

**Algorithm 2:** Auxiliary Algorithm for Algorithm 1

**Input:** Arriving rate vector $\lambda$. Capacity vector $\mathbf{m}$. Time horizon $(t_1, t_2)$. Fair parameters $(\alpha, \delta)$. Vector $\mathbf{z}$. Optimal primal solution $\mathbf{x}^*$.

**Output:** Vector $\mathbf{w}$, indicating the sum of rejected type $i$ customers.

Sample $y_i \sim z_i + \mathrm{Bin}((t_2 - t_1)\lambda_i, \frac{x_i^*}{\lambda_i})$.

**if** $t_1 \neq 1$ **then**

  Give an increasing grace period $[t_1, o_i - 1]$ to type $i$ customer, where $o_i$ is the time when the $(y_i - \bar{a}n\gamma)^{\text{th}}$ type $i$ customer arrives.

Give a decreasing grace period $[o_i, t_2]$ to type $i$ customer.

Return vector $\mathbf{w}$, whose $i^{\text{th}}$ element is the sum of number of rejected type $i$ customers in the increasing grace period $[t_1, o_i - 1]$ and in the decreasing grace period $[o_i, \tilde{o}_i]$, where $\tilde{o}_i$ is the time when the $y_i^{\text{th}}$ type $i$ customer arrives.

**for** $t \in \{t_1, t_1 + 1, \ldots, t_2\}$ **do**

  **if** $\min_{j \in [L]} m_j(t) - \bar{a}n\gamma \leq 0$ **then**

    Give a decreasing grace period $[t, T]$ to all types of customers.

THEOREM 2. *Given any $\alpha \in (0,1)$, $\delta = 1/T$, and any input RDLP-PA algorithm that attains a regret of $O(T^\beta)$, Algorithm GP-Enhanced-RDLP: (i) is $(\alpha, \delta)$-fair, and (ii) incurs regret $O(T^\beta)$.*

*Proof Sketch.* Algorithm GP-Enhanced-RDLP is $(\alpha, \delta)$-fair since it introduces a grace period every time that a decision transitions from acceptance to rejection and vice versa. To confirm that the regret of Algorithm GP-Enhanced-RDLP is $O(T^\beta)$, we have

$$\begin{aligned}\text{Regret} &= \text{Rev}^\star - \text{Rev}(\pi) = \text{Rev}^\star - \text{Rev}(\mathsf{RDLP\text{-}PA}) + \text{Rev}(\mathsf{RDLP\text{-}PA}) - \text{Rev}(\pi)\\ &= O(T^\beta) + \text{Rev}(\mathsf{RDLP\text{-}PA}) - \text{Rev}(\pi),\end{aligned}$$

where Rev(RDLP-PA), $\text{Rev}(\pi)$ are revenue generated by Algorithm RDLP-PA and Algorithm GP-Enhanced-RDLP respectively. It suffices to show that

$$\text{Rev}(\mathsf{RDLP\text{-}PA}) - \text{Rev}(\pi) = O(T^\beta)$$

Let $\tau$ be a random variable indicating the stopping time at which some resource becomes exhausted (triggering the termination of the algorithm), with $\tau$ belonging to time segment $k(\tau)$. Ignoring the grace period and accepting $y_i$ type $i$ customers in each segment, the expected revenue of Algorithm GP-Enhanced-RDLP almost matches that of Algorithm RDLP-PA for $t \in [0, (k(\tau) - 1)T^\beta]$.

Furthermore, although the application of both increasing and decreasing grace periods in each time segment may lead to the additional loss of up to $\Theta(\log T)$ customers of each type, we accept an equivalent number of customers of each type in the following time segment. By doing this, before $t^*$, the expected difference in the number of accepted customers of each type between Algorithm RDLP-PA and Algorithm GP-Enhanced-RDLP stems only from the last time segment before $t^*$, amounting to $\Theta(\log T)$. Therefore, the expected difference in remaining capacity between Algorithm RDLP-PA and Algorithm GP-Enhanced-RDLP is bounded by $\Theta(\log T)$ at time $t^*$. As such, the disparity in the re-solved optimal solutions at $t^*$ between Algorithm RDLP-PA and Algorithm GP-Enhanced-RDLP is negligible. Consequently, we eliminate all revenue loss due to the grace period for all $k(\tau) - 1$ time segments. For the time segment $k(\tau)$ of size $T^\beta$, we may lose a maximum of $O(T^\beta)$ in revenue compared to the RDLP-PA algorithm. Hence, we have

$$\text{Rev}(\mathsf{RDLP\text{-}PA}) - \text{Rev}(\pi) = O(T^\beta),$$

The full proof can be found in Appendix EC.4. Q.E.D.

**Algorithm 3:** GP-Enhanced-BL

**Input:** Preference matrix $\mathbf{A}$. Capacity vector $\mathbf{m}$. Booking limit $\mathbf{b}$. $\gamma = \log_{1-\alpha}\delta$.
**Output:** Decision on accepting customers over time.
**Initialize:** Number of accepted customers $s_i = 0$, $i \in [n]$.
**for** $t \in \{1, 2, ..., T\}$ **do**
  Observe customer of type $i \in [n]$.
  **if** $\min_{j\in[L]} m_j(t) - \bar{a}n\gamma \leq 0$ **then**
    Give a decreasing period $[t, T]$ to all types of customers.
  **else**
    **if** $s_i < b_i - \gamma$ **then**
      Accept the arriving customer, and set $s_i \leftarrow s_i + 1$.
    **else**
      Give a decreasing period $[t, T]$ to type $i$ customers.

### 4.2. Algorithm S-BPC

In the algorithm S-BPC, an FCFS approach is employed for each customer type. Drawing intuition from our treatment of FCFS, to guarantee fairness, we only need to offer a decreasing grace period to those types whose bid prices are greater than the aggregated bid prices. From Theorem 1, the extra revenue loss of the fair variant is $O(\log T)$ for any $\alpha \in (0, 1)$ and $\delta = 1/T$.

THEOREM 3. *Given any $\alpha \in (0, 1)$, $\delta = 1/T$, and any input S-BPC algorithm with a regret of $O(\sqrt{T})$, by giving a decreasing grace period $[\min_{j\in[L]} m_j - \bar{a}n\gamma, T]$ to each type $i$ customer (for $i$ such that $r_i > \sum_{j=1}^{L} \theta_j^* A_{ij}$): (i) the algorithm is $(\alpha, \delta)$-fair, (ii) incurs regret $O(\sqrt{T})$.*

## 5. Individually Fair RM Algorithms under Adversarial Arrivals

In this section, we focus on RM problems with adversarial arrivals, where an adversary dictates both the *quantity* and the *sequence* of arriving customers of each type. The two prevalent algorithm structures used in this setting are Booking Limit(BL) and NESTING. Next, we introduce how to make these algorithms individually fair.

### 5.1. Algorithm BL

Algorithm BL sets quotas for each customer $i$, and then makes decisions in an FCFS manner. Consequently, our method of ensuring individual fairness mirrors the modification of FCFS in Section 3. Algorithm GP-Enhanced-BL provides a decreasing grace period as the number of accepted type $i$ customers approaches the quota $b_i$, and extends this decreasing grace period to all customer types if the remaining capacity is almost depleted. Details can be found in Algorithm 3.

THEOREM 4. *Given any $\alpha \in (0, 1)$, $\delta = 1/T$, resource capacity $\mathbf{m}$ where $m_j = \Theta(m)$ for $j \in [L]$, and any input BL algorithm that attains a competitive ratio of $C$, GP-Enhanced-BL achieves the following: (i) is $(\alpha, \delta)$-fair, (ii) reaches a competitive ratio of $C - O(\log m/m)$.*

Theorem 4 implies that GP-Enhanced-BL incurs negligible additional loss asymptotically, even under the worst case instance. The detailed proof can be found in Appendix EC.6. To execute a competitive ratio analysis, we first observe that, given $b_i = \Theta(m)$, if the aggregate number of arrivals is $o(m)$, both the original and the revised booking limit algorithms yield a competitive ratio of 1 since all customers are accepted.

In the event that the total number of arrivals is $\Omega(m)$, the offline optimal revenue is $\Theta(m)$ as the resource capacity scales with $\Theta(m)$. Additionally, any extra loss incurred by GP-Enhanced-BL in comparison to BL is from the decreasing grace period. In the worst scenario, we may lose at most $(\bar{a}+1)n\gamma = \Theta(\log m)$ customers. Hence,

$$\inf_{I\in\mathcal{I}} \frac{\text{Rev}(\pi(I))}{\text{OPT}(I)} \geq \inf_{I\in\mathcal{I}} \frac{\text{Rev}(\mathsf{BL}(I)) - (\bar{a}+1)n\gamma\bar{r}}{\text{OPT}(I)} = C - O\left(\frac{\log m}{m}\right).$$

**Algorithm 4:** GP-Enhanced-NESTING

**Input:** Capacity $m$. Nesting quota $\mathbf{b}$.
**Output:** Decision on accepting customers over time.
**Initialize:** Number of accepted customers $s_i = 0$, $i \in [n]$. $\gamma = \log_{1-\alpha}\delta$.

```
for t ∈ {1,2,...,T} do
    Observe customer of type i ∈ [n].
    if m(t) − nγ ≤ 0 then
        Give a decreasing period [t,T] to all types of customers.
    else
        if Σ_{j=i}^{n} s_j < b_i − nγ then
            Accept the arriving customer, and set s_i ← s_i + 1.
        else
            Give a decreasing period [t,T] to type i customers.
```

### 5.2. Algorithm NESTING

NESTING ranks the customer types based on the revenue they bring to the supplier. Compared to BL, NESTING also sets quotas and accepts customers in a FCFS manner. Differently, each quota $b_i$ is the upper bound of number of customers can be accepted with type $i, i+1, \ldots, n$. As such, we still provide a decreasing grace period as the cumulative number of $i$, $i+1$, …, $n$ customers approaches the quota $b_i$. The rationale behind this modification aligns precisely with that used in BL. Details can be found in Algorithm 4.

THEOREM 5. *Given any $\alpha \in (0,1)$, $\delta = 1/T$, and any input NESTING algorithm that attains a competitive ratio of $C$, GP-Enhanced-NESTING: (i) is $(\alpha,\delta)$-fair, (ii) reaches a competitive ratio of $C - O(\log(m)/m)$.*

The proof is almost identical to that of Theorem 4 and hence we omit it.

## 6. Survey

In this section, we delineate the primary findings derived from our survey which was conducted between October and November 2023 on Prolific[5]. We collected responses from 150 respondents. The primary goal was to evaluate three hypotheses:

**H1.** When buyers observe "unfair" (defined according to our Definition 1) treatment in a store prior to shopping there, their purchasing intent diminishes.

**H2.** A customer who has previously made a purchase from a store and subsequently perceives unfair treatment is likely to exhibit adverse future buying behaviors.

**H3.** Individuals perceive disparate treatments occurring in the near term as more unfair compared to those in the distant past.

Comprehensive demographic data is available in Appendix EC.1.1. The respondents were all based in the United States, fluent in English, and mostly gender-balanced (about a half of them are identifying as male and about half as female). The majority identified as white. To ensure the quality of the responses, we incorporated three attention checks. Only the data from respondents who correctly answered at least two of these questions were considered valid. From the initial 150 participants, 140 responses met this criterion.

The goal of our survey was to assess individuals' buying behavior when they observed other customers receiving discounts of roughly 10% but they themselves did not. We recognize the significance of the specific discount value in this context; people's valuation of money varies, and a discount perceived as negligible by some might not induce feelings of unfairness to them. To address

[5] https://www.prolific.com/

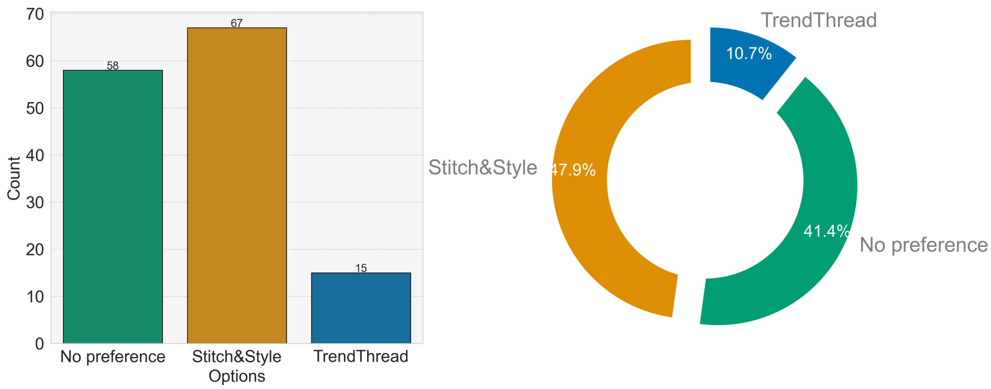


**Figure 2 Distribution of preferences in response to exposure to discount-related posts on social media**

this, we conducted a comparative survey, wherein the context incorporated a more substantial discount (specifically, 30%). This supplementary survey, which also comprised of 150 participants, is documented in Appendix EC.1.3. In all the scenarios we considered, we posit that there are two stores *TrendThread* and *Stitch&Style* which the survey respondent is considering. The products sold by the two places are *identical* in terms of quality and overall characteristics. The only difference is with regards to the pricing and the discounts that each stores offers.

### 6.1. Buying Behavior and Observed Pre-Purchased Unfairness

In this section, we seek to validate hypothesis **H1**. We begin by presenting two distinct scenarios designed to evoke feelings of unfairness *before* making a purchase among consumers.

The first scenario revolves around a consumer's interaction with social media posts describing others' purchasing experiences. Imagine you come across a social media post wherein a user claims to have a 10% discount on a hoodie from the store *TrendThread*. You promptly visit the online shopping website, but no such discount is extended to you from *TrendThread*. *Stitch&Style* is offering the identical hoodie at exactly the same price as *TrendThread*. We aim to capture whether consumers, upon encountering the absence of a promised discount at *TrendThread*, opt to transfer their patronage to *Stitch&Style* for their purchase[6].

As illustrated in Figure 2, 47.9% of the respondents perceive the different treatment as a significant factor influencing their purchasing decisions. Within the 41.5% who opted for *No Preference*, there emerges a question: is their indifference attributed to the perceived unfairness or the magnitude of the discount (10%)? In Figure 3, of those participants who initially selected *No Preference*, more than 20% would divert their purchase to an alternative store if faced with a more substantial discount disparity. From this observation, we infer that approximately 30% of the participants remain indifferent to differential treatment. In contrast, close to 60% would reconsider their store preference upon perceiving some form of unfairness prior to purchase.

The second scenario tests whether perceptions of pre-purchase unfairness can also come from customer reviews associated with a store. Consider two stores offering an identical hoodie at the same price and with equivalent ratings. The first store, *TrendThread*, displays one negative comment addressing discount unfairness among five positive comments[7]. Conversely, the second store, *Stitch&Style*, presents three negative comments to discount unfairness. The focus is to analyze how the volume of comments related to unfair treatments influences consumer shopping behavior.

As depicted in Figure 4, upon encountering few comments indicating unfair treatment, 28.6% of respondents express a positive inclination towards making a purchase, whereas 27.9% demonstrate

[6] A concise version is presented here for clarity, with the detailed questions available in Appendix EC.1.2

[7] An auxiliary analysis was conducted to discern whether the placement of this negative comment has any consequential impact. Refer to Appendix EC.1.2 for details.

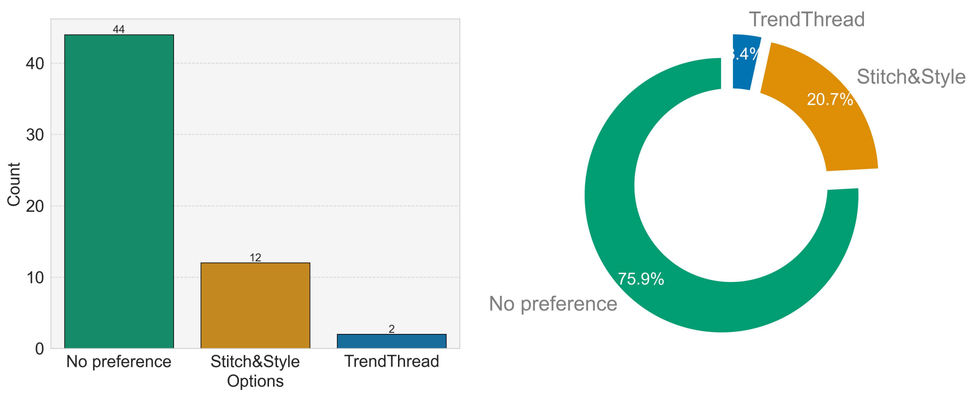


**Figure 3** **Impact of** $30\%$ **discount on initially indifferent consumers (i.e., "No preference" from Fig 2)**

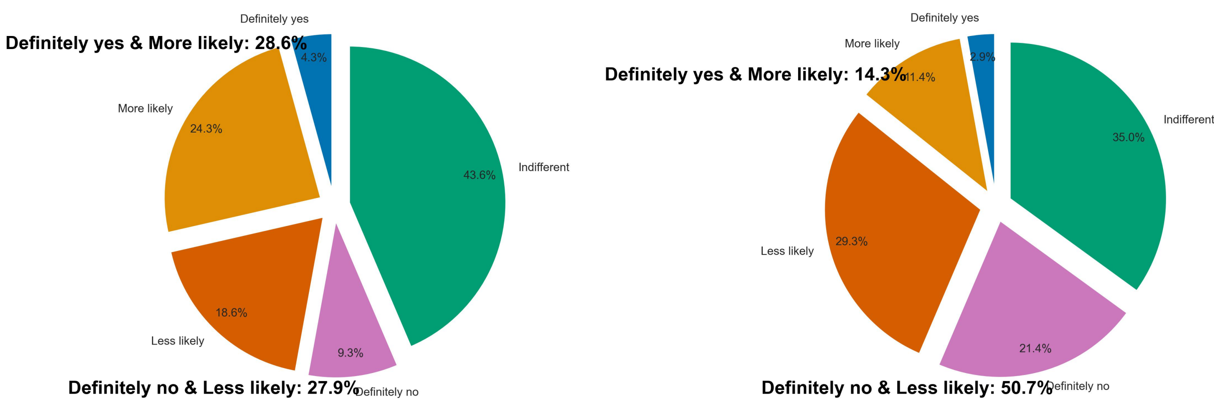


**Figure 4** **Left, Right Distribution of preferences for store TrendThread and Stitch&Style respectively.**

a negative inclination. However, in the presence of a *trend* of unfairness in the comments, the proportion of positive respondents diminishes to 14.3%, while those with a negative predisposition drastically surge to over 50%.

The findings from both scenarios underscore the observation that a pronounced perception of pre-purchase unfairness can significantly influence consumer behavior, leading many consumers to avoid shopping at the store they see as unfair. This verifies our hypothesis **H1**.

### 6.2. Buying Behavior and Observed Post-purchased Unfairness

Next, we test hypothesis **H2**. We propose a scenario where a customer, having made a purchase at the full price, subsequently learns of another individual who got a discount for the same item shortly thereafter, potentially triggering feelings of unfairness. To assess the repercussions of such a scenario on future consumer behavior, we initially investigate whether respondents perceive unfairness in this situation. Subsequently, we probe their likelihood of repeating purchases at the same store and their propensity to recommend the store to others.

In Figure 5, the left histogram depicts the perceived unfairness distribution, with 0 indicating no perceived unfairness and 1 denoting extreme unfairness. Note that in order to obtain this information, we asked respondents to choose how unfairly treated they feel, by moving a needle.

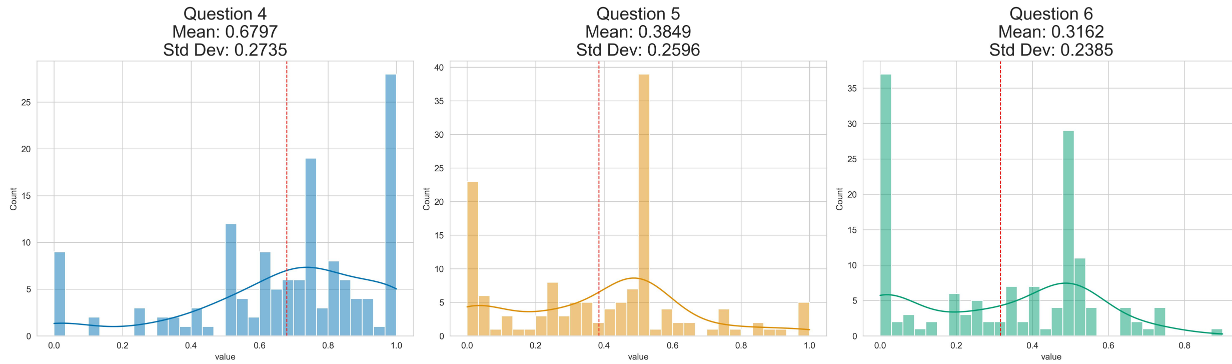


**Figure 5 Left: Perceived unfairness level (0 = Not at all, 1 = Extremely); Middle: Customer repurchase likelihood (0 = Definitely no, 1 = Definitely yes); Right: Customer recommendation likelihood (0 = Definitely no, 1 = Definitely yes)**

We ran the experiment with different initial placements of the needle, to abstract away from biases. For more details, we refer the reader to Appendix EC.1.

A significant number of respondents perceive high levels of unfairness, evidenced by an average response of approximately 0.68. This indicates that post-purchase differential treatment significantly influences perceptions of unfairness. The middle histogram illustrates respondents' willingness to repurchase from the store, where 0 represents a definite refusal and 1 a definite intention to repurchase. The mean response, around 0.38, suggests a reduced likelihood of repeat patronage due to observed disparities in post-purchase treatment. The right histogram, portrays the willingness to recommend the store with 0 indicating an unwillingness to recommend and 1 the opposite; it reveals a predominant reluctance to recommend among participants. This trend towards non-recommendation may adversely affect the store's potential revenue. These findings underline the importance of maintaining post-purchase individual fairness as a key factor influencing customer retention and advocacy, with direct implications for revenue streams.

### 6.3. Continuous Time Metrics of Individual Fairness

In this part, our objective is to validate hypothesis **H3**. This investigation also serves to corroborate the mathematical definition of individual fairness metrics presented in Definition 1. To assess whether individuals are more sensitive to recent disparities in treatment as opposed to those in the more distant past, we propose the following scenario: Imagine encountering two identical posts, each stating that a user got a 10% discount from *TrendThread*. The only distinction between these posts is their post time: one was created 5 minutes ago, while the other dates back to a month ago.

As illustrated in Figure 6, the predominant initial choice among respondents is the *5-minute* post, accounting for 33.6% of the selections. Incorporating those respondents who initially chose *Both* and subsequently indicated that the *5-minute* post evoked stronger feelings of unfairness, we find that 46.4% of participants perceive recent different treatments as more unfair than ones set further back in time. With this 46.4% surpassing the proportions of other available choices, our continuous metrics hypothesis is empirically validated.

## 7. Extension: Price-based Revenue Management

Price-based RM represents another significant class of RM problems Gallego and Van Ryzin (1994, 1997), Talluri and Van Ryzin (2004). In this paradigm, a seller markets one or multiple products over a finite horizon with a given initial resource inventory. Unlike the quantity-based RM, which determines whether to accept or reject an incoming customer, the decision in price-based RM involves setting a price for each arriving customer. Typically, we assume the existence of a bijective mapping between the price and the purchasing probability of each type of customer, whereby a higher price correlates with a lower purchasing probability. The objective is to maximize the total

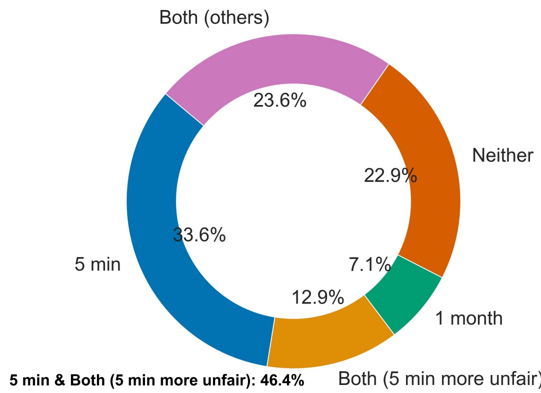


**Figure 6** **Distribution of opinions regarding which post makes the respondents feel more unfairly treated**

revenue over the given finite time horizon. We first define individual fairness for price-based RM, drawing upon a similar intuition to Definition 1, designed for quantity-based RM.

DEFINITION 6. Assume the algorithm $\mathcal{A}$ assigns a price $p_u^{(i)}$ to the $u^{\text{th}}$ customer of type $i$. An Algorithm $\mathcal{A}$ is $(\alpha, \delta)$-fair if for all $i \in [n]$, $u, v \in [n_i]$, where $n$ is the number of customer type and $n_i$ is the number of type $i$ customers who will arrive, with probability at least $1-\delta$, we have

$$\mathbb{P}(p_u^{(i)} \neq p_v^{(i)}) \leq \alpha |u - v|.$$

Traditional methods for solving price-based revenue management, for instance, those proposed by Gallego and Van Ryzin (1994, 1997), use the DLP-PA algorithm structure, while Jasin and Kumar (2012), Wang and Wang (2022) employs the RDLP-PA algorithm structure. These papers offer static or dynamic pricing strategies aimed at minimizing regret under a stochastic arrival setting. It is evident that neither static nor dynamic pricing strategies satisfy the individual fairness metrics as introduced in Definition 6. In the case of static pricing, as the capacity depletes, we assign a price of $+\infty$ to arriving customers. For dynamic programming, its inherent structure is entirely not individually fair.

First, we employ the concept of a decreasing grace period, adjusting it to suit the price-based context. Specifically, a decreasing grace period to type $i$ customer within the interval $[t_1, t_2]$ means that for every $u^{\text{th}}$ type $i$ customer arriving within the interval $[t_1, t_2]$, we assign them a price $p_{u-1}^{(i)}$ with a probability of $1-\alpha$, and a price of $+\infty$ with a probability of $\alpha$. This is analogous to the decreasing grace period for quantity-based RM.

Subsequently, we use the modification of the static pricing algorithm as an example to illustrate how to apply a decreasing grace period to guarantee individual fairness. A static pricing algorithm assigns a price $p^{(i)}$ to all type $i$ customers as long as sufficient resources exist, and assigns a price of $+\infty$ to all customers once resources are exhausted. The following theorem summarizes the results and the proof can be found in Appendix EC.7:

THEOREM 6. *For the price-based revenue management problem where the price $p^{(i)}$ is given to type $i$ customers, by assigning a decreasing grace period $[t_1(i), t_2(i)]$, where $t_1(i) = \inf\{t : \min_{j \in [L]} m_j(t) \leq \bar{a} n \gamma\}$ and $t_2(i) = T$ to each type $i$ customer, where $\gamma = \log_{1-\alpha} \delta$, the $(\alpha, \delta)$-fair metrics is satisfied. Moreover, compared to the original static pricing algorithm, the revenue loss is bounded by $\frac{\bar{a}}{\underline{a}} n \gamma \bar{p}$, where $\bar{p} = \sup_{i \in [n]} p^{(i)}$.*

## References


Shipra Agrawal, Zizhuo Wang, and Yinyu Ye. 2014. A dynamic near-optimal algorithm for online linear programming. *Operations Research* 62, 4 (2014), 876–890.

Mohammad Reza Aminian, Vahideh Manshadi, and Rad Niazadeh. 2023. Fair markovian search. *Available at SSRN 4347447* (2023).

Makis Arsenis and Robert Kleinberg. 2022. Individual Fairness in Prophet Inequalities. *arXiv preprint arXiv:2205.10302* (2022).

Santiago Balseiro, Christian Kroer, and Rachitesh Kumar. 2022. Single-leg revenue management with advice. *arXiv preprint arXiv:2202.10939* (2022).

Santiago Balseiro, Christian Kroer, and Rachitesh Kumar. 2023a. Online resource allocation under horizon uncertainty. *ACM SIGMETRICS Performance Evaluation Review* 51, 1 (2023), 63–64.

Santiago Balseiro, Haihao Lu, and Vahab Mirrokni. 2021. Regularized online allocation problems: Fairness and beyond. In *International Conference on Machine Learning*. PMLR, 630–639.

Santiago R Balseiro, Haihao Lu, and Vahab Mirrokni. 2023b. The best of many worlds: Dual mirror descent for online allocation problems. *Operations Research* 71, 1 (2023), 101–119.

Kristóf Bérczi, Laura Codazzi, Julian Golak, and Alexander Grigoriev. 2023. Envy-free dynamic pricing schemes. *arXiv preprint arXiv:2301.01529* (2023).

Semih Cayci, Swati Gupta, and Atilla Eryilmaz. 2020. Group-fair online allocation in continuous time. *Advances in Neural Information Processing Systems* 33 (2020), 13750–13761.

Kate Donahue and Jon Kleinberg. 2020. Fairness and utilization in allocating resources with uncertain demand. In *Proceedings of the 2020 conference on fairness, accountability, and transparency*. 658–668.

Benjamin D Douglas, Patrick J Ewell, and Markus Brauer. 2023. Data quality in online human-subjects research: Comparisons between MTurk, Prolific, CloudResearch, Qualtrics, and SONA. *Plos one* 18, 3 (2023), e0279720.

Guillermo Gallego, Anran Li, Van-Anh Truong, and Xinshang Wang. 2015. Online resource allocation with customer choice. *arXiv preprint arXiv:1511.01837* (2015).

Guillermo Gallego and Garrett Van Ryzin. 1994. Optimal dynamic pricing of inventories with stochastic demand over finite horizons. *Management science* 40, 8 (1994), 999–1020.

Guillermo Gallego and Garrett Van Ryzin. 1997. A multiproduct dynamic pricing problem and its applications to network yield management. *Operations research* 45, 1 (1997), 24–41.

Negin Golrezaei, Patrick Jaillet, and Zijie Zhou. 2023. Online Resource Allocation with Convex-set Machine-Learned Advice. *arXiv preprint arXiv:2306.12282* (2023).

Swati Gupta and Vijay Kamble. 2021. Individual fairness in hindsight. *The Journal of Machine Learning Research* 22, 1 (2021), 6386–6420.

Venkatesan Guruswami, Jason D Hartline, Anna R Karlin, David Kempe, Claire Kenyon, and Frank McSherry. 2005. On profit-maximizing envy-free pricing.. In *SODA*, Vol. 5. 1164–1173.

Tianming Huo and Wang Chi Cheung. 2022. Online reusable resource allocations with multi-class arrivals. *Available at SSRN 4320423* (2022).

Stefanus Jasin. 2014. Reoptimization and self-adjusting price control for network revenue management. *Operations Research* 62, 5 (2014), 1168–1178.

Stefanus Jasin and Sunil Kumar. 2012. A re-solving heuristic with bounded revenue loss for network revenue management with customer choice. *Mathematics of Operations Research* 37, 2 (2012), 313–345.

Jiashuo Jiang. 2023. Constant Approximation for Network Revenue Management with Markovian-Correlated Customer Arrivals. *arXiv preprint arXiv:2305.05829* (2023).

Jiashuo Jiang, Xiaocheng Li, and Jiawei Zhang. 2020. Online stochastic optimization with wasserstein based non-stationarity. *arXiv preprint arXiv:2012.06961* (2020).

Qian Liu and Garrett Van Ryzin. 2008. On the choice-based linear programming model for network revenue management. *Manufacturing & Service Operations Management* 10, 2 (2008), 288–310.

Will Ma, Pan Xu, and Yifan Xu. 2020. Group-level fairness maximization in online bipartite matching. *arXiv preprint arXiv:2011.13908* (2020).

Constantinos Maglaras and Joern Meissner. 2006. Dynamic pricing strategies for multiproduct revenue management problems. *Manufacturing & Service Operations Management* 8, 2 (2006), 136–148.

Vahideh Manshadi, Rad Niazadeh, and Scott Rodilitz. 2021. Fair dynamic rationing. In *Proceedings of the 22nd ACM Conference on Economics and Computation*. 694–695.

Martin I Reiman and Qiong Wang. 2008. An asymptotically optimal policy for a quantity-based network revenue management problem. *Mathematics of Operations Research* 33, 2 (2008), 257–282.

Sean R Sinclair, Siddhartha Banerjee, and Christina Lee Yu. 2022. Sequential fair allocation: Achieving the optimal envy-efficiency tradeoff curve. *ACM SIGMETRICS Performance Evaluation Review* 50, 1 (2022), 95–96.

Rui Sun, Xinshang Wang, and Zijie Zhou. 2020. Near-optimal primal-dual algorithms for quantity-based network revenue management. *arXiv preprint arXiv:2011.06327* (2020).

Kalyan Talluri and Garrett Van Ryzin. 1998. An analysis of bid-price controls for network revenue management. *Management science* 44, 11-part-1 (1998), 1577–1593.

Kalyan Talluri and Garrett Van Ryzin. 2004. Revenue management under a general discrete choice model of consumer behavior. *Management Science* 50, 1 (2004), 15–33.

Kalyan T Talluri, Garrett Van Ryzin, and Garrett Van Ryzin. 2004. *The theory and practice of revenue management*. Vol. 1. Springer.

Yining Wang and He Wang. 2022. Constant regret resolving heuristics for price-based revenue management. *Operations Research* 70, 6 (2022), 3538–3557.

Elizabeth Louise Williamson. 1992. *Airline network seat inventory control: Methodologies and revenue impacts*. Ph. D. Dissertation. Massachusetts Institute of Technology.

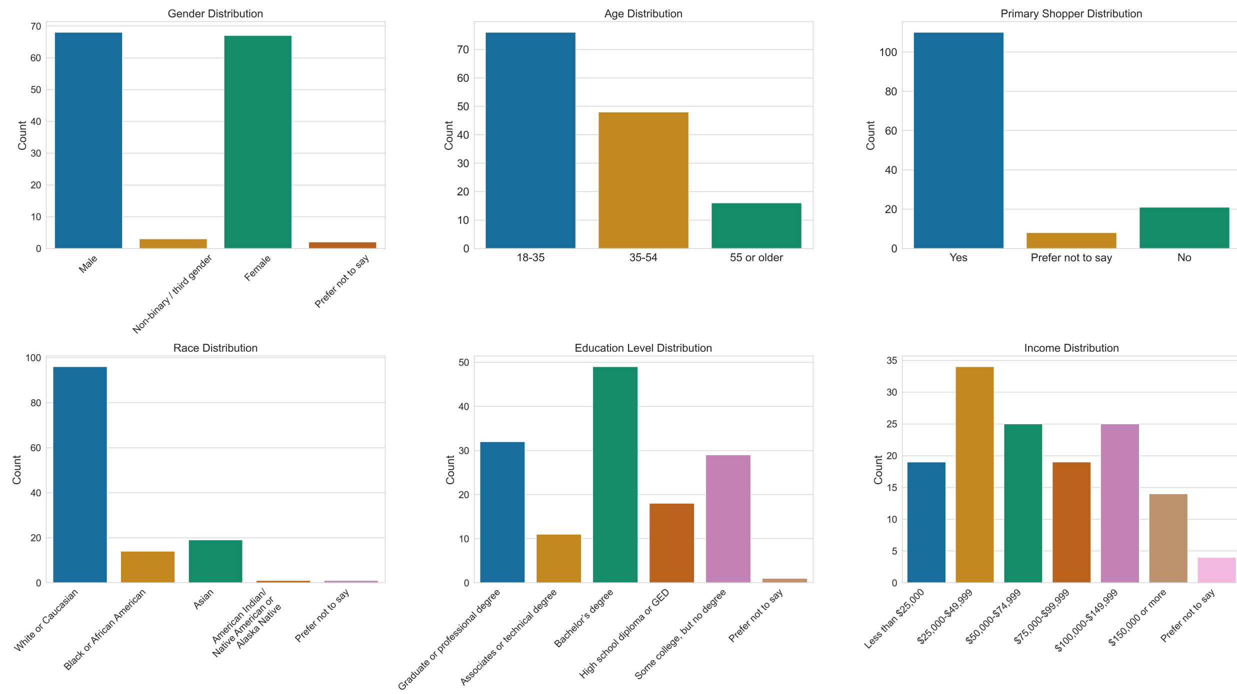


**Figure EC.1 Demographics of survey participants. The x axis corresponds to the categories for each plot and the y axis reports the number of participants per category**

# Online Appendix

## EC.1. Survey Appendix

### EC.1.1. Demographics

We conducted our survey through Prolific, ensuring gender balance—with an equal representation of male and female respondents—and fluency in English. Detailed demographic statistics are presented in Figure EC.1. The majority of our participants, over 50%, belong to the age group of 18-35 years. Approximately 30% are between 35 and 54 years old. Notably, more than 80% of the respondents identify as white or Caucasian. Our respondent population is representative of the Prolific respondent base (see e.g., Douglas et al. (2023)).

The respondents of our survey come from diverse educational backgrounds. About 60% have a bachelor's degree or have undertaken some form of graduate study. The rest have completed high school, earned an associate degree, or have some college education without obtaining a degree.

From an economic perspective, our sample spans various income brackets. Roughly 30% report an income exceeding $100K annually, while nearly 60% fall within the $25K-$100K income range. Additionally, over 85% of our participants identify as the primary shopper within their households.

### EC.1.2. Survey Questions

In this section, we display the questions and their respective distributions that were not introduced in Section 6. **'Question 1'**, aimed at verifying the authenticity of participants' shopping behavior, seeks to determine if respondents would, all other factors being equal, opt for a cheaper option. The related distribution, depicted in Figure EC.2, reveals that over 94% of participants chose the less expensive item, thereby indicating a general propensity for rational economic behavior.

**'Question 2'** and **'Question 3'** are aiming to test hypothesis **H1** mentioned in Section 6. In these questions, we examine a situation focusing on an individual's engagement with social media

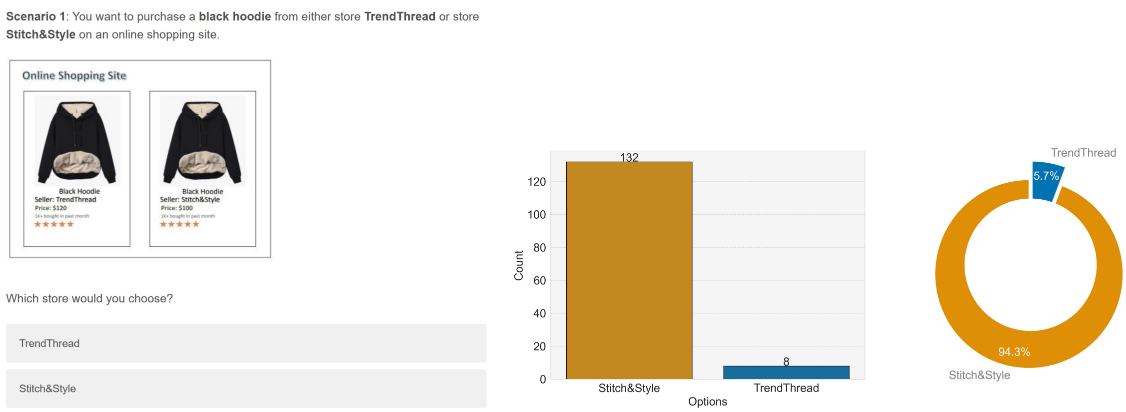


**Figure EC.2 Question 1 and its Answer Distribution**

content that depicts others' experiences with purchases. The detailed question can be found in Figure EC.3, and the distribution can be found in Figures 2 and 3.

**'Question 4'** still tests hypothesis **H1**. Contrasting the second and third questions, we highlight that pre-purchase unfairness can arise from customer reviews. We explore a scenario where two shops sell identical hoodies, each accompanied by varying review comments. The question can be found in Figure EC.4. Additionally, we sought to investigate whether the positioning of such comments influences consumer reactions and potentially introduces bias. To this end, we randomized the placement of comments from User 5, ensuring they appeared with equal frequency at any point from the first to the fifth position. The aggregated distribution is presented in Section 7, Figure 4. For a more detailed analysis on the order of comments, the distribution corresponding to each of the five cases is depicted in Figure EC.5.

In Figure EC.5, a significant bias is evident in the order of the comments. When comparing the same five comments, where the sole negative comment is placed last, the first diagram reveals a positive purchase incentive exceeding 44% and a negative purchase incentive of approximately 18%. However, if the negative comment is positioned at the beginning, the positive responses drop to merely 16%, while the negative responses surge to over 43%. This intriguing observation may open the door to numerous compelling research questions.

**'Question 5'**, **'Question 6'**, and **'Question 7'** are used to test hypothesis **H2** mentioned in Section 6. We suggest a situation in which a customer, who has bought an item at its full price, later discovers that another person obtained the same product at a discounted rate, possibly leading to a sense of perceived unfairness. The detailed questions can be found in Figure EC.6, and the distribution is in Figure 5.

An aspect of our investigation focused on the potential bias introduced by the initial positioning of the slider in three questionnaire items. To assess this, the sliders were initialized in four distinct scenarios, each distributed equally among participants:

- All sliders beginning at the midpoint (0.5).
- All sliders beginning at the maximum value (1).
- All sliders beginning at the minimum value (0).
- All sliders beginning at random positions.

The response distributions for each initialization scenario are illustrated in Figure EC.7, which suggests that there is no significant bias resulting from the initial location of the slider.

Finally, **'Question 8'** verifies hypothesis **H3**, which is to evaluate if people are more responsive to recent differences in treatment compared to those that occurred further back in time. The detailed question can be found in Figure EC.8, and the distribution has been mentioned in Figure 6.

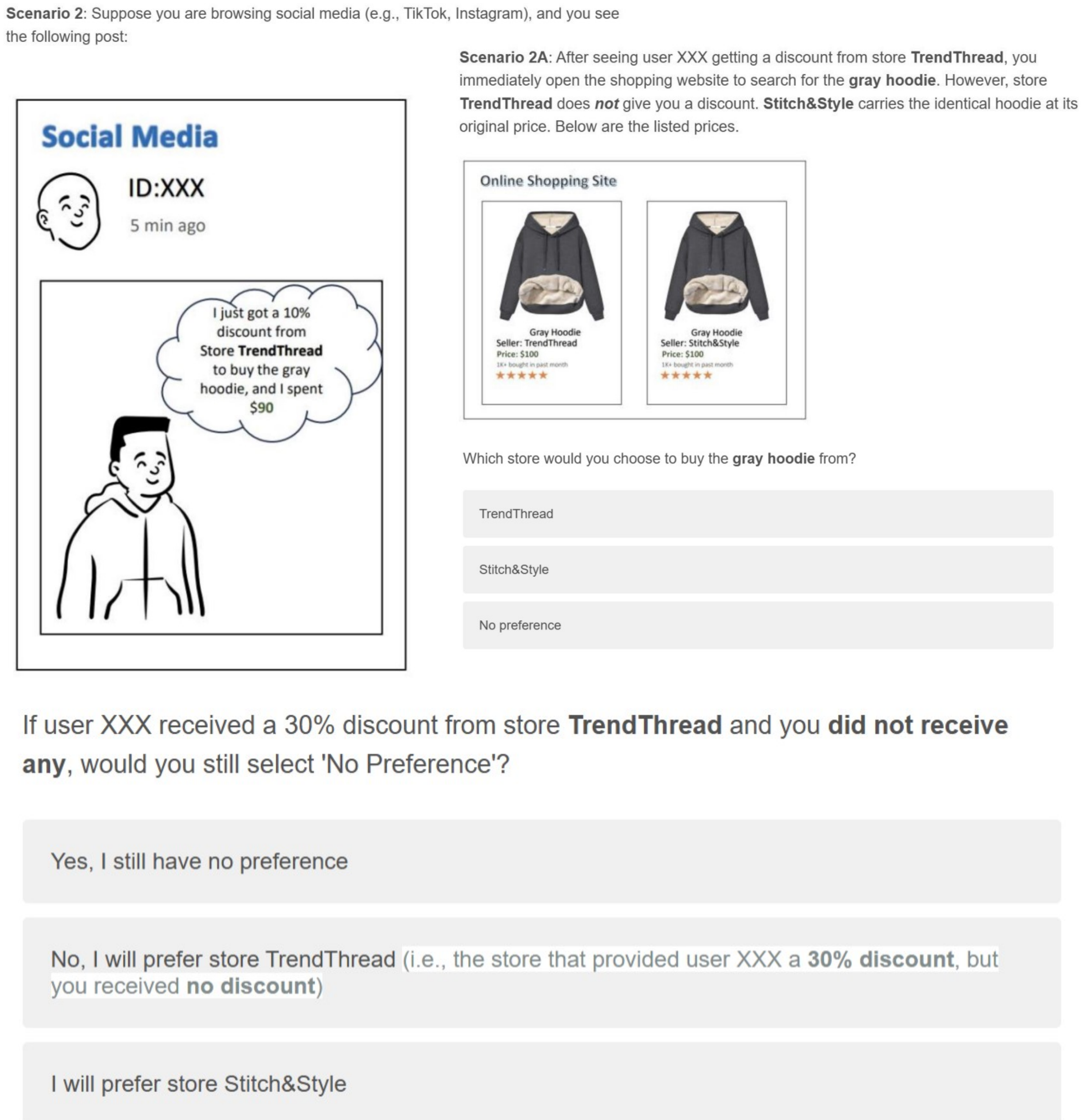


**Figure EC.3 Question 2 and Question 3**

### EC.1.3. 30% Discount Survey

We conducted a comparative survey, wherein the 10% discount is replaced by a 30% discount. In this section, we will show the distribution of the answers of Question 1 - 8.

Specifically, the distribution for **'Question 1'** can be found in Figure EC.9. **'Question 2'** is depicted in Figure EC.10, while **'Question 3'** is represented in Figure EC.11. The responses to **'Question 4'** are presented in Figure EC.12, and the distributions for **'Questions 5'** to **'Question 7'** are consolidated in Figure EC.13. Finally, the distribution for **'Question 8'** is available in Figure EC.14. Upon comparison with the data in Section 6, it appears that the magnitude of the discount offered does not significantly influence the distribution of responses to these questions.

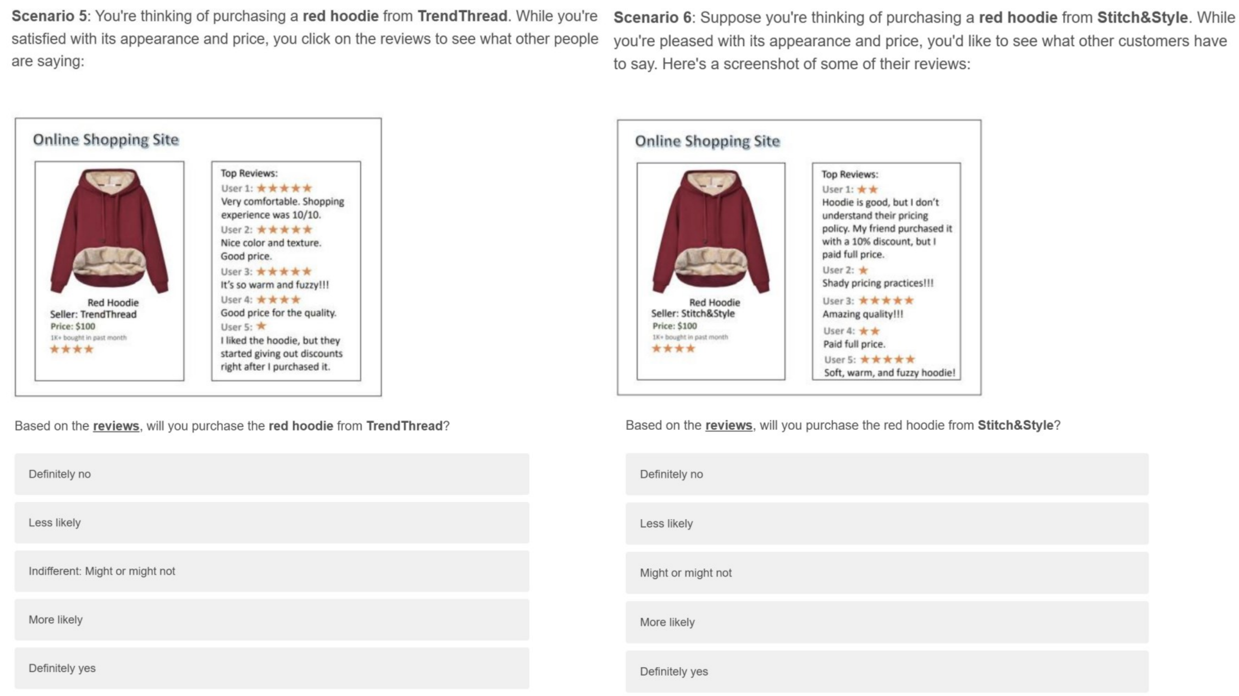


**Figure EC.4 Question 4**

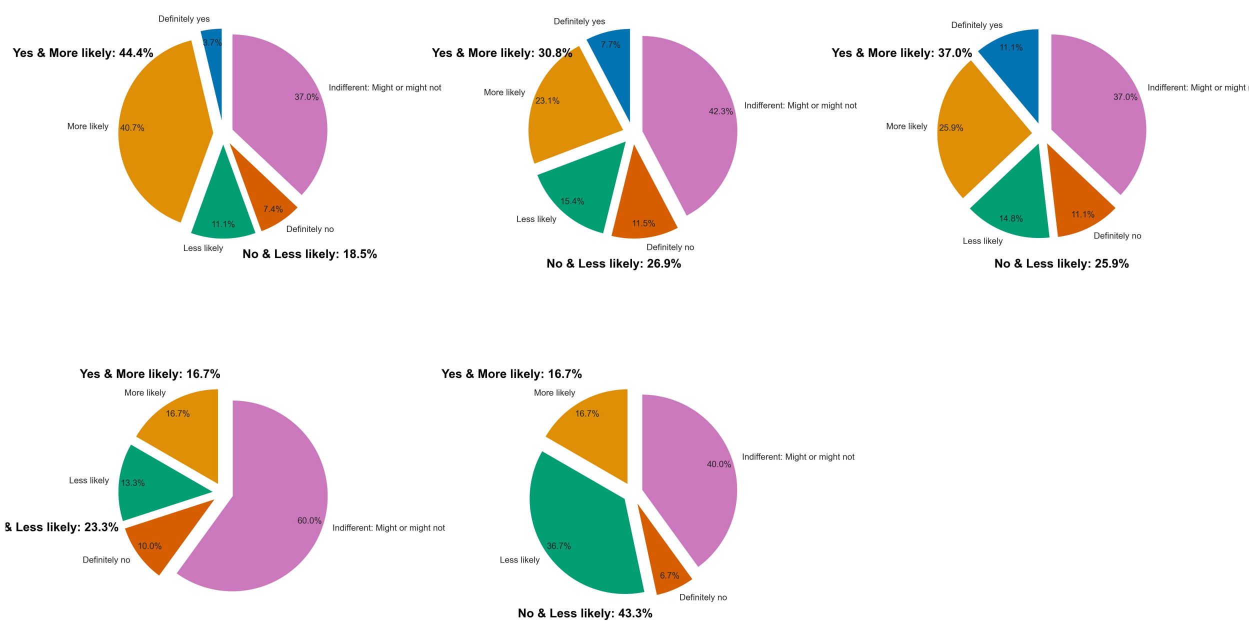


**Figure EC.5 Answer Distribution under Random Order of Comments**

# EC.2. Supplementary Material for Section 2

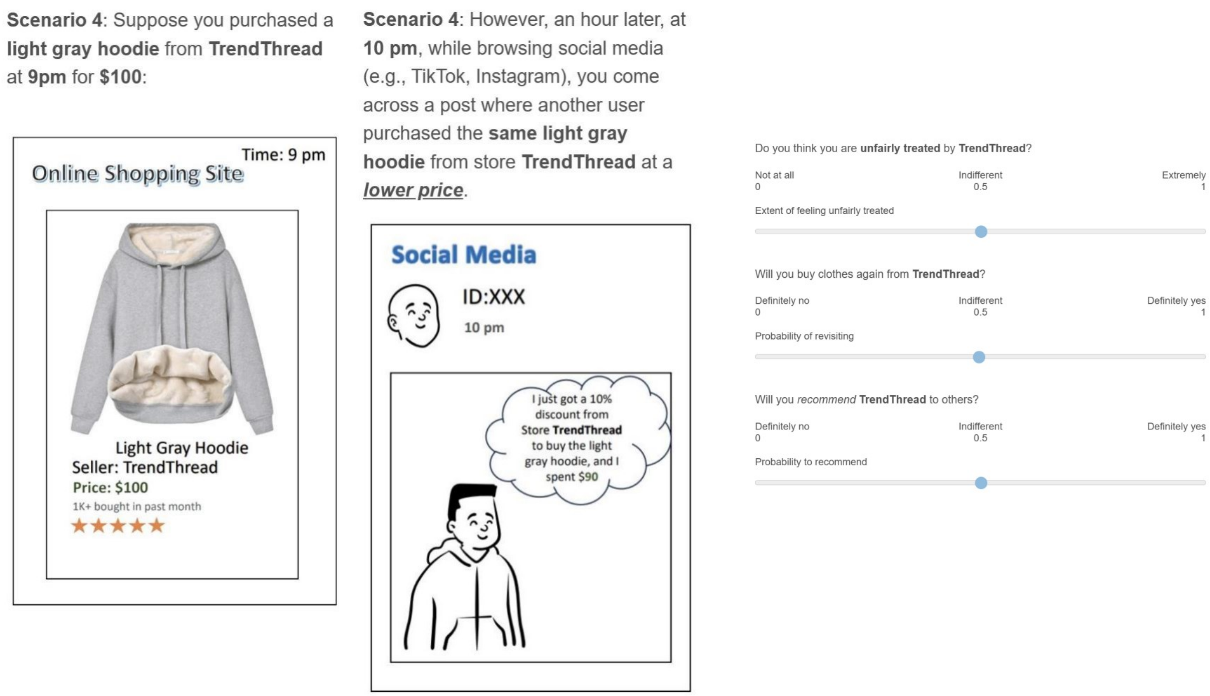


**Figure EC.6 Details of Questions 5-7**

---

**Algorithm 5:** DLP-PA

**Input:** Optimal primal variable $\mathbf{x}^*$. Preference matrix $\mathbf{A}$. Arriving rate vector $\lambda$. Capacity vector $\mathbf{m}$. Time horizon $[0,T]$.
**Output:** Acceptance decision for customers over time and update of capacity vector $\mathbf{m}$.
**for** $t \in \{1,2,...,T\}$ **do**
  Observe customer of type $i \in [n]$.
  **if** $\boldsymbol{A}_i \le \boldsymbol{m}(t)$ *for all* $i \in [n]$ **then**
    Accept customer with probability $\frac{x_i^*}{\lambda_i}$.
    **if** *customer is accepted* **then**
      $\mathbf{m}(t+1) \leftarrow \mathbf{m}(t) - \mathbf{A}_i$.
  **else**
    Reject the arriving customer and break.

---

**Algorithm 6:** RDLP-PA

**Input:** Preference matrix $\mathbf{A}$. Arriving rate vector $\lambda$. Capacity vector $\mathbf{m}$. Time horizon $[0,T]$. Resolving time point $t^*$.
**Output:** Updated optimal primal solutions $\mathbf{x}^*$ and $\tilde{\mathbf{x}}^*$ based on capacity adjustments at $t^*$.
At $t=0$, solve DLP based on the initial capacity. Denote the optimal primal solution as $\mathbf{x}^*$.
Run Algorithm 5 with initial capacity $\mathbf{m}$, horizon $[0,t^*]$, and optimal primal solution $\mathbf{x}^*$.
At $t=t^*$, solve DLP based on the remaining capacity. Let $\tilde{\mathbf{x}}^*$ be the optimal primal solution.
Run Algorithm 5 with remaining capacity, horizon $[t^*,T]$, and optimal primal solution $\tilde{\mathbf{x}}^*$.

---

## EC.3. Supplementary Material For Section 3

*Proof of Lemma 1* Let's focus on a type $i \in [n]$, to simplify the notation, we denote $p_u$ as the probability algorithm $\mathcal{A}$ accepts customer $i(u)$. For any $v > u$, we have $\mathbb{P}(i(v) \succ_{\mathcal{A}} i(u)) = p_v(1-p_u)$. We first show that for any $u < w < v$, we have $p_v(1-p_u) \le p_w(1-p_u) + p_v(1-p_w)$.

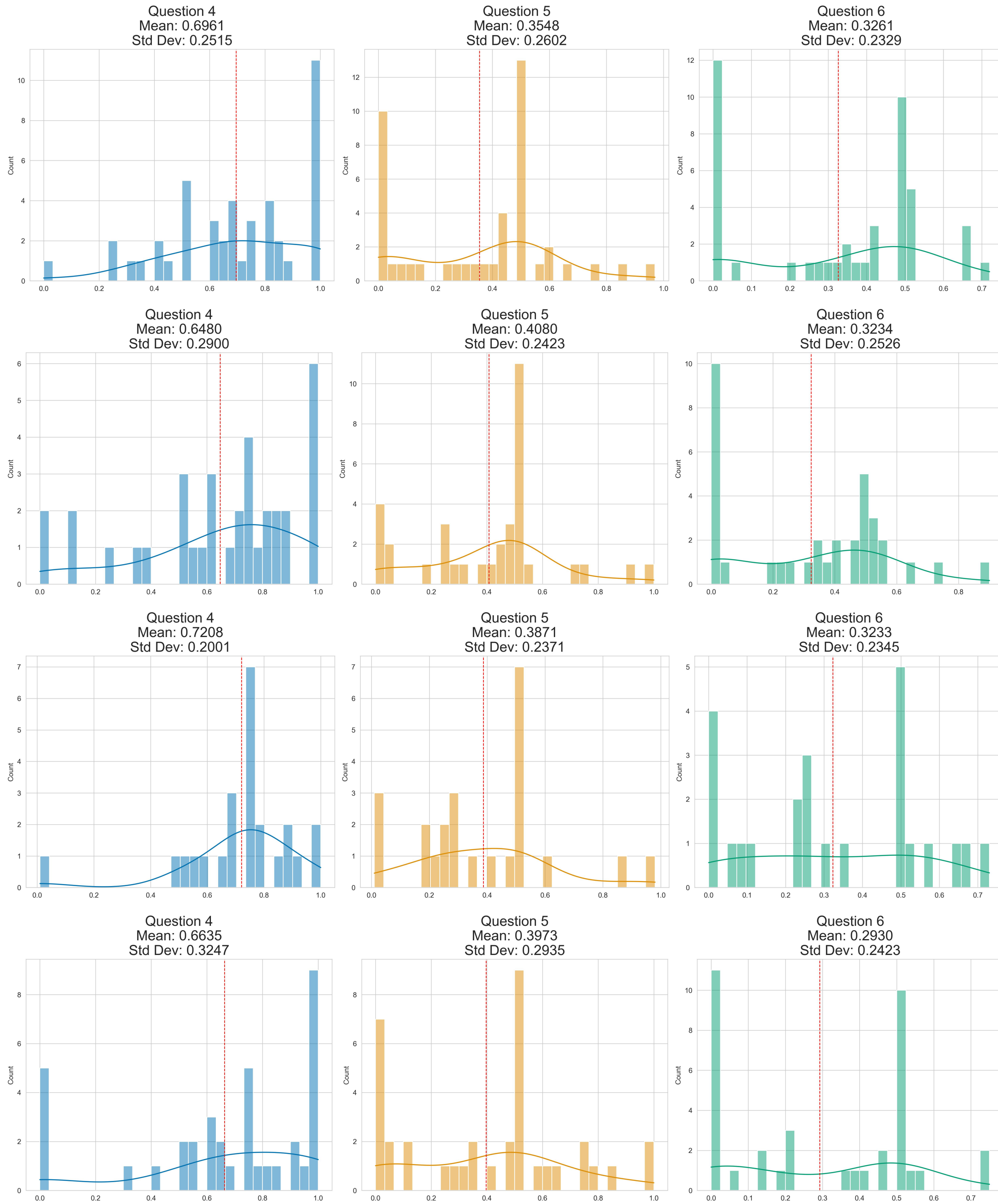


**Figure EC.7 Answer Distribution of Questions 5-7 with Different Starting Location of Sliders**

Observe that since $p_u, p_v \in [0,1]$, we have $(1-p_u)(1-p_v) \geq 0$. Then, we can obtain

$$
\begin{aligned}
& 0 \leq (1-p_u)(1-p_v) \\
\iff & p_u + p_v - 1 \leq p_u p_v
\end{aligned}
$$

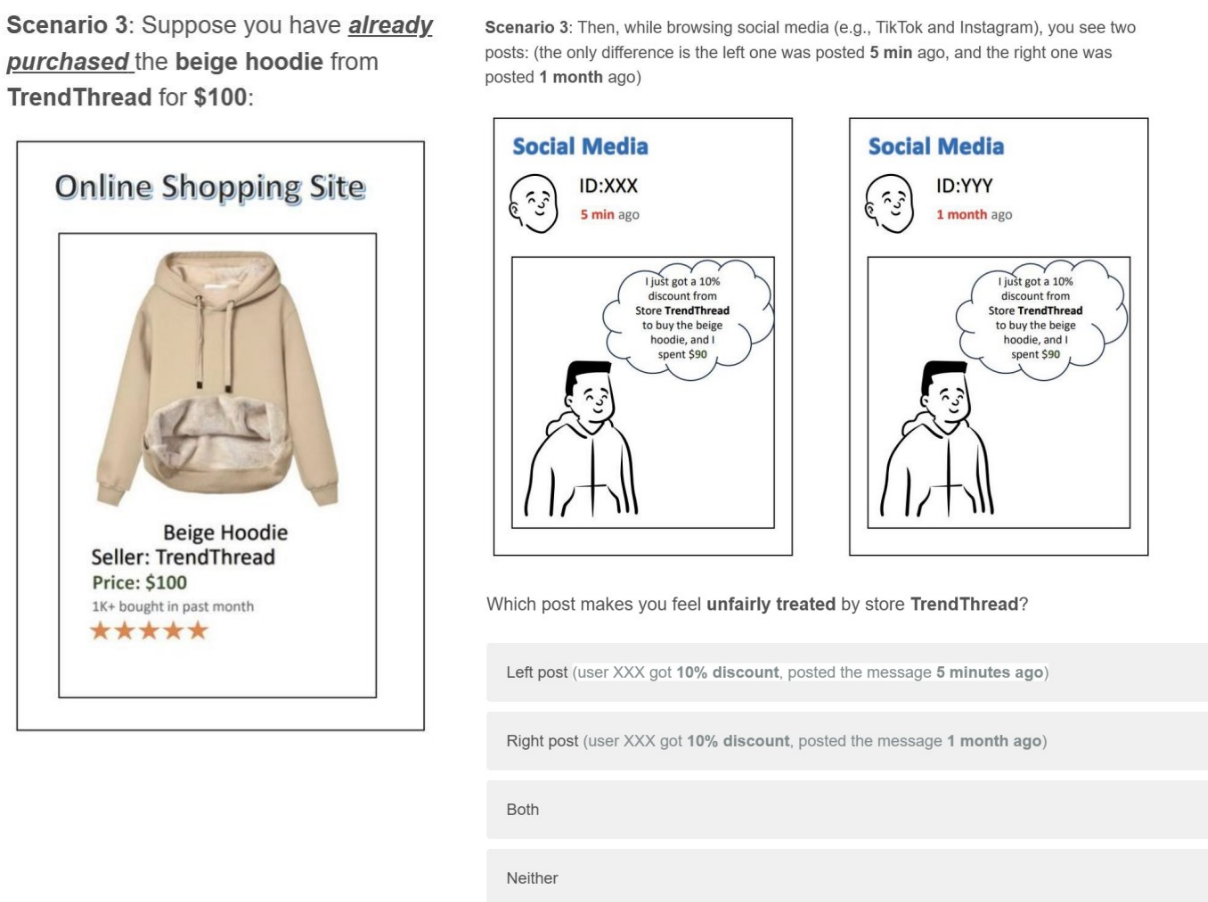


**Figure EC.8** Details of the Eighth Question

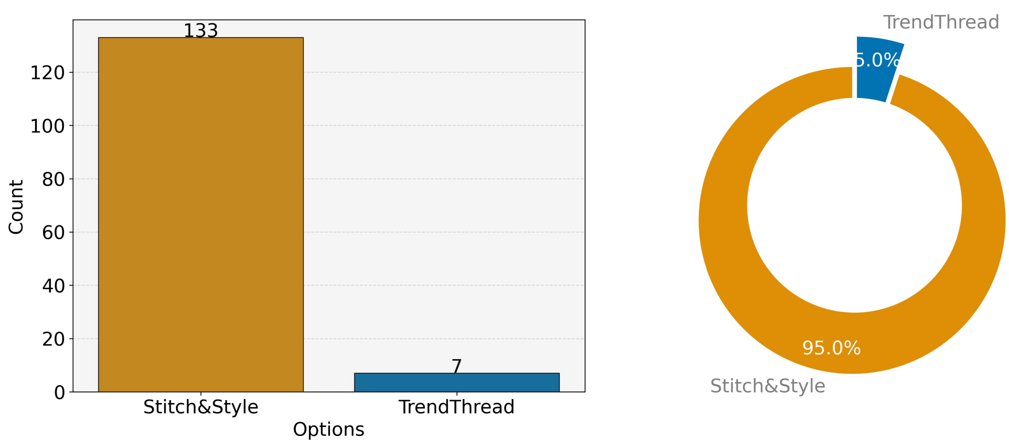


**Figure EC.9** Distribution of preferences among respondents of Question 1

$$
\begin{aligned}
&\Longrightarrow p_w(p_u + p_v - 1) \leq p_u p_v \qquad \text{(EC.1)}\\
&\Longleftrightarrow -p_u p_v \leq p_w - p_u p_w - p_w p_v\\
&\Longleftrightarrow p_v - p_u p_v \leq p_v + p_w - p_u p_w - p_w p_v\\
&\Longleftrightarrow p_v(1 - p_u) \leq p_w(1 - p_u) + p_v(1 - p_w),
\end{aligned}
$$

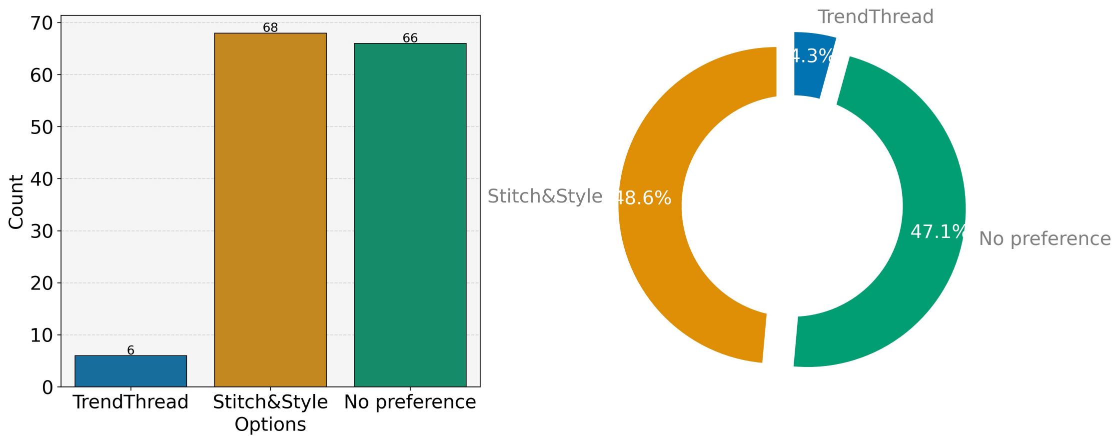


**Figure EC.10 Distribution of preferences among respondents of Question 2**

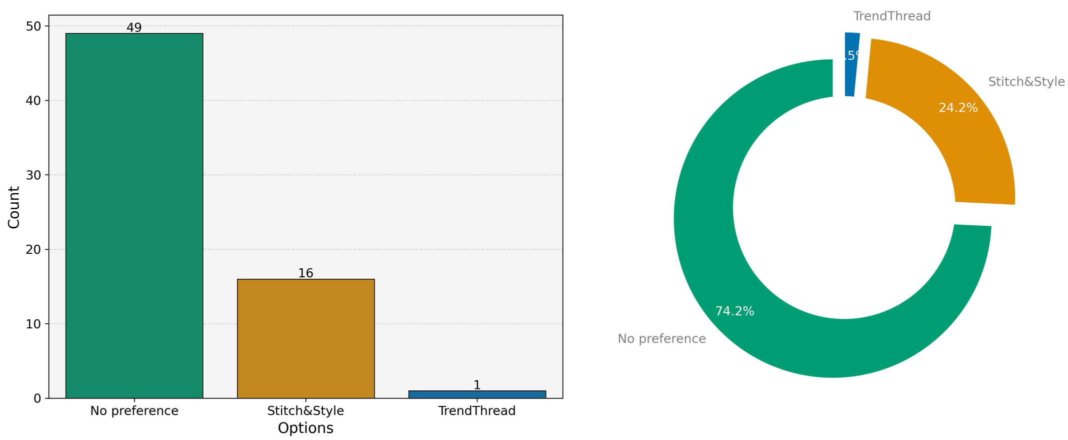


**Figure EC.11 Distribution of preferences among respondents of Question 3**

---

**Algorithm 7:** S-BPC

---

**Input:** Optimal dual variable $\theta^*$. Preference matrix $\mathbf{A}$. Reward vector $\mathbf{r}$. Capacity vector $\mathbf{m}$.

**Output:** Acceptance decision for customers over time and update of capacity vector $\mathbf{m}$.

**for** $t \in \{1, 2, ..., T\}$ **do**

  Observe customer of type $i$. Set $y_t \leftarrow \mathbf{1}\left(r_i > \sum_{j=1}^{L} \theta_j^* A_{ij}\right)$.

  **if** $\boldsymbol{A}_i \leq \boldsymbol{m}(t)$ *for all* $i \in [n]$ **then**

    **if** $y_t$ *equals to* 1 **then**

      Accept the arriving customer.

    **else**

      Reject the arriving customer.

    Set $\mathbf{m}(t+1) \leftarrow \mathbf{m}(t) - y_t A_i$.

  **else**

    Reject the arriving customer and break.

---

where (EC.1) is because $p_w \in [0,1]$. Therefore, for any $v > u$, we have

$$p_v(1-p_u) \leq \sum_{i=1}^{v-u} p_{u+i}(1-p_u) \leq \sum_{i=1}^{v-u} \alpha = \alpha(v-u).$$

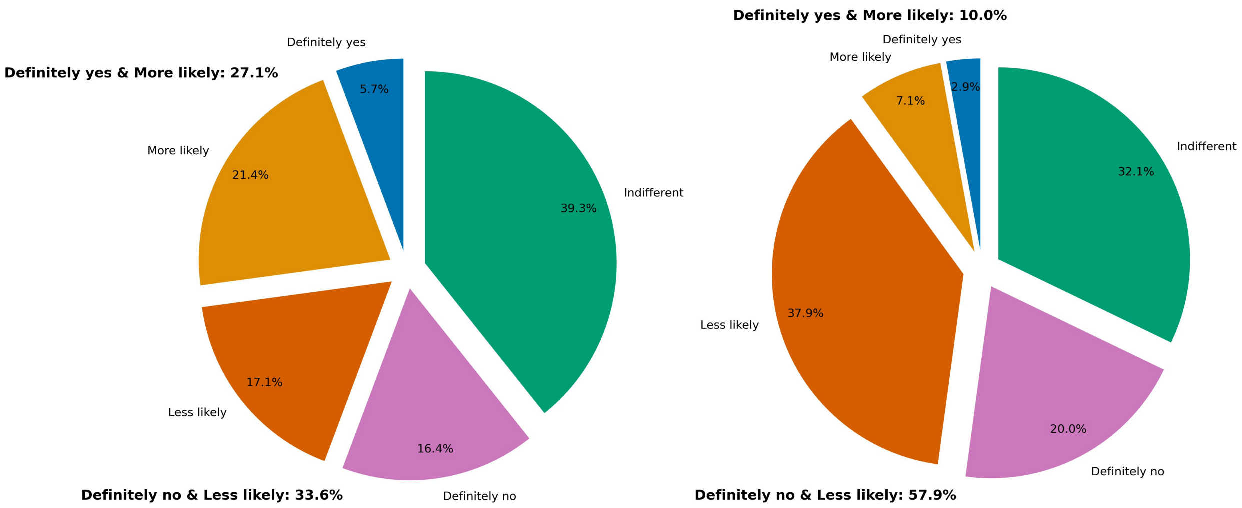


**Figure EC.12 Left: Distribution of preferences for store, TrendThread; Right: Distribution of preferences for store, Stitch&Style**

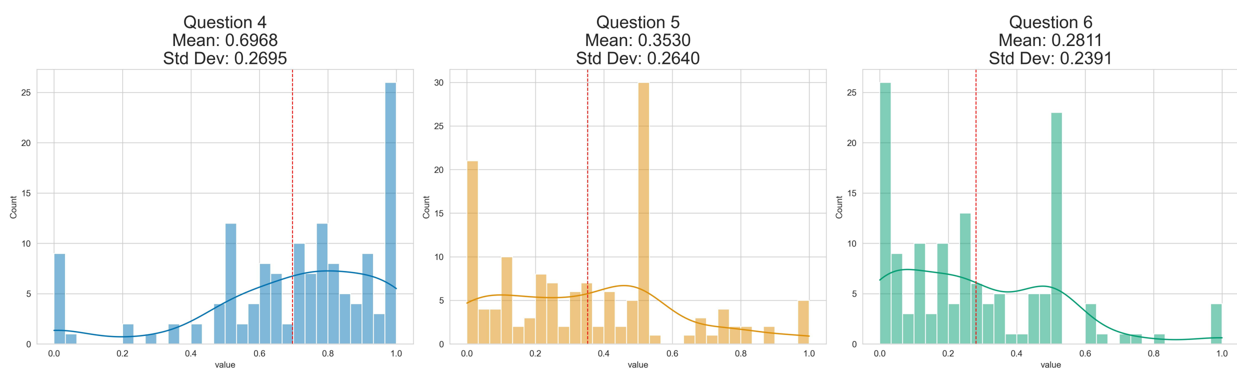


**Figure EC.13 Left: Distribution of preferences of Question 5; Middle: Distribution of preferences of Question 6; Right: Distribution of preferences of Question 7**

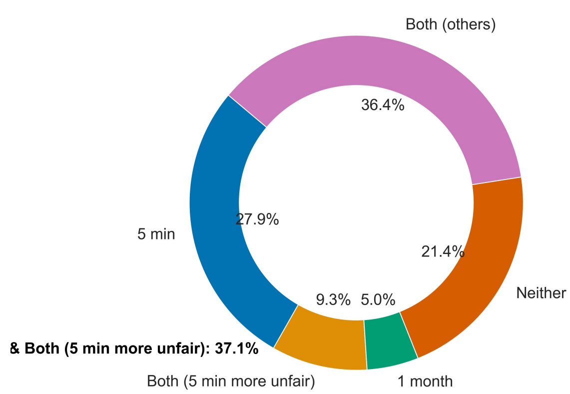


**Figure EC.14 Distribution of preferences among respondents of Question 8**

**Algorithm 8:** BL

**Input:** Preference matrix $\mathbf{A}$. Capacity vector $\mathbf{m}$. Booking limit $\mathbf{b}$.
**Output:** Acceptance decision for customers over time and update of capacity vector $\mathbf{m}$.
**Initialize:** Number of accepted customers $s_i = 0$, $i \in [n]$.
**for** $t \in \{1, 2, ..., T\}$ **do**
  Observe customer of type $i \in [n]$.
  **if** $\boldsymbol{A}_i \leq \boldsymbol{m}(t)$ *and* $s_i < b_i$ **then**
    Accept the arriving customer, and set $s_i \leftarrow s_i + 1$, $\mathbf{m}(t+1) \leftarrow \mathbf{m}(t) - \mathbf{A}_i$.
  **else**
    Reject the arriving customer.

**Algorithm 9:** NESTING

**Input:** Capacity $m$. Nesting quota $\mathbf{b}$.
**Output:** Acceptance decision for customers over time and update of capacity.
**Initialize:** Number of accepted customers $s_i = 0$, $i \in [n]$.
**for** $t \in \{1, 2, ..., T\}$ **do**
  Observe customer of type $i \in [n]$.
  **if** $m(t) \geq 1$ *and* $\sum_{j=i}^{n} s_j < b_i$ **then**
    Accept the arriving customer, and set $s_i \leftarrow s_i + 1$, $m(t+1) \leftarrow m(t) - 1$.
  **else**
    Reject the arriving customer.

A similar argument can be applied on the case $v < u$, and we omit the proof.

# EC.4. Supplementary material for Section 4

## EC.4.1. Proof of Theorem 2

[ Proof of Theorem 2] The proof of Theorem 2 is split into two parts: in the first one, we show that Algorithm 1 is $(\alpha,\delta)$-fair. In the second part, we show that Algorithm 1 has a regret of $O(T^{\beta})$.

*Part 1:* Define $\tau$ as the stopping time, where one of the resources depletes. Let $k(\tau)$ be the index of time segment containing $\tau$. Note that here both $\tau$ and $k(\tau)$ are random variables. For any type $i$ customer $i(u)$ arriving within $[1,(k(\tau)-1)T^{\beta}]$, by definition, both decreasing grace period and increasing grace period satisfy that $\mathbb{P}(i(u+1)\succ_{\mathcal{A}} i(u))\le\alpha$ and $\mathbb{P}(i(u-1)\succ_{\mathcal{A}} i(u))\le\alpha$ for any $u$. By Lemma 1, the $(\alpha,\delta)$-fair metrics is satisfied. For any customer $i(u)$ arriving within $[k(\tau)T^{\beta},T]$, Theorem 1 shows that Algorithm 1 is $(\alpha,\delta)$-fair. Note that Theorem 1 only proves the fairness of the decreasing grace period towards resource depletion. We can use a symmetric statement to prove the fairness of the increasing grace period towards resource depletion, and we omit the proof of this statement.

*Part 2:* To show the regret bound, we have

$$\begin{aligned}\text{Regret} &= \mathbb{E}[\text{Rev}^{\star}]-\text{Rev}(\pi) = (\mathbb{E}[\text{Rev}^{\star}]-\text{Rev}(\mathsf{RDLP\text{-}PA})) + (\text{Rev}(\mathsf{RDLP\text{-}PA})-\text{Rev}(\pi))\\ &= O(T^{\beta}) + (\text{Rev}(\mathsf{RDLP\text{-}PA})-\text{Rev}(\pi)).\end{aligned}$$

Next, we upper bound the value of $\text{Rev}(\mathsf{RDLP\text{-}PA})-\text{Rev}(\pi)$.

Define $\tau$ as the stopping time, where one of the resources depletes, and $t^*$ as the re-solving time point. Let $k(\tau)$, $k(t^*)$ be the index of time segment containing $\tau$, $t^*$, respectively. For any arriving instance $I$, for the time segment $1,2,\dots,k(\tau)-1$, we make the following coupling between R-DLP and GP Enhanced R-DLP: let $u_i(k)$ be the number of type $i$ customers accepted by DLP in the time segment $k$. We have $u_i(k)\sim\text{Bin}(\lambda_i\frac{t_2-t_1}{K},\frac{x_i^*}{\lambda_i})$. Then, we couple $y_i$ in time segment $k$ to have the value $u_i(k)$. We can make this coupling because $y_i$ in time segment $k$ and $u_i(k)$ follow the same distribution. Let $w_i(k)$ be the number of rejected type $i$ customers due to the grace period, and let $Y_i(k)$ be the realization of total number of accepted type $i$ customers in time segment. For any instance $\mathcal{I}$, any customer type $i$, and any time segment index $s\in[k(t^*)-1]$, we have

$$\sum_{k=1}^{s} u_i(k)-Y_i(k) = \sum_{k=1}^{s} u_i(k)-(y_i-w_i(k)+w_i(k-1)) = w_i(s)\le 2\frac{\bar{a}}{\underline{a}}\gamma, \tag{EC.2}$$

which implies that

$$\begin{aligned}&\text{Rev}(\mathsf{RDLP\text{-}PA})(1,(k(t^*)-1)T^{1/3})-\text{Rev}(\pi)(1,(k(t^*)-1)T^{\beta})\\ &=\sum_{k=1}^{k(t^*)-1}\left(\text{Rev}(\mathsf{RDLP\text{-}PA})\left((k-1)T^{\beta}+1,kT^{\beta}\right)-\text{Rev}(\pi)\left((k-1)T^{\beta}+1,kT^{\beta}\right)\right)\\ &=\sum_{i=1}^{n} w_i(k(t^*)-1)\le 2\frac{\bar{a}}{\underline{a}}n\gamma\bar{r}.\end{aligned}\tag{EC.3}$$

Next, let $\tilde{\mathbf{x}}^*$, $\tilde{\mathbf{y}}^*$ be the re-solved optimal solution by Algorithm RDLP-PA and Algorithm GP-Enhanced-RDLP at time $t^*$ respectively. By Equation (EC.2), the difference of remaining capacity Algorithm RDLP-PA and Algorithm GP-Enhanced-RDLP at time $t^*$ is at most $O(\log T)$. Therefore,

$$|\tilde{\mathbf{x}}^*-\tilde{\mathbf{y}}^*| = O\left(\frac{\log T}{T}\right).$$

For any instance $\mathcal{I}$, any customer type $i$, and any time segment index $s\in\{k(t^*),k(t^*)+1,\dots,k(\tau)-1\}$, we have

$$\sum_{k=k(t^*)}^{s} u_i(k)-Y_i(k) = \sum_{k=k(t^*)}^{s} u_i(k)-\left(y_i-w_i(k)+w_i(k-1)+O\left(\frac{\log T}{T}\right)\right)$$

$$= w_i(s) + (s - k(t^*))O\left(\frac{\log T}{T}\right)$$
$$\leq 2\frac{\bar{a}}{\underline{a}}\gamma + O(\log T), \tag{EC.4}$$

which implies that

$$\text{Rev}(\mathsf{RDLP\text{-}PA})(k(t^*)T^\beta), (k(\tau)-1)T^\beta) - \text{Rev}(\pi)(k(t^*)T^\beta), (k(\tau)-1)T^\beta)$$
$$= \sum_{k=k(t^*)}^{k(\tau)-1} \left(\text{Rev}(\mathsf{RDLP\text{-}PA})\left((k-1)T^\beta + 1, kT^\beta\right) - \text{Rev}(\pi)\left((k-1)T^\beta + 1, kT^\beta\right)\right)$$
$$= \sum_{i=1}^{n} w_i(k(\tau)-1) + O(\log T) \leq 2\frac{\bar{a}}{\underline{a}} n\gamma\bar{r} + O(\log T). \tag{EC.5}$$

As we can have at most $2T^\beta$ loss in the time segment $k(t^*)$ and $k(\tau)$, by Equations (EC.3) and (EC.5), we can obtain:

$$\text{Rev}(\mathsf{RDLP\text{-}PA}) - \text{Rev}(\pi) = 2T^\beta + O(\log T).$$

## EC.5. Supplementary Material for BPC-OGD

Balseiro et al. (2023b), Sun et al. (2020) propose a dynamic bid price control using first-order online optimization methods: online gradient descent or online mirror descent (BPC-OGD). Compared to S-BPC, BPC-OGD also incurs regret $O(\sqrt{T})$ under stochastic arrivals, yet it offers two advantages: (i) the algorithm can be applied even when the arrival processes $\lambda_i$ are entirely unknown; (ii) it does not require solving any primal or dual LP. The details of BPC-OGD can be found in Algorithm 10 below.

---

**Algorithm 10:** BPC-OGD

---

**Input:** Preference matrix $\mathbf{A}$. Reward vector $\mathbf{r}$. Capacity vector $\mathbf{m}$. OGD Parameter: $G$, $D$, $\bar{\theta}$.
**Output:** Update of dual variables $\theta$ over time.
**Initialize:** Dual variable $\theta^{(0)} \leftarrow \mathbf{0}$.
**for** $t = 1, 2, \dots, T$ **do**
  Observe customer of type $i$. Set $y_t \leftarrow \mathbf{1}\left(r_i > \sum_{j=1}^L \theta_j^{(t)} A_{ij}\right)$.
  **if** $\boldsymbol{A}_i \leq \boldsymbol{m}(t)$ *for all* $i \in [n]$ **then**
    **if** $y_t$ *equals to* 1 **then**
      Accept the arriving customer.
    **else**
      Reject the arriving customer.
    Set $\mathbf{m}(t+1) \leftarrow \mathbf{m}(t) - y_t A_i$.
  **else**
    Reject the arriving customer and break.
  Construct function $g_t(\theta) = \sum_{j=1}^L \theta_j \left(\frac{m_j(t)}{T} - y_t A_{ij}\right)$.
  Update the dual variables using the OGD procedure:
    $\eta_t \leftarrow \frac{D}{G\sqrt{T}}$,
    $\theta^{(t+1)} \leftarrow \theta^{(t)} - \eta_t \nabla_\theta g_t(\theta^{(t)})$,
    $\theta_j^{(t+1)} \leftarrow \min\left(\max\left(0, \theta_j^{(t+1)}\right), \bar{\theta}\right)$ for all $i \in [m]$.

---

---

**Algorithm 11:** GP-Enhanced-BPC-OGD

**Input:** Preference matrix $\mathbf{A}$. Reward vector $\mathbf{r}$. Capacity vector $\mathbf{m}$. OGD Parameter: $G$, $D$, $\bar{\theta}$. Fairness parameters $(\alpha, \delta)$.

**Output:** Update of dual variable $\theta$ and acceptance decisions over time.

**Initialize:** Dual variable $\theta^{(0)} \leftarrow \mathbf{0}$.

Reject all customers arriving between $t \in [0, T^{2/3}]$.

Denote $\Lambda_i$ as the number of arriving customers of type $i$ between $t \in [0, T^{2/3}]$, $i \in [n]$.

Run Algorithm BPC-OGD as an auxiliary algorithm for $t \in [0, T^{2/3}]$.

Denote $u_i$ as the number of accepted customers of type $i$ by BPC-OGD, $i \in [n]$.

Set $K = T^{1/3} - 1$, $\lambda_i = \frac{\Lambda_i}{T^{2/3}}$ and $x_i^* = \frac{u_i}{T^{2/3}}$, $\mathbf{z}^{(0)} = \mathbf{0}$.

**for** $k \in \{1, 2, ..., K\}$ **do**

  Run Algorithm 2 with $t_1 = kT^{2/3}$, $t_2 = (k+1)T^{2/3}$, $\mathbf{z} = \mathbf{z}^{(k-1)}$, optimal primal solution $\mathbf{x}^*$. Denote the output vector $\mathbf{w}$ as $\mathbf{z}^{(k)}$.

---

BPC-OGD can be markedly unfair. To see this, let us simplify the setting and assume that there is only one type of customer for $t = 1, 2, \ldots, T$. Also assume that there is only one type of resource and each arriving customer requests a single unit of this resource. Subsequently, if we follow steps 15 and 16 in Algorithm 10, we find that $\theta^{(t)}$ is a deterministic process that can be described as per the following formula:

$$\theta^{(t+1)} = \begin{cases} \theta^{(t)} - \frac{D}{G\sqrt{T}} \frac{m(t)}{T} + \frac{D}{G\sqrt{T}}, & \text{if } \theta^{(t)} < r \ , \\ \theta^{(t)} - \frac{D}{G\sqrt{T}} \frac{m(t)}{T}, & \text{if } \theta^{(t)} \geq r \ , \end{cases} \tag{EC.6}$$

Here, $r \in (0,1)$, $m(t)$ denotes the number of resources remaining at time $t$, with $m(0) = m$, and

$$m(t+1) = \begin{cases} m(t) - 1, & \text{if } \theta^{(t)} < r \ , \\ m(t), & \text{if } \theta^{(t)} \geq r \ , \end{cases}$$

The deterministic process $\theta^{(t)}$, as defined in Equation (EC.6), allows for the precise computation of how frequently decisions switch between accepting and rejecting adjacent customers. Regardless of the constants $D$, $G$, and $r$ (where $r$ lies between 0 and 1), this decision-making flips $\Theta(T)$ times. This behavior highlights the algorithm's significant unfairness, as it implies that almost every customer is treated differently compared to their immediate predecessor or successor.

### EC.5.1. Algorithm BPC-OGD

To modify BPC-OGD to comply with $(\alpha, \delta)$-fairness while minimally impacting revenue, we first reject all arrivals in the period $t \in [0, T^{2/3}]$. Meanwhile, we operate BPC-OGD as an auxiliary algorithm, documenting its decisions. We designate $u_i$ as the number of accepted type $i$ customers by BPC-OGD within the $[0, T^{2/3}]$ time horizon, and $\Lambda_i$ as the count of arriving type $i$ customers within $[0, T^{2/3}]$. Then, under mild non-degeneracy assumptions (as per assumption 3 in Agrawal et al. (2014)), we obtain that the Euclidean distance between the rate of accepted type $i$ customers by BPC-OGD ($\frac{u_i}{\Lambda_i}$), and that by DLP-PA ($\frac{x_i^*}{\lambda_i}$), is bounded with high probability. Thus, even without access to the $\lambda_i$ arrival rate and with the prohibition of linear programming, we can utilize the initial $T^{2/3}$ periods to approximate the optimal DLP solution $\mathbf{x}^*$. Finally, for $t \in [T^{2/3}, T]$, we execute the GP-Enhanced-RDLP Algorithm with a $T^{2/3}$ time segment length, setting $x_i^* = \frac{u_i}{T^{2/3}}$ and $\lambda_i = \frac{\Lambda_i}{T^{2/3}}$, culminating in a $(\alpha, \delta)$-fair algorithm with regret $O(T^{2/3} \log T)$. Full details can be found in Algorithm 11, and the following theorem summarizes the results.

Theorem EC.1. *Given any $\alpha \in (0,1)$, $\delta = 1/T$, and any input BPC-OGD algorithm that attains a regret of $O(\sqrt{T})$, under a mild non-degenerate assumption (assumption 3 in Agrawal et al. (2014)), Algorithm GP-Enhanced-BPC-OGD: (i) is $(\alpha, \delta)$-fair, (ii) incurs regret $O(T^{2/3} \log T)$.*

*Proof of Theorem EC.1* First, in the period $(0, T^{2/3})$, Algorithm GP-Enhanced-BPC-OGD rejects all customers, which is obviously $(\alpha,\delta)$-fair. As is shown in Theorem 2, Algorithm RDLP-PA satisfies $(\alpha,\delta)$-fair metrics, we can obtain that Algorithm GP-Enhanced-BPC-OGD is $(\alpha,\delta)$-fair.

Secondly, to address the regret bound, we observe that the regret bound can be formulated as

$$\begin{aligned}\text{Regret} &= \mathbb{E}[\text{Rev}^\star] - \text{Rev}(\pi) = \\ &= \Big(\mathbb{E}\big[\text{Rev}^\star(0,T^{2/3})\big] - \text{Rev}(\pi)(0,T^{2/3})\Big) + \Big(\mathbb{E}\big[\text{Rev}^\star(T^{2/3},T)\big] - \text{Rev}(\pi)(T^{2/3},T)\Big), \end{aligned} \tag{EC.7}$$

where $\text{Rev}(\pi)$ is revenue generated by Algorithm GP-Enhanced-BPC-OGD.

Upon rejecting all customers within $[0, T^{2/3}]$, the maximum loss incurred is $T^{2/3}$ during the initial $T^{2/3}$ periods, which implies that the first term in Equation (EC.7) is bounded by $T^{2/3}$. To bound the second term, we show that the Euclidean distance between the rate of accepted type $i$ customers by BPC-OGD ($\frac{u_i}{\Lambda_i}$), and that by DLP ($\frac{x_i^*}{\lambda_i}$) is close. Let the expected total revenue generated by BPC-OGD between $(t_1, t_2)$ be $\text{Rev}(\text{BPC-OGD})(t_1,t_2)$. By Balseiro et al. (2023b), we can obtain that BPC-OGD has the regret bound of $O(\sqrt{T})$, which implies that for $t \in [0, T^{2/3}]$,

$$\text{Rev}^\star(0,T^{2/3}) - \text{Rev}(\text{BPC-OGD})(0,T^{2/3}) = \text{Rev}^\star(0,T^{2/3}) - \sum_{i\in[n]} r_i u_i = O\big(\sqrt{T^{2/3}}\big) = O\big(T^{1/3}\big). \tag{EC.8}$$

A similar deduction holds for DLP given its regret bound of $O(\sqrt{T})$, leading to

$$\text{Rev}^\star(0,T^{2/3}) - \text{Rev}(\text{DLP-PA})(0,T^{2/3}) = \text{Rev}^\star(0,T^{2/3}) - \sum_{i\in[n]} r_i x_i^* T^{2/3} = O\big(\sqrt{T^{2/3}}\big) = O\big(T^{1/3}\big), \tag{EC.9}$$

where $x_i^*$ is the $i^{\text{th}}$ element of the optimal solution of DLP. By Equations (EC.8) and (EC.9), we have

$$\Big|\sum_{i\in[n]} r_i u_i - \sum_{i\in[n]} r_i x_i^* T^{2/3}\Big| = O\big(T^{1/3}\big). \tag{EC.10}$$

Building on assumption 3 of Agrawal et al. (2014), according to Sun et al. (2020), for any $i \in [n]$, the following equation holds with probability $1 - O(\frac{1}{T})$:

$$|u_i - x_i^* T^{2/3}| = O\big(T^{1/3}\log T\big). \tag{EC.11}$$

For a given $t \in [T^{2/3}, T]$, Algorithm GP-Enhanced-RDLP is executed with a time segment length of $T^{2/3}$, and with $x_i^* = \frac{u_i}{T^{2/3}}$. By Equation (EC.11), we have $|\frac{x_i^*}{\lambda_i} - \frac{u_i}{\lambda_i T^{2/3}}| = O\big(T^{-1/3}\big)$, and by Hoeffding's inequality, we have $|\frac{u_i}{\Lambda_i} - \frac{u_i}{\lambda_i T^{2/3}}| = O\big(T^{-1/3}\log T\big)$. Then, with triangle inequality, we have

$$\Big|\frac{x_i^*}{\lambda_i} - \frac{u_i}{\Lambda_i}\Big| = O\big(T^{-1/3}\log T\big), \tag{EC.12}$$

Therefore, to bound the second term in Equation (EC.7), we have

$$\begin{aligned}
&\text{Rev}^\star(T^{2/3},T) - \text{Rev}(\pi)(T^{2/3},T)\\
&= \Big(\text{Rev}^\star(T^{2/3},T) - \text{Rev}(\text{GP-Enhanced-RDLP})(T^{2/3},T)\Big)\\
&+ \Big(\text{Rev}(\text{GP-Enhanced-RDLP})(T^{2/3},T) - \text{Rev}(\pi)(T^{2/3},T)\Big)\\
&= O(\sqrt{T}) + \Big(\text{Rev}(\text{GP-Enhanced-RDLP})(T^{2/3},(k(\tau)-1)T^{2/3}) - \text{Rev}(\pi)(T^{2/3},(k(\tau)-1)T^{2/3})\Big)\\
&+ \Big(\text{Rev}(\text{GP-Enhanced-RDLP})((k(\tau)-1)T^{2/3},k(\tau)T^{2/3}) - \text{Rev}(\pi)((k(\tau)-1)T^{2/3},k(\tau)T^{2/3})\Big).
\end{aligned} \tag{EC.13}$$

By Equation (EC.12), we can obtain

$$\begin{aligned}\mathrm{Rev}(\mathsf{GP\text{-}Enhanced\text{-}RDLP})(T^{2/3},(k(\tau)-1)T^{2/3}) - \mathrm{Rev}(\pi)(T^{2/3},(k(\tau)-1)T^{2/3}) &= (T-T^{2/3})O\big(T^{-1/3}\log T\big)\\ &= O\big(T^{2/3}\log T\big).\end{aligned}$$

Moreover, as the length of the time segment $k(\tau)$ is $T^{2/3}$, we have

$$\mathrm{Rev}(\mathsf{GP\text{-}Enhanced\text{-}RDLP})((k(\tau)-1)T^{2/3},k(\tau)T^{2/3}) - \mathrm{Rev}(\pi)((k(\tau)-1)T^{2/3},k(\tau)T^{2/3}) = O(T^{2/3}).$$

Therefore, by Equation (EC.13), the second term in Equation (EC.7) is bounded by $O(\sqrt{T}) + O(T^{2/3}\log T) + O(T^{2/3}) = O(T^{2/3}\log T)$. Finally, by Equation (EC.7), we have the regret of Algorithm GP-Enhanced-BPC-OGD is

$$\mathrm{Rev}^\star - \mathrm{Rev}(\pi) = O(T^{2/3}) + O(T^{2/3}\log T) = O(T^{2/3}\log T).$$

## EC.6. Supplementary material for Section 5

*Proof of Theorem 4* First, we show that GP-Enhanced-BL is $(\alpha,\delta)$-fair. Without the capacity constraint, for each customer type $i$, GP-Enhanced-BL is the same as the FCFS algorithm under a resource capacity $b_i$, where $b_i$ is the booking limit. Therefore, by Theorem 1, GP-Enhanced-BL is $(\alpha,\delta)$-fair if the demand is less than the capacity. Then, let's consider the capacity constraint. Suppose that at a certain time period $t$, $\min_{j\in[L]} m_j(t) - \bar{a}n\gamma \le 0$, for each customer type $i$, if $s_i < b_i - \gamma$, then GP-Enhanced-BL starts a decreasing grace period to type $i$. If $s_i \ge b_i - \gamma$, then type $i$ has been already started a decreasing grace period, and this will not have any impact on the decision. Therefore, by Theorem 1, GP-Enhanced-BL is $(\alpha,\delta)$-fair.

Second, we show that GP-Enhanced-BL has a competitive ratio of $C - O(\frac{\log m}{m})$. If the total number of arrivals is $o(m)$, then with probability 1, no decreasing grace period will start because the number of accepted customers is much fewer than the booking limit or resource capacity. This leads to a competitive ratio of 1 because Algorithm 3 accepts everyone.

If the total number of arrivals is $\Omega(m)$, the offline optimal revenue is $\Theta(m)$ as the resource capacity scales with $m$. Compare to Algorithm BL, the loss is only from the decreasing grace period. The maximum number of customers who are rejected due to the decreasing grace period is the total length of decreasing grace periods: $\bar{a}n\gamma + n\gamma = (\bar{a}+1)n\gamma$, which will incur a maximum revenue loss of $(\bar{a}+1)n\gamma\bar{r}$. Therefore, the competitive ratio is

$$\inf_{I\in\mathcal{I}} \frac{\mathrm{Rev}(\pi(I))}{\mathrm{OPT}(I)} \ge \inf_{I\in\mathcal{I}} \frac{\mathrm{Rev}(\mathsf{BL}(I)) - (\bar{a}+1)n\gamma\bar{r}}{\mathrm{OPT}(I)} = C - O\left(\frac{\log m}{m}\right).$$

## EC.7. Supplementary material for Section 7

*Proof of Theorem 6* First, we show that by assigning a decreasing grace period $[t_1(i), t_2(i)]$, where $t_1(i) = \inf\{t : \min_{j\in[L]} m_j(t) \le \bar{a}n\gamma\}$ and $t_2(i) = T$ to each type $i$ customer, the static pricing algorithm is $(\alpha,\delta)$-fair. Similarly to Lemma 1, we only need to show that for any customer type $i$, with probability at least $1-\delta$, for any $u \in [n_i]$, $\mathbb{P}\left(p_u^{(i)} \ne p_u^{(i+1)}\right) \le \alpha$.

Define event $E$ as $E := \{\text{no resource is depleted in the time interval } [0,T]\}$. We show that $E$ happens with probability at least $1-\delta$ by showing that the complement of $E$ happens with probability at most $\delta$.

$$\begin{aligned}\mathbb{P}(E^{\mathsf{C}}) &\le \mathbb{P}(\text{more than } \gamma n \text{ customers are accepted in the grace period})\\ &\le \mathbb{P}(\exists i\in[n],\ \text{s.t. the number of type } i \text{ customers accepted in the grace period } \ge \gamma) \qquad \text{(EC.14)}\\ &= 1 - \mathbb{P}(\exists i\in[n],\ \text{s.t. the number of type } i \text{ customers accepted in the grace period } < \gamma)\\ &= (1-\alpha)^\gamma \qquad \text{(EC.15)}\\ &= (1-\alpha)^{\log_{1-\alpha}\delta} = \delta,\end{aligned}$$

where (EC.14) is due to the pigeonhole principle, and (EC.15) is because for each type $i$, the number of type $i$ customers accepted after the grace period starts is a geometric random variable with success probability $\alpha$. Therefore, the cdf is $1-(1-\alpha)^{\gamma}$.

Conditional on event $E$ happening, for any customer $i(u)$ arriving within $[1,t_1]$, the algorithm gives the same price $p_u^{(i)}$ to all of them. For any customer $i(u)$ arriving within $[t_1,t_2]$, by the definition of the grace period, we obtain that $\mathbb{P}\left(p_u^{(i)} \neq p_u^{(i+1)}\right) \leq \alpha$.

To complete the proof of the theorem, we need to show that the revenue loss is bounded by $\frac{\bar{a}}{\underline{a}} n\gamma\bar{p}$. In the worst case for the revenue, the algorithm rejects all customers after $t_1$. By the definition of $t_1$, the remaining units for each $j$ are $\bar{a}n\gamma$. Since $\bar{a}n\gamma$ units of resource can serve at most $\frac{\bar{a}}{\underline{a}} n\gamma$ customers, we have that the revenue loss is at most $\frac{\bar{a}}{\underline{a}} n\gamma\bar{p}$.