Two-stage Pandora’s Box for Product Ranking

(Authors’ names blinded for peer review)

In online platforms, consumers face an abundance of options that are displayed in the form of a position ranking. Only products placed in the first few positions are readily accessible to the consumer, and she needs to exert effort to access more options. We study how platforms with different business models should rank products to maximize their profit. We provide empirical evidence from a large online platform that products placed in higher positions have a favorable chance of being selected regardless of their utility. Further, the externality that high-positioned products impose on low-positioned ones substantially increases with their utility. Motivated by our empirical evidence, and building upon the seminal work of Weitzman (1979) (also known as the Pandora’s box), we develop a two-stage sequential search model; in the first stage, the consumer learns the intrinsic utility of products, already known to the platform, and forms a consideration set. While in the second stage, she inspects the products in her consideration set to learn the additional idiosyncratic utility she derives from a product. We develop optimal or FPTAS ranking algorithms under two diametric business models: one only concerned with maximizing consumer welfare - a suitable objective for platforms seeking consumer’s long-term engagement - and one concerned with maximizing short-term revenue. Further, somewhat surprisingly, we show that ranking products in a decreasing order of their intrinsic utilities does not necessarily maximize consumer welfare. Such ranking may shorten the consumer’s consideration set, due to the externality effect, leading to insufficient exploration.

Key words: two-stage consumer search, Pandora’s box, product ranking, search costs, online platforms
1. Introduction

With the record breaking growth in online shopping WSJ (2017a,b) and the profusion of choices that can be offered online, the role of online platforms in navigating consumer’s choice has become increasingly important. In a typical online shopping experience, upon entering a keyword, the consumer is presented with numerous search results displayed in a sequence of panels or web-pages, with most of the options being initially invisible. For instance, Expedia displays hotels vertically with only two or three options immediately visible to the consumer. The consumer needs to scroll down to access more options. For the most part, the number of options a platform offers far exceeds the consumer’s attention and cognitive resources. Millward Brown reports that 70% of Amazon users do not go beyond the first page of search results clavisinsight.com (2015). As a result of such consumer behavior, it is well-documented that ranking of the products displayed to the consumer has a substantial impact on her choice (Kim et al. 2010, De los Santos and Koulayev 2013, Jeziorski and Segal 2015, Chen and Yao 2016, Ursu 2016). This makes product ranking an instrumental lever for a platform to achieve various goals in line with its business model, such as driving up its short-term revenue or establishing a large consumer base in the long-run. In this work, we study the problem of product ranking that platforms with different business models face each time a consumer enters a keyword.

In the following, we first draw insights on consumer behavior by analyzing query-level data from a large online platform. Then, we provide a summary and an overview of our contributions.

Empirical Evidence on the Impact of Ranking on Consumer Choice: We analyze a dataset from a platform that, for each query, displays a number of options ranging from a few to a few dozens. We defer the details about the setting, dataset, and analyses to Appendix EC.1, and here, we just present our analysis on the impact of position and products’ externality. In Figure 1a, we present aggregate level data for queries where 30 options were displayed. We plot the probability of purchase for each position under two rankings: random ranking and the platform’s optimized ranking. Under both ranking mechanisms, products in lower positions are less likely to be purchased. Such a decreasing trend indicates that consumer search is sequential and costly. However, this decline is sharper under the platform’s ranking. A similar empirical observation has been made by Ursu (2016) using a dataset from Expedia.
top positions may be filled with low utility products. In that case, the consumer is more likely to continue her search beyond the few top positions. The platform’s ranking, on the other hand, tend to place higher utility products in higher positions;\(^5\) the benefit of doing so is evident in the increased purchase probability of the first 5 positions compared to the random ranking setting. At the same time, due to the externality that those high utility products impose in lower positions, the purchase probability for positions lower than 13 is smaller compared to that of random ranking.\(^6\)

To further investigate the externality effect, we analyze purchase probabilities on an individual product level for the subset of queries where 2 options were displayed. Under random ranking, for each pair of products, we traced all the queries in which these two products were displayed either under ranking \((1, 2)\) or \((2, 1)\) where in ranking \((i, j)\), product \(i/j\) is in the top/bottom position. We focus on a pair of products that is among the most frequently displayed pairs. The first product of this pair, denoted by Product 1, is significantly more popular than its second product, denoted by Product 2.\(^7\) Figure 1b compares the purchase probability for each product under rankings \((1, 2)\) and \((2, 1)\). We observe that placing Product 1 in position 2 results in a 37% decrease in its purchase probability; however, for Product 2, lowering its position reduces the purchase probability by 51%. This further highlights the higher externality that the popular product in the top position imposes on the product in the bottom position, compared to the case where the positions of these products are reversed. Moreover, the overall purchase probability under ranking \((2, 1)\) is 7% higher than under ranking \((1, 2)\). This suggests that placing the most popular product in the higher position is suboptimal for maximizing the overall probability of purchase.

**Summary of Contributions:** Motivated by our empirical observations, and focusing on the platform’s ranking problem, we (1) develop a novel two-stage consumer search model that captures key features observed in our empirical study as well as in other empirical work; (2) design polynomial-time optimal ranking algorithms tailored to platform’s business model; (3) derive insights into the structure of the optimal ranking and the possible poor performance of simple and commonly used policies (e.g., utility-ordered or revenue-ordered ranking).

In the rest of this introduction, we elaborate on each of our contributions.

**A New Two-Stage Consumer Search Model:** We develop a new consumer search and choice model that captures the key features of consumer behavior. At the same time, our model lends itself to designing efficient algorithms for the ranking problems. Note that in the absence of efficient algorithms, one needs to exhaustively search over an exponential number of rankings to find the

---

\(^5\) Based on personal communication with the platform.

\(^6\) The overall probability of purchase under the platform’s ranking is 5% higher than that under random ranking. This further highlights the significant positive impact of optimizing the ranking.

\(^7\) For each product, we have a “popularity” measure that can be viewed as a proxy for its expected utility.
optimal one. Consistent with the choice modeling literature, we assume the utility that a consumer
derives from each product consists of two parts: an intrinsic utility (IU) and an idiosyncratic
part; initially the consumer does not know either parts, and she needs to exert a search cost to
learn each part. The search cost for each position varies across types of consumers; while in all
types, the cost becomes higher for lower positions. We utilize the framework of Pandora’s box for
sequential search problems Weitzman (1979) in both stages of our model. In the first stage, the
consumer screens products to learn their IU, and forms a consideration set. The consideration set
formed through this process (see Eq. (1)) has several desired properties: (1) The consumer screens
products starting from the highest position and moving downward. (2) The consideration set is
endogenously determined by both the ranking and consumer type. (3) Placing products with high
IU in higher positions may reduce the size of the consideration set; this captures the externality
that such products impose on lower-positioned ones. (4) Consumer types with lower search costs
will form larger consideration sets.

After forming the consideration set, the consumer proceeds to the second stage, where she
inspects products in her consideration set to learn the idiosyncratic part of their utility. Her order
of search in the second stage, however, does not necessarily follow the displayed order. Instead, she
inspects products in a decreasing order of intrinsic utilities (see Lemma 1 and the description after
that).

Under our novel consumer search model, we study the optimal ranking problem for a platform
that pursues long-term business goals or short-term ones.

---

8 This happens under some mild and reasonable assumptions detailed in Section 3.1.1. Also, Chu et al. (2017) has
recently derived this property using a one-stage search model that bears some similarity to ours. For a detailed
comparison, see Section 2.
**Optimal Ranking for Long-term Business Goals:** For many platforms, maximizing consumer’s measure of “satisfaction”, such as her welfare or her probability of purchase, is the foremost objective, as these platforms seek to build long-run reputation of helping consumers find their favorite products without too much effort. As such, we study the problem of platform’s optimal ranking with the objective of maximizing a weighted sum of the consumer net value and her probability of purchase. We refer to this objective function as consumer welfare. For any fixed number of consumer types, we develop a polynomial-time algorithm that returns the optimal ranking by solving a polynomial number of LPs (Theorem 1). Somewhat surprisingly, we show that ordering products based on their IUs does not necessarily maximize consumer welfare (see Section 4.3). Said another way, it may not be optimal to display the best product in the highest position. Due to the idiosyncratic part of the utility, consumers may benefit from screening products beyond those with the highest IUs, even though screening is costly. This resembles the exploration-exploitation trade-off that arises in the learning literature and multi-armed bandit problems. The optimal ranking strikes a balance between such exploration, i.e., screening more products, and exploitation, i.e., including products with high IU in the consideration set. The algorithm also resolves another trade-off which arises when faced with multiple types of consumers; to keep the types with lower search costs screening further, the platform may forfeit the opportunity of displaying the highest IU products to all types.

**Optimal Ranking for Short-term Business Goals:** Many platforms derive revenue from a product sale that is proportional to its price. For instance, Airbnb collects 9%-15% of each booking from the host and the guest. In such a setting, platforms that are more focused on their short-term profit, may pursue the objective of maximizing the yield revenue from the displayed products. We study the problem of maximizing revenue in a special class of our (general) model, where upon forming the consideration set, the consumer makes decisions according to classic random utility choice models; in particular, according to the standard Multinomial Logit (MNL) model. In a homogenous population, we develop a polynomial-time algorithm (Theorem 2). In designing a revenue maximizing ranking, our algorithm faces yet a new trade-off between including high-priced products in the consideration set vs. including high IU ones. For a multi-type population, we show that the problem of maximizing revenue is NP-complete (Theorem 3). We design an FPTAS algorithm by exploiting certain functional properties of the objective function, the structural properties of feasible consideration sets, and casting the problem as a knapsack problem with side constraints (Theorem 3).

---

9 Such an objective function has been the main focus of the work on assortment planning; see for example Golrezaei et al. (2014), Aouad and Segev (2015).
The rest of the paper is organized as follows. In Section 2, we review the related literature. In Section 3, we introduce the setting, our model of consumer search, as well as the platform’s objective. In Sections 4 and 5, we present our algorithms and insights for the problems of maximizing consumer welfare and revenue, respectively. Section 6 concludes the paper. Appendix EC.1 is dedicated to describing our case study based on a real dataset and presenting our estimation results. For the sake of brevity, we only include proofs of selected results in the main text. The detailed proofs of the rest of the statements are deferred to appendices.

2. Related Work

Our work relates to and contributes to several streams of literature.

Product Ranking and Empirical Studies: A strand of papers in marketing literature empirically study the causal impact of ranking on consumer’s choice (Kim et al. 2010, De los Santos and Koulayev 2013, Ghose et al. 2014, Jeziorski and Segal 2015, Chen and Yao 2016, Ursu 2016). Even though the aforementioned papers differ in techniques used to address the endogeneity of ranking, they all found that placing a product at higher positions increases its probability of purchase. Another finding consistent across this line of work is that utility-ordered ranking improves consumer welfare compared to the currently employed ranking methods. The focus of our work is different from the above papers; besides proposing a new consumer search model, we also focus on designing efficient algorithms for the platform’s optimal ranking problems.

From a modeling perspective, closest to our work is Ursu (2016) that analyzed a unique dataset from Expedia which includes the purchasing outcomes of random ranking as well as optimized ranking. Ursu (2016) estimated a sequential search model utilizing the framework of Weitzman (1979). The main difference in the modeling approach of Ursu (2016) and that of ours is related to the a priori knowledge of consumers about products. In particular, Ursu (2016) assumed that consumers know the IU of all products (or equivalently learn them for free) and only pay a search cost for learning the idiosyncratic part of the utility. However, as discussed in the introduction, given that most products are not visible to the consumer and she needs to exert effort to even take a first glance, such an assumption may not bear out in practice. In our work, we relax such an assumption by proposing a two-stage search model. In the first stage, we explicitly account for the search cost of learning the IUs.

Rank Optimization: A few recent papers studied the platform rank optimization problems utilizing models that reflect the impact of position on consumer choice (Davis et al. 2013, Abeliuk et al. 2014, Aouad and Segev 2015, Gallego et al. 2016), and they all devise elegant methods to find the

10 Returning to the Expedia example, a typical search for a hotel includes more than a hundred search results, and only two or three of them are visible.
optimal or approximation solution for the underlying revenue maximization problems. Here, we mainly discuss the differences in the modeling approach of our work and the above papers. Davis et al. (2013), Abeliuk et al. (2014) studied an MNL choice model where the impact of position on a product’s intrinsic utility is incorporated exogenously. While resulting in a tractable framework, such a modeling approach cannot fully capture the externality effect. More generally, this model possesses the Independence of Irrelevant Alternatives property that has been challenged beyond its applicability to ranking problems McFadden et al. (1977). Aouad and Segev (2015) took a different modeling approach, and they proposed a consider-then-choose model in which the population is composed of different types of consumers; for each type, the size of consideration set is exogenously drawn from a distribution and is not a function of the ranking of the products. Gallego et al. (2016) took a similar modeling approach, and studied a closely related problem. Different from the above papers, we construct a choice model based on a microeconomic foundation in which consideration sets are endogenously determined by the ranking.

**Position Auctions:** Motivated by online advertising platforms that sell multiple ad slots through auctions and the externality that ads in higher positions impose on those in lower positions, a stream of papers in computer science and economics proposed various models that incorporate the impact of position on click-through-rates (CTR). This line of work mainly focused on properties of the GSP auction and its relation to the VCG mechanism. Varian (2007), Edelman et al. (2007) modeled the impact of position by multiplying the CTR by an exogenous position-dependent factor. This bears similarity to the modeling approach of Davis et al. (2013), Abeliuk et al. (2014); as mentioned for those papers, such a modeling approach cannot capture the externality effect. On the other hand, Aggarwal et al. (2008), Kempe and Mahdian (2008), Athey and Ellison (2011) proposed models that capture the externality of higher-positioned products on lower-positioned ones. All models assume the user views ads sequentially starting from the highest position; at any position the user decides to stop her search independently with an ad-specific probability. However, Jeziorski and Segal (2015) provided empirical evidence that some users click on a lower position before clicking on a higher one.

Closest to our work is Chu et al. (2017) that proposed a sequential search model based on Weitzman (1979) framework. Two main factors differentiate our model from theirs: (1) in Chu et al. (2017) consumers pay the same search cost for any position, whereas in our model, search costs are position-dependent (consistent with the empirical literature (Kim et al. 2010, Chen and Yao 2016, Ursu 2016)). (2) Chu et al. (2017) assumed that the consumer learns both the IU and

---

11 For example, by multiplying the IU of a product by an exogenous position-dependent factor.
12 Hummel and McAfee (2014) also proposed a model similar to that of Davis et al. (2013), Abeliuk et al. (2014) with the same practical limitations.
her idiosyncratic utility in a single stage of search. Consequential to the above assumptions, the consumer’s decision in Chu et al. (2017) simplifies to a single stopping time problem where the consumer searches downward and stops the first time she finds a product with utility larger than a (position-independent) threshold. Therefore, similar to the model of Aggarwal et al. (2008), Kempe and Mahdian (2008), Athey and Ellison (2011), products placed in lower positions may not impact the choice probability of higher-positioned products. In contrast, we consider a two-stage search model with multiple consumer types that allows for capturing richer interactions between substitutability of products in different positions as well as modeling externality.

**Consumer Search Behavior:** Building on the seminal work of Weitzman (1979), many studies model consumer’s search and learning behavior as a sequential process where the consumer decides on the order of her search. For a sample of recent work, see Ursu (2016) for structural estimation, Kleinberg et al. (2016), Chu et al. (2017) for the study of auctions, Armstrong (2016) for a competition setting between two sellers. Additionally, several papers study the impact of consumer search cost on pricing, assortment planning, and market expansion (Cachon et al. 2005, 2008, Wang and Sahin 2017). Our work contributes to this recent line of work by studying the ranking problem of an online platform under a novel two-stage sequential search model.

**Choice Modeling:** With the abundance of substitutable products, there has been a growing body of literature in choice-based demand modeling; see for example (Ulu et al. 2012, Honhon and Seshadri 2013, Golrezaei et al. 2014, Rusmevichientong et al. 2014, Agrawal et al. 2016, Bernstein et al. 2015, Besbes and Sauré 2016, Blanchet et al. 2016, Kallus and Udell 2016, Wang and Wang 2016, Bernstein et al. 2017, Vaccari et al. 2018). One of the modeling paradigms widely proposed and studied is the so-called *consider-then-choose* models, in which the consumer makes the choice according to a two-stage process; first forms a consideration set and then chooses one of the alternatives from it (see Roberts and Lattin (1991), Hauser (1978, 2014) for structural estimation and empirical analysis, and Aouad et al. (2015), Jagabathula and Rusmevichientong (2016), Wang and Sahin (2017) for the study of related optimization problems). Our choice model also constitutes a two-stage process, in first of which, the consideration set is formed endogenously through a sequential search process.

### 3. Model and Preliminaries

Consider an online platform that displays $n$ items indexed by $i \in [n] = \{1, 2, \ldots, n\}$. The items are ordered in $n$ positions, indexed by $j \in [n]$, where the positions with lower indices have a higher visibility.\(^{13}\) Depending on the market that the platform hosts, an item can be a product, an app,

\(^{13}\) For instance, in vertical search (e.g., Amazon, Trip Advisor, and Yelp), position one is the best position.
a service, or an ad featuring one. The platform displays the items in a form of an ordering to the consumer, seeking a desired action from her. The consumer’s action can take the form of a click, a purchase, or a download. To be concrete, hereafter, we refer to the item and the desired action as product and purchase. The platform’s goal is to find the optimal ranking of the products. To be more precise, the platform’s decision is a permutation \( \pi : [n] \rightarrow [n] \), where \( \pi(j) = i \) means product \( i \) is placed in position \( j \). We will elaborate on the platform’s objective in Section 3.2. In the next subsection, we explain how products’ position impacts the consumer’s purchasing decisions.

### 3.1. Consumer’s Search Behavior and Choice Model

We first discuss the consumer’s utility for each product \( i \in [n] \). The consumer’s utility consists of two parts: a shared intrinsic utility (IU) denoted by \( U_i \) and an idiosyncratic part denoted by \( Z_i \).

We adopt the convention of presenting random variables \( U_i \) and \( Z_i \) by uppercase letters, and their realizations by lowercases, \( u_i \) and \( z_i \). We assume the IU of all \( n \) products are i.i.d. samples from probability distribution \( f_U : [\bar{u}, \bar{u}] \rightarrow \mathbb{R}^+ \). Further, for each product \( i \) and each consumer, \( Z_i \) is drawn independently from probability distribution \( f_Z : \mathbb{R} \rightarrow \mathbb{R}^+ \). The consumer also has an outside option represented by product 0. We normalize the IU of the outside option by setting \( u_0 = 0 \). However, the idiosyncratic part \( Z_0 \) is an independent draw from distribution \( f_Z \).

The platform has perfect knowledge of distributions, \( f_U \) and \( f_Z \), as well as the IU of all \( n \) products. Access to an unprecedented amount of historical data and computational power makes it reasonable to assume that well-established online platforms have such perfect knowledge. On the consumer side, we assume she knows the distributions \( f_U \) and \( f_Z \), and that \( u_0 = 0 \). However, she does not know either a product’s IU or her idiosyncratic shock, and she needs to exert effort to learn both parts. The assumption that the consumer does not know the products’ IU is partly motivated by the large number of products in online platforms, most of which are initially invisible.

Next we describe the consumer’s search process to learn her utility for a product. We start with an overview of the seminal work of Weitzman (1979) on sequential search processes.

**Pandora’s Box: a Framework for Sequential Search:** Weitzman (1979) characterizes the optimal sequential search policy where the consumer (“Pandora”) looks for the best option (“box”) among a set of boxes with uncertain rewards. Pandora is presented by \( n \) closed boxes where box \( b \) contains a random reward \( X_b \), where \( X_b \) is drawn independently from the probability distribution \( f_b \). To discover the realization of her reward for box \( b \), Pandora needs to open the box and incur

---

14 Our main results can be extended to a setting where the probability distribution of the IUs depend on the position of the products; however, assuming that the consumer is oblivious to the platform’s ranking algorithm is practically reasonable due to complexity of such algorithms and consumer’s lack of knowledge with regard to platform’s objective; Chu et al. (2017) Chen and Yao (2016) make a similar assumption.
the non-refundable search cost \( s_b \), where \( s_b \leq \mathbb{E}_{X_b}[X_b] \) (otherwise box \( b \) is never worth opening).

Here, \( \mathbb{E}_Y \) denotes the expectation with respect to random variable \( Y \).

Pandora consumes one of the boxes and she wishes to maximize her net value, which is her reward of the chosen box minus the total search cost that she incurs. We note that to consume a box, the box needs to be opened first. To maximize her net value, Pandora decides in what order to open the boxes and when to stop her search process. When she stops her search, she consumes the best opened box, i.e., the one with the highest realized reward. Suppose \( \mathcal{O} \) represents the set of boxes she opened before stopping. We use calligraphic font for discrete sets and intervals. Then, her net value is given by \( \max_{b \in \mathcal{O}} \{ x_b \} - \sum_{b \in \mathcal{O}} s_b \).

Weitzman (1979) showed Pandora’s optimal sequential search policy admits a surprisingly simple solution: for each box, define a reservation price, \( r_b \), as follows: \( \mathbb{E}_{X_b}[(X_b - r_b)^+] = s_b \), where \( (y)^+ = \min\{y, 0\} \). Note that the reservation price is positive because \( s_b \leq \mathbb{E}_{X_b}[(X_b)^+] \). Intuitively the reservation price is the deterministic reward that makes Pandora indifferent between (i) opening box \( b \), paying cost \( s_b \), and getting a random reward of \( \max\{X_b, r_b\} \) (ii) leaving box \( b \) unopened and getting a deterministic reward of \( r_b \). We note that the reservation price \( r_b \) only depends on the search cost of box \( b \), i.e., \( s_b \), and the distribution of its reward, i.e., \( f_b \). Given the reservation prices, the optimal sequential search policy can be described as the following rules. (1) Selection Rule: If another box to be opened, it should be one of the closed boxes with the highest reservation price. (2) Stopping Rule: Stop the search process whenever the maximum realized reward of the opened boxes exceeds the reservation price of all closed ones.

3.1.1. Our Two-stage Search Model
With the above background, we describe our model of consumer search. The consumer performs a two-stage sequential search; in the first stage the consumer screens the products to learn the IUs, and forms a consideration set. In the second stage, she further inspects products in her consideration set to learn the idiosyncratic part of the utility she derives from each product.\footnote{Our two-stage search model is motivated by how consumers conduct their search in online platforms. In the first stage, consumers eye-ball the products in the higher positions and in the second stage, consumers choose some of the products that they eye-balled in the first stage and open new tabs for them in order to conduct another level of inspection.} We utilize the Pandora’s box framework in both stages as described below.

We utilize the Pandora’s box framework in both stages as described below.

First Stage: Forming the Consideration Set
The first stage of the screening highly depends on the position-dependent search cost of the consumers. As one can imagine, consumers can have heterogeneous search costs. While some consumers are willing to screen all the products in first few pages of the search results, some of them might hardly screen one product. To capture
this heterogeneity, we assume that our consumer population is comprised of $K$ types, where the position-dependent search cost varies across types. More precisely, each type $k \in [K]$ is characterized by a set of costs $\{s^k_1, s^k_2, \ldots, s^k_n\}$ where $s^k_j$ is the search cost of the consumer of type $k$ to learn the IU of a product in position $j$. We assume $s^k_{j+1} \geq s^k_j$, for $k \in [K]$ and $j \in [n-1]$. That is, for any type $k \in [K]$, higher positions have lower search costs. We further assume that as $k$ increases, consumers get more patient, i.e., their search cost decreases. That is, we assume that $s^k_j \geq s^{k+1}_j$ for any $j \in [n]$ and $k \in [K-1]$.

Before the first stage of screening, the consumer does not know the IU of the products. Thus, all the products are ex-ante symmetric and the only differentiating factor is the search cost associated with their position. This implies that the consumer starts screening the products with lower search costs, i.e., those in higher positions. This can be verified using Pandora’s box framework stated at the beginning of this section. Recall that Pandora starts opening the boxes in the decreasing order of their reservation prices. In our setting, for a consumer of type $k \in [K]$, the reservation price of a product in position $j \in [n]$, denoted by $r^k_j$, solves $E_U[(U - r^k_j)^+] = s^k_j$ where $U$ is a random variable drawn from the IU probability distribution, $f_U$. It is easy to observe that $r^k_j \geq r^{k+1}_{j+1}$ for $j \in [n-1]$, which implies that products in higher positions will be screened first. In what follows, we further define $r^k_{n+1} = -\infty$ for any $k \in [K]$.

Suppose $E_U[(U^+) \geq s^k_j$, otherwise the consumer does not screen any product. Then, the consumer starts screening the product in the first position and adds this product to her consideration set, i.e., $\pi(1) \in C^k_\pi$. Here, $C^k_\pi$ denotes the consideration set of a consumer of type $k$ when she faces ranking $\pi$, i.e., when product $\pi(j)$ is placed in position $j \in [n]$. After observing the IU of the product in the first position, she decides whether to stop her search. In particular, a consumer of type $k$ stops her search if $\max\{u_{\pi(1)}, u_0\} \geq r^k_2$ where $u_0 = 0$ is the IU of the outside option and $u_{\pi(j)}$ is the IU of the product in position $j$. If she continues her search, she screens the product in the second position and adds it to her consideration set $\pi(2) \in C^k_\pi$, and so on. After terminating her search process, the consideration set of a consumer consists of the $|C^k_\pi|$ products in higher positions. In other words, $C^k_\pi = \{\pi(1), \pi(2), \ldots, \pi(|C^k_\pi|)\}$. We note that Aouad and Segev (2015), Chu et al. (2017) assume that the consumer screens downward, starting from the top positions, and thus her consideration set consists of products in top positions. Here, we do not make this assumption: instead we present a model that can explain why consumers only consider the products in the top positions.

We point out that the consumer always screens the no-purchase option; that is, the actual consideration set of the consumer includes product 0. However, to simplify the notation, we exclude

---

16 A commonly used model for search cost, $s^k_j$, in the literature has an exponential form, i.e., $a^k \exp(b^k j)$ Kim et al. (2010), Ghose et al. (2014), Chen and Yao (2016), Ursu (2016). Then, given that $(a^1, b^1) \geq (a^2, b^2) \geq \ldots \geq (a^K, b^K)$, we get $s^k_j \geq s^{k+1}_j$ for any $j \in [n]$, where $(a^i, b^i)$ if $a^i \geq a^2$ and $b^i \geq b^2$. 
product 0 from the consideration $C^k_\pi$. Let $[n_1, n_2] = \{n_1, n_1 + 1, \ldots, n_2\}$ for any $n_1 < n_2 \in \mathbb{Z}$. The formation of the consideration set $C^k_\pi$ is summarized below.

**Forming Consideration Set $C^k_\pi$**

Initialize $C^k_\pi = \emptyset$. Then, for $j \in [n]$, stop if

$$\max_{j' \in [0:j]} \{u_{\pi(j')}\} \geq r^k_{j+1},$$

otherwise continue and add product $\pi(j)$ to the consideration set: $\pi(j) \in C^k_\pi$.

We point out that when $|C^k_\pi| = 0$, the consumer does not screen any product $i \in [n]$. As stated earlier, this happens when $\mathbb{E}_U[(U)^+] \leq s^k_1$, or equivalently $r^k_i \leq 0$.

Before proceeding to describe the second stage and the choice model, we highlight some features and structural properties of consideration sets. (i) The consideration set is endogenously determined by the ranking. That is, the order of the products changes the consideration set of the consumer. Note that in Aouad and Segev (2015), Gallego et al. (2016), the size of the consideration set of consumers is exogenously drawn from a distribution. (ii) Products with a high IU can impose significant externality on products that are placed in lower positions. This holds because the consumer stops her search whenever $\max_{j' \in [0:j]} \{u_{\pi(j')}\}$ exceeds the reservation price $r^k_{j+1}$. One can think about $\max_{j' \in [0:j]} \{u_{\pi(j')}\}$ as the externality that products in position 1, 2, ..., $j$ impose on the product in position $j+1$, i.e., $\pi(j+1)$.

**Second Stage: Choice Model of Consumers**

After forming a consideration set, the consumer proceeds to inspect the idiosyncratic part of her utility for some of the products in her consideration set through a sequential search process. We assume the consumer learns the idiosyncratic part of her utility for the outside option, i.e., $z_0$, cost free. However, for any other product $i$ she pays a search cost of $\sigma$ to learn $z_i$.\textsuperscript{17} Utilizing the Pandora’s box framework, we first compute the reservation prices as follows:\textsuperscript{18} We note that in the second stage of inspection, the search cost does not depend on the position of the products. Assuming position independent search cost in the second stage is reasonable, as the consumer has already made the effort needed to observe lower positioned products. In addition, this assumption can be justified by a recent empirical work by Ursu (2016). In this work, using a dataset from Expedia, the author empirically showed that with random ranking and conditioned on a click, the purchase probability of the hotels does not increase

\textsuperscript{17} Assuming that the search cost $\sigma$ is the same across consumer types and positions is consistent with prior literature Aouad and Segev (2015), Aouad et al. (2015), and empirical analysis of Ursu (2016).

\textsuperscript{18} The second stage of inspection happens after the platform has engaged the consumer and made her interested in a set of products. Because of this, it is reasonable to assume that in the this stage, the consumers are more homogeneous in their search process and are mainly concerned about choosing the best product that matches their idiosyncratic taste. We point out that many prior work including Aouad and Segev (2015) and Aouad et al. (2015) also assume that consumers are only heterogeneous in how they form their consideration sets.
as their position improves. This shows that after the first stage of screening, consumers are less concerned about the position of the products and are more interested in purchasing a product with a high utility.

Utilizing the Pandora’s box framework, we first compute the reservation prices as follows:

**Lemma 1 (Reservation Prices in the Second Stage).** Consider a consumer of type \( k \). For any product \( i \in C^k_\pi \), the reservation price in the second stage, denoted by \( r_i \), is given by \( u_i + r_Z \), where \( E_Z [(Z - r_Z)^+] = \sigma \). Here, random variable \( Z \) represents the idiosyncratic part of the utility for a product and is drawn from the probability distribution \( f_Z \). For the no-purchase option, \( r_0 = \infty \).

Lemma 1 shows that the consumer inspects the products in her consideration set in the order of the realized IU. Therefore, in this second stage, the consumer does not necessarily follow the order prescribed by the ranking. Let \( C^k_\pi = \{i_1, i_2, \ldots, i_{|C^k_\pi|}\} \) where \( u_{i_1} \geq u_{i_2} \geq \ldots \geq u_{i_{|C^k_\pi|}} \). Then, the consumer first inspects product \( i_1 \). If \( \max\{z_0, u_{i_1} + z_{i_1}\} \geq r_{i_2} = u_{i_2} + r_Z \), the consumer stops her second stage of inspection. In this case, she purchases product \( i_1 \) if \( u_{i_1} + z_{i_1} \geq z_0 \), and she leaves without making any purchase if \( u_{i_1} + z_{i_1} < z_0 \). Otherwise, if she continues her inspection, the next step proceeds alike. Then, this search process can be summarized as follows:

For \( j \in [|C^k_\pi| - 1] \): stop if

\[
\max_{j' \in [0, j]} \{u_{i_{j'}} + z_{i_{j'}}\} \geq r_{i_{j+1}} = u_{i_{j+1}} + r_Z.
\]

If the search is stopped at \( j \in [|C^k_\pi| - 1] \), the consumer purchases product \( i_{j^*} \) where \( j^* = \arg\max_{j' \in [0, j]} \{u_{i_{j'}} + z_{i_{j'}}\} \). When the search is not stopped at any \( j \in [|C^k_\pi| - 1] \), the consumer has inspected all the products in her consideration set, and she purchases product \( i_{j^*} \) where \( j^* = \arg\max_{j' \in [0, |C^k_\pi|]} \{u_{i_{j'}} + z_{i_{j'}}\} \).

### 3.2. Platform’s Objective and Optimal Ranking Problem

With the above description of consumer search behavior and its resulting choice model, we proceed to describing the platform’s objective and decisions. Online platforms employ diverse monetization schemes ranging from commission to subscription to online advertising marketplace (2016). Depending on the monetization scheme, competitive landscape, and sensitivity of consumers (to the quality of search results) platforms pursue different business models for profitability. For instance, a platform encountering fierce competition may pursue long-term goal of retaining its market share by maximizing consumer welfare, whereas a monopoly that derives revenue from a percentage-based commission (per purchase) may pursue the short-term goal of maximizing immediate revenue. In this paper, we mainly focus on these two diametric business models. We introduce two different
objectives - each suited for one of the business models - and study the platform’s ranking problems for each objective separately.

Before introducing the objectives, we describe the platform’s knowledge regarding consumer types. Suppose the consumer type distribution is given by $\theta_1, \theta_2, \ldots, \theta_K$; see Section 3.1 for the definition of the consumer type. The platform does not know the type of a consumer, however it does know the type distribution. Further, throughout the paper we assume that for all types of consumers $r_k^1 > 0$, i.e., all types screen at least one product. We note that this assumption is without loss of generality; a type with $r_k^1 \leq 0$ does not screen any product, therefore any ranking decision by platform does not impact such a consumer. Thus we can exclude such types and normalize the type distribution.

**Long-term Objective - Maximizing Consumer Welfare:** Let $p^k_{\pi(j)}$ be the probability that, under ranking $\pi$, a consumer of type $k$ purchases the product in position $j$; further let $v^k_\pi$ be the expected net value that a consumer of type $k$ derives under ranking $\pi$. For a ranking $\pi$ and each consumer type $k$, we define the expected consumer welfare as the weighted sum of her expected net value and her overall probability of purchase, and we define the platform’s optimal ranking problem as:

$$Wel = \max_{\pi \in \Pi} \sum_{k \in [K]} \theta_k \left( \alpha v^k_\pi + \beta \sum_{j: \pi(j) \in C^k_\pi} p^k_{\pi(j)} \right),$$  

(2)

where $\alpha, \beta \in \mathbb{R}^+$ are chosen by the platform. With a slight abuse of notation, we also use $Wel$ as our shorthand for consumer welfare maximization problem. Section 4 is dedicated to studying the above optimization problem.\textsuperscript{19}

**Short-term Objective - Maximizing Revenue:** Let $q_i$ be the revenue that the platform derives from product $i$. For the objective of maximizing expected revenue, platform’s optimal ranking problem can be summarized as:

$$Rev = \max_{\pi \in \Pi} \sum_{k \in [K]} \theta_k \left( \sum_{j: \pi(j) \in C^k_\pi} q_{\pi(j)} p^k_{\pi(j)} \right),$$

(3)

where $Rev$ is also our shorthand for the revenue maximization problem. We study the revenue maximization problem in Section 5. In the rest of the paper, for the sake of brevity, we suppress the word “expected.”

\textsuperscript{19}Note that for a platform that derives revenue from a fixed commission (per purchase), the overall probability of purchase is proportional to its expected revenue; therefore the objective in (2) can alternatively be viewed as the weighted sum of consumer net value, and the platform’s revenue.
4. Optimal Ranking for Maximizing Consumer Welfare

In this section, we study the ranking problem of a platform concerned with maximizing consumer welfare as defined in (2). The main result of this section is as follows:

**THEOREM 1 (Consumer Welfare Maximization).** Suppose that the number of types $K$ is constant; the consumer welfare maximization problem, defined in (2), admits a polynomial-time solution.

The proof of Theorem 1 consist of several steps that establish and utilize the structural properties of the consideration sets, consumer net value, and probability of purchases under our model. To simplify the exposition, in the following, we present our solution when the number of types is two, i.e., $K = 2$. In Remark EC.2 in Appendix EC.2, we explain how this solution can be generalized to more than two types. Our strategy in solving problem (2) is to first solve a constrained version of the problem where we enforce the size of consideration set $C^k_\pi$ to be $\nu^k$:

$$\text{Wel}(\nu^1, \nu^2) = \max_{\{\pi; |C^1_\pi| = \nu^1, |C^2_\pi| = \nu^2\}} \sum_{k \in [2]} \theta^k \left( \alpha v^k_n + \beta \sum_{j: \pi(j) \in C^k_\pi} p^k_{\pi(j)} \right),$$

(4)

where $\nu^k \in [n]$, for $k \in [2]$. Note that the number of possible combinations of $(\nu^1, \nu^2)$ is polynomial in $n$. Therefore, solving (4) polynomially results in having a polynomial-time algorithm for the unconstrained problem (2). To solve (4), we construct an integer linear program (ILP) with $4n$ binary variables and at most $5n + 4$ constraints; then we consider the LP relaxation of such ILP and show that the LP admits an integeral solution.

In the next two subsections, we provide the proof steps. We start with presenting auxiliary results that will be used to construct a feasible region of the ILP. In Section 4.2, we proceed to constructing an appropriate objective function of the ILP, and study its LP relaxation. Finally, to derive insights on the structure of the optimal ranking, in Section 4.3, we present an example and a simplified algorithm for the single-type problem, i.e., $K = 1$.

4.1. Constructing the ILP Feasible Region

We start by analyzing the consideration sets. Such analysis proves helpful in defining appropriate decision variables for the ILP and constructing a feasible region for problem (4). First with the following lemma, we establish the relationship between consideration sets of different types.

**LEMMA 2 (Nested Consideration Sets).** If $s^k_j \geq s^{k+1}_j$ for any $j \in [n]$ and $k \in [K - 1]$, then for any ranking $\pi$, $C^k_\pi \subseteq C^{k+1}_\pi$. 

Lemma 2 shows that the consideration sets of consumers are nested: when type \( k \in [K-1] \) consumers screen a product, consumers with type greater than \( k \) also screen that product. Next we proceed to studying the relationship between a ranking and the consideration sets; first we focus on the consideration set of one of the consumer types, and then move on to analyzing the relationship between the ranking and consideration sets of both types.

The platform cannot directly select a subset of products to be included in the consideration set of any type. Instead the subset needs to be endogenously induced by a ranking. Motivated by this observation, we define implementability of a consideration set as follows:

**Definition 1 (Implementability).** We call a subset \( C \subseteq \{n\} \) implementable for type \( k \), if there exists a ranking \( \pi \) for which \( C^k_\pi = C \).

In the following proposition, we establish an equivalence between implementability of a set and whether it is induced by a simple ranking construction.

**Proposition 1 (Ranking and Consideration Set).** Let \( C \subseteq \{n\} \) be a subset with size \( \nu \). For a consumer of type \( k \), suppose \( r^k_{\nu+1} > 0 \). Set \( C \) is an implementable consideration set iff it is induced by a ranking with the following structure: Position \( j \in [\nu] \) is filled by a product from set \( C \), and

- **Stopping Product:** Position \( \nu \) is filled by a product whose IU belongs to stopping region \( S := [r^k_{\nu+1}, \infty) \).

- **Engaging Products:** Position \( j \in [\nu-1] \) is filled by a product whose IU belongs to engaging region \( E := (-\infty, r^k_{\nu}) \).

Note that we use the symbol “:=” for definitions. Figure 2a illustrates the definition of \( S \) and \( E \) regions. We point out that regions \( S \) and \( E \) overlap. Further, we note that the above proposition does not result in a unique ranking, nor does it include all the rankings that induce consideration set of \( C \). However, given that the ranking has a simple structure (more specifically, position \( \nu \) is filled by a product from region \( S \)) it proves helpful in constructing a feasible region for our ILP. Further we remark that the ranking of products only impacts the consideration set formed in the first stage screening. For a fixed consideration set, the order in which the consumer inspects the products (to learn the idiosyncratic part of her utility) is independent of the ranking; as a result her final choice and her net value is also independent of the ranking (see Lemma 1 and the description after that).

Finally we comment on the assumption \( r^k_{\nu+1} > 0 \). First note this assumption also implies \( r^k_{\nu} > 0 \) (because \( r^k_{\nu} \geq r^k_{\nu+1} \)), which is a necessary condition for having a consideration set of size \( \nu \) (otherwise the consumer of type \( k \) stops before reaching position \( \nu \) regardless of the ranking). However, it

\(^{20}\) By definition, any ranking does not assign a product \( i \in C \) to more than one position.
is possible to have $r_{\nu+1}^k \leq 0 < r_{\nu}^k$. In that case, the consumer stops at position $\nu$ regardless of the product placed in this position. We note that all of our analyses can be modified to address this special case, details of which is deferred to Appendix EC.2 (Remark EC.1). In the rest of this section and Section 5, when we study optimization problems with constraint $|C^k_\pi| = \nu$ we always assume $r_{\nu+1}^k > 0$.

Next we proceed to analyzing the relationship between the ranking and consideration sets when there are two types of consumers. As it becomes more clear later, this helps us construct a feasible region for the ILP.

**Proposition 2 (Ranking and Consideration Sets of Both Types).** Suppose $|C^k| = \nu^k$, $k \in [2]$, and $r_{\nu+1}^k > 0$. Sets $C^1$ and $C^2$ are implementable consideration sets iff both sets are induced by a ranking with the following structure: Position $j \in [\nu^1]$ is filled by a product from set $C^1$, and

- **Engaging products for both types:** Position $j \in [\nu^1 - 1]$ is filled by a product whose IU belongs to region $E_b := (-\infty, \min\{r_{\nu+1}^1, r_{\nu+1}^2\})$.

- **Stopping product for type one:** Position $\nu^1$ is filled by a product whose IU belongs to region $S_o := [r_{\nu+1}^1, r_{\nu+1}^2]$.

- **Additional engaging products for type two:** Position $j \in [\nu^1 + 1, \nu^2 - 1]$ is filled by a product whose IU belongs $E_t := (-\infty, r_{\nu+1}^2)$.

- **Stopping product for type two:** Position $\nu^2$ is filled by a product whose IU belongs to region $S_t := [r_{\nu+1}^2, \infty)$.

In the above proposition, the subscripts in $E_b$, $S_o$, $E_t$, $S_t$ stand for “both”, “one”, and “two”, respectively. Figure 2b illustrates these four regions in an example with $\min\{r_{\nu+1}^1, r_{\nu+1}^2\} = r_{\nu+1}^1$. We remark that the above four regions overlap. As we will show later, this makes the design of an optimal ranking more involved. We note that depending on the values of $\nu^1$ and $\nu^2$, we may not need to consider some of the aforementioned regions. For instance, if $\nu^1 = 1$, we only need to specify
the stopping product for type 1 and region $E_b$ can be ignored. As another example, when $\nu^1 = \nu^2$, regions $S_o$ and $E_t$ can be ignored. However, to be as comprehensive as possible, we assume that none of the regions can be ignored.

In light of the above proposition, we define four binary (decision) variables for each product $i$: $x_{\rho,i}$, where region $\rho \in R = \{E_b, S_o, E_t, S_t\}$. We will use these variables to set up our ILP. These variables determine which product should be assigned to which position. Note that we do not need to decide about the exact position of each product, as the order of products in positions $j \in [\nu^1 - 1]$ and the order of the products in positions $j \in [\nu^1 + 1, \nu^2 - 1]$ can be chosen arbitrarily. It suffices to decide to which region, each product $i$ "contributes." We say product $i$ contributes to region $\rho = E_b$ and we set $x_{E_b,i} = 1$ when we assign this product to one of positions in $[\nu^1 - 1]$. When product $i$ does not contribute to region $\rho$, we set $x_{\rho,i} = 0$. Similarly, we say product $i$ contributes to region $\rho = S_o, E_t, S_t$ when we assign this product to position $\nu^1$, one of the positions in the range of $[\nu^1 + 1, \nu^2 - 1]$, and position $\nu^2$, respectively. Note that if a product does not contribute to any region, the product is an ignored product. That is, the product does not belong to any of the consideration sets.

We then define $c_\rho$ to be the number of products that should contribute to each region. That is, for $\rho \in \{E_b, S_o, E_t, S_t\}$, $c_\rho$ is, respectively, $\nu^1 - 1, 1, \nu^2 - \nu^1 - 1, 1$. With these definitions, conditions of Proposition 2 hold if the following constraints are satisfied.

$$
\sum_{i \in [n]} x_{\rho,i} = c_\rho, \rho \in R \\
\sum_{\rho \in R} x_{\rho,i} \leq 1, \quad i \in [n] \\
x_{\rho,i} = 0, \quad i \notin \rho, \rho \in R.
$$

(5)

These constraints will be used as the feasible region of our ILP. The first set of constraints enforces the number of the products that should contribute to each region. The second set of constraints guarantees that each product is assigned to at most one of the regions. The third set of constraints ensures that the products that do not fall in a region $\rho$ do not contribute to the region. Here, with a slight abuse of notation, $i \in \rho$ means that $u_i \in \rho$.

As stated earlier, a set of variables that satisfy Eq. (5) can correspond to more than one ranking. However, each feasible solution corresponds to a unique consideration set of size $\nu^k$ for type $k \in [2]$. Hereafter, we use a feasible solution and its corresponding pair of consideration sets interchangeably.

4.2. Constructing the ILP Objective

We start by analyzing the objective of Problem (4). Recall that the objective, which is called consumer welfare, is a weighted sum of the consumer’s net value and her probability of purchase. The following proposition - which is a crucial step in constructing the ILP objective - establishes a monotonic relationship between consumer welfare and the IU of the products in her consideration set.
Proposition 3 (Monotonicity of Consumer Welfare). Suppose that consideration sets $C$ and $\tilde{C}$ are both implementable for a consumer of type $k \in [K]$, and they differ in only one product, i.e., $\tilde{C} = C \setminus \{i_1\} \cup \{i_2\}$ and $i_1 \neq i_2$. If product $i_1$ has a higher IU that product $i_2$ (i.e., $u_{i_1} > u_{i_2}$), then the consumer welfare of type $k$ achieved by $C$ is greater than or equal to that achieved by $\tilde{C}$.

Utilizing monotonicity property established in Proposition 3 and with the formulation introduced in Section 4.1, we aim to find an objective function that results in welfare maximizing consideration sets. Before constructing such an objective, we define a re-indexing of products, which simplifies the notation in the rest of the section: We re-index the products in a decreasing order of their IU, i.e., $u_1 \geq u_2 \ldots \geq u_n$.

We motivate the construction of our objective function by making two crucial observations: Our first observation is that due to monotonicity of consumer welfare in the IUs (Proposition 3), when multiple products can be selected to satisfy a certain constraint, the objective function of our ILP must prioritize products with higher IUs. We formalize this observation in the following lemma:

Lemma 3 (Breaking Ties between Products). Let $\{x_{\rho,i} : \rho \in \mathcal{R}, i \in [n]\}$ and $\{\tilde{x}_{\rho,i} : \rho \in \mathcal{R}, i \in [n]\}$ be two feasible solutions, i.e., they satisfy Eq. (5). Suppose that these solutions are identical except for the following variables:

\[ x_{\rho_1,i_1} = 1, \quad \text{and} \quad x_{\rho,i_2} = 0 \quad \forall \rho \in \mathcal{R}, \quad \text{while} \quad \tilde{x}_{\rho_1,i_2} = 1, \quad \text{and} \quad \tilde{x}_{\rho,i_1} = 0 \quad \forall \rho \in \mathcal{R} \]

If $i_1 < i_2$, then consumer welfare achieved by $\{x_{\rho,i} : \rho \in \mathcal{R}, i \in [n]\}$ is greater than or equal to that achieved by $\{\tilde{x}_{\rho,i} : \rho \in \mathcal{R}, i \in [n]\}$.

Note that by our re-indexing, $i_1 < i_2$ implies that $u_{i_1} \geq u_{i_2}$. Lemma 3 shows that the first solution ($\{x_{\rho,i} : \rho \in \mathcal{R}, i \in [n]\}$) in which $i_1$ contributes to region $\rho_1$ and $i_2$ is an ignored product, weakly dominates the second one in which the role of $i_1$ and $i_2$ are reversed.\(^{21}\)

Our second observation is that when a product with a high IU can contribute to more than one constraints, the objective function of the ILP must break ties (among constraints) in favor of more "profitable" constraints. In particular, the constraints associated with regions $\mathcal{E}_b$ and $\mathcal{S}_o$ must be given higher priories because any products that contributes to these two regions is included in the consideration sets of both types. We formalize this observation in the following lemma:

Lemma 4 (Breaking Ties between Regions). Let $\{x_{\rho,i} : \rho \in \mathcal{R}, i \in [n]\}$ and $\{\tilde{x}_{\rho,i} : \rho \in \mathcal{R}, i \in [n]\}$ be two feasible solutions, i.e., they satisfy Eq. (5). Suppose that these two feasible solutions are identical except for the following variables:

\[ x_{\rho_1,i_1} = 1, \quad \text{and} \quad x_{\rho_2,i_2} = 1, \quad \text{while} \quad \tilde{x}_{\rho_2,i_1} = 1, \quad \text{and} \quad \tilde{x}_{\rho_1,i_2} = 1 \]

\(^{21}\)One can use the proof of Lemma 3 to show Proposition 3 or the other way around. Thus, in the appendix, we present a single proof for both of these results.
where \( \rho_1 \in \{E_b, S_o\} \) and \( \rho_2 \in \{E_t, S_t\} \). If \( i_1 < i_2 \) then consumer welfare, achieved by \( \{x_{\rho,i}: \rho \in \mathcal{R}, i \in [n]\} \), is greater than or equal to that achieved by \( \{\tilde{x}_{\rho,i}: \rho \in \mathcal{R}, i \in [n]\} \).

By Lemma 4, the first solution in which product \( i_1 \) contributes to region \( \rho_1 \in \{E_b, S_o\} \) and product \( i_2 \) contributes to region \( \rho_2 \in \{E_t, S_t\} \) weakly dominates the second solution in which the role of \( i_1 \) and \( i_2 \) are reversed.

Motivated by the above observations, we define our objective function to be the weighted sum of variables \( \{x_{\rho,i}: \rho \in \mathcal{R}, i \in [n]\} \), where the weights are defined as follows:

\[
\omega_{\rho,i} = 2(n+1-i), \quad \rho \in \{E_b, S_o\}, \quad i \in [n],
\]

\[
\omega_{\rho,i} = 2(n+1-i) - 1, \quad \rho \in \{E_t, S_t\}, \quad i \in [n],
\]

(6)

The following proposition, which proves Theorem 1, states that the following ILP

\[
\max_{\{x_{\rho,i}: \rho \in \mathcal{R}, i \in [n]\} \in \{0,1\}^{4n}} \sum_{i \in [n], \rho \in \mathcal{R}} \omega_{\rho,i} x_{\rho,i} \quad \text{s.t.} \quad \text{Constraints (5)}
\]

(ILP)

gives us an optimal (welfare maximizing) solution \( \{x^*_{\rho,i}: \rho \in \mathcal{R}, i \in [n]\} \) which can be used to construct an optimal ranking for problem (4). Weights \( \omega_{\rho,i}, \rho \in \mathcal{R} \) and \( i \in [n] \), are given in (6). Equally importantly, the proposition also states that the resulting ILP can be solved in polynomial time because its associated LP relaxation admits an integral solution.\(^{22}\)

**Proposition 4 (Optimality of ILP).** Suppose that the ILP is feasible and let \( x^* = \{x^*_{\rho,i}: \rho \in \mathcal{R}, i \in [n]\} \) be its optimal solution. Then,

1. A ranking associated with solution \( x^* \) is an optimal ranking.

2. Its LP relaxation, in which \( x_{\rho,i} \in [0,1], \rho \in \mathcal{R} \) and \( i \in [n] \), has an integral solution.

**4.3. Discussion on Optimal Ranking for Welfare Maximization**

Here, we provide further insights on the structure of the optimal solution for the welfare maximization problem by presenting an example as well as a simplified algorithm that returns the optimal ranking when there is a single type of consumer. The example shows that ranking products in a decreasing order of their IUs is not optimal. The single-type algorithm sheds light on the structure of the optimal ranking.

\(^{22}\) We discuss the running times of our algorithms inside the proofs in the appendices.
4.3.1. Sub-optimality of the Utility-ordered Ranking  First note that in light of Proposition 3 that states that consumer welfare is increasing in the IU of products included in her consideration set, one might speculate that ranking product in a decreasing order of their IU would result in deriving the maximum consumer welfare. In the following example, however, we show that such ranking can indeed be far from optimal.

**Example 1 (Suboptimality of IU-ordered Ranking).** Suppose that the platform displays five products. The IUs are uniformly distributed in interval $[0, 1]$, i.e., $f_U(u) = 1$ for $u \in [0, 1]$. There are two types of consumers with probability distribution $(\theta^1, \theta^2) = (0.5, 0.5)$. The search costs of types 1 and 2 and the IU of products $i \in [5]$ are given in Table 1. Further, we assume that $\sigma = 0$ and $f_Z$ is the standard Gumbel distribution. The weights in the consumer welfare objective (defined in (2)), $\alpha$ and $\beta$, are 1 and 0.25 respectively.

<table>
<thead>
<tr>
<th>Position</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>$s_1^j$</td>
<td>0</td>
<td>0.131</td>
<td>0.197</td>
<td>0.2</td>
<td>0.303</td>
</tr>
<tr>
<td>$s_2^j$</td>
<td>0</td>
<td>0.008</td>
<td>0.055</td>
<td>0.059</td>
<td>0.181</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Index</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>$u_j$</td>
<td>0.77</td>
<td>0.61</td>
<td>0.6</td>
<td>0.53</td>
<td>0.4</td>
</tr>
</tbody>
</table>

Table 1  Model Primitives of Example 1

The optimal welfare-maximizing ranking is $(5, 2, 3, 1, 4)$, which gives the welfare of 1.07. However, the welfare from IU-ordered ranking, i.e., $(1, 2, 3, 4, 5)$, is 0.94. This shows that the IU-order ranking reduces the consumer welfare by 12%, compared with the optimal ranking. To understand the sub-optimality of IU-ordered ranking, we focus on the consideration sets induced by the two rankings. The optimal ranking induces consideration sets of $\{2, 5\}$ (top two products) and $\{1, 2, 3, 5\}$ (top fours products) for types 1 and 2, respectively. Whereas the IU-ordered ranking induces sets of $\{1\}$ and $\{1, 2\}$, respectively. This highlights the inefficiency of the IU-ordered ranking due to the externality of high IU products when they are positioned in higher positions: by placing the best product, i.e., product 1, in the first position, the platform does not incentivize the consumers of types 1 to screen beyond the first position.\(^{23}\) We note that due to idiosyncratic part of the utility, consumers may benefit from screening products beyond those with the highest IUs, even though screening is costly. This resembles the exploration-exploitation trade-off that arises in the learning literature and multi-armed bandit problems.

The optimal ranking strikes a balance between such exploration, which involves costly screening, and exploitation, i.e., including products with the high enough IU in the consideration set. For instance here, the optimal ranking incentives the consumers of type 1 and 2 to screen 2 and

\(^{23}\) Similarly, for type 2 consumers, by placing products 1, 2 in the first and second positions, the platform does not incentivize them to screen beyond the second position.
4 products, respectively. We point out that addressing the exploration-exploitation trade-off is particularly challenging because of the heterogeneity of search costs across different types. In this example, under the optimal ranking, type 1 consumers (that are the less patient ones) do not even include best product in their consideration set, however, type 2 consumers do: in order to enlarge the consideration set of the more patient type (type 2), the platform sacrifices the additional welfare for type 1 consumers had they included the best product in their consideration set.

4.3.2. Single-Type Population In the following, we study the constrained welfare maximization problem in a homogenous population. In particular, using a similar notation as in (4), we aim to solve the following optimization problem.

\[ \text{Wel}(\nu) = \max_{\pi : |C_\pi| = \nu} \left( \alpha v_\pi + \beta \sum_{j : \pi(j) \in C_\pi} p_{\pi(j)} \right), \tag{7} \]

where \( \nu \) is the size of the consideration set. Note that for the single-type problem, we drop the superscripts associated with type.

For a consideration set size of \( \nu \), utilizing Propositions 1 and 3, we show that the optimal ranking has the following simple structure.

**Algorithm 1:** Optimal solution of Problem \( \text{Wel}(\nu) \), defined in (7).

Let \( S = \{i \mid i \in [n], u_i \geq r_{\nu+1}\} \) and \( E = \{i \mid i \in [n], u_i < r_{\nu}\} \).

**IF** (i) \( |S| < 1 \), or (ii) \( |E| < (\nu - 1) \), or (iii) \( |S \cap E| = 0 \) and \( |E| < \nu \), then the problem is infeasible.

**ELSEIF** \( |S \setminus E| > 0 \), then,

**Stopping Product:** Place product \( i^* = \arg\max_{i \in S \setminus E} \{u_i\} \) in position \( \nu \).

**Engaging Products:** Place \( \nu - 1 \) products with the highest IU from set \( E \) in positions \( 1, 2, \ldots, \nu - 1 \) in an arbitrary order.

**ELSE**

**Stopping Product:** Place products \( i^{**} = \arg\max_{i \in S \cap E} \{u_i\} \) in position \( \nu \).

**Engaging Products:** Place \( \nu - 1 \) products with the highest IU from set \( E \setminus \{i^{**} \} \) in positions \( 1, 2, \ldots, \nu - 1 \) in an arbitrary order.

**Proposition 5 (Algorithm 1 is Optimal).** Suppose that there is only one type of consumer. If Problem \( \text{Wel}(\nu) \), defined in (7), is feasible, then Algorithm 1 returns an optimal ranking.

The three feasibility conditions given in Algorithm 1 corresponds to (i) we need at least one product in the stoping region \( S \), (ii) we also need at least \( (\nu - 1) \) in the engaging region. (iii) because sets \( S \) and \( E \) overlap, if \( S \subseteq E \) then we need at least \( \nu \) products in \( E \).

The optimal ranking of Algorithm 1 has an illuminating structure: to increase the consumer welfare, the platform would like the consumer to include the best product, i.e., product with the highest IU, in her consideration set. At the same, given the externality that this best product
imposes on products at the lower positions, the platform would not necessarily benefit from placing it at position 1. Particularly when \( |S \setminus E| > 0 \), the best product, i.e., \( i^* = \arg \max_{i \in S \setminus E} \{ u_i \} \) is not engaging. Therefore, by placing it at position \( \nu \), the platform (a) avoids the consumers from stopping their screening early on (b) incentivizes them to screen the best product, and (c) stops them from screening products in position \( j \in [\nu + 1, n] \).

For the top \( \nu - 1 \) positions, if the platform fills the positions with low IU products, even though the consumer continues her screening, in expectation, she does not derive enough welfare from such products. Thus, to increase consumer welfare, the platform places the top products in set \( E \) in position \( j \in [\nu - 1] \). However, to keep the consumer engaged, the IU of all the products that are placed in the top \( \nu - 1 \) positions should be less than \( r_\nu \). In fact, the ranking of Algorithm 1 strikes the optimal balance between keeping the consumers engaged and increasing her welfare by displaying products with moderate IUs - that do not impose externality on the lower positions in the consideration set while maximally contributing to her welfare.\(^{24}\)

We conclude this section by noting that we cannot construct an optimal ranking for the two-type problem based on the above simple solution of the single-type one. In Appendix EC.5, we present such an example.

5. Optimal Ranking for Optimizing Revenue

In this section, we study the ranking problem of a platform concerned with maximizing revenue for a given set of prices, \( q_i \geq 0 \) for \( i \in [n] \). Here we restrict our attention to settings where the consumer learns idiosyncratic part of her utility without paying any cost. Such a setting arises when the idiosyncratic part of utility is readily available to the consumer but unobserved to the platform. For instance, for a platform that displays ads and derives revenue from consumer’s click, the product is an ad and the desired action is the consumer click. After the first-stage of screening and forming a consideration set, the consumer decides which ad to click (if any) without further inspection. In such a setting, our choice model reduces to the random utility choice models. Further, we assume that \( f_Z \) is the standard Gumbel distribution. In this case, after setting \( \sigma \) to 0, our choice model has the form of the standard Multinomial Logit (MNL) model:

\[
p^{k}_{i}(j) = \frac{w_{i}(j)}{1 + \sum_{j' \in C^{k}_{i}} w_{i}(j')}
\]

where \( w_{i} = \exp(u_{i}) \). We emphasize that only for the products in the consideration set \( C^{k}_{i} \) the purchase probability (8) has the same form as MNL. For the rest of the products that are not in the consideration set, the purchase probability is zero.

\(^{24}\) There exists empirical evidence that suggests the effectiveness of this technique; see for example Ngwe (2017), Ngwe and Teixeira (2017). In these works, the best products have not been shown in the best positions.
We note that the MNL models are widely used in studying various optimization problem under consumer choice (see for example Wang and Sahin (2017), Aouad and Segev (2015) and references therein); This is partly due to its special form of the purchase probability that leads to developing tractable solutions. In this section, we also make use of this special form to design an optimal algorithm for the revenue maximization problem in a homogenous population as well as an FPTAS for a population with multiple types. Using (8), we define the revenue maximization problem as follows:

\[
\text{Rev} = \max_{\pi \in \Pi} \sum_{k \in [K]} \theta^k \left( \sum_{j: \pi(j) \in C_k^\pi} \frac{q_{\pi(j)} w_{\pi(j)}}{1 + \sum_{j' \in C_k^\pi} w_{\pi(j')}} \right),
\]

(9)

We start by outlining the main results of this section: We first show that for a homogenous population, the revenue maximization problem can be solved polynomially; see Theorem 2 and Algorithm 2 in Section 5.1. We then show that the revenue maximization problem becomes NP-complete in a heterogenous population even with two types; However, we show that there exists an FPTAS for the problem; see Theorem 3 in Section 5.2.

5.1. Single-type Population

In this section, we show that the revenue maximization problem can be solved polynomially when there is only a single-type of consumers.

**Theorem 2 (Single-type Revenue Maximization).** Suppose the population is comprised of a single-type of consumers, i.e., \( K = 1 \). Then, the revenue maximization problem, defined in (9), admits a polynomial-time solution.

In the following, we explain how we design a polynomial-time algorithm for the single-type revenue maximization problem. Similar to the proof strategy of Theorem 1, we aim to first solve the constrained version of the revenue maximization problem, defined as follows.

\[
\text{Rev}(\nu) = \max_{\pi: |C_\pi| = \nu} \sum_{j: \pi(j) \in C_\pi} \frac{q_{\pi(j)} w_{\pi(j)}}{1 + \sum_{j' \in C_\pi} w_{\pi(j')}},
\]

(10)

where \( \nu \) is the size of consideration set. Also, for the single-type case, we drop the superscripts for the type index. We note that in the revenue-maximization problem, unlike the welfare-maximization problem, breaking ties in favor of products with higher IU is not optimal, as a product with a high IU may have a low revenue. That is, there is a tension between the revenue of the products, \( q_i \)'s, and their IU, \( u_i \)'s. To deal with this tension, we need to “sort” the products so that we can effectively break the ties between them. Following a similar idea used in Rusmevichientong et al. (2010),
Megiddo (1978), we can show that it suffices to consider $O(n^2)$ ways of sorting. To determine these $O(n^2)$ ways of sorting, we make use of the rational form of the objective function to express $\text{Rev}(\nu)$ as follows.

$$
\text{Rev}(\nu) = \max \left\{ \lambda \in \mathbb{R} : \exists \pi, |C_\pi| = \nu, \sum_{j: \pi(j) \in C_\pi} w_{\pi(j)} (q_{\pi(j)} - \lambda) \geq \lambda \right\}
$$

(11)

Let us define $h_i(\lambda) := w_i (q_i - \lambda)$. Eq. (11) implies that for any given $\lambda$, product $i$ is preferable to product $i'$ if $h_i(\lambda) > h_{i'}(\lambda)$. Thus, the ties should break in favor of products with higher $h_i$'s. Based on this observation, we need to sort the products based on $h_i(\lambda)$'s as $\lambda$ moves from $-\infty$ to $\infty$. We also note that as $\lambda$ ranges over the real values, the lines $h_i(\lambda)$ have $L \leq \binom{n+1}{2} = O(n^2)$ intersections. In between two consequent intersections, the ordering of lines does not change. As a result, one only needs to consider $O(n^2)$ ways of sorting.\(^{25}\) Then, given that we have $L$ unique intersection points, we can define $L + 1$ non-overlapping intervals such that in each interval the order of $h_i$'s remains the same.\(^{26}\) To represent an interval $l \in [L + 1]$, we choose an arbitrary point in it, denoted by $\lambda_l$. Then, solving problem (11) reduces to solving

$$
\text{Rev}(\nu, \lambda_l) = \max_{\pi: |C_\pi| = \nu} \sum_{j: \pi(j) \in C_\pi} h_{\pi(j)}(\lambda_l), \quad l \in [L + 1]
$$

(12)

Therefore, our strategy in designing a polynomial-time algorithm for Problem $\text{Rev}(\nu)$ is to solve Problem $\text{Rev}(\nu, \lambda_l)$ to find its best consideration set. Then, among the $L + 1$ resulting consideration sets, we exhaustively search for the best one. We note that Problem $\text{Rev}(\nu, \lambda_l)$ bears similarity to Problem $\text{Wel}(\nu)$. In problem $\text{Wel}(\nu)$, the objective was increasing in $u_i$'s and here the the objective is increasing in $h_i$'s. As a result, finding the optimal ranking boils down to selecting the products with highest values of $h_i$'s that belong to the corresponding stopping and engaging regions. Algorithm 2 presents our optimal algorithm.

\(^{25}\) The intersection point of two lines $h_i(\cdot)$ and $h_{i'}(\cdot)$ is point $y$ that solves $h_i(y) = h_{i'}(y)$. Also, we set $h_0(\lambda) = 0$ for the outside option, and we need to consider the intersection point of $h_i(\cdot)$'s and $h_0(\cdot)$.

\(^{26}\) The first interval ranges from $-\infty$ to the minimum intersection point and the last interval ranges from the maximum intersection point to $\infty$. 
ALGORITHM 2: Optimal solution of Problem \( \text{Rev}(\nu, \lambda_i) \), defined in (12).

Let \( S = \{ i | i \in [n], u_i \geq r_{\nu+1} \} \) and \( E = \{ i | i \in [n], u_i < r_\nu \} \). Construct two rankings and return the one with larger objective:

**Ranking 1:**

**Stopping Product:** Place product \( i^* = \arg\max_{i \in S \setminus E} \{ h_i(\lambda_i) \} \) in position \( \nu \).

**Engaging Products:** Place \( \nu - 1 \) products with the highest \( h_i(\lambda_i) \) from set \( E \) in positions \( 1, 2, \ldots, \nu - 1 \) in an arbitrary order.

**Ranking 2:**

**Stopping Product:** Place product \( i^{**} = \arg\max_{i \in S \cap E} \{ h_i(\lambda_i) \} \) in position \( \nu \).

**Engaging Products:** Place \( \nu - 1 \) products with the highest \( h_i(\lambda_i) \) from set \( E \setminus \{ i^{**} \} \) in positions \( 1, 2, \ldots, \nu - 1 \) in an arbitrary order.

Note that Ranking 1 has the exact same structure as that of Algorithm 1 once we substitute \( u_i \) by \( h_i(\lambda_i) \). However, in Algorithm 2, we also need to consider Ranking 2. The reason is that in Problem \( \text{Rev}(\nu, \lambda_i) \), even if product \( i^* \) exists, there is no relation between \( h_{i^*}(\lambda_i) \) and \( h_i(\lambda_i) \) for \( i \in E \). This is in contrast with Algorithm 1 in which if product \( i^* \) exists, it has an IU higher than any product in \( E \), and therefore it is preferable to be included as the stopping product.

The following proposition establishes the optimality of Algorithm 2 and proves Theorem 2.

**Proposition 6 (Algorithm 2 is Optimal).** Suppose \( K = 1 \). If Problem \( \text{Rev}(\nu, \lambda_i) \) is feasible, Algorithm 2 gives an optimal ranking.

### 5.2. Multi-type Population

The following is the main result of this section. We show that Problem \( \text{Rev} \), given in (9), is NP-complete even when the number of types is two. We further show that this problem admits a fully polynomial-time approximation scheme (FPTAS).\(^{27}\)

**Theorem 3 (Revenue Maximization Problem: Hardness Result and FPTAS).** Suppose \( K > 1 \) and \( K \) is a constant.

- The revenue maximization problem, defined in (9), is NP-complete.
- The revenue maximization problem, defined in (9), admits a fully polynomial-time approximation scheme (FPTAS).

\(^{27}\) In Appendix EC.6, we present an example that shows revenue-ordered ranking is suboptimal.
The hardness result uses ideas similar to Gallego et al. (2016), and it involves a reduction from the PARTITION problem that is a well-known NP-complete problem; see Appendix EC.4.

We now outline our ideas for developing an FPTAS for the revenue-maximization problem when the number of types is constant. An FPTAS is a family of algorithms parameterized by $\epsilon$ such that for any $\epsilon > 0$, there exists an algorithm that for any instance of the problem that returns a solution whose revenue is no less than $1 - \epsilon$ fraction of the optimal revenue, and whose running time is polynomial in the input size of the problem and $\frac{1}{\epsilon}$. To ease the exposition, we present the algorithm when the number of types is two. Similar to Section 4, we first consider a constrained version of the problem (9) where we enforce the size of consideration set $C^k$ to be $\nu^k$. Using the formulation we developed in Section 4, we write the constrained revenue maximization problem as a function of the decision variables $x_{\rho,i}$, where $i \in [n]$ and $\rho \in \mathcal{R} = \{\mathcal{E}_b, \mathcal{S}_o, \mathcal{E}_t, \mathcal{S}_t\}$, as follows:

$$\text{Rev}(\nu^1, \nu^2) = \max_{\{x_{\rho,i} : \rho \in \mathcal{R}, i \in [n]\}} \sum_{k \in [2]} \sum_{\rho \in \mathcal{R}^k} \sum_{i \in [n]} \theta^k q_i w_i x_{\rho,i}$$

$$\text{s.t.} \sum_{i \in [n]} x_{\rho,i} = c_\rho, \quad \rho \in \mathcal{R}, \quad \sum_{\rho \in \mathcal{R}} x_{\rho,i} \leq 1, \quad i \in [n], \quad x_{\rho,i} = 0, \quad i \notin \rho, \rho \in \mathcal{R} \quad (13)$$

where $\mathcal{R}^1 = \{\mathcal{E}_b, \mathcal{S}_o\}, \mathcal{R}^2 = \mathcal{R}$, and $c_\rho, \rho \in \mathcal{R}$, is defined in Section 4.1. Note that to present the denominator in a succinct way, we define variables $x_{\rho,0}$, $\rho \in \mathcal{R}$. However, these are not decision variables; we set $x_{\rho,0} = 1$ for $\rho = \mathcal{E}_b$, and $x_{\rho,0} = 0$ for $\rho \in \{\mathcal{S}_o, \mathcal{E}_t, \mathcal{S}_t\}$. Finally, we point out that by developing an FPTAS for problem $\text{Rev}(\nu^1, \nu^2)$, we will also have an FPTAS for the original problem, defined in (9). Our FPTAS algorithm, which is inspired by Lawler (1979), Désir et al. (2014), involves discretizing the coefficients of the denominators and numerators of the objective function of (13), defining a knapsack problem related to (13), and solving it using dynamic programming.

We start with discretizing the coefficients of the denominators and numerators of the objective function in order to find an approximate solution of Problem $\text{Rev}(\nu^1, \nu^2)$ efficiently. Let $\bar{q} = \min_{k \in [2]} \min_{i \in [n]} q_i \theta^k, \bar{\nu} = \max_{k \in [2]} \max_{i \in [n]} q_i \theta^k, \bar{w} = \min_{i \in [n]} w_i$, and $\bar{\bar{w}} = \max_{i \in [n]} w_i$. Then, for any given $\epsilon$, we define

$$\mathcal{A}_\epsilon = \left\{\frac{wq(1 + \epsilon)^\alpha}{\alpha = 0, 1, \ldots, A}\right\} \quad \text{and} \quad \mathcal{B}_\epsilon = \left\{\frac{w(1 + \epsilon)^\beta}{\beta = 0, 1, \ldots, B}\right\}$$

Here, $A = O(\log(n\bar{q}\bar{\bar{w}}/(qw))/\epsilon)$ and $B = O(\log((n + 1)\bar{\bar{w}})/\epsilon)$. We note that $n\bar{q}\bar{\bar{w}}$ and $qw$ are respectively the maximum and minimum values that the numerators in problem (13) can take. In constructing $\mathcal{A}_\epsilon$, we discretize the range of $[qw, n\bar{q}\bar{\bar{w}}]$ such that the ratio of two consecutive points is equal to $(1 + \epsilon)$. We followed a similar procedure for the denominators. We will use all the points in

$^{28}$ To be precise, $A = \log(n\bar{q}\bar{\bar{w}}/(qw))/\log(1 + \epsilon)$ and $B = \log((n + 1)\bar{\bar{w}})/\log(1 + \epsilon)$. 
$(\mathcal{A}_e)^2 \times (\mathcal{B}_e)^2$ as our guesses for the numerators and denominators of the objective in problem (13). To be precise, for any $(a, b) \in (\mathcal{A}_e)^2 \times (\mathcal{B}_e)^2$, we seek to determine whether the optimal objective of the following optimization problem is greater $a^1$, where $a = (a^1, a^2)$ and $b = (b^1, b^2)$.

$$\max_{\{x_{\rho,i}: \rho \in \mathcal{R}, i \in [n]\}} \sum_{\rho \in \mathcal{R}} \sum_{i \in [n]} \theta_{\rho,i} w_i x_{\rho,i}$$

$$s.t. \sum_{\rho \in \mathcal{R}} \sum_{i \in [0,n]} w_i x_{\rho,i} \leq b^k, \quad k \in [2], \quad \sum_{\rho \in \mathcal{R}} \sum_{i \in [n]} \theta_{\rho,i} w_i x_{\rho,i} \geq a^2,$$

$$\sum_{i \in [n]} x_{\rho,i} = c_{\rho}, \quad \rho \in \mathcal{R}, \quad \sum_{\rho \in \mathcal{R}} x_{\rho,i} \leq 1, \quad i \in [n], \quad x_{\rho,i} = 0, \quad i \notin \rho, \rho \in \mathcal{R}. \quad (14)$$

Next, we discuss how one can find an approximate solution of problem (14) polynomially. As a first step, we normalize and discretize the coefficients. We then use dynamic programming (DP) to find an approximate solution to problem (14). Given $(a, b)$, we define the new coefficients as

$$\tilde{q}_i^k = \left\lfloor \frac{\theta_{k,i} w_i}{\epsilon a^k/n} \right\rfloor, \quad \tilde{w}_i^k = \left\lfloor \frac{w_i}{\epsilon b^k/(n+1)} \right\rfloor, \quad k \in [2], \quad i \in [n]. \quad (15)$$

Having defined the normalized coefficients, we now formalize our DP. For each $(l, m, s)$, let

$$V_g(l, m, s) = \max_{\{x_{\rho,i}: \rho \in \mathcal{R}, i \in [g,n]\}} \sum_{\rho \in \mathcal{R}} \sum_{i = g}^{n} \tilde{q}_i x_{\rho,i}$$

$$s.t. \sum_{\rho \in \mathcal{R}} \sum_{i = g}^{n} \tilde{w}_i^k x_{\rho,i} \leq m^k, \quad k \in [2], \quad \sum_{\rho \in \mathcal{R}} \sum_{i = g}^{n} \tilde{q}_i x_{\rho,i} \geq l,$$

$$\sum_{i = g}^{n} x_{\rho,i} = s_{\rho}, \quad \rho \in \mathcal{R}, \quad \sum_{\rho \in \mathcal{R}} x_{\rho,i} \leq 1, \quad i \in [g,n], \quad x_{\rho,i} = 0, \quad i \notin \rho, \rho \in \mathcal{R}. \quad (16)$$

be our value function when only products $i \in [g,n]$ are considered. Here, $l \in [L], m = (m_1, m_2) \in [M]^2$, and $g \in [n+1]$, where $L = \left\lceil \frac{n}{\epsilon} \right\rceil - n$ and $M = \left\lceil (n+1)/\epsilon \right\rceil + (n+1)$. In addition, $s = \{s_{\rho}, \rho \in \mathcal{R}\}$, where for any $\rho \in \{\mathcal{S}_o, \mathcal{S}_t\}$, we have $s_{\rho} \in \{0, 1\}$; for $\rho = \mathcal{E}_b$, $s_{\rho} \in [0, \nu^1 - 1]$; finally for $\rho = \mathcal{E}_t$, we have $s_{\rho} \in [0, \nu^2 - \nu^1 - 1]$. Note that $s$ keeps track of the feasibility constraints of the consideration sets. Our ultimate goal is to compute $V_1(L, (M, M), c)$. Later in Proposition 7, we will show that $V_1(L, (M, M), c) \geq L$ when the optimal objective value of problem (14) is greater than $a^1$.

The value function $V_1(L, (M, M), c)$ can be computed recursively. We start with $g = n + 1$ and we set $V_{n+1}(l, m, s) = 0$ if $(m, s) \geq 0$. Otherwise, we set $V_{n+1}(l, m, s)$ to $-\infty$. In addition, the problem is also infeasible when any of $m_1, m_2, s_{\rho}, \rho \in \mathcal{R}$ is negative.\textsuperscript{29}

For any $g \in [n]$, we determine whether or not product $g$ should/contribute to any of the consideration sets feasibility constraints, i.e., $\sum_{i = g}^{n} x_{\rho,i} = s_{\rho}, \rho \in \mathcal{R}$. Also, we should decide to which

\textsuperscript{29}We also set $V_g(l, m, s) = -\infty$ if any of $m_1, m_2, s_{\rho}, \rho \in \mathcal{R}$ is less than zero.
of four aforementioned constraints, product $g$ contributes. By constraints $\sum_{\rho \in \mathcal{R}} x_{\rho, i} \leq 1$, $i \in [n]$, each product $g$ can contribute to at most one of the constraints. To make this decision, we compare the following cases:

- $\rho \in \mathcal{R}^1$: Product $g$ contributes to constraint $\sum_{i=g}^n x_{\rho, i} = s_{\rho}$, i.e., $x_{\rho, g} = 1$, where $\rho \in \mathcal{R}^1$. Then,

$$V_g(l, m, s) = \mathbb{I}(g \in \rho) + \tilde{q}_g^1 + V_{g+1}(l - \tilde{q}_g^2, (m^1 - \tilde{w}_g^1, m^2 - \tilde{w}_g^2), (s_{\rho} - 1, s_{-\rho})),$$

where $\mathbb{I}(g \in \rho) = 0$ when $g \in \rho$, and is $-\infty$ otherwise. $\mathbb{I}(g \in \rho)$ is added to make sure we do not construct an infeasible solution. Also, $s_{-\rho}$ is the vector of $s$ except $s_{\rho}$.

- $\rho \in \mathcal{R}^2 \setminus \mathcal{R}^1$: Product $g$ contributes to constraint $\sum_{i=g}^n x_{\rho, i} = s_{\rho}$; that is, $x_{\rho, g} = 1$ where $\rho \in (\mathcal{R}^2 \setminus \mathcal{R}^1)$. Then,

$$V_g(l, m, s) = \mathbb{I}(g \in \rho) + V_{g+1}(l - \tilde{q}_g^2, (m^1, m^2 - \tilde{w}_g^2), (s_{\rho} - 1, s_{-\rho})).$$

- $\rho \in \emptyset$: Product $g$ does not contributes to any of the constraints and the product is an ignored product, i.e., $x_{\rho, g} = 0$ for any $\rho \in \mathcal{R}$. In this case, we get

$$V_g(l, m, s) = V_{g+1}(l, m, s).$$

Putting these together, we get the following DP.

$$V_g(l, m, s) = \max_{\rho \in \mathcal{R}^1} \left\{ \max \left\{ \mathbb{I}(g \in \rho) + \tilde{q}_g^1 + V_{g+1}(l - \tilde{q}_g^2, (m^1 - \tilde{w}_g^1, m^2 - \tilde{w}_g^2), (s_{\rho} - 1, s_{-\rho})) \right\} , \right. \left. \max_{\rho \in \mathcal{R}^2 \setminus \mathcal{R}^1} \left\{ \mathbb{I}(g \in \rho) + V_{g+1}(l - \tilde{q}_g^2, (m^1, m^2 - \tilde{w}_g^2), (s_{\rho} - 1, s_{-\rho})) \right\} , V_{g+1}(l, m, s) \right\} \quad (17)$$

The term that achieves the maximum determines the value of $x_{\rho, g}$. For instance, if $V_{g+1}(l, m, s)$ obtains the maximum, $x_{\rho, g} = 0$ for any $\rho \in \mathcal{R}$.

We now summarize our FPTAS algorithm.

**ALGORITHM 3:** FPTAS Algorithm for Problem $\text{Rev}(\nu^1, \nu^2)$.

For any $(a, b) \in (A_i)^2 \times (B_i)^2$

- Compute $\tilde{q}_i^k$ and $\tilde{w}_i^k$ for any $i \in [n]$ and $k \in [2]$ according to Eq. (15).

- Using the computed coefficients and via the DP formulation given in Eq. (17), compute $V_1(L, (M, M), c)$.

  - If $V_1(L, (M, M), c) \geq L$, then keep the solution returned by the DP, denoted by $\{x_{\rho, i}, (a, b), i \in [n], \rho \in \mathcal{R}\}$. Otherwise, discard the solution.

  - Let $\mathcal{X}$ be the set of the solutions kept by the algorithm. Return the best solution in $\mathcal{X}$, i.e., the one that obtains the highest objective value in problem (13).

The following proposition proves the second part of Theorem 3.

**Proposition 7 (Performance Guarantee of Algorithm 3).** Algorithm 3 returns a solution to Problem $\text{Rev}(\nu^1, \nu^2)$ with performance guarantee of $(1 - \epsilon)$. Furthermore, the running time of the algorithm is polynomial in the input size of the problem and $\frac{1}{\epsilon}$.
6. Conclusion

Motivated by empirical evidence on consumer search behavior in online platforms, we develop a novel two-stage sequential search model based on a microeconomic foundation. Our model constitutes a two-stage search process; in the first stage, the consumer screens products to learn their intrinsic utility and form a consideration set; in the second stage, she inspects the products in her consideration set to learn the idiosyncratic part of utility. Our two-stage model can capture the externality effect as well as the position-dependent substitution effect between products. Under such a model, we study the rank optimization problem of a platform with the objective of maximizing welfare of its consumers - to retain its long-run market share - or its own short-term revenue. For each objective, we develop optimal or near optimal polynomial-time algorithms. One of the main insights of our work is that, in contrast to general wisdom, ranking products in a decreasing order of their intrinsic utilities does not necessarily maximize consumer welfare. This is due to an exploration-exploitation trade-off that exists because of the idiosyncratic part of utility. We believe that such a trade-off exists in brick and mortar companies as well as online platforms that experience with more sophisticated ways of displaying products. Thus, this work can serve as a starting point to design product display in more complex settings.

References


Daniel McFadden, William B Tye, and Kenneth Train. 1977. An application of diagnostic tests for the independence from irrelevant alternatives property of the multinomial logit model. Institute of Transportation Studies, University of California.


Appendix

EC.1. Case Study

In this section, we present details of the dataset from which we draw empirical evidence on the impact ranking. We presented some of our evidence in Section 1 (Figures 1a and 1b). In this section, we also present further evidence on the impact of ranking on platform’s revenue as well as an estimation model and its results. Due to sensitivity of the data, we are unable to disclose certain details, such as numerical values. In those cases, we present scaled quantities or qualitative description to highlight the main features.

Platform Structure: The dataset comes from an online platform. In this platform, when a user enters a keyword, the platform displays a number of “products” ranging from a few to a few dozens; these products are arranged in a horizontal strip. Products placed on right side of the stripe (worse) positions are usually not visible to the user, and she needs to swipe or scroll in order to observe them. The platform optimizes over the rankings to achieve a complex multi-criteria objective.

Data: Our dataset was collected over 30 consecutive days in Summer 2017; during this period, the platform conducted a randomized experiment that is of particular interest to our study; in this experiment, 0.26% of the traffic is assigned to a treatment group for which the ranking of products is not optimized, but rather is chosen uniformly at random. Another 0.26% is kept as the control group. Note that under random ranking, the products’ position is not a function of their characteristic or past demand, but it is chosen uniformly at random.

Such an experiment enables us to investigate the impact of ranking without being concerned about the endogeneity of positions. If the ranking were not random and were optimized by the platform, it would be rather difficult to separate the impact of ranking on consumers’ choices from those caused by the platform’s ranking algorithm itself. We would also like to highlight that running such an experiment can be quite costly for any platform. Nonetheless, many of online platforms run randomized ranking experiments to effectively train their learning algorithm without suffering from position bias caused by their ranking algorithm. In this section, we mainly focus on the treatment group in which the ranking of the products are random to estimate unbiased effect of ranking on consumers’ choices.

Next, we discuss our dataset in more details. There are more than twelve million queries in the treatment group. We have a similar number of queries in the control group. For each query, we have the following information: (1) category of the searched keyword (2) number of products shown n (3) the ranking π over the products (4) product IDs (5) whether product i in position j was purchased (6) a “popularity” measure for each product that can be viewed as a proxy for its IU.

We now provide additional information about the above fields in our data set.

- In our dataset, the number of categories is 20. We will focus on the category with the highest number of queries that contributes to more than 25% of the total number of queries.

- In most of the queries, the number of products, n, is either 2 or 30. In particular, the percentage of queries with n = 2 and 30 are approximately 23%, and 14%, respectively. However, as expected, the number of queries per product ID is the highest when the number of products is two. Because of that, we will restrict our attention to all the queries for which n = 2.

- The number of unique product IDs is more than thirty five millions. Then, given that the number of queries is around twelve million queries, it is not surprising that some of the products IDs appear only a few times in our dataset.

30 To preserve the identity of the platform, we follow the terminology introduced in Section 3, and use the generic terms of “product” and “purchase”.

EC.1.1. Further Empirical Evidence

In Section 1, we presented our empirical analysis of the impact of ranking and rank optimization on probability of purchase. Here, we also investigate the impact of rank optimization on revenue for platforms that derive revenue from a product sale that is proportional to its price. Such setting does not directly apply to the platform under our study, therefore we do not have data on the revenue of each products. Instead, we assume the revenue of products are i.i.d. samples from a Log-normal distribution with mean 0.5 and σ ranging from 0.05 to 0.2. Figure EC.1b plots the average revenue obtained under optimal, revenue order, and random ranking for Pair 1 (defined in Section 1). The width of each curve represents the mean standard error. This experiment highlights the substantial impact of optimal ranking on revenue: even for 2 products, optimal ranking can substantially increase revenue.

EC.1.2. Estimation

In this section, we briefly describe our estimation methodology and present representative estimation results and counterfactual analysis. We assume that the cost of second-stage search is zero and, as explained in Section 5, the choice model reduces to a random utility choice model. We further assume that $f_z$ is the standard Gumbel distribution and therefore the choice model in the second stage has the form of standard Multinomial Logit (MNL) as given in (8). The fact that the choice model of the consumers in the second stage can be described by the MNL model does not imply that ranking does not play a role in consumer’s choice. Ranking determines the consumer’s consideration set endogenously and the consideration set can impact the consumer’s choice.

As stated earlier, we restrict our attention to all the queries in which the number of products is two and consumers searched for the most popular category. Recall that queries with $n = 2$ have the highest occurrence across all values of $n$. By focusing on all the queries with $n = 2$, the number of queries that we get for each product ID will be also larger. Having access to more data-points makes our estimation more rigorous. Considering this, we identify the top twelve product IDs that have the highest frequency in our dataset. We then find all the queries in which at least one of the aforementioned top product IDs appear. In total, we got more than 2200 unique pairs. From these 2200 pairs, we zoom in to all the pairs that appear more than 1000 times in our dataset. We point out that there are less than 5 pairs that have this property. In the following, we explain how to estimate our model for such a pair.

For a pair of products 1 and 2, the input to the estimation can be summarized in a vector $p = [p_{1,0}, p_{1,1}, p_{1,2}, p_{2,0}, p_{2,1}, p_{2,2}]$, where $p_{i,j}$ represents purchase fraction of product $i = 0, 1, 2$ under ranking $(1, 2)$ and $p_{2,1}$ represents the purchase fraction of product 1 under ranking $(2, 1)$. Note that product 0 represents the no-purchase option.

The estimation in the general form involves estimating consumer-type distribution ($\theta^1, \ldots, \theta^K$), and their associated search costs (or equivalently reservation prices $r_j^k$), as well as the IUs, $u_1$ and $u_2$. The formation of the consideration set in the first stage implies that the consideration sets itself and purchase probabilities only depend on the relation between IUs and the reservation prices, and not the values of the reservation prices. This implies that with three products (the two products plus the no-purchase option), it suffices to determine whether $r_1$ - reservation price for the higher position - belongs to one of four regions: larger than the IU of all three products, between the largest and the second largest IU, between the second largest and the smallest IU, and smaller than the IU of all the products. Assuming one of these cases for $r_1$, we can consider all cases for $r_2$ with the constraint $r_2 \leq r_1$. This results in having ten consumer types. Therefore, in our estimation process, we do not determine the exact values of reservation prices, as they cannot be identified uniquely. Alternatively, we estimate the probability that a consumer belongs to one of the aforementioned types.

31 From now on, by our dataset, we mean all the queries with $n = 2$, which are affiliated with the most popular category.
Figure EC.1 Figure EC.1a shows that probabilities of purchase along with their estimates. The absolute values are hidden for confidentiality, but the ratios are preserved. Figure EC.1b shows the average revenue obtained under optimal, revenue-ordered, and random ranking. The width of each curve represents the mean standard error.

For the above model, we are faced with the problem of estimating a mixture model with consumer type as the latent variable. Utilizing the Expectation Maximization (EM) algorithm, we compute the maximum likelihood estimation of IUs and consumer-type distribution. In Figure EC.1a, we present the estimated purchase probabilities resulted from running the EM algorithm, along with the actual ones for two pairs of products that we call Pair 1 and Pair 2. Pair 1 has already been defined in Section 1. Pair 2 is also among the most frequent pairs shown. The products in Pair 1 are different from those in Pair 2. As mentioned in Section 1, the first product of Pair 1, denoted by Product 1, is significantly more popular than its second product, denoted by Product 2. In Pair 2, however, its first product, that (with a slight abuse of notation) we also call Product 1, is only slightly more popular than its second product. In the figure, the absolute values of the purchase probabilities are hidden for confidentiality, but the ratios are preserved, and they indicate the closeness of our estimated probabilities to the actual ones. We now comment on the estimated IUs which we cannot disclose; consistent with our data on the popularity measure of products - which we use as a proxy for its IU- for both pairs the estimated IU of Product 1 is higher than that of Product 2. Also, as expected, the gap in IUs of Pair 1 is much higher (twice) as that of Pair 2.

Finally, in Table EC.1, we report the estimation error for Pair 1 and 2 in two metrics of mean absolute percentage error (MAPE) and mean absolute error (MAE). Additionally, along with the estimate of our model, we also compute the estimation error for models resulting from two counterfactual hypotheses: (1) Suppose there is no search cost, and the consumer faces a standard choice model. In this case, we estimate the error resulting from fitting an MNL model. (2) Suppose the search cost is not position dependent. This results in a special case of our model in which \( r_k^1 = r_k^2 \) for any type \( k \); we call such a model “Pos-Ind Model”. Compared with these two models, ours results in significantly lower estimation errors. This highlights the fact that even with two products, search costs play an important role in explaining consumers’ choice. Moreover, it implies that consumers’ search costs increase as they move away from the higher position and failing to consider this can lead to a bad estimation. Finally, observe that the Pos-Ind model that consider search costs to some extent performs better than the MNL model that completely ignores them.

\[ \text{MAPE} = \frac{100}{6} \left( \sum_{\ell=1}^{2} \sum_{i=1}^{3} \left| \frac{\hat{p}_{\ell,i} - p_{\ell,i}}{p_{\ell,i}} \right| \right), \quad \text{MAE} = \frac{1}{6} \left( \sum_{\ell=1}^{2} \sum_{i=1}^{3} |\hat{p}_{\ell,i} - p_{\ell,i}| \right). \]
Table EC.1  Estimation error in two metrics of MAPE and MAE.

<table>
<thead>
<tr>
<th>Model</th>
<th>Pair 1</th>
<th></th>
<th>Pair 2</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MAPE</td>
<td>MAE</td>
<td>MAPE</td>
<td>MAE</td>
</tr>
<tr>
<td>Our Two-stage Model</td>
<td>3.8%</td>
<td>6.9e-3</td>
<td>3.2%</td>
<td>1.6e-3</td>
</tr>
<tr>
<td>Pos-Ind Model</td>
<td>17.0%</td>
<td>1.8e-2</td>
<td>14.7%</td>
<td>1.0e-2</td>
</tr>
<tr>
<td>MNL</td>
<td>21.0%</td>
<td>2.1e-2</td>
<td>20.9%</td>
<td>1.1e-2</td>
</tr>
</tbody>
</table>

**EC.2. Further Remarks**

**Remark EC.1** (Assumption $r^k_{ν+1} > 0$). In Section 4 and 5, when dealing with a constrained optimization with constraints of the form $|C_ν^k| = ν$, $k ∈ [K]$, we assume that $r^k_{ν+1} > 0$. We note that assuming $r^k_{ν} > 0$ is necessary; otherwise the consumer stops before position $ν$, independent of ranking. However, we can have $|C_ν^k| = ν$ for the case $r^k_{ν+1} ≤ 0 < r^k_{ν}$. Our formulation extends to such a setting after a small modification: Here there is no constraint on the product in position $ν$; because the consumer will stop at position $ν$ because she has a product, i.e., the outside option, whose IU is larger than the next reservation price, i.e., $r^k_{ν+1}$. Therefore, independent of the IU of the product at position $ν$, she will stop her screening. However, we still need to make sure that the products in positions 1, 2, ..., $ν − 1$ were engaging. Therefore we can modify Proposition 1 as follows:

**Proposition EC.1.** Let $C ⊆ [n]$ be a subset with size $ν$. For a consumer of type $k$, suppose $r^k_{ν+1} ≤ 0 < r^k_{ν}$. Set $C$ is an implementable consideration set iff it is induced by a ranking with the following structure: Position $j ∈ [ν]$ is filled by a product from set $C$, and

- *Engaging Products:* Position $j ∈ [ν − 1]$ is filled by a product whose IU belongs to engaging region $E := (−∞, r^k_{ν})$.

Proposition 2, and constraints for the feasible region (5) can be modified in a similar way.

**Remark EC.2** (Welfare maximization with $K > 2$). Solving the optimal ranking problem with more than two types involves similar steps; for every type $k$ we define an engaging region $E_{[ν]} \setminus [k−1]$ and a stopping one $S_k$, where a product that is engaging for type $k$ is also engaging for types $k + 1, ..., K$. If $ν^k < ν^{k+1}$, then the stopping product of type $k$ must be still engaging for type $k + 1$. We can construct $2Kn$ binary variables to encode the at most $(2K + 1)n + 2K$ constraints similar to constraints (5), as well as the weights used to build an appropriate objective function.

**EC.3. Proof of the Results in Sections 3 and 4**

**Proof:** Proof of Lemma 2

By Eq. (1), consumer of type $k$ screens positions 1, 2, ..., $ν$ if

$$\max_{j' ∈ [0, j]} \{u_π(j')\} < r^k_{j+1}, \quad j ∈ [ν − 1].$$

Considering the fact that $r^k_{1} ≥ r^k_{2} ≥ ... ≥ r^k_{n}$, the aforementioned condition implies that

$$\max_{j' ∈ [0, ν − 1]} \{u_π(j')\} < r^k_{ν}. \quad \text{(EC.1)}$$

In the following, we argue that consumer of type $k + 1$ also screens the products in the first $ν$ positions. For this to happen, we need to have $\max_{j' ∈ [0, ν − 1]} \{u_π(j')\} < r^{k+1}_{ν}$. This holds because of Eq. (EC.1) and the fact that $s^k_{ν} ≥ s^{k+1}_{ν}$. The latter implies that $r^k_{ν} ≤ r^{k+1}_{ν}$.

**EC.3.1. Proof of Proposition 1**

We first show that for a ranking that has the structure described in the proposition, the consideration set is $C$. To show this, we need to argue that the consumer only screens the first $ν$ products. By equations (1) and (EC.1), this happens if

$$\max_{j' ∈ [0, ν − 1]} \{u_π(j')\} < r^k_{ν}, \quad \max_{j' ∈ [0, ν]} \{u_π(j')\} ≥ r^k_{ν + 1}.$$
Under assumption $r_{\nu+1}^k > 0$, the conditions simplify to:

$$\max_{j' \in [\nu-1]} \{u_{\pi(j')}\} < r_{\nu}^k, \quad \max_{j' \in [\nu]} \{u_{\pi(j')}\} \geq r_{\nu+1}^k. \quad (EC.2)$$

We note that the first condition holds because by the imposed structure, the first $\nu - 1$ products belong to region $\mathcal{E}$, i.e., interval $(-\infty, r_{\nu}^k)$. Further, the second condition holds because the product in position $\nu$ belongs to region $\mathcal{S}$, i.e., interval $[r_{\nu+1}^k, \infty)$.

Next, we show that if set $\mathcal{C}$ is implementable, then there exists a ranking that has the following properties: (1) It induces the consideration set $\mathcal{C}$, and (2) it has the structure described in the proposition. To do so, we construct ranking $\pi_1$ and we show that this ranking has the aforementioned two properties. Ranking $\pi_1$ fills position $\nu$ with product $i_h = \arg\max_{i \in \mathcal{C}} \{u_i\}$, i.e., the product with the highest IU. Ranking $\pi_1$ fills position $j \in [\nu - 1]$ with products $\mathcal{C} \setminus \{i_h\}$ in an arbitrary order. We will show that ranking $\pi_1$ has the structure stated in the proposition, i.e.,

$$i_h \in \mathcal{S} \quad \text{and} \quad i \in \mathcal{E}, \quad i \in \mathcal{C} \setminus \{i_h\}. \quad (EC.3)$$

That is, ranking $\pi_1$ has the second property. By showing this, it is easy to argue that $\pi_1$ stops the consumer at position $\nu$, and thus, it also satisfies the first property.

We show this by contradiction. Suppose that one of the conditions in Eq. (EC.3) does not hold. We argue that set $\mathcal{C}$ is not implementable.

1. Suppose $i_h \notin \mathcal{S}$; then $u_i < r_{\nu+1}^k$, $i \in \mathcal{C}$. Then, by Eq. (EC.2), regardless on how products in $\mathcal{C}$ fill the first $\nu$ positions, when the consumer reaches position $\nu$, she does not stop.

2. Suppose there exists a product $i \neq i_h$ such that $i \notin \mathcal{E}$; this implies $u_{i_h} \geq u_i \geq r_{\nu}^k$. Regardless of how products in $\mathcal{C}$ fill the first $\nu$ positions, either $i$ or $i_h$ will be placed in a position $j \leq \nu - 1$. Therefore the consumer will stop before reaching position $\nu$.

Proof: Suppose there exists a product $i \neq i_h$ such that $i \notin \mathcal{E}$; this implies $u_{i_h} \geq u_i \geq r_{\nu}^k$. Regardless of how products in $\mathcal{C}$ fill the first $\nu$ positions, either $i$ or $i_h$ will be placed in a position $j \leq \nu - 1$. Therefore the consumer will stop before reaching position $\nu$.

Proof: The proof follows the exact same steps as that of Proposition 1, and therefore omitted.

**EC.3.2. Proof of Proposition 2**

The proof of Lemma 3 and Proposition 3 are the same. So, in the following, we only show the result in Lemma 3.

Recall that the consumer welfare is a weighted sum of the purchase probability and consumer’s net value. Thus, in the following, we show that both purchase probability and consumer’s net value are higher under solution $x := \{x_{\rho,i} : \rho \in \mathcal{R}, i \in [n]\}$ than those under solution $\tilde{x} := \{\tilde{x}_{\rho,i} : \rho \in \mathcal{R}, i \in [n]\}$.

**The purchase probability:** We start with presenting the purchase probability in a closed form when the consumer’s consideration set is $\mathcal{C}$. The purchase probability depends on the search process of the consumer in the second stage of the search process. Although, this stage has a dynamic nature, as shown by Kleinberg et al. (2016). Armstrong (2016), the final choice model can easily be expressed as a static discrete choice problem. In particular, the consumer with consideration set $\mathcal{C} = \{i_1, i_2, \ldots, i_{|\mathcal{C}|}\}$ purchases product $i_{j^*}$ where $j^* = \arg\max_{j \in [0, |\mathcal{C}|]} \{\min\{u_{i_j} + z_{i_j}, r_{i_j}\}\}$. Note that to find out what product will be purchased, we do not need to know in what order the consumer inspects the products and when she stops her inspection. The consumer always purchases a product in her consideration set ($\mathcal{C} \cup \{0\}$) that has the highest “adjusted utility” of

$$\min\{u_i + z_i, r_i\}.$$

By Lemma 1, the adjusted utility for any product $i \in [n]$ is $\min\{u_i + z_i, u_i + r_Z\}$ and is $z_0$ for the no-purchase option. Then, the total purchase probability of a consumer is given by

$$1 - \text{Pr} [Z_0 > \max_{i \in \mathcal{C}} \{u_i + \min\{Z_i, r_Z\}\}] \quad (EC.4)$$
Observe that the purchase probability is increasing in the IU of the products in the consideration set. This implies that for any type of consumer, solution $x$ outperforms $\tilde{x}$ in terms of the purchase probability. Recall that the consideration set under solution $x$ is either the same as that under $\tilde{x}$, or they differ in one product. If different, the consideration set of $x$ has product $i_1$, and the consideration set of $\tilde{x}$ has product $i_2$, where $u_{i_1} > u_{i_2}$.

**The consumer’s net value:** In the following, we show that the consumer’s net value is higher under solution $x$. With a slight abuse of notation, let the consideration set of type $k$ under solution $x$ and $\tilde{x}$ be $C_k(x)$ and $C_k(\tilde{x})$, respectively. We know that either $C_k(x) = C_k(\tilde{x})$ or $C_k(x) = C_k(\tilde{x}) \setminus \{i_1\} \cup \{i_2\}$. The case of $C_k(x) = C_k(\tilde{x})$ is trivial. Thus, we assume that $C_k(x) = C_k(\tilde{x}) \setminus \{i_1\} \cup \{i_2\}$.

Let us fix the realization of idiosyncratic part of the utility of the products in the consideration sets such that the idiosyncratic part of the utility of all the products in set $C_k(x) \setminus \{i_1\} = C_k(\tilde{x}) \setminus \{i_2\}$ is the same and the idiosyncratic part of the utility of products $i_1$ and $i_2$ is also the same. We show that for any realization of idiosyncratic parts, the consumer with consideration set $C_k(x)$ earns higher net value than the consumer with consideration set $C_k(\tilde{x})$. Assume that under set $C_k(\tilde{x})$, the consumer follows her optimal order and inspects the $J$ products with the highest IU in her consideration set. Let $J'$ denote the set of such products. We distinguish two cases:

**case 1:** Suppose $i_2 \in J'$; now assume the consumer with consideration set $C_k(x)$ follows the same order, and inspects $J \setminus \{i_2\} \cup \{i_1\}$. Because $u_{i_1} \geq u_{i_2}$, her net value is greater than or equal to that of the consumer with $C_k(\tilde{x})$.

**case 2:** Suppose $i_2 \notin J'$; now assume the consumer with consideration set $C_k(x)$ follows the same order, and inspects $J'$. Her net value is equal to that of the consumer with $C_k(\tilde{x})$.

We complete the proof by noting that the consumer with consideration set $C_k(x)$ can only increase her net value by taking an optimal inspection decision.

**EC.3.3. Proof of Proposition 4**

**Part (1) - The ILP returns the Optimal Solution:** We show the result by contradiction. Let $x^* = \{x^*_{p,i}, x_p \in R, i \in [n]\}$ be the optimal solution returned by the ILP, that induces consideration set $C^{\star k}$, $k \in [2]$. Suppose, contrary to our claim, that there exists another feasible solution $x = \{x_{p,i}, x_p \in R, i \in [n]\}$ that obtains strictly higher consumer welfare that solution $x^*$. Then, by Proposition 3 and Lemma 3, there should exist region $\rho \in R$ such that $x_{p,i_2} = 1$, $x^*_{p,i_1} = 0$, $x_{p,i_2} = 0$, and $x^*_{p,i_2} = 1$ where $u_{i_1} > u_{i_2}$. That is, in the ILP solution product $i_2$ contributes to region $\rho$ and in the other solution, product $i_1$ contributes to region $\rho$.

We consider the following two cases:

- $i_1 \in C^{\star 2}$: Here, in the LP solution, product $i_1$ is in the consideration set of type two consumers and contributes to one of the regions $\rho \in \{E_i, S_i\}$. Let us denote this region by $\rho_i$. Whereas, in the other solution, product $i_1$ is in the consideration set of both types and contributes to one of the regions $\rho \in \{E_b, S_b\}$, denoted by $\rho_o$. This implies that $i_1 \in \rho_i, \rho_o$ and $i_2 \notin \rho_o$.

By construction, the LP breaks the tie in favor of regions/constraints $\rho \in \{E_b, S_b\}$ unless all other products that already contributed to these regions have higher IU. The fact that in assigning product $i_1$, LP breaks the tie in favor of region $\rho_i \in \{E_i, S_i\}$ shows that all the products that contribute to region $\rho_o \in \{E_o, S_o\}$ in the LP solution have IU higher than $u_{i_1}$. As a result, in the LP solution, there does not exist any product $i_2$ such that $u_{i_2} < u_{i_1}$ and $x^*_{p_o,i_2} = 1$. This gives us a contradiction.

- $i_1 \notin C^{\star 2}$: In that case, in the LP solution, product $i_1$ is ignored; that is, $x_{p,i_1} = 0$ for any $\rho \in R$. In the following, we construct another feasible solution for which the objective function of the LP is higher than solution $x^*$. This contradicts that $x^*$ is the optimal solution of the LP.

Let region $\rho_L$ be the region that product $i_2$ contributes to in the LP solution. Define a new solution $\tilde{x}^* = \{\tilde{x}^*_{p,i}, x_p \in R, i \in [n]\}$ such that $\tilde{x}^*_{p,i_1} = 1$, $\tilde{x}^*_{p,i_2} = 0$, and $\tilde{x}^*_{p,i} = x^*_{p,i}$ when $(\rho, i) \neq (\rho_L, i_1)$ and $(\rho, i) \neq (\rho_L, i_2)$. Observe that $\tilde{x}^*$ is a feasible solution of the LP because $i_1 \notin \rho_L$. In addition, the objective function of the LP is higher under $\tilde{x}^*$ than $x^*$, as $u_{i_1} > u_{i_2}$. This contradicts that $x^*$ is the optimal solution of the LP.
**Part (2) - The ILP Can be Solved Effectively:** We will show that the ILP can be solved effectively by showing its LP relaxation, in which $x_{p,i} \in [0, 1]$, admits an integral solution. Note that if the ILP has the form of $\max c^T x$ such that $Ax = b$, where $A$, $b$, and $c$ have all integer entries and matrix $A$ is totally unimodular, then the solution returned by the LP relaxation is integral. In our ILP, $A$, $b$, and $c$ are integers. Thus, it suffices to write our constraints in the form of $Ax = b$ and show that $A$ is totally unimodular.

Let $y_{p,i} = 1$ if $i \in p$ and zero otherwise. Then, we can rewrite Problem ILP as follows.

$$\max \sum_{i \in [n], \rho \in \mathcal{R}} \omega_{p,i} x_{p,i}$$

s.t.

$$\sum_{i \in [n]} x_{p,i} y_{p,i} = c_p, \rho \in \mathcal{R}$$

$$\sum_{\rho \in \mathcal{R}} x_{p,i} y_{p,i} + a_i = 1, \quad i \in [n],$$

(EC.5)

Let $A$ be the constraint matrix associated with the above ILP, where the $i$-th row of $A$ is associated with the $i$-th constraint. Dantzig (1956) showed that if matrix $A$ satisfies the following conditions, it is totally unimodular. We will make use of this result to show that our constraint matrix $A$ is totally unimodular.

Let $A$ be a matrix whose rows can be partitioned into two disjoint sets $B$ and $C$. Then the following four conditions together are sufficient for $A$ to be totally unimodular:

- Every entry in $A$ is 0, +1, or -1;
- Every column of $A$ contains at most two non-zero entries;
- If two non-zero entries in a column of $A$ have the same sign, then the row of one is in $B$, and the other in $C$;
- If two non-zero entries in a column of $A$ have opposite signs, then the rows of both are in $B$, or both in $C$.

We will show that these four conditions are satisfied. First of all, all the elements in $A$ are either one or zero. Second, every column of $A$ contains at most two non-zero elements. To see why, note that if product $i$ belongs to region $\rho$, i.e., if $y_{p,i} = 1$, then, the column associated to $x_{p,i}$ has two non-zero entries both equal to +1. When $y_{p,i} = 0$, the column associated to $x_{p,i}$ is all zero. Also, the column associated to $a_i$ has only one non-zero element equal to +1. Finally, we partition the rows of $A$ as follows: set $B$ contains the first four rows, and $C$ contains the rest. As mentioned above, the only columns containing two non-zero entries are those associated with a region and a product $(\rho, i)$ such that $y_{p,i} = 1$. One of the +1 entries appears in set $B$, and the other one in set $C$. Given that all the above four conditions are satisfied, we conclude that matrix $A$ is totally unimodular.

**Running Time of Our Algorithm:** Our LP, which solves Problem $Wel(\nu^1, \nu^2)$, has $5n$ variables. Thus, its running time is in the order of $O(n^{3.5})$. Recall that we need to solve $Wel(\nu^1, \nu^2)$ for every $\nu^1, \nu^2 \in [n]$ in order to Problem $Wel$. This implies that the complexity of solving Problem $Wel$ is $O(n^{5.5})$.

EC.3.4. Proof of Proposition 5

By Proposition 1, any implementable consideration set has at least one product from region $S$ to be placed in position $\nu$, and $\nu - 1$ different products from region $E$ to be placed in position $j \in [\nu - 1]$. Then, the result follows from Lemma 3, where we show that we need to break the ties in favor of products with higher IU.
EC.3.5. Proof of Additional Lemmas

Proof: Proof of Lemma 4 The proof follows from the Pandora’s box stopping rule and the definition of reservation prices.

Proof: Proof of Lemma 4 Let \( C_k(x) \) and \( C_k(\tilde{x}) \) be the consideration sets of consumers of type \( k \in [2] \) under solution \( x \) and \( \tilde{x} \). By definition of solutions, \( C_2(x) = C_2(\tilde{x}) \) and \( C_1(x) = C_1(\tilde{x}) \setminus \{i_2\} \cup \{i_1\} \). Proposition 3 and the fact that \( u_{i_1} \geq u_{i_2} \) imply that under solution \( x \), consumers of type one obtain higher net value than under solution \( \tilde{x} \).

EC.4. Proof of the Results in Section 5

EC.4.1. Proof of Proposition 6

By Eq. (11) and definition of \( h_i(\cdot) \)'s, we have

\[
\text{Rev}(\nu) = \max \left\{ \lambda \in \mathbb{R} : \exists \pi, |C_{\pi}| = \nu, \sum_{j=\pi(j)\in C_{\pi}} h_{\pi(j)}(\lambda) \geq \lambda \right\}.
\]

Thus, for a given \( \lambda \), the optimal ranking can be obtained by solving the Problem \( \text{Rev}(\nu, \lambda) = \max_{|C_{\pi}|=\nu} \sum_{j=\pi(j)\in C_{\pi}} h_{\pi(j)}(\lambda) \). The optimal solution of Problem \( \text{Rev}(\nu, \lambda) \) depends on the order of \( h_i(\lambda) \)'s, and as long as the order of \( h_i(\lambda) \)'s remains the same, the optimal ranking does not change. Based on this observation, it suffices to only solve Problem \( \text{Rev}(\nu, \lambda_l) \), for any \( l \in [L+1] \). Recall that using the intersection points of \( h_i(\cdot) \)'s, we constructed \( L+1 \) non-overlapping intervals where the order of \( h_i(\cdot) \)'s did not change in each interval. Also, point \( \lambda_l, l \in [L+1] \), is an arbitrary point in interval \( l \).

In order to solve \( \text{Rev}(\nu, \lambda_l) \), we need to return an implementable consideration set that maximizes \( \text{Rev}(\nu, \lambda_l) \). By proposition 1, a consideration set is implementable iff there exists a ranking such that its stopping product, i.e., the product in position \( \nu \), belongs to region \( \mathcal{S} \) and its engaging products, products in positions \( j = 1, 2, \ldots, \nu - 1 \), belong to region \( \mathcal{E} \). We note that sets \( \mathcal{S} \) and \( \mathcal{E} \) overlap. Thus, if we choose a product from \( \mathcal{S} \cap \mathcal{E} \) as the stopping product, that product can no longer be used as an engaging product. Based on this observation, we need to consider two rankings. In the first ranking, we choose a product \( i^* \) from region \( \mathcal{S} \setminus \mathcal{E} \) that has the highest \( h_i(\lambda_l) \) as the stopping product, i.e., \( i^* = \arg\max_{i \in \mathcal{S} \setminus \mathcal{E}} \{h_i(\lambda_l)\} \). Note that this is the best we can do assuming that the stopping product should be chosen from \( \mathcal{S} \setminus \mathcal{E} \). Also, choosing this product as the stopping product does not limit our options for the engaging products. For the engaging products, we choose \( \nu - 1 \) products with the highest \( h_i(\lambda_l) \) from region \( \mathcal{E} \). This ensures that \( \text{Rev}(\nu, \lambda_l) \) is maximized. In the second ranking, we choose the stopping product from set \( \mathcal{S} \cap \mathcal{E} \). For the stopping product, we select product \( i^{**} = \arg\max_{i \in \mathcal{S} \cap \mathcal{E}} \{h_i(\lambda_l)\} \). For the engaging products, we select \( \nu - 1 \) products with the highest \( h_i(\lambda_l) \) from set \( \mathcal{E} \setminus \{i^{**}\} \). The best of the two rankings gives us the optimal ranking.

EC.4.2. Proof of the First Part of Theorem 3

In the following we prove the first part of Theorem 3. The second part is proven through Proposition 7 in the next section. To prove the first part, we consider an instance of our problem that reduces to the setting of Gallego et al. (2016). For their setting, Gallego et al. (2016) established an equivalence between solving the revenue maximization problem and an instance of PARTITION problem that is a well-known NP-complete problem.

For a given integer \( d \), define \( p = 200d^2 \). Suppose the number of products is \( n = 2(p + 2d) \). Further, suppose the population is comprised of two types of consumers with distribution \( (\theta^1, \theta^2) = (0.5, 0.5) \) and the following cost structure:

\[
s^1_j = \begin{cases} 
0 & 1 \leq j \leq p \\
+\infty & p + 1 \leq j \leq n
\end{cases} \quad s^2_j = \begin{cases} 
0 & 1 \leq j \leq 2p \\
+\infty & 2p + 1 \leq j \leq n
\end{cases}
\]
For such cost structure, under any ranking, type 1 consumers screen the first $p$ products, while type 2 consumers screen the first $2p$ products. Therefore the ranking problem in (9) simplifies to selecting two subsets of products; subset $C^1$ with size $p$ that constitutes that consideration set of type 1 (for any ranking) and subset $C^2$ with size $2p$ and $C^1 \subseteq C^2$ that constitutes that consideration set of type 2. Therefore we have:

$$\text{Rev} = \max_{c_1, c_2 : |c_1| = p, |c_2| = 2p, C^1 \subseteq C^2} \sum_{k \in [2]} 0.5 \frac{\sum_{i \in C^k} q_i w_i}{1 + \sum_{i \in C^k} w_i} \quad (EC.6)$$

Gallego et al. (2016) presents a reduction from the PARTITION problem to an instance of the revenue maximization as posed in (EC.6). Given that the PARTITION problem is known to be NP-complete, using their instance, we conclude that the ranking problem of (9) is NP-complete.

**EC.4.3. Proof of Proposition 7**

We start with the following claim.

**Claim:** For any $(a, b) \in (A)^2 \times (B)^2$, if there exists a feasible solution $\{x_{\rho,i} : \rho \in R, i \in [n]\}$ to problem (14) whose objective value, i.e., $\sum_{\rho \in R^1} \sum_{i \in [n]} \theta^1 q_i w_i x_{\rho,i}$ is greater than $a^1$, then Problem $V_i(L, (M, M), c)$ has a feasible solution, denoted by $\{x_{\rho,i}(a, b) : \rho \in R, i \in [n]\}$ whose objective value $V_i(L, (M, M), c) \geq L$. In addition,

$$\sum_{\rho \in R^k} \sum_{i \in [0, n]} w_i x_{\rho,i}(a, b) \leq b^k(1 + 2\epsilon), \quad \sum_{\rho \in R^k} \sum_{i \in [n]} \theta^k q_i w_i x_{\rho,i}(a, b) \geq a^k(1 - 2\epsilon), \quad k \in [2] \quad (EC.7)$$

**Proof of Claim:** Consider $\{x_{\rho,i} : \rho \in R, i \in [n]\}$ that is a feasible solution of problem (14). By the feasibility of the solution and by our assumption that $\sum_{\rho \in R^1} \sum_{i \in [n]} \theta^1 q_i w_i x_{\rho,i} \geq a^1$, for any $k \in [2]$, we have

$$\sum_{\rho \in R^k} \sum_{i \in [0, n]} \frac{w_i x_{\rho,i}}{\epsilon b^k/(n + 1)} \leq \frac{n + 1}{\epsilon}, \quad \sum_{\rho \in R^k} \sum_{i \in [n]} \frac{\theta^k q_i w_i x_{\rho,i}}{\epsilon a^k/n} \geq \frac{n}{\epsilon}$$

After rounding down and up and by definition of normalized coefficients given in Eq. (15), for $k \in [2]$ we get

$$\sum_{\rho \in R^k} \sum_{i \in [0, n]} \tilde{w}_i \leq \left\lceil \frac{n + 1}{\epsilon} \right\rceil + n + 1 = M, \quad \sum_{\rho \in R^k} \sum_{i \in [n]} \tilde{q}_i^k \geq \left\lfloor \frac{n}{\epsilon} \right\rfloor - n = L$$

This implies that the there exists a feasible solution $\{x_{\rho,i}(a, b) : \rho \in R, i \in [n]\}$ to problem $V_i(L, (M, M), c)$ whose objective value is greater than or equal to $L$. Next we show that Eq. (EC.7) holds at this feasible solution. Since $\{x_{\rho,i}(a, b) : \rho \in R, i \in [n]\}$ is a feasible solution of problem $V_i(L, (M, M), c)$ and $V_i(L, (M, M), c) \geq L$, we get

$$\sum_{\rho \in R^k} \sum_{i \in [0, n]} w_i x_{\rho,i}(a, b) \leq M \frac{\epsilon b^k}{(n + 1)} \leq b^k(1 + 2\epsilon), \quad k \in [2]$$

$$\sum_{\rho \in R^k} \sum_{i \in [n]} \theta^k q_i w_i x_{\rho,i}(a, b) \geq L \frac{\epsilon a^k}{n} \geq a^k(1 - 2\epsilon), \quad k \in [2]$$

The above inequalities complete the proof of the claim. Q.E.D.
Next, we use the claim to show that Algorithm 3, returns a \((1 - \epsilon)\)-optimal solution to Problem \(\text{Rev}(\nu^1, \nu^2)\), defined in Eq. (13). Let \(\{x_{\rho,i}^*, \rho \in \mathcal{R}, i \in [n]\}\) be the optimal solution of Problem \(\text{Rev}(\nu^1, \nu^2)\). Then, there exists \(\alpha, \beta\) such that

\[
\nu q(1 + \epsilon)^{\alpha k} \leq \sum_{\rho \in \mathcal{R}^k} \sum_{i \in [n]} \theta^k q_i w_i x_{\rho,i}^* \leq \nu q(1 + \epsilon)^{\alpha k + 1} \quad k \in [2]
\]

\[
(1 + \epsilon)^{\beta k} \leq \sum_{\rho \in \mathcal{R}^k} \sum_{i \in [0,n]} w_i x_{\rho,i}^* \leq (1 + \epsilon)^{\beta k + 1} \quad k \in [2]
\]

where \(\alpha = (\alpha_1, \alpha_2)\) and \(\beta = (\beta_1, \beta_2)\). Let \(\nu q(1 + \epsilon)^{\alpha} = (\nu q(1 + \epsilon)^{\alpha_1}, \nu q(1 + \epsilon)^{\alpha_2})\) and \((1 + \epsilon)^{\beta} = (w(1 + \epsilon)^{\beta_1}, w(1 + \epsilon)^{\beta_2})\). Then, by the claim, for \((a, b) = (\nu q(1 + \epsilon)^{\alpha}, w(1 + \epsilon)^{\beta})\), set \(X\), returned by the algorithm, contains solution \(\{x_{\rho,i}(a, b) : \rho \in \mathcal{R}, i \in [n]\}\) that satisfies Eq. (EC.7). This implies that the objective value of Problem \(\text{Rev}(\nu^1, \nu^2)\) at this solution fulfills

\[
\sum_{k \in [2]} \sum_{\rho \in \mathcal{R}^k} \sum_{i \in [n]} \theta^k q_i w_i x_{\rho,i}(a, b) \geq \frac{1 - 2\epsilon}{1 + 2\epsilon} \sum_{k \in [2]} \sum_{\rho \in \mathcal{R}^k} \sum_{i \in [n]} \theta^k q_i w_i x_{\rho,i}^* \\
\geq (1 - 4\epsilon) \sum_{k \in [2]} \sum_{\rho \in \mathcal{R}^k} \sum_{i \in [n]} \theta^k q_i w_i x_{\rho,i}^*
\]

Next, we show the running time of Algorithm 3 is polynomial in the input size of Problem \(\text{Rev}(\nu^1, \nu^2)\) and \(\frac{1}{\epsilon}\). For each \((a, b) \in (A^\nu)^2 \times (B^\nu)^2\), we need to solve a dynamic programming with \(O(\frac{q^k}{\epsilon})\) states. Then, considering the fact that the number of \(a^k \) and \(b^k\), \(k \in [2]\), that we consider in the algorithm is \(A = O(\log(nq w/(qw))/\epsilon)\) and \(B = O(\log((n+1)\bar{w}/q)/\epsilon)\), respectively, the running time of Algorithm 3 is in the order of \(O((\log(nq w/(qw)))^2(\log((n+1)\bar{w}/q))^2\frac{q^k}{\epsilon^2})\).

**EC.5. Suboptimality of Algorithm 1 in a Multi-type Population**

In the following, we revisit Example 1 to illustrate the suboptimality of Algorithm 1 when applied to a multi-type population.

**Example EC.1.** Consider the instance presented in Example 1 in which the optimal ranking, \((5, 2, 3, 1, 4)\), leads to consideration sets of \((2, 5)\), and \((1, 2, 3, 5)\) for types 1 and 2, respectively. Now suppose we solve the welfare maximization problem for the single-type problems for both types. For type 1, under the constraint of \(\nu^1 = 2\), an optimal ranking is \((5, 1, 3, 4, 2)\) and the consideration set of the consumer is \(\{1, 5\}\). For type 2 under constraint \(\nu^2 = 4\), an optimal ranking and its associated consideration set are \((2, 3, 4, 1, 5)\) and \((1, 2, 3, 4)\), respectively. Clearly, solution to neither single-type problem coincides with the optimal two-type solution.

**EC.6. An Example for Suboptimality of Revenue-ordered Ranking**

**Example EC.2 (Suboptimality of Revenue-ordered Ranking).** Consider the instance of example 1, and suppose the revenue of products \(i = 1, 2, \ldots, 5\), i.e., \(q_i\), is given by \(8.4, 15.8, 9.4, 4, 15.6\), respectively. The revenue-maximizing ranking is \((5, 2, 1, 4, 3)\) with revenue of 11.47 while the revenue-ordered ranking\(^{33}\) \((2, 5, 3, 1, 4)\) achieves revenue of 10.39. Thus, the revenue-ordered ranking is suboptimal with optimality gap of 9.4%.

To gain further insight we study the consideration sets induced by both rankings for both types. The consideration sets under the optimal ranking for types 1 and 2 are \((2, 5)\) and \((1, 2, 5)\) respectively. While under revenue-ordered ranking consideration sets are \((2)\) and \((1, 2, 3, 5)\).

\(^{33}\) In the revenue-ordered ranking, products are placed in positions 1, 2, \ldots, \(n\) in decreasing order of their revenues, i.e., the product with the highest revenue is placed in position 1.
The optimal ranking make use of product 5, which is a high revenue product with low IU that does not impose excessive externality on position 2; this way, by placing product 5 at position 1, and product 2 (the highest revenue product) at position 2 the optimal ranking enlarges the consideration set of type 1 consumer. While the revenue-ordered ranking fails to include product 5 in the type 1 consideration set.