# Assortment Optimization and Pricing under the Multinomial Logit Model with Impatient Customers: Sequential Recommendation and Selection

Pin Gao*[†1], Yuhang Ma*[‡2], Ningyuan Chen[†3], Guillermo Gallego[†1], Anran Li[†4], Paat Rusmevichientong[‡5], Huseyin Topaloglu[‡2]

[1]Department of Industrial Engineering and Decision Analytics, HKUST, Clear Water Bay, Hong Kong
[2]School of Operations Research and Information Engineering, Cornell Tech, New York, NY 10044, USA
[3]Department of Management, University of Toronto Mississauga, Mississauga, ON, Canada
[4]Department of Management, London School of Economics, London, WC2A 2AE, UK
[5]Marshall School of Business, University of Southern California, Los Angeles, CA 90089, USA

pgao@connect.ust.hk, ym367@cornell.edu, ningyuan.chen@utoronto.ca, ggallego@ust.hk, a.li26@lse.ac.uk, rusmevic@marshall.usc.edu, topaloglu@orie.cornell.edu

We develop a variant of the multinomial logit model with impatient customers and study assortment optimization and pricing problems under this choice model. In our choice model, a customer incrementally views the assortment of available products in multiple stages. The patience level of a customer determines the maximum number of stages in which she is willing to view the assortments of products. In each stage, if the product with the largest utility provides larger utility than a minimum acceptable utility, which we refer to as the utility of the outside option, then the customer purchases that product right away. Otherwise, the customer views the assortment of products in the next stage, as long as her patience level allows her to do so. Under the assumption that the utilities have the Gumbel distribution and are independent, we give a closed-form expression for the choice probabilities. For the assortment optimization problem, we develop a polynomial-time algorithm to find the revenue-maximizing sequence of assortments to offer. For the pricing problem, we show that if the sequence of offered assortments is fixed, then we can solve a convex program to find the revenue-maximizing prices, where the decision variables are the probabilities that a customer will reach different stages. We build on this result to give a 0.878-approximation algorithm, when both the sequence of assortments and the prices are decision variables. We consider the assortment optimization problem when each product occupies some space and there is a constraint on the total space consumption of the offered products. We give a fully polynomial-time approximation scheme for this constrained problem. We use a dataset from Expedia to demonstrate that incorporating patience levels, as in our model, can improve purchase predictions. We also check the practical performance of our approximation schemes, in terms of both the quality of solutions and the computation times. *Dated October 12, 2020.*

## 1. Introduction

A common assumption in traditional revenue management models is that each customer enters the system with the intention to purchase a particular product. If this product is available for purchase, then the customer purchases it. Otherwise, the customer leaves the system without a purchase. In many settings, however, the customers observe the assortment of available products and choose and substitute within this assortment, based on the features and prices of the offered products. In this case, the demand for a product depends on the availability of other products, along with their features and prices. In revenue management research, there has been a recent surge in using

*P. Gao and Y. Ma were graduate students at the time of this work. They are listed alphabetically, so are the remaining authors. The work in this paper is independent discovery by two different sets of authors, each indicated with † and ‡.

discrete choice models to capture the fact that the customers choose and substitute among the available products. A large portion of these models work under the assumption that the customers view the entire assortment of products offered to them simultaneously, but it is clear that in many cases, the customers incrementally view the assortment of offered products and make a purchase decision before viewing all the offered products. When purchasing products in online retail, for example, a customer may view the assortment of offered products on multiple webpages. When booking a healthcare appointment on the phone, the patient may be offered appointment slots gradually until she makes a choice. In both of these examples, the customer may make a purchase or leave without a purchase before viewing all the offered products or appointment slots. When the customers view the assortment of offered products incrementally, the question is not only what assortment of products to offer but also in which sequence to offer them.

We propose a variant of the multinomial logit model where the customers incrementally view the assortment of offered products in multiple stages. We study assortment and pricing problems under this choice model. In our choice model, each customer has a different patience level sampled from a known distribution, which determines the maximum number of stages in which she is willing to view the assortment of products. In each stage, if the utility of a product in the current stage is larger than the utility of the outside option, then the customer purchases this product and leaves the system. Otherwise, the customer views the assortment of products in the next stage, as long as her patience level allows her. Thus, in our model, customers impatiently leave for two reasons. First, a customer purchases a product in the current stage as soon as its utility exceeds a minimum acceptable utility, even though there may be a product with a larger utility in a later stage. Second, a customer runs out of patience and leaves without viewing the entire assortment.

## 1.1 Main Contributions

Our main contributions are the formulation of the multinomial logit model with impatient customers, as well as developing exact and approximate solution methods for assortment optimization and pricing problems under this choice model.

*Multinomial Logit Model with Impatient Customers*. We propose a new variant of the multinomial logit model with impatient customers. The choice model is based on random utilities. A customer arriving into the system associates random utilities with the products. Furthermore, she has a minimum acceptable utility and a patience level, which are also both random. We refer to the minimum acceptable utility as the utility of the outside option. The utilities of the products and the outside option are independent and have the Gumbel distribution with the same scale parameter. The patience level of the customer has a general distribution over the support

$\{1,\ldots,m\}$ for a fixed integer $m$ and is independent of the utilities. The customer incrementally views the assortment of offered products in multiple stages. In each stage, if the product with the largest utility provides larger utility than the outside option, then the customer purchases this product and the choice process terminates. If the utilities of all products that the customer views before she runs out of patience are smaller than the utility of the outside option, then the customer leaves without a purchase. Since the patience level of a customer is at most $m$, we choose pairwise disjoint assortments that we offer over $m$ stages. We give a closed-form expression for the choice probability of each product under any assortment (Theorem 2.1).

*Assortment Optimization*. In the assortment optimization problem, each product has a fixed revenue and the goal is to find a revenue-maximizing sequence of assortments to offer. For this problem, we give a polynomial-time algorithm using the following steps. First, we show that there exists a revenue-ordered optimal solution. That is, letting $n$ be the number of products and $r_i$ be the revenue of product $i$, indexing the products so that $r_1 \geq r_2 \geq \ldots \geq r_n$, the optimal assortment to offer in stage $k$ is of the form $\{j^*_k+1,\ldots,j^*_{k+1}\}$ for some $0=j^*_1 \leq j^*_2 \leq \ldots \leq j^*_m \leq j^*_{m+1}$ (Theorem 3.1). Second, exploiting the revenue-ordered property, we give a dynamic program that finds the best sequence of revenue-ordered assortments using $O(mn^2)$ operations (Theorem 3.3).

*Joint Pricing and Assortment Optimization*. In the pricing setting, the mean utility of a product depends on its price. Following the standard assumption in the pricing literature, we assume that the products have the same price sensitivity; see Song and Xue (2007), Li and Huh (2011). We start with the case where the sequence of offered assortments is fixed and the goal is to find the revenue-maximizing prices. The expected revenue is not concave in the prices, but we give a reformulation where the decision variables are the probabilities that a customer will reach different stages. We show that the expected revenue is concave in these decision variables and we can recover the optimal prices after solving our reformulation (Theorem 4.2).

Next, we consider the case where both the sequence of offered assortments and the prices are decision variables. We give an approximation algorithm that obtains at least 87.8% of the optimal expected revenue (Theorem 4.4). This approximation algorithm is based on showing that if we offer all products in the first stage and compute the corresponding optimal prices, then the solution that we obtain has an 87.8% performance guarantee. In our computational experiments, starting from such a solution that is obtained by offering all products in the first stage, we use a neighborhood search algorithm to further improve the quality of this solution.

*Space Constraints*. We consider the assortment optimization problem where each product occupies a certain amount of space and there is a constraint on the total space consumption of the offered products. We give a fully polynomial-time approximation scheme (FPTAS) (Theorem 5.2). In the special case where there is a constraint on the total number of offered products, we can improve the running time of our FPTAS. We also give an exact algorithm whose running time depends exponentially on the number of stages $m$, but polynomially on the number of products $n$. A constraint on the total space consumption or the total number of offered products may arise, for example, when we want to avoid overwhelming a patient with too many appointment slot options or when we have a limited budget and offering a product requires a capital investment.

*Numerical Results*. Using a dataset from Expedia, we check the performance of our choice model to predict the customer purchases, when compared against the standard multinomial logit benchmark. We use two metrics. The first metric is out-of-sample log-likelihood. The second metric is the fraction of customers whose bookings are predicted correctly. In the first and second metrics, our choice model improves upon the benchmark by, respectively, 1.95% and 4.49%, on average. Also, we test the practical performance of our approximation schemes for joint pricing and assortment optimization, as well as for assortment optimization under space constraints.

In many online retail settings, the products are offered on multiple webpages, but the number of products on a webpage is at the discretion of the retailer, since the products are simply presented as a list, as on Amazon, for example. Our unconstrained and constrained assortment optimization problems, as well as our joint pricing and assortment optimization problem, find applications in such settings. Our choice model is motivated by the satisficing behavior of customers, especially when purchasing leisure products, such as hotel rooms, where the customer directly proceeds to purchasing a product once the utility of the product exceeds a minimum acceptable utility.

### 1.2 Literature Review

There is recent assortment optimization work where customers view only a portion of the offered assortment due to either search behavior or consideration sets. In Gallego et al. (2016), the customers decide on the number of webpages to view based on an exogenous distribution and choose within the entire assortment on these webpages according to a general choice model. Wang and Sahin (2018) consider a model where the customers focus on a portion of the products by trading off the expected utility from the purchase with the search effort, but they do not view the assortment incrementally. Derakhshan et al. (2018) examine a product ranking problem where the customers build a consideration set as a function of the search cost. In Aouad and Segev (2018), each customer views a random number of webpages and makes a choice within these webpages according to the multinomial logit model. The customers do not view the products sequentially. Aouad et al. (2019) focus on a setting where each product is included in the consideration set of a customer with a fixed probability. In Feldman et al. (2019), the choice model is based on short preference lists, corresponding to the case with small consideration sets. In all these papers, the assortment optimization problems are NP-hard and the authors give approximation methods.

Liu et al. (2019) and Feldman and Segev (2019) study assortment optimization problems under another variant of the multinomial logit model with multiple stages. The two papers use the same choice model, where the utility of the outside option is re-sampled when the customer considers the products at each stage during the course of her choice process and the utilities of the outside option at different stages are independent. Thus, a customer may associate a high utility with the outside option in one stage, but a low utility in another stage. Under this choice model, it is NP-hard to find the revenue-maximizing sequence of assortments to offer. Both papers give approximation schemes, the main difference being the running time of the approximation scheme depends exponentially on the number of stages $m$ in Liu et al. (2019), but polynomially in Feldman and Segev (2019). In our choice model, the utility of the outside option is sampled once at the beginning of the choice process of the customer. We give a polynomial-time exact algorithm.

Gallego et al. (2004) and Talluri and van Ryzin (2004) study the assortment optimization problem under the standard multinomial logit model and show that it is optimal to offer a revenue-ordered assortment. Rusmevichientong et al. (2010), Wang (2012), Jagabathula (2016) and Sumida et al. (2019) impose various constraints on the offered assortment. Bront et al. (2009), Mendez-Diaz et al. (2014) and Rusmevichientong et al. (2014) consider the problem under a mixture of multinomial logit models. Flores et al. (2019) use a two-stage multinomial logit model where the products that can be offered in each of the two stages are fixed a priori. For the pricing problem, Song and Xue (2007), Hopp and Xu (2005) and Li and Huh (2011) show that the expected revenue is concave in the product market shares and the optimal prices of the products exceed their marginal costs by the same mark-up, as long as the products have the same price sensitivity. Zhang et al. (2018) show that these two results hold under all generalized extreme value models.

We limit our literature review to the multinomial logit model, but we refer to Farias et al. (2013), Davis et al. (2014), Gallego and Wang (2014), Aouad et al. (2016), Blanchet et al. (2016), Desir et al. (2016a) and Li and Webster (2017) for work under other choice models.

### 1.3 Organization

In Section 2, we define our choice model and derive an expression for the choice probabilities. In Section 3, we consider the unconstrained assortment optimization problem. In Section 4, we examine the joint pricing and assortment problem. In Section 5, we study constraints on the space consumption of the offered products. In Section 6, we give computational experiments.

## 2. Multinomial Logit Model with Impatient Customers

We describe our choice model and give an expression for the choice probabilities of the products. The set of products is $\mathcal{N}=\{1,\ldots,n\}$. The set of stages is $\mathcal{M}=\{1,\ldots,m\}$. We use $(S_1,\ldots,S_m)$ to denote the sequence of assortments that we offer over all $m$ stages, where $S_k \subseteq \mathcal{N}$ is the assortment that we offer in stage $k$. The assortments that we offer in different stages are disjoint, so $S_k \cap S_\ell = \varnothing$ for all $k \neq \ell$. The utility of product $i$ is given by the random variable $U_i$, which has the Gumbel distribution with location-scale parameters $(\mu_i, 1)$. Letting $v_i = e^{\mu_i}$, we refer to $v_i$ as the preference weight of product $i$. The utility of the outside option is given by the random variable $U_0$, which has the Gumbel distribution with location-scale parameters $(0,1)$. The patience level of a customer is given by the random variable $Y$ taking values in $\mathcal{M}$. A customer with patience level $k$ is willing to view the assortments in the first $k$ stages. We let $\lambda_k = \mathbb{P}\{Y \geq k\}$, so $1 = \lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_m > 0$. The random variables $\{U_i : i \in \mathcal{N}\}$, $U_0$ and $Y$ are independent. In our choice model, an arriving customer is characterized by the utilities she associates with the different products and the outside option, along with her patience level, all sampled from their distributions.

As stated in the introduction, the utility of the outside option corresponds to the minimum acceptable utility for the customer. A customer chooses among the products by sequentially viewing the assortments in each stage. Given that the customer is currently in stage $k$, if the product with the largest utility in stage $k$ provides larger utility than the outside option, then the customer purchases this product. Otherwise, the customer moves on to stage $k+1$. If stage $k+1$ is beyond the patience level of the customer, then the customer leaves without a purchase. Otherwise, the customer views the products in stage $k+1$. The customer leaves the system for two reasons. First, if the product with the largest utility in stage $k$ provides larger utility than the outside option, then the customer purchases this product right away, even though there may be a product in a subsequent stage with larger utility. Second, due to her patience level, a customer may not view all assortments in all stages. As a function of the assortments $(S_1,\ldots,S_m)$, let $\phi_i^k(S_1,\ldots,S_m)$ be the probability that a customer will choose product $i \in S_k$. In the next theorem, we give an expression for this choice probability. Throughout the paper, we let $V(S) = \sum_{i \in S} v_i$.

**Theorem 2.1 (Choice Probabilities)** *If we offer assortments $(S_1,\ldots,S_m)$ over $m$ stages, then a customer purchases product $i \in S_k$ with probability*

$$\phi_i^k(S_1,\ldots,S_m) = \frac{\lambda_k\, v_i}{(1+\sum_{\ell=1}^{k-1} V(S_\ell))\,(1+\sum_{\ell=1}^{k} V(S_\ell))}.$$

*Proof:* Letting $X_1$ and $X_2$ be independent Gumbel random variables with location-scale parameters $(\mu_1, 1)$ and $(\mu_2, 1)$, we use three properties of Gumbel random variables. First, $\max\{X_1, X_2\}$

is a Gumbel random variable with location-scale parameters $(\log(e^{\mu_1}+e^{\mu_2}),1)$. Second, we have $\mathbb{P}\{X_1 \geq X_2\} = \frac{e^{\mu_1}}{e^{\mu_1}+e^{\mu_2}}$. Third, letting $\mathbf{1}(\cdot)$ be the indicator function, the random variables $\max\{X_1, X_2\}$ and $\mathbf{1}(X_1 \geq X_2)$ are independent. The first and second properties are discussed in Sections 7.2.2.2-3 and Appendix B in Talluri and van Ryzin (2005). We show the third property in Appendix A. For a customer to purchase product $i \in S_k$, her patience level must be at least $k$, the utility of the outside option must exceed the utilities of all products in stages $1,\ldots,k-1$, and the utility of product $i$ must exceed both the utility of the outside option and the utilities of all other products in stage $k$. Letting $\widehat{U}_{k-1} = \max_{j\in S_1\cup\ldots\cup S_{k-1}} U_j$ and $\widetilde{U}_k = \max_{j\in S_k\setminus\{i\}} U_j$, we have

$$\begin{aligned}\phi_i^k(S_1,\ldots,S_k) &= \mathbb{P}\Big\{Y \geq k\Big\}\cdot\mathbb{P}\Big\{U_0 \geq \max_{j\in S_1\cup\ldots\cup S_{k-1}} U_j,\ U_i \geq \max\Big\{U_0, \max_{j\in S_k\setminus\{i\}} U_j\Big\}\Big\} \\ &= \lambda_k\cdot\mathbb{P}\Big\{U_0 \geq \widehat{U}_{k-1}\Big\}\cdot\mathbb{P}\Big\{U_i \geq \max\Big\{U_0,\widetilde{U}_k\Big\}\ \Big|\ U_0 \geq \widehat{U}_{k-1}\Big\}. \qquad (1)\end{aligned}$$

By the first property, $\widehat{U}_{k-1}$ and $\widetilde{U}_k$ are Gumbel random variables with location-scale parameters $(\log\sum_{j\in S_1\cup\ldots\cup S_{k-1}} e^{\mu_j},1)$ and $(\log\sum_{j\in S_k\setminus\{i\}} e^{\mu_j},1)$. Moreover, the random variables $U_i$, $U_0$, $\widehat{U}_{k-1}$, and $\widetilde{U}_k$ are independent. Considering the second probability on the right side of (1), we have

$$\begin{aligned}\mathbb{P}\Big\{U_i \geq \max\{U_0,\widetilde{U}_k\}\ \big|\ U_0 \geq \widehat{U}_{k-1}\Big\} &= \mathbb{P}\Big\{U_i \geq \max\{U_0,\widetilde{U}_k,\widehat{U}_{k-1}\}\ \big|\ U_0 \geq \widehat{U}_{k-1}\Big\} \\ &= \mathbb{P}\Big\{U_i \geq \max\{U_0,\widehat{U}_{k-1}\},\ U_i \geq \widetilde{U}_k\ \big|\ U_0 \geq \widehat{U}_{k-1}\Big\} \\ &= \mathbb{P}\Big\{U_i \geq \max\{U_0,\widehat{U}_{k-1}\}\ \big|\ U_0 \geq \widehat{U}_{k-1}\Big\}\cdot\mathbb{P}\Big\{U_i \geq \widetilde{U}_k\ \big|\ U_i \geq \max\{U_0,\widehat{U}_{k-1}\},\ U_0 \geq \widehat{U}_{k-1}\Big\} \\ &\overset{(a)}{=} \mathbb{P}\Big\{U_i \geq \max\{U_0,\widehat{U}_{k-1}\}\Big\}\cdot\mathbb{P}\Big\{U_i \geq \widetilde{U}_k\ \big|\ U_i \geq \max\{U_0,\widehat{U}_{k-1}\},\ U_0 \geq \widehat{U}_{k-1}\Big\} \\ &\overset{(b)}{=} \mathbb{P}\Big\{U_i \geq \max\{U_0,\widehat{U}_{k-1}\}\Big\}\cdot\mathbb{P}\Big\{U_i \geq \widetilde{U}_k\ \big|\ U_i \geq \max\{U_0,\widehat{U}_{k-1}\}\Big\} \\ &= \mathbb{P}\Big\{U_i \geq \max\{U_0,\widehat{U}_{k-1}\},\ U_i \geq \widetilde{U}_k\Big\} \\ &= \mathbb{P}\Big\{U_i \geq \max\{U_0,\widetilde{U}_k,\widehat{U}_{k-1}\}\Big\} \overset{(c)}{=} \frac{e^{\mu_i}}{1+\sum_{j\in S_1\cup\ldots\cup S_k} e^{\mu_j}} = \frac{v_i}{1+\sum_{\ell=1}^k V(S_\ell)}. \qquad (2)\end{aligned}$$

Both $(a)$ and $(b)$ use the fact that $\max\{U_0,\widehat{U}_{k-1}\}$ and $\mathbf{1}(U_0 \geq \widehat{U}_{k-1})$ are independent by the third property, so knowing that $U_0 \geq \widehat{U}_{k-1}$ does not change the distribution of $\max\{U_0,\widehat{U}_{k-1}\}$, along with $U_i$ and $\widetilde{U}_k$ are independent of $U_0$ and $\widehat{U}_{k-1}$. Lastly, $(c)$ uses the first and second properties.

Considering the first probability on the right side of (1), using the second property and the fact that $v_i = e^{\mu_i}$, we get $\mathbb{P}\{U_0 \geq \widehat{U}_{k-1}\} = 1/(1+\sum_{j\in S_1\cup\ldots\cup S_{k-1}} e^{\mu_j}) = 1/(1+\sum_{\ell=1}^{k-1} V(S_\ell))$. Plugging this equality and (2) into (1) gives the desired result. ■

As an extension to our model, we can consider the case where a customer, after not making a purchase in stage $k$, decides to continue to the next stage with probability $\beta_k$. If the decision to continue to the next stage is independent of $\{U_i : i \in \mathcal{N}\}$, $U_0$ and $Y$, then all we need to do is to multiply the choice probability in the theorem with $\beta_1\,\beta_2\,\ldots\beta_{k-1}$. Thus, this extension is equivalent to using a patience level distribution with $\mathbb{P}\{Y \geq k\} = \beta_1\,\beta_2\,\ldots\beta_{k-1}\,\lambda_k$.

## 3. Unconstrained Assortment Optimization

We focus on the assortment optimization problem with no constraints on the offered assortment and give a polynomial-time algorithm. We use $r_i > 0$ to denote the revenue of product $i$. Throughout the paper, we let $W(S) = \sum_{i \in S} r_i v_i$. Noting the choice probability in Theorem 2.1, if we offer assortments $(S_1, \ldots, S_m)$ over $m$ stages, then the expected revenue from a customer is

$$\Pi(S_1, \ldots, S_m) \;=\; \sum_{k \in \mathcal{M}} \sum_{i \in S_k} r_i \, \phi_i^k(S_1, \ldots, S_m) \;=\; \sum_{k \in \mathcal{M}} \sum_{i \in S_k} \frac{\lambda_k \, r_i \, v_i}{(1 + \sum_{\ell=1}^{k-1} V(S_\ell))\,(1 + \sum_{\ell=1}^{k} V(S_\ell))}$$

$$= \sum_{k \in \mathcal{M}} \frac{\lambda_k \, W(S_k)}{(1 + \sum_{\ell=1}^{k-1} V(S_\ell))\,(1 + \sum_{\ell=1}^{k} V(S_\ell))}. \tag{3}$$

The assortments offered over $m$ stages are disjoint, so the set of feasible solutions is $\mathcal{F} = \{(S_1, \ldots, S_m) : S_k \subseteq \mathcal{N} \;\forall\, k \in \mathcal{M},\; S_k \cap S_\ell = \varnothing \;\forall\, k \neq \ell\}$. We want to solve the problem

$$\max_{(S_1, \ldots, S_m) \in \mathcal{F}} \;\; \Pi(S_1, \ldots, S_m). \qquad (\textsc{Assortment})$$

We use two steps to give a polynomial-time algorithm for the Assortment problem. First, we show that there exists an optimal solution to the Assortment problem that is revenue-ordered. Specifically, we index the products in the order of decreasing revenues so that $r_1 \geq r_2 \geq \ldots \geq r_n$. Then, there exists an optimal solution $(S_1^*, \ldots, S_m^*)$ such that $S_k^* = \{j_k^* + 1, \ldots, j_{k+1}^*\}$ for $j_1^*, \ldots, j_{m+1}^*$ that satisfy $0 = j_1^* \leq j_2^* \leq \ldots \leq j_{m+1}^*$. Thus, the assortment offered in each stage follows the order of the revenues of the products. Noting $j_1^* = 0$, the choice of the products $j_2^*, \ldots, j_{m+1}^*$ determines an optimal solution to the Assortment problem. Knowing that there exists an optimal solution that is revenue-ordered reduces the number of possible optimal solutions to $O(n^m)$, which is polynomial in $n$ but still exponential in $m$. Second, exploiting the revenue-ordered property, we find an optimal sequence of revenue-ordered assortments by solving a dynamic program in $O(mn^2)$ operations.

**Optimality of Revenue-Ordered Assortments:**

For two solutions $(S_1, \ldots, S_m)$ and $(T_1, \ldots, T_m)$, we say that the solution $(S_1, \ldots, S_m)$ dominates the solution $(T_1, \ldots, T_m)$ if $|S_1| = |T_1| \ldots |S_k| = |T_k|$ and $|S_{k+1}| > |T_{k+1}|$ for some $k \in \mathcal{M}$. Intuitively speaking, a dominating solution offers an assortment with a larger cardinality in an earlier stage. If there are multiple optimal solutions for the Assortment problem, then we choose an optimal solution that is non-dominated by any other optimal solution. To establish that an optimal solution to the Assortment problem satisfies the revenue ordered property, we construct revenue thresholds for each stage such that if the revenue of a product falls within the thresholds for stage $k$, then it is optimal to offer the product in stage $k$. The next theorem is the main result of this section, establishing the existence of such revenue thresholds. The proof follows from an intermediate lemma, which we give after the theorem.

**Theorem 3.1 (Optimal Revenue-Ordered Assortments)** *There exists an optimal solution* $(S_1^*,\ldots,S_m^*)$ *to the* ASSORTMENT *problem such that* $S_k^* = \{i\in\mathcal{N} : t_{k+1}^* \le r_i < t_k^*\}$ *for some revenue thresholds* $t_1^*,\ldots,t_{m+1}^*$ *that satisfy* $+\infty = t_1^* \ge t_2^* \ge \ldots \ge t_{m+1}^*$.

To construct the revenue thresholds, let $R_k(S_1,\ldots,S_m) = \frac{\lambda_k W(S_k)}{(1+\sum_{\ell=1}^{k-1} V(S_\ell))\,(1+\sum_{\ell=1}^{k} V(S_\ell))}$ denote the expected revenue obtained in stage $k$, and define

$$\begin{aligned} t_k(S_1,\ldots,S_m) &= \frac{R_{k-1}(S_1,\ldots,S_m)+R_k(S_1,\ldots,S_m)}{\frac{\lambda_{k-1}}{1+\sum_{\ell=1}^{k-2}V(S_\ell)} - \frac{\lambda_k}{1+\sum_{\ell=1}^{k}V(S_\ell)}} \qquad \forall\, k\in\mathcal{M}\setminus\{1\},\\ t_{m+1}(S_1,\ldots,S_m) &= \frac{R_m(S_1,\ldots,S_m)}{\frac{\lambda_m}{1+\sum_{\ell=1}^{m-1}V(S_\ell)}}. \end{aligned}$$

We set $t_1(S_1,\ldots,S_m)=+\infty$. In the next lemma, we quantify the change in the expected revenue when we move a product from one stage to another. The proof is in Appendix B.

**Lemma 3.2 (Product Exchanges)** *For each sequence of assortments* $(S_1,\ldots,S_m)\in\mathcal{F}$ *offered over* $m$ *stages, we have the following three identities.*

(*a*) $\Pi(S_1,\ldots,S_{k-1}\cup\{i\},S_k\setminus\{i\},\ldots,S_m) - \Pi(S_1,\ldots,S_m)$

$$= \frac{\frac{\lambda_{k-1}}{1+\sum_{\ell=1}^{k-2}V(S_\ell)} - \frac{\lambda_k}{1+\sum_{\ell=1}^{k}V(S_\ell)}}{1+\sum_{\ell=1}^{k-1}V(S_\ell)+v_i}\, v_i\,(r_i - t_k(S_1,\ldots,S_m)) \quad \forall\, k=\mathcal{M}\setminus\{1\},\ i\in S_k,$$

(*b*) $\Pi(S_1,\ldots,S_k\setminus\{i\},S_{k+1}\cup\{i\}\ldots,S_m) - \Pi(S_1,\ldots,S_m)$

$$= \frac{\frac{\lambda_{k}}{1+\sum_{\ell=1}^{k-1}V(S_\ell)} - \frac{\lambda_{k+1}}{1+\sum_{\ell=1}^{k+1}V(S_\ell)}}{1+\sum_{\ell=1}^{k}V(S_\ell)-v_i}\, v_i\,(t_{k+1}(S_1,\ldots,S_m) - r_i) \quad \forall\, k\in\mathcal{M}\setminus\{m\},\ i\in S_k,$$

(*c*) $\Pi(S_1,\ldots,S_{m-1},S_m\setminus\{i\}) - \Pi(S_1,\ldots,S_m)$

$$= \frac{\frac{\lambda_m}{1+\sum_{\ell=1}^{m-1}V(S_\ell)}}{1+\sum_{\ell=1}^{m}V(S_\ell)-v_i}\, v_i\,(t_{m+1}(S_1,\ldots,S_m) - r_i) \quad \forall\, i\in S_m.$$

The proof of the lemma is based on directly evaluating the changes in the expected revenue using (3). Since $\lambda_k \ge \lambda_{k+1}$ and $\sum_{\ell=1}^{k}V(S_\ell) \le \sum_{\ell=1}^{k+1}V(S_\ell)$, by this lemma, we can compare $r_i$ only with $t_k(S_1,\ldots,S_m)$ or $t_{k+1}(S_1,\ldots,S_m)$ to check whether moving product $i$ from stage $k$ to stage $k-1$ or to stage $k+1$ improves the expected revenue. Below is the proof of Theorem 3.1.

**Proof of Theorem 3.1:** Let $(S_1^*,\ldots,S_m^*)$ be a non-dominated optimal solution to the ASSORTMENT problem. Without loss of generality, $S_1^*\ne\varnothing,\ldots,S_\ell^*\ne\varnothing$, $S_{\ell+1}^*=\varnothing,\ldots,S_m^*=\varnothing$ for some $\ell\in\mathcal{M}$. In particular, if $S_\ell^*=\varnothing$ and $S_{\ell+1}^*\ne\varnothing$, then the solution $(S_1^*,\ldots,S_{\ell-1}^*,S_{\ell+1}^*,S_\ell^*,\ldots,S_m^*)$ dominates the solution $(S_1^*,\ldots,S_m^*)$, but since $\lambda_1\ge\lambda_2\ge\ldots\ge\lambda_m$, noting (3), we can check that the expected revenue of the solution $(S_1^*,\ldots,S_{\ell-1}^*,S_{\ell+1}^*,S_\ell^*,\ldots,S_m^*)$ is at least as large as that of the solution $(S_1^*,\ldots,S_m^*)$. Thus, since $S_{\ell+1}^*=\varnothing,\ldots,S_m^*=\varnothing$, a customer does not make a purchase after stage $\ell$, so if we consider the Assortment problem with only $\ell$ stages, then $(S_1^*, \ldots, S_\ell^*)$ must be a non-dominated optimal solution. For all $k = 1, \ldots, \ell$, we let $t_k^* = t_k(S_1^*, \ldots, S_\ell^*)$ and focus on the Assortment problem with $\ell$ stages, where the set of stages is $\mathcal{L} = \{1, \ldots, \ell\}$.

First, we claim that $S_k^* \subseteq \{i \in \mathcal{N} : t_{k+1}^* \leq r_i < t_k^*\}$ for all $k \in \mathcal{L}$. Specifically, for $k \in \mathcal{L} \setminus \{1\}$, if $i \in S_k^*$ and $r_i \geq t_k^*$, then Lemma 3.2(a) implies that moving product $i$ from assortment $S_k^*$ to $S_{k-1}^*$ does not degrade the expected revenue from the solution $(S_1^*, \ldots, S_\ell^*)$, which contradicts the fact that $(S_1^*, \ldots, S_\ell^*)$ is a non-dominated optimal solution. For $k = 1$, we cannot have $r_i \geq t_1^*$, since $t_1^* = +\infty$. For $k \in \mathcal{L} \setminus \{\ell\}$, if $i \in S_k^*$ and $r_i < t_{k+1}^*$, then Lemma 3.2(b) implies that moving product $i$ from assortment $S_k^*$ to $S_{k+1}^*$ strictly increases the expected revenue from the solution $(S_1^*, \ldots, S_\ell^*)$, which contradicts the fact that $(S_1^*, \ldots, S_\ell^*)$ is an optimal solution. For $k = \ell$, if $i \in S_\ell^*$ and $r_i < t_{\ell+1}^*$, then Lemma 3.2(c) implies that removing product $i$ from assortment $S_\ell^*$ strictly improves the expected revenue from the solution $(S_1^*, \ldots, S_\ell^*)$. So, the claim holds. Also, for all $k \in \mathcal{L}$, $S_k^* \neq \varnothing$ includes some product $i$, so $t_{k+1}^* \leq r_i < t_k^*$ for all $k \in \mathcal{L}$. Thus, $t_{k+1}^* \leq t_k^*$ for all $k \in \mathcal{L}$.

Second, we claim that $S_k^* \supseteq \{i \in \mathcal{N} : t_{k+1}^* \leq r_i < t_k^*\}$ for all $k \in \mathcal{L}$. Specifically, if $t_{k+1}^* \leq r_i < t_k^*$ and $i \notin S_k^*$ for some $k \in \mathcal{L}$, then it must be the case that $i \notin S_q^*$ for all $q \in \mathcal{L}$, since by the first claim above, we have $S_q^* \subseteq \{i \in \mathcal{N} : t_{q+1}^* \leq r_i < t_q^*\}$ for all $q \in \mathcal{L}$ and $+\infty = t_1^* \geq t_2^* \geq \cdots \geq t_{\ell+1}^*$. In this case, using the fact that $r_i \geq t_{k+1}^* \geq t_{\ell+1}^*$, replacing the preference weight of product $i$ in Lemma 3.2(c) with $-v_i$, note that adding product $i$ to assortment $S_\ell^*$ does not degrade the expected revenue from the solution $(S_1^*, \ldots, S_\ell^*)$, which contradicts the fact that $(S_1^*, \ldots, S_\ell^*)$ is a non-dominated optimal solution to the Assortment problem with $\ell$ stages. So, the claim holds. By the two claims, we get $S_k^* = \{i \in \mathcal{N} : t_{k+1}^* \leq r_i < t_k^*\}$ for all $k \in \mathcal{L}$ and $+\infty = t_1^* \geq t_2^* \geq \ldots \geq t_{\ell+1}^*$. For the problem with $m$ stages, noting that $S_k^* = \varnothing$ for all $k = \ell + 1, \ldots, m$, we set $t_k^* = t_{\ell+1}^*$ for all $k = \ell + 2, \ldots, m + 1$, so we have $S_k^* = \{i \in \mathcal{N} : t_{k+1}^* \leq r_i < t_k^*\}$ for all $k \in \mathcal{M}$ and $+\infty = t_1^* \geq t_2^* \geq \ldots \geq t_{m+1}^*$. ■

**Finding an Optimal Sequence of Revenue-Ordered Assortments:**

We show how to find an optimal sequence of revenue-ordered assortments. By Theorem 3.1, we can consider solutions $(S_1, \ldots, S_m)$ of the form $S_k = \{j_k + 1, \ldots, j_{k+1}\}$ for $j_1, \ldots, j_{m+1}$ that satisfy $0 = j_1 \leq j_2 \leq \ldots \leq j_{m+1}$. If we offer assortments of this form, then $S_1 \cup \ldots \cup S_{k-1} = \{1, \ldots, j_k\}$ and $\sum_{\ell=1}^{k-1} V(S_\ell) = V(\{1, \ldots, j_k\})$. Thus, we can solve a dynamic program to pick assortments of this form to offer in each stage so that we maximize the expected revenue. The decision epochs are the stages. The state variable at decision epoch $k$ is the value of $j$ such that the assortments $S_1, \ldots, S_{k-1}$ offered in the previous stages satisfy $S_1 \cup \ldots \cup S_{k-1} = \{1, \ldots, j\}$. The action at decision epoch $k$ is the value of $p$ such that the assortment offered in stage $k$ is $\{j+1, \ldots, p\}$. Let $J_k(j)$ denote the maximum expected revenue obtained from stages $k, k+1, \ldots, m$, given that $S_1 \cup \ldots \cup S_{k-1} = \{1, \ldots, j\}$. The next theorem gives a dynamic programming formulation to compute $\{J_k(j) : j \in \mathcal{N},\ k \in \mathcal{M}\}$.

**Theorem 3.3 (Dynamic Program for an Optimal Sequence of Assortments)** *Letting $J_{m+1}(\cdot) = 0$, for all $k \in \mathcal{M}$ and $j \in \mathcal{N}$, we have*

$$J_k(j) = \max_{p \in \{j,\dots,n\}} \left\{ \frac{\lambda_k W(\{j+1,\dots,p\})}{(1+V(\{1,\dots,j\}))\,(1+V(\{1,\dots,p\}))} \; + \; J_{k+1}(p) \right\},$$

*and we can solve the dynamic program in $O(mn^2)$ operations.*

*Proof:* The dynamic program follows from the discussion above. Precomputing $W(\{j+1,\dots,p\})$ and $V(\{1,\dots,j\})$ for all $j,p \in \mathcal{N}$ with $p > j$ in $O(n^2)$ operations, since there are $m$ decision epochs, $n$ states and $n$ actions, we can solve the dynamic program in $O(mn^2)$ operations. ■

## 4. Joint Pricing and Assortment Optimization

We consider the joint pricing and assortment optimization problem, where we choose the assortment of products to offer in each stage as well as the prices of the products.

### 4.1 Optimal Prices under Fixed Assortments

In this section, we assume that the assortments $(S_1,\dots,S_m)$ offered over $m$ stages are fixed. We give a convex program to choose the prices to maximize the expected revenue. In Section 4.2, we build on this result give an approximation algorithm for joint pricing and assortment optimization. We use $p_i$ to denote the price for product $i$. For fixed parameters $\alpha_i$ and $\beta$, if we charge the price $p_i$ for product $i$, then the utility of product $i$ has the Gumbel distribution with location-scale parameters $(\alpha_i - \beta p_i, 1)$, with the corresponding mean $\alpha_i - \beta p_i + \gamma$, where $\gamma$ is the Euler-Mascheroni constant. Thus, the mean utility of a product depends linearly on its price. Such linear dependence of the mean utility on the price is often used in the literature; see Song and Xue (2007), Gallego and Wang (2014), Li and Webster (2017). The parameter $\alpha_i$ captures the intrinsic mean utility of product $i$, whereas $\beta$ captures the sensitivity of the mean utility to price. If we charge the price $p_i$ for product $i$, then the preference weight of the product is $e^{\alpha_i - \beta p_i}$. As a function of the prices $\boldsymbol{p} = (p_1,\dots,p_n)$, let $V_k(\boldsymbol{p}) = \sum_{i \in S_k} e^{\alpha_i - \beta p_i}$ capture the total preference weight of the products in stage $k$. Since the assortment of products offered in each stage is fixed, we do not make the dependence of $V_k(\boldsymbol{p})$ on the assortment $S_k$ explicit. By Theorem 2.1, if the prices of the products are given by $\boldsymbol{p}$, then a customer chooses product $i \in S_k$ with probability $\phi_i^k(\boldsymbol{p}) = \lambda_k e^{\alpha_i - \beta p_i} / ((1 + \sum_{\ell=1}^{k-1} V_\ell(\boldsymbol{p}))\,(1 + \sum_{\ell=1}^{k} V_\ell(\boldsymbol{p})))$. As a function of the prices $\boldsymbol{p}$, the expected revenue obtained from a customer is

$$\Pi(\boldsymbol{p}) \; = \; \sum_{k \in \mathcal{M}} \sum_{i \in S_k} p_i \, \phi_i^k(\boldsymbol{p}) \; = \; \sum_{k \in \mathcal{M}} \frac{\lambda_k \sum_{i \in S_k} p_i \, e^{\alpha_i - \beta p_i}}{(1 + \sum_{\ell=1}^{k-1} V_\ell(\boldsymbol{p}))\,(1 + \sum_{\ell=1}^{k} V_\ell(\boldsymbol{p}))}.$$

In the pricing literature, it is customary to include a marginal cost $c_i$ for product $i$ so that the objective function presented above reads $\sum_{k \in \mathcal{M}} \sum_{i \in S_k} (p_i - c_i)\, \phi_i^k(\boldsymbol{p})$. Including a marginal cost for product $i$ is equivalent to simply shifting the price of product $i$ by $c_i$ and the constant $\alpha_i$ by $\beta c_i$. We want to find the product prices to maximize the expected revenue, yielding the problem

$$\max_{\boldsymbol{p}\in\mathbb{R}^n} \ \Pi(\boldsymbol{p}). \tag{Pricing}$$

The prices are not constrained to be nonnegative above, allowing us to use first-order conditions to characterize an optimal solution. In Appendix C, we show that the optimal prices are nonnegative.

**Stage-Specific Optimal Prices:**

The following theorem shows that the prices in a particular stage are the same in an optimal solution. We use this result to give a convex program to solve the Pricing problem.

**Theorem 4.1 (Stage-Specific Optimal Prices)** *There exists an optimal solution $\boldsymbol{p}^*$ to the* Pricing *problem such that if $i,j\in S_\ell$ for some $\ell\in\mathcal{M}$, then $p_i^*=p_j^*$.*

*Proof:* Letting $\boldsymbol{p}^*$ be an optimal solution to the Pricing problem, assume that $p_i^*\neq p_j^*$ for some $\ell\in\mathcal{M}$ and $i,j\in S_\ell$. We construct another solution $\widehat{\boldsymbol{p}}$ with $\widehat{p}_i=\widehat{p}_j$ and $\widehat{p}_t=p_t^*$ for all $t\in\mathcal{N}\setminus\{i,j\}$ such that the solution $\widehat{\boldsymbol{p}}$ provides an expected revenue that is at least as large as the one provided by the solution $\boldsymbol{p}^*$. Specifically, set $\widehat{p}_t=p_t^*$ for all $t\in\mathcal{N}\setminus\{i,j\}$. Thus, letting $W_k(\boldsymbol{p})=\sum_{t\in S_k} p_t\, e^{\alpha_t-\beta\, p_t}$, we have $V_k(\widehat{\boldsymbol{p}})=V_k(\boldsymbol{p}^*)$ and $W_k(\widehat{\boldsymbol{p}})=W_k(\boldsymbol{p}^*)$ for all $k\in\mathcal{M}\setminus\{\ell\}$. Letting $K^*=e^{\alpha_i-\beta\, p_i^*}+e^{\alpha_j-\beta\, p_j^*}$, set $(\widehat{p}_i,\widehat{p}_j)$ as an optimal solution to the problem

$$\max_{(p_i,p_j)\in\mathbb{R}^2}\Big\{p_i\, e^{\alpha_i-\beta\, p_i}+p_j\, e^{\alpha_j-\beta\, p_j}\ :\ e^{\alpha_i-\beta\, p_i}+e^{\alpha_j-\beta\, p_j}=K^*\Big\}. \tag{4}$$

Since $(p_i^*,p_j^*)$ is a feasible solution to problem (4) and $\widehat{p}_t=p_t^*$ for all $t\in S_\ell\setminus\{i,j\}$, we have $W_\ell(\widehat{\boldsymbol{p}})\geq W_\ell(\boldsymbol{p}^*)$. Also, noting the constraint, we have $e^{\alpha_i-\beta\,\widehat{p}_i}+e^{\alpha_j-\beta\,\widehat{p}_j}=K^*=e^{\alpha_i-\beta\, p_i^*}+e^{\alpha_j-\beta\, p_j^*}$, so $V_\ell(\boldsymbol{p}^*)=V_\ell(\widehat{\boldsymbol{p}})$. In this case, since $V_k(\widehat{\boldsymbol{p}})=V_k(\boldsymbol{p}^*)$ and $W_k(\widehat{\boldsymbol{p}})=W_k(\boldsymbol{p}^*)$ for all $k\in\mathcal{M}\setminus\{\ell\}$, noting the definition of $\Pi(\boldsymbol{p})$, we get $\Pi(\widehat{\boldsymbol{p}})\geq\Pi(\boldsymbol{p}^*)$, as desired. It remains to show that the optimal solution $(\widehat{p}_i,\widehat{p}_j)$ to problem (4) satisfies $\widehat{p}_i=\widehat{p}_j$. Using the change of variables $d_i=e^{\alpha_i-\beta\, p_i}$ and solving for $p_i$, we have $p_i=\frac{1}{\beta}(\alpha_i-\log d_i)$. Thus, problem (4) is equivalent to the problem $\frac{1}{\beta}\max_{(d_i,d_j)\in\mathbb{R}^2_+}\big\{(\alpha_i-\log d_i)\, d_i+(\alpha_j-\log d_j)\, d_j : d_i+d_j=K^*\big\}$. Since $-x\log x$ is concave in $x$, we can solve the last problem using Lagrangian relaxation. Associating the Lagrange multiplier $\theta$ with the constraint, the Lagrangian is $(\alpha_i-\log d_i)\, d_i+(\alpha_j-\log d_j)\, d_j+\theta(K^*-d_i-d_j)$. Differentiating the Lagrangian, the optimal solution $(\widehat{d}_i,\widehat{d}_j)$ to the last problem satisfies $\alpha_i-\log\widehat{d}_i-1-\theta=0$ and $\alpha_j-\log\widehat{d}_j-1-\theta=0$, so $\alpha_i-\log\widehat{d}_i=1+\theta=\alpha_j-\log\widehat{d}_j$. Recalling that $p_i=\frac{1}{\beta}(\alpha_i-\log d_i)$, the optimal solution $(\widehat{p}_i,\widehat{p}_j)$ to (4) satisfies $\widehat{p}_i=\frac{1}{\beta}(\alpha_i-\log\widehat{d}_i)=\frac{1}{\beta}(1+\theta)=\frac{1}{\beta}(\alpha_j-\log\widehat{d}_j)=\widehat{p}_j$. ■

By the theorem above, we can focus on the solutions where all products in each stage have the same price. Next, we give a convex reformulation of the Pricing problem.

**Convex Reformulation of the Pricing Problem:**

By Theorem 4.1, letting $\rho_k$ be the price that we charge for all products in stage $k$, we can use stage-specific prices $\boldsymbol{\rho} = (\rho_1, \ldots, \rho_m)$, instead of product-specific prices $\boldsymbol{p} = (p_1, \ldots, p_n)$, as the decision variables. It is simple to give examples to demonstrate that the expected revenue is not a concave function of either of $\boldsymbol{p}$ or $\boldsymbol{\rho}$. We give an equivalent formulation for the Pricing problem, which has a concave objective function and linear constraints. Using stage-specific prices $\boldsymbol{\rho} = (\rho_1, \ldots, \rho_m)$, the total preference weight of the products in stage $k$ is $\widehat{V}_k(\boldsymbol{\rho}) = \sum_{i \in S_k} e^{\alpha_i - \beta \rho_k} = e^{-\beta\rho_k} \sum_{i\in S_k} e^{\alpha_i}$. Noting the definition of the purchase probability for product $i$ in Theorem 2.1, if we charge the stage-specific prices $\boldsymbol{\rho}$, then the probability that a customer purchases some product in stage $k$ is $\frac{\lambda_k \sum_{i\in S_k} e^{\alpha_i - \beta\,\rho_k}}{(1+\sum_{\ell=1}^{k-1} \widehat{V}_\ell(\boldsymbol{p}))\,(1+\sum_{\ell=1}^{k} \widehat{V}_\ell(\boldsymbol{p}))} = \frac{\lambda_k\, \widehat{V}_k(\boldsymbol{\rho})}{(1+\sum_{\ell=1}^{k-1} \widehat{V}_\ell(\boldsymbol{p}))\,(1+\sum_{\ell=1}^{k} \widehat{V}_\ell(\boldsymbol{p}))}$, in which case, the price of the purchased product is $\rho_k$. Throughout the rest of this section, we let $q_k(\boldsymbol{\rho}) = 1/(1+\sum_{\ell=1}^{k} \widehat{V}_\ell(\boldsymbol{\rho}))$ with the convention that $q_0(\boldsymbol{\rho}) = 1$. For each $k \in \mathcal{M}$, $q_k(\boldsymbol{\rho})$ is the probability that the utility of the outside option exceeds the utility of all products offered in the first $k$ stages.

We refer to $q_k(\boldsymbol{\rho})$ as the no-purchase probability over the first $k$ stages, but we understand that this probability is actually the no-purchase probability for a customer with patience level exceeding $k$. The idea behind our convex reformulation is to express the probability that a customer makes a purchase in each stage and the stage-specific price for each stage as functions of the no-purchase probabilities over different numbers of stages. By doing so, we will express the expected revenue as a function of the no-purchase probabilities over different numbers of stages as well. Specifically, noting that $q_k(\boldsymbol{\rho}) = \frac{1}{1+\sum_{\ell=1}^{k} \widehat{V}_\ell(\boldsymbol{\rho})}$, we have $q_{k-1}(\boldsymbol{\rho}) - q_k(\boldsymbol{\rho}) = \frac{\widehat{V}_k(\boldsymbol{\rho})}{(1+\sum_{\ell=1}^{k-1} \widehat{V}_\ell(\boldsymbol{\rho}))\,(1+\sum_{\ell=1}^{k} \widehat{V}_\ell(\boldsymbol{\rho}))}$. Therefore, given the no-purchase probabilities $\boldsymbol{q} = (q_1, \ldots, q_m)$ over different numbers of stages, the probability that a customer purchases a product in stage $k$ is $\lambda_k\,(q_{k-1} - q_k)$. Moreover, using the fact that $q_k(\boldsymbol{\rho}) = \frac{1}{1+\sum_{\ell=1}^{k} \widehat{V}_\ell(\boldsymbol{\rho})}$, we get $\frac{1}{q_k(\boldsymbol{\rho})} - \frac{1}{q_{k-1}(\boldsymbol{\rho})} = \widehat{V}_k(\boldsymbol{\rho}) = e^{-\beta\rho_k} \sum_{i\in S_k} e^{\alpha_i}$. In this case, solving for $\rho_k$ in the equality $\frac{1}{q_k(\boldsymbol{\rho})} - \frac{1}{q_{k-1}(\boldsymbol{\rho})} = e^{-\beta\rho_k} \sum_{i\in S_k} e^{\alpha_i}$, given the no-purchase probabilities $\boldsymbol{q} = (q_1, \ldots, q_m)$ over different numbers of stages, the stage-specific price for stage $k$ is

$$\rho_k(\boldsymbol{q}) = \frac{1}{\beta}\left\{ \log\left( \sum_{i\in S_k} e^{\alpha_i} \right) - \log\left( \frac{1}{q_k} - \frac{1}{q_{k-1}} \right) \right\}.$$

Thus, for given no-purchase probabilities $\boldsymbol{q} = (q_1, \ldots, q_m)$, the customer makes a purchase in stage $k$ with probability $\lambda_k\,(q_{k-1} - q_k)$. If she does so, then the price of the purchased product is $\rho_k(\boldsymbol{q})$.

By the discussion in the above paragraph, we can express the expected revenue as a function of the no-purchase probabilities. That is, we have

$$\widehat{\Pi}(\boldsymbol{q}) \;=\; \sum_{k\in\mathcal{M}} \lambda_k\,(q_{k-1} - q_k)\,\rho_k(\boldsymbol{q}) \;=\; \sum_{k\in\mathcal{M}} \frac{\lambda_k}{\beta}\,(q_{k-1} - q_k) \left\{ \log\left( \sum_{i\in S_k} e^{\alpha_i} \right) - \log\left( \frac{1}{q_k} - \frac{1}{q_{k-1}} \right) \right\}. \quad (5)$$

In the next theorem, we show that the expected revenue function above is concave in the no-purchase probabilities and we can recover the optimal prices using its maximizer.

**Theorem 4.2 (Convex Reformulation for Pricing)** *The expected revenue* $\widehat{\Pi}(\boldsymbol{q})$ *in* (5) *is a concave function of* $\boldsymbol{q}$. *Furthermore, letting* $\boldsymbol{q}^*$ *be an optimal solution to the problem*

$$\max_{\boldsymbol{q}\in\mathbb{R}^m} \left\{ \widehat{\Pi}(\boldsymbol{q}) \;:\; q_{k-1} \geq q_k \quad \forall\, k \in \mathcal{M} \right\} \tag{6}$$

*with the convention that* $q_0 = 1$, *if we set* $\rho_k^* = \rho_k(\boldsymbol{q}^*)$ *for all* $k \in \mathcal{M}$, *then* $\boldsymbol{\rho}^*$ *are optimal stage-specific prices to charge in the* Pricing *problem.*

*Proof:* To show that $\widehat{\Pi}(\boldsymbol{q})$ is a concave function of $\boldsymbol{q}$, noting that $\sum_{k\in\mathcal{M}} \lambda_k\,(q_{k-1}-q_k)\,\log\sum_{i\in S_k} e^{\alpha_i}$ is linear in $\boldsymbol{q}$, it suffices to prove that $\sum_{k\in\mathcal{M}} \lambda_k\,(q_{k-1}-q_k)\,\log(\frac{1}{q_k}-\frac{1}{q_{k-1}})$ is convex in $\boldsymbol{q}$. We have $\sum_{k\in\mathcal{M}} \lambda_k\,(q_{k-1}-q_k)\,\log(\frac{1}{q_k}-\frac{1}{q_{k-1}}) = \sum_{k\in\mathcal{M}} \lambda_k\,(q_{k-1}-q_k)\,\log\frac{q_{k-1}-q_k}{q_{k-1}} - \sum_{k\in\mathcal{M}} \lambda_k\,(q_{k-1}-q_k)\,\log q_k$. First, we show that $\sum_{k\in\mathcal{M}} \lambda_k\,(q_{k-1}-q_k)\,\log\frac{q_{k-1}-q_k}{q_{k-1}}$ is convex in $\boldsymbol{q}$. The relative entropy function $x\log(x/y)$ is convex in $(x,y)\in\mathbb{R}_+^2$; see Example 3.19 in Boyd and Vandenberghe (2004). Moreover, composing a convex function with an affine function preserves its convexity; see Section 3.2.2 in Boyd and Vandenberghe (2004). Thus, $(q_{k-1}-q_k)\,\log\frac{q_{k-1}-q_k}{q_{k-1}}$ is convex in $\boldsymbol{q}$, in which case, $\sum_{k\in\mathcal{M}} \lambda_k\,(q_{k-1}-q_k)\,\log\frac{q_{k-1}-q_k}{q_{k-1}}$ is convex in $\boldsymbol{q}$. Second, we show that $-\sum_{k\in\mathcal{M}} \lambda_k\,(q_{k-1}-q_k)\,\log q_k$ is convex in $\boldsymbol{q}$. Noting that $q_0 = 1$ and rearranging the terms in the sum, we have

$$\begin{aligned}
-\sum_{k\in\mathcal{M}} \lambda_k\,(q_{k-1}-q_k)\,\log q_k &= -\lambda_1\,\log q_1 + \sum_{k=1}^{m-1} q_k\,(\lambda_k\,\log q_k - \lambda_{k+1}\,\log q_{k+1}) + \lambda_m\,q_m\,\log q_m \\
&= -\lambda_1\,\log q_1 + \sum_{k=1}^{m-1} \lambda_{k+1}\,q_k(\log q_k - \log q_{k+1}) + \sum_{k=1}^{m-1}(\lambda_k-\lambda_{k+1})\,q_k\,\log q_k + \lambda_m\,q_m\,\log q_m.
\end{aligned}$$

Because $x\log(x/y)$ and $x\log x$ are convex in $(x,y)\in\mathbb{R}_+^2$ and $\lambda_k \geq \lambda_{k+1}$, $-\sum_{k\in\mathcal{M}} \lambda_k\,(q_{k-1}-q_k)\log q_k$ is convex in $\boldsymbol{q}$. The second part of the theorem holds by the discussion before the theorem. ■

Problem (6) has a concave objective function and linear constraints, so we can solve it efficiently using convex optimization tools. Also, by (5), once we fix the values of $(q_1,\ldots,q_{k-1})$, the optimal values of $(q_k,\ldots,q_m)$ only depend on $q_{k-1}$. In Appendix D, using this observation, we give a dynamic program to find a solution with an additive performance guarantee of $\theta$ with a running time polynomial in $1/\theta$. Numerically, solving problem (6) through convex optimization tools turns out to be faster, but the dynamic program does not require convex optimization software.

**Monotonicity of Optimal Prices:**

In the next theorem, we compare the optimal prices in different stages. If $\lambda_k = 1$ for all $k \in \mathcal{M}$, then the patience level of the customers is $m$ with probability one, which is to say that the customers leave the system only when they have found a product with utility exceeding the utility of the outside option, or they have exhausted all stages and still have not found a product with utility exceeding the utility of the outside option. By the next theorem, if $\lambda_k = 1$ for all $k \in \mathcal{M}$, then the optimal prices in stage $k$ are at least as large as those in stage $k+1$ for each $k \in \mathcal{M}\setminus\{m\}$.

**Theorem 4.3 (Monotonicity of Scaled Prices)** *There exist optimal stage-specific prices* $\boldsymbol{\rho}^* = (\rho_1^*, \ldots, \rho_m^*)$ *in the* Pricing *problem such that* $\lambda_k \, \rho_k^* \geq \lambda_{k+1} \, \rho_{k+1}^*$ *for all* $k = 1, \ldots, m-1$.

*Proof:* Letting $q_k^* = q_k(\boldsymbol{\rho}^*)$ for all $k \in \mathcal{M}$ with $q_0^* = 1$, in Appendix E, we use the first-order condition for the Pricing problem to show that the optimal stage-specific prices satisfy

$$\frac{1}{\beta} - \frac{q_\ell^*}{q_{\ell-1}^*} \, \rho_\ell^* + \frac{1}{\lambda_\ell \, q_\ell^* \, q_{\ell-1}^*} \sum_{k=\ell+1}^{m} \rho_k^* \, \lambda_k \Big\{ (q_{k-1}^*)^2 - (q_k^*)^2 \Big\} = 0. \tag{7}$$

Letting $Q_{\ell+1}^* = \sum_{k=\ell+1}^m \rho_k^* \, \lambda_k \, ((q_{k-1}^*)^2 - (q_k^*)^2)$, (7) reads as $\frac{q_\ell^*}{q_{\ell-1}^*} \, \rho_\ell^* = \frac{1}{\beta} + \frac{1}{\lambda_\ell \, q_\ell^* \, q_{\ell-1}^*} \, Q_{\ell+1}^*$. Similarly, using (7) for stage $\ell+1$, we have $\frac{q_{\ell+1}^*}{q_\ell^*} \, \rho_{\ell+1}^* = \frac{1}{\beta} + \frac{1}{\lambda_{\ell+1} \, q_{\ell+1}^* \, q_\ell^*} \, Q_{\ell+2}^*$, which is equivalent to

$$\frac{1}{\lambda_\ell \, q_\ell^* \, q_{\ell-1}^*} \, Q_{\ell+2}^* = \frac{\lambda_{\ell+1} \, (q_{\ell+1}^*)^2}{\lambda_\ell \, q_\ell^* \, q_{\ell-1}^*} \, \rho_{\ell+1}^* - \frac{\lambda_{\ell+1} \, q_{\ell+1}^*}{\beta \, \lambda_\ell \, q_{\ell-1}^*}. \tag{8}$$

Noting the definition of $Q_{\ell+1}^*$, we have $Q_{\ell+1}^* = \rho_{\ell+1}^* \, \lambda_{\ell+1} \, ((q_\ell^*)^2 - (q_{\ell+1}^*)^2) + Q_{\ell+2}^*$. Therefore, using the fact that $\frac{q_\ell^*}{q_{\ell-1}^*} \, \rho_\ell^* = \frac{1}{\beta} + \frac{1}{\lambda_\ell \, q_\ell^* \, q_{\ell-1}^*} \, Q_{\ell+1}^*$, we obtain the chain of equalities

$$\begin{aligned}
\frac{q_\ell^*}{q_{\ell-1}^*} \, \rho_\ell^* \; &= \; \frac{1}{\beta} + \frac{1}{\lambda_\ell \, q_\ell^* \, q_{\ell-1}^*} \Big\{ \rho_{\ell+1}^* \, \lambda_{\ell+1} \, ((q_\ell^*)^2 - (q_{\ell+1}^*)^2) + Q_{\ell+2}^* \Big\} \\
&\overset{(a)}{=} \; \frac{1}{\beta} + \frac{1}{\lambda_\ell \, q_\ell^* \, q_{\ell-1}^*} \, \rho_{\ell+1}^* \, \lambda_{\ell+1} \, ((q_\ell^*)^2 - (q_{\ell+1}^*)^2) + \frac{\lambda_{\ell+1} \, (q_{\ell+1}^*)^2}{\lambda_\ell \, q_\ell^* \, q_{\ell-1}^*} \, \rho_{\ell+1}^* - \frac{\lambda_{\ell+1} \, q_{\ell+1}^*}{\beta \, \lambda_\ell \, q_{\ell-1}^*} \\
&= \; \left( 1 - \frac{\lambda_{\ell+1} \, q_{\ell+1}^*}{\lambda_\ell \, q_{\ell-1}^*} \right) \frac{1}{\beta} + \frac{\lambda_{\ell+1} \, q_\ell^*}{\lambda_\ell \, q_{\ell-1}^*} \, \rho_{\ell+1}^*,
\end{aligned}$$

where $(a)$ uses (8). By the chain of equalities above, $\frac{q_\ell^*}{\lambda_\ell \, q_{\ell-1}^*} (\lambda_\ell \, \rho_\ell^* - \lambda_{\ell+1} \, \rho_{\ell+1}^*) = (1 - \frac{\lambda_{\ell+1} \, q_{\ell+1}^*}{\lambda_\ell \, q_{\ell-1}^*}) \frac{1}{\beta}$, so since $q_{\ell+1}^* \leq q_{\ell-1}^*$ and $\lambda_{\ell+1} \leq \lambda_\ell$, we get $\lambda_\ell \, \rho_\ell^* \geq \lambda_{\ell+1} \, \rho_{\ell+1}^*$. ■

Our results in this section use the assumption that the products have the same price sensitivity, which is reasonable for the products in the same product category. When studying pricing problems even under the standard multinomial logit model with a single stage, it is common to use the assumption that the products have the same price sensitivity. For example, Hopp and Xu (2005), Song and Xue (2007), Li and Huh (2011) and Zhang and Lu (2013) use this assumption when working with the standard multinomial logit model. These papers use market shares of the products as the decision variables to give convex reformulations, whereas we use the no-purchase probabilities. When different products have different price sensitivities, we are not aware of convex reformulations of the pricing problem even under the standard multinomial logit model, but it is not known whether such convex reformulations provably do not exist.

### 4.2 Optimal Assortments and Prices

In this section, we consider the case where both the assortments $(S_1,\ldots,S_m)$ offered over $m$ stages and the prices charged for the products are decision variables. By the discussion in the previous section, for any fixed sequence of assortments, it is optimal to charge stage-specific prices. Therefore, it suffices to focus on stage-specific prices when both the sequence of assortments to offer and the prices to charge are decision variables, but to simplify the proofs of our results, we revert to using product-specific prices. As a function of the product-specific prices $\boldsymbol{p}=(p_1,\ldots,p_n)$, we let $V(\boldsymbol{p},S)=\sum_{i\in S} e^{\alpha_i-\beta p_i}$ to capture the total preference weight of the products in $S$. Noting Theorem 2.1, if we charge the prices $\boldsymbol{p}$ and offer the assortments $(S_1,\ldots,S_m)$, then a customer purchases product $i\in S_k$ with probability $\lambda_k\, e^{\alpha_i-\beta p_i}/((1+\sum_{\ell=1}^{k-1}V(\boldsymbol{p},S_\ell))\,(1+\sum_{\ell=1}^{k}V(\boldsymbol{p},S_\ell)))$. Thus, as a function of the product-specific prices $\boldsymbol{p}$ and the assortments $(S_1,\ldots,S_m)$ over $m$ stages, the expected revenue is

$$\Pi(\boldsymbol{p},S_1,\ldots,S_m) \;=\; \sum_{k\in\mathcal{M}} \frac{\lambda_k \sum_{i\in S_k} p_i\, e^{\alpha_i-\beta p_i}}{(1+\sum_{\ell=1}^{k-1}V(\boldsymbol{p},S_\ell))\,(1+\sum_{\ell=1}^{k}V(\boldsymbol{p},S_\ell))}.$$

We continue to use $\mathcal{F}$ to denote the set of feasible assortments that we can offer over $m$ stages, ensuring that the assortments offered over different stages are disjoint.

Our goal is to find the assortment to offer in each stage and the prices to charge for the products to maximize the expected revenue, yielding the problem

$$\max_{(\boldsymbol{p},S_1,\ldots,S_m)\in\mathbb{R}^n\times\mathcal{F}} \quad \Pi(\boldsymbol{p},S_1,\ldots,S_m). \qquad (\textsc{Pricing-Assortment})$$

This problem involves both continuous and discrete decision variables. In the rest of this section, we focus on obtaining solutions with performance guarantees for this problem.

**Approximation Algorithm for Joint Pricing and Assortment Optimization:** In the next theorem, we show that if we offer all products simply in the first stage and compute the corresponding optimal prices, then we obtain a 0.878-approximate solution. By Theorem 4.2, we can solve a convex program to compute the optimal prices for a fixed sequence of assortments.

**Theorem 4.4 (87.8% Approximation for Joint Pricing and Assortment Optimization)** *Letting $\pi^*$ be the optimal objective value of the* Pricing-Assortment *problem, we have* $\max_{\boldsymbol{p}\in\mathbb{R}^n}\Pi(\boldsymbol{p},\mathcal{N},\varnothing,\ldots,\varnothing)\geq 0.878\,\pi^*$.

In Appendix F, we give a proof for Theorem 4.4 and show that the performance guarantee of 87.8% is tight. The proof is based on a sequence of upper bounds. First, we consider a variant of the Pricing-Assortment problem, where the patience levels of the customers are, intuitively speaking, infinite. We argue that the optimal expected revenue of the variant with infinite patience levels provides an upper bound on that of the PRICING-ASSORTMENT problem. Second, by treating $(\sum_{i\in S_1} e^{\alpha_i}, \ldots, \sum_{i\in S_m} e^{\alpha_i})$ in (5) as continuous quantities, we formulate a smooth variant of the PRICING-ASSORTMENT problem. We argue that the optimal expected revenue of the smooth variant is an upper bound on that of the variant with infinite patience levels. Third, we give a closed-form upper bound on the optimal expected revenue of the smooth variant. Chaining all upper bounds, the closed-form upper bound is also an upper bound on the optimal expected revenue of the PRICING-ASSORTMENT problem. Lastly, the closed-form upper bound is simple enough to allow us to show that the expected revenue obtained by offering all products in the first stage and computing the corresponding optimal prices is at least 87.8% of the closed-form upper bound.

Theorem 4.4 also allows us to make an interesting contrast between the ASSORTMENT and PRICING-ASSORTMENT problems. For the ASSORTMENT problem, we can show that if we offer the empty assortment in all stages except for the first stage and find the revenue-maximizing assortment to offer in the first stage, then we obtain a solution that provides at least 50% of the optimal expected revenue in the ASSORTMENT problem. In Appendix G, we give a proof for the performance guarantee of 50% and show that it is tight. Thus, for the PRICING-ASSORTMENT problem, finding the revenue-maximizing prices to charge in the first stage while offering the empty assortment in all other stages yields a tight performance guarantee of 87.8%, whereas for the ASSORTMENT problem, finding the revenue-maximizing assortment to offer in the first stage while offering the empty assortment in all other stages yields a tight performance guarantee of 50%.

In our computational experiments, we use a neighborhood search algorithm to further improve the performance of the solution obtained by offering all products in the first stage and computing the corresponding optimal prices. In particular, we start with a sequence of assortments that offers all products in the first stage. Given the current sequence of assortments, we check all neighbors of the current sequence of assortments for an appropriately defined neighborhood. For each sequence of assortments in the neighborhood, we compute the corresponding optimal prices to charge. Among all sequences of assortments in the neighborhood and their corresponding optimal prices, we pick the best one. We repeat the process starting from the best sequence of assortments in the neighborhood, until we cannot improve the expected revenue. This algorithm is guaranteed to provide a solution that is at least as good as the solution that we start with.

In this section, we gave an approximation algorithm for the PRICING-ASSORTMENT problem. In this problem, the prices take values over a continuum and the preference weight of product $i$ is given by $e^{\alpha_i - \beta p_i}$ as a function of its price $p_i$. A natural question is the computational complexity of this problem. In Appendix H, we show that if the prices take values over a discrete set, then the PRICING-ASSORTMENT problem is NP-hard, but the computational complexity of the problem with the prices taking values over a continuum remains an open question.

## 5. Assortment Optimization under a Space Constraint

We consider the assortment problem when each product occupies a certain amount of space and there is a limit on the total space consumption of the products offered in all stages. As in Section 3, the revenue of product $i$ is $r_i$ and we index the products such that $r_1 \geq r_2 \geq \ldots \geq r_n$. The space consumption of product $i$ is $c_i$. We let $C(S) = \sum_{i \in S} c_i$. The total amount of space available is $b$. Noting the expected revenue function in the ASSORTMENT problem, we want to solve

$$\max_{(S_1,\ldots,S_m)\in\mathcal{F}} \left\{ \sum_{k\in\mathcal{M}} \frac{\lambda_k\, W(S_k)}{(1+\sum_{q=1}^{k-1} V(S_q))\,(1+\sum_{q=1}^{k} V(S_q))} \;:\; \sum_{k\in\mathcal{M}} C(S_k) \leq b \right\}. \quad \text{(CAPACITATED)}$$

**Overview of Our Approach:**

The CAPACITATED problem is NP-hard; see Rusmevichientong et al. (2009). So, we focus on developing an FPTAS. In the next lemma, shown in Appendix I, we give a structural property of an optimal solution to the CAPACITATED problem that will be useful to develop an FPTAS.

**Lemma 5.1 (Solutions for the Capacitated Problem)** *In a non-dominated optimal solution* $(S_1^*,\ldots,S_m^*)$ *to the* CAPACITATED *problem, for all* $k \in \mathcal{M}$*, we have* $S_k^* \subseteq \{j_k^*+1,\ldots,j_{k+1}^*\}$ *for* $j_1^*,\ldots,j_{m+1}^*$ *with* $0 = j_1^* \leq j_2^* \leq \ldots \leq j_m^* \leq j_{m+1}^* = n$.

This lemma does not immediately yield an efficient algorithm, since the optimal assortment $S_k^*$ in stage $k$ may omit products in $\{j_k^*+1,\ldots,j_{k+1}^*\}$. In the ASSORTMENT problem, by Theorem 3.1, the optimal assortment $S_k^*$ in stage $k$ satisfies $S_k^* = \{j_k^*+1,\ldots,j_{k+1}^*\}$ for $j_k^*, j_{k+1}^*$ with $0 \leq j_k^* \leq j_{k+1}^* \leq n$.

To give an FPTAS for the CAPACITATED problem, we fix a value of $\epsilon \in (0,1)$ and proceed as described in the following two parts.

**Part 1. Constructing Candidates:** For each $j,\ell \in \{0,\ldots,n\}$ with $j \leq \ell$, we will construct a collection of candidate assortments $\textsc{Cand}(j,\ell)$ that satisfies the following two properties.

- **(Correct Product Interval)** For each $\widehat{S} \in \textsc{Cand}(j,\ell)$, we have $\widehat{S} \subseteq \{j+1,\ldots,\ell\}$. Thus, a candidate assortment in $\textsc{Cand}(j,\ell)$ can include only the products in $\{j+1,\ldots,\ell\}$.
- **(Limited Degradation)** For each $S \subseteq \{j+1,\ldots,\ell\}$, there exists $\widehat{S} \in \textsc{Cand}(j,\ell)$ such that $W(\widehat{S}) \geq (1-\epsilon/4)\,W(S)$, $V(\widehat{S}) \leq (1+\epsilon/4)\,V(S)$, and $C(\widehat{S}) \leq C(S)$.

Intuitively speaking, noting the objective function of the CAPACITATED problem, we prefer $S \subseteq \mathcal{N}$ with larger $W(S)$, smaller $V(S)$, and smaller $C(S)$. By the second property, for any assortment $S \subseteq \{j+1,\ldots,\ell\}$, there exists a candidate assortment $\widehat{S} \in \textsc{Cand}(j,\ell)$ that is almost as preferable. Throughout this section, let $v_{\min} = \min\{v_i : i \in \mathcal{N}\}$, $v_{\max} = \max\{v_i : i \in \mathcal{N}\}$, $w_{\min} = \min\{v_i\, r_i : i \in \mathcal{N}\}$, and $w_{\max} = \max\{v_i\, r_i : i \in \mathcal{N}\}$. We will construct all collections of candidate assortments $\{\textsc{Cand}(j,\ell) : j,\ell \in \{0,\ldots,n\},\ j \le \ell\}$ in $O(\frac{n^4}{\epsilon^2} \log(\frac{n w_{\max}}{w_{\min}}) \log(\frac{n v_{\max}}{v_{\min}}))$ operations. Each collection $\textsc{Cand}(j,\ell)$ will include $O(\frac{n^2}{\epsilon^2} \log(\frac{n w_{\max}}{w_{\min}}) \log(\frac{n v_{\max}}{v_{\min}}))$ candidate assortments.

**Part 2. Combining Candidates:** Having obtained the collection of candidate assortments from Part 1, we solve an approximate version of the Capacitated problem given by

$$\max_{(S_1,\ldots,S_m,j_1,\ldots,j_m)} \left\{ \sum_{k\in\mathcal{M}} \frac{\lambda_k\, W(S_k)}{(1+\sum_{q=1}^{k-1} V(S_q))\,(1+\sum_{q=1}^{k} V(S_q))} \ : \right.$$
$$\left. S_k \in \textsc{Cand}(j_k, j_{k+1}) \ \ \forall\, k \in \mathcal{M},\ j_k \le j_{k+1} \ \ \forall\, k \in \mathcal{M},\ \sum_{k\in\mathcal{M}} C(S_k) \le b \right\}, \quad (9)$$

where we follow the convention that $j_{m+1} = n$. Comparing this problem with the Capacitated problem, we have $S_k \in \textsc{Cand}(j_k, j_{k+1})$ above. Also, we do not explicitly impose the constraint that $S_k \cap S_q = \varnothing$ for $k \ne q$ above, but the constraint $S_k \in \textsc{Cand}(j_k, j_{k+1})$ for all $k \in \mathcal{M}$, along with $j_k \le j_{k+1}$ for all $k \in \mathcal{M}$, ensures that $S_k \cap S_q = \varnothing$ for $k \ne q$. Problem (9) is an approximate version of the Capacitated problem, where we can offer only candidate assortments.

Letting $a \vee b = \max\{a,b\}$, we will obtain a $(1-\frac{\epsilon}{4})$-approximate solution to problem (9) in $O(\frac{n^4 m^3}{\epsilon^4} \log(\frac{n w_{\max}}{w_{\min}}) \log(\frac{n w_{\max}\,(1 \vee n v_{\max})}{\lambda_m\, w_{\min}}) \log^2(\frac{n v_{\max}}{v_{\min}}))$ operations.

In this case, executing the two parts, we get an FPTAS given by the following result.

**Theorem 5.2 (FPTAS under a Space Constraint)** *For each $\epsilon \in (0,1)$, we can obtain a $(1-\epsilon)$-approximate solution to the* Capacitated *problem in the number of operations*

$$O\left( \frac{n^4 m^3}{\epsilon^4} \log\left(\frac{n w_{\max}}{w_{\min}}\right) \log\left(\frac{n w_{\max}\,(1 \vee n v_{\max})}{\lambda_m\, w_{\min}}\right) \log^2\left(\frac{n v_{\max}}{v_{\min}}\right) \right).$$

*Proof:* We execute the two parts discussed above. By the discussion just before the theorem, the number of operations to execute the two parts is given by the expression in the theorem. Let $(S_1^*,\ldots,S_m^*)$ be an optimal solution to the Capacitated problem. By Lemma 5.1, we know that there exist $0 = j_1^* \le j_2^* \le \ldots \le j_m^* \le j_{m+1}^* = n$ such that $S_k^* \subseteq \{j_k^*+1,\ldots,j_{k+1}^*\}$ for all $k \in \mathcal{M}$. After executing Part 1, by the second property in Part 1, for each $k \in \mathcal{M}$, there exists $\widehat{S}_k \in \textsc{Cand}(j_k^*, j_{k+1}^*)$ such that $W(\widehat{S}_k) \ge (1-\epsilon/4)\, W(S_k^*)$, $V(\widehat{S}_k) \le (1+\epsilon/4)\, V(S_k^*)$ and $C(\widehat{S}_k) \le C(S_k^*)$. Since $(S_1^*,\ldots,S_m^*)$ is an optimal solution to the Capacitated problem, we have $\sum_{k\in\mathcal{M}} C(S_k^*) \le b$, so noting that $C(\widehat{S}_k) \le C(S_k^*)$ for all $k \in \mathcal{M}$, the solution $(\widehat{S}_1,\ldots,\widehat{S}_m, j_1^*,\ldots,j_m^*)$ is feasible for problem (9). After executing Part 2, we have a $(1-\frac{\epsilon}{4})$-approximate solution to problem (9) as well, which we denote by $(\widetilde{S}_1,\ldots,\widetilde{S}_m,\widetilde{j}_1,\ldots,\widetilde{j}_m)$. Therefore, we get

$$\begin{aligned}\sum_{k\in\mathcal{M}}\frac{\lambda_k\,W(\widetilde{S}_k)}{(1+\sum_{q=1}^{k-1}V(\widetilde{S}_q))\,(1+\sum_{q=1}^{k}V(\widetilde{S}_q))} &\overset{(a)}{\geq} \left(1-\frac{\epsilon}{4}\right)\sum_{k\in\mathcal{M}}\frac{\lambda_k\,W(\widehat{S}_k)}{(1+\sum_{q=1}^{k-1}V(\widehat{S}_q))\,(1+\sum_{q=1}^{k}V(\widehat{S}_q))}\\ &\overset{(b)}{\geq} \left(1-\frac{\epsilon}{4}\right)\sum_{k\in\mathcal{M}}\frac{(1-\frac{\epsilon}{4})\,\lambda_k\,W(S_k^*)}{(1+\sum_{q=1}^{k-1}(1+\frac{\epsilon}{4})V(S_q^*))\,(1+\sum_{q=1}^{k}(1+\frac{\epsilon}{4})V(S_q^*))}\\ &\geq \frac{(1-\frac{\epsilon}{4})^2}{(1+\frac{\epsilon}{4})^2}\sum_{k\in\mathcal{M}}\frac{\lambda_k\,W(S_k^*)}{(1+\sum_{q=1}^{k-1}V(S_q^*))\,(1+\sum_{q=1}^{k}V(S_q^*))}\,.\end{aligned}$$

In this chain of inequalities, $(a)$ holds because $(\widetilde{S}_1,\ldots,\widetilde{S}_m,\widetilde{j}_1,\ldots,\widetilde{j}_m)$ is a $(1-\frac{\epsilon}{4})$-approximate solution to problem (9), whereas $(\widehat{S}_1,\ldots,\widehat{S}_m,j_1^*,\ldots,j_m^*)$ is only a feasible solution to problem (9), whereas $(b)$ holds because $W(\widehat{S}_k)\geq(1-\epsilon/4)\,W(S_k^*)$ and $V(\widehat{S}_k)\leq(1+\epsilon/4)\,V(S_k^*)$. For $\epsilon\in(0,1)$, we have $\frac{(1-\epsilon/4)^2}{(1+\epsilon/4)^2}\geq(1-\frac{\epsilon}{4})^4\geq 1-\epsilon$. In this case, letting $z^*$ be the optimal objective value of the Capacitated problem and noting that $(S_1^*,\ldots,S_m^*)$ is the optimal solution to the Capacitated problem, the chain of inequalities presented above implies that the solution $(\widetilde{S}_1,\ldots,\widetilde{S}_m)$ provides an objective value of at least $(1-\epsilon)\,z^*$ to the Capacitated problem. ■

The number of operations in Theorem 5.2 is polynomial in the input size and $1/\epsilon$, giving an FPTAS. In the next two sections, we discuss how to execute the two parts.

### 5.1 Part 1: Constructing Collections of Candidate Assortments

We focus on executing Part 1. For each $j,\ell\in\{0,\ldots,n\}$ with $j\leq\ell$, we separately construct the collection of candidate assortments $\textsc{Cand}(j,\ell)$. Therefore, we fix $j,\ell$ throughout this section. Intuitively speaking, to construct the collection of candidate assortments $\textsc{Cand}(j,\ell)$, we use a geometric grid to guess the values of $W(S)$ and $V(S)$ for each possible assortment $S\subseteq\{j+1,\ldots,\ell\}$. For each guess for the values of $W(S)$ and $V(S)$, we use a dynamic program to find an assortment $\widehat{S}$ such that $W(\widehat{S})$ and $V(\widehat{S})$ are not too far from the guess and the capacity consumption of $\widehat{S}$ is as small as possible. The dynamic program that we use is, in spirit, similar to the one that is used for solving the knapsack problem; see, for example, Chapter 3 in Williamson and Shmoys (2011) and Desir et al. (2016b). In particular, for fixed $\rho>0$, we define the geometric grid $\textsc{Dom}=\{(1+\rho)^r : r\in\mathbb{Z}\}\cup\{0\}$. We define the round down operator $\lfloor\cdot\rfloor$ that rounds its argument down to the closest point in Dom when the argument is positive. That is, if $a\geq 0$, then we have $\lfloor a\rfloor=\max\{b\in\textsc{Dom}: b\leq a\}$. If $a<0$, then we follow the convention that $\lfloor a\rfloor=0$. Similarly, we define the round up operator $\lceil\cdot\rceil$ that rounds its argument up to the closest point in Dom when the argument is positive. That is, if $a\geq 0$, then we have $\lceil a\rceil=\min\{b\in\textsc{Dom}: b\geq a\}$. If $a<0$, then we follow the convention that $\lceil a\rceil=-\infty$. For given $(x,y)\in\textsc{Dom}^2$ and $(j,\ell)$, we consider finding the smallest possible capacity consumption of any assortment $S \subseteq \{j+1,\dots,\ell\}$ that satisfies $W(S) \geq x$ and $V(S) \leq y$. For this purpose, we use the dynamic program

$$\Theta_i^\ell(x,y) \;=\; \min_{u_i \in \{0,1\}} \Big\{ c_i\, u_i + \Theta_{i+1}^\ell(\lfloor x - v_i\, r_i\, u_i \rfloor \,,\, \lceil y - v_i\, u_i \rceil) \Big\}, \tag{10}$$

where we use the boundary condition that $\Theta_{\ell+1}^\ell(x,y) = 0$ if $x \leq 0$ and $y \geq 0$. If, on the other hand, $x > 0$ or $y < 0$, then we have $\Theta_{\ell+1}^\ell(x,y) = +\infty$.

In (10), the decision epochs are the products. The action at decision epoch $i$ is whether we offer product $i$. If we drop the round down and up operators on the right side of (10), then $\Theta_{j+1}^\ell(x,y)$ gives the smallest possible capacity consumption of any assortment $S \subseteq \{j+1,\dots,\ell\}$ that satisfies $W(S) \geq x$ and $V(S) \leq y$. If there is no assortment $S \subseteq \{j+1,\dots,\ell\}$ such that $W(S) \geq x$ and $V(S) \leq y$, then $\Theta_{j+1}^\ell(x,y) = +\infty$. With the round down and up operators on the right side of (10), this dynamic program is only an approximation. Shortly, we will put bounds on the two components of the state variable $(x,y)$, in which case, noting that $(x,y) \in \textsc{Dom}^2$, the number of operations required to solve the dynamic program in (10) will be a polynomial in the input size.

To construct the collection of candidate assortments $\textsc{Cand}(j,\ell)$, we compute the value functions $\{\Theta_i^\ell(x,y) : (x,y) \in \textsc{Dom}^2,\ i = j+1,\dots,\ell+1\}$ using the dynamic program in (10). Once we compute the value functions, for each $(x,y) \in \textsc{Dom}^2$, starting with state $(x,y)$ and decision epoch $j+1$, we follow the sequence of optimal state-action pairs in the dynamic program in (10). In this way, we obtain an assortment $\widehat{S}_{x,y}$ for each $(x,y) \in \textsc{Dom}^2$, which we use as one of the candidate assortments in the collection $\textsc{Cand}(j,\ell)$. Specifically, for each $(x,y) \in \textsc{Dom}^2$, if $\Theta_{j+1}^\ell(x,y) < +\infty$, then we construct the assortment $\widehat{S}_{x,y}$ using the following algorithm. Throughout this section, we refer to this algorithm as the candidate construction algorithm.

**Candidate Construction:**

**Initialization:** Compute the value functions $\{\Theta_i^\ell(\cdot,\cdot) :\ i = j+1,\dots,\ell+1\}$ using (10). Set $i = j+1$, $\widehat{x}_i = x$, and $\widehat{y}_i = y$.

**Step 1.** Set $\widehat{u}_i = \arg\min_{u_i \in \{0,1\}} \Big\{ c_i\, u_i + \Theta_{i+1}^\ell(\lfloor \widehat{x}_i - v_i\, r_i\, u_i \rfloor \,,\, \lceil \widehat{y}_i - v_i\, u_i \rceil) \Big\}$.

**Step 2.** Set $\widehat{x}_{i+1} = \lfloor \widehat{x}_i - v_i\, r_i\, \widehat{u}_i \rfloor$ and $\widehat{y}_{i+1} = \lceil \widehat{y}_i - v_i\, \widehat{u}_i \rceil$. Increase $i$ by one. If $i < \ell+1$, then go to Step 1; otherwise, stop.

**Output:** Return $\widehat{S}_{x,y} = \{i \in \{j+1,\dots,\ell\} : \widehat{u}_i = 1\}$.

In the next lemma, we show useful properties of the assortment $\widehat{S}_{x,y}$ obtained by the algorithm above. Specifically, by the next lemma, if there exists an assortment $S \subseteq \{j+1,\dots,\ell\}$ with $W(S) \geq x$ and $V(S) \leq y$, then we have $\Theta_{j+1}^\ell(x,y) < +\infty$, so we execute the algorithm above.

In this case, considering the assortment $S \subseteq \{j+1,\ldots,\ell\}$ with $W(S) \geq x$ and $V(S) \leq y$, once again, by the next lemma, the output of the candidate construction algorithm $\widehat{S}_{x,y}$ satisfies $W(\widehat{S}_{x,y}) \geq \frac{1}{(1+\rho)^n}\, x$, $V(\widehat{S}_{x,y}) \leq (1+\rho)^n\, y$, and $C(\widehat{S}_{x,y}) \leq C(S)$. Recall that we prefer an assortment $S$ with larger $W(S)$, smaller $V(S)$ and smaller $C(S)$. Thus, if $\rho$ is small, then the candidate assortment $\widehat{S}_{x,y}$ is almost as preferable as the assortment $S$.

**Lemma 5.3 (Candidate Assortments)** *If there exists an assortment $S \subseteq \{j+1,\ldots,\ell\}$ such that $W(S) \geq x$ and $V(S) \leq y$, then we have $\Theta^{\ell}_{j+1}(x,y) < +\infty$, $W(\widehat{S}_{x,y}) \geq \frac{1}{(1+\rho)^n}\, x$, $V(\widehat{S}_{x,y}) \leq (1+\rho)^n\, y$, and $C(\widehat{S}_{x,y}) \leq C(S)$.*

The proof of the lemma is in Appendix J. Intuitively speaking, the proof is based on accumulating the errors due to the round down and up operators in (10). Next, we discuss how to use the lemma and the candidate construction algorithm to execute Part 1. Given $\epsilon \in (0,1)$, we set the accuracy parameter of the geometric grid as $\rho = \frac{1}{8\,(n+1)}\,\epsilon$. For all $S \neq \varnothing$, we have $W(S) \in [w_{\min}, n\, w_{\max}]$ and $V(S) \in [v_{\min}, n\, v_{\max}]$, so we construct the collection $\textsc{Cand}(j,\ell)$ as

$$\textsc{Cand}(j,\ell) = \Big\{ \widehat{S}_{x,y} : (x,y) \in \textsc{Dom}^2,\ x \in [\lfloor w_{\min} \rfloor, \lceil n\, w_{\max} \rceil] \cup \{0\},\ y \in [\lfloor v_{\min} \rfloor, \lceil n\, v_{\max} \rceil] \cup \{0\} \Big\}. \quad (11)$$

Noting that $\widehat{S}_{x,y} \subseteq \{j+1,\ldots,\ell\}$, we have $\widehat{S} \subseteq \{j+1,\ldots,\ell\}$ for all $\widehat{S} \in \textsc{Cand}(j,\ell)$. Thus, the collection of candidate assortments $\textsc{Cand}(j,\ell)$ given above satisfies the correct product interval property in Part 1. It remains to argue that the collection of candidate assortments $\textsc{Cand}(j,\ell)$ in (11) satisfies the limited degradation property in Part 1 as well. In the next lemma, we show that our collection of candidate assortments indeed satisfies this property.

**Lemma 5.4 (Limited Degradation)** *Considering the collection of candidate assortments $\textsc{Cand}(j,\ell)$ in (11), for each $S \subseteq \{j+1,\ldots,\ell\}$, there exists $\widehat{S} \in \textsc{Cand}(j,\ell)$ such that $W(\widehat{S}) \geq (1-\epsilon/4)\, W(S)$, $V(\widehat{S}) \leq (1+\epsilon/4)\, V(S)$, and $C(\widehat{S}) \leq C(S)$.*

*Proof:* Fix $S \subseteq \{j+1,\ldots,\ell\}$ and let $(x,y) \in \textsc{Dom}^2$ be such that $x \in [\lfloor w_{\min} \rfloor, \lceil n\, w_{\max} \rceil] \cup \{0\}$, $y \in [\lfloor v_{\min} \rfloor, \lceil n\, v_{\max} \rceil] \cup \{0\}$, $x \leq W(S) \leq (1+\rho)\, x$ and $y/(1+\rho) \leq V(S) \leq y$. Noting that we have $W(S) \in [w_{\min}, n\, w_{\max}] \cup \{0\}$ and $V(S) \in [v_{\min}, n\, v_{\max}] \cup \{0\}$, there always exists such $(x,y) \in \textsc{Dom}^2$. In this case, since we have $W(S) \geq x$ and $V(S) \leq y$, by Lemma 5.3, the candidate assortment $\widehat{S}_{x,y} \in \textsc{Cand}(j,\ell)$ satisfies the inequalities

$$W(\widehat{S}_{x,y}) \geq \frac{1}{(1+\rho)^n}\, x, \quad V(\widehat{S}_{x,y}) \leq (1+\rho)^n\, y, \quad C(\widehat{S}_{x,y}) \leq C(S).$$

Moreover, noting the fact that $W(S) \leq (1+\rho)\, x$ and $V(S) \geq y/(1+\rho)$, the first two inequalities above yield $W(\widehat{S}_{x,y}) \geq \frac{1}{(1+\rho)^{n+1}}\, W(S)$ and $V(\widehat{S}_{x,y}) \leq (1+\rho)^{n+1}\, V(S)$. For all $\delta \in [0,1/2]$ and $n \in \mathbb{Z}_+$, we have the standard inequalities $(1+\delta/n)^n \le \exp(\delta) \le 1+2\delta$. Thus, since $\epsilon/8 \le 1/2$, we get the two chains of inequalities

$$W(\widehat{S}_{x,y}) \;\ge\; \frac{1}{(1+\rho)^{n+1}}\,W(S) \;=\; \frac{1}{\left(1+\frac{\epsilon}{8\,(n+1)}\right)^{n+1}}\,W(S) \;\ge\; \frac{1}{1+\epsilon/4}\,W(S) \;\ge\; (1-\epsilon/4)\,W(S),$$

$$V(\widehat{S}_{x,y}) \le (1+\rho)^{n+1}\,V(S) = \left(1+\frac{\epsilon}{8\,(n+1)}\right)^{n+1} V(S) \le (1+\epsilon/4)\,V(S).$$

Thus, given an assortment $S \subseteq \{j+1,\ldots,\ell\}$, there exists $\widehat{S}_{x,y} \in \textsc{Cand}(j,\ell)$ such that $W(\widehat{S}_{x,y}) \ge (1-\epsilon/4)\,W(S)$, $V(\widehat{S}_{x,y}) \le (1+\epsilon/4)\,V(S)$, and $C(\widehat{S}_{x,y}) \le C(S)$. ■

Closing this section, we briefly explain that we can construct all collections of candidate assortments in $O(\frac{n^4}{\epsilon^2}\log(\frac{nw_{\max}}{w_{\min}})\log(\frac{nv_{\max}}{v_{\min}}))$ operations. In particular, to solve the dynamic program in (10), in Appendix K, we argue that the smallest nonzero values of $x$ and $y$ in the state variable $(x,y) \in \textsc{Dom}^2$ are, respectively, $\lfloor w_{\min} \rfloor$ and $\lfloor v_{\min} \rfloor$, whereas the largest values of $x$ and $y$ in the state variable $(x,y) \in \textsc{Dom}^2$ are, respectively, $\lceil n w_{\max} \rceil$ and $\lceil n v_{\max} \rceil$. In this case, noting that $\rho = \frac{1}{8\,(n+1)}\epsilon$, the number of state variables that we need to consider is

$$O\left(\frac{\log(\frac{nw_{\max}}{w_{\min}})}{\log(1+\rho)} \cdot \frac{\log(\frac{nv_{\max}}{v_{\min}})}{\log(1+\rho)}\right) \;=\; O\left(\frac{n^2}{\epsilon^2}\log\left(\frac{nw_{\max}}{w_{\min}}\right)\log\left(\frac{nv_{\max}}{v_{\min}}\right)\right). \tag{12}$$

Thus, we can compute $\Theta_i^\ell(x,y)$ for all values of the state variable $(x,y)$, $i \in \mathcal{N}$ and $\ell \in \{0,\ldots,n\}$ with $i \le \ell+1$ in $O(\frac{n^4}{\epsilon^2}\log(\frac{nw_{\max}}{w_{\min}})\log(\frac{nv_{\max}}{v_{\min}}))$ operations. The number of other operations to construct the collections of candidate assortments is negligible, resulting in the desired number of operations to construct all collections. Also, by (11), the collection $\textsc{Cand}(j,\ell)$ includes one assortment for each $(x,y) \in \textsc{Dom}^2$ such that $x \in [\lfloor w_{\min} \rfloor, \lceil n\,w_{\max} \rceil] \cup \{0\}$ and $y \in [\lfloor v_{\min} \rfloor, \lceil n\,v_{\max} \rceil] \cup \{0\}$, so by (12), the collection $\textsc{Cand}(j,\ell)$ includes $O(\frac{n^2}{\epsilon^2}\log(\frac{nw_{\max}}{w_{\min}})\log(\frac{nv_{\max}}{v_{\min}}))$ assortments.

## 5.2 Part 2: Combining Candidate Assortments

We focus on executing Part 2, which obtains an approximate solution to problem (9). We can solve problem (9) using dynamic programming. The decision epochs are the stages. At decision epoch $k$, the action is the candidate assortment $S_k$ offered, whereas the state variable keeps track of $j_k$ such that $S_{k-1} \subseteq \{j_{k-1}+1,\ldots,j_k\}$, the accumulated value of $\sum_{q=1}^{k-1} V(S_q)$, and a target expected revenue to generate from the future stages. Thus, we consider the dynamic program

$$\Psi_k(j,u,z) = \min_{\substack{(\ell,S)\,:\,\ell \in \{j,\ldots,n\},\\ S \in \textsc{Cand}(j,\ell)}} \left\{ C(S) + \Psi_{k+1}\left(\ell,\, \lceil u+V(S) \rceil,\, \left\lceil z - \frac{\lambda_k\, W(S)}{(1+u)\,(1+u+V(S))} \right\rceil \right) \right\} \tag{13}$$

with the boundary condition that $\Psi_{m+1}(j,u,z) = 0$ if $z \le 0$. Otherwise, we have $\Psi_{m+1}(j,u,z) = +\infty$. If we drop the round up operators on the right side of (13), then $\Psi_k(j,u,z)$ gives the smallest total capacity consumption of assortments $(S_k,\ldots,S_m)$ such that $S_\ell \in \textsc{Cand}(j_\ell, j_{\ell+1})$ for some $j = j_k \le j_{k+1} \le \ldots \le j_m$ and these assortments provide an expected revenue of at least $z$ in stages $k,\ldots,m$, when the assortments $(S_1,\ldots,S_{k-1})$ offered in the previous stages satisfy $\sum_{q=1}^{k-1} V(S_q) = u$. In this case, the optimal objective value of problem (9) is given by $\max\{z \in \mathbb{R} : \Psi_1(0,0,z) \le b\}$. With the round up operator, the dynamic program in (13) is only an approximation.

To obtain an approximate solution to problem (9), we use the dynamic program in (13) to compute the value functions $\{\Psi_k(j,u,z) : j = 0,\ldots,n,\ (u,z) \in \textsc{Dom}^2,\ k \in \mathcal{M}\}$. Approximating the optimal objective value of problem (9) as $\widehat{z}_{\text{APP}} = \max\{z \in \textsc{Dom} : \Psi_1(0,0,z) \le b\}$, we start with the state $(0,0,\widehat{z}_{\text{APP}})$ and follow the optimal state-action pairs in the dynamic program in (13). Specifically, we use the following algorithm to follow the optimal state-action pairs.

**Candidate Stitching:**

**Initialization:** Compute the value functions $\{\Psi_k(j,u,z) : j = 0,\ldots,n,\ (u,z) \in \textsc{Dom}^2,\ k \in \mathcal{M}\}$ using (13). Set $\widehat{z}_{\text{APP}} = \max\{z \in \textsc{Dom} : \Psi_1(0,0,z) \le b\}$. Initialize $k = 1$, $\widehat{j}_k = 0$, $\widehat{u}_k = 0$, and $\widehat{z}_k = \widehat{z}_{\text{APP}}$.

**Step 1.** Set

$$(\widehat{j}_{k+1}, \widehat{S}_k) = \underset{\substack{(\ell,S)\,:\,\ell \in \{\widehat{j}_k,\ldots,n\},\\ S \in \textsc{Cand}(\widehat{j}_k,\ell)}}{\arg\min} \left\{ C(S) + \Psi_{k+1}\left(\ell,\ \lceil \widehat{u}_k + V(S) \rceil,\ \left\lceil \widehat{z}_k - \frac{\lambda_k W(S)}{(1+\widehat{u}_k)(1+\widehat{u}_k + V(S))} \right\rceil \right) \right\}.$$

**Step 2.** Set $\widehat{u}_{k+1} = \lceil \widehat{u}_k + V(\widehat{S}_k) \rceil$ and $\widehat{z}_{k+1} = \left\lceil \widehat{z}_k - \dfrac{\lambda_k W(\widehat{S}_k)}{(1+\widehat{u}_k)(1+\widehat{u}_k + V(\widehat{S}_k))} \right\rceil$.
Increase $k$ by one. If $k < m+1$, then go to Step 1; otherwise, stop.

**Output:** Return $(\widehat{S}_1,\ldots,\widehat{S}_m)$.

Throughout this section, we refer to this algorithm as the candidate stitching algorithm, since this algorithm stitches together a solution to problem (9) using the candidate assortments for different stages. In the next lemma, we show useful properties of the output $(\widehat{S}_1,\ldots,\widehat{S}_m)$ of the candidate stitching algorithm. In particular, by the next lemma, the output of the candidate stitching algorithm is feasible for problem (9), satisfying $\sum_{k\in\mathcal{M}} C(\widehat{S}_k) \le b$. Furthermore, once again, by the next lemma, using $\textsc{Rev}(S_1,\ldots,S_m)$ to denote the expected revenue from the solution $(S_1,\ldots,S_m)$, $\widetilde{z}$ to denote the optimal objective value of problem (9), and $\widehat{z}_{\text{APP}} = \max\{z \in \textsc{Dom} : \Psi_1(0,0,z) \le b\}$ to denote our approximation of the optimal objective value of problem (9), we have $\textsc{Rev}(\widehat{S}_1,\ldots,\widehat{S}_m) \ge \widehat{z}_{\text{APP}} \ge \widetilde{z}/(1+\rho)^{3(m+1)}$. Thus, the expected revenue provided by the output of the candidate stitching algorithm is at least as large as our approximation of the optimal objective value of problem (9). Also, if $\rho$ is small, then our approximation of the optimal objective value of problem (9) is not too far from the optimal objective value of this problem. The proof of the lemma uses induction over the stages. It is in Appendix L.

**Lemma 5.5 (Stitching Candidates)** *Let* $(\widehat{S}_1,\ldots,\widehat{S}_m)$ *be the output of the candidate stitching algorithm,* $\widetilde{z}$ *be the optimal objective of problem* (9)*, and* $\widehat{z}_{\text{APP}} = \max\{z \in \text{DOM} : \Psi_1(0,0,z) \le b\}$*. We have* $\sum_{k\in\mathcal{M}} C(\widehat{S}_k) \le b$ *and* $\text{REV}(\widehat{S}_1,\ldots,\widehat{S}_m) \ge \widehat{z}_{\text{APP}} \ge \widetilde{z}/(1+\rho)^{3m+1}$.

Next, we discuss how to use the lemma and the candidate stitching algorithm to execute Part 2. Given $\epsilon \in (0,1)$, we set the accuracy parameter of the geometric grid as $\rho = \frac{1}{8(3m+1)}\,\epsilon$. Since $\epsilon/8 \le 1/2$ and $(1+\delta/n)^n \le \exp(\delta) \le 1+2\delta$ for all $\delta \in [0,1/2]$ and $n \in \mathbb{Z}_+$, by Lemma 5.5, we get

$$\text{REV}(\widehat{S}_1,\ldots,\widehat{S}_m) \ge \frac{1}{(1+\rho)^{3m+1}}\,\widetilde{z} = \frac{1}{\left(1+\frac{\epsilon}{8(3m+1)}\right)^{3m+1}}\,\widetilde{z} \ge \frac{1}{1+\frac{\epsilon}{4}}\,\widetilde{z} \ge \left(1-\frac{\epsilon}{4}\right)\widetilde{z},$$

so the output of the candidate stitching algorithm is a $(1-\frac{\epsilon}{4})$-approximate solution to problem (9), as desired. Closing this section, we explain that we can execute the candidate stitching algorithm with $\rho = \frac{1}{8(3m+1)}\,\epsilon$ in $O(\frac{n^4 m^3}{\epsilon^4}\log(\frac{n w_{\max}}{w_{\min}})\log(\frac{n w_{\max}(1\vee n v_{\max})}{\lambda_m w_{\min}})\log^2(\frac{n v_{\max}}{v_{\min}}))$ operations.

To solve the dynamic program in (13), in Appendix M, we argue that the largest values of $u$ and $z$ in the state variable $(j,u,z) \in \mathcal{N} \times \text{DOM}^2$ are, respectively, $\lceil 2n\,v_{\max}\rceil$ and $\lceil n w_{\max}\rceil$, whereas the smallest nonzero values of $u$ and $z$ in the state variable $(j,u,z) \in \mathcal{N} \times \text{DOM}^2$ are, respectively, $\lfloor v_{\min}\rfloor$ and $\lfloor \lambda_m \frac{w_{\min}}{(1+2n\,v_{\max})^2}\rfloor$. Since $j$ in the state variable $(j,u,z)$ takes $O(n)$ possible values and we set $\rho = \frac{1}{8(3m+1)}\,\epsilon$, the number of state variables we need to consider is

$$O\left(n\frac{\log(\frac{n v_{\max}}{v_{\min}})}{\log(1+\rho)} \cdot \frac{\log(\frac{n w_{\max}}{\lambda_m w_{\min}/(1+2n\,v_{\max})^2})}{\log(1+\rho)}\right) = O\left(\frac{n\,m^2}{\epsilon^2}\log\left(\frac{n v_{\max}}{v_{\min}}\right)\log\left(\frac{n w_{\max}(1\vee n\,v_{\max})}{\lambda_m\,w_{\min}}\right)\right).$$

In decision epoch $k$, there are $\sum_{p=\widehat{j}^k}^{n}\text{CAND}(\widehat{j}_k,p) = O(\frac{n^3}{\epsilon^2}\log(\frac{n w_{\max}}{w_{\min}})\log(\frac{n v_{\max}}{v_{\min}}))$ possible actions. Moreover, there are $m$ decision epochs, one for each stage. Therefore, we can solve the dynamic program in (13) in $O(\frac{n^4 m^3}{\epsilon^4}\log(\frac{n w_{\max}}{w_{\min}})\log(\frac{n w_{\max}(1\vee n v_{\max})}{\lambda_m w_{\min}})\log^2(\frac{n v_{\max}}{v_{\min}}))$ operations. The number of other operations to execute the candidate stitching algorithm is negligible, resulting in the desired number of operations to execute the candidate stitching algorithm.

In Appendix N, we tailor our FPTAS to the case where there is a constraint on the total number of offered products and slightly improve its running time. For this case, we also give an exact algorithm with running time polynomial in $n$, but exponential in $m$.

## 6. Computational Experiments

We provide three sets of computational experiments. First, we use a dataset from Expedia to check the ability of our choice model to predict customer purchases. Second, we test the performance of our approximation algorithm for the joint pricing and assortment optimization problem. Third, we test the performance of our FPTAS under space constraints. In the second and third sets, we develop upper bounds on the optimal expected revenue and use them to check optimality gaps.

### 6.1 Prediction Ability on the Dataset from Expedia

We use a dataset provided by Expedia as a part of a Kaggle competition; see Kaggle (2013). Our goal is to test the ability of our choice model to predict the purchases of customers.

**Experimental Setup:** The dataset gives the results of search queries for hotels on Expedia. In the dataset, the rows correspond to different hotels that are displayed in different search queries. The columns give information on the attributes of the displayed hotel, the results of the search query, and the booking decision of the customer. We preprocess the dataset to remove the values that are either missing or uninterpretable as a result of which, we end up with 595,965 rows and 15 columns. The first three columns in the dataset include the following information. The first column is the unique code for each search query. Using this column, we can have access to all of the hotels that are displayed in a particular search query, which is the set of products among which a particular customer makes a choice. The second column is an indicator of whether the customer booked the hotel in the search query. We use this column to identify the purchase of the customer. A customer books at most one hotel in a search query, but it is possible that she does not book any hotels. The third column is the display position of the hotel in the search query, which becomes useful when fitting our multinomial logit model with multiple stages. The remaining 12 columns give information on the characteristics of the hotel, such as the star rating, average review score, and displayed price. In Appendix O, we explain our approach for preprocessing the dataset and give a detailed discussion of the 15 columns that we use.

After processing the dataset, the 595,965 rows that we end up with represent 34,561 search queries. The average number of hotels displayed in a search query is 17.24, with the maximum number of hotels being 37. In 83% of the search queries, the customer did not make a booking. To enrich our experimental setup, we use bootstrapping on the data to generate multiple datasets. In each dataset, we vary the fraction of the search queries that did not result in a booking. There are a total of 10,000 search queries in each dataset that we bootstrap. Using $P_0$ to denote the fraction of the search queries that did not result in a booking, we sample $10,000\,P_0$ search queries among the original Expedia search queries that did not result in a booking. Similarly, among the original Expedia search queries that resulted in a booking, we sample $10,000\,(1-P_0)$ search queries. Putting these two samples together, we get a dataset with 10,000 search queries in which $P_0$ fraction of them did not result in a booking. For each value of $P_0$, we repeat the bootstrapping process 50 times to get 50 different datasets. We vary $P_0$ over $\{0.5, 0.7, 0.9\}$. In this way, we obtain 150 datasets in our computational experiments. The value of $P_0$ dictates the balance between the customers making and not making a booking in the dataset. In our choice model, we capture the preference weight of hotel $i$ in a search query by $v_i = \exp\big(\beta^0 + \sum_{\ell=1}^{12} \beta^\ell \, x_i^\ell\big)$, where $(x_i^1, \ldots, x_i^{12})$

are the values in the last 12 columns giving the characteristics of hotel $i$ and $(\beta^0, \beta^1, \ldots, \beta^{12})$ are coefficients that we estimate from the dataset. Therefore, the parameters of our choice model are the coefficients $(\beta^0, \beta^1, \ldots, \beta^{12})$ and the patience level distribution.

We randomly split each dataset into training, validation, and testing data, each of which, respectively, includes 64%, 16%, and 20% of the search queries. The data provides the display position of each hotel in the search query, but fitting our choice model requires having access to the stage in which each hotel is displayed. We proceed under the assumption that each stage corresponds to $b$ hotels in consecutive display positions and choose the best value of $b$ using cross-validation. Specifically, we use the values of $b \in \{1, 3, 5, 10, 20\}$. For each value of $b$, we use maximum likelihood to fit our choice model to the training data and check the log-likelihood of our fitted choice model on the validation data. We choose the value of $b$ that provides the largest log-likelihood on the validation data. See, for example, the approach used by Vulcano et al. (2012) to fit choice models using maximum likelihood. As a benchmark, we also fit a standard multinomial logit model to the training data. The preference weight of hotel $i$ under this choice model is $v_i = \exp\big(\beta_0 + \sum_{\ell=1}^{12} \beta_\ell \, x_\ell\big)$. In Appendix P, we compare the runtimes for fitting the two choice models. Throughout this section, we refer to our multinomial logit model with impatient customers as IML and the standard multinomial logit model as SML.

**Computational Results:** We use two performance measures to compare IML and SML. The first measure is the out-of-sample log-likelihood on the testing data. The second is the $k$-hit score on the testing data. To compute the $k$-hit score of the fitted IML model, we use $\mathcal{T}$ to denote the set of search queries in the testing data in which the customer made a booking. For each $t \in \mathcal{T}$, we let $S_t$ be the assortment of hotels offered in this search query and $i_t$ be the hotel booked. Using $\phi_i(S)$ to denote the purchase probability of hotel $i$ within assortment $S$ under the fitted IML model, for each $t$, we let $A_t^k$ be the set of $k$ alternatives with the largest purchase probabilities, which are given by the $k$ largest elements of $\{\phi_i(S_t) : i \in S_t\}$. If $i_t \in A_t^k$, then the hotel booked in search query $t$ has one of the $k$ largest choice probabilities under the fitted IML model. So, the $k$-hit score of the fitted IML model is $\frac{1}{|\mathcal{T}|} \sum_{t \in \mathcal{T}} \mathbf{1}(i_t \in A_t^k)$. The $k$-hit score of the fitted SML model is similar. We use $k \in \{1, 2, 3\}$. For the $k$-hit score, we focus only on the search queries resulting in a booking, because a large fraction of the customers do not book. If we included the search queries not resulting in a booking in the $k$-hit score, then the $k$-hit scores would be driven mainly by the customers who do not book, but we want to test our ability to predict the specific hotel booked.

We give our computational results in Table 1. Each row in the table corresponds to a different value of $P_0$. Recall that we generate 50 datasets for each value of $P_0$. In the top portion, we compare the out-of-sample log-likelihoods of IML and SML. The first column shows the number of datasets

| | Out-of-Sample Log-Likelihood | | | | |
|---|---|---|---|---|---|
| $P_0$ | IML≻ SML | IML Like. | SML Like. | Avg. %Gap | S. Er. %Gap |
| 0.5 | 50 | −3899.65 | −3963.64 | 1.64% | 0.07% |
| 0.7 | 50 | −2701.65 | −2766.36 | 2.40% | 0.09% |
| 0.9 | 47 | −1145.78 | −1169.93 | 2.12% | 0.18% |

| | 1-Hit Score | | | | 2-Hit Score | | | | 3-Hit Score | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $P_0$ | IML≻ SML | IML 1-hit | Avg. %Gap. | Std. Err. | IML≻ SML | IML 2-hit | Avg. %Gap | Std. %Gap | IML≻ SML | IML 3-hit | Avg. %Gap. | Std. %Gap | $\lvert\mathcal{T}\rvert$ |
| 0.5 | 36 | 0.25 | 3.81% | 0.88% | 37 | 0.39 | 3.48% | 0.61% | 36 | 0.50 | 2.04% | 0.57% | 1003.04 |
| 0.7 | 42 | 0.24 | 6.61% | 0.91% | 42 | 0.37 | 4.24% | 0.71% | 32 | 0.48 | 3.11% | 0.68% | 599.06 |
| 0.9 | 34 | 0.22 | 3.04% | 2.08% | 35 | 0.36 | 5.80% | 1.57% | 40 | 0.47 | 7.74% | 1.16% | 199.84 |

**Table 1** Comparison of the fitted IML and SML models on the dataset from Expedia.

out of 50 where the out-of-sample log-likelihood of the fitted IML model is larger than that of SML. The second and third columns, respectively, show the average out-of-sample log-likelihood of the fitted IML and SML models, where the average is over the 50 datasets. The fourth and fifth columns, respectively, show the average and standard error of the percent gaps between the out-of-sample log-likelihoods of the two fitted choice models, where the standard error is the standard deviation of the percent gaps over the 50 datasets divided by $\sqrt{50}$. In the bottom portion, we compare the $k$-hit scores. The first column shows the number of datasets out of 50 where the 1-hit score of the fitted IML model is larger than that of SML. The second column shows the average 1-hit score of the fitted IML model over the 50 datasets. The fourth and fifth columns, respectively, show the average and standard error of the percent gaps between the 1-hit scores of the two fitted choice models. Positive values favor IML. We compare the 2-hit and 3-hit scores similarly. Lastly, to put the $k$-hit scores in perspective, we give the average of $|\mathcal{T}|$ over the 50 datasets.

The fitted IML model improves the out-of-sample log-likelihoods of the fitted SML model in 147 out of 150 datasets. To quantify the improvements in the prediction accuracies more clearly, we turn to $k$-hit scores. The fitted IML model improves the 1-hit score of the fitted SML model in 112 out of 150 datasets, providing an average improvement of 4.49%. Noting the 3-hit scores, one of the three alternatives with the largest purchase probabilities ends up being the hotel booked by the customer about 50% of the time. The gaps between the $k$-hit scores are maintained for $k \in \{2, 3\}$, but for large values of $k$, the $k$-hit scores for both choice models will naturally be one, since as $k$ gets large, the hotel booked by the customer will be one of a large number of hotels with a large probability. Our bootstrapped datasets are independent samples. All average gaps in the out-of-sample log-likelihoods and $k$-hit scores, except for one, are statistically significant in paired $t$-test at the 99% level; see Chapter 4.6 in Goulden (1939). For $P_0 = 0.9$, the average gap in the 1-hit scores is statistically significant at the 90% level.

To get a feel for how the different characteristics of the hotels affect their mean utilities, in Table 2, we provide the estimated values for the coefficients $(\beta^1, \ldots, \beta^{12})$ in the fitted IML model. Recall

| Attribute | Coeff. | Attribute | Coeff. | Attribute | Coeff. |
|---|---|---|---|---|---|
| Star rating | 0.287 | Accessibility score | 0.386 | Number of days until stay | −0.111 |
| Average review score | 0.110 | Average historical price | −0.038 | Number of adults staying | −0.128 |
| Part of a chain indicator | 0.116 | Displayed price | −1.202 | Number of children staying | 0.065 |
| Location score | −0.192 | Promotion indicator | 0.127 | Saturday stay indicator | −0.051 |

**Table 2** Estimated values for the 12 coefficients in the fitted IML model averaged over the bootstrapped datasets.

that the preference weight of hotel $i$ is given by $v_i = \exp\big(\beta^0 + \sum_{\ell=1}^{12} \beta^\ell \, x_i^\ell\big)$, where $(x_i^1, \ldots, x_i^{12})$ are the values in the last 12 columns giving the characteristics of hotel $i$, such as the star rating, average review score, and displayed price. Before fitting choice models, we shift and scale the entries in each of the 12 columns of the dataset so that the entries in each column have mean zero and variance one, in which case, the entries corresponding to different characteristics of the hotels have roughly the same order of magnitude. The estimated values for the coefficients $(\beta^1, \ldots, \beta^{12})$ are relatively stable from one bootstrapped dataset to another, so we provide the average of the estimated parameters over the bootstrapped datasets. Going over some of the values for the coefficients, not surprisingly, larger star rating and larger average review score positively impact the mean utility, whereas larger displayed price negatively impacts the mean utility. Being part of a hotel chain, providing brand familiarity to the customer, positively impacts the mean utility. More interestingly, larger number of days until the actual day of stay and duration of stay including a Saturday night negatively impact the mean utility. It is reasonable that the customers booking earlier and staying over Saturday night are leisure travelers, so they are more likely to leave without making a booking, resulting in a smaller mean utility. The effects of some characteristics, such as the location and accessibility scores, and the numbers of adults and children on the booking, are harder to interpret, but all of the estimated coefficients are statistically significant with $p$-values less than $10^{-5}$ when we use the $t$-test to test the null hypothesis that a coefficient is zero; see Chapter 3.1.2 in James et al. (2014).

## 6.2 Joint Pricing and Assortment Optimization

For joint pricing and assortment optimization, based on Theorem 4.4, we give a simple neighborhood search algorithm with 87.8% performance guarantee and test its performance.

**Experimental Setup:** In our neighborhood search algorithm, we start with a solution to the Pricing-Assortment problem that offers all products in the first stage and charges the corresponding optimal prices for the products. By Theorem 4.4, this solution provides at least 87.8% of the optimal expected revenue. Given the current sequence of assortments that we offer, we check all neighbors of the current sequence of assortments, using an appropriately defined neighborhood. For each sequence of assortments in the neighborhood, we compute the corresponding optimal prices to charge. Among all sequences of assortments in the neighborhood and their corresponding prices, we pick the one that provides the largest expected revenue. Recall that we can use the approach in Section 4.1 to compute the optimal prices to charge for a given sequence of assortments. We update the current sequence of assortments to be this best sequence in the neighborhood and repeat the process starting from the updated current sequence of assortments. If no sequence of assortments in the neighborhood, along with the corresponding optimal prices, improves the expected revenue from the current sequence of assortments, then we stop. In our neighborhood search algorithm, we define the neighborhood of a sequence of assortments as all sequences of assortments obtained by moving one product from one stage to another, so the neighborhood of the sequence of assortments $(S_1,\ldots,S_m)$ is $\big\{(S_1,\ldots,S_k\setminus\{i\},\ldots,S_\ell\cup\{i\},\ldots,S_m):\forall\, i\in S_k,\ k,\ell\in\mathcal{M},\ k\neq\ell\big\}$. To complement the neighborhood search algorithm, in Appendix Q, we also give an efficiently computable upper bound on the optimal expected revenue in the Pricing-Assortment problem. In our computational experiments, we randomly generate a large number of test problems and compare the expected revenue from the solution obtained by our neighborhood search algorithm with the upper bound on the optimal expected revenue.

In all of our test problems, the number of products is $n=20$ and the price sensitivity is $\beta=1$. Working with other values for the price sensitivity is equivalent to scaling the prices of the products with the same constant. We use the following approach to come up with the parameters $\{\alpha_i : i\in\mathcal{N}\}$. We have $C$ product clusters. We randomly assign each product to a cluster. If products $i$ and $j$ are in the same cluster, then the values of $\alpha_i$ and $\alpha_j$ are close. Specifically, cluster $c$ has the centroid $\gamma_c$. We set the centroid of cluster $c$ as $\gamma_c=c-0.5$ for all $c=1,\ldots,C$. If product $i$ belongs to cluster $c$, then we generate $\kappa_i$ from the normal distribution with mean $\gamma_c$ and standard deviation $\sigma$, where $\sigma$ is a parameter that we vary. We set $\alpha_i=\kappa_i-\Delta$, where we have $\Delta=\log\sum_{i\in\mathcal{N}}e^{\kappa_i}-\log 9$. In this case, if we offer all products in the first stage and charge a price of zero for them, then a customer leaves without a purchase with probability 0.1. Using the random variable $Y$ to capture the patience level of a customer, the probability mass function of $Y$ is given by $\mathbb{P}\{Y=k\}=e^{a\cdot k}/\sum_{\ell\in\mathcal{M}}e^{a\cdot\ell}$, where $a$ is another parameter that we vary. Negative and positive values for $a$ yield, respectively, left-skewed and right-skewed distributions. If $a=+\infty$, then $Y=m$ with probability one, so the customers are willing to wait until the last stage.

Recalling that $m$ is the number of stages, varying $m\in\{6,8,10\}$, $C\in\{3,5\}$, $\sigma\in\{0.5,1.0\}$, and $a\in\{+\infty,0.5,0.0,-0.1\}$, we obtain 48 parameter configurations. In each parameter configuration, we generate 25 problem instances using the approach in the previous paragraph.

**Computational Results:** We show our computational results in Table 3. In this table, the first column shows the parameter configuration using the tuple $(C,\sigma,a)$, where $C$, $\sigma$ and $a$ are as discussed in our experimental setup. In the rest of the table, there are three blocks, each with four

| Param. Conf. $(C,\sigma,a)$ | $m=6$ Avg. Gap | Max. Gap | Std. Gap | Avg. Imp. | $m=8$ Avg. Gap | Max. Gap | Sd. Gap | Avg. Imp. | $m=10$ Avg. Gap | Max. Gap | Std. Gap | Avg. Imp. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $(3, 0.5, +\infty)$ | 1.38% | 1.59% | 0.09% | 4.98% | 1.95% | 2.17% | 0.09% | 4.99% | 2.52% | 2.74% | 0.09% | 4.99% |
| $(3, 0.5, 0.5)$ | 1.15% | 1.17% | 0.01% | 2.73% | 1.63% | 1.66% | 0.01% | 3.70% | 2.12% | 2.22% | 0.03% | 4.28% |
| $(3, 0.5, 0.0)$ | 0.78% | 0.79% | 0.00% | 0.55% | 1.04% | 1.10% | 0.01% | 1.01% | 1.29% | 1.38% | 0.02% | 1.36% |
| $(3, 0.5, -0.1)$ | 0.70% | 0.70% | 0.00% | 0.24% | 0.89% | 0.90% | 0.01% | 0.50% | 1.06% | 1.07% | 0.00% | 0.71% |
| $(3, 1.0, +\infty)$ | 1.53% | 2.61% | 0.30% | 4.82% | 2.10% | 3.18% | 0.31% | 4.83% | 2.67% | 3.75% | 0.31% | 4.83% |
| $(3, 1.0, 0.5)$ | 1.18% | 1.57% | 0.08% | 2.70% | 1.66% | 2.12% | 0.12% | 3.66% | 2.20% | 2.86% | 0.17% | 4.21% |
| $(3, 1.0, 0.0)$ | 0.78% | 0.90% | 0.02% | 0.55% | 1.03% | 1.04% | 0.00% | 1.01% | 1.29% | 1.38% | 0.02% | 1.36% |
| $(3, 1.0, -0.1)$ | 0.70% | 0.70% | 0.00% | 0.24% | 0.89% | 0.92% | 0.01% | 0.50% | 1.06% | 1.14% | 0.02% | 0.70% |
| $(5, 0.5, +\infty)$ | 1.49% | 1.77% | 0.16% | 4.87% | 2.06% | 2.34% | 0.17% | 4.88% | 2.63% | 2.92% | 0.17% | 4.88% |
| $(5, 0.5, 0.5)$ | 1.16% | 1.18% | 0.01% | 2.73% | 1.64% | 1.72% | 0.03% | 3.69% | 2.15% | 2.27% | 0.05% | 4.25% |
| $(5, 0.5, 0.0)$ | 0.78% | 0.79% | 0.00% | 0.55% | 1.04% | 1.09% | 0.02% | 1.00% | 1.29% | 1.33% | 0.01% | 1.36% |
| $(5, 0.5, -0.1)$ | 0.70% | 0.71% | 0.00% | 0.24% | 0.89% | 0.91% | 0.01% | 0.50% | 1.06% | 1.11% | 0.02% | 0.70% |
| $(5, 1.0, +\infty)$ | 1.84% | 2.95% | 0.46% | 4.49% | 2.41% | 3.52% | 0.46% | 4.50% | 2.98% | 4.09% | 0.46% | 4.50% |
| $(5, 1.0, 0.5)$ | 1.21% | 2.28% | 0.22% | 2.67% | 1.78% | 2.59% | 0.24% | 3.54% | 2.36% | 3.25% | 0.31% | 4.04% |
| $(5, 1.0, 0.0)$ | 0.78% | 0.86% | 0.02% | 0.55% | 1.07% | 1.21% | 0.06% | 0.97% | 1.31% | 1.46% | 0.05% | 1.34% |
| $(5, 1.0, -0.1)$ | 0.70% | 0.75% | 0.01% | 0.23% | 0.88% | 0.89% | 0.00% | 0.50% | 1.07% | 1.40% | 0.07% | 0.69% |

**Table 3** Performance of the neighborhood search algorithm for joint pricing and assortment optimization.

columns. Each block corresponds to a particular value for the parameter $m$. In each block, the first column shows the average percent gap between the upper bound on the optimal expected revenue and the expected revenue from the solution obtained by our neighborhood search algorithm, where the average is computed over the 25 problem instances in a parameter configuration. Specifically, using $\mathsf{Rev}^k$ and $\mathsf{UB}^k$ to denote, respectively, the expected revenue from the solution obtained by the neighborhood search algorithm and the upper bound on the optimal expected revenue for problem instance $k$, the first column shows the average of the data $\{100 \cdot \frac{\mathsf{UB}^k - \mathsf{Rev}^k}{\mathsf{UB}^k} : k = 1, \ldots, 25\}$. The second and third columns, respectively, show the maximum and standard deviation of the same data. The fourth column shows the average percent gap between the expected revenues from the initial and final solutions in the neighborhood search algorithm, capturing the improvement provided by this algorithm over using the solution that offers all products in the first stage.

Our results indicate that our neighborhood search algorithm performs quite well. Over all of our test problems, the average gap between the upper bound on the optimal expected revenue and the expected revenue from the neighborhood search algorithm is 1.43%. The gaps tend to increase as the number of stages gets larger. Without knowing the optimal expected revenue, it is difficult to tell whether the increase in the gaps is due to a degradation in the upper bounds or a degradation in the expected revenues from the neighborhood search algorithm. However, the upper bound that we give in Appendix Q is based on treating $(\sum_{i \in S_1} e^{\alpha_i}, \ldots, \sum_{i \in S_m} e^{\alpha_i})$ in the expected revenue expression in (5) as continuous quantities whose sum does not exceed $\sum_{i \in \mathcal{N}} e^{\alpha_i}$. Intuitively speaking, this assumption becomes harder to justify when the number of stages gets larger. Overall, the performance of the neighborhood search algorithm is better than its 87.8% theoretical performance guarantee. The improvement provided by the neighborhood search algorithm over the initial solution that offers all products in the first stage can get as large as 4.99%. The improvements are most visible for problem instances with $a \in \{+\infty, 0.5\}$. For these problem instances, the patience level distribution has a significant mass for larger patience levels, so the customers tend to have larger patience levels, in which case, it becomes more important to explore solutions that offer assortments in the later stages. On the other hand, for problem instances with $a \in \{0.0, -0, 1\}$, the patience level distribution has a significant mass for smaller patience levels. In this case, the customers tend to have smaller patience levels, so focusing on solutions that offer assortments only in a few earlier stages appears to be adequate to get good solutions. For problem instances with $a = +\infty$, $a = 0.5$, $a = 0.0$, and $a = -0.1$, the average number of neighbors that the neighborhood search algorithm visits before termination are, respectively, 38.31, 9.02, 1.89, and 1.74, which also indicates that as the value of $a$ gets larger, so that the customers tend stay in the system for larger number of stages before they run of patience, it becomes more important to explore solutions that offer assortments in later stages. The runtime for the neighborhood search algorithm ranges from 0.11 and 5.14 seconds, with larger runtimes corresponding to problem instances with larger values for $m$ and larger values for $a$.

### 6.3 Assortment Optimization under a Space Constraint

We test the practical performance of the FPTAS that we give in Section 5 for the assortment optimization problem under a space constraint.

**Experimental Setup:** To assess the optimality gaps for the solutions obtained by our FPTAS, in Appendix R, we give an efficiently computable upper bound on the optimal objective value of the CAPACITATED problem. In our computational experiments, we randomly generate a large number of test problems and compare the expected revenue from the solution obtained by our FPTAS with the upper bound on the optimal expected revenue. We use the following approach to generate our test problems. In all of our test problems, the number of products is $n = 20$. To come up with the revenue $r_i$ of product $i$, we generate $r_i$ from the uniform distribution over $[1, 10]$. We reindex $(r_1, \ldots, r_n)$ so that $r_1 \geq r_2 \geq \ldots \geq r_n$. To come up with the preference weight $v_i$ of product $i$, we generate $\gamma_i$ from the uniform distribution $[1, 10]$ and set $v_i = \gamma_i / \Delta$, where we have $\Delta = P_0 \sum_{i \in \mathcal{N}} \gamma_i / (1 - P_0)$ and $P_0$ is a parameter that we vary. In this case, if we offer all products in the first stage, then a customer will leave without a purchase with probability $P_0$. After generating the preference weights, we process them to come up with two problem classes for the preference weights. In the first problem class, we leave the preference weights untouched. In the second problem class, we reindex $(v_1, \ldots, v_n)$ so that $v_1 \leq v_2 \leq \ldots \leq v_n$. Thus, recalling that $r_1 \geq r_2 \geq \ldots \geq r_n$, in the second problem class, the products with larger revenues have smaller preference weights, so the more expensive products are less attractive. We refer to the first and second problem classes, respectively, as "U" and "O," where "U" stands for unordered and "O" stands for ordered. We use the same approach that we use in Section 6.2 for the joint pricing and assortment optimization problem to come up with the distribution for the patience levels. Recall that the parameter $a$ controls the skewness of the distribution of the patience levels. In all of our test problems, to come up with the space consumptions $\{c_i : i \in \mathcal{N}\}$ and the space availability $b$, we generate $c_i$ from the uniform distribution over $[0,1]$ and set $b = 5$.

Using $T$ denote the problem class for the preference weights, varying $m \in \{6,8,10\}$, $P_0 \in \{0.1, 0,3\}$, $T \in \{\text{U}, \text{O}\}$, and $a \in \{+\infty, 0.5, 0.0, -0.1\}$, we obtain 48 parameter configurations. We generate 25 problem instances in each parameter configuration.

**Computational Results:** We executed our FPTAS with $\epsilon = 1/2$ to obtain a $\frac{1}{2}$-approximate solution to the Capacitated problem. Even with this setting, our FPTAS obtains solutions with expected revenues within 5% of the upper bound on the optimal expected revenue. The large number of test problems in our experimental setup prevented us from reporting results for theoretical performance guarantees better than 50%, but a limited number of runs indicated that if we use $\epsilon = \frac{1}{4}$, then we decrease the percent gap between the upper bound and the performance of our FPTAS by about 1%. We show our computational results in Table 4. The layout of this table is similar to that of Table 3. In the first column, we use the tuple $(P_0, T, a)$ to show the parameter configuration. The rest of the table has three blocks of three columns. Each block corresponds to a particular value for the parameter $m$. In each block, the three columns, respectively, show the average, maximum, and standard deviation for the percent gap between the upper bound on the optimal expected revenue and the expected revenue from the solution obtained by our FPTAS, where the average, maximum, and standard deviation are computed over the 25 problem instances in a parameter configuration. Over all of our test problems, the average gap between the upper bound and the expected revenue from our FPTAS is 2.25% and the maximum gap is 4.47%. The gaps increase only slightly as the number of stages gets larger. Overall, the performance of our FPTAS is substantially stronger than its theoretical performance guarantee of 50%.

The runtime for our FPTAS ranges from 26.23 to 36.12 minutes. Note that the size of our test problems makes full enumeration impossible, because the number of possible sequences of assortments is $O(m^n)$. Considering the candidate construction and candidate stitching algorithms in Sections 5.1 and 5.2, a major portion of the runtime is spent for candidate construction, particularly due to the fact that many of the candidate assortments that we construct end up being duplicates of each other, resulting in substantial savings in the runtime for candidate stitching. The

| | $m=6$ | | | $m=8$ | | | $m=10$ | | |
|---|---|---|---|---|---|---|---|---|---|
| Param. Conf. $(P_0,T,a)$ | Avg. Gap | Max. Gap | Std. Gap | Avg. Gap | Max. Gap | Sd. Gap | Avg. Gap | Max. Gap | Std. Gap |
| $(0.1,\mathrm{U},+\infty)$ | 2.92% | 3.88% | 0.56% | 3.18% | 4.20% | 0.56% | 3.27% | 4.31% | 0.58% |
| $(0.1,\mathrm{U},\ \ 0.5)$ | 2.40% | 3.22% | 0.42% | 2.53% | 3.44% | 0.56% | 2.79% | 3.71% | 0.47% |
| $(0.1,\mathrm{U},\ \ 0.0)$ | 2.17% | 3.09% | 0.49% | 2.28% | 3.71% | 0.51% | 2.19% | 3.73% | 0.65% |
| $(0.1,\mathrm{U},-0.1)$ | 2.12% | 3.09% | 0.55% | 2.09% | 3.07% | 0.50% | 2.11% | 3.21% | 0.56% |
| $(0.1,\mathrm{O},+\infty)$ | 2.77% | 4.00% | 0.60% | 3.11% | 4.27% | 0.60% | 3.30% | 4.47% | 0.59% |
| $(0.1,\mathrm{O},\ \ 0.5)$ | 2.14% | 2.77% | 0.40% | 2.38% | 3.72% | 0.60% | 2.72% | 4.00% | 0.57% |
| $(0.1,\mathrm{O},\ \ 0.0)$ | 1.96% | 3.32% | 0.62% | 2.15% | 3.34% | 0.61% | 1.99% | 3.81% | 0.53% |
| $(0.1,\mathrm{O},-0.1)$ | 2.25% | 3.54% | 0.53% | 2.01% | 2.93% | 0.54% | 2.03% | 3.04% | 0.54% |
| $(0.3,\mathrm{U},+\infty)$ | 2.52% | 3.58% | 0.45% | 2.80% | 3.84% | 0.43% | 2.93% | 4.08% | 0.46% |
| $(0.3,\mathrm{U},\ \ 0.5)$ | 2.15% | 3.22% | 0.52% | 2.25% | 3.01% | 0.45% | 2.16% | 3.03% | 0.47% |
| $(0.3,\mathrm{U},\ \ 0.0)$ | 1.87% | 2.71% | 0.40% | 1.78% | 2.59% | 0.45% | 1.87% | 2.86% | 0.49% |
| $(0.3,\mathrm{U},-0.1)$ | 1.80% | 2.65% | 0.43% | 1.79% | 2.71% | 0.42% | 2.02% | 2.94% | 0.45% |
| $(0.3,\mathrm{O},+\infty)$ | 2.37% | 3.20% | 0.47% | 2.69% | 3.58% | 0.46% | 2.85% | 3.85% | 0.46% |
| $(0.3,\mathrm{O},\ \ 0.5)$ | 1.74% | 2.67% | 0.50% | 1.96% | 3.19% | 0.52% | 2.07% | 2.99% | 0.43% |
| $(0.3,\mathrm{O},\ \ 0.0)$ | 1.59% | 2.73% | 0.55% | 1.52% | 2.48% | 0.46% | 1.73% | 2.63% | 0.53% |
| $(0.3,\mathrm{O},-0.1)$ | 1.50% | 2.52% | 0.53% | 1.54% | 2.47% | 0.48% | 1.55% | 2.75% | 0.52% |

**Table 4** Performance of the FPTAS for assortment optimization under a space constraint.

runtime for the candidate stitching algorithm ranges from 2.08 to 18.09 seconds, where the larger runtimes correspond to the test problems with larger number of stages. The remaining portion of the runtime is for the candidate construction algorithm. In Table 5, we give the runtime in minutes for our FPTAS for the values of $\epsilon \in \{3/4, 1/2, 1/4, 1/8\}$, averaged over four representative problem instances with $m=10$ stages. We use four problem instances, as the runtime with $\epsilon = 1/8$ is a few hours. In our computational experiments, the runtime of our FPTAS increases roughly quadratically with $\frac{1}{\epsilon}$, due to the fact that many of the candidate assortments that we construct, as mentioned earlier in this paragraph, end up being duplicates of each other.

## 7. Conclusions

Our work in this paper opens up several research directions to pursue. We can explore efficient solution methods for the assortment optimization problem when there is a constraint on the space consumption or the number of products offered in each stage. An analogue of Lemma 5.1 does not hold under a constraint on the space consumption or the number of products offered in each stage. That is, we can have a pair of products such that it is optimal to offer the one with the smaller revenue in an earlier stage and the one with the larger revenue in a later stage. For example, consider a problem instance with three products and two stages. The revenues and preference weights of the products are $r_1 = 10$, $r_2 = 7$, $r_3 = 6$, $v_1 = 0.4$, $v_2 = 0.9$ and $v_3 = 0.3$. The distribution of the patience levels is given by $\lambda_1 = \lambda_2 = 1$. If we can offer at most two products in the first stage and at most one product in the second stage, then the unique optimal assortments in the first and second stages are, respectively, $\{1, 3\}$ and $\{2\}$, offering product 3 with revenue 6 in the first stage and product 2 with revenue 7 in the second stage. Similarly, we have not been able to characterize

| $\epsilon$ | 3/4 | 1/2 | 1/4 | 1/8 |
|---|---|---|---|---|
| Runtime (Min.) | 14.37 | 31.05 | 118.43 | 464.27 |

**Table 5** Runtime for the FPTAS for different values of $\epsilon$.

structural properties of a near-optimal sequence of assortments to offer in the joint pricing and assortment optimization problem when there is a constraint on the number of products offered in each stage. Thus, the joint pricing and assortment optimization problem under a constraint on the number of products offered in each stage is an open problem. Considering another research direction, in Appendix D, we give a dynamic program to find a solution for the Pricing problem with an additive performance guarantee. It would be useful to extend this work to find a solution with a multiplicative performance guarantee without using convex optimization tools.

Also, the running time of our FPTAS is $O(\frac{n^4 m^3}{\epsilon^4}\log(\frac{n w_{\max}}{w_{\min}})\log(\frac{n w_{\max}(1\vee n v_{\max})}{\lambda_m w_{\min}})\log^2(\frac{n v_{\max}}{v_{\min}}))$, depending on the parameters $w_{\max}$, $w_{\min}$, $v_{\max}$, $v_{\min}$, and $\lambda_m$. We can work on removing the dependence on these parameters to obtain a strongly polynomial running time. We can remove these dependencies partially. In particular, when solving the dynamic program in (10), we can guess the largest value of $v_i r_i$ for an offered product $i$ and the largest value of $v_j$ for an offered product $j$. Letting $\widehat{w}$ and $\widehat{v}$ be these two guesses, we can argue that the largest values of $x$ and $y$ in the state variable $(x,y)\in \text{Dom}^2$ would, respectively, be $\lceil n\widehat{w}\rceil$ and $\lceil n\widehat{v}\rceil$, whereas the smallest nonzero values of $x$ and $y$ in the state variable $(x,y)\in\text{Dom}^2$ would, respectively, be $\lfloor\widehat{w}\rfloor$ and $\lfloor\widehat{v}\rfloor$. Thus, noting that there are $n^2$ possible guesses for $(\widehat{w},\widehat{v})$, the number of candidate assortments in the collection $\text{Cand}(j,\ell)$ would be $O(\frac{n^4}{\epsilon^2}(\log n)^2)$, which is strongly polynomial, but it is not necessarily smaller than the number of candidate assortments $O(\frac{n^2}{\epsilon^2}\log(\frac{n w_{\max}}{w_{\min}})\log(\frac{n v_{\max}}{v_{\min}}))$ discussed at the end of Section 5.1. In this case, following the same line of analysis in Section 5.2, the running time of our FPTAS would be $O(\frac{n^6 m^3}{\epsilon^4}(\log n)^2\log(\frac{n v_{\max}}{v_{\min}})\log(\frac{n w_{\max}(1\vee n v_{\max})}{\lambda_m w_{\min}}))$. We can use a similar idea for the dynamic program in (13) to deal with $u$ in the state variable $(j,u,z)\in\mathcal{N}\times\text{Dom}^2$, yielding the running time $O(\frac{n^7 m^3}{\epsilon^4}(\log n)^3\log(\frac{n w_{\max}(1\vee n v_{\max})}{\lambda_m w_{\min}}))$ for our FPTAS, but dealing with $z$ in the state variable $(j,u,z)\in\mathcal{N}\times\text{Dom}^2$ appears to be difficult due to the term $\lambda_k$ in (13).

Lastly, in Appendix H, we show that the joint pricing and assortment optimization problem is NP-hard when the prices take values over a discrete set. We do not know the computational complexity of the problem when the prices take values over a continuum. Similarly, in Appendix N, considering the case where there is a constraint on the total number of offered products, we give an algorithm to find the optimal sequence of assortments to offer, but the running time of this algorithm increases exponentially with the number of stages. We do not know the complexity of the problem when the number of stages is also an input.

**Acknowledgements:** The authors thank the associate editor and two anonymous referees. This work was supported by NSF grant CMMI 1825406. The current proof of Theorem 4.1 was suggested by one of the referees, simplifying our earlier proof.

## References


Aouad, A., V. Farias, R. Levi. 2016. Assortment optimization under consider-then-choose choice models. Tech. rep., MIT, Massachusetts, MA.

Aouad, A., J. Feldman, D. Segev, D. J. Zhang. 2019. Click-based MNL: Algorithmic frameworks for modeling click data in assortment optimization. Tech. rep., Washington University, St. Louis, MO.

Aouad, A., D. Segev. 2018. Display optimization for vertically differentiated locations under multinomial logit preferences. Tech. rep., MIT, Massachusetts, MA.

Blanchet, J., G. Gallego, V. Goyal. 2016. A Markov chain approximation to choice modeling. *Operations Research* **64**(4) 886–905.

Boyd, S., L. Vandenberghe. 2004. *Convex optimization*. Cambridge University Press.

Bront, J. J. M., I. Mendez Diaz, G. Vulcano. 2009. A column generation algorithm for choice-based network revenue management. *Operations Research* **57**(3) 769–784.

Davis, J. M., G. Gallego, H. Topaloglu. 2014. Assortment optimization under variants of the nested logit model. *Operations Research* **62**(2) 250–273.

Derakhshan, M., N. Golrezaei, V. Manhshadi, V. Mirrokni. 2018. Product ranking on online platforms. Tech. rep., MIT, Massachusetts, MA.

Desir, A., V. Goyal, S. Jagabathula, D. Segev. 2016a. Assortment optimization under the Mallows model. D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, R. Garnett, eds., *Advances in Neural Information Processing Systems 29*. Curran Associates, Inc., 4700–4708.

Desir, A., V. Goyal, J. Zhang. 2016b. Near-optimal algorithms for capacity constrained assortment optimization. Tech. rep., Columbia University, New York, NY.

Farias, Vivek F., Srikanth Jagabathula, Devavrat Shah. 2013. A non-parametric approach to modeling choice with limited data. *Management Science* **59**(2) 305–322.

Feldman, J., A. Paul, H. Topaloglu. 2019. Assortment optimization with small consideration sets. *Operations Research* (to appear).

Feldman, J., D. Segev. 2019. Improved approximation schemes for MNL-driven sequential assortment optimization. Tech. rep., Washington University, St. Louis, MO.

Flores, A., G. Berbeglia, P. van Hentenryck. 2019. Assortment optimization under the sequential multinomial logit model. *European Journal of Operational Research* **273**(3) 1052–1064.

Gallego, G., G. Iyengar, R. Phillips, A. Dubey. 2004. Managing flexible products on a network. CORC Technical Report TR-2004-01.

Gallego, G., A. Li, V. A. Truong, X. Wang. 2016. Approximation algorithms for product framing and pricing. *Operations Research* (to appear).

Gallego, G., R. Wang. 2014. Multi-product price optimization and competition under the nested attraction model. *Operations Research* **62**(2) 450–461.

Garey, M. R., D. S. Johnson. 1979. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, New York, NY.

Goulden, C. H. 1939. *Methods of Statistical Analysis*. John Wiley & Sons, New York, NY.

Hoorfar, A., M. Hassani. 2008. Inequalities on the Lambert-$W$ function and hyperpower function. *Journal of Inequalities in Pure & Applied Mathematics* **9**(2) 51.1–51.11.

Hopp, W. J., X. Xu. 2005. Product line selection and pricing with modularity in design. *Manufacturing & Service Operations Management* **7**(3) 172–187.

Jagabathula, S. 2016. Assortment optimization under general choice. Tech. rep., NYU, New York, NY.

James, G., D. Witten, T. Hastie, R. Tibshirani. 2014. *An Introduction to Statistical Learning*. Springer, New York, NY.

Kaggle. 2013. Personalize Expedia hotel searches. Last checked: August 5, 2019. URL `https://www.kaggle.com/c/expedia-personalized-sort`.

Li, H., W. T. Huh. 2011. Pricing multiple products with the multinomial logit and nested models: Concavity and implications. *Manufacturing & Service Operations Management* **13**(4) 549–563.

Li, H., S. Webster. 2017. Optimal pricing of correlated product options under the paired combinatorial logit model. *Operations Research* **65**(5) 1215–1230.

Liu, N., Y. Ma, H. Topaloglu. 2019. Assortment optimization under the multinomial logit model with sequential offerings. *INFORMS Journal on Computing* (to appear).

Mendez-Diaz, I., J. J. M. Bront, G. Vulcano, P. Zabala. 2014. A branch-and-cut algorithm for the latent-class logit assortment problem. *Discrete Applied Mathematics* **164**(1) 246–263.

Rusmevichientong, P., Z.-J. M. Shen, D. B. Shmoys. 2009. A PTAS for capacitated sum-of-ratios optimization. *Operations Research Letters* **37**(4) 230–238.

Rusmevichientong, P., D. Shmoys, C. Tong, H. Topaloglu. 2014. Assortment optimization under the multinomial logit model with random choice parameters. *Production and Operations Management* **23**(11) 2023–2039.

Rusmevichientong, Paat, Zuo-Jun Max Shen, David B. Shmoys. 2010. Dynamic assortment optimization with a multinomial logit choice model and capacity constraint. *Operations Research* **58**(6) 1666–1680.

Song, J.-S., Z. Xue. 2007. Demand management and inventory control for substitutable products. Tech. rep., Duke University, Durham, NC.

Sumida, M., G. Gallego, P. Rusmevichientong, H. Topaloglu, J. M. Davis. 2019. Revenue-utility tradeoff in assortment optimization under the multinomial logit model with totally unimodular constraints. Tech. rep., Cornell University, Ithaca, NY.

Talluri, K., G. J. van Ryzin. 2004. Revenue management under a general discrete choice model of consumer behavior. *Management Science* **50**(1) 15–33.

Talluri, K. T., G. J. van Ryzin. 2005. *The theory and practice of revenue management*. Kluwer Academic Publishers, Boston, MA.

Vulcano, G., G. J. van Ryzin, R. Ratliff. 2012. Estimating primary demand for substitutable products from sales transaction data. *Operations Research* **60**(2) 313–334.

Wang, R. 2012. Capacitated assortment and price optimization under the multinomial logit model. *Operations Research Letters* **40**(6) 492–497.

Wang, R., O. Sahin. 2018. The impact of consumer search cost on assortment planning and pricing. *Management Science* **64**(8) 3649–3666.

Williamson, D. P., D. B. Shmoys. 2011. *The design of approximation algorithms*. Cambridge University Press.

Zhang, D., Z. Lu. 2013. Assessing the value of dynamic pricing in network revenue management. *INFORMS Journal on Computing* **25**(1) 102–115.

Zhang, H., P. Rusmevichientong, H. Topaloglu. 2018. Technical note: Multiproduct pricing under the generalized extreme value models with homogeneous price sensitivity parameters. *Operations Research* **66**(6) 1559–1570.

# Appendix: Assortment Optimization and Pricing under the Multinomial Logit Model with Impatient Customers: Sequential Recommendation and Selection

## Appendix A: Maximum of Independent Gumbel Random Variables

Let $X$ and $Y$ be independent Gumbel random variables with location-scale parameters $(\mu, 1)$ and $(\eta, 1)$, respectively. So, the probability density function of $X$ is $f(x) = \exp(-(x-\mu+e^{-(x-\mu)}))$ and the cumulative distribution function of $Y$ is $G(x) = \exp(-e^{-(x-\eta)})$. Thus, we have

$$\begin{aligned}
&\mathbb{P}\{\mathbf{1}(X \geq Y) = 1,\ \max\{X,Y\} \geq u\} \;=\; \mathbb{P}\{X \geq Y,\ \max\{X,Y\} \geq u\} \;=\; \mathbb{P}\{X \geq Y,\ X \geq u\} \\
&\quad = \int_u^\infty \mathbb{P}\{Y \leq x\} \cdot \mathbb{P}\{X \in dx\} \;=\; \int_u^\infty \exp(-e^{-(x-\eta)}) \exp(-(x-\mu+e^{-(x-\mu)}))\, dx \\
&\quad = e^\mu \int_u^\infty \exp(-(x+e^{-x}(e^\mu+e^\eta)))\, dx \;=\; e^\mu \int_u^\infty \exp(-(x+e^{-(x-\log(e^\mu+e^\eta))}))\, dx \\
&\quad = \frac{e^\mu}{e^\mu+e^\eta} \int_u^\infty \exp(-(x-\log(e^\mu+e^\eta)+e^{-(x-\log(e^\mu+e^\eta))}))\, dx \;\stackrel{(a)}{=}\; \mathbb{P}\{X \geq Y\} \cdot \mathbb{P}\{\max\{X,Y\} \geq u\},
\end{aligned}$$

where (a) holds because $\mathbb{P}\{X \geq Y\} = \frac{e^\mu}{e^\mu+e^\eta}$, and $\max\{X,Y\}$ has the Gumbel distribution with location-scale parameters $(\log(e^\mu+e^\eta), 1)$ by the first and second properties in the proof of Theorem 2.1. Thus, $\mathbb{P}\{\mathbf{1}(X \geq Y) = 1,\ \max\{X,Y\} \geq u\} = \mathbb{P}\{\mathbf{1}(X \geq Y) = 1\} \cdot \mathbb{P}\{\max\{X,Y\} \geq u\}$, as desired.

## Appendix B: Proof of Lemma 3.2

In (3), we have one term for each stage. Considering $\Pi(S_1, \ldots, S_{k-1} \cup \{i\}, S_k \setminus \{i\}, \ldots, S_m)$ and $\Pi(S_1, \ldots, S_m)$, the terms for stages other than $k-1$ and $k$ are identical. Thus, fixing $(S_1, \ldots, S_m)$ and letting $\widehat{W}_k = W(S_k)$, $\widehat{\theta}_k = \sum_{\ell=1}^k V(S_\ell)$ and $\widehat{R}_k = R_k(S_1, \ldots, S_m)$, we have

$$\begin{aligned}
&\Pi(S_1, \ldots, S_{k-1} \cup \{i\}, S_k \setminus \{i\}, \ldots, S_m) - \Pi(S_1, \ldots, S_m) \\
&\stackrel{(a)}{=} \frac{\lambda_{k-1}(\widehat{W}_{k-1} + v_i r_i)}{(1+\widehat{\theta}_{k-2})(1+\widehat{\theta}_{k-1}+v_i)} + \frac{\lambda_k(\widehat{W}_k - v_i r_i)}{(1+\widehat{\theta}_{k-1}+v_i)(1+\widehat{\theta}_k)} - \frac{\lambda_{k-1}\widehat{W}_{k-1}}{(1+\widehat{\theta}_{k-2})(1+\widehat{\theta}_{k-1})} - \frac{\lambda_k \widehat{W}_k}{(1+\widehat{\theta}_{k-1})(1+\widehat{\theta}_k)} \\
&= \frac{v_i r_i}{1+\widehat{\theta}_{k-1}+v_i}\left(\frac{\lambda_{k-1}}{1+\widehat{\theta}_{k-2}} - \frac{\lambda_k}{1+\widehat{\theta}_k}\right) + \frac{\lambda_{k-1}\widehat{W}_{k-1}}{1+\widehat{\theta}_{k-2}}\left(\frac{1}{1+\widehat{\theta}_{k-1}+v_i} - \frac{1}{1+\widehat{\theta}_{k-1}}\right) \\
&\qquad + \frac{\lambda_k \widehat{W}_k}{1+\widehat{\theta}_k}\left(\frac{1}{1+\widehat{\theta}_{k-1}+v_i} - \frac{1}{1+\widehat{\theta}_{k-1}}\right),
\end{aligned}$$

where we follow the convention that $\widehat{\theta}_0 = 0$. In the chain of equalities above, $(a)$ uses the fact that $W(S_{k-1} \cup \{i\}) = W(S_{k-1}) + v_i r_i$ and $W(S_k \setminus \{i\}) = W(S_k) - v_i r_i$, along with $\sum_{\ell=1}^{k-2} V(S_\ell) + V(S_{k-1} \cup \{i\}) = \sum_{\ell=1}^{k-1} V(S_\ell) + v_i$ and $\sum_{\ell=1}^{k-2} V(S_\ell) + V(S_{k-1} \cup \{i\}) + V(S_k \setminus \{i\}) = \sum_{\ell=1}^{k} V(S_\ell)$.

Arranging the terms on the right side of the chain of equalities above, the right side of the chain of equalities above is equivalent to

$$\frac{v_i\, r_i}{1+\widehat{\theta}_{k-1}+v_i}\left(\frac{\lambda_{k-1}}{1+\widehat{\theta}_{k-2}}-\frac{\lambda_k}{1+\widehat{\theta}_k}\right)-\left(\frac{\lambda_{k-1}\,\widehat{W}_{k-1}}{1+\widehat{\theta}_{k-2}}+\frac{\lambda_k\,\widehat{W}_k}{1+\widehat{\theta}_k}\right)\left(\frac{v_i}{(1+\widehat{\theta}_{k-1}+v_i)\,(1+\widehat{\theta}_{k-1})}\right)$$

$$\stackrel{(b)}{=}\ \frac{v_i\, r_i}{1+\widehat{\theta}_{k-1}+v_i}\left(\frac{\lambda_{k-1}}{1+\widehat{\theta}_{k-2}}-\frac{\lambda_k}{1+\widehat{\theta}_k}\right)-\frac{v_i}{1+\widehat{\theta}_{k-1}+v_i}\,(\widehat{R}_{k-1}+\widehat{R}_k)$$

$$=\ \frac{\frac{\lambda_{k-1}}{1+\widehat{\theta}_{k-2}}-\frac{\lambda_k}{1+\widehat{\theta}_k}}{1+\widehat{\theta}_{k-1}+v_i}\,v_i\left(r_i-\frac{\widehat{R}_{k-1}+\widehat{R}_k}{\frac{\lambda_{k-1}}{1+\widehat{\theta}_{k-2}}-\frac{\lambda_k}{1+\widehat{\theta}_k}}\right)\ =\ \frac{\frac{\lambda_{k-1}}{1+\widehat{\theta}_{k-2}}-\frac{\lambda_k}{1+\widehat{\theta}_k}}{1+\widehat{\theta}_{k-1}+v_i}\,v_i\,(r_i-t_k(S_1,\ldots,S_m)),$$

where $(b)$ uses the definition of $R_k(S_1,\ldots,S_m)$. The two chains of equalities above show that the first identity in the lemma holds. The proofs for the second and third identities are similar. ■

## Appendix C: Nonnegative Optimal Prices

In the next lemma, we show that the prices are nonnegative in any optimal solution to the Pricing problem, which allows us to use first-order conditions to characterize the optimal prices.

**Lemma C.1** *Letting $\boldsymbol{p}^*$ be an optimal solution to the* Pricing *problem, the prices in the optimal solution satisfies $p_i^*\geq 0$ for all $i\in\mathcal{N}$.*

*Proof:* Using $\boldsymbol{p}^*$ to denote an optimal solution to the Pricing problem, let $\mathcal{N}^+$ and $\mathcal{N}^-$ be such that $p_i^*\geq 0$ for all $i\in\mathcal{N}^+$ and $p_i^*<0$ for all $i\in\mathcal{N}^-$. To get a contradiction, assume that $\mathcal{N}^-\neq\varnothing$. Let $\widehat{\boldsymbol{p}}$ be defined as

$$\widehat{p}_i=p_i^*\quad\forall\, i\in\mathcal{N}^+\quad\text{and}\quad\widehat{p}_i=0\quad\forall\, i\in\mathcal{N}^-.$$

We claim that the choice probabilities corresponding to the prices $\widehat{\boldsymbol{p}}$ satisfy $\phi_i^k(\widehat{\boldsymbol{p}})\geq\phi_i^k(\boldsymbol{p}^*)$ for all $i\in S_k\cap\mathcal{N}^+$, $k\in\mathcal{M}$. In particular, by the definition of $\widehat{\boldsymbol{p}}$, we have $\widehat{p}_i\geq p_i^*$ for all $i\in\mathcal{N}$, so $e^{\alpha_i-\beta\,\widehat{p}_i}\leq e^{\alpha_i-\beta\,p_i^*}$ for all $i\in\mathcal{N}$. Thus, we get $V_k(\widehat{\boldsymbol{p}})\leq V_k(\boldsymbol{p}^*)$ for all $k\in\mathcal{M}$. In this case, since $\widehat{p}_i=p_i^*$ for all $i\in\mathcal{N}^+$, we have $\phi_i^k(\widehat{\boldsymbol{p}})=\lambda_k\frac{e^{\alpha_i-\beta\,\widehat{p}_i}}{((1+\sum_{\ell=1}^{k-1}V_\ell(\widehat{\boldsymbol{p}}))\,(1+\sum_{\ell=1}^{k}V_\ell(\widehat{\boldsymbol{p}})))}\geq\lambda_k\frac{e^{\alpha_i-\beta\,p_i^*}}{((1+\sum_{\ell=1}^{k-1}V_\ell(\boldsymbol{p}^*))\,(1+\sum_{\ell=1}^{k}V_\ell(\boldsymbol{p}^*)))}=\phi_i^k(\boldsymbol{p}^*)$ for all $i\in S_k\cap\mathcal{N}^+$, $k\in\mathcal{M}$. Thus, the claim holds. Letting $S_k^+=S_k\cap\mathcal{N}^+$ and $S_k^-=S_k\cap\mathcal{N}^-$ for notational brevity, we have $\widehat{p}_i=p_i^*$ for all $i\in S_k^+$, as well as $\widehat{p}_i=0>p_i^*$ for all $i\in S_k^-$. Furthermore, since $\mathcal{N}^-\neq\varnothing$, we have $S_k^-\neq\varnothing$ for some $k\in\mathcal{M}$. In this case, we obtain

$$\Pi(\boldsymbol{p}^*)=\sum_{k\in\mathcal{M}}\sum_{i\in S_k^+}p_i^*\,\phi_i^k(\boldsymbol{p}^*)+\sum_{k\in\mathcal{M}}\sum_{i\in S_k^-}p_i^*\,\phi_i^k(\boldsymbol{p}^*)\stackrel{(a)}{<}\sum_{k\in\mathcal{M}}\sum_{i\in S_k^+}p_i^*\,\phi_i^k(\boldsymbol{p}^*)\stackrel{(b)}{\leq}\sum_{k\in\mathcal{M}}\sum_{i\in S_k^+}\widehat{p}_i\,\phi_i^k(\widehat{\boldsymbol{p}})$$
$$=\sum_{k\in\mathcal{M}}\sum_{i\in S_k^+}\widehat{\boldsymbol{p}}_i\,\phi_i^k(\widehat{\boldsymbol{p}})+\sum_{k\in\mathcal{M}}\sum_{i\in S_k^-}\widehat{\boldsymbol{p}}_i\,\phi_i^k(\widehat{\boldsymbol{p}})=\Pi(\widehat{\boldsymbol{p}}),$$

where $(a)$ holds since $p_i^*<0$ for all $i\in S_k^-$ and $S_k^-\neq\varnothing$ for some $k\in\mathcal{M}$, whereas $(b)$ holds since $p_i^*=\widehat{p}_i$ and $\phi_i^k(\boldsymbol{p}^*)\leq\phi_i^k(\widehat{\boldsymbol{p}})$ for all $i\in S_k\cap\mathcal{N}^+$, $k\in\mathcal{M}$. The chain of inequalities above contradicts the fact that $\boldsymbol{p}^*$ is an optimal solution to the Pricing problem. ■

## Appendix D: Additive Performance Guarantee for Optimal Prices under Fixed Assortments

In this section, we give a dynamic programming approach to obtain a solution to the PRICING problem with an additive performance guarantee. Letting $\pi^*$ be the optimal objective value of the PRICING problem, for any $\theta > 0$, our approach comes up with a solution that provides an expected revenue of at least $\pi^* - \theta$ and the number of operations required to obtain this solution is polynomial in $1/\theta$. At the end of this section, we explain that we can easily use a lower bound on the optimal expected revenue to numerically evaluate the multiplicative performance guarantee of the solution that has an additive performance guarantee. To construct a solution with an additive performance guarantee, fixing an integer $K > 0$, we construct the grid points $\mathsf{Grid} = \{\ell/K : \ell = 1, \ldots, K\}$ over the interval $[0, 1]$. Noting the expected revenue expression in (5), we use the no-purchase probabilities over different numbers of stages as the decision variables in the PRICING problem. We focus on only the values of the no-purchase probabilities that take values in $\mathsf{Grid}$. Let $\Theta_k(q_{k-1})$ denote the maximum expected revenue that can be obtained from stages $k, k+1, \ldots, m$, given that the no-purchase probability over the first $k-1$ stages is $q_{k-1}$. In this case, letting $\Theta_{m+1}(\cdot) = 0$ and recalling that $q_0 = 1$, by (5), for all $k \in \mathcal{M}$ and $q_{k-1} \in \mathsf{Grid}$, we have the recursion

$$\Theta_k(q_{k-1}) \;=\; \max_{\substack{q_k \,\in\, \mathsf{Grid}\,: \\ q_{k-1} \geq q_k}} \left\{ \frac{\lambda_k}{\beta}\,(q_{k-1} - q_k) \left\{ \log\left( \sum_{i \in S_k} e^{\alpha_i} \right) - \log\left( \frac{1}{q_k} - \frac{1}{q_{k-1}} \right) \right\} + \Theta_{k+1}(q_k) \right\}. \qquad (14)$$

Thus, $\Theta_1(q_0) = \Theta_1(1)$ corresponds to the largest expected revenue provided by no-purchase probabilities of the form $\boldsymbol{q} = (q_1, \ldots, q_m)$ with $q_k \in \mathsf{Grid}$ and $q_{k-1} \geq q_k$ for all $k \in \mathcal{M}$.

In the next theorem, we show that we can use the dynamic program above to come up with a solution to the PRICING problem with an additive performance guarantee.

**Theorem D.1** *Letting $\pi^*$ be the optimal objective value of the PRICING problem and $\{\Theta_k(\cdot) : k \in \mathcal{M}\}$ be obtained through (14) with $\mathsf{Grid} = \{\ell/K : \ell = 1, \ldots, K\}$ and $K \geq 3$, we have*

$$\Theta_1(1) \;\geq\; \pi^* - \frac{1}{\beta K} \left( \sum_{k \in \mathcal{M}} \left| \log \sum_{i \in S_k} e^{\alpha_i} \right| + m + 3m \log K \right).$$

*Proof:* Let $\boldsymbol{q}^*$ be an optimal solution to problem (6). By Theorem 4.2, we have $\pi^* = \widehat{\Pi}(\boldsymbol{q}^*)$, where $\widehat{\Pi}(\boldsymbol{q})$ is as in (5). As discussed immediately before the theorem, $\Theta_1(1)$ corresponds to the largest expected revenue provided by no-purchase probabilities of the form $\boldsymbol{q} = (q_1, \ldots, q_m)$ with $q_k \in \mathsf{Grid}$ and $q_{k-1} \geq q_k$ for all $k \in \mathcal{M}$. Thus, we have $\Theta_1(1) \geq \widehat{\Pi}(\widehat{\boldsymbol{q}})$ for any $\widehat{\boldsymbol{q}}$ that satisfies $\widehat{q}_k \in \mathsf{Grid}$ and $\widehat{q}_{k-1} \geq \widehat{q}_k$ for all $k \in \mathcal{M}$. We show that there exists some $\widehat{\boldsymbol{q}} = (\widehat{q}_1, \ldots, \widehat{q}_m)$ that satisfies $\widehat{q}_k \in \mathsf{Grid}$ and $\widehat{q}_{k-1} \geq \widehat{q}_k$ for all $k \in \mathcal{M}$ such that $\widehat{\Pi}(\widehat{\boldsymbol{q}}) \geq \widehat{\Pi}(\boldsymbol{q}^*) - \frac{1}{\beta K}\big( \sum_{k \in \mathcal{M}} |\log \sum_{i \in S_k} e^{\alpha_i}| + m + 3m \log K \big)$, in which case, the desired result follows by noting that $\Theta_1(1) \geq \widehat{\Pi}(\widehat{\boldsymbol{q}})$ and $\pi^* = \widehat{\Pi}(\boldsymbol{q}^*)$. Define $\widehat{\boldsymbol{q}}$

as $\widehat{\boldsymbol{q}}_k = \min\{q_k \in \mathsf{Grid} : q_k \geq q_k^*\}$ for all $k \in \mathcal{M}$, so $\widehat{q}_k$ is obtained by rounding $q_k^*$ up to the nearest point in $\mathsf{Grid}$. In this case, since $1 \geq q_1^* \geq \ldots \geq q_m^* \geq 0$, we have $1 \geq \widehat{q}_1 \geq \ldots \geq \widehat{q}_m \geq 0$. Furthermore, since $\widehat{q}_k$ is obtained by rounding $q_k^*$ up to the nearest point in $\mathsf{Grid}$, we have $1 \geq \widehat{q}_k \geq q_k^*$ and $\widehat{q}_k \geq 1/K$ for all $k \in \mathcal{M}$. Lastly, since the two successive points in $\mathsf{Grid}$ are separated by $1/K$, we have $0 \leq \widehat{q}_k - q_k^* \leq 1/K$ and $-1/K \leq q_{k-1}^* - \widehat{q}_{k-1} \leq 0$, in which case, adding the two yields $-1/K \leq (q_{k-1}^* - q_k^*) - (\widehat{q}_{k-1} - \widehat{q}_k) \leq 1/K$. For notational brevity, let $\Delta_k^* = q_{k-1}^* - q_k^*$ and $\widehat{\Delta}_k = \widehat{q}_{k-1} - \widehat{q}_k$, so we write the last chain of inequalities as $-1/K \leq \Delta_k^* - \widehat{\Delta}_k \leq 1/K$. Noting that $q_{k-1}^* \geq q_k^*$ and $\widehat{q}_{k-1} \geq \widehat{q}_k$, we have $\Delta_k^* \geq 0$ and $\widehat{\Delta}_k \geq 0$. By the definition of $\widehat{\Pi}(\boldsymbol{q})$ in (5), we have

$$\begin{aligned}\widehat{\Pi}(\boldsymbol{q}^*) - \widehat{\Pi}(\widehat{\boldsymbol{q}}) \;=\; & \sum_{k\in\mathcal{M}} \frac{\lambda_k}{\beta}(\Delta_k^* - \widehat{\Delta}_k)\log\left(\sum_{i\in S_k} e^{\alpha_i}\right) + \sum_{k\in\mathcal{M}} \frac{\lambda_k}{\beta}\Big\{\Delta_k^* \log q_{k-1}^* - \widehat{\Delta}_k \log \widehat{q}_{k-1}\Big\} \\ & + \sum_{k\in\mathcal{M}} \frac{\lambda_k}{\beta}\Big\{\Delta_k^* \log q_k^* - \widehat{\Delta}_k \log \widehat{q}_k\Big\} - \sum_{k\in\mathcal{M}} \frac{\lambda_k}{\beta}\Big\{\Delta_k^* \log \Delta_k^* - \widehat{\Delta}_k \log \widehat{\Delta}_k\Big\}. \qquad (15)\end{aligned}$$

We bound each one of the four sums above separately. To bound the first sum, noting that $|\Delta_k^* - \widehat{\Delta}_k| \leq 1/K$ by the discussion at the beginning of the proof, we obtain

$$(\Delta_k^* - \widehat{\Delta}_k)\log\left(\sum_{i\in S_k} e^{\alpha_i}\right) \;\leq\; |\Delta_k^* - \widehat{\Delta}_k| \left|\log \sum_{i\in S_k} e^{\alpha_i}\right| \;\leq\; \frac{1}{K}\left|\log\sum_{i\in S_k} e^{\alpha_i}\right|. \qquad (16)$$

To bound the second sum, note that $\Delta_k^* \geq 0$ and $q_{k-1}^* \leq \widehat{q}_{k-1}$, so $\Delta_k^* \log q_{k-1}^* \leq \Delta_k^* \log \widehat{q}_{k-1}$. Also, $\Delta_k^* - \widehat{\Delta}_k \geq -1/K$ and $\log \widehat{q}_{k-1} \leq 0$. Lastly, since $\widehat{q}^k \geq 1/K$, $-\log \widehat{q}_k \leq \log K$. Thus, we get

$$\Delta_k^* \log q_{k-1}^* - \widehat{\Delta}_k \log \widehat{q}_{k-1} \;\leq\; (\Delta_k^* - \widehat{\Delta}_k)\log \widehat{q}_{k-1} \;\leq\; -\frac{1}{K}\log \widehat{q}_{k-1} \;\leq\; \frac{1}{K}\log K. \qquad (17)$$

Similarly, we have $\Delta_k^* \log q_k^* - \widehat{\Delta}_k \log \widehat{q}_k \leq \frac{1}{K}\log K$, bounding the third sum. To bound the fourth sum, consider the case $\widehat{\Delta}_k \geq 1/K$. Since $x\log x$ convex in $x$, the subgradient inequality yields

$$\begin{aligned}\Delta_k^* \log \Delta_k^* - \widehat{\Delta}_k \log \widehat{\Delta}_k \;& \overset{(a)}{\geq}\; (1 + \log \widehat{\Delta}_k)(\Delta_k^* - \widehat{\Delta}_k) \\ & =\; \Delta_k^* - \widehat{\Delta}_k + \log \widehat{\Delta}_k (\Delta_k^* - \widehat{\Delta}_k) \;\overset{(b)}{\geq}\; -\frac{1}{K} + \frac{1}{K}\log \widehat{\Delta}_k \;\overset{(c)}{\geq}\; -\frac{1}{K} - \frac{1}{K}\log K, \qquad (18)\end{aligned}$$

where $(a)$ holds since the derivative of $x\log x$ is $1+\log x$, $(b)$ holds since $\widehat{\Delta}_k \leq 1$, so $\log \Delta_k \leq 0$ and $-1/K \leq \Delta_k^* - \widehat{\Delta}_k \leq 1/K$, and $(c)$ holds since $\widehat{\Delta}_k \geq 1/K$, so $\log \widehat{\Delta}_k \geq -\log K$.

Consider the case $\widehat{\Delta}_k < 1/K$. Since $\widehat{q}_{k-1}$ and $\widehat{q}_k$ are, respectively, obtained by rounding $q_{k-1}^*$ and $q_k^*$ up to the nearest point in $\mathsf{Grid}$, if $\widehat{\Delta}_k = \widehat{q}_{k-1} - \widehat{q}_k < 1/K$, then we must have $\widehat{\Delta}_k = \widehat{q}_{k-1} - \widehat{q}_k = 0$, in which case, we must have $q_{k-1}^* - q_k^* \leq 1/K$. Furthermore, $x\log x$ is decreasing in $x$ for $x \in (0, e^{-1})$, so since $K \geq 3$, we have $e^{-1} > 1/K \geq q_{k-1}^* - q_k^* = \Delta_k^*$, which implies that $-\frac{1}{K}\log K \leq \Delta_k^* \log \Delta_k^*$. Thus, noting that $\lim_{x\to 0} x\log x = 0$, we get $\Delta_k^* \log \Delta_k^* - \widehat{\Delta}_k \log \widehat{\Delta}_k = \Delta_k^* \log \Delta_k^* \geq -\frac{1}{K}\log K$, indicating that the inequality in (18) holds under the case $\widehat{\Delta}_k < 1/K$ as well. Adding up the inequalities in (16), (17) and (18), recalling that we also have $\Delta_k^* \log q_k^* - \widehat{\Delta}_k \log \widehat{q}_k \leq \frac{1}{K} \log K$ for the third sum and noting that $\lambda_k \leq 1$, by (15), we get $\widehat{\Pi}(\boldsymbol{q}^*) - \widehat{\Pi}(\widehat{\boldsymbol{q}}) \leq \frac{1}{\beta K} \sum_{k \in \mathcal{M}} |\log \sum_{i \in S_k} e^{\alpha_i}| + \frac{2m}{\beta K} \log K + \frac{m}{\beta K} + \frac{m}{\beta K} \log K = \frac{1}{\beta K} \big( \sum_{k \in \mathcal{M}} |\log \sum_{i \in S_k} e^{\alpha_i}| + m + 3m \log K \big)$. ■

We can follow the optimal state-action trajectory in the dynamic program in (14) to obtain the no-purchases probabilities $\widehat{\boldsymbol{q}} = (\widehat{q}_1, \ldots, \widehat{q}_m)$ that provide the additive performance guarantee of $\frac{1}{\beta K} \big( \sum_{k \in \mathcal{M}} |\log \sum_{i \in S_k} e^{\alpha_i}| + m + 3m \log K \big)$. In particular, after computing $\{\Theta_k(\cdot) : k \in \mathcal{M}\}$ through the dynamic program (14), we set $\widehat{q}_0 = 1$. For each $k \in \mathcal{M}$, we compute $\widehat{q}_k$ as an optimal solution to the problem on the right side of (14) when we solve this problem with $q_{k-1} = \widehat{q}_{k-1}$. Once we have these no-purchase probabilities, noting the expression right before (5), we can compute the corresponding stage-specific prices $\widehat{\boldsymbol{\rho}} = (\widehat{\rho}_1, \ldots, \widehat{\rho}_m)$ as $\widehat{\rho}_k = \frac{1}{\beta} \big\{ \log \big( \sum_{i \in S_k} e^{\alpha_i} \big) - \log \big( \frac{1}{\widehat{q}_k} - \frac{1}{\widehat{q}_{k-1}} \big) \big\}$ for all $k \in \mathcal{M}$. These stage-specific prices yield the same additive performance guarantee.

The number of operations to obtain an additive performance guarantee of $\theta > 0$ is polynomial in $1/\theta$. By the theorem above, to get an additive performance guarantee $\theta > 0$, we need to choose $K$ such that $\frac{1}{\beta K} \big( \sum_{k \in \mathcal{M}} |\log \sum_{i \in S_k} e^{\alpha_i}| + m + 3m \log K \big) \leq \theta$, but since $1/K \leq 1/\sqrt{K}$ and $\log K / K \leq 1/\sqrt{K}$ for $K \geq 3$, it is enough to choose $K$ such that $\frac{1}{\beta \sqrt{K}} \big( \sum_{k \in \mathcal{M}} |\log \sum_{i \in S_k} e^{\alpha_i}| + m + 3m \big) \leq \theta$, so we can set $K = \frac{1}{(\theta\beta)^2} \big( \sum_{k \in \mathcal{M}} |\log \sum_{i \in S_k} e^{\alpha_i}| + m + 3m \big)^2$. The dynamic program in (14) has $K$ possible states, $K$ possible actions, and $m$ decision epochs. Thus, we can solve this dynamic program in $m K^2$ operations, so noting the choice of $K$, we can obtain a solution that provides an additive performance guarantee of $\theta$ in $\frac{m}{(\theta\beta)^4} \big( \sum_{k \in \mathcal{M}} |\log \sum_{i \in S_k} e^{\alpha_i}| + m + 3m \big)^4$ operations. Our analysis for the number of operations is rather loose. For any $\epsilon > 0$, we have $1/K \leq 1/K^{1-\epsilon}$ and $\log K / K \leq 1/K^{1-\epsilon}$ for large enough $K$. In this case, following the same line of reasoning in this paragraph, we can obtain a solution that provides an additive performance guarantee of $\theta$ in $\frac{m}{(\theta\beta)^{2/(1-\epsilon)}} \big( \sum_{k \in \mathcal{M}} |\log \sum_{i \in S_k} e^{\alpha_i}| + m + 3m \big)^{2/(1-\epsilon)}$ operations, so the number of operations scale a bit faster than quadratically in $1/\theta$ as long as the number of points $K$ in Grid is large, which is more aligned with our experience with the dynamic program in (14), as we report shortly.

If we have a lower bound on the optimal objective value of the PRICING problem, then we can always use the theorem above to numerically obtain a multiplicative performance guarantee. In particular, for any $\delta \in (0, 1)$, letting $\widehat{\pi}$ be a lower bound on the optimal objective value of the PRICING problem, we can use Theorem D.1 to obtain a solution that provides an additive performance guarantee of $\delta \widehat{\pi}$. In other words, using $\pi^*$ to denote the optimal objective value of the PRICING problem, this solution provides an expected revenue of at least $\pi^* - \delta \widehat{\pi}$. Noting that $\pi^* - \delta \widehat{\pi} \geq (1 - \delta) \pi^*$, the solution that provides an additive performance guarantee of $\delta \widehat{\pi}$ also provides an expected revenue of at least $(1 - \delta) \pi^*$, corresponding to a solution with a multiplicative

| $(m,C)=(6,3)$ | | |
|---|---|---|
| $a$ | Avg. Gap | CPU Secs. |
| $+\infty$ | $7.2\cdot 10^{-5}\%$ | 34 |
| 0.5 | $9.6\cdot 10^{-5}\%$ | 30 |
| 0.0 | $1.2\cdot 10^{-4}\%$ | 31 |
| $-0.1$ | $1.4\cdot 10^{-4}\%$ | 33 |

| $(m,C)=(6,5)$ | | |
|---|---|---|
| $a$ | Avg. Gap | CPU Secs. |
| $+\infty$ | $1.5\cdot 10^{-4}\%$ | 43 |
| 0.5 | $1.6\cdot 10^{-4}\%$ | 31 |
| 0.0 | $1.8\cdot 10^{-4}\%$ | 30 |
| $-0.1$ | $1.7\cdot 10^{-4}\%$ | 32 |

| $(m,C)=(10,3)$ | | |
|---|---|---|
| $a$ | Avg. Gap | CPU Secs. |
| $+\infty$ | $3.9\cdot 10^{-5}\%$ | 423 |
| 0.5 | $5.3\cdot 10^{-5}\%$ | 358 |
| 0.0 | $5.0\cdot 10^{-5}\%$ | 278 |
| $-0.1$ | $8.0\cdot 10^{-5}\%$ | 260 |

| $(m,C)=(10,5)$ | | |
|---|---|---|
| $a$ | Avg. Gap | CPU Secs. |
| $+\infty$ | $7.5\cdot 10^{-5}\%$ | 520 |
| 0.5 | $6.4\cdot 10^{-5}\%$ | 426 |
| 0.0 | $1.0\cdot 10^{-4}\%$ | 340 |
| $-0.1$ | $9.2\cdot 10^{-5}\%$ | 336 |

**Table EC.1** Optimality gaps of the prices obtained by using the dynamic program in (14).

performance guarantee of $1-\delta$. To obtain a lower bound on the optimal objective value of the PRICING problem, we can, for example, charge the same price for all products in all stages, in which case, we have a single decision variable in the PRICING problem. We can carry out a numerical search to find the best single price to charge in all stages.

**Computational Experiments:** We give a small set of computational experiments to understand the quality of the additive performance guarantee given in Theorem D.1. We randomly generate a number of test problems. For each problem instance, we use the approach in Section 4.1 to compute the optimal prices, as well as the dynamic program in (14) to compute prices with an additive performance guarantee. To choose the number of points $K$ in Grid, we compute a lower bound on the optimal objective value of the PRICING problem by charging the same price for all products in all stages and finding the best single price to charge through numerical search. Letting $\widehat{\pi}$ be this lower bound, we choose the value of $K$ such that we obtain an additive performance guarantee of $\frac{1}{2}\widehat{\pi}$. By the discussion in the previous paragraph, this approach yields a multiplicative performance guarantee of 50%. The approach to generate our test problems closely follows the one in Section 6.2. We briefly describe our approach and refer to Section 6.2 for details.

The number of products is $n=20$ and the price sensitivity is $\beta=1$. We come up with the parameters $\{\alpha_i : i\in\mathcal{N}\}$ as follows. We have $C$ product clusters. We randomly assign each product to a cluster. If products $i$ and $j$ are in the same cluster, then the values of $\alpha_i$ and $\alpha_j$ are close. Specifically, cluster $c$ has the centroid $\gamma_c$. We set the centroid of cluster $c$ as $\gamma_c=c-0.5$ for all $c=1,\dots,C$. If product $i$ belongs to cluster $c$, then we generate $\kappa_i$ from the normal distribution with mean $\gamma_c$ and standard deviation one. We set $\alpha_i=\kappa_i-\Delta$, where $\Delta=\log\sum_{i\in\mathcal{N}}e^{\kappa_i}-\log 9$. Thus, if all products were offered in the first stage at zero price, then a customer would leave without a purchase with probability 0.1. We randomly assign each product to one of the assortments $(S_1,\dots,S_m)$. Letting the random variable $Y$ be the patience level of a customer, the probability mass function of $Y$ is $\mathbb{P}\{Y=k\}=\frac{e^{a\cdot k}}{\sum_{\ell\in\mathcal{M}}e^{a\cdot\ell}}$, where $a$ is a parameter that we vary.

Varying $m\in\{6,10\}$, $C\in\{3,5\}$, and $a\in\{+\infty,0.5,0.0,-0.1\}$, we get 16 parameter configurations. In each parameter configuration, we generate 25 problem instances. For each problem instance, we compute the optimal expected revenue by solving the convex program in (6) through the `fmincon` routine in Matlab. In Table EC.1, we show the average percent gap between the optimal expected revenue and the expected revenue from the prices obtained through the dynamic program in (14), averaged over the 25 instances in a parameter configuration. The second column in the table shows the runtime to solve the dynamic program in (14). Note that we choose the value of $K$ for a performance guarantee of 50%, but the optimality gap of the prices that we obtain is less than $1.8 \cdot 10^{-4}$%. Although we report average optimality gaps, each optimality gap deviates from the average by no more than $0.2 \cdot 10^{-4}$%. For our problem instances, the runtime to solve the dynamic program in (14) ranges between half a minute to nine minutes. For comparison, although we do not report in Table EC.1, the runtime to solve problem (6) through the `fmincon` routine in Matlab takes a few seconds. Thus, solving (14) does not require convex optimization tools, but solving problem (6) through convex optimization software is faster.

## Appendix E: First-Order Conditions for Optimal Prices

The proof of Theorem 4.3 uses the following lemma, which gives a characterization of the optimal stage-specific prices for the Pricing problem by using first order conditions.

**Lemma E.1** *Letting $\boldsymbol{\rho}^* = (\rho_1^*, \ldots, \rho_m^*)$ be the optimal stage-specific prices in the* Pricing *problem and $q_k^* = q_k(\boldsymbol{\rho}^*)$ for all $k \in \mathcal{M}$ with $q_0^* = 1$, we have*

$$\frac{1}{\beta} - \frac{q_\ell^*}{q_{\ell-1}^*}\,\rho_\ell^* + \frac{1}{\lambda_\ell\, q_\ell^*\, q_{\ell-1}^*} \sum_{k=\ell+1}^{m} \rho_k^*\, \lambda_k \Big\{ (q_{k-1}^*)^2 - (q_k^*)^2 \Big\} = 0.$$

*Proof:* Since $\widehat{V}_k(\boldsymbol{\rho}) = e^{-\beta\,\rho_k} \sum_{i \in S_k} e^{\alpha_i}$, we have $\frac{\partial \widehat{V}_k(\boldsymbol{\rho})}{\partial \rho_k} = -\beta\, \widehat{V}_k(\boldsymbol{\rho})$ and $\frac{\partial \widehat{V}_k(\boldsymbol{\rho})}{\partial \rho_\ell} = 0$ for all $\ell \neq k$. In this case, noting that $q_k(\boldsymbol{\rho}) = \frac{1}{1 + \sum_{\ell=1}^k \widehat{V}_\ell(\boldsymbol{\rho})}$, for $k \geq \ell$, we get $\frac{\partial q_k(\boldsymbol{\rho})}{\partial \rho_\ell} = \beta\, \widehat{V}_\ell(\boldsymbol{\rho})\, q_k(\boldsymbol{\rho})^2$. Also, we have

$$q_{k-1}(\boldsymbol{\rho}) - q_k(\boldsymbol{\rho}) = \frac{1}{1 + \sum_{\ell=1}^{k-1} \widehat{V}_\ell(\boldsymbol{\rho})} - \frac{1}{1 + \sum_{\ell=1}^{k} \widehat{V}_\ell(\boldsymbol{\rho})} = \frac{\widehat{V}_k(\boldsymbol{\rho})}{(1 + \sum_{\ell=1}^{k-1} \widehat{V}_\ell(\boldsymbol{\rho}))\,(1 + \sum_{\ell=1}^{k} \widehat{V}_\ell(\boldsymbol{\rho}))},$$

so $q_{k-1}^* - q_k^* = \widehat{V}_k(\boldsymbol{\rho}^*)\, q_{k-1}^*\, q_k^*$. Lastly, the optimal prices are finite, since decreasing all infinite prices to the largest finite price charged in any stage improves the expected revenue.

By (5), $\Pi(\boldsymbol{\rho}) = \sum_{k \in \mathcal{M}} \lambda_k\, \rho_k\, (q_{k-1}(\boldsymbol{\rho}) - q_k(\boldsymbol{\rho}))$ is the expected revenue as a function of stage-specific prices. Note that $q_k(\boldsymbol{\rho})$ depends on $\rho_\ell$ only if $k \geq \ell$. Thus, differentiating $\Pi(\boldsymbol{\rho})$, we get

$$\frac{\partial \Pi(\boldsymbol{\rho})}{\partial \rho_\ell} \;=\; \lambda_\ell\, (q_{\ell-1}(\boldsymbol{\rho}) - q_\ell(\boldsymbol{\rho})) - \lambda_\ell\, \rho_\ell\, \beta\, \widehat{V}_\ell(\boldsymbol{\rho})\, q_\ell(\boldsymbol{\rho})^2 + \sum_{k=\ell+1}^{m} \lambda_k\, \rho_k\, \beta\, \widehat{V}_\ell(\boldsymbol{\rho}) \Big\{ q_{k-1}(\boldsymbol{\rho})^2 - q_k(\boldsymbol{\rho})^2 \Big\},$$

where we use the fact that $\frac{\partial q_k(\boldsymbol{\rho})}{\partial \rho_\ell} = \beta\, \widehat{V}_\ell(\boldsymbol{\rho})\, q_k(\boldsymbol{\rho})^2$ for $k \geq \ell$, but $q_k(\boldsymbol{\rho})$ does not depend on $\rho_\ell$ for $k < \ell$, so we have $\frac{\partial q_k(\boldsymbol{\rho})}{\partial \rho_\ell} = 0$ for $k < \ell$. The optimal stage-specific prices $\boldsymbol{\rho}^*$ satisfies the first order condition $\frac{\partial \Pi(\boldsymbol{\rho})}{\partial \rho_\ell}\Big|_{\boldsymbol{\rho}=\boldsymbol{\rho}^*} = 0$. Therefore, using the equality above, along with the fact that $q^*_{\ell-1} - q^*_\ell = \widehat{V}_\ell(\boldsymbol{\rho}^*)\, q^*_{\ell-1}\, q^*_\ell$, we get

$$\frac{\partial \Pi(\boldsymbol{\rho})}{\partial \rho_\ell}\bigg|_{\boldsymbol{\rho}=\boldsymbol{\rho}^*} = \lambda_\ell\, \beta\, \widehat{V}_\ell(\boldsymbol{\rho}^*)\, q^*_{\ell-1}\, q^*_\ell \left\{ \frac{1}{\beta} - \frac{q^*_\ell}{q^*_{\ell-1}}\, \rho^*_\ell + \frac{1}{\lambda_\ell\, q^*_\ell\, q^*_{\ell-1}} \sum_{k=\ell+1}^m \rho^*_k\, \lambda_k \Big\{ (q^*_{k-1})^2 - (q^*_k)^2 \Big\} \right\} = 0.$$

Since the optimal prices are finite, $\widehat{V}_\ell(\boldsymbol{\rho}^*) \neq 0$, along with $q^*_{\ell-1} = q_{\ell-1}(\boldsymbol{\rho}^*) \neq 0$ and $q^*_\ell = q_\ell(\boldsymbol{\rho}^*) \neq 0$, in which case $\boldsymbol{\rho}^*$ satisfies the equality in the lemma. ■

## Appendix F: Proof of Theorem 4.4 and Tightness of the Performance Guarantee of 87.8%

The proof uses a chain of upper bounds. Setting $\lambda_k = 1$ for all $k \in \mathcal{M}$ enlarges the objective function of the Pricing-Assortment problem. By (5), we can express the expected revenue as a function of the no-purchase probabilities. So, setting $\lambda_k = 1$ for all $k \in \mathcal{M}$ in (5), as a function of no-purchase probabilities $\boldsymbol{q}$ and assortments $(S_1, \ldots, S_m)$, we can upper bound the expected revenue by

$$\widehat{\Pi}(\boldsymbol{q}, S_1, \ldots, S_m) = \frac{1}{\beta} \sum_{k \in \mathcal{M}} (q_{k-1} - q_k) \left\{ \log\left( \sum_{i \in S_k} e^{\alpha_i} \right) - \log\left( \frac{1}{q_k} - \frac{1}{q_{k-1}} \right) \right\}. \tag{19}$$

So, we can upper bound the optimal objective value of the Pricing-Assortment problem by maximizing $\widehat{\Pi}(\boldsymbol{q}, S_1, \ldots, S_m)$ over all $(\boldsymbol{q}, S_1, \ldots, S_m) \in \mathbb{R}^m_+ \times \mathcal{F}$ such that $q_{k-1} \geq q_k$ for all $k \in \mathcal{M}$.

Throughout this section, we set $\beta = 1$ for notational brevity, which simply scales the expected revenue by $\beta$. Also, recall that $q_0 = 1$. Letting $T = \sum_{i \in \mathcal{N}} e^{\alpha_i}$, we define $R^{(\ell)}(q_1, \ldots, q_\ell)$ as

$$R^{(\ell)}(q_1, \ldots, q_\ell) = \sum_{k=1}^{\ell} (q_{k-1} - q_k) \log(q_{k-1}\, q_k) + (1 - q_\ell) \log\left( \frac{T}{1 - q_\ell} \right). \tag{20}$$

We have the superscript $(\ell)$ in $R^{(\ell)}(q_1, \ldots, q_\ell)$ since we will work with different numbers of stages. In the next lemma, we show that $R^{(m)}(q_1, \ldots, q_m)$ is an upper bound on $\widehat{\Pi}(\boldsymbol{q}, S_1, \ldots, S_m)$.

**Lemma F.1** *If $(\boldsymbol{q}, S_1, \ldots, S_m) \in \mathbb{R}^m_+ \times \mathcal{F}$ satisfies $q_{k-1} \geq q_k$ for all $k \in \mathcal{M}$, then we have $R^{(m)}(q_1, \ldots, q_m) \geq \widehat{\Pi}(\boldsymbol{q}, S_1, \ldots, S_m)$.*

*Proof:* If $(S_1, \ldots, S_m) \in \mathcal{F}$, then we have $\sum_{k \in \mathcal{M}} \sum_{i \in S_k} e^{\alpha_i} \leq T$. Thus, noting the definition of $\widehat{\Pi}(\boldsymbol{q}, S_1, \ldots, S_m)$ and using the decision variables $\boldsymbol{x} = (x_1, \ldots, x_m)$, we get

$$\widehat{\Pi}(\boldsymbol{q}, S_1, \ldots, S_m) \leq \max_{\boldsymbol{x} \in \mathbb{R}^m_+} \left\{ \sum_{k \in \mathcal{M}} (q_{k-1} - q_k) \left\{ \log x_k - \log\left( \frac{1}{q_k} - \frac{1}{q_{k-1}} \right) \right\} \; : \; \sum_{k \in \mathcal{M}} x_k \leq T \right\}. \tag{21}$$

Since $q_{k-1} \geq q_k$ for all $k \in \mathcal{M}$ with $q_0 = 1$, if $q_m = 1$, then we have $q_k = 1$ for all $k \in \mathcal{M}$, so using the fact that $\lim_{x \to 0} x \log x = 0$, we get $\widehat{\Pi}(\boldsymbol{q}, S_1, \ldots, S_m) = 0 = R^{(m)}(q_1, \ldots, q_m)$.

In the rest of the proof, we consider the case $q_m < 1$. We can solve the problem on the right side of (21) by using Lagrangian relaxation. For $(a_1, \ldots, a_m) \in \mathbb{R}^m_+$, the optimal solution $\boldsymbol{x}^*$ to the problem $\max_{\boldsymbol{x}\in\mathbb{R}^m_+} \left\{ \sum_{k\in\mathcal{M}} a_k \log x_k : \sum_{k\in\mathcal{M}} x_k \le T \right\}$ is obtained by setting $x^*_k = \frac{T}{\sum_{\ell\in\mathcal{M}} a_\ell}\, a_k$ for all $k\in\mathcal{M}$. To show this result, we can relax the constraint $\sum_{k\in\mathcal{M}} x_k \le T$ by using a Lagrange multiplier and compute the optimal value of the Lagrange multiplier by noting that this constraint must be tight at optimality. Using this result with $a_k = q_{k-1} - q_k$, since $\sum_{k\in\mathcal{M}}(q_{k-1}-q_k) = q_0 - q_m = 1 - q_m$, the optimal solution $\boldsymbol{x}^*$ to the problem on the right side of (21) is obtained by setting $x^*_k = \frac{T}{1-q_m}\,(q_{k-1}-q_k)$ for all $k\in\mathcal{M}$. Plugging this optimal solution into (21), we get

$$\begin{aligned}
\widehat{\Pi}(\boldsymbol{q}, S_1,\ldots,S_m) \;&\le\; \sum_{k\in\mathcal{M}} (q_{k-1}-q_k) \left\{ \log\left(\frac{T}{1-q_m}\right) + \log(q_{k-1}-q_k) - \log\left(\frac{1}{q_k} - \frac{1}{q_{k-1}}\right)\right\} \\
&=\; (1-q_m)\log\left(\frac{T}{1-q_m}\right) + \sum_{k\in\mathcal{M}} (q_{k-1}-q_k) \left\{ \log(q_{k-1}-q_k) - \log\left(\frac{q_{k-1}-q_k}{q_{k-1}\,q_k}\right)\right\} \\
&=\; (1-q_m)\log\left(\frac{T}{1-q_m}\right) + \sum_{k\in\mathcal{M}} (q_{k-1}-q_k)\,\log(q_{k-1}\,q_k).
\end{aligned}$$

The desired result follows by noting that the expression on the right side of the chain of inequalities above corresponds to $R^{(m)}(q_1,\ldots,q_m)$. ∎

To get an upper bound on the optimal objective value of the Pricing-Assortment problem, we can maximize the upper bound on the objective function, yielding the problem

$$\widehat{z}^{(\ell)} \;=\; \max_{(q_1,\ldots,q_\ell)\in\mathbb{R}^\ell_+} \left\{ R^{(\ell)}(q_1,\ldots,q_\ell) \;:\; 1 \ge q_1 \ge \ldots \ge q_\ell \ge 0 \right\}. \tag{22}$$

In the next lemma, we show that an optimal solution to the problem above occurs in the strict interior of the feasible set and the optimal objective value in (22) is strictly increasing in $\ell$.

**Lemma F.2** *Letting $(q^*_1,\ldots,q^*_\ell)$ be an optimal solution to problem (22), for all $\ell = 1,2,\ldots$, we have $1 > q^*_1 > \ldots > q^*_\ell > 0$ and $\widehat{z}^{(\ell)} > \widehat{z}^{(\ell-1)}$ with the convention that $\widehat{z}^{(0)} = 0$.*

*Proof:* We show the result by using induction on the number of stages. For $\ell = 1$, we have $R^{(1)}(q_1) = (1-q_1)\log q_1 + (1-q_1)\log\frac{T}{1-q_1}$, so that $R^{(1)}(0) = -\infty$, $R^{(1)}(1) = 0$ and $\frac{\partial R^{(1)}(q_1)}{\partial q_1}\Big|_{q_1=1} = -\infty$. Thus, the value of $R^{(1)}(q_1)$ at $1-\epsilon$ is strictly greater than zero for small enough $\epsilon > 0$, which implies that $\widehat{z}^{(1)} > 0 = \widehat{z}^{(0)}$ and the maximizer of $R^{(1)}(q_1)$ over the interval $[0,1]$ is in the strict interior of the interval $[0,1]$. Therefore, the result holds for $\ell = 1$. Assuming that the result holds for $\ell$ stages, we show that the result holds for $\ell+1$ stages. We have

$$\begin{aligned}
R^{(\ell+1)}(q, q_1,\ldots,q_\ell) \;&=\; (1-q)\log q + (q-q_1)\log(q\,q_1) \\
&\qquad + \sum_{k=2}^{\ell} (q_{k-1}-q_k)\log(q_{k-1}\,q_k) + (1-q_\ell)\log\left(\frac{T}{1-q_\ell}\right),
\end{aligned}$$

which follows by using the definition of $R^{(\ell)}(q_1,\ldots,q_\ell)$. In this case, subtracting the expression above from $R^{(\ell)}(q_1,\ldots,q_\ell)$, we have $R^{(\ell+1)}(q,q_1,\ldots,q_\ell) = R^{(\ell)}(q_1,\ldots,q_\ell) + f(q,q_1)$, where $f(q,q_1)$ is given by $f(q, q_1) = (1-q)\log q + (q - q_1)\,\log(q\,q_1) - (1-q_1)\,\log q_1$. By the subgradient inequality, we have $\log x < x - 1$ for all $x \in (0,1)$. Also, using the definition of $f(q,q_1)$, we have $f(1,q_1) = 0$ and $\frac{\partial f(q,q_1)}{\partial q}\big|_{q=1} = 1 - q_1 + \log q_1$. Let $(r_1^*, \ldots, r_\ell^*)$ be an optimal solution to problem (22) when we solve this problem with $\ell$ stages. By the induction assumption, we have $1 > r_1^* > \ldots > r_\ell^* > 0$. Since $r_1^* \in (0,1)$, we get $f(1, r_1^*) = 0$ and $\frac{\partial f(q,r_1^*)}{\partial q}\big|_{q=1} = 1 - r_1^* + \log r_1^* < 0$. Therefore, letting $q^*$ be an optimal solution to the problem $\max_{q \in [r_1^*,1]} f(q, r_1^*)$, the objective value of this problem at $q = 1$ is zero, but since the derivative of the objective function at $q = 1$ is strictly negative, the objective value of this problem at $q = 1 - \epsilon$ is strictly greater than zero for small enough $\epsilon > 0$. Therefore, it follows that $f(q^*, r_1^*) > 0$. In this case, we get

$$\widehat{z}^{(\ell)} \;=\; R^{(\ell)}(r_1^*, \ldots, r_\ell^*) \;<\; R^{(\ell)}(r_1^*, \ldots, r_\ell^*) + f(q^*, r_1^*) \;=\; R^{(\ell+1)}(q^*, r_1^*, \ldots, r_\ell^*) \;\overset{(a)}{\leq}\; \widehat{z}^{(\ell+1)},$$

where $(a)$ holds since $1 \geq q^* \geq r_1^* > \ldots > r_\ell^* \geq 0$, so $(q^*, r_1^*, \ldots, r_\ell^*)$ is a feasible, but not necessarily an optimal, solution to problem (22) with $\ell + 1$ stages. Thus, we have $\widehat{z}^{(\ell+1)} > \widehat{z}^{(\ell)}$.

Let $(q_1^*, \ldots, q_{\ell+1}^*)$ be an optimal solution to problem (22) with $\ell + 1$ stages. We show that $1 > q_1^* > \ldots > q_{\ell+1}^* > 0$. To get a contradiction, assume that $q_{\tau-1}^* = q_\tau^*$ for some $\tau \leq \ell + 1$, so

$$\begin{aligned}
\widehat{z}^{(\ell+1)} \;&=\; R^{(\ell+1)}(q_1^*, \ldots, q_{\ell+1}^*) \\
&=\; \sum_{k=1}^{\ell+1} (q_{k-1}^* - q_k^*)\,\log(q_{k-1}^*\,q_k^*) \;+\; (1 - q_{\ell+1}^*)\log\left(\frac{T}{1 - q_{\ell+1}^*}\right) \\
&\overset{(b)}{=}\; \sum_{k=1}^{\tau-1} (q_{k-1}^* - q_k^*)\,\log(q_{k-1}^*\,q_k^*) + \sum_{k=\tau+1}^{\ell+1} (q_{k-1}^* - q_k^*)\,\log(q_{k-1}^*\,q_k^*) + \;(1 - q_{\ell+1}^*)\log\left(\frac{T}{1 - q_{\ell+1}^*}\right) \\
&\overset{(c)}{=}\; R^{(\ell)}(q_1^*, \ldots, q_{\tau-1}^*, q_{\tau+1}^*, \ldots, q_{\ell+1}^*) \;\overset{(d)}{\leq}\; \widehat{z}^{(\ell)},
\end{aligned}$$

where $(b)$ and $(c)$ hold since $q_{\tau-1}^* = q_\tau^*$ and $(d)$ holds since $(q_1^*, \ldots, q_{\ell+1}^*)$ is a feasible solution to problem (22) with $\ell+1$ stages so $1 \geq q_1^* \geq \ldots \geq q_{\ell+1}^* \geq 0$, in which case, $(q_1^*, \ldots, q_{\tau-1}^*, q_{\tau+1}^*, \ldots, q_{\ell+1}^*)$ is a feasible, but not necessarily an optimal, solution to problem (22). The chain of inequalities above contradict the fact that $\widehat{z}^{(\ell+1)} > \widehat{z}^{(\ell)}$. Therefore, we have $q_{k-1}^* > q_k^*$ for all $k = 1, \ldots, \ell+1$. Noting the convention that $q_0 = 1$, we get $1 > q_1^* > \ldots > q_{\ell+1}^*$. Lastly, if we have $q_{\ell+1}^* = 0$, then there must exist some $k = 1, \ldots, \ell + 1$ such that $q_{k-1}^* > q_k^* = 0$, which implies that $(q_{k-1}^* - q_k^*)\,\log(q_{k-1}^*\,q_k^*) = -\infty$. Thus, by the definition of $R^{(\ell)}(q_1, \ldots, q_\ell)$, we get $\widehat{z}^{(\ell+1)} = R^{(\ell+1)}(q_1^*, \ldots, q_{\ell+1}^*) = -\infty$, contradicting the fact that $\widehat{z}^{(\ell+1)} > \widehat{z}^{(\ell)} > \ldots > \widehat{z}^{(0)} = 0$. Therefore, we have $1 > q_1^* > \ldots > q_{\ell+1}^* > 0$. In the previous paragraph, we also had $\widehat{z}^{(\ell+1)} > \widehat{z}^{(\ell)}$, so the result holds for $\ell + 1$ stages. ■

In the next lemma, we build on the lemma above to give a simple expression for the objective function of problem (22) when evaluated at its optimal solution.

**Lemma F.3** *Letting* $(q_1^*,\ldots,q_\ell^*)$ *be an optimal solution to problem* (22)*, this solution satisfies the two identities given by*

$$\begin{aligned} R^{(\ell)}(q_1^*,\ldots,q_\ell^*) &= \log q_1^* - (1+q_1^*) + \frac{q_{\ell-1}^*}{q_\ell^*} - \log(q_{\ell-1}^*\, q_\ell^*) + q_\ell^*, \\ \frac{1-q_\ell^*}{q_{\ell-1}^*\, q_\ell^*} \exp\left(\frac{q_{\ell-1}^*}{q_\ell^*}\right) &= T. \end{aligned}$$

*Proof:* Using the definition of $R^{(\ell)}(q_1,\ldots,q_\ell)$ in (20), directly by differentiating this function, we have the partial derivatives

$$\frac{\partial R^{(\ell)}(q_1,\ldots,q_\ell)}{\partial q_k} = \begin{cases} \log\left(\dfrac{q_{k+1}}{q_{k-1}}\right) + \dfrac{q_{k-1}-q_{k+1}}{q_k} & \text{if } k=1,\ldots,\ell-1 \\ \dfrac{q_{\ell-1}}{q_\ell} - \log(q_{\ell-1}\, q_\ell) - \log\left(\dfrac{T}{1-q_\ell}\right) & \text{otherwise.} \end{cases} \tag{23}$$

By Lemma F.2, $(q_1^*,\ldots,q_\ell^*)$ is in the strict interior of the feasible set of problem (22), so it satisfies the first order condition $\frac{\partial R^{(\ell)}(q_1,\ldots,q_\ell)}{\partial q_k}\big|_{(q_1,\ldots,q_\ell)=(q_1^*,\ldots,q_\ell^*)} = 0$ for all $k=1,\ldots,\ell$. By (23), we get $\log\big(\frac{q_{k+1}^*}{q_{k-1}^*}\big) = -\frac{q_{k-1}^*-q_{k+1}^*}{q_k^*}$ for all $k=1,\ldots,\ell-1$ and $\frac{q_{\ell-1}^*}{q_\ell^*} = \log\big(\frac{T}{1-q_\ell^*}\, q_{\ell-1}^*\, q_\ell^*\big)$. Solving for $T$ in the last equality yields the second identity in the lemma. By the definition of $R^{(\ell)}(q_1,\ldots,q_\ell)$, we get

$$\begin{aligned} R^{(\ell)}(q_1^*,\ldots,q_\ell^*) &= \sum_{k=1}^{\ell} q_{k-1}^* \log(q_{k-1}^*\, q_k^*) - \sum_{k=1}^{\ell} q_k^* \log(q_{k-1}^*\, q_k^*) + (1-q_\ell^*)\log\big(\tfrac{T}{1-q_\ell^*}\big) \\ &= \log q_1^* + \sum_{k=1}^{\ell-1} q_k^* \log(q_k^*\, q_{k+1}^*) - \sum_{k=1}^{\ell} q_k^* \log(q_{k-1}^*\, q_k^*) + (1-q_\ell^*)\log\big(\tfrac{T}{1-q_\ell^*}\big) \\ &= \log q_1^* + \sum_{k=1}^{\ell-1} q_k^* \log\big(\tfrac{q_{k+1}^*}{q_{k-1}^*}\big) - q_\ell^* \log(q_{\ell-1}^*\, q_\ell^*) + (1-q_\ell^*)\log\big(\tfrac{T}{1-q_\ell^*}\big) \\ &= \log q_1^* + \sum_{k=1}^{\ell-1} q_k^* \log\big(\tfrac{q_{k+1}^*}{q_{k-1}^*}\big) - q_\ell^*\Big\{\log(q_{\ell-1}^*\, q_\ell^*) + \log\big(\tfrac{T}{1-q_\ell^*}\big)\Big\} + \log\big(\tfrac{T}{1-q_\ell^*}\big) \\ &\overset{(a)}{=} \log q_1^* - \sum_{k=1}^{\ell-1} (q_{k-1}^* - q_{k+1}^*) - q_{\ell-1}^* + \frac{q_{\ell-1}^*}{q_\ell^*} - \log(q_{\ell-1}^*\, q_\ell^*) \\ &\overset{(b)}{=} \log q_1^* - (1+q_1^*) + q_\ell^* + \frac{q_{\ell-1}^*}{q_\ell^*} - \log(q_{\ell-1}^*\, q_\ell^*), \end{aligned}$$

where $(a)$ holds since $\log\big(\frac{q_{k+1}^*}{q_{k-1}^*}\big) = -\frac{q_{k-1}^*-q_{k+1}^*}{q_k^*}$ for all $k=1,\ldots,\ell-1$ and $\frac{q_{\ell-1}^*}{q_\ell^*} = \log\big(\frac{T}{1-q_\ell^*}\, q_{\ell-1}^*\, q_\ell^*\big)$, and $(b)$ follows by cancelling the telescoping terms, so the first identity in the lemma holds. ∎

In the next lemma, we give a simple inequality that will allow us to upper bound $\widehat{z}^{(\ell)}$.

**Lemma F.4** *If* $s,t>1$ *satisfies* $\log(st) + \frac{1}{t} - s = 0$*, then we have* $s \ge 2 - \frac{1}{t}$.

*Proof:* Letting $h(s) = 2(1-s) + \log s - \log(2-s)$, we have $h'(s) = -2 + \frac{1}{s} + \frac{1}{2-s} = 2\frac{(s-1)^2}{s(2-s)}$. Thus, $h(s)$ is strictly increasing in $s$ for all $s \in (1,2)$, which implies that $h(s) > h(1) = 0$ for all $s \in (1,2)$.

Also, letting $f(x) = x - \log x$, we have $f'(x) = 1 - \frac{1}{x}$, so $f(x)$ is strictly decreasing in $x$ for all $x \in (0,1)$. To get a contradiction, assume that $s, t > 1$ satisfies $\log(st) + \frac{1}{t} - s = 0$ and we have $s < 2 - \frac{1}{t}$. Since $0 < \frac{1}{t} < 2 - s < 1$ and $f(x)$ is is strictly decreasing in $x$ for all $x \in (0,1)$, we get $f(\frac{1}{t}) > f(2-s)$. Noting the definition of $f(x)$, the last inequality is equivalent to

$$\frac{1}{t} + \log(st) - s \;>\; (2-s) - \log(2-s) + \log s - s \;=\; h(s).$$

Since $1 < s < 2 - \frac{1}{t}$, we have $s \in (1,2)$. Noting that $h(s) > 0$ for all $s \in (1,2)$, the inequality above yields $\log(st) + \frac{1}{t} - s > 0$, which contradicts the fact that $\log(st) + \frac{1}{t} - s = 0$. ■

In the next proposition, we use Lemmas F.3 and F.4 to upper bound $\widehat{z}^{(\ell)}$ with a closed-form.

**Proposition F.5** *Defining the function* $G_T(x) = \frac{1}{2}\left(\sqrt{1 + 4T/e^x} + 1\right)$ *and noting that* $\widehat{z}^{(\ell)}$ *is the optimal objective value of problem* (22)*, we have*

$$\widehat{z}^{(\ell)} \;\le\; 2\log(G_T(1)) + \frac{1}{G_T(1 + \frac{1}{\ell}\, G_T(1))} - 1 + \frac{G_T(1)}{\ell}.$$

*Proof:* Let $(q_1^*, \ldots, q_\ell^*)$ be an optimal solution to problem (22). First, we give an upper bound on $1/q_\ell^*$. Letting $f(x) = \frac{1}{x}\exp(x/q_\ell^*)$, we have $f'(x) = f(x)\,(\frac{1}{q_\ell^*} - \frac{1}{x})$. Therefore, $f(x)$ is increasing in $x$ for all $x \ge q_\ell^*$. Since $q_{\ell-1}^* > q_\ell^*$, we obtain $\frac{1}{q_{\ell-1}^*}\exp(q_{\ell-1}^*/q_\ell^*) = f(q_{\ell-1}^*) \ge f(q_\ell^*) = e/q_\ell^*$. In this case, since $(q_1^*, \ldots, q_\ell^*)$ satisfies the second identity in Lemma F.3, we have

$$T \;=\; \frac{1 - q_\ell^*}{q_\ell^*}\left\{\frac{1}{q_{\ell-1}^*}\exp\left(\frac{q_{\ell-1}^*}{q_\ell^*}\right)\right\} \;\ge\; e\,\frac{1 - q_\ell^*}{(q_\ell^*)^2} \;=\; e\left\{\frac{1}{(q_\ell^*)^2} - \frac{1}{q_\ell^*}\right\}.$$

For fixed $x \in \mathbb{R}$, the only positive root of the quadratic equation $z^2 - z - \frac{T}{e^x} = 0$ is $G_T(x)$. Since $\frac{1}{(q_\ell^*)^2} - \frac{1}{q_\ell^*} - \frac{T}{e} \le 0$ by the chain of inequalities above, we get $\frac{1}{q_\ell^*} \le G_T(1)$.

Second, we give a lower bound on $1/q_\ell^*$. Letting $t_k^* = q_{k-1}^*/q_k^*$ for all $k = 1, \ldots, \ell$, by Lemma F.2, $t_k^* > 1$ for all $k = 1, \ldots, \ell$. Also, we can write the first order condition in the first case in (23) as $-\log(t_k^*\, t_{k+1}^*) + t_k^* - \frac{1}{t_{k+1}^*} = 0$ for all $k = 1, \ldots, \ell - 1$. In this case, by Lemma F.4, we get $t_k^* \ge 2 - \frac{1}{t_{k+1}^*}$ for all $k = 1, \ldots, \ell - 1$. Thus, letting $V_k^* = \frac{T}{1 - q_\ell^*}\,(q_{k-1}^* - q_k^*)$, we obtain

$$\frac{V_k^*}{V_{k+1}^*} \;=\; \frac{q_{k-1}^* - q_k^*}{q_k^* - q_{k+1}^*} \;=\; \frac{t_k^* - 1}{1 - \frac{1}{t_{k+1}^*}} \;\ge\; 1,$$

where the last inequality holds since $t_k^* \ge 2 - \frac{1}{t_{k+1}^*}$. Thus, we get $V_k^* \ge V_{k+1}^*$. By the definition of $V_k^*$, we have $\sum_{k=1}^{\ell} V_k^* = T$. Since $V_1^* \ge \ldots \ge V_\ell^*$ and $\sum_{k=1}^{\ell} V_k^* = T$, it follows that $V_\ell^* \le T/\ell$.

We have $V_\ell^* = \frac{T}{1 - q_\ell^*}\,(q_{\ell-1}^* - q_\ell^*) \le \frac{T}{\ell}$, which implies that $q_{\ell-1}^* - q_\ell^* \le \frac{1}{1 - q_\ell^*}\,(q_{\ell-1}^* - q_\ell^*) \le \frac{1}{\ell}$, in which case, dividing both sides of the last inequality by $q_\ell^*$, we get $\frac{q_{\ell-1}^*}{q_\ell^*} \le 1 + \frac{1}{\ell\, q_\ell^*} \le 1 + \frac{G_T(1)}{\ell}$, where the last inequality holds due to the fact that $\frac{1}{q_\ell^*} \le G_T(1)$. Since $(q_1^*,\ldots,q_\ell^*)$ satisfies the second identity in Lemma F.3, noting that $q_{\ell-1}^* \ge q_\ell^*$ and $\frac{q_{\ell-1}^*}{q_\ell^*} \le 1 + \frac{G_T(1)}{\ell}$, we have

$$T = \frac{1-q_\ell^*}{q_{\ell-1}^*\, q_\ell^*} \exp\left(\frac{q_{\ell-1}^*}{q_\ell^*}\right) \;\le\; \frac{1-q_\ell^*}{(q_\ell^*)^2} \exp\left(1+\frac{G_T(1)}{\ell}\right) \;=\; \exp\left(1+\frac{G_T(1)}{\ell}\right)\left\{\frac{1}{(q_\ell^*)^2} - \frac{1}{q_\ell^*}\right\}.$$

Since the only positive root of the quadratic equation $z^2 - z - \frac{T}{e^x} = 0$ is $G_T(x)$, noting that $\frac{1}{(q_\ell^*)^2} - \frac{1}{q_\ell^*} - \frac{T}{e^{1+G_T(1)/\ell}} \ge 0$ by the chain of inequalities above, we get $\frac{1}{q_\ell^*} \ge G_T(1 + \frac{1}{\ell}\, G_T(1))$.

By the subgradient inequality, we have $\log x \le x - 1$ for all $x > 0$. Noting that $(q_1^*,\ldots,q_\ell^*)$ satisfies the first identity in Lemma F.3, we get

$$\begin{aligned}
\widehat{z}^{(\ell)} \;=\; R^{(\ell)}(q_1^*,\ldots,q_\ell^*) \;&=\; \log q_1^* - (1+q_1^*) + \frac{q_{\ell-1}^*}{q_\ell^*} - \log(q_{\ell-1}^*\, q_\ell^*) + q_\ell^* \\
&\overset{(a)}{\le}\; \log q_1^* - (1+q_1^*) + 1 + \frac{G_T(1)}{\ell} - 2\log(q_\ell^*) + \frac{1}{G_T(1+\frac{1}{\ell}\, G_T(1))} \\
&\overset{(b)}{\le}\; -1 + \frac{G_T(1)}{\ell} + 2\,\log(G_T(1)) + \frac{1}{G_T(1+\frac{1}{\ell}\, G_T(1))},
\end{aligned}$$

where $(a)$ holds since we have $\frac{q_{\ell-1}^*}{q_\ell^*} \le 1 + \frac{G_T(1)}{\ell}$, $q_{\ell-1}^* \ge q_\ell^*$ and $\frac{1}{q_\ell^*} \ge G_T(1 + \frac{1}{\ell}\, G_T(1))$, whereas $(b)$ holds since $\log q_1^* \le q_1^* - 1$ and $\frac{1}{q_\ell^*} \le G_T(1)$. ∎

If we offer all products in the first stage, then the Pricing-Assortment problem reduces to the standard pricing problem under the multinomial logit model with the same price sensitivity for all products. In this case, using $W(\cdot)$ to denote the Lambert-$W$ function, it is a standard result that the optimal price to charge for all products is $\frac{1}{\beta}\,(1 + W(T/e))$, yielding the optimal expected revenue $\frac{1}{\beta}\, W(T/e)$; see Proposition 3.2 in Zhang et al. (2018). Recalling that we set $\beta = 1$, if we offer all products in the first stage, then the optimal expected revenue is $W(T/e)$. In the next theorem, we compare the optimal expected revenue that we obtain when we offer all products in the first stage with the optimal expected revenue in the Pricing-Assortment problem.

**Theorem F.6** *Noting that* $\Pi(\boldsymbol{p}, S_1,\ldots,S_m)$ *is the objective function of the* Pricing-Assortment *problem as a function of the prices* $\boldsymbol{p}$ *and the assortments* $(S_1,\ldots,S_m)$*, we have*

$$\frac{\max_{\boldsymbol{p}\in\mathbb{R}^n} \Pi(\boldsymbol{p},\mathcal{N},\varnothing,\ldots,\varnothing)}{\max_{(\boldsymbol{p},S_1,\ldots,S_m)\in\mathbb{R}^n\times\mathcal{F}} \Pi(\boldsymbol{p},S_1,\ldots,S_m)} \;\ge\; \min_{x\ge 0}\left\{\frac{(1+x)\,W(x(1+x))}{2\,(1+x)\,\log(1+x) - x}\right\}.$$

*Proof:* Let $\pi^*$ be the optimal objective value of the Pricing-Assortment problem, corresponding to the denominator of the first fraction in the theorem. Note that $\widehat{\Pi}(\boldsymbol{q}, S_1,\ldots,S_m)$ in (19) is an upper bound on the expected revenue from the no-purchase probabilities $\boldsymbol{q}$ and the assortments $(S_1,\ldots,S_m)$. Thus, we have $\pi^* \le \max_{(\boldsymbol{q},S_1,\ldots,S_m)\in\mathbb{R}_+^m\times\mathcal{F}}\{\widehat{\Pi}(\boldsymbol{q},S_1,\ldots,S_m) : q_{k-1}\ge q_k \;\forall\, k\in\mathcal{M}\}$, so by Lemma F.1, we obtain $\pi^* \le \max_{(q_1,\ldots,q_m)\in\mathbb{R}_+^m}\{R^{(m)}(q_1,\ldots,q_m) : q_{k-1}\ge q_k \;\forall\, k\in\mathcal{M}\} = \widehat{z}^{(m)}$, where the equality uses the fact that $\widehat{z}^{(m)}$ is the optimal objective value of problem (22) with $\ell = m$. By the last chain of inequalities, for all $\ell \geq m$, we get

$$\pi^* \;\leq\; \widehat{z}^{(m)} \;\overset{(a)}{\leq}\; \widehat{z}^{(\ell)} \;\overset{(b)}{\leq}\; 2\,\log(G_T(1)) + \frac{1}{G_T(1+\frac{1}{\ell}\,G_T(1))} - 1 + \frac{G_T(1)}{\ell},$$

where $(a)$ holds since $\widehat{z}^{(\ell)} \geq \widehat{z}^{(m)}$ for all $\ell \geq m$ by Lemma F.2 and $(b)$ uses Proposition F.5. Thus, for any $\ell \geq m$, we have $\pi^* \leq 2\,\log(G_T(1)) + \frac{1}{G_T(1+G_T(1)/\ell)} - 1 + \frac{G_T(1)}{\ell}$.

The last inequality holds for all $\ell \geq m$. Taking the limit as $\ell \to \infty$ and noting that $G_T(x)$ is continuous in $x$, we can upper bound $\pi^*$ as

$$\begin{aligned}
\pi^* \;&\leq\; \lim_{\ell\to\infty} \left\{ 2\,\log(G_T(1)) + \frac{1}{G_T(1+\frac{1}{\ell}\,G_T(1))} - 1 + \frac{G_T(1)}{\ell} \right\} \\
&\leq\; 2\,\log(G_T(1)) + \frac{1}{G_T(1)} - 1 \;=\; 2\,\log\left(\frac{\sqrt{1+4T/e}+1}{2}\right) + \frac{2}{\sqrt{1+4T/e}+1} - 1,
\end{aligned} \tag{24}$$

where the last equality uses the definition of $G_T(x)$ in Proposition F.5. Also, we know that if we offer all products in the first stage, then the optimal expected revenue is $W(T/e)$.

The function $f(x) = e\,x\,(1+x)$ is strictly increasing in $x$ for $x \geq 0$. Making the change of variables $T = e\,x\,(x+1)$, we have $\frac{1}{2}(\sqrt{1+4T/e}+1) = 1+x$. Therefore, we obtain

$$\begin{aligned}
\frac{\max_{\boldsymbol{p}\in\mathbb{R}^n} \Pi(\boldsymbol{p},\mathcal{N},\varnothing,\ldots,\varnothing)}{\max_{(\boldsymbol{p},S_1,\ldots,S_m)\in\mathbb{R}^n\times\mathcal{F}} \Pi(\boldsymbol{p},S_1,\ldots,S_m)} \;&=\; \frac{W(T/e)}{\pi^*} \;\overset{(c)}{\geq}\; \frac{W(T/e)}{2\,\log\left(\frac{\sqrt{1+4T/e}+1}{2}\right) + \frac{2}{\sqrt{1+4T/e}+1} - 1} \\
&\overset{(d)}{\geq}\; \min_{x\geq 0}\left\{ \frac{W(x\,(1+x))}{2\,\log(1+x) + \frac{1}{1+x} - 1} \right\} \;=\; \min_{x\geq 0}\left\{ \frac{(1+x)\,W(x(1+x))}{2\,(1+x)\,\log(1+x) - x} \right\},
\end{aligned}$$

where $(c)$ follows from (24), and $(d)$ follows by making the change of variables $T = e\,x\,(1+x)$ and minimizing the lower bound over all $x \geq 0$. ■

Here is the proof of Theorem 4.4.

**Proof of Theorem 4.4:** We argue that the optimal objective value of the minimization problem on the right side of the inequality in Theorem F.6 is at least 0.878. Let $H(x) = \frac{(1+x)\,W(x(1+x))}{2\,(1+x)\,\log(1+x)-x}$. In Figure EC.1, we plot $H(x)$ as a function of $x$ over the interval $[0, 1827]$, which is at least 0.878. It remains to demonstrate that $\min_{x\geq 1827} H(x) \geq 0.878$. For two functions $f(\cdot)$ and $g(\cdot)$ that take nonnegative values over the interval $[1827, +\infty)$, fixing $\beta = 0.878$, we have $\min_{x\geq 1827} \frac{f(x)}{g(x)} \geq \beta$ if and only if $f(x) - \beta\,g(x) \geq 0$ for all $x \geq 1827$. So, having $\min_{x\geq 1827} H(x) = 0.878$ is equivalent to having $(1+x)\,W(x(1+x)) - \beta\,(2\,(1+x)\,\log(1+x) - x) \geq 0$ for all $x \geq 1827$.

For $x \geq e$, we have $W(x) \geq \log(x) - \log\log(x)$; see Hoorfar and Hassani (2008) Therefore, it is enough to argue that $(1+x)\,(\log(x\,(1+x)) - \log\log(x\,(1+x))) - \beta\,(2\,(1+x)\,\log(1+x) - x) \geq 0$ for

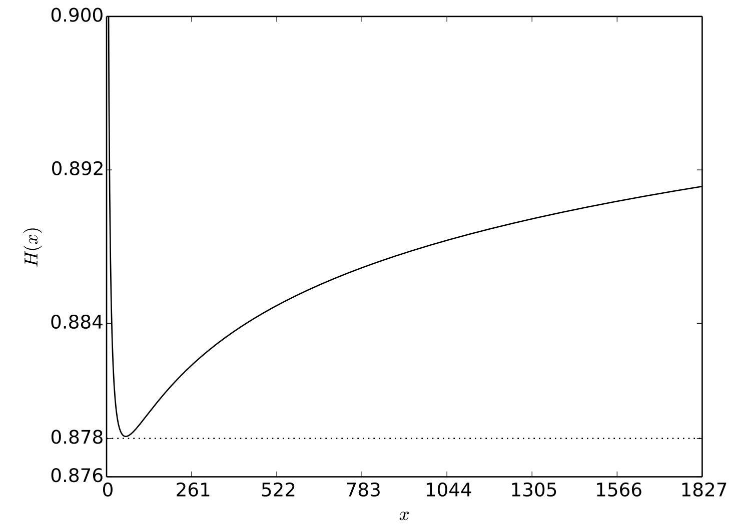


**Figure EC.1** Plot of $H(x)$ as a function of $x$.

all $x \geq 1827$. We use $F(x)$ to denote the expression on the left side of the last inequality. By direct computation with $\beta = 0.878$, $F(1827) \geq 0.31$. Next, we check that $F'(x) \geq 0$ for $x \geq 1827$. We have

$$\begin{aligned} F'(x) &= \log x + \log(1+x) + \frac{1+x}{x} + 1 - \log\log(x\,(1+x)) \\ &\qquad - \frac{1+x}{\log x + \log(1+x)}\left(\frac{1}{x} + \frac{1}{1+x}\right) - \beta - 2\,\beta\,\log(1+x) \\ &\geq 2\log x + 2 - \log 2 - \log\log(1+x) - \frac{3655/3654}{\log x} - \beta - 2\,\beta\,\log(1+x) \\ &= (2-2\beta)\log x + 2\beta\,\log\left(\frac{x}{x+1}\right) + 2 - \log 2 - \log\log(1+x) - \frac{3655/3654}{\log x} - \beta, \end{aligned} \tag{25}$$

where the first inequality holds since $\log\log(x\,(1+x)) \leq \log(2\,\log(1+x))$ and $\frac{1+x}{x} + 1 \leq \frac{3655}{1827}$ for all $x \geq 1827$. We split the expression on the right side above into two expressions.

First, consider the function $P(x) = 2\beta\,\log(\frac{x}{x+1}) + 2 - \log 2 - \frac{3651/3650}{\log x} - \beta$, which is increasing in $x$. Thus, for all $x \geq 1827$, we have $P(x) \geq P(1827) \geq 0.29$, where the second inequality is by direct computation. Second, consider the function $Q(x) = (2-2\beta)\log x - \log\log(1+x)$. By direct computation, we have $Q(1827) \geq -0.19$. Also, noting that $2 - 2\beta - \frac{1}{\log 1828} \geq 0.11$, for all $x \geq 1827$, we have $Q'(x) = (2-2\beta)\frac{1}{x} - \frac{1}{(1+x)\,\log(1+x)} \geq \frac{1}{1+x}(2 - 2\beta - \frac{1}{\log(1+x)}) \geq \frac{1}{1+x}\,(2 - 2\beta - \frac{1}{\log 1828}) \geq 0$. Thus, $Q(x)$ is increasing in $x$ for all $x \geq 1827$, so $Q(x) \geq Q(1827) \geq -0.19$. The expression on the right side of (25) is $P(x) + Q(x)$, so $F'(x) = P(x) + Q(x) \geq 0.29 - 0.19 \geq 0$ for all $x \geq 1827$. ∎

**Tightness of the Performance Guarantee of 87.8%:**

Intuitively speaking, we can reverse-engineer the sequence of steps in our proof of the performance guarantee of 87.8% to come up with a problem instance to demonstrate that this performance guarantee is tight. In particular, we use the following steps. First, we find the value of $x^*$ that minimizes the function $H(x)$ in the proof of Theorem 4.4. This value is approximately 58.83. Noting the change of variables $T = e\,x\,(1+x)$ in the proof of Theorem F.6, we set $T^* = e\,x^*\,(1+x^*)$, which is approximately 9567.33. We fix the number of stages $m$ to any positive integer. Second, setting $T = T^*$ in (20), we solve problem (22) with $\ell = m$. We let $(q_1^*, \ldots, q_m^*)$ be an optimal solution to this problem. Third, noting that the optimal value of $x_k^*$ in problem (21) in the proof of Lemma F.1 is $x_k^* = \frac{T}{1-q_m}\,(q_{k-1} - q_k)$, we set $T_k^* = \frac{T^*}{1-q_m^*}\,(q_{k-1}^* - q_k^*)$ for all $k = 1, \ldots, m$, where $T^*$ is as obtained in the first step and $(q_1^*, \ldots, q_m^*)$ is as obtained in the second step. Once we compute the values of $(T_1^*, \ldots, T_m^*)$, we construct our problem instance as follows. We have $\lambda_1 = \ldots = \lambda_m = 1$. The price sensitivity is $\beta = 1$. There is one product associated with each stage, so we index the products by $\{1, \ldots, m\}$. The parameter $\alpha_i$ for product $i$ is such that $e^{\alpha_i} = T_i^*$.

Noting that $\widehat{z}^{(m)}$ is the optimal objective value of problem (22) with $\ell = m$, by Lemma F.1, $\widehat{z}^{(m)}$ is an upper bound on the optimal objective value of the Pricing-Assortment problem. In this case, for the problem instance that we constructed as in the previous paragraph, we can follow the proof of Lemma F.1 line by line to show that if we offer product $i$ in stage $i$ and optimize only over the prices of the products in the Pricing-Assortment problem, then the optimal objective value that we obtain is equal to $\widehat{z}^{(m)}$, achieving the upper bound of $\widehat{z}^{(m)}$. Thus, for the problem instance that we constructed, the optimal solution for the Pricing-Assortment problem involves offering each product $i$ in stage $i$. In other words, for the problem instance that we constructed, we can solve the Pricing-Assortment problem efficiently. We offer each product $i$ in stage $i$ and use the approach in Section 4.1 to find the optimal prices to charge for the products.

In this case, we can follow the proofs of Proposition F.5 and Theorem F.6 line by line to show that the performance guarantee of 87.8% is tight for the problem instance that we constructed, as long as the number of stages $m$ gets arbitrarily large. The number of stages needs to get arbitrarily large due to the limit in (24). In Figure EC.2, we numerically verify the tightness of the performance guarantee of 87.8%. For each $m \in \mathbb{Z}_+$, we construct a problem instance as described in this section. For each problem instance that we construct, we solve the Pricing-Assortment problem to get the optimal objective value, which we denote by $\overline{\pi}^{(m)}$. Also, we offer all products in the first stage and compute the optimal prices to charge for the products. We denote the corresponding optimal objective value by $\underline{\pi}^{(m)}$. In the figure, as a function of the number of stages $m$ in the problem instance that we construct, we plot the ratio $\underline{\pi}^{(m)}/\overline{\pi}^{(m)}$.

Naturally, by Theorem 4.4, $\underline{\pi}^{(m)}/\overline{\pi}^{(m)}$ never falls below 87.8%. For smaller values of $m$, the ratio $\underline{\pi}^{(m)}/\overline{\pi}^{(m)}$ can be noticeably far from 87.8%, but $m$ does not need to get too large for the ratio to be close to 87.8%. Once $m$ reaches about 15, $\underline{\pi}^{(m)}/\overline{\pi}^{(m)}$ gets remarkably close to 87.8%, verifying that our analysis in the proof of Theorem 4.4 is tight for the problem instances constructed by using the approach discussed in this section, as long as $m$ gets large.

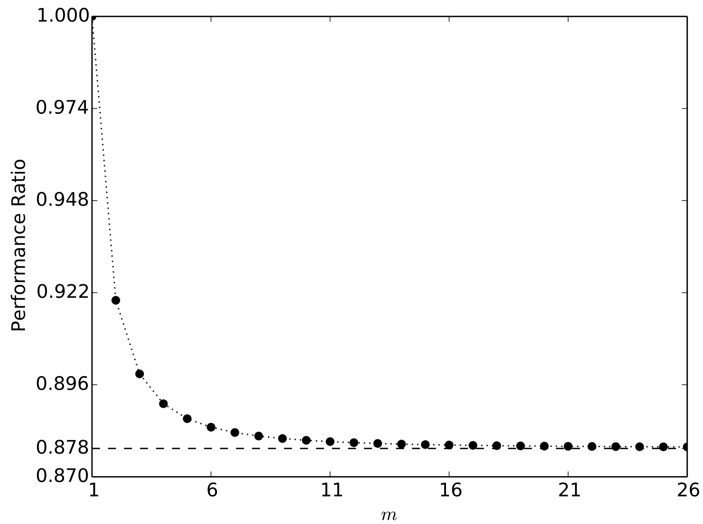


**Figure EC.2** Plot of $\underline{\pi}^{(m)}/\overline{\pi}^{(m)}$ as a function of $m$.

## Appendix G: Performance Guarantee of 50% and its Tightness

In the following lemma, we give a 50% performance guarantee for the Assortment problem by offering a nonempty assortment only in the first stage.

**Lemma G.1** *Letting $\pi^*$ be the optimal objective value of the* Assortment *problem, we have* $\max_{S\subseteq\mathcal{N}}\Pi(S,\varnothing,\ldots,\varnothing)\geq\frac{1}{2}\pi^*$.

*Proof:* Let $(S_1^*,\ldots,S_m^*)$ be an optimal solution to the Assortment problem and $T_k^*=S_1^*\cup\ldots\cup S_k^*$ with $T_0^*=\varnothing$. Noting the definition of $\Pi(S_1,\ldots,S_m)$, we get

$$
\begin{aligned}
\pi^* \;=\; \Pi(S_1^*,\ldots,S_m^*) \;&=\; \sum_{k\in\mathcal{M}}\frac{\lambda_k W(T_k^*)-\lambda_k W(T_{k-1}^*)}{(1+V(T_{k-1}^*))\,(1+V(T_k^*))}\\
&\overset{(a)}{=}\sum_{k=1}^{m-1}\frac{W(T_k^*)}{1+V(T_k^*)}\left\{\frac{\lambda_k}{1+V(T_{k-1}^*)}-\frac{\lambda_{k+1}}{1+V(T_{k+1}^*)}\right\}+\frac{\lambda_m W(T_m^*)}{(1+V(T_{m-1}^*))\,(1+V(T_m^*))}\\
&\leq\max_{S\subseteq\mathcal{N}}\left\{\frac{W(S)}{1+V(S)}\right\}\left(\sum_{k=1}^{m-1}\left\{\frac{\lambda_k}{1+V(T_{k-1}^*)}-\frac{\lambda_{k+1}}{1+V(T_{k+1}^*)}\right\}+\frac{\lambda_m}{1+V(T_{m-1}^*)}\right)\\
&\overset{(b)}{=}\max_{S\subseteq\mathcal{N}}\Big\{\Pi(S,\varnothing,\ldots,\varnothing)\Big\}\left(\lambda_1+\sum_{k=2}^{m}\lambda_k\left\{\frac{1}{1+V(T_{k-1}^*)}-\frac{1}{1+V(T_k^*)}\right\}\right)\\
&\leq\max_{S\subseteq\mathcal{N}}\Big\{\Pi(S,\varnothing,\ldots,\varnothing)\Big\}\left(1+\sum_{k=2}^{m}\left\{\frac{1}{1+V(T_{k-1}^*)}-\frac{1}{1+V(T_k^*)}\right\}\right)\\
&=\max_{S\subseteq\mathcal{N}}\Big\{\Pi(S,\varnothing,\ldots,\varnothing)\Big\}\left(1+\frac{1}{1+V(T_1^*)}-\frac{1}{1+V(T_m^*)}\right)\;\leq 2\max_{S\subseteq\mathcal{N}}\Big\{\Pi(S,\varnothing,\ldots,\varnothing)\Big\},
\end{aligned}
$$

where $(a)$ follows by by arranging the terms and $(b)$ follows by noting that $\Pi(S,\varnothing,\ldots,\varnothing)=\lambda_1\frac{W(S)}{1+V(S)}$ and $\lambda_1=1$, as well as arranging the terms in the sum on the left side of the equality. ■

| $(S_1,S_2)$ | Exp. Rev |
|---|---|
| $(\{1\},\varnothing)$ | $\frac{r_1\,v_1}{1+v_1}=\frac{(1+1/\epsilon)\epsilon}{1+\epsilon}=1\xrightarrow{\epsilon\to 0}1$ |
| $(\{2\},\varnothing)$ | $\frac{r_2\,v_2}{1+v_2}=\frac{1/\epsilon}{1+1/\epsilon}=\frac{1}{1+\epsilon}\xrightarrow{\epsilon\to 0}1$ |
| $(\{1,2\},\varnothing)$ | $\frac{r_1\,v_1+r_2\,v_2}{1+v_1+v_2}=\frac{(1+1/\epsilon)\epsilon+1/\epsilon}{1+\epsilon+1/\epsilon}=1\xrightarrow{\epsilon\to 0}1$ |
| $(\{1\},\{2\})$ | $\frac{r_1\,v_1}{1+v_1}+\frac{r_2\,v_2}{(1+v_1)\,(1+v_1+v_2)}=\frac{(1+1/\epsilon)\epsilon}{1+\epsilon}+\frac{1/\epsilon}{(1+\epsilon)\,(1+\epsilon+1/\epsilon)}=1+\frac{1}{(1+\epsilon)\,(1+\epsilon+\epsilon^2)}\xrightarrow{\epsilon\to 0}2$ |
| $(\{2\},\{1\})$ | $\frac{r_2\,v_2}{1+v_2}+\frac{r_1\,v_1}{(1+v_2)\,(1+v_2+v_1)}=\frac{1/\epsilon}{1+1/\epsilon}+\frac{(1+1/\epsilon)\,\epsilon}{(1+1/\epsilon)\,(1+1/\epsilon+\epsilon)}=\frac{1}{1+\epsilon}+\frac{\epsilon^2(1+\epsilon)}{(1+\epsilon)\,(1+\epsilon+\epsilon^2)}\xrightarrow{\epsilon\to 0}1$ |

**Table EC.2** Expected revenue from non-dominated assortments.

**Tightness of the Performance Guarantee of 50%:**

We give a problem instance to demonstrate that the performance guarantee of 50% that we give for the Assortment problem in Lemma G.1 is tight. We consider a problem instance with two products and two stages. The revenues and preference weights of the products are $r_1=1+1/\epsilon$, $r_2=1$, $v_1=\epsilon$, and $v_2=1/\epsilon$. The distribution of the patience level is given by $\lambda_1=\lambda_2=1$. In Table EC.2, we give the expected revenue from each non-dominated solution, along with the limit of the expected revenue as $\epsilon\to 0$. If we offer the empty assortment in all stages except for the first one, then the largest expected revenue that we can obtain is the expected revenue from one of the solutions $(\{1\},\varnothing)$, $(\{2\},\varnothing)$ and $(\{1,2\},\varnothing)$, all of which get arbitrarily close to one, as we choose $\epsilon$ arbitrarily small. On the other hand, as we choose $\epsilon$ arbitrarily small, noting the solution $(\{1\},\{2\})$, the largest expected revenue from any solution is arbitrarily close to two. Thus, the performance guarantee of 50% that we give for the Assortment problem in Lemma G.1 is tight. To make the contrast, for the Pricing-Assortment problem, offering the empty assortment in all stages except for the first one and finding the revenue-maximizing prices in the first stage provides a tight performance guarantee of 87.8%. On the other hand, for the Assortment problem, offering the empty assortment in all stages except for the first one and finding the revenue-maximizing assortment in the first stage provides a tight performance guarantee of 50%.

## Appendix H: Complexity of Joint Pricing and Assortment Optimization

We consider the Pricing-Assortment problem when the prices of the products take values only over a finite set. We show that the problem is NP-hard even when we have only two possible price levels for the products and the choice process of the customers involves only two stages with $\lambda_1=\lambda_2=1$. Consider the following instance. The set of products is $\mathcal{N}=\{1,2,\ldots,n\}$. We have two stages with $\lambda_1=\lambda_2=1$. We have two price levels, which we denote by $p_H$ and $p_L$ with $p_H>p_L$. For each product $i$, if we offer it at price $q\in\{p_H,p_L\}$, then its preference weight is given by $v_{iq}$. We want to find the sequence of assortments to offer in the two stages and the prices to charge for the products to maximize the expected revenue. Using the vector $\boldsymbol{p}=(p_1,\ldots,p_n)$ to denote the prices that we charge for the products and $(S_1, S_2)$ to denote the assortments that we offer in the two stages, noting the expected revenue expression in (3), we want to solve the problem

$$\max_{(\boldsymbol{p}, S_1, S_2) \in \{p_L, p_H\}^n \times \mathcal{F}} \left\{ \frac{\sum_{i \in S_1} p_i \, v_{i,p_i}}{1 + \sum_{i \in S_1} v_{i,p_i}} + \frac{\sum_{i \in S_2} p_i \, v_{i,p_i}}{(1 + \sum_{i \in S_1} v_{i,p_i})\,(1 + \sum_{i \in S_1 \cup S_2} v_{i,p_i})} \right\}. \tag{26}$$

In the next lemma, we give a structural property of an optimal solution to the problem above to express it in a simpler fashion. We defer the proofs of auxiliary lemmas to the end of this section.

**Lemma H.1** *There exists an optimal solution $(\boldsymbol{p}^*, S_1^*, S_2^*)$ to problem (26) such that all products are offered; that is, $S_1^* \cup S_2^* = \mathcal{N}$. All products in the first stage have the high price and all products in the second stage have the low price; that is, $p_i^* = p_H$ for all $i \in S_1^*$ and $p_i^* = p_L$ for all $i \in S_2^*$.*

By Lemma H.1, the critical decision is the assortment offered $S$ in the first stage, in which case, we offer the assortment $\mathcal{N} \setminus S$ in the second stage. Thus, problem (26) is equivalent to

$$\max_{S \subseteq \mathcal{N}} \left\{ \frac{p_H \sum_{i \in S} v_{i,H}}{1 + \sum_{i \in S} v_{i,H}} + \frac{p_L \sum_{i \notin S} v_{i,L}}{(1 + \sum_{i \in S} v_{i,H})\,(1 + \sum_{i \in S} v_{i,H} + \sum_{i \notin S} v_{i,L})} \right\}, \tag{27}$$

where we let $v_{i,H} = v_{i,p_H}$ and $v_{i,L} = v_{i,p_L}$ for notational brevity. To establish the computational complexity, we will consider the following decision-theoretic version of the problem above.

**Two Stages and Two Price Levels:**

**Inputs:** A set of products index by $\mathcal{N} = \{1, \ldots, n\}$, two price levels $p_H$ and $p_L$ with $p_H > p_L > 0$, two preference weights $v_{i,H}$ and $v_{i,L}$ for each product $i \in \mathcal{N}$, and an expected revenue target $T$.

**Question:** Does there exist a subset of products $S \subseteq \mathcal{N}$ that provides an expected revenue of $T$ or more in problem (27)?

The main result of this section is given in the following theorem, showing that the Two Stages and Two Price Levels problem is NP-complete.

**Theorem H.2** *The* Two Stages and Two Price Levels *problem is NP-complete.*

We will use two auxiliary lemmas in the proof of the theorem above. The first lemma focuses on the complexity of the following variant of the subset sum problem.

**Three-Quarters Subset Sum:**

**Inputs:** A collection of weights $w_1, w_2, \ldots, w_n$ such that $w_i \in \mathbb{Q}_{++}$ for all $i = 1, \ldots, n$.

**Question:** Does there exist a subset $S \subseteq \{1, \ldots, n\}$ such that $\sum_{i \in S} w_i = \frac{3}{4} \sum_{i=1}^n w_i$.

In the following lemma, we show that the Three-Quarters Subset Sum problem is NP-complete. We give the proof of this lemma also at the end of this section.

**Lemma H.3** *The* Three-Quarters Subset Sum *problem is NP-complete.*

Lastly, in the next lemma, we characterize the maximizer of a function that is crucial in our NP-completeness proof. The proof of this lemma is at the end of this section as well.

**Lemma H.4** *For each $\pi > 1$ and $\alpha > 1$ such that $1 < \frac{\alpha-1}{\pi-1} < (1+\alpha)^2$, define $f_{\pi,\alpha} : [0,1] \to \mathbb{R}_+$ as*

$$f_{\pi,\alpha}(x) \;=\; \frac{\pi x}{1+x} + \frac{\alpha(1-x)}{(1+x)(1+\alpha-(\alpha-1)x)}.$$

*Then, $f_{\pi,\alpha}$ achieves its unique maximum at* $x^* = \dfrac{1+\alpha-\sqrt{(\alpha-1)/(\pi-1)}}{-1+\alpha+\sqrt{(\alpha-1)/(\pi-1)}}$.

Here is the proof of Theorem H.2.

**Proof of Theorem H.2:** We will use a reduction from the Three-Quarters Subset Sum problem, which is NP-complete by Lemma H.3. Consider an arbitrary instance of the Three-Quarters Subset Sum problem with the weights $w_1, w_2, \ldots, w_n$. Without loss of generality, we assume that $\sum_{i=1}^n w_i = 1$, because we can normalize all of the weights by dividing them by $\sum_{i=1}^n w_i$ without changing the answer to the problem. The Three-Quarters Subset Sum problem asks whether there exists a subset $S \subseteq \{1, \ldots, n\}$ such that $\sum_{i\in S} w_i = \frac{3}{4}$.

We construct an instance of the Two Stages and Two Price Levels problem as follows. The set of products is $\{1, \ldots, n\}$. The two price levels are $p_H = \frac{5}{2}$ and $p_L = 1$ with the corresponding preference weights $v_{i,H} = w_i$ and $v_{i,L} = 7w_i$ for each product $i$. Considering the function $f_{\pi,\alpha}$ in Lemma H.4 with $\pi = \frac{5}{2}$ and $\alpha = 7$, we set the expected revenue target as $T = f_{\frac{5}{2},7}(3/4)$. Let $\text{Rev}(S)$ be the expected revenue in this Two Stages and Two Price Levels problem. Noting that $\sum_{i\notin S} w_i = \sum_{i=1}^n w_i - \sum_{i\in S} w_i = 1 - \sum_{i\in S} w_i$, by (27), we have

$$\begin{aligned}\text{Rev}(S) \;&=\; \frac{\frac{5}{2}\sum_{i\in S} w_i}{1+\sum_{i\in S} w_i} \;+\; \frac{7\sum_{i\notin S} w_i}{(1+\sum_{i\in S} w_i)\,(1+\sum_{i\in S} w_i + 7\sum_{i\notin S} w_i)}\\ &=\; \frac{\frac{5}{2}\sum_{i\in S} w_i}{1+\sum_{i\in S} w_i} \;+\; \frac{7(1-\sum_{i\in S} w_i)}{(1+\sum_{i\in S} w_i)\,(8-6\sum_{i\in S} w_i)}.\end{aligned}$$

We will show that there exists a subset $A \subseteq \{1, \ldots, n\}$ such that $\sum_{i\in A} w_i = \frac{3}{4}$ in the Three-Quarters Subset Sum Problem if and only if there exists a subset $S \subseteq \{1, \ldots, n\}$ in the Two-Stages and Two Price Levels problem such that $\text{Rev}(S) \ge T$.

By the definitions of $\text{Rev}(S)$ above and $f_{\pi,\alpha}$ in Lemma H.4, $\text{Rev}(S) = f_{\frac{5}{2},7}(\sum_{i\in S} w_i)$. Also, our choice of $\pi = \frac{5}{2}$ and $\alpha = 7$ satisfies $\pi > 1$, $\alpha > 1$ and $1 < \frac{\alpha-1}{\pi-1} < (1+\alpha)^2$, so by Lemma H.4, the function $f_{\frac{5}{2},7}$ achieves its unique maximum at $x^* = \frac{1+7-\sqrt{(7-1)/(\frac{5}{2}-1)}}{-1+7+\sqrt{(7-1)/(\frac{5}{2}-1)}} = \frac{3}{4}$. Thus, for any subset $S \subseteq \{1, \ldots, n\}$, we have $\text{Rev}(S) = f_{\frac{5}{2},7}(\sum_{i\in S} w_i) \le f_{\frac{5}{2},7}(3/4) = T$ and the inequality holds as an equality if and only if $\sum_{i\in S} w_i = 3/4$. Therefore, there exists a subset $S \subseteq \{1, \ldots, n\}$ such that $\text{Rev}(S) \ge T$ if and only if there exists a subset $S \subseteq \{1, \ldots, n\}$ such that $\sum_{i\in S} w_i = \frac{3}{4}$. ■

**Proofs of Auxiliary Lemmas:**

In the rest of this section, we give the proofs for the auxiliary lemmas that we used to show Theorem H.2. Here is the proof of Lemma H.1.

**Proof of Lemma H.1:** Let $(\boldsymbol{p}^*, S_1^*, S_2^*)$ be an optimal solution to problem (26). If $i \notin S_1^* \cup S_2^*$, then offering product $i$ in the second stage at price level $p_H$ does not degrade the expected revenue, so we can assume that $S_1^* \cup S_2^* = \mathcal{N}$. Considering the prices $\boldsymbol{p}^*$, let $H^* = \{i \in \mathcal{N} : p_i^* = p_H\}$ and $L^* = \{i \in \mathcal{N} : p_i^* = p_L\}$ be the sets of products for which we charge the two price levels. Fixing the prices at $\boldsymbol{p}^*$ and optimizing over the sequence of assortments $(S_1, S_2) \in \mathcal{F}$, problem (26) becomes equivalent to the ASSORTMENT problem. Thus, by the revenue-ordered property in Theorem 3.1, one of the three solutions $(H^* \cup L^*, \varnothing)$, $(H^*, L^*)$, and $(H^*, \varnothing)$ is optimal to this problem. In particular, we have two stages, so noting the revenue thresholds $+\infty = t_1^* \geq t_2^* \geq t_3^*$ in Theorem 3.1, the solutions $(H^* \cup L^*, \varnothing)$, $(H^*, L^*)$, and $(H^*, \varnothing)$, respectively, correspond to the cases $+\infty = t_1^* > p_H > p_L \geq t_2^* \geq t_3^*$, $+\infty = t_1^* > p_H \geq t_2^* > p_L \geq t_3^*$, and $+\infty = t_1^* > p_H \geq t_2^* \geq t_3^* > p_L$. The solution $(H^*, L^*)$ does not degrade the expected revenue from the solution $(H^*, \varnothing)$, since offering some product at the second stage provides additional expected revenue without changing the expected revenue from the first stage. Thus, it is enough to show that the solution $(H^*, L^*)$ does not degrade the expected revenue from the solution $(H^* \cup L^*, \varnothing)$. Let $V_{H^*} = \sum_{i \in H^*} v_{i,H}$ and $V_{L^*} = \sum_{i \in L^*} v_{i,L}$, so

$$
\begin{aligned}
\frac{p_H V_{H^*} + p_L V_{L^*}}{1 + V_{H^*} + V_{L^*}} &= \frac{p_H V_{H^*}}{1 + V_{H^*}} + \frac{p_H V_{H^*}}{1 + V_{H^*} + V_{L^*}} - \frac{p_H V_{H^*}}{1 + V_{H^*}} \\
&\quad + \frac{p_L V_{L^*}}{(1 + V_{H^*})(1 + V_{H^*} + V_{L^*})} + \frac{p_L V_{L^*}}{1 + V_{H^*} + V_{L^*}} - \frac{p_L V_{L^*}}{(1 + V_{H^*})(1 + V_{H^*} + V_{L^*})} \\
&= \frac{p_H V_{H^*}}{1 + V_{H^*}} - \frac{p_H V_{H^*} V_{L^*}}{(1 + V_{H^*})(1 + V_{H^*} + V_{L^*})} \\
&\quad + \frac{p_L V_{L^*}}{(1 + V_{H^*})(1 + V_{H^*} + V_{L^*})} + \frac{p_L V_{L^*} V_{H^*}}{(1 + V_{H^*})(1 + V_{H^*} + V_{L^*})} \\
&\leq \frac{p_H V_{H^*}}{1 + V_{H^*}} + \frac{p_L V_{L^*}}{(1 + V_{H^*})(1 + V_{H^*} + V_{L^*})},
\end{aligned}
$$

where the inequality uses the fact that $p_H^* > p_L^*$. The first and last expressions above are, respectively, the expected revenues from the solutions $(H^* \cup L^*, \varnothing)$ and $(H^*, L^*)$. ■

Next, we give a proof for Lemma H.3.

**Proof of Lemma H.3:** We use a reduction from the standard PARTITION problem, which is a well-known NP-complete problem; see Garey and Johnson (1979). Consider an arbitrary instance of the PARTITION problem, where we have a collection of $n$ items with weights $\{s_1, s_2, \ldots, s_n\} \in \mathbb{Q}_{++}$. Letting $T = \sum_{i=1}^n s_i$, the question is to determine whether there exists a subset $S \subseteq \{1, \ldots, n\}$ such that $\sum_{i \in S} s_i = T/2$. Given an instance of the PARTITION problem, we construct instance of the THREE-QUARTERS SUBSET SUM problem with $n+1$ items, where $w_1 = s_1, \ldots, w_n = s_n$ and

$w_{n+1}=T$. We will show there exists a subset $S\subseteq\{1,\ldots,n\}$ in the Partition problem such that $\sum_{i\in S} w_i = T/2$ if and only if there exists a subset $A\subseteq\{1,\ldots,n,n+1\}$ in the Three-Quarters Subset Sum problem such that $\sum_{i\in A} w_i = \frac{3}{4}\sum_{i=1}^{n+1} w_i$.

Assume that there exists a subset $S\subseteq\{1,\ldots,n\}$ such that $\sum_{i\in S} s_i = T/2$. In this case, the subset $A=S\cup\{n+1\}\subseteq\{1,\ldots,n\}$ satisfies $\sum_{i\in A} w_i = w_{n+1}+\sum_{i\in S} w_i = T+(T/2)=\frac{3}{2}T=\frac{3}{4}(2T)=\frac{3}{4}\sum_{i=1}^{n+1} w_i$. On the other hand, assume that there exists a subset $A\subseteq\{1,\ldots,n,n+1\}$ such that $\sum_{i\in A} w_i = \frac{3}{4}\sum_{i=1}^{n+1} w_i = \frac{3}{2}T$. In this case, note that we must have $n+1\in A$, because $\sum_{i=1}^{n} w_i = T$, but $\sum_{i\in A} w_i = \frac{3}{2}T$. Since $w_{n+1}=T$ and $n+1\in A$, it follows that the subset $S=A\setminus\{n+1\}\subseteq\{1,\ldots,n\}$ satisfies $\sum_{i\in S} w_i = \sum_{i\in A} w_i - w_{n+1} = \frac{3}{2}T - T = T/2$. ■

Here is the proof of Lemma H.4.

**Proof of Lemma H.4:** Letting $G(x)=1+\alpha-(\alpha-1)\,x$ for notational brevity, we have $G(x)-2\alpha=(1-\alpha)\,(1+x)$ and $G(x)-\alpha\,(1-x)=1+x$. Furthermore, we can express $f_{\pi,\alpha}(x)$ as $f_{\pi,\alpha}(x)=\frac{\pi\, x}{1+x}+\alpha\,\frac{1-x}{1+x}\cdot\frac{1}{G(x)}$. Noting that $G'(x)=-(\alpha-1)$, differentiating $f_{\pi,\alpha}(x)$, we get

$$
\begin{aligned}
f'_{\pi,\alpha}(x) &= \frac{\pi}{(1+x)^2}-\alpha\,\frac{2}{(1+x)^2}\cdot\frac{1}{G(x)}+\alpha\,\frac{1-x}{1+x}\cdot\frac{\alpha-1}{G(x)^2}\\
&= \frac{1}{G(x)^2}\left\{\frac{\pi\, G(x)^2-2\alpha\, G(x)+\alpha(\alpha-1)\,(1-x)\,(1+x)}{(1+x)^2}\right\}\\
&= \frac{1}{G(x)^2}\left\{\frac{(\pi-1)\,G(x)^2+G(x)\,(G(x)-2\alpha)+\alpha(\alpha-1)\,(1-x)\,(1+x)}{(1+x)^2}\right\}\\
&\overset{(a)}{=} \frac{1}{G(x)^2}\left\{\frac{(\pi-1)\,G(x)^2+(1-\alpha)\,(1+x)\,(G(x)-\alpha\,(1-x))}{(1+x)^2}\right\}\\
&\overset{(b)}{=} \frac{1}{G(x)^2}\left\{\frac{(\pi-1)\,G(x)^2+(1-\alpha)\,(1+x)^2}{(1+x)^2}\right\}\\
&= \frac{\pi-1}{G(x)^2}\left\{\left(\frac{G(x)}{1+x}\right)^2-\frac{\alpha-1}{\pi-1}\right\} \overset{(c)}{=} \frac{\pi-1}{(1+\alpha-(\alpha-1)\,x)^2}\left\{\left(1+\alpha\,\frac{1-x}{1+x}\right)^2-\frac{\alpha-1}{\pi-1}\right\},
\end{aligned}
$$

where $(a)$ holds since $G(x)-2\alpha=(1-\alpha)\,(1+x)$, $(b)$ holds since $G(x)-\alpha\,(1-x)=1+x$ and $(c)$ holds by the definition of $G(x)$. Therefore, defining the function $g:[0,1]\to\mathbb{R}$ as $g(x)=\left(1+\alpha\,\frac{1-x}{1+x}\right)^2-\frac{\alpha-1}{\pi-1}$, the sign of $f'_{\pi,\alpha}(x)$ is determined by the sign of $g(x)$.

We have $g'(x)=-\frac{4\alpha}{(1+x)^2}\left(1+\alpha\frac{1-x}{1+x}\right)<0$ for all $x\in[0,1]$, so $g(x)$ is strictly decreasing in $x$ over the interval $[0,1]$. Also, since $1<\frac{\alpha-1}{\pi-1}<(1+\alpha)^2$, we get $(1+\alpha)^2-\frac{\alpha-1}{\pi-1}=g(0)>0>g(1)=1-\frac{\alpha-1}{\pi-1}$, which implies that $g(x)$ crosses zero at a unique point over the interval $[0,1]$.

By the discussion in the previous paragraph, $f_{\pi,\alpha}(x)$ is strictly increasing, then strictly decreasing in $x$ over the interval $[0,1]$, so it has a unique maximizer over this interval. To find the maximizer $x^*$ of $f_{\pi,\alpha}(x)$, we set $g(x^*)=\left(1+\alpha\,\frac{1-x^*}{1+x^*}\right)^2-\frac{\alpha-1}{\pi-1}=0$, yielding $\alpha\,\frac{1-x^*}{1+x^*}=\sqrt{\frac{\alpha-1}{\pi-1}}-1$. For constants

$a$ and $b$, the value of $x$ that solves $a\frac{1-x}{1+x}=b$ is $\frac{a-b}{a+b}$. Thus, setting $a=\alpha$ and $b=\sqrt{\frac{\alpha-1}{\pi-1}}-1$ in the last equality, we get $x^*=\frac{1+\alpha-\sqrt{(\alpha-1)/(\pi-1)}}{-1+\alpha+\sqrt{(\alpha-1)/(\pi-1)}}$, as desired. Lastly, we check that this value of $x^*$ is in the interval $[0,1]$. Noting that $\sqrt{\frac{\alpha-1}{\pi-1}}<1+\alpha$ and $\alpha>1$, we get $x^*\geq 0$. Also, $1-\sqrt{\frac{\alpha-1}{\pi-1}}<-1+\sqrt{\frac{\alpha-1}{\pi-1}}$, so adding $\alpha$ to both sides of this inequality and dividing by $-1+\alpha+\sqrt{\frac{\alpha-1}{\pi-1}}$, we get $x^*\leq 1$. ■

## Appendix I: Proof of Lemma 5.1

For $i\in S_k^*$ and $j\in S_{k+1}^*$, we must have $r_i\geq t_{k+1}(S_1^*,\ldots,S_m^*)>r_j$. In particular, by the first part of Lemma 3.2, if $r_j\geq t_{k+1}(S_1^*,\ldots,S_m^*)$, then we can move product $j$ from stage $k+1$ to stage $k$ without degrading the expected revenue provided by the solution $(S_1^*,\ldots,S_m^*)$, which contradicts the fact that $(S_1^*,\ldots,S_m^*)$ is non-dominated. By the second part of Lemma 3.2, if $r_i<t_{k+1}(S_1^*,\ldots,S_m^*)$, then we can move product $i$ from stage $k$ to stage $k+1$ to obtain a solution strictly better than the solution $(S_1^*,\ldots,S_m^*)$, which contradicts the fact that $(S_1^*,\ldots,S_m^*)$ is an optimal solution. Thus, if $r_i\in S_k^*$ and $j\in S_{k+1}^*$, then we must have $r_i>r_j$. ■

## Appendix J: Proof of Lemma 5.3

In this section, we give a proof for Lemma 5.3. We need the next intermediate lemma, where we show a monotonicity property for the value functions computed through (10).

**Lemma J.1** *If the value functions* $\{\Theta_i^\ell(x,y):(x,y)\in \text{DOM}^2,\ i=1,\ldots,\ell+1\}$ *are computed through the dynamic program in* (10)*, then* $\Theta_i^\ell(x,y)$ *is increasing in* $x$ *and decreasing in* $y$*.*

*Proof:* We show the result by using induction over the decision epochs. By the boundary condition, $\Theta_{\ell+1}^\ell(x,y)$ is increasing in $x$ and decreasing in $y$. Assuming that $\Theta_{i+1}^\ell(x,y)$ is increasing in $x$ and decreasing in $y$, we proceed to showing that $\Theta_i^\ell(x,y)$ is increasing in $x$ and decreasing in $y$. Since $\lfloor a\rfloor$ and $\lceil a\rceil$ are increasing in $a$, $\lfloor x-v_i\,r_i\,u_i\rfloor$ and $\lceil y-v_i\,u_i\rceil$ are increasing in $x$ and $y$, in which case, by the induction hypothesis, $\Theta_{i+1}^\ell(\lfloor x-v_i\,r_i\,u_i\rfloor,\lceil y-v_i\,u_i\rceil)$ is increasing in $x$ and decreasing in $y$. Thus, for a fixed value of $u_i$, the objective function of the minimization problem in (10) is increasing in $x$ and decreasing in $y$. So, the optimal objective value of this minimization problem, which is equal to $\Theta_i^\ell(x,y)$, must be increasing in $x$ and decreasing in $y$ as well. ■

Here is the proof of Lemma 5.3.

**Proof of Lemma 5.3:** Throughout the proof, let $S\subseteq\{j+1,\ldots,\ell\}$ be such that $W(S)\geq x$ and $V(S)\leq y$. Our proof proceeds in three parts.

**Part 1:** First, assuming that such an assortment $S$ exists, we show that $\Theta_{j+1}^\ell(x,y)<+\infty$. For notational brevity, we let $S^i=S\cap\{i,\ldots,\ell\}$. Also, we define $\widetilde{u}_i\in\{0,1\}$ as $\widetilde{u}_i=1$ if and only if

$i \in S$. Since $\widetilde{u}_i$ is a feasible but not necessarily an optimal solution to the minimization problem on the right side of (10) with $x = W(S^i)$ and $y = V(S^i)$, we have

$$\begin{aligned}\Theta_i^\ell(W(S^i), V(S^i)) &\le c_i\,\widetilde{u}_i + \Theta_{i+1}^\ell(\lfloor W(S^i) - v_i\,r_i\,\widetilde{u}_i\rfloor, \lceil V(S^i) - v_i\,\widetilde{u}_i\rceil) \\ &= c_i\,\widetilde{u}_i + \Theta_{i+1}^\ell(\lfloor W(S^{i+1})\rfloor, \lceil V(S^{i+1})\rceil) \\ &\le c_i\,\widetilde{u}_i + \Theta_{i+1}^\ell(W(S^{i+1}), V(S^{i+1})),\end{aligned}$$

where the last inequality uses the fact that $\Theta_{i+1}^\ell(x,y)$ is increasing in $x$ and decreasing in $y$ by Lemma J.1, along with the fact that $\lfloor W(S^{i+1})\rfloor \le W(S^{i+1})$ and $\lceil V(S^{i+1})\rceil \ge V(S^{i+1})$.

Since $S \subseteq \{j+1, \ldots, \ell\}$, we have $S^{j+1} = S$ and $S^{\ell+1} = \varnothing$ by the definition of $S^i$ in the previous paragraph. Thus, adding the chain of inequalities above over all $i = j+1, \ldots, \ell$, we obtain

$$\Theta_{j+1}^\ell(W(S), V(S)) \le \sum_{i=j+1}^{\ell} c_i\,\widetilde{u}_i + \Theta_{\ell+1}^\ell(W(\varnothing), V(\varnothing)) = C(S),$$

where the equality uses the fact that $\Theta_{\ell+1}^\ell(0,0) = 0$. Since $W(S) \ge x$, $V(S) \le y$, using Lemma J.1 once again, we get $\Theta_{j+1}^\ell(x,y) \le \Theta_{j+1}^\ell(W(S), V(S)) \le C(S) < +\infty$.

**Part 2:** Second, we show that $C(\widehat{S}_{x,y}) \le C(S)$. Noting the last chain of inequalities at the end of the previous paragraph, it is enough to show that $\Theta_{j+1}^\ell(x,y) = C(\widehat{S}_{x,y})$.

Consider executing the candidate construction algorithm with $(x,y) \in \textsc{Dom}^2$. By Steps 2 and 3 in the candidate construction algorithm, along with the dynamic program in (10), we have

$$\Theta_i^\ell(\widehat{x}_i, \widehat{y}_i) = c_i\,\widehat{u}_i + \Theta_{i+1}^\ell(\widehat{x}_{i+1}, \widehat{y}_{i+1}).$$

Adding this equality over all $i = j+1, \ldots, \ell$, we get $\Theta_{j+1}^\ell(\widehat{x}_{j+1}, \widehat{y}_{j+1}) = C(\widehat{S}_{x,y}) + \Theta_{\ell+1}^\ell(\widehat{x}_{\ell+1}, \widehat{y}_{\ell+1})$. Since we start the candidate construction algorithm with $\widehat{x}_{j+1} = x$ and $\widehat{y}_{j+1} = y$, the last equality yields $\Theta_{j+1}^\ell(x,y) = C(\widehat{S}_{x,y}) + \Theta_{\ell+1}^\ell(\widehat{x}_{\ell+1}, \widehat{y}_{\ell+1})$. By Part 1, $\Theta_{j+1}^\ell(x,y) < +\infty$. Also, $\Theta_{\ell+1}^\ell(\widehat{x}_{\ell+1}, \widehat{y}_{\ell+1})$ takes the value $+\infty$ or zero. If $\Theta_{\ell+1}^\ell(\widehat{x}_{\ell+1}, \widehat{y}_{\ell+1}) = +\infty$, then we get a contradiction to the fact that $\Theta_{j+1}^\ell(x,y) < +\infty$ and $\Theta_{j+1}^\ell(x,y) = C(\widehat{S}_{x,y}) + \Theta_{\ell+1}^\ell(\widehat{x}_{\ell+1}, \widehat{y}_{\ell+1})$. Thus, we have $\Theta_{\ell+1}^\ell(\widehat{x}_{\ell+1}, \widehat{y}_{\ell+1}) = 0$, in which case, having $\Theta_{j+1}^\ell(x,y) = C(\widehat{S}_{x,y}) + \Theta_{\ell+1}^\ell(\widehat{x}_{\ell+1}, \widehat{y}_{\ell+1})$ yields $\Theta_{j+1}^\ell(x,y) = C(\widehat{S}_{x,y})$.

**Part 3:** Third, we show that $W(\widehat{S}_{x,y}) \ge x/(1+\rho)^n$. Letting $\widehat{S}_{x,y}^i = \widehat{S}_{x,y} \cap \{i, \ldots, \ell\}$, we use induction over the decision epochs to show that $W(\widehat{S}_{x,y}^i) \ge \widehat{x}_i/(1+\rho)^{\ell+1-i}$, where $\widehat{x}_i$ is as in the candidate construction algorithm. By the discussion in the previous paragraph, we have $\Theta_{\ell+1}^\ell(\widehat{x}_{\ell+1}, \widehat{y}_{\ell+1}) = 0$, in which case, by the boundary condition of the dynamic program in (10), we must have $\widehat{x}_{\ell+1} \le 0$. Also, $\widehat{S}_{x,y}^{\ell+1} = \varnothing$. Therefore, we get $W(\widehat{S}_{x,y}^{\ell+1}) = 0 \ge \widehat{x}_{\ell+1}$, so the result holds for decision epoch $\ell+1$. Assuming that $W(\widehat{S}_{x,y}^{i+1}) \ge \widehat{x}_{i+1}/(1+\rho)^{\ell-i}$, we proceed to showing that

$W(\widehat{S}^i_{x,y}) \geq \widehat{x}_i/(1+\rho)^{\ell+1-i}$. Since $\widehat{x}_{i+1} = \lfloor \widehat{x}_i - v_i\, r_i\, \widehat{u}_i \rfloor$, we have $\widehat{x}_{i+1} \geq \frac{1}{1+\rho}\,(\widehat{x}_i - v_i\, r_i\, \widehat{u}_i)$. Noting that $\lfloor a \rfloor = 0$ for $a < 0$, the last inequality holds when $\widehat{x}_i - v_i\, r_i\, \widehat{u}_i < 0$ as well. The last inequality yields $(1+\rho)\,\widehat{x}_{i+1} + v_i\, r_i\, \widehat{u}_i \geq \widehat{x}_i$. Since $\widehat{S}^i_{x,y} \setminus \{i\} = \widehat{S}^{i+1}_{x,y}$ and $\widehat{u}_i = 1$ if and only if $i \in \widehat{S}^i_{x,y}$, we get

$$\begin{aligned} W(\widehat{S}^i_{x,y}) \;=\; W(\widehat{S}^{i+1}_{x,y}) + v_i\, r_i\, \widehat{u}_i \;&\geq\; \frac{\widehat{x}_{i+1}}{(1+\rho)^{\ell-i}} + v_i\, r_i\, \widehat{u}_i \\ &\geq\; \frac{1}{(1+\rho)^{\ell+1-i}} \Big\{ (1+\rho)\,\widehat{x}_{i+1} + v_i\, r_i\, \widehat{u}_i \Big\} \;\geq\; \frac{\widehat{x}_i}{(1+\rho)^{\ell+1-i}}, \end{aligned}$$

where the first inequality uses the induction hypothesis. Thus, the induction argument is complete. Since $\widehat{S}^{j+1}_{x,y} = \widehat{S}_{x,y}$ and $\widehat{x}_{j+1} = x$, we get $W(\widehat{S}_{x,y}) = W(\widehat{S}^{j+1}_{x,y}) \geq \widehat{x}_{j+1}/(1+\rho)^{\ell-j} \geq x/(1+\rho)^n$. Lastly, we can follow a similar argument to also show that $V(\widehat{S}_{x,y}) \leq (1+\rho)^n\, y$. ■

## Appendix K: Bounds on the State Variable for Constructing Candidate Assortments

To construct the collection of candidate assortments as in (11), we need the value functions $\Theta^\ell_i(x,y)$ through the dynamic program in (10) for $(x,y) \in \textsc{Dom}^2$ such that $x \in [\lfloor w_{\min} \rfloor, \lceil n\, w_{\max} \rceil] \cup \{0\}$, $y \in [\lfloor v_{\min} \rfloor, \lceil n\, v_{\max} \rceil] \cup \{0\}$, and $i \in \mathcal{N}$, $\ell \in \{0, \ldots, n\}$ with $i \leq \ell + 1$. Therefore, the largest values of $x$ and $y$ in the state variable $(x,y)$ are, respectively, $\lceil n\, w_{\max} \rceil$ and $\lceil n\, v_{\max} \rceil$. Since $\lfloor a - b \rfloor \leq a$ and $\lceil a - b \rceil \leq a$ for $a \in \textsc{Dom}$ and $a, b \in \mathbb{R}_+$, from one decision epoch to another, the values of $x$ and $y$ in the state variable $(x,y)$ in (10) go down. Moreover, the boundary condition in (10) depends only on the sign of $x$ and $y$. Thus, if the value of the state variable $x$ goes below $\lfloor w_{\min} \rfloor$ but it is still strictly positive, then without loss of generality, we can bump the value of the state variable $x$ up to $\lfloor w_{\min} \rfloor$, because offering any of the products would immediately turn the value of the state variable $x$ to negative. Similarly, if the value of the state variable $y$ goes below $\lfloor v_{\min} \rfloor$ but is still strictly positive, then we can bump the value of the state variable $y$ up to $\lfloor v_{\min} \rfloor$. Lastly, once the value of $x$ and $y$ in the state variable $(x,y)$ turns negative, we do not need to keep their exact values, since each component of the state variable can only go down and the boundary condition at state $(x,y)$ with $y < 0$ always yields a value function of $+\infty$. Thus, the smallest nonzero values of $x$ and $y$ in the state variable $(x,y) \in \textsc{Dom}^2$ are, respectively, $\lfloor w_{\min} \rfloor$ and $\lfloor v_{\min} \rfloor$.

## Appendix L: Proof of Lemma 5.5

In this section, we give a proof for Lemma 5.5. We need the next intermedia lemma, where we give two monotonicity properties of the value functions $\{\Psi_k(\ell,u,z) : \ell = 0, \ldots, n,\; (u,z) \in \textsc{Dom}^2,\; k \in \mathcal{M}\}$ computed through the dynamic program in (13). Intuitively speaking, the second one of these properties states that we can compensate for an increase by a factor of $(1+\rho)^2$ in the state variable $z$ by an increase by a factor of $1+\rho$ in the state variable $u$. This result becomes critical in ultimately proving the performance guarantee of our FPTAS.

**Lemma L.1** *If the value functions* $\{\Psi_k(j,u,z) : j = 0,\ldots,n,\ (u,z)\in \textsc{Dom}^2,\ k\in\mathcal{M}\}$ *are computed through the dynamic program in* (13)*, then* $\Psi_k(j,u,z)$ *is increasing in* $j$*,* $u$ *and* $z$*. Furthermore, we have* $\Psi_k(j,(1+\rho)\,u,z)\le \Psi_k(j,u,(1+\rho)^2\,z)$*.*

*Proof:* The fact that $\Psi_k(j,u,z)$ is increasing in $j$, $u$ and $z$ follows from an induction argument that is similar to the one in the proof of Lemma J.1. To show that $\Psi_k(j,(1+\rho)\,u,z)\le \Psi_k(j,u,(1+\rho)^2\,z)$, we use induction over the decision epochs. Since $\Psi_{m+1}(j,(1+\rho)\,u,z)$ depends only on the sign of $z$ and the signs of $z$ and $(1+\rho)^2\,z$ are the same, we have $\Psi_{m+1}(j,(1+\rho)\,u,z) = \Psi_{m+1}(j,u,(1+\rho)^2\,z)$. Assuming that $\Psi_{k+1}(j,(1+\rho)\,u,z)\le\Psi_{k+1}(j,u,(1+\rho)^2\,z)$, we proceed to showing that $\Psi_k(j,(1+\rho)\,u,z)\le\Psi_k(j,u,(1+\rho)^2\,z)$. We have $(1+\rho)\,u+V(S)\le(1+\rho)\lceil u+V(S)\rceil$. Since $(1+\rho)\lceil u+V(S)\rceil\in\textsc{Dom}$, the last inequality implies that $\lceil(1+\rho)\,u+V(S)\rceil\le(1+\rho)\lceil u+V(S)\rceil$. In this case, we have

$$\begin{aligned} C(S)+\Psi_{k+1}\bigg(\ell,\,\lceil(1+\rho)\,u+V(S)\rceil,\,\bigg\lceil z-\frac{\lambda_k\,W(S)}{(1+(1+\rho)\,u)\,(1+(1+\rho)\,u+V(S))}\bigg\rceil\bigg)&\\ \le C(S)+\Psi_{k+1}\bigg(\ell,\,(1+\rho)\,\lceil u+V(S)\rceil,\,\bigg\lceil z-\frac{\lambda_k\,W(S)}{(1+(1+\rho)\,u)\,(1+(1+\rho)\,u+V(S))}\bigg\rceil\bigg)&\\ \le C(S)+\Psi_{k+1}\bigg(\ell,\,\lceil u+V(S)\rceil,(1+\rho)^2\bigg\lceil z-\frac{\lambda_k\,W(S)}{(1+(1+\rho)\,u)\,(1+(1+\rho)\,u+V(S))}\bigg\rceil\bigg),& \qquad (28)\end{aligned}$$

where the first inequality follows from the fact that $\Psi_k(\ell,u,z)$ is increasing in $u$ and the second inequality follows from the induction argument.

Note that $(1+\rho)^2\lceil a\rceil\le\lceil(1+\rho)^2\,a\rceil$. If $a<0$, then the inequality is trivial. For $a\ge0$, $a\le\frac{1}{(1+\rho)^2}\lceil(1+\rho)^2a\rceil$. Since $\frac{\lceil(1+\rho)^2a\rceil}{(1+\rho)^2}\in\textsc{Dom}$, the last inequality yields $\lceil a\rceil\le\frac{1}{(1+\rho)^2}\lceil(1+\rho)^2a\rceil$. So,

$$\begin{aligned}(1+\rho)^2&\bigg\lceil z-\frac{\lambda_k\,W(S)}{(1+(1+\rho)u)\,(1+(1+\rho)u+V(S))}\bigg\rceil\\ &\le\bigg\lceil(1+\rho)^2z-\frac{(1+\rho)^2\,\lambda_k\,W(S)}{(1+(1+\rho)u)\,(1+(1+\rho)u+V(S))}\bigg\rceil\le\bigg\lceil(1+\rho)^2z-\frac{\lambda_k\,W(S)}{(1+u)\,(1+u+V(S))}\bigg\rceil,\end{aligned}$$

where the second inequality uses the fact that $\lceil a\rceil$ is increasing in $a$. Note that $\lceil a\rceil$ is increasing in $a$ even with the convention that $\lceil a\rceil=-\infty$ for $a<0$.

Using the chain of inequalities above and the fact that $\Psi_{k+1}(j,u,z)$ is increasing in $z$, we can bound the expression on the right side of (28) as

$$\begin{aligned}C(S)+\Psi_{k+1}\bigg(\ell,\,\lceil u+V(S)\rceil,(1+\rho)^2\bigg\lceil z-\frac{\lambda_k\,W(S)}{(1+(1+\rho)\,u)\,(1+(1+\rho)\,u+V(S))}\bigg\rceil\bigg)&\\ \le C(S)+\Psi_{k+1}\bigg(\ell,\,\lceil u+V(S)\rceil,\,\bigg\lceil(1+\rho)^2z-\frac{\lambda_k\,W(S)}{(1+u)\,(1+u+V(S))}\bigg\rceil\bigg).&\qquad(29)\end{aligned}$$

By (28) and (29), we have $C(S)+\Psi_{k+1}\big(\ell,\,\lceil(1+\rho)\,u+V(S)\rceil,\,\lceil z-\frac{\lambda_k\,W(S)}{(1+(1+\rho)\,u)\,(1+(1+\rho)\,u+V(S))}\rceil\big)\le C(S)+\Psi_{k+1}\big(\ell,\,\lceil u+V(S)\rceil,\lceil(1+\rho)^2z-\frac{\lambda_k\,W(S)}{(1+u)\,(1+u+V(S))}\rceil\big)$ for all $S$ and $\ell$. In this case, minimizing both sides of the inequality over $(\ell, S)$ with $\ell \geq j$ and $S \in \textsc{Cand}(j,\ell)$, the inequality is still preserved, but noting (13), the left side of the inequality gives $\Psi_k(j,(1+\rho)\,u,z)$, whereas the right side gives $\Psi_k(j,u,(1+\rho)^2\,z)$. Thus, we have $\Psi_k(\ell,(1+\rho)\,u,z) \leq \Psi_k(\ell,u,(1+\rho)^2\,z)$. ■

Next, we give a proof for Lemma 5.5.

**Proof of Lemma 5.5:** Let $(\widehat{S}_1,\ldots,\widehat{S}_m)$ be the output of the candidate stitching algorithm, $\widetilde{z}$ be the optimal objective value of problem (9), and $\widehat{z}_{\text{APP}} = \max\{z \in \textsc{Dom} : \Psi_1(0,0,z) \leq b\}$. Our proof proceeds in three parts.

**Part 1:** First, we show that $\sum_{k\in\mathcal{M}} C(\widehat{S}_k) \leq b$. Noting Steps 1 and 2 in the candidate stitching algorithm, along with the dynamic program in (13), we have $\Psi_k(\widehat{j}_k,\widehat{u}_k,\widehat{z}_k) = C(\widehat{S}_k) + \Psi_{k+1}(\widehat{j}_{k+1},\widehat{u}_{k+1},\widehat{z}_{k+1})$. Adding this equality over all $k\in\mathcal{M}$ and noting that we start the candidate stitching algorithm with $\widehat{j}_1 = 0$, $\widehat{u}_1 = 0$ and $\widehat{z}_1 = \widehat{z}_{\text{APP}}$, we obtain $\Psi_1(0,0,\widehat{z}_{\text{APP}}) = \sum_{k\in\mathcal{M}} C(\widehat{S}_k) + \Psi_{m+1}(\widehat{j}_{m+1},\widehat{u}_{m+1},\widehat{z}_{m+1})$. By the initialization of candidate stitching, we have $\Psi_1(0,0,\widehat{z}_{\text{APP}}) \leq b$, in which case, the last equality implies that $\sum_{k\in\mathcal{M}} C(\widehat{S}_k) + \Psi_{m+1}(\widehat{j}_{m+1},\widehat{u}_{m+1},\widehat{z}_{m+1}) \leq b$. By the boundary condition of the dynamic program in (13), $\Psi_{m+1}(\widehat{j}_{m+1},\widehat{u}_{m+1},\widehat{z}_{m+1})$ takes the value $+\infty$ or zero. If we have $\Psi_{m+1}(\widehat{j}_{m+1},\widehat{u}_{m+1},\widehat{z}_{m+1}) = +\infty$, then we get a contradiction to the fact that $\sum_{k\in\mathcal{M}} C(\widehat{S}_k) + \Psi_{m+1}(\widehat{j}_{m+1},\widehat{u}_{m+1},\widehat{z}_{m+1}) \leq b$. Thus, we must have $\Psi_{m+1}(\widehat{j}_{m+1},\widehat{u}_{m+1},\widehat{z}_{m+1}) = 0$, so having $\sum_{k\in\mathcal{M}} C(\widehat{S}_k) + \Psi_{m+1}(\widehat{j}_{m+1},\widehat{u}_{m+1},\widehat{z}_{m+1}) \leq b$ implies that $\sum_{k\in\mathcal{M}} C(\widehat{S}_k) \leq b$.

**Part 2:** Second, we show that $\textsc{Rev}(\widehat{S}_1,\ldots,\widehat{S}_m) \geq \widehat{z}_{\text{APP}}$. By Step 2 of the candidate stitching algorithm, we have $\widehat{u}_{k+1} \geq \widehat{u}_k + V(\widehat{S}_k)$. Adding this inequality over all $k = 1,\ldots,q-1$ and noting that $\widehat{u}_1 = 0$ in the initialization of the algorithm, we get $\widehat{u}_q \geq \sum_{k=1}^{q-1} V(\widehat{S}_k)$. For notational brevity, we let $\widehat{R}_k = \sum_{q=k}^{m} \frac{\lambda_q\,W(\widehat{S}_q)}{(1+\sum_{r=1}^{q-1} V(\widehat{S}_r))\,(1+\sum_{r=1}^{q} V(\widehat{S}_r))}$ with the convention that $\widehat{R}_{m+1} = 0$. We use induction over the stages to show that $\widehat{R}_k \geq \widehat{z}_k$ for all $k = 1,\ldots,m+1$. By the discussion in the previous paragraph, $\Psi_{m+1}(\widehat{j}_{m+1},\widehat{u}_{m+1},\widehat{z}_{m+1}) = 0$, in which case, by the boundary condition in (13), we must have $\widehat{z}_{m+1} \leq 0$. Thus, we have $\widehat{R}_{m+1} = 0 \geq \widehat{z}_{m+1}$. Assuming that $\widehat{R}_{k+1} \geq \widehat{z}_{k+1}$, we proceed to showing that $\widehat{R}_k \geq \widehat{z}_k$. Noting Step 2 of the candidate stitching algorithm and using the induction hypothesis, if $\widehat{z}_{k+1} \geq 0$, then $\widehat{z}_k - \frac{\lambda_k\,W(\widehat{S}_k)}{(1+\widehat{u}_k)\,(1+\widehat{u}_k+V(\widehat{S}_k))} \leq \widehat{z}_{k+1} \leq \widehat{R}_{k+1}$. Also, if $\widehat{z}_{k+1} < 0$, then $\widehat{z}_k - \frac{\lambda_k\,W(\widehat{S}_k)}{(1+\widehat{u}_k)\,(1+\widehat{u}_k+V(\widehat{S}_k))} < 0 \leq \widehat{R}_{k+1}$. So, $\widehat{z}_k \leq \widehat{R}_{k+1} + \frac{\lambda_k\,W(\widehat{S}_k)}{(1+\widehat{u}_k)\,(1+\widehat{u}_k+V(\widehat{S}_k))}$ in both cases. Thus, we get

$$\widehat{R}_k \;=\; \widehat{R}_{k+1} + \frac{\lambda_k\,W(\widehat{S}_k)}{(1+\sum_{q=1}^{k-1} V(\widehat{S}_q))\,(1+\sum_{q=1}^{k} V(\widehat{S}_q))} \;\geq\; \widehat{R}_{k+1} + \frac{\lambda_k\,W(\widehat{S}_k)}{(1+\widehat{u}_k)\,(1+\widehat{u}_k+V(\widehat{S}_k))} \;\geq\; \widehat{z}_k,$$

where we use the fact that $\widehat{u}_k \geq \sum_{q=1}^{k-1} V(\widehat{S}_q)$. The induction argument is complete, in which case, we have $\widehat{R}_1 \geq \widehat{z}_1$. Noting that $\widehat{R}_1 = \textsc{Rev}(\widehat{S}_1,\ldots,\widehat{S}_m)$ and $\widehat{z}_1 = \widehat{z}_{\text{APP}}$, the result follows.

**Part 3:** Third, we show that $\widehat{z}_{\text{APP}} \geq \widetilde{z}/(1+\rho)^{3m+1}$. Let $(\widetilde{S}_1,\ldots,\widetilde{S}_m,\widetilde{j}_1,\ldots,\widetilde{j}_m)$ be an optimal solution to problem (9). For notational brevity, we let $\widetilde{C}_k = \sum_{q=k}^{m} C(\widetilde{S}_q)$, $\widetilde{u}_k = \sum_{q=1}^{k-1} V(\widetilde{S}_q)$ and

$\widetilde{z}_k = \sum_{q=k}^{m} \frac{\lambda_q W(\widetilde{S}_q)}{(1+\widetilde{u}_q)(1+\widetilde{u}_{q+1})}$ with the convention that $\widetilde{C}_{m+1} = 0$, $\widetilde{u}_1 = 0$ and $\widetilde{z}_{m+1} = 0$. We use induction over the stages to show that $\Psi_k(\widetilde{j}_k, \widetilde{u}_k, \widetilde{z}_k/(1+\rho)^{3(m+1-k)}) \le \widetilde{C}_k$. We have $\widetilde{z}_{m+1} = 0$ and $\widetilde{C}_{m+1} = 0$, in which case, noting the boundary condition in (13), we have $\Psi_{m+1}(\widetilde{j}_{m+1}, \widetilde{u}_{m+1}, \widetilde{z}_{m+1}) = \Psi_{m+1}(\widetilde{j}_{m+1}, \widetilde{u}_{m+1}, 0) = 0 = \widetilde{C}_{m+1}$. Assuming that $\Psi_{k+1}(\widetilde{j}_{k+1}, \widetilde{u}_{k+1}, \widetilde{z}_{k+1}/(1+\rho)^{3(m-k)}) \le \widetilde{C}_{k+1}$, we proceed to showing that $\Psi_k(\widetilde{j}_k, \widetilde{u}_k, \widetilde{z}_k/(1+\rho)^{3(m+1-k)}) \le \widetilde{C}_k$. We have

$$
\begin{aligned}
&(1+\rho)^2 \left\lceil \frac{\widetilde{z}_k}{(1+\rho)^{3(m+1-k)}} - \frac{\lambda_k W(\widetilde{S}_k)}{(1+\widetilde{u}_k)(1+\widetilde{u}_{k+1})} \right\rceil \\
&\le (1+\rho)^3 \left( \frac{\widetilde{z}_k}{(1+\rho)^{3(m+1-k)}} - \frac{\lambda_k W(\widetilde{S}_k)}{(1+\widetilde{u}_k)(1+\widetilde{u}_{k+1})} \right) \\
&\overset{(a)}{=} \frac{\widetilde{z}_{k+1}}{(1+\rho)^{3(m-k)}} + \frac{\lambda_k W(\widetilde{S}_k)}{(1+\rho)^{3(m-k)}(1+\widetilde{u}_k)(1+\widetilde{u}_{k+1})} - \frac{(1+\rho)^3 \lambda_k W(\widetilde{S}_k)}{(1+\widetilde{u}_k)(1+\widetilde{u}_{k+1})} \overset{(b)}{\le} \frac{\widetilde{z}_{k+1}}{(1+\rho)^{3(m-k)}}, \qquad (30)
\end{aligned}
$$

where $(a)$ follows from the fact that $\widetilde{z}_k = \widetilde{z}_{k+1} + \frac{\lambda_k W(\widetilde{S}_k)}{(1+\widetilde{u}_k)(1+\widetilde{u}_{k+1})}$ by the definition of $\widetilde{z}_k$ and $(b)$ holds because we have $k \le m$.

In (13), the action $(\widetilde{j}_{k+1}, \widetilde{S}_k)$ is feasible when the state of the system at decision epoch $k$ is $(\widetilde{j}_k, \widetilde{u}_k, \widetilde{z}_k/(1+\rho)^{3(m+1-k)})$. In particular, since $(\widetilde{S}_1, \ldots, \widetilde{S}_m, \widetilde{j}_1, \ldots, \widetilde{j}_m)$ is a feasible solution to problem (9), we have $\widetilde{j}_{k+1} \ge \widetilde{j}_k$ and $\widetilde{S}_k \in \textsc{Cand}(\widetilde{j}_k, \widetilde{j}_{k+1})$. Since, the action $(\widetilde{j}_{k+1}, \widetilde{S}_k)$ is feasible to the minimization problem in (13) with $(j, u, z) = (\widetilde{j}_k, \widetilde{u}_k, \widetilde{z}_k/(1+\rho)^{3(m+1-k)})$, we get

$$
\begin{aligned}
&\Psi_k\left(\widetilde{j}_k, \widetilde{u}_k, \frac{\widetilde{z}_k}{(1+\rho)^{3(m+1-k)}}\right) \\
&\quad \le C(\widetilde{S}_k) + \Psi_{k+1}\left(\widetilde{j}_{k+1}, \lceil \widetilde{u}_k + V(\widetilde{S}_k) \rceil, \left\lceil \frac{\widetilde{z}_k}{(1+\rho)^{3(m+1-k)}} - \frac{\lambda_k W(\widetilde{S}_k)}{(1+\widetilde{u}_k)(1+\widetilde{u}_k + V(\widetilde{S}_k))} \right\rceil \right) \\
&\quad = C(\widetilde{S}_k) + \Psi_{k+1}\left(\widetilde{j}_{k+1}, \lceil \widetilde{u}_{k+1} \rceil, \left\lceil \frac{\widetilde{z}_k}{(1+\rho)^{3(m+1-k)}} - \frac{\lambda_k W(\widetilde{S}_k)}{(1+\widetilde{u}_k)(1+\widetilde{u}_{k+1})} \right\rceil \right) \\
&\quad \overset{(c)}{\le} C(\widetilde{S}_k) + \Psi_{k+1}\left(\widetilde{j}_{k+1}, (1+\rho)\,\widetilde{u}_{k+1}, \left\lceil \frac{\widetilde{z}_k}{(1+\rho)^{3(m+1-k)}} - \frac{\lambda_k W(\widetilde{S}_k)}{(1+\widetilde{u}_k)(1+\widetilde{u}_{k+1})} \right\rceil \right) \\
&\quad \overset{(d)}{\le} C(\widetilde{S}_k) + \Psi_{k+1}\left(\widetilde{j}_{k+1}, \widetilde{u}_{k+1}, (1+\rho)^2 \left\lceil \frac{\widetilde{z}_k}{(1+\rho)^{3(m+1-k)}} - \frac{\lambda_k W(\widetilde{S}_k)}{(1+\widetilde{u}_k)(1+\widetilde{u}_{k+1})} \right\rceil \right) \\
&\quad \overset{(e)}{\le} C(\widetilde{S}_k) + \Psi_{k+1}\left(\widetilde{j}_{k+1}, \widetilde{u}_{k+1}, \frac{\widetilde{z}_{k+1}}{(1+\rho)^{3(m-k)}}\right) \\
&\quad \overset{(f)}{\le} C(\widetilde{S}_k) + \widetilde{C}_{k+1} = \widetilde{C}_k,
\end{aligned}
$$

where $(c)$ follows from the fact that $\Psi_k(\ell, u, z)$ is increasing in $u$ and $(1+\rho)\,u \ge \lceil u \rceil$, $(d)$ follows by the second part of Lemma L.1, $(e)$ follows by noting the fact that $\Psi_k(j, u, z)$ is increasing in $z$ and using the inequality in (30), and $(f)$ is by the induction hypothesis. Thus, the induction argument is complete, so it follows that $\Psi_k(\widetilde{j}_k, \widetilde{u}_k, \widetilde{z}_k/(1+\rho)^{3(m+1-k)}) \le \widetilde{C}_k$.

By the definition of $\widetilde{z}_k$ and $\widetilde{u}_k$, $\widetilde{z}_1 = \sum_{k \in \mathcal{M}} \frac{\lambda_k W(\widetilde{S}_k)}{(1+\widetilde{u}_k)(1+\widetilde{u}_{k+1})} = \sum_{k \in \mathcal{M}} \frac{\lambda_k W(\widetilde{S}_k)}{(1+\sum_{q=1}^{k-1} V(\widetilde{S}_q))(1+\sum_{q=1}^{k} V(\widetilde{S}_q))} = \textsc{Rev}(\widetilde{S}_1, \ldots, \widetilde{S}_m) = \widetilde{z}$, where the last equality uses the fact that $(\widetilde{S}_1, \ldots, \widetilde{S}_m, \widetilde{j}_1, \ldots, \widetilde{j}_m)$ is an optimal solution to problem (9), so $\widetilde{z}_1 = \widetilde{z}$. Thus, using the inequality $\Psi_k(\widetilde{j}_k, \widetilde{u}_k, \widetilde{z}_k/(1+\rho)^{3(m+1-k)}) \le \widetilde{C}_k$ with $k=1$, we get $\Psi_1(\widetilde{j}_1, 0, \widetilde{z}/(1+\rho)^{3m}) \le \widetilde{C}_1 \le b$, where the last inequality uses the fact that $(\widetilde{S}_1,\ldots,\widetilde{S}_m,\widetilde{j}_1,\ldots,\widetilde{j}_m)$ is an optimal solution to problem (9) so that $\widetilde{C}_1 = \sum_{k\in\mathcal{M}} C(\widetilde{S}_k) \le b$. Since $\Psi_k(j,u,z)$ is increasing in $j$ and $z$ by Lemma L.1, we obtain

$$\Psi_1(0,0,\lfloor\widetilde{z}\rfloor/(1+\rho)^{3m}) \le \Psi_1(\widetilde{j}_1, 0, \widetilde{z}/(1+\rho)^{3m}) \le b,$$

which implies that $\lfloor\widetilde{z}\rfloor/(1+\rho)^{3m} \in \textsc{Dom}$ is a feasible solution to the problem $\widehat{z}_{\text{App}} = \max\{z \in \textsc{Dom} : \Psi_1(0,0,z) \le b\}$. Therefore, $\widehat{z}_{\text{App}} \ge \lfloor\widetilde{z}\rfloor/(1+\rho)^{3m} \ge \widetilde{z}/(1+\rho)^{3m+1}$. ■

## Appendix M: Bound on the State Variable for Combining Candidate Assortments

To solve the dynamic program in (13), we argue that that the largest values of $u$ and $z$ that we need to consider in the state variable $(j,u,z)\in\mathcal{N}\times\textsc{Dom}^2$ are, respectively, $\lceil 2n\,v_{\max}\rceil$ and $\lceil nw_{\max}\rceil$. Similarly, the smallest nonzero values of $u$ and $z$ that we need to consider in the state variable $(j,u,z)\in\mathcal{N}\times\textsc{Dom}^2$ are, respectively, $\lfloor v_{\min}\rfloor$ and $\lfloor\lambda_m \frac{w_{\min}}{(1+2n\,v_{\max})^2}\rfloor$. In particular, a simple lemma, given as Lemma M.1 at the end of this section, shows that if we compute $\{\widehat{u}_k : k=1,\ldots,m+1\}$ as $\widehat{u}_{k+1} = \lceil \widehat{u}_k + V(S_k)\rceil$ with $\widehat{u}_1 = 0$ and $S_k\cap S_q = \varnothing$ for all $k\ne q$, then $\widehat{u}_k \le 2n\,v_{\max}$ for all $k\in\mathcal{M}$. Thus, the value of $u$ in the state variable $(j,u,z)$ in the dynamic program in (13) is at most $\lceil 2n\,v_{\max}\rceil$. A strictly positive value of $u$ in the state variable $(j,u,z)$ is at least $\lfloor v_{\min}\rfloor$, as the initial value of this state variable is zero and the preference weight of any product is at least $v_{\min}$. Therefore, the desired upper and lower bounds for $u$ in the state variable $(j,u,z)$ follow.

If the initial state variable $(j,u,z)$ satisfies $z > n\,w_{\max}$, then since $\sum_{k\in\mathcal{M}} W(S_k) \le n\,w_{\max}$ for any $(S_1,\ldots,S_m)$ with $S_k\cap S_q=\varnothing$ for all $k\ne q$, no matter which assortments we offer, the final state variable $(j,u,z)$ satisfies $z>0$, in which case the value function $\Psi_1(0,0,z)$ takes the value $+\infty$. Thus, we do not need to consider the values of $z$ that exceed $n\,w_{\max}$ in the state variable $(j,u,z)$. So, we can assume that the value of $z$ in the state variable $(j,u,z)$ is at most $\lceil n\,w_{\max}\rceil$. Finally, if the value of $z$ in the state variable goes below $\lfloor\lambda_m\,w_{\min}/(1+2n\,v_{\max})^2\rfloor$ but is still strictly positive, then without of loss generality, we can bump the value of $z$ up to $\lfloor\lambda_m\,w_{\min}/(1+2n\,v_{\max})^2\rfloor$, since offering any nonempty candidate assortment would immediately turn the value of the state variable to negative. Therefore, it follows that we can assume that a strictly positive value of $z$ in the state variable $(j,u,z)$ is at least $\lfloor\lambda_m\,w_{\min}/(1+2n\,v_{\max})^2\rfloor$.

We used the next lemma in our discussion earlier in this section. Recall that we choose the accuracy parameter for the geometric grid as $\rho = \frac{1}{8(3m+1)}\epsilon$ for $\epsilon\in(0,1)$, so $\rho\le\frac{1}{2m}$.

**Lemma M.1** *For $\rho\le\frac{1}{2m}$, if we compute $\{\widehat{u}_k : k=1,\ldots,m+1\}$ as $\widehat{u}_{k+1} = \lceil\widehat{u}_k + V(S_k)\rceil$ with $\widehat{u}_1=0$ and $S_k\cap S_q=\varnothing$ for all $k\ne q$, then $\widehat{u}_{m+1}\le 2n\,v_{\max}$.*

*Proof:* We use induction to show that $\widehat{u}_k \le (1+\rho)^{k-1}\,(V(S_1)+\ldots+V(S_{k-1}))$. For $k=1$, we have $\widehat{u}_1=0$. Therefore, the result holds for $k=1$. Assuming that $\widehat{u}_k \le (1+\rho)^{k-1}\,(V(S_1)+\ldots+V(S_{k-1}))$, we proceed to showing that $\widehat{u}_{k+1} \le (1+\rho)^k\,(V(S_1)+\ldots+V(S_k))$. We have

$$\begin{aligned}\widehat{u}_{k+1} \;=\; \lceil \widehat{u}_k + V(S_k)\rceil \;\le\; (1+\rho)\,(\widehat{u}_k + V(S_k))\\ \le\; (1+\rho)\Big((1+\rho)^{k-1}\,(V(S_1)+\ldots+V(S_{k-1})) + V(S_k)\Big)\\ \le\; (1+\rho)^k(V(S_1)+\ldots+V(S_{k-1})+V(S_k)),\end{aligned}$$

which completes the induction argument. Thus, we have $\widehat{u}_{m+1} \le (1+\rho)^m(V(S_1)+\ldots+V(S_m)) \le (1+\rho)^m\,nv_{\max}$. In this case, the result follows because $(1+\rho)^m \le \left(1+\frac{1}{2m}\right)^m \le \exp(1/2) \le 2$. ■

## Appendix N: Assortment Optimization under a Cardinality Constraint

In this section, we consider a version of the CAPACITATED problem, where each product occupies one unit of space. Therefore, we can express the constraint $\sum_{k\in\mathcal{M}} C(S_k) \le b$ as $\sum_{k\in\mathcal{M}} |S_k| \le b$, in which case, we ensure that the total number of products offered over all stages does not exceed $b$. Note that $b$ is an integer without loss of generality. Otherwise, we can round it down to the nearest integer. In this section, we give three results. First, we give an algorithm that finds an exact solution. The running time of this algorithm is polynomial in the number of products, but exponential in the number of stages. Second, we give a pseudo polynomial-time algorithm that finds an exact solution. Assuming that the preference weight of the products take on integer values, the running time of this algorithm is polynomial in the number of products, number of stages, and $v_{\max}$. Third, we give an FPTAS to get a $(1-\epsilon)$-approximate solution, whose running time is polynomial in all of the input parameters and $1/\epsilon$. Next, we go into the details of each of these results, compare them with each other and explain their common components.

First, we show that we can obtain an optimal solution by checking the expected revenue from $O(b^m\,n^{3m-1})$ possible solutions. The running time of this approach is polynomial in the number of products for a fixed number of stages. In general, since each one of as many as $b$ products in an optimal solution can be offered in one of the $m$ stages, the number of all possible solutions to the CAPACITATED problem under a cardinality constraint is $O(\binom{n}{b}\,b^m) = O(n^b\,b^m)$, which is exponential in the number of products even for a fixed number of stages. Second, treating the preference weights as the problem input, if all of the preference weights take on integer values, then we give a pseudo polynomial-time algorithm that obtains an optimal solution in $O(v_{\max} m n^5 b^2)$ operations. This algorithm is based on a dynamic programming formulation of the problem. If the preference weights take on rational values, then we can ensure that the preference weights take on integer values by scaling all of the preference weights by a constant, since the choice probabilities do not change by doing so. Third, by discretizing the state variable in the dynamic program that we use in the pseudo polynomial-time algorithm through a geometric grid, we obtain an FPTAS. Our FPTAS obtains a $(1-\epsilon)$-approximate solution in $O(m^2\, n^4\, b^2 \log(n v_{\max}/v_{\min})/\epsilon)$ operations. All of these three results, the exact algorithm whose running time is exponential in the number of stages, the pseudo polynomial-time algorithm and the FPTAS, are based on constructing a collection of candidate assortments for each stage so that an optimal assortment to offer in a stage lies within this collection. Therefore, we start by focusing on constructing the collections of candidate assortments for the different stages. Throughout this section, when we refer to the Capacitated problem, we refer to the version where each product occupies one unit of space, so we have a constraint on the number of offered products.

**Constructing Collections of Candidate Assortments:**

Note that Lemma 5.1 continues to hold when each product occupies one unit of space. Thus, there exists an optimal solution $(S_1^*, \ldots, S_m^*)$ such that $S_k^* \subseteq \{j_k^*+1, \ldots, j_{k+1}^*\}$ and $S_q^* \cap \{j_k^*+1, \ldots, j_{k+1}^*\} = \varnothing$ for all $q \neq k$, for some $j_1^*, \ldots, j_{m+1}^*$ that satisfy $0 = j_1^* \le j_2^* \le \ldots \le j_m^* \le j_{m+1}^* = n$. To construct the collection of candidate assortments for stage $k$, we proceed under the assumption that we know the values of $j_k^*$, $j_{k+1}^*$, $|\cup_{q\neq k} S_q^*|$ along with $V(S_q^*)$ and $W(S_q^*)$ for all $q \neq k$. In this case, since the assortment that we offer in stage $k$ affects the expected revenue in stages $k, \ldots, m$, we can recover an optimal assortment to offer in stage $k$ by solving

$$\max_{\substack{S \subseteq \{j_k^*+1, \ldots, j_{k+1}^*\},\\ |S| \le b - |\cup_{q\neq k} S_q^*|}} \left\{ \frac{\lambda_k\, W(S)}{(1+\sum_{q=1}^{k-1} V(S_q^*))\,(1+\sum_{q=1}^{k-1} V(S_q^*) + V(S))} \right. \\ \left. + \sum_{\ell=k+1}^{m} \frac{\lambda_\ell\, W(S_\ell^*)}{(1+\sum_{q=1,q\neq k}^{\ell-1} V(S_q^*) + V(S))\,(1+\sum_{q=1,q\neq k}^{\ell} V(S_q^*) + V(S))} \right\},$$

where we use the fact that if we know the value of $|\cup_{q\neq k} S_q^*|$, then we can offer at most $b - |\cup_{q\neq k} S_q^*|$ products in stage $k$.

For notational brevity, we let $b_k^* = b - |\cup_{q\neq k} S_q^*|$, $f_\ell^* = \lambda_\ell\, W(S_\ell^*)/V(S_\ell^*)$ and $u_\ell^* = \sum_{q=1,q\neq k}^{\ell} V(S_q^*)$. We write the objective function of the problem above as

$$\frac{\lambda_k\, W(S)}{(1+u_{k-1}^*)\,(1+u_{k-1}^*+V(S))} + \sum_{\ell=k+1}^{m} f_\ell^* \left\{ \frac{1}{1+u_{\ell-1}^*+V(S)} - \frac{1}{1+u_\ell^*+V(S)} \right\} \\ = \frac{\lambda_k\, W(S)}{(1+u_{k-1}^*)\,(1+u_{k-1}^*+V(S))} + \sum_{\ell=k+1}^{m} (f_\ell^* - f_{\ell+1}^*) \left\{ \frac{1}{1+u_{k-1}^*+V(S)} - \frac{1}{1+u_\ell^*+V(S)} \right\}$$

with the convention that $f_{m+1}^* = 0$. The equality above follows by noting that the sum on the left side of the equality is equivalent to $f_{k+1}^* \frac{1}{1+u_k^*+V(S)} + \sum_{\ell=k+1}^{m} (f_{\ell+1}^* - f_\ell^*) \frac{1}{1+u_\ell^*+V(S)} =$

$\sum_{\ell=k+1}^{m}(f_\ell^* - f_{\ell+1}^*)\frac{1}{1+u_k^*+V(S)} - \sum_{\ell=k+1}^{m}(f_\ell^* - f_{\ell+1}^*)\frac{1}{1+u_\ell^*+V(S)}$, along with the fact that $u_k^* = u_{k-1}^*$. In this case, to recover an optimal assortment to offer in stage $k$, we can solve the problem

$$\max_{\substack{S\subseteq\{j_k^*+1,\ldots,j_{k+1}^*\},\\ |S|\le b_k^*}} \left\{ \frac{\lambda_k W(S)}{(1+u_{k-1}^*)\,(1+u_{k-1}^*+V(S))} + \sum_{\ell=k+1}^{m}(f_\ell^* - f_{\ell+1}^*)\left\{\frac{1}{1+u_{k-1}^*+V(S)} - \frac{1}{1+u_\ell^*+V(S)}\right\}\right\}. \tag{31}$$

In the next lemma, we show that we can efficiently construct a collection of candidate assortments that includes an optimal solution to problem (31) for any values of $\{(f_\ell^*, u_\ell^*) : \ell\in\mathcal{M},\ \ell\neq k\}$.

**Lemma N.1** *Given $j_k^*$, $j_{k+1}^*$ and $b_k^*$, there exists a collection of candidate assortments $\textsc{Cand}_k(j_k^*, j_{k+1}^*, b_k^*)$ with $|\textsc{Cand}_k(j_k^*, j_{k+1}^*, b_k^*)| = O(n^2)$ that includes an optimal solution to problem (31) for any values of $\{(f_\ell^*, u_\ell^*) : \ell\in\mathcal{M},\ \ell\neq k\}$.*

*Proof:* Let $g_\ell^* = (f_\ell^* - f_{\ell+1}^*)\,(u_\ell^* - u_{k-1}^*)$. Multiplying the objective function of problem (31) by the constant $1+u_{k-1}^*$, we can obtain an optimal assortment to offer in stage $k$ by solving

$$\max_{\substack{S\subseteq\{j_k^*+1,\ldots,j_{k+1}^*\},\\ |S|\le b_k^*}} \left\{\frac{1}{1+u_{k-1}^*+V(S)}\left\{\lambda_k W(S) + (1+u_{k-1}^*)\sum_{\ell=k+1}^{m}\frac{g_\ell^*}{1+u_\ell^*+V(S)}\right\}\right\}.$$

Letting $t^*$ be the optimal objective value of the problem above, $t^*$ is no smaller than the objective function of the problem above at each $S$ such that $S\subseteq\{j_k^*+1,\ldots,j_{k+1}^*\}$ and $|S|\le b_k^*$.

Therefore, letting $\mathcal{G} = \{S\subseteq\{j_k^*+1,\ldots,j_{k+1}^*\} : |S|\le b_k^*\}$, we can obtain an optimal solution to the problem above by using the so-called dual formulation, which is given by

$$\begin{aligned}
&\min\left\{t \;:\; t \;\ge\; \frac{1}{1+u_{k-1}^*+V(S)}\left\{\lambda_k W(S) + (1+u_{k-1}^*)\sum_{\ell=k+1}^{m}\frac{g_\ell^*}{1+u_\ell^*+V(S)}\right\}\quad \forall\, S\in\mathcal{G}\right\}\\
&\quad=\min\left\{t \;:\; t \;\ge\; \frac{\lambda_k W(S)}{1+u_{k-1}^*} - \frac{t\,V(S)}{1+u_{k-1}^*} + \sum_{\ell=k+1}^{m}\frac{g_\ell^*}{1+u_\ell^*+V(S)}\quad \forall\, S\in\mathcal{G}\right\}\\
&\quad=\min\left\{t \;:\; t \;\ge\; \max_{S\in\mathcal{G}}\left\{\frac{\lambda_k W(S)}{1+u_{k-1}^*} - \frac{t\,V(S)}{1+u_{k-1}^*} + \sum_{\ell=k+1}^{m}\frac{g_\ell^*}{1+u_\ell^*+V(S)}\right\}\right\}
\end{aligned}$$

where the first equality follows by multiplying both sides of the constraint in the first minimization problem above by $1+u_{k-1}^*+V(S)$ and arranging the terms.

By the discussion so far, if $t^*$ is an optimal solution to the last minimization problem above, then we can recover an optimal assortment to offer in stage $k$ by replacing $t$ in the maximization problem on the right side of the constraint with $t^*$ and solving this maximization problem. Thus, we can obtain an optimal assortment to offer in stage $k$ by solving the problem

$$\max_{S\in\mathcal{G}}\left\{\frac{\lambda_k W(S)}{1+u^*_{k-1}}-\frac{t\,V(S)}{1+u^*_{k-1}}+\sum_{\ell=k+1}^{m}\frac{g^*_\ell}{1+u^*_\ell+V(S)}\right\} \tag{32}$$

for some value of $t$. We will construct a collection of $O(n^2)$ candidate assortments that includes an optimal solution to the problem above for any values of $\{(g^*_\ell,u^*_\ell):\ell\in\mathcal{M},\ \ell\neq k\}$ and $t$.

Note that $\lambda_k W(S)=\lambda_k\sum_{i\in S}r_i\,v_i$ and $t\,V(S)=t\sum_{i\in S}v_i$. In this case, using the decision variables $\boldsymbol{x}=(x_1,\ldots,x_n)$ and noting the definition of $\mathcal{G}$, we write problem (32) equivalently as

$$\max_{\boldsymbol{x}\in\{0,1\}^n}\left\{\frac{\lambda_k}{1+u^*_{k-1}}\sum_{i\in\mathcal{N}}r_i\,v_i\,x_i-\frac{t}{1+u^*_{k-1}}\sum_{i\in\mathcal{N}}v_i\,x_i+\sum_{\ell=k+1}^{m}\frac{g^*_\ell}{1+u^*_\ell+\sum_{i\in\mathcal{N}}v_i\,x_i}\right.$$
$$\left.:\ \sum_{i\in\mathcal{N}}x_i\le b^*,\ x_i=0\ \ \forall\,i\notin\{j^*_k+1,\ldots,j^*_{k+1}\}\right\}. \tag{33}$$

If $g^*_\ell\ge 0$, then the objective function of the problem above is convex in $\boldsymbol{x}$, in which case, an optimal solution occurs at an extreme point, so we can relax $\boldsymbol{x}\in\{0,1\}^n$ to $\boldsymbol{x}\in[0,1]^n$.

Indeed, we have $g^*_\ell\ge 0$. Note that $W(S^*_\ell)/V(S^*_\ell)$ is the weighted average of the revenues of the products in $S^*_\ell$. By Lemma 5.1, the revenues of the products in $S^*_\ell$ are larger than those of the products in $S^*_{\ell+1}$, so we have $W(S^*_\ell)/V(S^*_\ell)\ge W(S^*_{\ell+1})/V(S^*_{\ell+1})$. Furthermore, we have $\lambda_\ell\ge\lambda_{\ell+1}$, in which case, we get $f^*_\ell=\lambda_\ell\,W(S^*_\ell)/V(S^*_\ell)\ge\lambda_{\ell+1}\,W(S^*_{\ell+1})/V(S^*_{\ell+1})=f^*_{\ell+1}$. We have $u^*_\ell\ge u^*_{k-1}$ for all $\ell\ge k+1$ as well, so $g^*_\ell=(f^*_\ell-f^*_{\ell+1})\,(u^*_\ell-u^*_{k-1})\ge 0$. We solve problem (33) with $\boldsymbol{x}\in[0,1]^n$ in two stages. First, intuitively speaking, we guess the value of $\sum_{i\in\mathcal{N}}v_i\,x_i$. Second, we find solution $\boldsymbol{x}$ that maximizes the objective function, while satisfying our guess.

Using $w$ to denote our guess of $\sum_{i\in\mathcal{N}}v_i\,x_i$, we can write the last problem in two stages. In particular, problem (33) is equivalent to the problem

$$\max_{w\in\mathbb{R}_+}\max_{\boldsymbol{x}\in[0,1]^n}\left\{\frac{\lambda_k}{1+u^*_{k-1}}\sum_{i\in\mathcal{N}}r_i\,v_i\,x_i-\frac{t\,w}{1+u^*_{k-1}}+\sum_{\ell=k+1}^{m}\frac{g^*_\ell}{1+u^*_\ell+w}\right.$$
$$\left.:\ \sum_{i\in\mathcal{N}}x_i\le b^*,\quad\sum_{i\in\mathcal{N}}v_i\,x_i\le w,\quad x_i=0\ \ \forall\,i\notin\{j^*_k+1,\ldots,j^*_{k+1}\}\right\}$$
$$=\ \max_{w\in\mathbb{R}_+}\left\{-\frac{t\,w}{1+u^*_{k-1}}+\sum_{\ell=k+1}^{m}\frac{g^*_\ell}{1+u^*_\ell+w}+\frac{\lambda_k}{1+u^*_{k-1}}\max_{\boldsymbol{x}\in[0,1]^n}\left\{\sum_{i\in\mathcal{N}}r_i\,v_i\,x_i\right.\right.$$
$$\left.\left.:\ \sum_{i\in\mathcal{N}}x_i\le b^*,\quad\sum_{i\in\mathcal{N}}v_i\,x_i\le w,\quad x_i=0\ \ \forall\,i\notin\{j^*_k+1,\ldots,j^*_{k+1}\}\right\}\right\}. \tag{34}$$

The first problem above is equivalent to problem (33) since $g^*_\ell\ge 0$, in which case, the objective function of the first problem above is decreasing in $w$. Therefore, $w$ takes the value $\sum_{i\in\mathcal{N}}v_i\,x_i$

in an optimal solution to the first problem above. Considering the second problem above, the inner maximization problem is a linear program with two constraints. We let $Q(w)$ be the optimal objective value and $\boldsymbol{x}^*(w)$ be an optimal solution of this linear program as a function of $w$. It is a standard result in linear programming theory that $Q(w)$ is a piecewise linear function of $w$ with $O(n^2)$ points of nondifferentiability. Furthermore, these points of nondifferentiability for $Q(\cdot)$ do not depend on the values of $\{(f^*_\ell, u^*_\ell) : \ell \in \mathcal{M},\ \ell \neq k\}$ and $t$.

Letting $T = \sum_{i\in\mathcal{N}} v_i$, $\sum_{i\in\mathcal{N}} v_i\, x_i \in [0,T]$. We use $\{\widehat{w}_s : s \in \mathcal{Q}\}$ to denote the points of nondifferentiability of $Q(\cdot)$ with the convention that $0, T \in \mathcal{Q}$. We write problem (34) as

$$\max_{w\in\mathbb{R}_+}\left\{-\frac{t\,w}{1+u^*_{k-1}} + \sum_{\ell=k+1}^{m} \frac{g^*_\ell}{1+u^*_\ell + w} + \frac{\lambda_k}{1+u^*_{k-1}}\, Q(w)\right\}$$
$$= \max_{s\in\mathcal{Q}}\left\{-\frac{t\,\widehat{w}_s}{1+u^*_{k-1}} + \sum_{\ell=k+1}^{m} \frac{g^*_\ell}{1+u^*_\ell + \widehat{w}_s} + \frac{\lambda_k}{1+u^*_{k-1}}\, Q(\widehat{w}_s)\right\},$$

where the equality holds since the objective function of the first problem above is convex in $w$, in which case, an optimal solution must occur at a point of nondifferentiability.

Thus, the collection $\{\boldsymbol{x}^*(\widehat{w}_s) : s \in \mathcal{Q}\}$ with $|\mathcal{Q}| = O(n^2)$ includes an optimal solution to problem (32) for any value of $\{(g^*_\ell, u^*_\ell) : \ell \in \mathcal{M},\ \ell \neq k\}$ and $t$. ■

The main computational effort in constructing the collection of candidate assortments $\textsc{Cand}_k(j_k, j_{k+1}, b_k)$ is to solve a parametric linear program with $O(n^2)$ points of nondifferentiability.

**A Polynomial-Time Algorithm for Fixed Number of Stages:**

We can solve the Capacitated problem as follows. We construct the collection of candidate assortments $\textsc{Cand}_k(j_k, j_{k+1}, b_k)$ for all $j_k, j_{k+1} \in \mathcal{N}$, $b_k \le b$, $k \in \mathcal{M}$. There are $O(n^{m-1})$ choices of $(j_1, \ldots, j_m)$ such that $0 = j_1 \le j_2 \le \ldots \le j_m \le j_{m+1} = n$, as well as $O(b^m)$ choices of $(b_1, \ldots, b_m)$ such that $\sum_{k\in\mathcal{M}} b_k = b$. For each choice of $(j_1, \ldots, j_m)$ and $(b_1, \ldots, b_m)$, since $|\textsc{Cand}_k(j_k, j_{k+1}, b_k)| = O(n^2)$, there are $O(n^{2m})$ ways of picking an assortment from the collection for each stage to construct a possible solution to the Capacitated problem. Thus, we get the next result.

**Theorem N.2** *We can construct a collection of $O(b^m\, n^{3m-1})$ possible solutions to the* Capacitated *problem that is guaranteed to include an optimal solution to this problem. Letting* LP *be the number of operations to solve a parametric linear program with $O(n^2)$ points of nondifferentiability, constructing these solutions requires $O(b\, n^2 \mathsf{LP} + b^m\, n^{3m-1})$ operations.*

**A Pseudo Polynomial-Time Algorithm:**

Noting the objective function of the Capacitated problem, knowing the value of $j^*_k$ such that $S^*_1 \cup \ldots \cup S^*_{k-1} \subseteq \{1, \ldots, j^*_k\}$, the value of $b^*_k$ such that $|S^*_1 \cup \ldots \cup S^*_{k-1}| = b^*_k$, and the value of

$u^*_{k-1}$ such that $\sum_{q=1}^{k-1} V(S^*_q) = u^*_{k-1}$ is enough to compute the optimal expected revenue in stages $k+1,\ldots,m$. Thus, we can solve the CAPACITATED problem by using dynamic programming. The decision epochs are the stages. The state variable at decision epoch $k$ is $(j_k, b_k, u_{k-1})$ such that the assortments $S_1,\ldots,S_{k-1}$ offered in the previous stages satisfy $S_1 \cup \ldots \cup S_{k-1} \subseteq \{1,\ldots,j_k\}$, $|S_1 \cup \ldots \cup S_{k-1}| = b_k$ and $\sum_{q=1}^{k-1} V(S_q) = u_{k-1}$. The action at decision epoch $k$ is the value of $j_{k+1}$ such that the assortment offered in stage $k$ satisfies $S_k \subseteq \{j_k+1,\ldots,j_{k+1}\}$, along with the assortment $S_k \in \cup_{d=0}^{b} \textsc{Cand}_k(j_k, j_{k+1}, d)$ offered in stage $k$. So, we consider the dynamic program

$$J_k(j,c,u) \quad = \max_{\substack{(\ell,S)\,:\,\ell\in\{j,\ldots,n\}\\ S\in\cup_{d=0}^{b}\textsc{Cand}_k(j,\ell,d)}} \left\{ \frac{\lambda_k\, W(S)}{(1+u)\,(1+u+V(S))} + J_{k+1}(\ell\,,\, c+|S|\,,\, u+V(S)) \right\}$$

with the boundary condition that $J_{m+1}(j,c,u) = -\infty$ if $c > b$. If $c \le b$, then $J_{m+1}(j,c,u) = 0$. Solving the dynamic program above requires constructing the collections of candidate assortments a priori.

Since $|\textsc{Cand}_k(j,\ell,d)| = O(n^2)$, at each decision epoch, there are $O(v_{\max}\, b\, n^2)$ possible values of the state variable and $O(bn^3)$ possible values of the action. So, we have the next result.

**Theorem N.3** *Letting* $\mathsf{LP}$ *be as in Theorem N.2, we can obtain an optimal solution to the* CAPACITATED *problem in* $O(b\,n^2\mathsf{LP} + v_{\max}\, m\, n^5\, b^2)$ *operations.*

**Fully Polynomial-Time Approximation Scheme:**

To obtain an FPTAS, we discretize the state variable in the dynamic program that we use to construct a pseudo polynomial-time algorithm. We consider the dynamic program

$$\Psi_k(j,c,u) \quad = \max_{\substack{(\ell,S)\,:\,\ell\in\{j,\ldots,n\}\\ S\in\cup_{d=0}^{b}\textsc{Cand}_k(j,\ell,d)}} \left\{ \frac{\lambda_k\, W(S)}{(1+u)\,(1+u+V(S))} + \Psi_{k+1}(\ell\,,\, c+|S|\,,\, \lceil u+V(S)\rceil) \right\}$$

with the boundary condition that $\Psi_{m+1}(j,c,u) = -\infty$ if $c > b$. If $c \le b$, then $\Psi_{m+1}(j,c,u) = 0$. In the dynamic program above, the roundup operator $\lceil\cdot\rceil$ is as in Section 5.1.

Building on the dynamic program above, we can give an FPTAS by using an argument similar to the one in Section 5. In particular, once we compute the value functions $\{\Psi_k(j,c,u) : j = 0,\ldots,n+1,\ c = 0,\ldots,b,\ u \in \textsc{Dom},\ k \in \mathcal{M}\}$ through the dynamic program above, starting from state $(0,0,0)$, we follow the sequence of optimal state-action pairs to obtain the assortments $(\widehat{S}_1,\ldots,\widehat{S}_m)$ over $m$ stages. We can show that expected revenue from the assortments $(\widehat{S}_1,\ldots,\widehat{S}_m)$ deviate from the optimal expected revenue by at most a factor of $(1+\rho)^{2m}$, where $\rho$ is the size of the geometric grid. For given $\epsilon \in (0,1)$, setting $\rho = \epsilon/(2m)$, we get the next result.

**Theorem N.4** *Letting* $\mathsf{LP}$ *be as in Theorem N.2, for each* $\epsilon \in (0,1)$*, we can obtain a* $(1-\epsilon)$*-approximate solution to the* CAPACITATED *problem in* $O(b\,n^2\mathsf{LP} + \frac{m^2\,n^4\,b^2}{\epsilon}\log(\frac{n\,v_{\max}}{v_{\min}}))$ *operations.*

| $P_0$ | $b=1$ | $b=3$ | $b=5$ | $b=10$ | $b=20$ |
|---|---|---|---|---|---|
| 0.5 | 251.65 | 126.49 | 108.12 | 86.39 | 73.53 |
| 0.7 | 281.75 | 143.99 | 119.20 | 90.04 | 74.66 |
| 0.9 | 278.77 | 143.25 | 117.72 | 85.93 | 70.85 |

**Table EC.3** CPU seconds to estimate the parameters of our choice model.

### Appendix O: Preprocessing the Dataset from Expedia

We explain our approach for preprocessing the dataset from Expedia and give a full description of the columns. The raw dataset includes about ten million rows and 54 columns. In some of the search queries, the price is given as the total amount over the whole length of the stay, whereas in some others, the price is given as the amount per night. It is not possible to reliably tell which approach is used in each search query. To avoid ambiguity, we focused our attention on the search queries for a single night stay and dropped the remaining search queries. Furthermore, we dropped the columns for which the entries are missing for more than 25% of the rows. Considering the remaining columns, we dropped the search queries for which the entries were missing in one of the remaining columns. Lastly, some rows in the dataset included entries that are too large or too small. We dropped all search queries which had an entry in a column that falls outside the 0.5-th and 99.5-th percentile band of the entries in the corresponding column. After preprocessing the dataset, we end up with 595,965 rows representing 34,561 search queries and 15 columns. We describe the first three columns in the main text.

The remaining 12 columns give the star rating and the average review score for the hotel, an indicator for whether the hotel is part of a chain, two location desirability scores, the average price of the hotel over the last trading period, the displayed price, an indicator for whether the hotel is on promotion, the number of days until the day of stay, the number of adults and children in the search query, and an indicator for whether the stay is over the weekend.

### Appendix P: Running Time for Fitting the Choice Models

We used the routine `fmincon` in Matlab to maximize the log-likelihood functions for both choice models under consideration. In Table EC.3, we give the average CPU seconds to estimate the parameters of our multinomial logit model with impatient customers for different values of $P_0$ and $b$, where the average is computed over the 50 datasets. We observe that the CPU seconds to estimate the parameters of our choice model increase as $b$ gets smaller so that we have more stages in the choice model. For a fixed value of $b$, the CPU seconds showed less than 20% variation from one dataset to another. For comparison purposes, we note that the average CPU seconds to estimate the parameters of the standard multinomial logit model is 18.34 seconds.

## Appendix Q: Upper Bound for Joint Pricing and Assortment Optimization

We give an upper bound on the optimal objective value of the PRICING-ASSORTMENT problem. For given assortments $(S_1,\ldots,S_m)$ and no-purchase probabilities $\boldsymbol{q}$ satisfying $q_{k-1}\geq q_k$ for all $k\in\mathcal{M}$, the expected revenue is given by (5). Making its dependence on the assortments explicit, we use $\widehat{\Pi}(\boldsymbol{q},S_1,\ldots,S_m)$ to denote the expected revenue in (5). We construct an upper bound on the expected revenue by treating $\sum_{i\in S_k}e^{\alpha_i}$ in (5) as a continuous quantity.

Specifically, letting $T=\sum_{i\in\mathcal{N}}e^{\alpha_i}$, for each $(S_1,\ldots,S_m)\in\mathcal{F}$, we have $\sum_{k\in\mathcal{M}}\sum_{i\in S_k}e^{\alpha_i}\leq T$. In this case, using the decision variables $\boldsymbol{x}=(x_1,\ldots,x_m)$, by (5), we have

$$\widehat{\Pi}(\boldsymbol{q},S_1,\ldots,S_m)\ \leq\ \frac{1}{\beta}\max_{\boldsymbol{x}\in\mathbb{R}^m_+}\left\{\sum_{k\in\mathcal{M}}\lambda_k\,(q_{k-1}-q_k)\left\{\log x_k-\log\left(\frac{1}{q_k}-\frac{1}{q_{k-1}}\right)\right\}\ :\ \sum_{k\in\mathcal{M}}x_k\leq T\right\},$$

where we use the fact that $(\sum_{i\in S_1}e^{\alpha_i},\ldots,\sum_{i\in S_m}e^{\alpha_i})$ is a feasible but not necessarily an optimal solution to the problem on the right side above.

Using the Lagrange multiplier $\alpha\geq 0$, we relax the constraint $\sum_{k\in\mathcal{M}}x_k\leq T$. Thus, for each $\alpha\geq 0$, we can upper bound the optimal objective value of the problem on the right side above by

$$\frac{1}{\beta}\max_{\boldsymbol{x}\in\mathbb{R}^m_+}\left\{\sum_{k\in\mathcal{M}}\lambda_k\,(q_{k-1}-q_k)\left\{\log x_k-\log\left(\frac{1}{q_k}-\frac{1}{q_{k-1}}\right)\right\}-\sum_{k\in\mathcal{M}}\alpha\,x_k+\alpha\,T\right\}.$$

This problem decomposes by the stages. By the first-order condition for the problem $\max_{x_k\in\mathbb{R}_+}\lambda_k\,(q_{k-1}-q_k)\log x_k-\alpha x_k$, the optimal value of $x_k$ is $\lambda_k\,(q_{k-1}-q_k)/\alpha$.

Plugging the optimal value of $x_k$ into the objective function of the problem presented immediately above, the optimal objective value of the problem is

$$\frac{1}{\beta}\sum_{k\in\mathcal{M}}\lambda_k\,(q_{k-1}-q_k)\left\{\log\frac{\lambda_k\,(q_{k-1}-q_k)}{\alpha}-\log\left(\frac{1}{q_k}-\frac{1}{q_{k-1}}\right)-1\right\}+\frac{\alpha\,T}{\beta}.$$

By the discussion so far, for any $\alpha\geq 0$, the quantity shown above provides an upper bound on $\widehat{\Pi}(\boldsymbol{q},S_1,\ldots,S_m)$, as long as $(S_1,\ldots,S_m)\in\mathcal{F}$ and $q_{k-1}\geq q_k$ for all $k\in\mathcal{M}$. We simplify this quantity by noting that $\log\frac{\lambda_k\,(q_{k-1}-q_k)}{\alpha}-\log(\frac{1}{q_k}-\frac{1}{q_{k-1}})-1=\log(q_{k-1}\,q_k)+\log\frac{\lambda_k}{\alpha}-1$. Thus, we can upper bound the optimal expected revenue in the PRICING-ASSORTMENT problem as

$$\begin{aligned}&\max_{(\boldsymbol{q},S_1,\ldots,S_m)\in\mathbb{R}^m_+\times\mathcal{F}}\left\{\widehat{\Pi}(\boldsymbol{q},S_1,\ldots,S_m)\ :\ q_{k-1}\geq q_k\quad\forall\,k\in\mathcal{M}\right\}\\&\leq\ \frac{1}{\beta}\max_{\boldsymbol{q}\in\mathbb{R}^m_+}\left\{\sum_{k\in\mathcal{M}}\lambda_k\,(q_{k-1}-q_k)\,(\log(q_{k-1}\,q_k)+\log\tfrac{\lambda_k}{\alpha}-1)\ :\ q_{k-1}\geq q_k\ \forall\,k\in\mathcal{M}\right\}+\frac{\alpha\,T}{\beta}.\end{aligned}\tag{35}$$

In the problem shown on the right side above, intuitively speaking, only the no-purchase probabilities in two successive stages $k$ and $k-1$ interact, which indicates that we can solve this problem using dynamic programming. To obtain a dynamic program with a finite number of possible states, we discretize the state variable. It is never optimal to charge negative prices in the joint pricing and assortment problem, since dropping a product with a negative price always increases the expected revenue. Thus, we can lower bound the no-purchase probability in any stage as $q_k(\boldsymbol{\rho}) = 1/(1+\widehat{V}_k(\boldsymbol{\rho})) = 1/(1+\sum_{i\in S_k} e^{\alpha_i - \beta\,\rho_k}) \geq 1/(1+\sum_{i\in\mathcal{N}} e^{\alpha_i}) = \frac{1}{1+T}$. We divide the interval $[\frac{1}{1+T}, 1]$ into $L+1$ subintervals using $\nu_0, \ldots, \nu_{L+1}$ that satisfy $\frac{1}{1+T} = \nu_0 < \nu_1 < \ldots < \nu_L < \nu_{L+1} = 1$. Let $G_k^\alpha(p,r)$ be such that $G_k^\alpha(p,r) \geq \lambda_k\,(q_{k-1}-q_k)\,(\log(q_{k-1}\,q_k) + \log\frac{\lambda_k}{\alpha} - 1)$ all $q_{k-1} \in [\nu_p, \nu_{p+1}]$ and $q_k \in [\nu_r, \nu_{r+1}]$. Coming up with such an upper bound $G_k^\alpha(p,r)$ is not difficult. The first derivatives of $(q_{k-1}-q_k)\log(q_{k-1}\,q_k)$ with respect to $q_{k-1}$ and $q_k$ are, respectively, negative and positive, so $(q_{k-1}-q_k)\log(q_{k-1}\,q_k)$ is decreasing in $q_{k-1}$ and increasing in $q_k$ Thus, if $\log\frac{\lambda_k}{\alpha} - 1 \geq 0$, then we set $G_k^\alpha(p,r) = \lambda_k\,(\nu_p - \nu_{r+1})\log(\nu_p\,\nu_{r+1}) + \lambda_k\,(\nu_{p+1} - \nu_r)\,(\log\frac{\lambda_k}{\alpha} - 1)$. If $\log\frac{\lambda_k}{\alpha} - 1 < 0$, then we set $G_k^\alpha(p,r) = \lambda_k\,(\nu_p - \nu_{r+1})\log(\nu_p\,\nu_{r+1}) + \lambda_k\,(\nu_p - \nu_{r+1})\,(\log\frac{\lambda_k}{\alpha} - 1)$. In our dynamic program, we focus on the possible intervals that can include the no-purchase probabilities $(q_1, \ldots, q_m)$. The decision epochs are the stages. The state at decision epoch $k$ is the interval that includes $q_{k-1}$. The action at decision epoch $k$ is the interval that includes $q_k$. Since the no-purchase probabilities in problem (35) satisfy $q_{k-1} \geq q_k$, we impose the constraint that the interval that includes $q_k$ should not lie to the right of the interval that includes $q_{k-1}$. Thus, we consider the dynamic program

$$J_k^\alpha(p) \;=\; \max_{r\in\{0,\ldots,p\}} \Big\{ G_k^\alpha(p,r) + J_{k+1}^\alpha(r) \Big\} \tag{36}$$

with the boundary condition that $J_{m+1}^\alpha(p) = \alpha T$. Next, we show that $\frac{1}{\beta}\,J_1^\alpha(L)$ is an upper bound on the optimal objective value of the Pricing-Assortment problem.

**Proposition Q.1** *For each $\alpha \geq 0$, $\frac{1}{\beta}\,J_1^\alpha(L)$ is an upper bound on the optimal expected revenue of the* Pricing-Assortment *problem.*

*Proof:* By the discussion earlier in this section, it is enough to show that $J_1^\alpha(L)$ is an upper bound on the optimal objective value of the problem

$$\max_{\boldsymbol{q}\in\mathbb{R}_+^m} \left\{ \sum_{k\in\mathcal{M}} \lambda_k\,(q_{k-1}-q_k)\,(\log(q_{k-1}\,q_k) + \log\tfrac{\lambda_k}{\alpha} - 1) \;:\; q_{k-1} \geq q_k \;\forall\, k \in \mathcal{M} \right\} + \alpha T.$$

Let $\boldsymbol{q}^*$ be an optimal solution to the problem above and $p_k^*$ be such that $q_k^* \in [\nu_{p_k^*}, \nu_{p_k^*+1}]$. Since $q_{k-1}^* \geq q_k^*$, we have $p_{k-1}^* \geq p_k^*$. Also, since $q_0^* = 1$, we have $p_0^* = L$.

Let $Z_k = \sum_{\ell=k}^m \lambda_\ell\,(q_{\ell-1}^* - q_\ell^*)\,(\log(q_{\ell-1}^*\,q_\ell^*) + \log\frac{\lambda_\ell}{\alpha} - 1) + \alpha T$ with $Z_{m+1} = \alpha T$. We use induction over the stages to show that $J_k^\alpha(p_{k-1}^*) \geq Z_k$. Since $J_{m+1}^\alpha(p) = \alpha T$, the result holds for stage $m+1$. Assuming that $J_{k+1}^\alpha(p_k^*) \geq Z_{k+1}$, we proceed to showing that $J_k^\alpha(p_{k-1}^*) \geq Z_k$. Since $p_k^* \leq p_{k-1}^*$, when computing $J_k^\alpha(p_{k-1}^*)$ though the dynamic program in (36), $p_k^*$ is a feasible but not necessarily an optimal decision. Therefore, we get

$$\begin{aligned} J_k^\alpha(p_{k-1}^*) &\geq G_k^\alpha(p_{k-1}^*, p_k^*) + J_{k+1}^\alpha(p_k^*) \\ &\geq \lambda_k\,(q_{k-1}^* - q_k^*)\,(\log(q_{k-1}^*\, q_k^*) + \log\tfrac{\lambda_k}{\alpha} - 1) + Z_{k+1} \;=\; Z_k, \end{aligned}$$

where the second inequality uses the fact that $J_{k+1}^\alpha(p_k^*) \geq Z_{k+1}$ by the induction hypothesis, along with the fact that $G_k^\alpha(p,r) \geq \lambda_k\,(q_{k-1} - q_k)\,(\log(q_{k-1}\,q_k) + \log\frac{\lambda_k}{\alpha} - 1)$ for all $q_{k-1} \in [\nu_p, \nu_{p+1}]$ and $q_k \in [\nu_r, \nu_{r+1}]$ by the definition of $G_k^\alpha(p,r)$, as well as noting that $q_{k-1}^* \in [\nu_{p_{k-1}^*}, \nu_{p_{k-1}^*+1}]$ and $q_k^* \in [\nu_{p_k^*}, \nu_{p_k^*+1}]$. Thus, the induction argument is complete. Therefore, we have $J_1^\alpha(L) = J_1^\alpha(p_0^*) \geq Z_1$, in which case, the desired result follows by observing that $Z_1$ is the optimal objective value of the problem at the beginning of the proof. ■

By the proposition above, the quantity $\frac{1}{\beta}\,J_1^\alpha(L)$ is an upper bound on the optimal objective value of the PRICING-ASSORTMENT problem for any $\alpha \geq 0$, so computing $\frac{1}{\beta}\,J_1^\alpha(L)$ for any $\alpha \geq 0$ provides an upper bound on the optimal expected revenue. To get a reasonably tight upper bound on the optimal expected revenue, we use a few iterations of the golden ratio search to find an approximate solution to the problem $\frac{1}{\beta}\min_{\alpha\geq 0} J_1^\alpha(L)$. This approach amounts to computing $J_1^\alpha(L)$ for a few different values of $\alpha$. To obtain the results reported in our computational experiments in Section 6.2, we choose the end points $\nu_0, \ldots, \nu_{L+1}$ of the intervals $\{[\nu_p, \nu_{p+1}] : p = 0, \ldots, L\}$ such that $\nu_{p+1} - \nu_p$ is approximately 0.001 for all $p = 0, \ldots, L$.

## Appendix R: Upper Bound under a Space Constraint

To obtain an upper bound on the optimal expected revenue in the assortment problem under a space constraint, we consider the linear program

$$\text{CAP}(j,\ell,x,y) \;=\; \min_{\boldsymbol{w}\in[0,1]^{\ell-j}} \left\{ \sum_{i=j+1}^{\ell} c_i\, w_i \;:\; \sum_{i=j+1}^{\ell} v_i\, r_i\, w_i \geq x, \quad \sum_{i=j+1}^{\ell} v_i\, w_i \leq y \right\}. \tag{37}$$

If we impose the constraints $\boldsymbol{w} \in \{0,1\}^{\ell-j}$ in the problem above and drop the round down and up operators in (10), then the problem above and (10) solve the same knapsack problem.

If the problem above is infeasible, then we set $\text{CAP}(j,\ell,x,y) = +\infty$. Note that $W(S) \leq n\, w_{\max}$ and $V(S) \leq n\, v_{\max}$ for all $S \subseteq \mathcal{N}$. Also, letting $r_{\max} = \max\{r_i : i \in \mathcal{N}\}$, we have $\Pi(S_1, \ldots, S_m) \leq r_{\max}$ for all $(S_1, \ldots, S_m) \in \mathcal{F}$. Letting $B = \max\{n\, w_{\max}, n\, v_{\max}, r_{\max}\}$, we divide the interval $[0, B]$ into $L+1$ subintervals using $\nu_0, \ldots, \nu_{L+1}$ that satisfy $0 = \nu_0 < \nu_1 < \ldots < \nu_L < \nu_{L+1} = B$. Throughout this section, we define the round down operator "$\lfloor \cdot \rfloor$" that rounds its argument down to the closest point in $\{\nu_p : p = 0, \ldots, L+1\}$ when the argument is positive. That is, if $a \geq 0$, then $\lfloor a \rfloor = \max\{\nu_r : \nu_r \leq a,\ r = 0, \ldots, L+1\}$. If $a < 0$, then $\lfloor a \rfloor = -\infty$. We consider the dynamic program

$$\overline{\Psi}_k(j,u,z) = \min_{\substack{(\ell,p,r)\,:\,\ell \in \{j,\ldots,n\},\\ p \in \{0,\ldots,L\},\\ r \in \{1,\ldots,L+1\}}} \Bigg\{ \textsc{Cap}(j,\ell,\nu_p,\nu_r) + \overline{\Psi}_{k+1}\bigg(\ell,\, \lfloor u + \nu_{r-1} \rfloor,\, \bigg\lfloor z - \frac{\lambda_k\, \nu_{p+1}}{(1+u)\,(1+u+\nu_{r-1})} \bigg\rfloor \bigg) \Bigg\} \tag{38}$$

with the boundary condition that $\overline{\Psi}_{m+1}(j,u,z) = 0$ if $z \leq 0$. Otherwise, we have $\overline{\Psi}_{m+1}(j,u,z) = +\infty$. Note that the dynamic program above is analogous to the one in (13).

In the next proposition, we show that we obtain an upper bound on the optimal expected revenue in the Capacitated problem by solving the dynamic program above.

**Proposition R.1** *Letting $\overline{z}_{\text{App}} = \max\{z \in \mathbb{R}_+ : \overline{\Psi}_1(0,0,z) \leq b\}$, $\overline{z}$ is an upper bound on the optimal expected revenue in the* Capacitated *problem.*

*Proof:* Using an induction argument that is similar to the one in the proof of Lemma J.1, it follows that $\overline{\Psi}_k(j,u,z)$ is increasing in $j$, $u$ and $z$. Let $(S_1^*, \ldots, S_m^*)$ be an optimal solution to the Capacitated problem. By Lemma 5.1, there exist $j_1^*, \ldots, j_{m+1}^*$ satisfying $0 = j_1^* \leq j_2^* \leq \ldots \leq j_m^* \leq j_{m+1}^* = n$ such that $S_k^* \subseteq \{j_k^*+1, \ldots, j_{k+1}^*\}$. Also, let $p_k^* = 0, \ldots, L$ and $r_k^* = 1, \ldots, L+1$ be such that $W(S_k^*) \in [\nu_{p_k^*}, \nu_{p_k^*+1}]$ and $V(S_k^*) \in [\nu_{r_k^*-1}, \nu_{r_k^*}]$. Consider solving problem (37) with $j = j_k^*$, $\ell = j_{k+1}^*$, $x = \nu_{p_k^*}$ and $y = \nu_{r_k^*}$. Setting $w_i = 1$ if $i \in S_k$ and $w_i = 0$ if $i \notin S_k$ provides a feasible solution to this problem with the objective value $C(S_k^*)$. Thus, $\textsc{Cap}(j_k^*, j_{k+1}^*, \nu_{p_k^*}, \nu_{r_k^*}) \leq C(S_k^*)$.

For notational brevity, we let $C_k^* = \sum_{q=k}^m C(S_q^*)$, $u_k^* = \sum_{q=1}^{k-1} V(S_q^*)$ and $z_k^* = \sum_{q=k}^m \frac{\lambda_q\, W(S_q^*)}{(1+u_q^*)\,(1+u_{q+1}^*)}$ with the convention that $C_{m+1}^* = 0$, $u_1^* = 0$ and $z_{m+1}^* = 0$. Observe that $z_1^*$ corresponds to the optimal objective value of the Capacitated problem. We use induction over the stages to show that $\overline{\Psi}_k(j_k^*, u_k^*, z_k^*) \leq C_k^*$. Since $z_{m+1}^* = 0$, we have $\overline{\Psi}_{m+1}(j_{m+1}^*, u_{m+1}^*, z_{m+1}^*) = 0 = C_{m+1}^*$. Therefore, the result holds for the base case. Assuming that $\overline{\Psi}_{k+1}(j_{k+1}^*, u_{k+1}^*, z_{k+1}^*) \leq C_{k+1}^*$, we proceed to showing that $\overline{\Psi}_k(j_k^*, u_k^*, z_k^*) \leq C_k^*$. Using the fact that $\overline{\Psi}(j,u,z)$ is increasing in $u$ and $z$ along with $\lfloor a \rfloor \leq a$ and noting that $W(S_k^*) \leq \nu_{p_k^*+1}$ and $V(S_k^*) \geq \nu_{r_k^*-1}$, we have

$$\begin{aligned} &\overline{\Psi}_{k+1}\bigg(j_{k+1}^*,\, \lfloor u_k^* + \nu_{r_k^*-1} \rfloor,\, \bigg\lfloor z_k^* - \frac{\lambda_k\, \nu_{p_k^*+1}}{(1+u_k^*)\,(1+u_k^*+\nu_{r_k^*-1})} \bigg\rfloor \bigg) \\ &\qquad \leq\ \overline{\Psi}_{k+1}\bigg(j_{k+1}^*,\, u_k^* + V(S_k^*),\, z_k^* - \frac{\lambda_k\, W(S_k^*)}{(1+u_k^*)\,(1+u_k^*+V(S_k^*))}\bigg) \ =\ \overline{\Psi}_{k+1}(j_{k+1}^*,\, u_{k+1}^*,\, z_{k+1}^*), \end{aligned}$$

where the equality above uses the definition of $u_k^*$ and $z_k^*$. Consider computing $\overline{\Psi}_k(j_k^*, u_k^*, z_k^*)$ through the dynamic program in (38). Since $j_{k+1}^* \geq j_k^*$, the solution $(j_{k+1}^*, p_k^*, r_k^*)$ is feasible but not necessarily optimal to the minimization problem on the right side of (38) when we solve this problem with $(j,u,z)=(j^*_k,u^*_k,z^*_k)$. Therefore, we have the chain of inequalities

$$\begin{aligned}\overline{\Psi}_k(j^*_k,u^*_k,z^*_k) \;&\leq\; \textsc{Cap}(j^*_k,j^*_{k+1},\nu_{p^*_k},\nu_{r^*_k})+\overline{\Psi}_{k+1}\left(j^*_{k+1}\,,\,\lfloor u^*_k+\nu_{r^*_k-1}\rfloor\,,\,\left\lfloor z^*_k-\frac{\lambda_k\,\nu_{p^*_k+1}}{(1+u^*_k)\,(1+u^*_k+\nu_{r^*_k-1})}\right\rfloor\right)\\ &\overset{(a)}{\leq}\; C(S^*_k)+\overline{\Psi}_{k+1}(j^*_{k+1}\,,\,u^*_{k+1}\,,\,z^*_{k+1})\\ &\overset{(b)}{\leq}\; C(S^*_k)+C^*_{k+1}\;\overset{(c)}{=}\;C^*_k,\end{aligned}$$

where $(a)$ follows from the inequality that we give earlier in this paragraph and the fact that $\textsc{Cap}(j^*_k,j^*_{k+1},\nu_{p^*_k},\nu_{r^*_k})\leq C(S^*_k)$, $(b)$ uses the induction hypothesis and $(c)$ is by the definition of $C^*_k$. Thus, the induction argument is complete, in which case, noting that $j^*_1=0$ and $u^*_1=0$, we obtain $\overline{\Psi}_1(0,0,z^*_1)\leq C^*_1=\sum_{k\in\mathcal{M}}C_k(S^*_k)\leq b$, where the last inequality uses the fact that $(S^*_1,\ldots,S^*_m)$ is a feasible solution to the Capacitated problem. Therefore, $z^*_1$ is a feasible solution to the problem $\overline{z}_{\text{App}}=\max\{z\in\mathbb{R}_+:\overline{\Psi}_1(0,0,z)\leq b\}$, which implies that the optimal objective value of this problem is at least as large as $z^*_1$. In other words, we have $\overline{z}_{\text{App}}\geq z^*_1$. In this case, the result follows by noting that $z^*_1$ is the optimal objective value of the Capacitated problem. ■

Note that the upper bound in the proposition above holds for any choice of $\nu_0,\ldots,\nu_{L+1}$ that satisfy $0=\nu_0<\nu_1<\ldots<\nu_L<\nu_{L+1}=B$.