Dynamic Assortment Optimization for Reusable Products with Random Usage Durations

Paat Rusmevichientong
Marshall School of Business, University of Southern California, Los Angeles, CA 90089, rusmevic@marshall.usc.edu

Mika Sumida and Huseyin Topaloglu
School of Operations Research and Information Engineering, Cornell Tech, NY, NY 10044, {ms3268,ht88}@cornell.edu

December 9, 2017

We consider multi-product dynamic assortment problems with reusable products, in which each arriving customer chooses a product within an offered assortment, uses the product for a random duration of time, and returns the product back to the firm to be used by other customers. The goal is to find a policy for deciding on the assortment to offer to each customer so that the total expected revenue over a finite selling horizon is maximized. The dynamic programming formulation of this problem requires a high-dimensional state variable that keeps track of the on-hand product inventories, as well as the products that are currently in use. We present a tractable approach to compute a policy that is guaranteed to obtain at least 50% of the optimal total expected revenue. This policy is based on constructing linear approximations to the optimal value functions. The approximations are computed through an efficient backward recursion over the time periods in the selling horizon. When the usage duration is infinite or follows a negative binomial distribution, we also discuss how to efficiently perform rollout on a simple static policy. Performing rollout corresponds to using separable and nonlinear value function approximations. The resulting policy is also guaranteed to obtain at least 50% of the optimal total expected revenue. The special case of our model with infinite usage durations captures the revenue management problem under customer choice over parallel flight legs operating between the same origin-destination pair. We provide computational experiments based on simulated data for parallel flights and real parking transaction data for the city of Seattle. Our computational experiments demonstrate that the practical performance of our policies is substantially better than their performance guarantees and performing rollout yields noticeable improvements.

Key words: dynamic assortment optimization, reusable products, and choice modeling

1. Introduction

Revenue management problems focus on making capacity allocation decisions for limited inventories of products over a finite selling horizon. These problems find applications in areas as diverse as airline, hotel, electric power, health care, consumer credit, cruise line, and advertising capacity management (Ozer and Phillips 2012). The dynamic programming formulations of revenue management problems are generally intractable because they require high-dimensional state variables that keep track of the remaining inventory of each product. Thus, computing the optimal policy is computationally difficult and researchers have focused on approximate policies.

In traditional application areas of revenue management problems, the customers purchase the products for final consumption. However, some emerging industries focus on renting out computing capacity, physical storage space, and fashion items. In these industries, each customer requests a product, uses the product for a possibly random duration of time, and returns the product back to the firm, at which point the product can be used by other customers. When making capacity allocation decisions in such environments, the firms must consider the inventories of products on-hand, along with the products that are currently in use.

In this paper, we consider dynamic assortment problems with reusable products. Customers randomly arrive into the system. We offer an assortment of products to each arriving customer. The customer decides to leave the system or chooses a product from the offered assortment. If a product is chosen, then the customer uses the product for a random duration of time. After a usage duration, the customer returns the product back. Our goal is to find a policy for deciding on the assortment to offer to each arriving customer so that the total expected revenue over a finite selling horizon is maximized. The dynamic programming formulation of this problem requires a high-dimensional state variable that keeps track of the remaining product inventories, as well as the products that are currently in use by the customers. Therefore, computing the optimal policy is computationally difficult. We propose tractable policies that provide performance guarantees.

Main Contributions: Working with a general dynamic assortment optimization problem, we construct linear approximations to the optimal value functions. We establish a half-approximate performance guarantee for the greedy policy with respect to the linear approximations, meaning that the total expected revenue of the greedy policy is guaranteed to be at least 50% of the optimal total expected revenue. For special cases of random usage durations, corresponding to negative binomial and infinite usage durations, we can perform rollout on a static policy to obtain separable and nonlinear approximations to the optimal value functions. The nonlinear value function approximations also yield a policy with the half-approximate performance guarantee. Our computational experiments indicate that the practical performance of our policies are remarkably good, exceeding the theoretical guarantees by substantial margins.

We proceed to elaborating on each one of these contributions in detail. In Section 2, we formulate a rather general dynamic assortment problem with random usage durations. In particular, our formulation allows for a general class of choice models for describing the choice process of the customers and arbitrary distributions for the random usage durations. We use a dynamic programming formulation whose state variable keeps track of the numbers of time periods that each unit of a product has been in use, as well as the numbers of units of products that are available on-hand. Therefore, the number of dimensions of the state variable can be rather large. Working with such a dynamic programming formulation, we construct tractable approximations to the optimal value functions that yield policies with a half-approximate performance guarantee.

In Section 3, we make algorithmic contributions by proposing a new method to construct linear value function approximations. Our approach uses an efficient backward recursion over the time periods. At each time period, we solve a myopic static assortment optimization problem, where we adjust the product revenues by time-dependent constants computed from the recursion and find an assortment of products that maximizes the expected adjusted revenue (Section 3.1). Such myopic static assortment optimization problems are tractable under a variety of choice models. We show that the greedy policy with respect to our linear value function approximations is guaranteed to obtain at least 50% of the optimal total expected revenue (Section 3.2).

Our proposed method has two novel aspects. First, the half-approximation guarantee provided by our linear value function approximations appears to be the first of its kind. A common approach for constructing linear value function approximations is the approximate linear programming approach, which uses the linear programming representation of the dynamic programming formulation of the dynamic assortment problem (Adelman 2007, Zhang and Adelman 2009). Existing results show that this approach yields upper bounds on the optimal value functions, but do not give performance guarantees for the resulting policies. Our approach comes with a performance guarantee. Second, our construction of the approximations is more tractable than the existing methods. The number of constraints in the linear program used in the approximate linear programming approach grows linearly with the size of the state and action spaces in the dynamic programming formulation of the dynamic assortment problem. Thus, this linear program can get quite large and it is often solved by using column generation on its dual. By contrast, our approach does not solve any linear program, but computes the coefficients of the linear value function approximations one time period at a time. Interestingly, the proof for our performance guarantee considers the linear program used in the approximate linear programming approach. However, we use this linear program only to facilitate our proof, but not to construct our linear value function approximations.

In Section 4, we focus on constructing separable and nonlinear approximations to the optimal value functions. We start with a static policy that simply offers the same assortment at a particular time period, irrespective of the state of the system (Section 4.1). We perform rollout on the static policy to obtain separable and nonlinear value function approximations, by treating each product separately. The policy obtained through the rollout approach is also guaranteed to yield at least 50% of the optimal total expected revenue (Section 4.2). We show that we can efficiently perform rollout on the static policy when the usage duration follows a negative binomial distribution (Section 4.3) or when the usage duration is infinite (Section 4.4). The case with infinite usage durations corresponds to the situation where the customers purchase the product outright, rather than renting. As our rollout approach computes separable and nonlinear value function approximations, it is similar to the decomposition techniques used in the literature to decompose the dynamic programming formulation of multi-product revenue management problems by treating each product separately (Liu and van Ryzin 2008, Zhang and Adelman 2009, Kunnumkal and Topaloglu 2010). To our knowledge, the decomposition techniques in the literature have no performance guarantee, whereas our approach provides a half-approximate performance guarantee.

Our dynamic assortment problem with infinite usage durations corresponds to the choice-based revenue management problem over parallel flight legs operating between the same origin-destination pair (Zhang and Cooper 2005, Liu and van Ryzin 2008, Dai et al. 2014). Thus, this special case represents an important problem class that has been studied in the literature. In our computational experiments, in addition to the revenue management problem over parallel flight legs, we work on the problem of dynamically adjusting the menu of offered prices for parking spaces. We treat each parking space as a reusable product with a random usage duration. We use actual parking transaction data from the city of Seattle to estimate the model parameters. Our computational experiments demonstrate that our policies perform well and performing rollout yields noticeable improvements in terms of policy performance.

Literature Review: There is limited work on revenue management with random usage durations. Levi and Radovanovic (2010) study a model that assumes independent demands across products, without any choice behavior for the customers. The authors establish a performance guarantee for a static policy that does not consider the real-time state of the system. They focus on the infinite horizon setting with a long-run average revenue criterion, which allows them to establish their performance guarantees by characterizing the so-called blocking probabilities. By contrast, we develop policies that consider the real-time state of the system, and focus on the finite horizon setting, which makes the blocking probabilities hard to characterize. Recently, Owen and Simchi-Levi (2017) extend the work of Levi and Radovanovic (2010) to model the choice behavior, but they consider only static policies and work with the long-run average revenue criterion.

Motivated by the online resource allocation setting, Stein et al. (2016), Wang et al. (2016) and Gallego et al. (2016) consider a problem related to our dynamic assortment problem. This stream of work focuses on allocating products to customers arriving over time. In relation to this literature, we make two contributions. Our problem setup involves reusable resources with random usage durations. Both reusability and random usage durations bring non-trivial challenges and it is immediately not clear how to extend the existing work in this direction. More importantly, the existing work requires solving linear programs, whose sizes can be exponentially large in the number of products that can be offered to the customers, when the customers choose among products. We avoid such large-scale linear programs in our approach altogether.

Finally, our work is related to revenue management problems under customer choice. Zhang and Cooper (2005) compute upper bounds on the optimal value functions for the choice-based parallel flights problem, and use these bounds to derive a booking limit policy. Gallego et al. (2004) focus on network revenue management problems and study static policies extracted from a deterministic linear program. Adelman (2007) constructs linear approximations in network revenue management problems. To choose the parameters of the approximations, the author uses the linear programming representation of the dynamic programming formulation for the network revenue management problem. This linear program has a large number of constraints, so it is solved by using column generation on its dual. Liu and van Ryzin (2008) build on the deterministic linear program proposed by Gallego et al. (2004) to develop dynamic programming decomposition methods for decomposing the dynamic programming formulation of the network revenue management problem by the flight legs. Zhang and Adelman (2009) construct linear approximations in choice-based network revenue management problems, and show that the dynamic programming decomposition idea provides upper bounds on the optimal value functions. Kunnumkal and Topaloglu (2010) discuss another dynamic programming decomposition method that is based on allocating the revenue from an itinerary over the different flight legs that it uses. Tong and Topaloglu (2013) and Vossen and Zhang (2015) show that the number of constraints in the linear program proposed by Adelman (2007) can be reduced a priori, eliminating the need to use column generation on the dual of the linear program. Unlike our performance guarantees, the approaches in this stream of literature do not provide any constant-factor performance guarantee on the resulting policy.

Organization: In Section 2, we formulate the dynamic assortment problem with reusable products and random usage durations as a dynamic program. In Section 3, we design a policy that is guaranteed to obtain at least 50% of the optimal total expected revenue. This policy is based on building linear approximations to the optimal value functions. Then, in Section 4, we focus on a static policy and perform rollout on the static policy. This approach yields separable and nonlinear approximations to the optimal value functions, and the resulting policy is also guaranteed to yield at least 50% of the optimal total expected revenue. In Section 5, we extend our approach to the case in which we have heterogeneous customer types, we make pricing rather than assortment decisions, and we can solve the assortment problems only approximately. In Section 6, we give computational experiments within the settings of making revenue management decisions for parallel flight legs and making pricing decisions for parking spaces. Conclusions are discussed in Section 7.

2. Problem Formulation

We have a set of products with limited inventories. At each time period in the selling horizon, we decide on the set of products to offer. A customer arriving into the system either chooses to rent one of the offered products or decides to leave the system without renting any of the products. We capture the choice process of the customers through a discrete choice model. If the customer chooses to rent one of the offered products, then she uses the product for a random duration of time by paying an upfront fee and a per-period rental fee for each time period that she uses the product. After using the product for a random duration of time, the customer returns the product, at which point, we can rent the product to another customer. Our goal is to find a policy for maximizing the total expected revenue over the selling horizon. We proceed to describe the primitives of the problem, followed by the state, and the transition dynamics. We conclude this section by giving a dynamic programming formulation of the problem.

Problem Primitives: We have n products indexed by $\mathcal{N} = \{1, \ldots, n\}$. For each product $i \in \mathcal{N}$, let $C_i \in \mathbb{Z}_+$ denote its initial inventory level. There are T time periods in the selling horizon indexed by $\mathcal{T} = \{1, \ldots, T\}$. Each time period corresponds to a small interval of time and there is exactly one customer arrival at each time period. It is not difficult to extend our model to the case where there is at most one customer arrival at each time period. A customer chooses among the offered products according to a discrete choice model $\{\phi_i(S) : i \in \mathcal{N},\ S \subseteq \mathcal{N}\}$, where $\phi_i(S)$ is the probability that the customer chooses product i when we offer the subset S of products. If a customer chooses to rent product i, then she pays two types of fees. She immediately pays a one-time upfront fee of r_i, and a fee of π_i for each time period she rents the product. Depending on the specific application under consideration, one of the fees r_i or π_i can be zero.

We use the generic random variable $\mathsf{Duration}_i$ to represent the random rental duration of product i. The random variable $\mathsf{Duration}_i$ has a probability mass function $f_i : \mathbb{Z}_{++} \mapsto [0,1]$, where $\sum_{\ell=1}^{\infty} f_i(\ell) = 1$. We describe the rental duration in terms of its hazard rate $\rho_{i,\ell}$ associated with the probability mass function f_i, where for each $\ell \in \mathbb{Z}_+$, we have

$$\rho_{i,\ell} \;=\; \Pr\{\mathsf{Duration}_i = \ell + 1 \mid \mathsf{Duration}_i > \ell\} \;=\; \frac{f_i(\ell+1)}{\sum_{s=\ell+1}^{\infty} f_i(s)}.$$

The hazard rate $\rho_{i,\ell}$ is the probability that each unit of product i is returned after $\ell + 1$ periods, given that it has been used for more than ℓ periods. Since $\sum_{s=1}^{\infty} f_i(s) = 1$, we have $\rho_{i,0} = f_i(1)$, so that $\rho_{i,0}$ is the probability that a unit of product i is used for exactly one time period. The usage duration of different units are assumed to be independent of each other.

At each time period t, the following sequence of events happen. We observe whether each customer with a rented unit of product decides to return the unit, and gather the returned units. Next, we observe the state of the system, which consists of the current on-hand units and the outstanding units that are being rented by the customers. Based on the state, we decide which subset of products to offer to customers at time period t. The customer arriving at time period t chooses a unit to rent or leaves the system without renting. Finally, we collect the upfront fee for the rented unit and the rent from all customers still using their rented units.

State and Transition Dynamics: As discussed in the previous paragraph, we observe the state of the system at each time period after the returns have been realized. To capture the state of the system at a generic time period, we use $q_{i,0}$ to denote the number of units of product i available as on-hand inventory. For $\ell \geq 1$, let $q_{i,\ell}$ denote the number of units of product i that have been used for exactly ℓ time periods, after all of the returns have been realized. Therefore, we can describe the state of the system by using the vector $\boldsymbol{q} = (q_{i,\ell} : i \in \mathcal{N},\ \ell = 0, 1, \ldots)$. Since $\sum_{\ell=0}^{\infty} q_{i,\ell} = C_i$, let $\mathcal{Q} = \{(q_{i,\ell} : i \in \mathcal{N},\ \ell = 0, 1, \ldots) : \sum_{\ell=0}^{\infty} q_{i,\ell} = C_i \ \forall\, i \in \mathcal{N}\}$ denote the set of all possible states. We assume that we start the system with no units in use and that all C_i units of each product i are available on-hand. Thus, there will never be a unit in use for more than T time periods, indicating that the effective set of possible states is always finite.

After the returns are realized, consider the state $\boldsymbol{q}$ at time period t. There are $q_{i,\ell}$ units of product i that have been used for exactly ℓ periods. Each of these units will be used by customers for at least one more time period, which is time period t. By definition of the hazard rate, with probability $\rho_{i,\ell}$, each of the $q_{i,\ell}$ units will be returned at the beginning of period $t+1$. Therefore, if there is no purchase at time period t, then the number of units that will be available as on-hand inventory at time period $t+1$ is $q_{i,0} + \sum_{\ell=1}^{\infty} \mathsf{Bin}(q_{i,\ell}, \rho_{i,\ell})$, where $\mathsf{Bin}(k,p)$ denotes a binomial random variable with parameters $k \in \mathbb{Z}_{++}$ and $p \in (0,1)$. At time period $t+1$, after the returns are realized, the number of units of product i that will have been rented out for $\ell + 1$ periods will be $q_{i,\ell} - \mathsf{Bin}(q_{i,\ell}, \rho_{i,\ell})$, where the second term reflects the units that will be returned. So, given the state $\boldsymbol{q}$ at time period t, if there is no purchase by a customer, then the state $\boldsymbol{X}(\boldsymbol{q}) = (X_{i,\ell}(\boldsymbol{q}) : i \in \mathcal{N},\ \ell = 0, 1, \ldots)$ at time period $t+1$ is given by

$$X_{i,\ell}(\boldsymbol{q}) = \begin{cases} q_{i,0} + \sum_{s=1}^{\infty} \mathsf{Bin}(q_{i,s}, \rho_{i,s}) & \text{if } \ell = 0, \\ 0 & \text{if } \ell = 1, \\ q_{i,\ell-1} - \mathsf{Bin}(q_{i,\ell-1}, \rho_{i,\ell-1}) & \text{if } \ell \geq 2. \end{cases} \tag{1}$$

Note that when there is no purchase in the current time period, we have $X_{i,1}(\boldsymbol{q}) = 0$ because each unit that has been in use will either return to the firm or continue to be in use. In the latter case, its usage duration will be at least two time periods.

Dynamic Programming Formulation: We use $\mathcal{F}$ to denote the collection of feasible subsets of products that we can offer to the customers at each time period, capturing the constraints that we may impose on the offered subset of products. To formulate the problem as a dynamic program, we denote a Bernoulli random variable with parameter ρ by $\mathsf{Z}(\rho)$; that is, we have $\Pr\{\mathsf{Z}(\rho)=1\}=\rho$ and $\Pr\{\mathsf{Z}(\rho)=0\}=1-\rho$. Lastly, viewing the state $\boldsymbol{q}=(q_{i,\ell} : i\in\mathcal{N},\ \ell=0,1,\dots)$ as a vector, we let $\boldsymbol{e}_{i,k}$ be a unit vector with one in the (i,k)-th coordinate and zero everywhere else. Let $J^t(\boldsymbol{q})$ denote the maximum total expected revenue over the time periods $t,t+1,\dots,T$, given that the system is in state $\boldsymbol{q}$ at time period t. Then, using $\mathbb{1}_{\{\cdot\}}$ to denote the indicator function, we can compute the optimal value functions $\{J^t\ :\ t\in\mathcal{T}\}$ by solving the dynamic program

$$\begin{aligned} J^t(\boldsymbol{q}) &= \sum_{i\in\mathcal{N}}\pi_i\sum_{\ell=1}^{\infty}q_{i,\ell} \\ &+\ \max_{S\in\mathcal{F}}\left\{\sum_{i\in\mathcal{N}}\mathbb{1}_{\{q_{i,0}\geq 1\}}\phi_i(S)\left(r_i+\pi_i+\mathbb{E}\Big\{\mathsf{Z}(\rho_{i,0})\,J^{t+1}(\boldsymbol{X}(\boldsymbol{q}))+(1-\mathsf{Z}(\rho_{i,0}))\,J^{t+1}\left(\boldsymbol{X}(\boldsymbol{q})-\boldsymbol{e}_{i,0}+\boldsymbol{e}_{i,1}\right)\Big\}\right)\right. \\ &\qquad\qquad\left.+\left(1-\sum_{i\in\mathcal{N}}\mathbb{1}_{\{q_{i,0}\geq 1\}}\phi_i(S)\right)\mathbb{E}\Big\{J^{t+1}(\boldsymbol{X}(\boldsymbol{q}))\Big\}\right\}, \end{aligned} \tag{2}$$

with the boundary condition that $J^{T+1}=0$. In the dynamic programming formulation above, we implicitly assume that even if $q_{i,0}=0$ so that we do not have any on-hand inventory for product i, we can offer an assortment that includes product i. Noting the indicator function, if a customer chooses a product with zero on-hand inventory, then she leaves the system without renting any products. The possibility of offering products with zero on-hand inventory may be unrealistic in certain settings. Later in this section, in Assumption 2.1, we impose rather mild assumptions on the discrete choice model $\{\phi_i(S): i\in\mathcal{N},\ S\subseteq\mathcal{N}\}$ and the set of feasible decisions $\mathcal{F}$ to ensure that the optimal policy never offers a product with zero on-hand inventory, even if we are allowed to do so. In this case, it follows that the dynamic programming formulation above is equivalent to a dynamic programming formulation that explicitly imposes a constraint to ensure that we must have non-zero on-hand inventory for each product that we offer.

In the dynamic program in (2), the term $\sum_{i\in\mathcal{N}}\pi_i\sum_{\ell=1}^{\infty}q_{i,\ell}=\sum_{i\in\mathcal{N}}\pi_i(C_i-q_{i,0})$ captures the rent payments from customers with already rented units. After observing the returns at time period t, there are $\sum_{\ell=1}^{\infty}q_{i,\ell}=C_i-q_{i,0}$ units of product i that are in use. All of these units will be used during time period t, so the customers who are using these units will pay the rent in the amount of $\pi_i\sum_{\ell=1}^{\infty}q_{i,\ell}$ for this period. On the other hand, the term $r_i+\pi_i+\mathbb{E}\Big\{\mathsf{Z}(\rho_{i,0})\,J^{t+1}(\boldsymbol{X}(\boldsymbol{q}))+(1-\mathsf{Z}(\rho_{i,0}))\,J^{t+1}\left(\boldsymbol{X}(\boldsymbol{q})-\boldsymbol{e}_{i,0}+\boldsymbol{e}_{i,1}\right)\Big\}$ corresponds to the expected revenue from a customer who selects product i. Here, $r_i+\pi_i$ reflects the upfront payment and the per-period rent for the first rental period. Noting the definition of the hazard rate, we have $\rho_{i,0}=f_i(1)$. Therefore, the Bernoulli random variable $\mathsf{Z}(\rho_{i,0})$ takes a value of 1 if and only if the customer renting a unit of product i at time period t uses the product for exactly one time period. If $\mathsf{Z}(\rho_{i,0})=1$, then the unit is returned to the firm at the beginning of period $t+1$, in which case, the state at time period $t+1$ is $\boldsymbol{X}(\boldsymbol{q})$, identical to the state that we would have obtained when no rentals were made at time period t. On the other hand, if $\mathsf{Z}(\rho_{i,0})=0$, then the selected unit of product i will not be returned at the beginning of time period $t+1$. In this case, when compared to the state $\boldsymbol{X}(\boldsymbol{q})$ with no rentals at time period t, we will have one fewer on-hand unit for product i and one more unit with one time period in use. So, the state of the system at time period $t+1$ will be $\boldsymbol{X}(\boldsymbol{q})-\boldsymbol{e}_{i,0}+\boldsymbol{e}_{i,1}$. To simplify our dynamic programming formulation, we observe that since the rental durations of different units are independent of each other, $\boldsymbol{X}(\boldsymbol{q})$ and $\mathsf{Z}(\rho_{i,0})$ are independent of each other as well. Therefore, we obtain

$$\begin{aligned}
&\mathbb{E}\Big\{\mathsf{Z}(\rho_{i,0})\,J^{t+1}(\boldsymbol{X}(\boldsymbol{q}))+(1-\mathsf{Z}(\rho_{i,0}))\,J^{t+1}\left(\boldsymbol{X}(\boldsymbol{q})-\boldsymbol{e}_{i,0}+\boldsymbol{e}_{i,1}\right)\Big\}-\mathbb{E}\Big\{J^{t+1}(\boldsymbol{X}(\boldsymbol{q}))\Big\}\\
&\quad=\;\rho_{i,0}\,\mathbb{E}\Big\{J^{t+1}\left(\boldsymbol{X}(\boldsymbol{q})\right)\Big\}+(1-\rho_{i,0})\,\mathbb{E}\Big\{J^{t+1}\left(\boldsymbol{X}(\boldsymbol{q})-\boldsymbol{e}_{i,0}+\boldsymbol{e}_{i,1}\right)\Big\}-\mathbb{E}\Big\{J^{t+1}(\boldsymbol{X}(\boldsymbol{q}))\Big\}\\
&\quad=\;-(1-\rho_{i,0})\,\mathbb{E}\Big\{J^{t+1}\left(\boldsymbol{X}(\boldsymbol{q})\right)-J^{t+1}\left(\boldsymbol{X}(\boldsymbol{q})-\boldsymbol{e}_{i,0}+\boldsymbol{e}_{i,1}\right)\Big\},
\end{aligned}$$

in which case, simply by rearranging the terms, we can write the dynamic programming formulation in (2) equivalently as

$$\begin{aligned}
J^t(\boldsymbol{q})=&\sum_{i\in\mathcal{N}}\pi_i\sum_{\ell=1}^{\infty}q_{i,\ell}\;+\;\mathbb{E}\Big\{J^{t+1}(\boldsymbol{X}(\boldsymbol{q}))\Big\}\\
&+\max_{S\in\mathcal{F}}\left\{\sum_{i\in\mathcal{N}}\mathbb{1}_{\{q_{i,0}\geq 1\}}\,\phi_i(S)\left(r_i+\pi_i-(1-\rho_{i,0})\,\mathbb{E}\Big\{J^{t+1}\left(\boldsymbol{X}(\boldsymbol{q})\right)-J^{t+1}\left(\boldsymbol{X}(\boldsymbol{q})-\boldsymbol{e}_{i,0}+\boldsymbol{e}_{i,1}\right)\Big\}\right)\right\}. \quad (3)
\end{aligned}$$

Note that $J^{t+1}\left(\boldsymbol{X}(\boldsymbol{q})\right)-J^{t+1}\left(\boldsymbol{X}(\boldsymbol{q})-\boldsymbol{e}_{i,0}+\boldsymbol{e}_{i,1}\right)$ captures the marginal value of renting one unit of product i to the customer at time period t.

Throughout the paper, we impose a mild assumption on the discrete choice model $\{\phi_i(S):i\in\mathcal{N},\ S\subseteq\mathcal{N}\}$ and the set of feasible decisions $\mathcal{F}$ to ensure that the optimal policy never offers a product with zero on-hand inventory. This assumption is given below.

Assumption 2.1 (Substitutability and Feasibility) *Adding more products to an assortment does not increase the selection probability; that is, for all $S\subseteq\mathcal{N}$ and $k\in\mathcal{N}$, $\phi_i(S\cup\{k\})\leq\phi_i(S)$ for all $i\in S$. In addition, if a set of products is feasible to offer, then so are all of its subsets; that is, if $A\in\mathcal{F}$, then $S\in\mathcal{F}$ for all $S\subseteq A$.*

The first assumption ensures that products are substitutable, and thus, the probability of choosing any product never increases if more options become available. This assumption is rather mild and it holds for all choice models satisfying the random utility maximization principle, including the multinomial logit, nested logit, paired combinatorial logit, and many others. In addition, the feasibility assumption on the collection of subsets $\mathcal{F}$ also holds for a broad class of assortment constraints, such as a shelf-space constraint $\mathcal{F} = \left\{S \subseteq \mathcal{N} : \sum_{i \in S} c_i \leq B\right\}$, where c_i is the space consumed by product i and B is the total shelf-space available. Under the assumption above, it is not difficult to see that the optimal policy never offers a product with zero on-hand inventory. In the maximization problem in (3), the profit contribution of product i is $\mathbb{1}_{\{q_{i,0} \geq 1\}} \times (r_i + \pi_i - (1 - \rho_{i,0}) \mathbb{E}\{J^{t+1}(\boldsymbol{X}(\boldsymbol{q})) - J^{t+1}(\boldsymbol{X}(\boldsymbol{q}) - \boldsymbol{e}_{i,0} + \boldsymbol{e}_{i,1})\})$. Let S^* be an optimal solution to this maximization problem. In this case, observe that we can drop all products with non-positive profit contributions from S^* because if we drop such products, then by the substitutability assumption, the selection probabilities of all other products increase, whereas by the feasibility assumption, the subset we obtain remains feasible. The new subset that we obtain in this fashion provides an objective value to the maximization problem in (3) that is at least as large as that provided by S^*. As the profit contribution of product i is zero when $\mathbb{1}_{\{q_{i,0} \geq 1\}} = 0$, there exists an optimal policy that never offers a product with zero on-hand inventory.

Because all products are available at the beginning of the selling horizon, the optimal total expected revenue is given by $J^1\left(\sum_{i \in \mathcal{N}} C_i \, \boldsymbol{e}_{i,0}\right)$. One potential source of difficulty in computing the optimal value functions $\{J^t : t \in \mathcal{T}\}$ is that the maximization problem in (3) is a combinatorial optimization problem that chooses the set of products to offer. However, this problem has been studied for many different discrete choice models, including the multinomial logit, nested logit, d-level logit, and paired combinatorial logit, and under many different types of feasible sets $\mathcal{F}$. Later in the paper, we also discuss how our results extend when we can solve this maximization problem only approximately. Thus, the difficulty due to having to solve the maximization problem in (3) is not a huge concern. A more serious source of difficulty is that we need to compute the value function $J^t(\boldsymbol{q})$ for each $\boldsymbol{q} \in \mathcal{Q}$, and the number of possible states $|\mathcal{Q}|$ grows exponentially with n and T. Therefore, throughout the rest of the paper, we focus on developing approximate policies that are efficient to compute and have provable performance guarantees.

3. Linear Value Function Approximations

We develop an approach to construct linear approximations to the optimal value functions and analyze the performance of a policy that uses these approximations. In particular, we give a tractable recursion to come up with linear value function approximations. We show that if we use the greedy policy with respect to these linear value function approximations, then we obtain a policy that is guaranteed to obtain at least 50% of the optimal total expected revenue.

3.1 Specification of Linear Value Function Approximations

We consider an approximation $\hat{J}^t$ to the optimal value function J^t given by

$$\hat{J}^t(\boldsymbol{q}) \;=\; \hat{\beta}^t \;+\; \sum_{i\in\mathcal{N}}\sum_{\ell=0}^{\infty}\hat{\nu}^t_{i,\ell}\,q_{i,\ell},$$

where, for $\ell \geq 1$, the parameter $\hat{\nu}^t_{i,\ell}$ represents the marginal value at time period t of each unit of product i that has been in use for ℓ periods, whereas $\hat{\nu}^t_{i,0}$ denotes the marginal value of each unit of product i that is currently available as on-hand inventory at time period t. The parameter $\hat{\beta}^t$ is a simple intercept. We propose computing $\hat{\nu}^t_{i,\ell}$ and $\hat{\beta}^t$ recursively as follows.

- **Initialization:** Set $\hat{\nu}^{T+1}_{i,\ell} = 0$ for all $i \in \mathcal{N},\ \ell \geq 0$ and set $\hat{\beta}^{T+1} = 0$.
- **Backward Recursion:** For $t = T, T-1, \ldots, 1$, we compute $\hat{\nu}^t_{i,\ell}$ and $\hat{\beta}^t$ by using $\{\hat{\nu}^{t+1}_{i,\ell} : i \in \mathcal{N},\ \ell \geq 0\}$ as follows. Let $\hat{A}^t \in \mathcal{F}$ be an assortment such that

$$\hat{A}^t \;=\; \arg\max_{S\in\mathcal{F}} \sum_{i\in\mathcal{N}}\phi_i(S)\Big[r_i + \pi_i - (1-\rho_{i,0})\left(\hat{\nu}^{t+1}_{i,0} - \hat{\nu}^{t+1}_{i,1}\right)\Big]. \tag{4}$$

  Once $\hat{A}^t$ is computed, for all $i \in \mathcal{N}$, let

$$\begin{aligned}
\hat{\nu}^t_{i,0} \;&=\; \hat{\nu}^{t+1}_{i,0} + \frac{1}{C_i}\phi_i(\hat{A}^t)\Big[r_i + \pi_i - (1-\rho_{i,0})\left(\hat{\nu}^{t+1}_{i,0} - \hat{\nu}^{t+1}_{i,1}\right)\Big] \qquad (5)\\
\hat{\nu}^t_{i,\ell} \;&=\; \pi_i \;+\; \rho_{i,\ell}\,\hat{\nu}^{t+1}_{i,0} + (1-\rho_{i,\ell})\,\hat{\nu}^{t+1}_{i,\ell+1} \qquad \forall\,\ell = 1,2,\ldots\\
\hat{\beta}^t \;&=\; \sum_{i\in\mathcal{N}} C_i\,\hat{\nu}^t_{i,0}.
\end{aligned}$$

The above description completes the specification of the approximate value function $\hat{J}^t$. We shortly give the intuition behind our approach. Because we start the system with all available units as on-hand inventory, no unit will be in use for more than T time periods. Thus, we only need to compute $\hat{\nu}^t_{i,\ell}$ for $\ell = 0, 1, \ldots, T$, so we can execute the above recursion in finite time.

We provide intuition into the computation of $\hat{A}^t$. Intuitively speaking, we can interpret $\hat{A}^t$ as an ideal assortment to offer at time period t under the linear value function approximation when we ignore the inventory availability. In particular, if we replace the value function J^{t+1} in the maximization problem on the right side of (3) with the linear approximation $\hat{J}^{t+1}(\boldsymbol{q}) = \hat{\beta}^{t+1} + \sum_{i\in\mathcal{N}}\sum_{\ell=0}^{\infty}\hat{\nu}^{t+1}_{i,\ell}\,q_{i,\ell}$ and we drop the indicator function $\mathbb{1}_{\{q_{i,0}\geq 1\}}$ to ignore the inventory availability, then the objective function of this maximization problem takes the form $\sum_{i\in\mathcal{N}}\phi_i(S)\left[r_i + \pi_i - (1-\rho_{i,0})\left(\hat{\nu}^{t+1}_{i,0} - \hat{\nu}^{t+1}_{i,1}\right)\right]$, which is the same as the objective function of the maximization problem in (4). Next, we provide some intuition into the computation of $\nu^t_{i,0}$, which measures the value of a unit of on-hand inventory for product i at time period t. Roughly speaking, assume that we offer the ideal assortment $\hat{A}^t$ at time period t, and if a customer selects product i

at time period t, then we "route" the customer to one of the C_i copies of product i with equal probability of $1/C_i$. In this case, the probability that a unit of product i "sees" a demand at time period t is $\phi_i(\hat{A}^t)\,\frac{1}{C_i}$. We write the recursion that we use to compute $\hat{\nu}^t_{i,0}$ equivalently as

$$\hat{\nu}^t_{i,0} \;=\; \frac{1}{C_i}\phi_i(\hat{A}^t)\Big[r_i + \pi_i + \rho_{i,0}\,\hat{\nu}^{t+1}_{i,0} + (1-\rho_{i,0})\,\hat{\nu}^{t+1}_{i,1}\Big] + \left(1 - \frac{1}{C_i}\,\phi_i(\hat{A}^t)\right)\hat{\nu}^{t+1}_{i,0}.$$

On the left side above, $\hat{\nu}^t_{i,0}$ is the value of a unit of product i on-hand at time period t. If we offer the ideal assortment $\hat{A}^t$ at time period t, then a unit of product i "sees" a demand with probability $\frac{1}{C_i}\phi_i(\hat{A}^t)$. In this case, we collect the upfront fee r_i and the rent π_i for the first time period. As discussed earlier, with probability $\rho_{i,0} = f_i(1)$, the customer rents product i for exactly one time period, in which case she returns the product at time period $t+1$. The value of a unit of on-hand inventory of product i at time period $t+1$ is $\hat{\nu}^{t+1}_{i,0}$. With probability $1-\rho_{i,0}$, the customer rents product i for more than one time period, in which case the product will have been rented out at time period $t+1$ for exactly one period. The value of a unit of product i at time period $t+1$ that has been in use for one period is $\hat{\nu}^{t+1}_{i,1}$. This discussion provides the intuition for the term $r_i + \pi_i + \rho_{i,0}\,\hat{\nu}^{t+1}_{i,0} + (1-\rho_{i,0})\,\hat{\nu}^{t+1}_{i,1}$ on the right side above. With probability $1 - \frac{1}{C_i}\phi_i(\hat{A}^t)$, the unit of product i does not "see" a demand, in which case this unit is still available at time period $t+1$ and the value of this unit is given by $\hat{\nu}^{t+1}_{i,0}$.

We can give a similar intuition for the recursion that is used to compute $\hat{\nu}^t_{i,\ell}$ for all $\ell = 1, 2, \ldots$. Noting the recursion $\hat{\nu}^t_{i,\ell} = \pi_i + \rho_{i,\ell}\,\hat{\nu}^{t+1}_{i,0} + (1-\rho_{i,\ell})\,\hat{\nu}^{t+1}_{i,\ell+1}$, recall that $\hat{\nu}^t_{i,\ell}$ on the left side is the value of a unit of product i that has been in use for ℓ periods at time period t. As the state of the system is accounted for after observing the rental returns in the current time period, this product will certainly be used until the end of time period t and we will obtain the rental fee of π_i. Furthermore, by the definition of the hazard rate $\rho_{i,\ell}$, a unit of product that has been in use for ℓ periods at time period t will be returned in the next time period with probability $\rho_{i,\ell}$, in which case, the value of this on-hand unit at time period $t+1$ is $\hat{\nu}^{t+1}_{i,0}$, yielding the term $\rho_{i,\ell}\,\hat{\nu}^{t+1}_{i,0}$ on the right side. Finally, a unit of product that has been in use for ℓ periods at time period t will not be returned in the next time period with probability $1-\rho_{i,\ell}$. Therefore, this unit of product will have been used for $\ell+1$ periods at the next time period and the value of this unit at time period $t+1$ is $\hat{\nu}^{t+1}_{i,\ell+1}$, yielding the term $(1-\rho_{i,\ell})\,\hat{\nu}^{t+1}_{i,\ell+1}$ on the right side.

Our choice for $\hat{\beta}^t$ is motivated by the fact that we will use a balancing argument to ultimately obtain a half-approximate policy. In particular, if all products are available at time period t, then the two terms in the value function approximation $\hat{\beta}^t + \sum_{i\in\mathcal{N}}\sum_{\ell=0}^{\infty}\hat{\nu}^t_{i,\ell}\,q_{i,\ell}$ evaluate to the same quantity $\sum_{i\in\mathcal{N}} C_i\,\hat{\nu}^t_{i,0}$. In the next lemma, we give a useful property for the marginal value $\hat{\nu}^t_{i,\ell}$, where we show that the marginal value of a unit of product becomes smaller as the end of the selling horizon approaches. We use this property several times throughout the paper.

Lemma 3.1 (Properties of the Marginal Values) *The marginal value of on-hand inventory decreases over time; that is,* $\hat{\nu}^t_{i,0} \geq \hat{\nu}^{t+1}_{i,0}$ *for all* $t \in \mathcal{T}$ *and* $i \in \mathcal{N}$.

Proof: For notational brevity, let $\Delta^t_i = r_i + \pi_i - (1-\rho_{i,0})\left(\hat{\nu}^{t+1}_{i,0} - \hat{\nu}^{t+1}_{i,1}\right)$. Shortly, we show the claim that $\phi_i(\hat{A}^t)\,\Delta^t_i \geq 0$ for all $i \in \mathcal{N}$. In this case, noting the recursion that we use to compute $\hat{\nu}^t_{i,0}$, we have $\hat{\nu}^t_{i,0} = \hat{\nu}^{t+1}_{i,0} + \frac{1}{C_i}\,\phi_i(\hat{A}^t)\,\Delta^t_i \geq \nu^{t+1}_{i,0}$, which is the desired result. To see the claim that $\phi_i(\hat{A}^t)\,\Delta^t_i \geq 0$ for all $i \in \mathcal{N}$, assume on the contrary that there exists some $k \in \mathcal{N}$ such that $\phi_k(\hat{A}^t)\,\Delta^t_k < 0$. Let $\mathcal{N}^+ = \{i \in \mathcal{N} : \Delta^t_i \geq 0\}$ and $\mathcal{N}^- = \{i \in \mathcal{N} : \Delta^t_i < 0\}$. By our assumption, there exists some $k \in \mathcal{N}^-$ such that $\phi_k(\hat{A}^t) > 0$. Furthermore, by Assumption 2.1, $\phi_i(\hat{A}^t \cap \mathcal{N}^+) \geq \phi_i(\hat{A}^t)$ for all $i \in \hat{A}^t \cap \mathcal{N}^+$. By the same assumption, because $\hat{A}^t \in \mathcal{F}$, we have $\hat{A}^t \cap \mathcal{N}^+ \in \mathcal{F}$. In this case, we obtain

$$\begin{aligned}\sum_{i\in\mathcal{N}} \phi_i(\hat{A}^t)\,\Delta^t_i \;&=\; \sum_{i\in\mathcal{N}^+} \phi_i(\hat{A}^t)\,\Delta^t_i \;+\; \sum_{i\in\mathcal{N}^-} \phi_i(\hat{A}^t)\,\Delta^t_i \;<\; \sum_{i\in\mathcal{N}^+} \phi_i(\hat{A}^t)\,\Delta^t_i \\ &=\; \sum_{i\in\hat{A}^t\cap\mathcal{N}^+} \phi_i(\hat{A}^t)\,\Delta^t_i \;\leq\; \sum_{i\in\hat{A}^t\cap\mathcal{N}^+} \phi_i(\hat{A}^t\cap\mathcal{N}^+)\,\Delta^t_i \;=\; \sum_{i\in\mathcal{N}} \phi_i(\hat{A}^t\cap\mathcal{N}^+)\,\Delta^t_i,\end{aligned}$$

where the first inequality is by the fact that there exists some $k \in \mathcal{N}^-$ such that $\phi_k(\hat{A}^t) > 0$, the second equality holds since $\phi_i(\hat{A}^t) = 0$ for all $i \notin \hat{A}^t$, and the second inequality uses the fact that $\phi_i(\hat{A}^t \cap \mathcal{N}^+) \geq \phi_i(\hat{A}^t)$ for all $i \in \hat{A}^t \cap \mathcal{N}^+$. Since $\hat{A}^t \cap \mathcal{N}^+ \in \mathcal{F}$, the chain of inequalities above contradicts the fact that $\hat{A}^t$ is an optimal solution to problem (4). ■

3.2 An Approximate Policy Using Marginal Values

We consider the greedy policy with respect to the value function approximations $\{\hat{J}^t : t \in \mathcal{T}\}$. If the system is in state $\boldsymbol{q}$ at time period t, then this policy offers the assortment $\hat{S}^t(\boldsymbol{q})$ given by

$$\begin{aligned}\hat{S}^t(\boldsymbol{q}) \;&=\; \arg\max_{S\in\mathcal{F}} \left\{ \sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\geq 1\}} \phi_i(S)\Big[r_i + \pi_i - (1-\rho_{i,0})\,\mathbb{E}\Big\{\hat{J}^{t+1}(\boldsymbol{X}(\boldsymbol{q})) - \hat{J}^{t+1}(\boldsymbol{X}(\boldsymbol{q}) - \boldsymbol{e}_{i,0} + \boldsymbol{e}_{i,1})\Big\}\Big]\right\} \\ &=\; \arg\max_{S\in\mathcal{F}} \;\sum_{i=1}^{n} \mathbb{1}_{\{q_{i,0}\geq 1\}} \phi_i(S)\Big[r_i + \pi_i - (1-\rho_{i,0})\left(\hat{\nu}^{t+1}_{i,0} - \hat{\nu}^{t+1}_{i,1}\right)\Big]. \qquad (6)\end{aligned}$$

The main result of this section is stated in the following theorem, which gives a performance guarantee for the greedy policy with respect to the value function approximations $\{\hat{J}^t : t \in \mathcal{T}\}$.

Theorem 3.2 (Performance of the Greedy Policy) *The total expected revenue of the greedy policy with respect to the value function approximations* $\left\{\hat{J}^t : t \in \mathcal{T}\right\}$ *is at least 50% of the optimal total expected revenue.*

The proof of this theorem makes use of the following lemma. Because we do not have any products in use at the beginning of the selling horizon, the initial state is $\sum_{i\in\mathcal{N}} C_i\,\boldsymbol{e}_{i,0}$. The following lemma relates $\sum_{i\in\mathcal{N}} C_i\,\hat{\nu}^1_{i,0}$ to the optimal total expected revenue $J^1\left(\sum_{i\in\mathcal{N}} C_i\,\boldsymbol{e}_{i,0}\right)$.

Lemma 3.3 (Expected Revenue Upper Bound) $J^1\left(\sum_{i\in\mathcal{N}} C_i\, \boldsymbol{e}_{i,0}\right) \le 2\sum_{i\in\mathcal{N}} C_i\,\hat{\nu}^1_{i,0}$.

Proof: By Adelman (2007), we can obtain an upper bound on the optimal total expected revenue by using the objective value provided by any feasible solution to the linear program

$$\begin{aligned}
\min\quad & \tilde{J}^1\left(\sum_{i\in\mathcal{N}} C_i \boldsymbol{e}_{i,0}\right) \\
\text{s.t.}\quad & \tilde{J}^t(\boldsymbol{q}) \;\ge\; \sum_{i\in\mathcal{N}} \pi_i \sum_{\ell=1}^{\infty} q_{i,\ell} \;+\; \mathbb{E}\left\{\tilde{J}^{t+1}(\boldsymbol{X}(\boldsymbol{q}))\right\} \\
& \qquad +\; \sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\ge 1\}}\, \phi_i(S)\left[r_i + \pi_i - (1-\rho_{i,0})\,\mathbb{E}\left\{\tilde{J}^{t+1}\left(\boldsymbol{X}(\boldsymbol{q})\right) - \tilde{J}^{t+1}\left(\boldsymbol{X}(\boldsymbol{q}) - \boldsymbol{e}_{i,0} + \boldsymbol{e}_{i,1}\right)\right\}\right] \\
& \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \forall\, \boldsymbol{q}\in\mathcal{Q},\; S\in\mathcal{F},\; t\in\mathcal{T},
\end{aligned}$$

where the decision variables are $\{\tilde{J}^t(\boldsymbol{q}) : \boldsymbol{q}\in\mathcal{Q},\; t\in\mathcal{T}\}$ and we follow the convention that $\tilde{J}^{T+1} = 0$. We proceed to showing that $\{\hat{J}^t(\boldsymbol{q}) : \boldsymbol{q}\in\mathcal{Q},\; t\in\mathcal{T}\}$ with $\hat{J}^t(\boldsymbol{q}) = \hat{\beta}^t + \sum_{i\in\mathcal{N}}\sum_{\ell=0}^{\infty} \hat{\nu}^t_{i,\ell}\, q_{i,\ell}$ is a feasible solution to the linear program above. As all products are available at the beginning of the selling horizon, there will never be a product that is in use for more than T time periods. Thus, we can assume that $\mathcal{Q}$ is a finite set, which implies that the numbers of decision variables and constraints above are finite. By the definition of $\hat{J}^{t+1}(\boldsymbol{q})$, along with $\boldsymbol{X}(\boldsymbol{q})$ in (1), we obtain

$$\begin{aligned}
\mathbb{E}\left\{\hat{J}^{t+1}(\boldsymbol{X}(\boldsymbol{q}))\right\} &= \hat{\beta}^{t+1} + \sum_{i\in\mathcal{N}}\left\{\hat{\nu}^{t+1}_{i,0}\left[q_{i,0} + \sum_{\ell=1}^{\infty}\rho_{i,\ell}\, q_{i,\ell}\right] + \sum_{\ell=1}^{\infty}\hat{\nu}^{t+1}_{i,\ell+1}\left[q_{i,\ell} - \rho_{i,\ell}\, q_{i,\ell}\right]\right\} \\
&= \hat{\beta}^{t+1} + \sum_{i\in\mathcal{N}}\left\{q_{i,0}\,\hat{\nu}^{t+1}_{i,0} + \sum_{\ell=1}^{\infty} q_{i,\ell}\left[\rho_{i,\ell}\,\hat{\nu}^{t+1}_{i,0} + (1-\rho_{i,\ell})\,\hat{\nu}^{t+1}_{i,\ell+1}\right]\right\}.
\end{aligned}$$

Similarly, $\mathbb{E}\left\{\hat{J}^{t+1}\left(\boldsymbol{X}(\boldsymbol{q})\right) - \hat{J}^{t+1}\left(\boldsymbol{X}(\boldsymbol{q}) - \boldsymbol{e}_{i,0} + \boldsymbol{e}_{i,1}\right)\right\} = \hat{\nu}^{t+1}_{i,0} - \hat{\nu}^{t+1}_{i,1}$. So, if we evaluate the right side of the constraint in the linear program above at $\{\hat{J}^t(\boldsymbol{q}) : \boldsymbol{q}\in\mathcal{Q},\; t\in\mathcal{T}\}$, then we obtain

$$\begin{aligned}
& \sum_{i\in\mathcal{N}} \pi_i \sum_{\ell=1}^{\infty} q_{i,\ell} \;+\; \mathbb{E}\left\{\hat{J}^{t+1}(\boldsymbol{X}(\boldsymbol{q}))\right\} \\
&\qquad +\; \sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\ge 1\}}\, \phi_i(S)\left[r_i + \pi_i - (1-\rho_{i,0})\,\mathbb{E}\left\{\hat{J}^{t+1}\left(\boldsymbol{X}(\boldsymbol{q})\right) - \hat{J}^{t+1}\left(\boldsymbol{X}(\boldsymbol{q}) - \boldsymbol{e}_{i,0} + \boldsymbol{e}_{i,1}\right)\right\}\right] \\
&= \sum_{i\in\mathcal{N}} \pi_i \sum_{\ell=1}^{\infty} q_{i,\ell} + \hat{\beta}^{t+1} + \sum_{i\in\mathcal{N}}\left\{q_{i,0}\,\hat{\nu}^{t+1}_{i,0} + \sum_{\ell=1}^{\infty} q_{i,\ell}\left[\rho_{i,\ell}\,\hat{\nu}^{t+1}_{i,0} + (1-\rho_{i,\ell})\,\hat{\nu}^{t+1}_{i,\ell+1}\right]\right\} \\
&\qquad +\; \sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\ge 1\}}\, \phi_i(S)\left[r_i + \pi_i - (1-\rho_{i,0})\,(\hat{\nu}^{t+1}_{i,0} - \hat{\nu}^{t+1}_{i,1})\right] \\
&= \sum_{i\in\mathcal{N}} \hat{\nu}^{t+1}_{i,0}\, C_i + \sum_{i\in\mathcal{N}}\left\{q_{i,0}\,\hat{\nu}^{t+1}_{i,0} + \sum_{\ell=1}^{\infty} q_{i,\ell}\hat{\nu}^t_{i,\ell}\right\} + \sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\ge 1\}}\, \phi_i(S)\left[r_i + \pi_i - (1-\rho_{i,0})\,(\hat{\nu}^{t+1}_{i,0} - \hat{\nu}^{t+1}_{i,1})\right],
\end{aligned}$$

where the second equality follows by noting the definitions of $\hat{\beta}^{t+1}$ and $\hat{\nu}^t_{i,\ell}$, where we have $\hat{\beta}^{t+1} = \sum_{i\in\mathcal{N}} \hat{\nu}^{t+1}_{i,0}\, C_i$ and $\hat{\nu}^t_{i,\ell} = \pi_i + \rho_{i,\ell}\,\hat{\nu}^{t+1}_{i,0} + (1-\rho_{i,\ell})\,\hat{\nu}^{t+1}_{i,\ell+1}$. A simple lemma, given as Lemma A.1 in

Appendix A, shows that if we let $\Delta_i^t = r_i + \pi_i - (1-\rho_{i,0})\,(\hat{\nu}_{i,0}^{t+1} - \hat{\nu}_{i,1}^{t+1})$, then for all $S \in \mathcal{F}$, we have $\sum_{i \in \mathcal{N}} \mathbb{1}_{\{q_{i,0} \geq 1\}}\, \phi_i(S)\, \Delta_i^t \leq \sum_{i \in \mathcal{N}} \phi_i(\hat{A}^t)\, \Delta_i^t$. The proof of this lemma follows from an argument similar to that in the proof of Lemma 3.1. Thus, noting the chain of equalities above, we upper bound the right side of the constraint in the linear program as

$$\begin{aligned}
&\sum_{i \in \mathcal{N}} \hat{\nu}_{i,0}^{t+1}\, C_i + \sum_{i \in \mathcal{N}} \left\{ q_{i,0}\, \hat{\nu}_{i,0}^{t+1} + \sum_{\ell=1}^{\infty} q_{i,\ell}\, \hat{\nu}_{i,\ell}^{t} \right\} + \sum_{i \in \mathcal{N}} \mathbb{1}_{\{q_{i,0} \geq 1\}}\, \phi_i(S)\, \left[r_i + \pi_i - (1-\rho_{i,0})\,(\hat{\nu}_{i,0}^{t+1} - \hat{\nu}_{i,1}^{t+1}) \right] \\
&\leq \;\; \sum_{i \in \mathcal{N}} \hat{\nu}_{i,0}^{t+1}\, C_i + \sum_{i \in \mathcal{N}} \sum_{\ell=0}^{\infty} q_{i,\ell}\, \hat{\nu}_{i,\ell}^{t} + \sum_{i \in \mathcal{N}} \phi_i(\hat{A}^t)\, \left[r_i + \pi_i - (1-\rho_{i,0})\,(\hat{\nu}_{i,0}^{t+1} - \hat{\nu}_{i,1}^{t+1}) \right] \\
&= \;\; \sum_{i \in \mathcal{N}} \hat{\nu}_{i,0}^{t+1}\, C_i + \sum_{i \in \mathcal{N}} \sum_{\ell=0}^{\infty} q_{i,\ell}\, \hat{\nu}_{i,\ell}^{t} + \sum_{i \in \mathcal{N}} C_i\, (\hat{\nu}_{i,0}^{t} - \hat{\nu}_{i,0}^{t+1}) \\
&= \;\; \sum_{i \in \mathcal{N}} \hat{\nu}_{i,0}^{t}\, C_i + \sum_{i \in \mathcal{N}} \sum_{\ell=0}^{\infty} q_{i,\ell}\, \hat{\nu}_{i,\ell}^{t} \;\; = \;\; \hat{\beta}^t + \sum_{i \in \mathcal{N}} \sum_{\ell=0}^{\infty} q_{i,\ell}\, \hat{\nu}_{i,\ell}^{t} \;\; = \;\; \hat{J}^t(\boldsymbol{q}),
\end{aligned}$$

where the first inequality follows from the fact $\hat{\nu}_{i,0}^t \geq \hat{\nu}_{i,0}^{t+1}$ by Lemma 3.1, the first equality follows from the definition of $\hat{\nu}_{i,0}^t$ and the third equality follows from the definition of $\hat{\beta}^t$. By the chain of inequalities above, for any $\boldsymbol{q} \in \mathcal{Q}$, $S \in \mathcal{F}$, $t \in \mathcal{T}$, if we evaluate the right side of the constraint at $\{\hat{J}^t(\boldsymbol{q}) : \boldsymbol{q} \in \mathcal{Q},\ t \in \mathcal{T}\}$, then the right side of the constraint is upper bounded by $\hat{J}^t(\boldsymbol{q})$. Thus, the solution $\{\hat{J}^t(\boldsymbol{q}) : \boldsymbol{q} \in \mathcal{Q},\ t \in \mathcal{T}\}$ is feasible to the linear program, which implies that the objective value of the linear program evaluated at this solution is an upper bound on the optimal expected revenue. The objective value of the linear program evaluated at the solution $\{\hat{J}^t(\boldsymbol{q}) : \boldsymbol{q} \in \mathcal{Q},\ t \in \mathcal{T}\}$ is $\hat{J}^1\left(\sum_{i \in \mathcal{N}} C_i\, \boldsymbol{e}_{i,0}\right) = \hat{\beta}^1 + \sum_{i \in \mathcal{N}} \hat{\nu}_{i,0}^1\, C_i = 2\, \sum_{i \in \mathcal{N}} \hat{\nu}_{i,0}^1\, C_i$, where the last inequality uses the definition of $\hat{\beta}^1$. Thus, $2\, \sum_{i \in \mathcal{N}} \hat{\nu}_{i,0}^1\, C_i$ is an upper bound on the optimal total expected revenue. ∎

The greedy policy with respect to the value function approximations $\{\hat{J}^t : t \in \mathcal{T}\}$ offers the assortment $\hat{S}^t(\boldsymbol{q})$ when the system is in state $\boldsymbol{q}$ at time period t. Let $U^t(\boldsymbol{q})$ denote the total expected revenue under this greedy policy over the time periods $t, \ldots, T$, given that we are in state $\boldsymbol{q}$ at time period t. We can compute $\{U^t : t \in \mathcal{T}\}$ by using the recursion

$$\begin{aligned}
U^t(\boldsymbol{q}) &= \sum_{i \in \mathcal{N}} \pi_i \sum_{\ell=1}^{\infty} q_{i,\ell} \\
&+ \sum_{i \in \mathcal{N}} \mathbb{1}_{\{q_{i,0} \geq 1\}} \phi_i(\hat{S}^t(\boldsymbol{q})) \left(r_i + \pi_i + \mathbb{E}\Big\{ \mathsf{Z}(\rho_{i,0})\, U^{t+1}(\boldsymbol{X}(\boldsymbol{q})) + (1 - \mathsf{Z}(\rho_{i,0}))\, U^{t+1}\,(\boldsymbol{X}(\boldsymbol{q}) - \boldsymbol{e}_{i,0} + \boldsymbol{e}_{i,1}) \Big\} \right) \\
&\qquad + \Big(1 - \sum_{i \in \mathcal{N}} \mathbb{1}_{\{q_{i,0} \geq 1\}} \phi_i(\hat{S}^t(\boldsymbol{q})) \Big)\, \mathbb{E}\Big\{ U^{t+1}(\boldsymbol{X}(\boldsymbol{q})) \Big\},
\end{aligned}$$

with the boundary condition that $U^{T+1} = 0$. In the recursion above, we use the same line of reasoning that we used for the dynamic programming formulation in (2), but the decision is fixed as $\hat{S}^t(\boldsymbol{q})$. An observation that will shortly be useful is that U^{t+1} appears with a positive coefficient on the right side above. Therefore, if we replace U^{t+1} with a function H^{t+1} that satisfies

$U^{t+1}(\boldsymbol{q}) \geq H^{t+1}(\boldsymbol{q})$, then the right side of the expression above becomes smaller. By using the same sequence of manipulations that we used to obtain the dynamic program in (3), we can write the above recursion equivalently as

$$\begin{aligned} U^t(\boldsymbol{q}) \;=\; & \sum_{i\in\mathcal{N}} \pi_i \sum_{\ell=1}^{\infty} q_{i,\ell} \;+\; \mathbb{E}\Big\{U^{t+1}(\boldsymbol{X}(\boldsymbol{q}))\Big\} \\ & + \sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\geq 1\}}\, \phi_i(\hat{S}^t(\boldsymbol{q}))\Big(r_i+\pi_i-(1-\rho_{i,0})\,\mathbb{E}\Big\{U^{t+1}\left(\boldsymbol{X}(\boldsymbol{q})\right)-U^{t+1}\left(\boldsymbol{X}(\boldsymbol{q})-\boldsymbol{e}_{i,0}+\boldsymbol{e}_{i,1}\right)\Big\}\Big). \end{aligned} \quad (7)$$

The coefficients of U^{t+1} are not necessarily all positive on the right side above, but the last two recursions are equivalent. So, if we replace U^{t+1} on the right side above with a function H^{t+1} that satisfies $U^{t+1}(\boldsymbol{q}) \geq H^{t+1}(\boldsymbol{q})$, then the right side of the expression above still gets smaller.

Here is the proof of Theorem 3.2.

Proof of Theorem 3.2: For all $t\in\mathcal{T}$ and $\boldsymbol{q}\in\mathcal{Q}$, let $H^t(\boldsymbol{q}) = \sum_{i\in\mathcal{N}}\sum_{\ell=0}^{\infty} q_{i,\ell}\,\hat{\nu}^t_{i,\ell}$. We will use induction over the time periods to show that $U^t(\boldsymbol{q})\geq H^t(\boldsymbol{q})$ for all $\boldsymbol{q}\in\mathcal{Q}$ and $t\in\mathcal{T}$. By definition, $\hat{\nu}^{T+1}_{i,\ell}=0$ for all $i\in\mathcal{N}$, $\ell=0,1,\ldots$, so that $H^{T+1}=0$. Furthermore, we have $U^{T+1}=0$. Thus, the result holds at time period $T+1$. Assuming that $U^{t+1}(\boldsymbol{q})\geq H^{t+1}(\boldsymbol{q})$ for all $\boldsymbol{q}\in\mathcal{Q}$, we proceed to showing that $U^t(\boldsymbol{q})\geq H^t(\boldsymbol{q})$ for all $\boldsymbol{q}\in\mathcal{Q}$. Noting that H^{t+1} is linear, by using the same argument in the proof of Lemma 3.3, we have

$$\mathbb{E}\Big\{H^{t+1}(\boldsymbol{X}(\boldsymbol{q}))\Big\} = \sum_{i\in\mathcal{N}}\left\{q_{i,0}\,\hat{\nu}^{t+1}_{i,0}+\sum_{\ell=1}^{\infty} q_{i,\ell}\left[\rho_{i,\ell}\,\hat{\nu}^{t+1}_{i,0}+(1-\rho_{i,\ell})\,\hat{\nu}^{t+1}_{i,\ell+1}\right]\right\}.$$

Similarly, we have $\mathbb{E}\Big\{H^{t+1}(\boldsymbol{X}(\boldsymbol{q})) - H^{t+1}\left(\boldsymbol{X}(\boldsymbol{q})-\boldsymbol{e}_{i,0}+\boldsymbol{e}_{i,1}\right)\Big\} = \hat{\nu}^{t+1}_{i,0} - \hat{\nu}^{t+1}_{i,1}$. Thus, by the inductive hypothesis and the recursion defining $U^t(\boldsymbol{q})$ in (7), we have

$$\begin{aligned} U^t(\boldsymbol{q}) \;\geq\; & \sum_{i\in\mathcal{N}} \pi_i \sum_{\ell=1}^{\infty} q_{i,\ell} \;+\; \mathbb{E}\Big\{H^{t+1}(\boldsymbol{X}(\boldsymbol{q}))\Big\} \\ & + \sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\geq 1\}}\, \phi_i(\hat{S}^t(\boldsymbol{q}))\Big(r_i+\pi_i-(1-\rho_{i,0})\,\mathbb{E}\Big\{H^{t+1}\left(\boldsymbol{X}(\boldsymbol{q})\right)-H^{t+1}\left(\boldsymbol{X}(\boldsymbol{q})-\boldsymbol{e}_{i,0}+\boldsymbol{e}_{i,1}\right)\Big\}\Big) \\ =\; & \sum_{i\in\mathcal{N}} \pi_i \sum_{\ell=1}^{\infty} q_{i,\ell} \;+\; \sum_{i\in\mathcal{N}}\left\{q_{i,0}\,\hat{\nu}^{t+1}_{i,0}+\sum_{\ell=1}^{\infty} q_{i,\ell}\left[\rho_{i,\ell}\,\hat{\nu}^{t+1}_{i,0}+(1-\rho_{i,\ell})\,\hat{\nu}^{t+1}_{i,\ell+1}\right]\right\} \\ & + \sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\geq 1\}}\, \phi_i(\hat{S}^t(\boldsymbol{q}))\left[r_i+\pi_i-(1-\rho_{i,0})(\hat{\nu}^{t+1}_{i,0}-\hat{\nu}^{t+1}_{i,1})\right] \\ =\; & \sum_{i\in\mathcal{N}}\left\{q_{i,0}\,\hat{\nu}^{t+1}_{i,0}+\sum_{\ell=1}^{\infty} q_{i,\ell}\left[\pi_i+\rho_{i,\ell}\,\hat{\nu}^{t+1}_{i,0}+(1-\rho_{i,\ell})\,\hat{\nu}^{t+1}_{i,\ell+1}\right]\right\} \\ & + \max_{S\in\mathcal{F}}\sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\geq 1\}}\, \phi_i(S)\left[r_i+\pi_i-(1-\rho_{i,0})\,(\hat{\nu}^{t+1}_{i,0}-\hat{\nu}^{t+1}_{i,1})\right], \end{aligned}$$

where the last equality follows from the fact that $\hat{S}^t(\boldsymbol{q})$, by its definition, is an the optimal solution to the maximization problem on the right side above. By the definition of $\hat{\nu}^t_{i,\ell}$, we have

$\hat{\nu}^t_{i,\ell} = \pi_i + \rho_{i,\ell}\,\hat{\nu}^{t+1}_{i,0} + (1-\rho_{i,\ell})\,\hat{\nu}^{t+1}_{i,\ell+1}$. In this case, the expression on the right side of the chain of inequalities above can equivalently be written as

$$\begin{aligned}
&\sum_{i\in\mathcal{N}}\left\{q_{i,0}\,\hat{\nu}^{t+1}_{i,0} + \sum_{\ell=1}^{\infty} q_{i,\ell}\,\hat{\nu}^t_{i,\ell}\right\} \;+\; \max_{S\in\mathcal{F}} \sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\geq 1\}}\,\phi_i(S)\,\big[r_i + \pi_i - (1-\rho_{i,0})\,(\hat{\nu}^{t+1}_{i,0} - \hat{\nu}^{t+1}_{i,1})\big] \\
&\qquad \geq \;\sum_{i\in\mathcal{N}}\left\{q_{i,0}\,\hat{\nu}^{t+1}_{i,0} + \sum_{\ell=1}^{\infty} q_{i,\ell}\,\hat{\nu}^t_{i,\ell}\right\} \;+\; \sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\geq 1\}}\,\phi_i(\hat{A}^t)\,\big[r_i + \pi_i - (1-\rho_{i,0})\,(\hat{\nu}^{t+1}_{i,0} - \hat{\nu}^{t+1}_{i,1})\big] \\
&\qquad \geq \;\sum_{i\in\mathcal{N}}\left\{q_{i,0}\,\hat{\nu}^{t+1}_{i,0} + \sum_{\ell=1}^{\infty} q_{i,\ell}\,\hat{\nu}^t_{i,\ell}\right\} \;+\; \sum_{i\in\mathcal{N}} \frac{q_{i,0}}{C_i}\,\phi_i(\hat{A}^t)\,\big[r_i + \pi_i - (1-\rho_{i,0})\,(\hat{\nu}^{t+1}_{i,0} - \hat{\nu}^{t+1}_{i,1})\big\} \\
&\qquad = \;\sum_{i\in\mathcal{N}}\left\{q_{i,0}\,\hat{\nu}^{t+1}_{i,0} + \sum_{\ell=1}^{\infty} q_{i,\ell}\,\hat{\nu}^t_{i,\ell}\right\} \;+\; \sum_{i\in\mathcal{N}} q_{i,0}(\hat{\nu}^t_{i,0} - \hat{\nu}^{t+1}_{i,0}) \\
&\qquad = \;\sum_{i\in\mathcal{N}}\sum_{\ell=0}^{\infty} q_{i,\ell}\,\hat{\nu}^t_{i,\ell} \;=\; H^t(\boldsymbol{q}).
\end{aligned}$$

In the chain of inequalities above, to see that second inequality holds, by the discussion in the proof of Lemma 3.1, we have $\phi_i(\hat{A}^t)\,[r_i + \pi_i - (1-\rho_{i,0})\,\big(\hat{\nu}^{t+1}_{i,0} - \hat{\nu}^{t+1}_{i,1}\big)] \geq 0$ for all $i \in \mathcal{N}$. Furthermore, noting the definition of $\mathcal{Q}$, we have $q_{i,0} \leq C_i$ whenever $\boldsymbol{q} \in \mathcal{Q}$, which implies that $\mathbb{1}_{\{q_{i,0}\geq 1\}} \geq \frac{q_{i,0}}{C_i}$. The first equality follows from the definition of $\hat{\nu}^t_{i,0}$. The chain of inequalities above completes the induction argument so that we have $U^t(\boldsymbol{q}) \geq H^t(\boldsymbol{q})$ for all $\boldsymbol{q} \in \mathcal{Q}$ and $t \in \mathcal{T}$. As the initial state of the system is $\sum_{i\in\mathcal{N}} C_i\,\boldsymbol{e}_{i,0}$, the total expected revenue collected by the greedy policy is $U^1\left(\sum_{i\in\mathcal{N}} C_i \boldsymbol{e}_{i,0}\right)$. In this case, using the last inequality with $t=1$ and $\boldsymbol{q} = \sum_{i\in\mathcal{N}} C_i \boldsymbol{e}_{i,0}$, we obtain $U^1\left(\sum_{i\in\mathcal{N}} C_i\,\boldsymbol{e}_{i,0}\right) \geq H^1\left(\sum_{i\in\mathcal{N}} C_i\,\boldsymbol{e}_{i,0}\right) = \sum_{i\in\mathcal{N}} C_i\,\hat{\nu}^1_{i,0} \geq \frac{1}{2} J^1\left(\sum_{i\in\mathcal{N}} C_i\,\boldsymbol{e}_{i,0}\right)$, where the second inequality follows by Lemma 3.3. ■

The greedy policy with respect to the value function approximations $\{\hat{J}^t : t \in \mathcal{T}\}$ is guaranteed to obtain at least 50% of the optimal total expected revenue. In our computational experiments, we demonstrate that the practical performance of the policy can be substantially better than this theoretical performance guarantee. Nevertheless, despite having a performance guarantee, this greedy policy has a somewhat undesirable attribute. Consider two states $\boldsymbol{q} \in \mathcal{Q}$ and $\boldsymbol{q}' \in \mathcal{Q}$ such that $\{i \in \mathcal{N} : q_{i,0} \geq 1\} = \{i \in \mathcal{N} : q'_{i,0} \geq 1\}$. In other words, the set of products for which we have on-hand inventory is the same in the two states. In this case, noting the definition of $\hat{S}^t(\boldsymbol{q})$, we have $\hat{S}^t(\boldsymbol{q}) = \hat{S}^t(\boldsymbol{q}')$. Therefore, the decisions of the greedy policy depend on the set of products for which we have on-hand inventory, but not on the level of inventory for these products. The greedy policy does not differentiate between having too much or too little inventory of a product, as long as we have on-hand inventory for this product. In the next section, we develop a more sophisticated policy that explicitly takes the inventory levels into consideration, while maintaining the performance guarantee of the greedy policy. Our computational experiments demonstrate that the latter policy can perform noticeably better than the greedy policy.

4. Improving the Policy Performance through Rollout

To develop a policy that explicitly takes the inventory levels of the products into consideration, we build on a static policy that offers a fixed assortment at each time period. With the assortment $\hat{A}^t$ defined in (4), the static policy always offers the assortment $\hat{A}^t$ at time period t. Using an analysis similar to the one for the greedy policy with respect to the linear value function approximations discussed in the previous section, we show that the static policy obtains at least 50% of the optimal total expected revenue. Furthermore, the value functions associated with the static policy are separable by the products. We perform rollout on the static policy to obtain a policy that takes the inventory levels of the products into consideration, while maintaining the performance guarantee of the static policy. Exploiting the fact that the value functions associated with the static policy are separable by the products, we show that we can efficiently perform rollout on the static policy when the usage durations follow a negative binomial distribution or when the customers purchase the products outright without returning them at all.

4.1 Properties of the Static Policy

We consider a static policy that always offers the assortment $\hat{A}^t$ at time period t regardless of the product availabilities, where $\hat{A}^t$ is defined in (4). If a customer chooses a product that is not available, then she leaves the system. In the next lemma, we show that the static policy obtains at least 50% of the optimal total expected revenue. The proof is similar to the analysis of the greedy policy with respect to the linear value function approximations. The details are in Appendix B.

Lemma 4.1 (Performance of the Static Policy) *The total expected revenue of the static policy that offers assortment $\hat{A}^t$ at time period t is at least 50% of the optimal total expected revenue.*

Let $V^t(\boldsymbol{q})$ denote the total expected revenue under the static policy over the time periods $t, \ldots, T$, given that we are in state $\boldsymbol{q}$ at time period t. Similar to the dynamic program in (3), we can compute $\{V^t : t \in \mathcal{T}\}$ by using the recursion

$$\begin{aligned} V^t(\boldsymbol{q}) = \sum_{i \in \mathcal{N}} \pi_i \sum_{\ell=1}^{\infty} q_{i,\ell} \; + \; \mathbb{E}\Big\{ V^{t+1}(\boldsymbol{X}(\boldsymbol{q})) \Big\} \\ + \sum_{i \in \mathcal{N}} \mathbb{1}_{\{q_{i,0} \geq 1\}} \, \phi_i(\hat{A}^t) \bigg(r_i + \pi_i - (1 - \rho_{i,0}) \mathbb{E}\Big\{ V^{t+1}(\boldsymbol{X}(\boldsymbol{q})) - V^{t+1}\left(\boldsymbol{X}(\boldsymbol{q}) - \boldsymbol{e}_{i,0} + \boldsymbol{e}_{i,1}\right) \Big\} \bigg), \end{aligned}$$

with the boundary condition that $V^{T+1} = 0$. The following lemma shows that $V^t(\boldsymbol{q})$ decomposes by products. The proof is in Appendix C. To facilitate our exposition, let $\boldsymbol{q}_i = (q_{i,\ell} \,:\, \ell = 0, 1, \ldots)$ denote the numbers of units of product i that have been in use for different numbers of time periods. By (1), we observe that the state of the units of product i in the next time period depends on the state of the units of product i in the current time period, but not on the state of the units for other products. Therefore, $X_{i,\ell}(\boldsymbol{q})$ is a function of $\boldsymbol{q}_i$ only, which implies that we can write $X_{i,\ell}(\boldsymbol{q})$ as $X_{i,\ell}(\boldsymbol{q}_i)$, in which case, we can define the vector $\boldsymbol{X}_i(\boldsymbol{q}_i) = (X_{i,\ell}(\boldsymbol{q}_i) : \ell = 0, 1, \ldots)$.

Lemma 4.2 (Decomposability by Products) *For each $t \in \mathcal{T}$ and $\boldsymbol{q} \in \mathcal{Q}$, we have $V^t(\boldsymbol{q}) = \sum_{i\in\mathcal{N}} V_i^t(\boldsymbol{q}_i)$, where for each $i \in \mathcal{N}$, $\{V_i^t : t \in \mathcal{T}\}$ is computed by using the recursion*

$$\begin{aligned} V_i^t(\boldsymbol{q}_i) = \pi_i \sum_{\ell=1}^{\infty} q_{i,\ell} \;+\; \mathbb{E}\Big\{ V_i^{t+1}(\boldsymbol{X}_i(\boldsymbol{q}_i)) \Big\} \\ + \mathbb{1}_{\{q_{i,0}\geq 1\}}\, \phi_i(\hat{A}^t) \bigg(r_i + \pi_i - (1-\rho_{i,0})\, \mathbb{E}\Big\{ V_i^{t+1}(\boldsymbol{X}_i(\boldsymbol{q}_i)) - V_i^{t+1}\left(\boldsymbol{X}_i(\boldsymbol{q}_i) - \boldsymbol{e}_0 + \boldsymbol{e}_1\right) \Big\} \bigg), \end{aligned} \tag{8}$$

with the boundary condition that $V_i^{T+1} = 0$.

4.2 Rollout Policy Based on the Static Policy

We perform rollout on the static policy to obtain a policy that takes the inventory levels of the products into consideration. To perform rollout on the static policy, given that we are in a particular state in the current time period, we choose the decision that maximizes the immediate expected revenue in the current time period plus the expected revenue from the static policy starting from the state in the next time period. We refer to the policy obtained by performing rollout on the static policy as the rollout policy. The rollout policy ultimately corresponds to using $V^t(\boldsymbol{q}) = \sum_{i\in\mathcal{N}} V_i^t(\boldsymbol{q}_i)$ as a separable nonlinear approximation to $J^t(\boldsymbol{q})$. Let $S^t_{\mathsf{rollout}}(\boldsymbol{q})$ be the assortment offered by the rollout policy given that we are in state $\boldsymbol{q}$ at time period t. As $V^{t+1}(\boldsymbol{q})$ is the total expected revenue obtained by the static policy starting in state $\boldsymbol{q}$ at time period $t+1$, $S^t_{\mathsf{rollout}}(\boldsymbol{q})$ is given by

$$\begin{aligned} &S^t_{\mathsf{rollout}}(\boldsymbol{q}) \\ &= \arg\max_{S\in\mathcal{F}} \Bigg\{ \sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\geq 1\}}\, \phi_i(S) \bigg(r_i + \pi_i + \mathbb{E}\Big\{ \mathsf{Z}(\rho_{i,0})\, V^{t+1}(\boldsymbol{X}(\boldsymbol{q})) + (1-\mathsf{Z}(\rho_{i,0}))\, V^{t+1}(\boldsymbol{X}(\boldsymbol{q}) - \boldsymbol{e}_{i,0} + \boldsymbol{e}_{i,1}) \Big\} \bigg) \\ &\qquad\qquad + \bigg(1 - \sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\geq 1\}} \phi_i(S) \bigg) \mathbb{E}\Big\{ V^{t+1}(\boldsymbol{X}(\boldsymbol{q})) \Big\} \Bigg\} \\ &= \arg\max_{S\in\mathcal{F}} \; \sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\geq 1\}}\, \phi_i(S) \bigg(r_i + \pi_i - (1-\rho_{i,0})\, \mathbb{E}\Big\{ V^{t+1}(\boldsymbol{X}(\boldsymbol{q})) - V^{t+1}\left(\boldsymbol{X}(\boldsymbol{q}) - \boldsymbol{e}_{i,0} + \boldsymbol{e}_{i,1}\right) \Big\} \bigg) \\ &= \arg\max_{S\in\mathcal{F}} \; \sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\geq 1\}} \phi_i(S) \bigg(r_i + \pi_i - (1-\rho_{i,0})\, \mathbb{E}\Big\{ V_i^{t+1}(\boldsymbol{X}_i(\boldsymbol{q}_i)) - V_i^{t+1}\left(\boldsymbol{X}_i(\boldsymbol{q}_i) - \boldsymbol{e}_0 + \boldsymbol{e}_1\right) \Big\} \bigg). \end{aligned}$$

In the first equality above, we follow the same argument that we used to construct the dynamic program in (2), in which we find an assortment that maximizes the immediate expected revenue and the expected value function in the next time period under the optimal policy, but we use the value function of the static policy in the next time period above. The second equality is by the same reasoning that we used to obtain the dynamic program in (3) from the dynamic program in (2). The third equality follows from the fact that the value functions of the static policy decompose by the products, as shown in Lemma 4.2.

It is a well-known result that the policy obtained by performing rollout on a base policy always performs at least as well as the base policy; see Section 6.1.3 in Bertsekas and Tsitsiklis (1996). Therefore, the total expected revenue obtained by our rollout policy is at least as large as the total expected revenue obtained by the static policy. By Lemma 4.1, the rollout policy thus obtains at least 50% of the optimal total expected revenue as well. In many applications, a policy based on rollout tends to offer a dramatic improvement over the base policy. The key question is whether the rollout assortment $S^t_{\mathsf{rollout}}(\boldsymbol{q})$ can be computed efficiently. Lemma 4.2 shows that the value function of the static policy is separable by the products, indicating that computing the value functions of the static policy through the recursion in (8) is more manageable than computing the value functions of the optimal policy through the dynamic program in (3). As discussed earlier, without loss of generality, we can assume that the vector $\boldsymbol{q}_i = (q_{i,\ell} : \ell = 0, 1, \ldots)$ is finite-dimensional, because we start with no units in use so that we always have have $q_{i,\ell} = 0$ for all $\ell \geq T$. However, the state variable $\boldsymbol{q}_i = (q_{i,\ell} : \ell = 0, 1, \ldots)$ in the recursion in (8) is still a high-dimensional vector. In particular, the state space in this recursion is given by $\mathcal{Q}_i = \{(q_{i,\ell} \in \mathbb{Z}_+ : \ell = 0, 1, \ldots) \mid \sum_{\ell=0}^{\infty} q_{i,\ell} = C_i\}$ and computing the value function $V_i^t(\boldsymbol{q}_i)$ of the static policy for all $\boldsymbol{q}_i \in \mathcal{Q}_i$ is difficult.

In the remainder of this section, we consider two cases. First, if the usage duration follows a negative binomial distribution, then the value functions of the static policy can be computed efficiently. Second, if the customers purchase the products outright and never return them, then the value functions of the static policy can be computed efficiently as well. Once we compute the value functions $\{V^t : t \in \mathcal{T}\}$ of the static policy efficiently, we can solve the maximization problem above that defines $S^t_{\mathsf{rollout}}(\boldsymbol{q})$ to find the assortment offered by the rollout policy. Note that the maximization problem that we solve to obtain the assortment $S^t_{\mathsf{rollout}}(\boldsymbol{q})$ has the same structure as the maximization problem on the right side of the dynamic program in (3). Thus, once we compute the value functions $\{V^t : t \in \mathcal{T}\}$ of the static policy, as discussed at the end of Section 2, there are numerous choice models that render this maximization problem tractable. Lastly, we emphasize that even if we cannot compute the value functions $\{V^t : t \in \mathcal{T}\}$ of the static policy, we can use simulation to estimate the expected revenue of the static policy, which still allows performing rollout on the static policy. Section 6.1.3 in Bertsekas and Tsitsiklis (1996) discusses using simulation to perform rollout. Naturally, the computational requirements of performing rollout inflate when we use simulation to estimate the total expected revenue of the static policy. Next, we discuss how to perform rollout efficiently when the usage durations have a negative binomial distribution.

4.3 Negative Binomial Usage Duration

In this section, we assume that for each product $i \in \mathcal{N}$, the usage duration is given $\mathsf{Duration}_i = 1 + \mathsf{NegBin}(s_i, \eta_i)$, where $\mathsf{NegBin}(s_i, \eta_i)$ denotes a negative binomial random variable with parameters $s_i \in \mathbb{Z}_{++}$ and $\eta_i \in (0,1)$ taking values over $\{0, 1, \ldots\}$. A negative binomial random variable with parameters (s_i, η_i) corresponds to the sum of s_i independent geometric random variables, each with parameter η_i. Thus, a negative binomial random variable with parameters $(1, \eta_i)$ is equivalent to a geometric random variable with parameter η_i. As s_i increases, the probability mass function of a negative binomial random variable with parameters (s_i, η_i) becomes more symmetric. Even with $s_i = 3$, the probability mass function is rather symmetric. Therefore, a negative binomial random variable is quite flexible for modeling usage durations. Noting that a negative binomial random variable with parameters (s_i, η_i) corresponds to the sum of s_i geometric random variables, we provide the following interpretation for our use of a negative binomial random variable for modeling the usage durations. At each time period, a customer is satisfied with product i with probability η_i. As soon as a customer is dissatisfied with the product for s_i times, she returns the product back, ending her rental duration. Naturally, we do not advocate this interpretation as a model of how customers make a decision for keeping the product, but this interpretation provides us the vocabulary to explain our model more clearly as follows. If the usage durations have negative binomial distributions, then our state variable does not need to keep track of the numbers of units of each product i that have been in use for a certain duration of time. It is enough to use a state variable that keeps track of the numbers of customers who are using each product i and have been dissatisfied for a certain number of times. In this case, we can efficiently compute the value functions of the static policy, as long as s_i is relatively small.

Next, we discuss how we can compute the value functions of the static policy by using a recursion similar to the one in (8) when the usage durations are negative binomial random variables.

State and Transition Dynamics: To compute the value functions of the static policy through a recursion similar to the one in (8), we define

$$w_{i,d} \; = \; \text{number of customers who are using product } i \text{ and have been dissatisfied for } d \text{ times.}$$

A customer using product i returns the product once she has been dissatisfied for s_i times, in which case, the product becomes available on-hand. Therefore, the s_i-dimensional vector $(w_{i,0}, \ldots, w_{i,s_i-1})$ captures the state of the customers using product i. The on-hand inventory of product i is given by $C_i - \sum_{d=0}^{s_i-1} w_{i,d}$. Under negative binomial usage durations, we use $\boldsymbol{w}_i = (w_{i,d} : 0 \le d \le s_i - 1)$ to denote the state vector for product i in the current time period. With this state representation, if no purchase is made at the current time period, then the new random state $\boldsymbol{F}_i(\boldsymbol{w}_i) = (F_{i,d}(\boldsymbol{w}_i) \,:\, 0 \le d \le s_i - 1)$ at the next time period is given by

$$F_{i,d}(\boldsymbol{w}_i) = \begin{cases} \mathsf{Bin}(w_{i,0}, \eta_i) & \text{if} \quad d = 0, \\ \mathsf{Bin}(w_{i,d}, \eta_i) + (w_{i,d-1} - \mathsf{Bin}(w_{i,d-1}, \eta_i)) & \text{if} \quad d = 1, 2, \ldots, s_i - 1, \end{cases}$$

where we use the fact that for each d, the number of customers who continue to remain dissatisfied for d times at the next time period is equal to $\mathsf{Bin}(w_{i,d}, \eta_i)$ since each customer is satisfied with the product with probability η_i, independently of each other. Furthermore, $w_{i,d-1} - \mathsf{Bin}(w_{i,d-1}, \eta_i)$ captures the number of customers who were dissatisfied for $d-1$ times at the beginning of the current time period and they were dissatisfied one more time in the current time period, in which case, these customers are dissatisfied d times in the next time period. These customers add up to the number of customers dissatisfied d times in the next time period.

With this state representation, we can compute the value functions of the static policy for each product i through the following recursion. We use $\boldsymbol{w}_i = (w_{i,0}, \ldots, w_{i,s_i-1})$ to capture the state of product i. Recall that the static policy offers the assortment $\hat{A}^t$ at each time period t. Given that the state of product i at time period t is $\boldsymbol{w}_i$, let $V_i^t(\boldsymbol{w}_i)$ be the total expected revenue from product i under the static policy over the time periods $t, \ldots, T$. Using the vectors $\boldsymbol{e}_0 = (1, 0, 0, \ldots, 0) \in \mathbb{R}^{s_i}$ and $\boldsymbol{e}_1 = (0, 1, 0, \ldots, 0) \in \mathbb{R}^{s_i}$, we can compute $\{V_i^t : t \in \mathcal{T}\}$ by using the recursion

$$\begin{aligned} V_i^t(\boldsymbol{w}_i) &= \pi_i \sum_{d=0}^{s_i-1} w_{i,d} + \left(1 - \mathbb{1}_{\left\{\sum_{d=0}^{s_i-1} w_{i,d} \,<\, C_i\right\}} \phi_i(\hat{A}^t)\right) \mathbb{E}\left\{ V_i^{t+1}\left(\boldsymbol{F}_i(\boldsymbol{w}_i)\right) \right\} \\ &\quad + \mathbb{1}_{\left\{\sum_{d=0}^{s_i-1} w_{i,d} \,<\, C_i\right\}} \phi_i(\hat{A}^t) \left(r_i + \pi_i + \eta_i \, \mathbb{E}\left\{ V_i^{t+1}\left(\boldsymbol{F}_i(\boldsymbol{w}_i) + \boldsymbol{e}_0\right) \right\} + (1 - \eta_i) \, \mathbb{E}\left\{ V_i^{t+1}\left(\boldsymbol{F}_i(\boldsymbol{w}_i) + \boldsymbol{e}_1\right) \right\} \right) \\ &= \pi_i \sum_{d=0}^{s_i-1} w_{i,d} + \mathbb{E}\left\{ V_i^{t+1}\left(\boldsymbol{F}_i(\boldsymbol{w}_i)\right) \right\} \\ &\quad + \mathbb{1}_{\left\{\sum_{d=0}^{s_i-1} w_{i,d} \,<\, C_i\right\}} \phi_i(\hat{A}^t) \left(r_i + \pi_i - \eta_i \, \mathbb{E}\left\{ V_i^{t+1}\left(\boldsymbol{F}_i(\boldsymbol{w}_i)\right) - V_i^{t+1}\left(\boldsymbol{F}_i(\boldsymbol{w}_i) + \boldsymbol{e}_0\right) \right\} \right. \\ &\qquad\qquad \left. - (1 - \eta_i) \, \mathbb{E}\left\{ V_i^{t+1}\left(\boldsymbol{F}_i(\boldsymbol{w}_i)\right) - V_i^{t+1}\left(\boldsymbol{F}_i(\boldsymbol{w}_i) + \boldsymbol{e}_1\right) \right\} \right), \end{aligned}$$

with the boundary condition that $V_i^{T+1} = 0$. In the first equality above, for a customer to rent a unit of product i, we need to have product i available on-hand and the customer needs to choose product i. The number of units of product i available on-hand is given by $C_i - \sum_{d=0}^{s_i-1} w_{i,d}$, so the expression $1 - \mathbb{1}_{\left\{\sum_{d=0}^{s_i-1} w_{i,d} \,<\, C_i\right\}} \phi_i(\hat{A}^t)$ captures the probability that a customer does not rent product i when we offer the assortment $\hat{A}^t$. If $\sum_{d=0}^{s_i-1} w_{i,d} < C_i$, then we have product i available on-hand. If the customer chooses product i, then she rents this product. With probability η_i, the customer renting product i in the current time period is satisfied and she ends up being a customer with no dissatisfactions at the beginning of the next time period. With probability $1 - \eta_i$, the customer renting product i in the current time period is dissatisfied and she becomes a customer who is dissatisfied for once at the beginning of next time period. The second equality follows by arranging the terms. If $s_i = 1$, so that the usage durations for product i are geometric random variables, then the recursion above continues to hold as long as we set $\boldsymbol{e}_0 = 1$ and $\boldsymbol{e}_1 = 0$

In the recursion above, the state variable is an s_i-dimensional vector $(w_{i,0}, \ldots, w_{i,s_i-1})$ such that $\sum_{d=0}^{s_i-1} w_{i,d} \le C_i$, so the number of states is $O((s_i + C_i)^{s_i})$. Thus, when s_i is relatively small, we can compute the value functions of the static policy efficiently. For example, in our computational experiments, using transaction data from the city of Seattle, we find that the negative binomial distribution provides a reasonably good model for the amount of time that drivers park their vehicles at a meter. In our experiments, the best fitted value for the parameter s_i was 2.

4.4 Infinite Usage Duration

In this section, we focus on the case in which the usage duration is infinity. This case corresponds to the situation where the customers buy the products outright, never returning them. Infinite usage durations have a number of interesting applications. In the retail setting, customers make purchases among substitutable products, in which case, our model dynamically makes product assortment offerings to each individual customer as a function of the remaining product inventories (Topaloglu 2013, Golrezaei et al. 2014). Also, an important class of revenue management problems occurs on a flight network with parallel flights operating between the same origin-destination pair. In this setting, the customers make a purchase among multiple parallel flights on a particular departure date. Our model captures dynamically adjusts the assortment of flights offered to each individual customer as a function of the remaining flight capacities (Zhang and Cooper 2005, Liu and van Ryzin 2008, Dai et al. 2014). We proceed to discussing how we can compute the value functions of the static policy by using a recursion similar to the one in (8). As the products are purchased outright, we assume that $\pi_i = 0$ so that there is no per-period fee. Since the products are not returned, we only need to keep track of the on-hand inventory of product i. We use $q_{i,0}$ to denote the number of units of product i on-hand. Given that we have $q_{i,0}$ units of product i on-hand, let $V_i^t(q_{i,0})$ be the total expected revenue from product i under the static policy over the time periods $t, \ldots, T$. We can compute $\{V_i^t : t \in \mathcal{T}\}$ by using the recursion

$$\begin{aligned} V_i^t(q_{i,0}) &= \left(1 - \mathbb{1}_{\{q_{i,0} \ge 1\}} \phi_i(\hat{A}^t)\right) V_i^{t+1}(q_{i,0}) + \mathbb{1}_{\{q_{i,0} \ge 1\}}\, \phi_i(\hat{A}^t) \left(r_i + V_i^{t+1}(q_{i,0} - 1)\right) \\ &= V_i^{t+1}(q_{i,0}) + \mathbb{1}_{\{q_{i,0} \ge 1\}}\, \phi_i(\hat{A}^t) \left(r_i - \left\{V_i^{t+1}(q_{i,0}) - V_i^{t+1}(q_{i,0} - 1)\right\}\right), \end{aligned}$$

with the boundary condition that $V_i^{T+1} = 0$. In the first equality above, if we have on-hand units of product i and a customer chooses product i, then she makes a purchase for product i, in which case, we have one less on-hand unit of product i in the next time period. The second equality follows by arranging the terms. As the state variable in the recursion above is scalar, we can efficiently compute the value functions of the static policy under infinite usage durations.

Thus, under both negative binomial and infinite usage durations, we can efficiently perform rollout on the static policy to obtain a policy that takes the inventory levels of the products into consideration, while still obtaining at least 50% of the optimal total expected revenue.

5. Extensions

We discuss extensions to the case in which we have multiple customer types, we make pricing decisions instead of assortment offering decisions, and we can solve the assortment optimization problems only approximately. We show that our earlier performance guarantees continue to hold when we have multiple customer types and when we make pricing decisions. Furthermore, we show that if we can solve the assortment optimization problems approximately, then our performance guarantees hold with appropriate modifications to reflect the solution accuracy in the assortment problems. Some of these extensions are used in our computational experiments.

5.1 Heterogeneous Customer Types

We have m customer types indexed by $\mathcal{M} = \{1, 2, \ldots, m\}$. At time period $t \in \mathcal{T}$, a customer of type j arrives with probability $p^{t,j}$, where we have $\sum_{j \in \mathcal{M}} p^{t,j} = 1$, so that each time period has exactly one customer arrival. We observe the type of each arriving customer. Each customer type has its own choice model, reward structure, assortment constraints, and usage duration. Therefore, a customer of type j chooses a product according to a discrete choice model $\{\phi_i^j(S) : i \in \mathcal{N},\ S \subseteq \mathcal{N}\}$, where $\phi_i^j(S)$ is the probability that a customer of type j chooses product i when we offer the subset S of products. Note that if we do not observe the type of each arriving customer, then we can continue using the model in Section 2, where the discrete choice model $\{\phi_i(S) : i \in \mathcal{N},\ S \subseteq \mathcal{N}\}$ is obtained by mixing the choice models corresponding to different customer types. If a customer of type j selects product i, then she pays an upfront fee of r_i^j and a fee of π_i^j for each time period she rents the product. The usage duration of product i by a customer of type j is given by the random variable $\mathsf{Duration}_i^j$. We let $\rho_{i,\ell}^j$ be the hazard rate of the usage duration of product i for a customer of type j, which is defined by $\rho_{i,\ell}^j \;=\; \Pr\{\mathsf{Duration}_i^j = \ell + 1 \mid \mathsf{Duration}_i^j \geq \ell + 1\}$. Lastly, the assortments offered to customers of different types have different feasibility requirements. We use $\mathcal{F}^j$ to denote the set of feasible assortments that can be offered to customers of type j.

We can extend all of our results to the case with heterogeneous customer types. Here, we focus on the essentials and we defer the details to Appendix D. To capture the state of the system, because each customer type has its own reward structure and usage duration, we need to keep track of the number of units that are currently in use by each customer type. We use $q_{i,0}$ to denote the number of units of product i on-hand. For $\ell \geq 1$, we use $q^j_{i,\ell}$ to denote the number of units of product i that have been used for exactly ℓ time periods by a customer of type j. Therefore, we can describe the state of the system by using the vector $\boldsymbol{q} = (q_{i,0}, q^j_{i,\ell} : i \in \mathcal{N},\ j \in \mathcal{M},\ \ell \geq 1)$. Using $\boldsymbol{q}$ as the state variable, we can give a dynamic programming formulation of the problem that resembles the one in (2). In this case, we use value function approximations of the form

$$\hat{J}^t(\boldsymbol{q}) = \hat{\beta}^t + \sum_{i \in \mathcal{N}} \hat{\theta}^t_i \, q_{i,0} + \sum_{i \in \mathcal{N}} \sum_{j \in \mathcal{M}} \sum_{\ell=1}^{\infty} \hat{\nu}^{t,j}_{i,\ell} \, q^j_{i,\ell},$$

where $\hat{\theta}^t_i$ captures the marginal value of a unit of product i on-hand at time period t and $\hat{\nu}^{t,j}_{i,\ell}$ captures the marginal value of a unit of product i that has been in use for ℓ periods by a customer of type j at time period t. Similar to our approach in Section 3.1, we propose computing $\hat{\beta}^t$, $\hat{\theta}^t_i$ and $\hat{\nu}^{t,j}_{i,\ell}$ recursively as follows.

- **Initialization:** Set $\hat{\theta}^{T+1}_i = 0$, $\hat{\nu}^{T+1,j}_{i,\ell} = 0$ for all $i \in \mathcal{N},\ j \in \mathcal{M},\ \ell \geq 1$ and set $\hat{\beta}^{T+1} = 0$.
- **Recursion:** For $t = T, T-1, \ldots, 1$, we compute $\hat{\theta}^t_i$, $\hat{\nu}^{t,j}_{i,\ell}$ and $\hat{\beta}^t$ by using $\{\hat{\theta}^{t+1}_i : i \in \mathcal{N}\}$ and $\{\hat{\nu}^{t+1,j}_{i,\ell} : i \in \mathcal{N},\ j \in \mathcal{M},\ \ell \geq 1\}$ as follows. For each $j \in \mathcal{M}$, let $\hat{A}^{t,j} \in \mathcal{F}^j$ be such that

$$\hat{A}^{t,j} \;=\; \arg\max_{S \in \mathcal{F}^j} \sum_{i \in \mathcal{N}} \phi^j_i(S) \Big[r^j_i + \pi^j_i - (1 - \rho^j_{i,0}) \left(\hat{\theta}^{t+1}_i - \hat{\nu}^{t+1,j}_{i,1} \right) \Big].$$

Once $\hat{A}^{t,j}$ is computed for all $j \in \mathcal{M}$, for each $i \in \mathcal{N}$ and $j \in \mathcal{M}$, let

$$\begin{aligned}
\hat{\theta}^t_i \;&=\; \hat{\theta}^{t+1}_i + \frac{1}{C_i} \sum_{j \in \mathcal{M}} p^{t,j} \phi^j_i(\hat{A}^{t,j}) \Big[r^j_i + \pi^j_i - (1 - \rho^j_{i,0}) \left(\hat{\theta}^{t+1}_i - \hat{\nu}^{t+1,j}_{i,1} \right) \Big] \qquad (9) \\
\hat{\nu}^{t,j}_{i,\ell} \;&=\; \pi^j_i + \rho^j_{i,\ell} \, \hat{\theta}^{t+1}_i + (1 - \rho^j_{i,\ell}) \, \hat{\nu}^{t+1,j}_{i,\ell+1} \qquad \forall \, \ell = 1, 2, \ldots \\
\hat{\beta}_t \;&=\; \sum_{i \in \mathcal{N}} C_i \, \hat{\theta}^t_i.
\end{aligned}$$

The above discussion completes the specification of the approximate value function $\hat{J}^t$. The intuition for the above specification of the parameters is similar to the one discussed in Section 3.1. Using an argument similar to the one in the previous two sections, we can show that the greedy policy with respect to value function approximations $\{\hat{J}^t : t \in \mathcal{T}\}$ obtains at least 50% of the optimal total expected revenue. We can also perform rollout on a static policy to obtain a policy that takes the inventory levels of the products into consideration, while ensuring that we still obtain at least 50% of the optimal total expected revenue. We give both of these results in Appendix D.

5.2 Price Optimization with Discrete Prices

So far in the paper, we assume that the upfront and per period rental fees for the products are fixed and we decide on the set of products to make available to the customers. It is not difficult to adopt our results to the case in which we decide the upfront and per period rental fees for the products and the customers choose based on the prices we charge. In particular, we create multiple copies of each product i, where the different copies correspond to charging different prices for product i. We call each copy of a product a virtual product. Let $\mathcal{H}$ denote the set of possible copies of each product. We write $(i,h) \in \mathcal{N} \times \mathcal{H}$ to denote copy h of product i. Thus, the pairs $\{(i,h) : i \in \mathcal{N},\ h \in \mathcal{H}\}$ are the set of all virtual products that we can offer to the customers. Offering virtual product (i,h) means that we offer product i at the price level corresponding to copy h of this product. In this case, the question becomes that of choosing an assortment of virtual products to offer at each time period to maximize the total expected revenue. As we can offer a product at no more than one price level, among all virtual copies of a particular product, we can offer at most one virtual copy. Thus, the set of possible assortments of virtual products that we can offer at each time period is given by $\mathcal{F} = \{S \subseteq \mathcal{N} \times \mathcal{H} : |S \cap (\{i\} \times \mathcal{H})| \le 1 \ \ \forall\ i \in \mathcal{N}\}$. Using $r_{i,h}$ to denote the upfront fee when we charge the price level corresponding to copy h for product i, and $\pi_{i,h}$ to denote the per period fee when we charge the price level corresponding to copy h of product i, we can follow the same outline in the previous two sections to come up with a policy that is guaranteed to obtain at least 50% of the optimal total expected revenue. The only difference is that we treat the virtual products $\mathcal{N} \times \mathcal{H}$ as the products.

5.3 Solving the Assortment Optimization Problem Approximately

The maximization problem in (3) is a combinatorial optimization problem. Under many choice models, we can solve this problem tractably, but it is not possible to solve this problem tractably under every choice model. In this section, we discuss how we can adopt our approach in principle to the case where we have a fully polynomial-time approximation scheme (FPTAS) for problem (3). For any $\epsilon > 0$, the FPTAS returns a $1/(1+\epsilon)$-approximate solution to problem (3) and the running time to do so is polynomial in n and $1/\epsilon$. It turns out that we can leverage the FPTAS to obtain a $1/(2(1+\epsilon))$-approximate policy and the running time to obtain and execute the approximate policy is polynomial in n, $1/\epsilon$ and T. In particular, assume that we have an FPTAS such that for any $\epsilon > 0$, the FPTAS finds an assortment $\hat{A}^t$ satisfying

$$(1+\epsilon) \sum_{i \in \mathcal{N}} \phi_i(\hat{A}^t) \Big[r_i + \pi_i - (1-\rho_{i,0}) \left(\hat{\nu}_{i,0}^{t+1} - \hat{\nu}_{i,1}^{t+1} \right) \Big] \ge \max_{S \in \mathcal{F}} \sum_{i \in \mathcal{N}} \phi_i(S) \Big[r_i + \pi_i - (1-\rho_{i,0}) \left(\hat{\nu}_{i,0}^{t+1} - \hat{\nu}_{i,1}^{t+1} \right) \Big]$$

in running time that is polynomial in n and $1/\epsilon$. In the next theorem, we show how to leverage this FPTAS to find a $1/(2(1+\epsilon))$-approximate policy. The proof is in Appendix E.

Theorem 5.1 (Approximate Solution) *Assume that for any $\epsilon > 0$, we can find a $1/(1+\epsilon)$-approximate solution to problem* (3) *in running time that is polynomial in n and $1/\epsilon$. Then, we can construct value function approximations $\{\hat{J}^t : t \in \mathcal{T}\}$ such that the greedy policy with respect to these value function approximations is a $1/(2(1+\epsilon))$-approximate policy and the running time to obtain and execute the approximate policy is polynomial in n, $1/\epsilon$ and T.*

6. Computational Experiments

In Section 6.1, we describe an approach to obtain an upper bound on the optimal total expected revenue, which is useful for assessing the optimality gaps of our policies. In Sections 6.2 and 6.3, we give our computational results on parallel flights and pricing parking spaces in the city of Seattle.

6.1 Upper Bound on the Optimal Total Expected Revenue

To compute an upper bound on the optimal total expected revenue, we formulate a linear program, in which the choices of the customers and the transition dynamics take on their expected values. We use the decision variables $(z^t(A) : A \in \mathcal{F},\ t \in \mathcal{T})$ and $(q^t_{i,\ell} : i \in \mathcal{N},\ \ell \geq 0,\ t \in \mathcal{T})$, where $z^t(A)$ is the frequency with which we offer assortment A at time period t and $q^t_{i,\ell}$ is the expected number of units of product i that have been in use for exactly ℓ time periods at time period t. To construct the constraints in our linear program, noting the dynamic programming formulation in (2), if the state of the system at time period t is $\boldsymbol{q}^t = (q^t_{i,\ell} : i \in \mathcal{N},\ \ell \geq 0)$ and the customer arriving at this time period chooses product i, then the state of the system at the next time period is given by the random variable $\mathsf{Z}(\rho_{i,0})\, \boldsymbol{X}(\boldsymbol{q}^t) + (1 - \mathsf{Z}(\rho_{i,0}))\,(\boldsymbol{X}(\boldsymbol{q}^t) - \boldsymbol{e}_{i,0} + \boldsymbol{e}_{i,1})$, where $\mathsf{Z}(\rho)$ is a Bernoulli random variable with parameter ρ. If the customer does not choose any product, then the state of the system is $\boldsymbol{X}(\boldsymbol{q}^t)$. Furthermore, if we offer the assortment A at time period t with frequency $z^t(A)$, then the probability that a customer chooses product i is $\sum_{A \in \mathcal{F}} \phi_i(A)\, z^t(A)$. In this case, if the state of the system at time period t is $\boldsymbol{q}^t$ and we offer assortment A with frequency $z^t(A)$, then the expected state of the system at the next time period is given by $\sum_{i \in \mathcal{N}} \left\{\sum_{A \in \mathcal{F}} \phi_i(A)\, z^t(A)\right\} \mathbb{E}\left\{\mathsf{Z}(\rho_{i,0})\, \boldsymbol{X}(\boldsymbol{q}^t) + (1 - \mathsf{Z}(\rho_{i,0}))\,(\boldsymbol{X}(\boldsymbol{q}^t) - \boldsymbol{e}_{i,0} + \boldsymbol{e}_{i,1})\right\}$ + $\left\{1 - \sum_{i \in \mathcal{N}} \left\{\sum_{A \in \mathcal{F}} \phi_i(A)\, z^t(A)\right\}\right\} \mathbb{E}\left\{\boldsymbol{X}(\boldsymbol{q}^t)\right\}$. Using the fact that $\mathbb{E}\{\mathsf{Z}(\rho_{i,0})\} = 1 - \rho_{i,0}$ and arranging the terms, the expected state at the next time period is $\mathbb{E}\left\{\boldsymbol{X}(\boldsymbol{q})\right\} - \sum_{i \in \mathcal{N}} \left\{\sum_{A \in \mathcal{F}} \phi_i(A)\, z^t(A)\right\} \times (1 - \rho_{i,0})\,(\boldsymbol{e}_{i,0} - \boldsymbol{e}_{i,1})$. By (1), $\mathbb{E}\{X_{i,0}(\boldsymbol{q})\} = q_{i,0} + \sum_{s=1}^{\infty} \rho_{i,s}\, q_{i,s}$, $\mathbb{E}\{X_{i,1}(\boldsymbol{q})\} = 0$ and $\mathbb{E}\{X_{i,\ell}(\boldsymbol{q})\} = q_{i,\ell-1} - \rho_{i,\ell-1}\, q_{i,\ell-1}$ for $\ell \geq 2$, which implies that the expected next state in the last expression is linear in the decision variables $\boldsymbol{q}^t$ and $\boldsymbol{z}^t = (z^t(A) : A \in \mathcal{F})$. To get an upper bound on the optimal total expected revenue in our dynamic assortment problem, we use the linear program

$$\max \quad \sum_{t \in \mathcal{T}} \sum_{i \in \mathcal{N}} r_i \sum_{A \in \mathcal{F}} \phi_i(A)\, z^t(A) + \sum_{t \in \mathcal{T}} \sum_{i \in \mathcal{N}} \pi_i \sum_{\ell=1}^{\infty} q^t_{i,\ell} \tag{10}$$

$$\begin{aligned} \text{s.t.} \quad & \boldsymbol{q}^{t+1} \;=\; \mathbb{E}\Big\{\boldsymbol{X}(\boldsymbol{q})\Big\} - \sum_{i\in\mathcal{N}} \left\{ \sum_{A\in\mathcal{F}} \phi_i(A)\, z^t(A) \right\} (1-\rho_{i,0})\,(\boldsymbol{e}_{i,0} - \boldsymbol{e}_{i,1}) \quad \forall\, t \in \mathcal{T}\setminus\{T\} \\ & \boldsymbol{q}^1 = \sum_{i\in\mathcal{N}} C_i\, \boldsymbol{e}_{i,0} \\ & \sum_{A\in\mathcal{F}} z^t(A) = 1 \quad \forall\, t\in\mathcal{T} \\ & z^t(A) \geq 0 \quad \forall A\in\mathcal{F},\ t\in\mathcal{T},\ q^t_{i,\ell}\geq 0 \quad \forall\, i\in\mathcal{N},\ \ell\geq 0,\ t\in\mathcal{T}. \end{aligned}$$

From the discussion right before the above problem, the objective function and the constraints are linear in $(z^t(A) : A\in\mathcal{F},\ t\in\mathcal{T})$ and $(q^t_{i,\ell} : i\in\mathcal{N},\ \ell\geq 0,\ t\in\mathcal{T})$. Therefore, the problem above is indeed a linear program. Since $\sum_{A\in\mathcal{F}} \phi_i(A)\, z^t(A)$ is the expected number of customers that choose product i at time period t, and $\sum_{\ell=1}^{\infty} q^t_{i,\ell}$ is the expected number of units of product i that are in use at time period t, the objective function above computes the total expected revenue over the selling horizon. The first constraint keeps track of the numbers of products with different durations of use under the assumption that the customer choices and transition dynamics take on their expected values. The second constraint initializes the state of the system. The third constraint ensures that we offer an assortment at each time period, but this assortment can be empty. By the same argument in Section 2, because the products are all available on-hand at the beginning of the selling horizon, we have $q^t_{i,\ell}=0$ for all $\ell\geq T+1$ in a feasible solution to the linear program above. Therefore, we do not need to define the decision variable $q^t_{i,\ell}$ for $\ell \geq T+1$, indicating that the numbers of decision variables and constraints are finite. Thus, we can solve the linear program above using standard linear programming software. In the next proposition, we show that the optimal objective value of the linear program above is an upper bound on the optimal total expected revenue in our dynamic assortment problem. The proof follows from an argument that often appears in the revenue management literature. We defer the proof to Appendix F.

Proposition 6.1 *Letting Z^* be the optimal objective value of problem* (10)*, we have $Z^* \geq J^1(\sum_{i\in\mathcal{N}} C_i\, \boldsymbol{e}_{i,0})$.*

In problem (10), we have one decision variable $z^t(A)$ for each assortment $A\in\mathcal{F}$. Therefore, the number of decision variables increases exponentially with the number of products. We can solve problem (10) by using column generation. The column generation subproblem has the same structure as the maximization problem in (3). As discussed at the end of Section 2, this problem is tractable under a variety of choice models. Lastly, we formulate problem (10) under the assumption that there is a single customer type and we make assortment decisions. However, it is not difficult to give analogues of problem (10) when the customers are heterogeneous and we make pricing decisions, reflecting the extensions in the previous section.

6.2 Revenue Management over Parallel Flight Legs

In our first set of computational experiments, the products are not reusable. We focus on the problem of making assortment decisions for parallel flights operating between an origin-destination pair. We treat each flight as one product. Assuming no cancellations occur, a seat purchased by a customer on a flight is never returned. This situation corresponds to having infinite usage durations. The dynamics of the problem are as follows. When a customer arrives into the system, we offer an assortment of flights. The customer either makes a purchase within the offered assortment or leaves the system. If the customer makes a purchase, then we generate a revenue and consume a unit of capacity on the flight chosen by the customer. Our goal is to find a policy to decide what assortment of flights to offer to each customer so that the total expected revenue is maximized.

Experimental Setup: In our test problems, there are two customer types indexed by $\mathcal{M} = \{1, 2\}$. In Section 5.1, we discuss how to extend our model to the case with multiple customer types. We continue indexing the products by $\mathcal{N} = \{1, \ldots, n\}$, corresponding to flights. Customers of type 1 are less price conscious than those of type 2. In particular, the upfront fee r_i^1 paid by customers of type 1 for each product i is generated from the uniform distribution over $[50, 100]$, whereas the upfront fee r_i^2 paid by customers of type 2 for each product i is generated from the uniform distribution over $[0, 50]$. Because the products are purchased rather than rented, we set the per period rental fee $\pi_i^j = 0$ for all $i \in \mathcal{N}$, $j \in \mathcal{M}$. The probability of having an arrival of each customer type at each time period is given by $p^{t,1} = 0.3$ and $p^{t,2} = 0.7$ for all $t \in \mathcal{T}$.

Customers choose according to the multinomial logit model. A customer of type j associates the preference weight v_i^j with product i and the preference weight v_0^j with the no purchase option. If we offer the assortment S, then a customer of type j chooses product $i \in S$ with probability $\phi_i^j(S) = v_i^j / (v_0^j + \sum_{\ell \in S} v_\ell^j)$. We generate v_i^j from the uniform distribution over $[0, 10]$ for all $i \in \mathcal{N}$, $j \in \mathcal{M}$. We calibrate the preference weight of the no purchase option so that if we offer all of the flights, then a customer leaves without a purchase with probability P_0, where P_0 is a parameter that we vary. Therefore, we calibrate v_0^j to satisfy $P_0 = v_0^j / (v_0^j + \sum_{\ell \in \mathcal{N}} v_\ell^j)$. To generate the capacities on the flights, we let S^j be the revenue maximizing assortment for customers of type j; that is, $S^j = \arg\max_{S \subseteq N} \sum_{i \in \mathcal{N}} r_i^j \, \phi_i^j(S)$. If we offer the assortment S^j to customers of type j, then the total expected demand for the capacity for product i is $\sum_{t \in \mathcal{T}} \sum_{j \in \mathcal{M}} p^{t,j} \, \phi_i^j(S^j)$. We set the capacity C_i of product i such that the total demand for product i exceeds the capacity by a factor of ρ, where ρ is another parameter that we vary, so $\sum_{t \in \mathcal{T}} \sum_{j \in \mathcal{M}} p^{t,j} \, \phi_i^j(S^j) = \rho \, C_i$ for all $i \in \mathcal{N}$. We refer to the parameter ρ as the load factor. We fix the length of the selling horizon at $T = 35 \times n \times \rho$.

Varying the number of products n over $\{6, 8\}$, the no purchase probability P_0 over $\{0.1, 0.4\}$ and the load factor ρ over $\{1.0, 1.2, 1.6\}$, we obtain 12 test problems.

Benchmarks: We compare the performance of the following four benchmark strategies.

Greedy Policy (GR). In this benchmark, we use the greedy policy with respect to the value function approximations $\{\hat{J}^t : t \in \mathcal{T}\}$, as discussed in Section 3.

Rollout Policy (RO). This benchmark is the policy obtained by applying rollout on the static policy, as discussed in Section 4.

Bid-Prices (BP). This benchmark is the classical bid-price policy. We use a variant of the linear program in (10) to estimate the value of a unit of capacity on each flight, called its bid-price. We offer the revenue maximizing set of flights at each time period, after adjusting the revenues from the flights by their bid-prices; see Section 5.2 in Zhang and Adelman (2009).

Decomposition (DC). This benchmark corresponds to the classical dynamic programming decomposition method. The idea is to decompose the dynamic programming formulation of the problem by the products and to obtain approximations to the value functions by solving a separate dynamic program for each product; see Section 6.2 in Liu and van Ryzin (2008).

To further improve the performance of the benchmarks, we divide the selling horizon into three equal segments and recompute the policy parameters at the beginning of each segment. For GR, if the remaining capacities of the products at the beginning of a segment are $(C_i' : i \in \mathcal{N})$ and the set of remaining time periods in the selling horizon is $\mathcal{T}' \subseteq \mathcal{T}$, then we apply the recursive computation at the beginning of Section 3.1 after replacing C_i with C_i' and $\mathcal{T}$ with $\mathcal{T}'$, which yields new value function approximations. We use the new value function approximations until we reach the next segment. We recompute the policy parameters for RO, BP and DC similarly.

Computational Results: We give our computational results in Table 1. The first column in this table shows the parameters for each test problem by using the tuple (n, ρ, P_0), where n, ρ and P_0 are as described above. The second column shows the upper bound on the optimal total expected revenue provided by the optimal objective value of the linear program in (10). The third through sixth columns show the total expected revenue obtained by each of the benchmarks RO, DC, GR and BP. We estimate these total expected revenues by simulating the performance of each benchmark over 1,000 sample paths. The seventh through ninth columns show the percent gaps between the total expected revenues obtained by RO and every other benchmark. The standard errors of these percent gaps are given in parentheses. For example, if a certain percent gap exceeds the corresponding standard error by more than a factor of 1.65, which is the 95-th standard normal percentile, then the percent gap is statistically significant at the 95% level.

Our computational results indicate that RO provides noticeable improvements over GR and BP. The improvements tend to be larger when ρ is large so that the capacities are scarce or when

Table 1 Computational results for revenue management over parallel flight legs.

Prob. Params.	Upp.	Total Expected Revenue				% Gain of RO over Benchmarks					
(n, ρ, P_0)	Bnd.	RO	DC	GR	BP	DC		GR		BP	
(6, 1.0, 0.1)	6,809	6,392	6,655	6,525	6,198	-3.95	(0.19)	-2.04	(0.06)	3.12	(0.22)
(6, 1.0, 0.4)	4,500	4,213	4,282	4,168	3,947	-1.61	(0.32)	1.09	(0.08)	6.73	(0.37)
(6, 1.2, 0.1)	7,217	7,025	6,720	7,027	6,549	4.53	(0.20)	-0.03	(0.05)	7.27	(0.21)
(6, 1.2, 0.4)	4,851	4,651	4,649	4,633	4,345	0.04	(0.27)	0.40	(0.08)	7.05	(0.30)
(6, 1.6, 0.1)	7,910	7,699	7,426	7,628	6,697	3.67	(0.21)	0.93	(0.05)	14.96	(0.25)
(6, 1.6, 0.4)	5,413	5,230	5,291	5,190	4,798	-1.17	(0.24)	0.75	(0.08)	8.99	(0.30)
(8, 1.0, 0.1)	8,308	7,794	7,993	7,894	7,761	-2.50	(0.18)	-1.27	(0.06)	0.42	(0.18)
(8, 1.0, 0.4)	6,538	6,153	6,277	6,078	5,990	-1.96	(0.29)	1.24	(0.06)	2.73	(0.32)
(8, 1.2, 0.1)	8,700	8,475	8,125	8,371	7,813	4.30	(0.17)	1.24	(0.05)	8.47	(0.18)
(8, 1.2, 0.4)	7,103	6,813	6,817	6,765	6,392	-0.05	(0.27)	0.71	(0.06)	6.60	(0.30)
(8, 1.6, 0.1)	9,382	9,154	8,561	9,026	7,912	6.93	(0.18)	1.41	(0.06)	15.70	(0.22)
(8, 1.6, 0.4)	8,025	7,736	7,821	7,677	7,121	-1.09	(0.23)	0.77	(0.07)	8.64	(0.27)
					Average	0.60		0.43		7.56	

P_0 is large so that the customers are more likely to leave without making a purchase. For the test problems with $\rho = 1.0$ and $P_0 = 0.1$, the performance gap between RO and GR is -3.22%, whereas the same performance gap is 5.30% for the test problems with $\rho = 1.6$ and $P_0 = 0.4$. The benchmarks RO and DC are competitive with each other, but to our knowledge, DC does not have any theoretical performance guarantees.

6.3 Street Parking Pricing in the City of Seattle

In our second set of computational experiments, the products are reusable. We focus on the problem of dynamically pricing street parking spaces. We treat the parking spaces within close proximity to each other as one product. After having been used by a driver for a certain duration of time, a parking space can be used by another driver, so the parking spaces are reusable products. The dynamics of the problem are as follows. When a driver arrives into the system with an intention to park in a certain region, as a function of the remaining parking space inventory in the nearby regions, we decide on the prices to charge for the parking spaces in different regions. The driver is informed about the prices in real time, possibly through a smartphone application. The driver either parks at a particular parking space or decides to leave the system. If the driver parks, then the parking space generates revenue for a random usage duration, after which the space becomes available. Our goal is to find a policy for deciding on the parking spaces to offer and the prices to charge for these offered parking spaces so that the total expected revenue is maximized.

Transaction Data: For brevity of discussion, we describe the most essential elements of the data that we use in our computational experiments, the approach that we use to augment and modify the data for compliance with our modeling assumptions, and the methodology that we use to estimate the model parameters. We defer the details to Appendix G. We build on the data provided by the Open Data Program in the city of Seattle; see Seattle Open Data (2017). Seattle uses parking rates that are dependent on the time of the day and location. Through the Open Data Program, we have access to transaction data on the use of the street parking spaces during 20 weekdays of June 2017. Each transaction record shows a parking event, documenting the start time, duration, and location of the parking event, along with the rate paid. We focus on 40 blocks in the downtown area between the hours of 11AM and 4PM. We partition this area into 11 block clusters, each including approximately four blocks laid out in a two-by-two configuration. We refer to each two-by-two block cluster as a locale.

The street parking spaces in each locale correspond to a different product in our model. Thus, we have $n = 11$ products. To comply with our modeling assumptions, we augment and modify the data provided by the Open Data Program as follows. We assume each driver arrives into the system with the intention to park in a particular locale. The intended locale of a driver determines the type of the driver. In Section 5.1, we discuss the extension of our model to the case in which we have multiple customer types. Because the intended locale of a driver determines her type, there are $m = 11$ customer types. In the data, we have access to the locale at which a driver actually parked, but we do not have access to the intended locale of a driver. For each driver, we randomly sample one of the five locales that are closest to the locale that the driver actually parked. We set the intended locale of the driver as this sampled locale. Once we augment the data in this way, each transaction record gives the start time, duration, intended locale, actual parked locale, and per hour rate for each parking event. Because the data that we use are obtained by augmenting the data from the Open Data Program, we caution the reader against comparing our results with the real parking operations in the city of Seattle.

The set of feasible locales we can offer to a driver are the five locales that are closest to her intended one. As a function of the remaining parking space inventories in these locales, we decide on the prices to charge for each locale. In Section 5.2, we discuss the extension of our model to the case in which we make pricing decisions. The driver either decides to park in one of these locales or leaves the system. If the driver parks, then we generate a certain revenue depending on the parking duration and the charged price. Although we discussed the extensions of our model to multiple customer types and to pricing decisions separately, it is not difficult to combine these extensions and to come up with a variant of our model that makes pricing decisions under multiple customer types. It is also not difficult to extend the linear program in (10) to the case in which we make pricing decisions under multiple customer types.

Experimental Setup: As discussed in Section 5.2, when making pricing decisions, we create multiple copies of each product, whereby the different copies correspond to charging different prices for the product. Using $\mathcal{H}$ to denote the set of possible prices that we can charge for a parking space and recalling that $\mathcal{N}$ is the set of possible parking locales, offering product copy $(i,h) \in \mathcal{N} \times \mathcal{H}$ represents charging price level h for locale i. We use $\pi_{i,h}$ to denote the per period fee when we charge the price level h for locale i. We assume that the choices of the drivers are governed by the multinomial logit model. In particular, if we offer the assortment $S \subseteq \mathcal{N} \times \mathcal{H}$ of locale and price combinations to a driver with intended locale j, then she chooses to park in locale i with probability $\phi_i^j(S) = \frac{e^{\alpha^j + \beta\,\pi_{i,h}}}{1+\sum_{(\ell,g)\in S} e^{\alpha^j+\beta\,\pi_{\ell,g}}}$ as long as $(i,h) \in S$. The parameter β captures the price sensitivity of the drivers and it is assumed to be constant over all drivers.

Throughout the paper so far, we assumed that there is one customer arrival at each time period. This assumption is not appropriate here, because the arrival rate of the drivers vary during the day, but extending our model to the case in which there is at most one arrival at each time period is straightforward. We scale the time so that each time period in our model corresponds to a time interval of 30 seconds. A time interval of 30 seconds is short enough to ensure that there is at most one driver arrival in the region of our focus. We use $p^{t,j}$ to denote the probability that a driver with intended locale j arrives at time period t. We estimate the parameters β, $(\alpha^j : j \in \mathcal{M})$ and $(p^{t,j} : t \in \mathcal{T},\ j \in \mathcal{M})$ by using maximum likelihood.

We model the parking duration in locale i as $1 + \mathsf{NegBin}(s_i, \eta_i)$, where $\mathsf{NegBin}(s_i, \eta_i)$ is a negative binomial random variable with parameters $s_i \in \mathbb{Z}_{++}$ and $\eta_i \in (0,1)$. As discussed in Section 4.3, if s_i is small, then we can perform rollout on the static policy in a tractable fashion. For each locale i, a negative binomial distribution with the parameter $s_i = 2$ provided a sensible fit.

Ultimately, in our experimental setup, we vary the length of the selling horizon over three values, 11AM-2PM, 11AM-3PM and 11AM-4PM. To obtain problems with different load factors, we scale the arrival rates with three different factors, 2.5, 3.0 and 3.5. Also, we vary the number of parking spaces over two values, 55 and 79. This experimental setup yields 18 parameter combinations for our test problems. From the rates used by the city of Seattle, the possible rates we can charge are within the menu of \$2, \$4 and \$6 per hour.

Benchmarks: We continue using the benchmarks greedy policy (GR) and rollout policy (RO), as discussed for parallel flights in Section 6.2. We make the necessary modifications in these benchmarks to ensure that we can handle multiple customer types and we choose the prices of the offered products. The benchmark decomposition (DC) does not extend when the products are reusable, so we do not use that benchmark. Instead of using the benchmark bid-prices (BP), we use the linear program in (10) in a different way to come up with the following benchmark.

Linear Program (LP). Let $(\hat{z}^t(A) : A \in \mathcal{F},\ t \in \mathcal{T})$ and $(\hat{q}^t_{i,\ell} : i \in \mathcal{N},\ \ell \geq 0,\ t \in \mathcal{T})$ be an optimal solution to problem (10). Using N^t to denote the set of products for which we have on-hand

Table 2 Computational results for street parking pricing in the city of Seattle.

Prob. Params. $(\mathcal{T}, \sigma, C)$	Load factor	Upp. Bnd.	Total Expected Revenue RO	GR	LP	FP	% Gain of RO over Benchmarks GR	LP	FP
(11AM-2PM, 2.5, 79)	0.79	344	329	320	325	321	2.82 (0.11)	1.01 (0.11)	2.25 (0.10)
(11AM-2PM, 3.0, 79)	0.95	410	385	375	379	375	2.62 (0.08)	1.73 (0.10)	2.69 (0.08)
(11AM-2PM, 3.5, 79)	1.11	474	439	429	430	423	2.20 (0.08)	2.08 (0.09)	3.59 (0.07)
(11AM-2PM, 2.5, 55)	1.14	338	306	301	300	296	1.82 (0.12)	2.16 (0.12)	3.55 (0.11)
(11AM-2PM, 3.0, 55)	1.37	397	353	348	345	336	1.69 (0.09)	2.57 (0.11)	5.10 (0.09)
(11AM-2PM, 3.5, 55)	1.60	452	396	390	385	370	1.61 (0.08)	2.96 (0.10)	6.87 (0.09)
(11AM-3PM, 2.5, 79)	0.76	495	466	454	460	455	2.77 (0.07)	1.36 (0.08)	2.56 (0.06)
(11AM-3PM, 3.0, 79)	0.91	588	545	532	533	528	2.42 (0.05)	2.10 (0.07)	3.15 (0.05)
(11AM-3PM, 3.5, 79)	1.06	676	618	605	603	593	2.20 (0.06)	2.49 (0.07)	4.29 (0.05)
(11AM-3PM, 2.5, 55)	1.09	482	430	423	420	413	1.72 (0.08)	2.55 (0.09)	4.21 (0.07)
(11AM-3PM, 3.0, 55)	1.31	564	495	487	482	467	1.63 (0.06)	2.68 (0.08)	6.02 (0.06)
(11AM-3PM, 3.5, 55)	1.53	640	553	544	538	513	1.69 (0.06)	2.84 (0.07)	7.89 (0.06)
(11AM-4PM, 2.5, 79)	0.72	631	591	575	582	575	2.82 (0.06)	1.49 (0.07)	2.77 (0.05)
(11AM-4PM, 3.0, 79)	0.86	749	689	673	674	667	2.46 (0.05)	2.27 (0.06)	3.42 (0.04)
(11AM-4PM, 3.5, 79)	1.01	860	781	766	761	747	1.93 (0.05)	2.60 (0.06)	4.52 (0.05)
(11AM-4PM, 2.5, 55)	1.04	613	543	534	528	520	1.61 (0.06)	2.72 (0.08)	4.48 (0.06)
(11AM-4PM, 3.0, 55)	1.25	717	624	614	609	587	1.53 (0.05)	2.51 (0.07)	6.23 (0.05)
(11AM-4PM, 3.5, 55)	1.46	812	696	686	679	644	1.55 (0.05)	2.49 (0.06)	8.16 (0.06)
						Average	2.06	2.26	4.54

inventory at time period t, we sample an assortment S with respect to the probabilities $(\hat{z}^t(A) : A \in \mathcal{F})$ and offer the assortment $S \cap N^t$ at time period t. Because of the third set of constraints in problem (10), the probabilities $(\hat{z}^t(A) : A \subseteq N)$ indeed add up to one. Offering the assortment $S \cap N^t$ ensures that we offer only products that are currently available. We also added the following benchmark to our experimental setup.

Fixed Price (FP). In this benchmark, we charge one fixed price for all locales at all time periods. We test the performance of the rates \$2, \$4 and \$6 per hour, which is the price menu used in other benchmarks. We select the best constant price. This benchmark is clearly not sophisticated but it serves as a simple baseline. In all of our test problems, the rate \$4 per hour provided the best performance.

Computational Results: We present our computational results in Table 2. The first column in this table shows the parameters for each test problem by using the tuple $(\mathcal{T}, \sigma, C)$, where $\mathcal{T} \in \{11\text{AM-2PM}, 11\text{AM-3PM}, 11\text{AM-4PM}\}$ is the selling horizon, $\sigma \in \{2.5, 3.0, 3.5\}$ is the multiplier for the arrival rates, and $C \in \{55, 79\}$ is the total number of parking spaces. The second column shows the load factor for the test problems. Noting that $\mathbb{E}\{\mathsf{Duration}_i\}$ is the expected parking duration in locale i, we estimate the number of times that we can turn over a parking space in locale i as $T/\mathbb{E}\{\mathsf{Duration}_i\}$. The total expected demand for parking is $\sum_{t \in \mathcal{T}} \sum_{j \in \mathcal{M}} p^{t,j}$. With C_i parking spaces available in locale i, the load factor is given by $\frac{\sum_{t \in \mathcal{T}} \sum_{j \in \mathcal{M}} p^{t,j}}{\sum_{i \in \mathcal{N}} C_i \, T/\mathbb{E}\{\mathsf{Duration}_i\}}$. The organization of the rest of the table closely mirrors that of Table 1, but we use 10,000 sample paths, rather than 1,000, to estimate the total expected revenues in the fourth through seventh columns. As a result, the standard errors in Table 2 are smaller than those in Table 1.

Across all of our test problems, the strongest benchmark is consistently RO. This benchmark provides an average improvement of 2.06%, 2.26% and 4.54% over GR, LP and FP in terms of total expected revenues. The performance of GR is comparable to that of LP. For problem instances with smaller load factors, LP tends to perform better, whereas for problem instances with larger load factors, GR tends to perform better. Nevertheless, it is important to note that LP requires solving a relatively large linear program, whereas the computation of the value function approximations used by GR requires a simple recursion over the selling horizon. The performance of RO is superior to that of LP. Furthermore, the performance gap between RO and LP also tends to be larger for the problem instances with larger load factors. The solution times for the linear program in (10) range from 894 to 6,535 seconds depending on the size of the problem instance. The computation times to compute the value functions $\{\hat{J}^t : t \in \mathcal{T}\}$ for GR range from 362 to 672 seconds, whereas the computation times to compute the value functions $\{V_i^t : i \in \mathcal{N},\ t \in \mathcal{T}\}$ for RO range from 1,550 to 17,201 seconds. A few preliminary runs indicated that the performance of none of the benchmarks improve noticeably with recomputations of the policy parameters. Considering the fact that the run times are relatively long, we do not recompute the policy parameters for any of the benchmarks. The computation times for RO are significantly longer, but this benchmark, by explicitly taking the inventory levels of the products into consideration, provides significant improvements over others in terms of total expected revenues.

7. Conclusions

We studied a dynamic assortment problem with reusable products, and provided policies with half-approximate performance guarantees. Our rollout of the static policy decomposes the problem by the products, which is reminiscent of dynamic programming decomposition techniques in revenue management. To our knowledge, existing decomposition methods do not provide any performance guarantees. An exciting future research area is to construct decomposition methods with performance guarantees for other revenue management problems. Moreover, reusable products frequently appear in sharing economies, but such products often have modifiable attributes. For example, a driver in a ride-sharing setting is a reusable product, but the location of the driver can be modified by the decision maker. The extension of our work to this setting is highly non-trivial, but it would significantly enhance the applicability of our approach.

References

Adelman, D. 2007. Dynamic bid prices in revenue management. *Operations Research* **55**(4) 647–661.

Bertsekas, D. P., J. N. Tsitsiklis. 1996. *Neuro-Dynamic Programming*. Athena Scientific, Belmont, MA.

Dai, J., W. Ding, A. J. Kleywegt, X. Wang, Y. Zhang. 2014. Choice based revenue management for parallel flights. Tech. rep., Georgia Tech, Atlanta, GA.

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. Online bipartite matching. Working Paper, Columbia University.

Golrezaei, N., H. Nazerzadeh, P. Rusmevichientong. 2014. Real-time optimization of personalized assortments. *Management Science* **60**(6) 1532–1551.

Kunnumkal, S., H. Topaloglu. 2010. A new dynamic programming decomposition method for the network revenue management problem with customer choice behavior. *Production and Operations Management* **19**(5) 575–590.

Levi, R., A. Radovanovic. 2010. Provably near-optimal LP-based policies for revenue management in systems with reusable resources. *Operations Research* **58**(2) 503–507.

Liu, Q., G. J. van Ryzin. 2008. On the choice-based linear programming model for network revenue management. *Manufacturing & Service Operations Management* **10**(2) 288–310.

Owen, Z., D. Simchi-Levi. 2017. Price and assortment optimization in systems of reusable resources. *Presentation at INFORMS Revenue Management & Pricing Section Conference*.

Ozer, Ozalp, Robert Phillips, eds. 2012. *The Oxford Handbook of Pricing Management*. Oxford University Press, UK.

Seattle Open Data. 2017. Seattle parking transactions. URL `https://data.seattle.gov/Transportation/Seattle-Parking-Transactions/updn-y53g`.

Stein, C., V. A. Truong, X. Wang. 2016. Advance reservations with heterogeneous customers. Working Paper, Columbia University.

Tong, C., H. Topaloglu. 2013. On the approximate linear programming approach for network revenue management problems. *INFORMS Journal on Computing* **26**(1) 121–134.

Topaloglu, H. 2013. Joint stocking and product offer decisions under the multinomial logit model. *Production and Operations Management* **22**(5) 1182–1199.

Vossen, T. W. M., D. Zhang. 2015. Reductions of approximate linear programs for network revenue management. *Operations Research* **63**(6) 1352–1371.

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, X., V. A. Truong, D. Bank. 2016. Online advance admission scheduling for services, with customer preferences. Working Paper, Columbia University.

Zhang, D., D. Adelman. 2009. An approximate dynamic programming approach to network revenue management with customer choice. *Transportation Science* **43**(3) 381–394.

Zhang, D., W. L. Cooper. 2005. Revenue management for parallel flights with customer choice behavior. *Operations Research* **53**(3) 415–431.

This page is intentionally blank. Proper e-companion title page, with INFORMS branding and exact metadata of the main paper, will be produced by the INFORMS office when the issue is being assembled.

Online Appendix

Dynamic Assortment Optimization for Reusable Products with Random Usage Duration

Appendix A: Expected Contribution of Ideal Assortment

For notational brevity, we let $\Delta_i^t = r_i + \pi_i - (1-\rho_{i,0})\left(\hat{\nu}_{i,0}^{t+1} - \hat{\nu}_{i,1}^{t+1}\right)$. In the next lemma, we show that $\sum_{i\in\mathcal{N}} \phi_i(\hat{A}^t)\,\Delta_i^t \ge \sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\ge 1\}}\,\phi_i(S)\,\Delta_i^t$ for all $S\in\mathcal{F}$.

Lemma A.1 *For all $S\in\mathcal{F}$, we have $\sum_{i\in\mathcal{N}} \phi_i(\hat{A}^t)\,\Delta_i^t \ge \sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\ge 1\}}\,\phi_i(S)\,\Delta_i^t$.*

Proof: Note that we have $\mathbb{1}_{\{\Delta_i^t\ge 0\}}\,\Delta_i^t \ge \mathbb{1}_{\{\Delta_i^t\ge 0\}}\,\mathbb{1}_{\{q_{i,0}\ge 1\}}\,\Delta_i^t \ge \mathbb{1}_{\{q_{i,0}\ge 1\}}\,\Delta_i^t$, where the first inequality is by the fact that $\mathbb{1}_{\{\Delta_i^t\ge 0\}}\,\Delta_i^t\ge 0$ and the second inequality is by the fact that $\mathbb{1}_{\{\Delta_i^t\ge 0\}}\,\mathbb{1}_{\{q_{i,0}\ge 1\}}\,\Delta_i^t\ge 0$, but $\mathbb{1}_{\{q_{i,0}\ge 1\}}\,\Delta_i^t$ can be positive or negative. We will shortly establish the claim that $\sum_{i\in\mathcal{N}} \phi_i(\hat{A}^t)\,\Delta_i^t \ge \sum_{i\in\mathcal{N}} \phi_i(S)\,\mathbb{1}_{\{\Delta_i^t\ge 0\}}\,\Delta_i^t$ for all $S\in\mathcal{F}$. In this case, noting the chain of inequalities at the beginning of the proof, we obtain $\sum_{i\in\mathcal{N}} \phi_i(\hat{A}^t)\,\Delta_i^t \ge \sum_{i\in\mathcal{N}} \phi_i(S)\,\mathbb{1}_{\{\Delta_i^t\ge 0\}}\,\Delta_i^t \ge \sum_{i\in\mathcal{N}} \phi_i(S)\,\mathbb{1}_{\{\Delta_i^t\ge 0\}}\,\mathbb{1}_{\{q_{i,0}\ge 1\}}\,\Delta_i^t \ge \sum_{i\in\mathcal{N}} \phi_i(S)\,\mathbb{1}_{\{q_{i,0}\ge 1\}}\,\Delta_i^t$ for all $S\in\mathcal{F}$, which is the desired result. We proceed to establishing the claim that $\sum_{i\in\mathcal{N}} \phi_i(\hat{A}^t)\,\Delta_i^t \ge \sum_{i\in\mathcal{N}} \phi_i(S)\,\mathbb{1}_{\{\Delta_i^t\ge 0\}}\,\Delta_i^t$ for all $S\in\mathcal{F}$. Assume on the contrary that there exists $\hat{S}\in\mathcal{F}$ such that $\sum_{i\in\mathcal{N}} \phi_i(\hat{A}^t)\,\Delta_i^t < \sum_{i\in\mathcal{N}} \phi_i(\hat{S})\,\mathbb{1}_{\{\Delta_i^t\ge 0\}}\,\Delta_i^t$. We define the assortment S^* as $S^* = \{i\in\hat{S} : \Delta_i^t\ge 0\}$. Therefore, we have $S^*\subseteq\hat{S}\subseteq\mathcal{N}$, in which case, since $\hat{S}\in\mathcal{F}$, we have $S^*\in\mathcal{F}$ by Assumption 2.1. For all $i\in S^*$, we have $\Delta_i^t\ge 0$ by the definition of S^*, so $\sum_{i\in S^*} \phi_i(S^*)\,\Delta_i^t = \sum_{i\in S^*} \phi_i(S^*)\,\mathbb{1}_{\{\Delta_i^t\ge 0\}}\,\Delta_i^t$. Also, for all $i\in\hat{S}\setminus S^*$, we have $\phi_i(S^*)=0$ and $\Delta_i^t<0$, in which case, we have $\sum_{i\in\hat{S}\setminus S^*} \phi_i(S^*)\,\Delta_i^t = 0 = \sum_{i\in\hat{S}\setminus S^*} \phi_i(\hat{S})\,\mathbb{1}_{\{\Delta_i^t\ge 0\}}\,\Delta_i^t$. Lastly, for all $i\in\mathcal{N}\setminus\hat{S}$, we have $\phi_i(S^*) = 0 = \phi_i(\hat{S})$. Therefore, we have $\sum_{i\in\mathcal{N}\setminus\hat{S}} \phi_i(S^*)\,\Delta_i^t = 0 = \sum_{i\in\mathcal{N}\setminus\hat{S}} \phi_i(\hat{S})\,\mathbb{1}_{\{\Delta_i^t\ge 0\}}\,\Delta_i^t$. Noting Assumption 2.1, since $S^*\subseteq\hat{S}$, we have $\phi_i(S^*)\ge\phi_i(\hat{S})$ for all $i\in S^*$. In this case, we obtain

$$
\begin{aligned}
\sum_{i\in\mathcal{N}} \phi_i(S^*)\,\Delta_i^t &= \sum_{i\in S^*} \phi_i(S^*)\,\Delta_i^t + \sum_{i\in\hat{S}\setminus S^*} \phi_i(S^*)\,\Delta_i^t + \sum_{i\in\mathcal{N}\setminus\hat{S}} \phi_i(S^*)\,\Delta_i^t \\
&= \sum_{i\in S^*} \phi_i(S^*)\,\mathbb{1}_{\{\Delta_i^t\ge 0\}}\,\Delta_i^t + \sum_{i\in\hat{S}\setminus S^*} \phi_i(\hat{S})\,\mathbb{1}_{\{\Delta_i^t\ge 0\}}\,\Delta_i^t + \sum_{i\in\mathcal{N}\setminus\hat{S}} \phi_i(\hat{S})\,\mathbb{1}_{\{\Delta_i^t\ge 0\}}\,\Delta_i^t \\
&\ge \sum_{i\in S^*} \phi_i(\hat{S})\,\mathbb{1}_{\{\Delta_i^t\ge 0\}}\,\Delta_i^t + \sum_{i\in\hat{S}\setminus S^*} \phi_i(\hat{S})\,\mathbb{1}_{\{\Delta_i^t\ge 0\}}\,\Delta_i^t + \sum_{i\in\mathcal{N}\setminus\hat{S}} \phi_i(\hat{S})\,\mathbb{1}_{\{\Delta_i^t\ge 0\}}\,\Delta_i^t \\
&= \sum_{i\in\mathcal{N}} \phi_i(\hat{S})\,\mathbb{1}_{\{\Delta_i^t\ge 0\}}\,\Delta_i^t.
\end{aligned}
$$

Thus, noting the assumption $\sum_{i\in\mathcal{N}} \phi_i(\hat{S})\,\mathbb{1}_{\{\Delta_i^t\ge 0\}}\,\Delta_i^t > \sum_{i\in\mathcal{N}} \phi_i(\hat{A}^t)\,\Delta_i^t$, we get $\sum_{i\in\mathcal{N}} \phi_i(S^*)\,\Delta_i^t > \sum_{i\in\mathcal{N}} \phi_i(\hat{A}^t)\,\Delta_i^t$, which contradicts the fact that $\hat{A}^t$ is an optimal solution to problem (4). ■

Appendix B: Performance of the Static Policy

In this section, we give a proof for Lemma 4.1, which shows that the total expected revenue of the static policy is at least 50% of the optimal total expected revenue.

Proof: Under the static policy, we offer the assortment $\hat{A}^t$ at time period t regardless of the product availabilities, where $\hat{A}^t$ is given by an optimal solution to problem (4). If a customer chooses a product that does not have on-hand inventory, then the customer leaves without using the product. Let $V^t(\boldsymbol{q})$ denote the total expected revenue under this static policy over the time periods $t,\ldots,T$, given that we are in state $\boldsymbol{q}$ at time period t. Similar to the dynamic program in (3), we can compute $\{V^t : t \in \mathcal{T}\}$ by using the recursion

$$V^t(\boldsymbol{q}) = \sum_{i\in\mathcal{N}} \pi_i \sum_{\ell=1}^{\infty} q_{i,\ell} \; + \; \mathbb{E}\Big\{V^{t+1}(\boldsymbol{X}(\boldsymbol{q}))\Big\} \\ + \sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\geq 1\}}\, \phi_i(\hat{A}^t)\Big(r_i + \pi_i - (1-\rho_{i,0})\mathbb{E}\Big\{V^{t+1}(\boldsymbol{X}(\boldsymbol{q})) - V^{t+1}\left(\boldsymbol{X}(\boldsymbol{q}) - \boldsymbol{e}_{i,0} + \boldsymbol{e}_{i,1}\right)\Big\}\Big),$$

with the boundary condition that $V^{T+1} = 0$. For all $t \in \mathcal{T}$ and $\boldsymbol{q} \in \mathcal{Q}$, let $H^t(\boldsymbol{q}) = \sum_{i\in\mathcal{N}} \sum_{\ell=0}^{\infty} q_{i,\ell}\, \hat{\nu}^t_{i,\ell}$. We will use induction over the time periods to show that $V^t(\boldsymbol{q}) \geq H^t(\boldsymbol{q})$ for all $\boldsymbol{q} \in \mathcal{Q}$ and $t \in \mathcal{T}$. We have $\hat{\nu}^{T+1}_{i,\ell} = 0$ for all $i \in \mathcal{N}$, $\ell = 0,1,\ldots$ by definition, so that $H^{T+1} = 0$. Also, we have $V^{T+1} = 0$, which implies that the result holds at time period $T+1$. Assuming that $V^{t+1}(\boldsymbol{q}) \geq H^{t+1}(\boldsymbol{q})$ for all $\boldsymbol{q} \in \mathcal{Q}$, we will show that $V^t(\boldsymbol{q}) \geq H^t(\boldsymbol{q})$ for all $\boldsymbol{q} \in \mathcal{Q}$. In the proof of Theorem 3.2, we show the equalities

$$\mathbb{E}\Big\{H^{t+1}(\boldsymbol{X}(\boldsymbol{q}))\Big\} \; = \; \sum_{i\in\mathcal{N}} \left\{ q_{i,0}\, \hat{\nu}^{t+1}_{i,0} + \sum_{\ell=1}^{\infty} q_{i,\ell}\left[\rho_{i,\ell}\, \hat{\nu}^{t+1}_{i,0} + (1-\rho_{i,\ell})\, \hat{\nu}^{t+1}_{i,\ell+1}\right] \right\}.$$

$$\mathbb{E}\Big\{H^{t+1}(\boldsymbol{X}(\boldsymbol{q})) - H^{t+1}\left(\boldsymbol{X}(\boldsymbol{q}) - \boldsymbol{e}_{i,0} + \boldsymbol{e}_{i,1}\right)\Big\} \; = \; \hat{\nu}^{t+1}_{i,0} - \hat{\nu}^{t+1}_{i,1} \; .$$

Also, recall that we have $\phi_i(\hat{A}^t)\,[r_i + \pi_i - (1-\rho_{i,0})\left(\hat{\nu}^{t+1}_{i,0} - \hat{\nu}^{t+1}_{i,1}\right)] \geq 0$ for all $i \in \mathcal{N}$ by the discussion in the proof of Lemma 3.1. In this case, by the inductive hypothesis and the above recursion defining $V^t(\boldsymbol{q})$, we obtain the chain of inequalities

$$\begin{aligned} V^t(\boldsymbol{q}) \;&\geq\; \sum_{i\in\mathcal{N}} \pi_i \sum_{\ell=1}^{\infty} q_{i,\ell} \; + \; \mathbb{E}\Big\{H^{t+1}(\boldsymbol{X}(\boldsymbol{q}))\Big\} \\ &\quad + \sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\geq 1\}}\, \phi_i(\hat{A}^t)\Big(r_i + \pi_i - (1-\rho_{i,0})\,\mathbb{E}\Big\{H^{t+1}\left(\boldsymbol{X}(\boldsymbol{q})\right) - H^{t+1}\left(\boldsymbol{X}(\boldsymbol{q}) - \boldsymbol{e}_{i,0} + \boldsymbol{e}_{i,1}\right)\Big\}\Big) \\ &= \; \sum_{i\in\mathcal{N}} \pi_i \sum_{\ell=1}^{\infty} q_{i,\ell} \; + \; \sum_{i\in\mathcal{N}} \left\{ q_{i,0}\, \hat{\nu}^{t+1}_{i,0} + \sum_{\ell=1}^{\infty} q_{i,\ell}\left[\rho_{i,\ell}\, \hat{\nu}^{t+1}_{i,0} + (1-\rho_{i,\ell})\, \hat{\nu}^{t+1}_{i,\ell+1}\right] \right\} \\ &\qquad + \; \sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\geq 1\}}\, \phi_i(\hat{A}^t)\,\left[r_i + \pi_i - (1-\rho_{i,0})(\hat{\nu}^{t+1}_{i,0} - \hat{\nu}^{t+1}_{i,1})\right] \end{aligned}$$

$$
\begin{aligned}
\geq \quad & \sum_{i\in\mathcal{N}} \pi_i \sum_{\ell=1}^{\infty} q_{i,\ell} \; + \; \sum_{i\in\mathcal{N}} \left\{ q_{i,0}\, \hat{\nu}_{i,0}^{t+1} + \sum_{\ell=1}^{\infty} q_{i,\ell} \left[\rho_{i,\ell}\, \hat{\nu}_{i,0}^{t+1} + (1-\rho_{i,\ell})\, \hat{\nu}_{i,\ell+1}^{t+1} \right] \right\} \\
& \qquad + \; \sum_{i\in\mathcal{N}} \frac{q_{i,0}}{C_i}\, \phi_i(\hat{A}^t) \left[r_i + \pi_i - (1-\rho_{i,0})(\hat{\nu}_{i,0}^{t+1} - \hat{\nu}_{i,1}^{t+1}) \right],
\end{aligned}
$$

where the last inequality uses the fact that $\phi_i(\hat{A}^t)\left[r_i + \pi_i - (1-\rho_{i,0})(\hat{\nu}_{i,0}^{t+1} - \hat{\nu}_{i,1}^{t+1})\right] \geq 0$ and $\mathbb{1}_{\{q_{i,0}\geq 1\}} \geq q_{i,0}/C_i$ for all $\boldsymbol{q} \in \mathcal{Q}$. By the definition of $\hat{\nu}_{i,0}^t$, we have $\hat{\nu}_{i,0}^t - \hat{\nu}_{i,0}^{t+1} = \frac{1}{C_i}\, \phi_i(\hat{A}^t) \times [r_i + \pi_i - (1-\rho_{i,0})(\hat{\nu}_{i,0}^{t+1} - \hat{\nu}_{i,1}^{t+1})]$. In this case, the expression on the right side of the chain of inequalities above can equivalently be written as

$$
\begin{aligned}
& \sum_{i\in\mathcal{N}} \pi_i \sum_{\ell=1}^{\infty} q_{i,\ell} \; + \; \sum_{i\in\mathcal{N}} \left\{ q_{i,0}\, \hat{\nu}_{i,0}^{t+1} + \sum_{\ell=1}^{\infty} q_{i,\ell} \left[\rho_{i,\ell}\, \hat{\nu}_{i,0}^{t+1} + (1-\rho_{i,\ell})\, \hat{\nu}_{i,\ell+1}^{t+1} \right] \right\} + \sum_{i\in\mathcal{N}} q_{i,0}\, (\hat{\nu}_{i,0}^t - \hat{\nu}_{i,0}^{t+1}) \\
& = \; \sum_{i\in\mathcal{N}} \left\{ q_{i,0}\, \hat{\nu}_{i,0}^{t+1} + \sum_{\ell=1}^{\infty} q_{i,\ell} \left[\pi_i + \rho_{i,\ell}\, \hat{\nu}_{i,0}^{t+1} + (1-\rho_{i,\ell})\, \hat{\nu}_{i,\ell+1}^{t+1} \right] \right\} + \sum_{i\in\mathcal{N}} q_{i,0}\, (\hat{\nu}_{i,0}^t - \hat{\nu}_{i,0}^{t+1}) \\
& = \; \sum_{i\in\mathcal{N}} \left\{ q_{i,0}\, \hat{\nu}_{i,0}^{t+1} + \sum_{\ell=1}^{\infty} q_{i,\ell}\, \hat{\nu}_{i,\ell}^t \right\} + \sum_{i\in\mathcal{N}} q_{i,0}\, (\hat{\nu}_{i,0}^t - \hat{\nu}_{i,0}^{t+1}) \; = \; \sum_{i\in\mathcal{N}} \sum_{\ell=0}^{\infty} \hat{\nu}_{i,\ell}^t\, q_{i,\ell} \; = \; H(\boldsymbol{q}),
\end{aligned}
$$

where the second equality uses the fact that $\hat{\nu}_{i,\ell}^t = \pi_i \; + \; \rho_{i,\ell}\, \hat{\nu}_{i,0}^{t+1} + (1-\rho_{i,\ell})\, \hat{\nu}_{i,\ell+1}^{t+1}$ by definition. The two chains of equalities and inequalities above complete our induction argument, so that $V^t(\boldsymbol{q}) \geq H^t(\boldsymbol{q})$ for all $\boldsymbol{q} \in \mathcal{Q}$ and $t \in \mathcal{T}$. By Lemma 3.3, we also have $J^1\left(\sum_{i\in\mathcal{N}} C_i\, \boldsymbol{e}_{i,0}\right) \leq 2 \sum_{i\in\mathcal{N}} C_i\, \hat{\nu}_{i,0}^1$, where $J^1\left(\sum_{i\in\mathcal{N}} C_i\, \boldsymbol{e}_{i,0}\right)$ is the optimal total expected revenue. Thus, we obtain

$$
V^1\left(\sum_{i\in\mathcal{N}} C_i \boldsymbol{e}_{i,0}\right) \; \geq \; H^1\left(\sum_{i\in\mathcal{N}} C_i \boldsymbol{e}_{i,0}\right) \; = \; \sum_{i\in\mathcal{N}} C_i \hat{\nu}_{i,0}^1 \; \geq \; \frac{1}{2} J^1\left(\sum_{i\in\mathcal{N}} C_i \boldsymbol{e}_{i,0}\right).
$$

■

Appendix C: Decomposability of the Value Functions of the Static Policy

In this section, we give a proof of Lemma 4.2.

Proof of Lemma 4.2: We will prove the result by using induction over the time periods. The result holds at time period $T+1$ because $V^{T+1} = 0 = V_i^{T+1}$ by definition. Assuming that the result holds at time period $t+1$, we proceed to showing that the result holds at time period t. Letting $\boldsymbol{e}_\ell$ be the standard unit vector with one in the ℓ-th coordinate, by the inductive hypothesis, we have $\mathbb{E}\Big\{V^{t+1}(\boldsymbol{X}(\boldsymbol{q})) - V^{t+1}\left(\boldsymbol{X}(\boldsymbol{q}) - \boldsymbol{e}_{i,0} + \boldsymbol{e}_{i,1}\right)\Big\} = \mathbb{E}\Big\{V_i^{t+1}(\boldsymbol{X}_i(\boldsymbol{q}_i)) - V_i^{t+1}\left(\boldsymbol{X}_i(\boldsymbol{q}_i) - \boldsymbol{e}_0 + \boldsymbol{e}_1\right)\Big\}$. In this case, by the recursion that we use to compute $\{V^t : t \in \mathcal{T}\}$, we obtain

$$
\begin{aligned}
V^t(\boldsymbol{q}) = & \sum_{i\in\mathcal{N}} \pi_i \sum_{\ell=1}^{\infty} q_{i,\ell} \; + \; \sum_{i\in\mathcal{N}} \mathbb{E}\Big\{V_i^{t+1}(\boldsymbol{X}_i(\boldsymbol{q}_i))\Big\} \\
& + \sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\geq 1\}}\, \phi_i(\hat{A}^t) \bigg(r_i + \pi_i - (1-\rho_{i,0})\, \mathbb{E}\Big\{V_i^{t+1}(\boldsymbol{X}_i(\boldsymbol{q}_i)) - V_i^{t+1}\left(\boldsymbol{X}_i(\boldsymbol{q}_i) - \boldsymbol{e}_0 + \boldsymbol{e}_1\right)\Big\}\bigg).
\end{aligned}
$$

By (8), the expression at the right side above is equal to $\sum_{i\in\mathcal{N}} V_i^t(\boldsymbol{q}_i)$. Therefore, the result holds at time period t as well. ■

Appendix D: Heterogeneous Customer Types

In this section, we discuss the extension of our approach to the case where there are multiple customer types. In Section 5.1, we already discuss the notation that we use under heterogeneous customer types. We do not repeat the discussion of the notation here. We proceed to giving a dynamic programming formulation under heterogeneous customer types. We use the vector $\boldsymbol{q} = ((q_{i,0}, q^j_{i,\ell}) : i \in \mathcal{N},\ j \in \mathcal{M},\ \ell \geq 1)$ as the state variable, where $q_{i,0}$ is the number of units of product i available on-hand and $q^j_{i,\ell}$ is the number of units of product i that have been used for exactly ℓ time periods by a customer of type j. In this case, state space is given by $\mathcal{Q} = \{\big((q_{i,0} \in \mathbb{Z}_+, q^j_{i,\ell} \in \mathbb{Z}_+) \ :\ i \in \mathcal{N},\ j \in \mathcal{M},\ \ell \geq 1\big) \ :\ q_{i,0} + \sum_{j \in \mathcal{M}} \sum_{\ell=1}^{\infty} q^j_{i,\ell} = C_i \ \forall\, i \in \mathcal{N}\}$. We capture the decision at each time period by $(S^1, \ldots, S^m)$, where $S^j \subseteq \mathcal{N}$ is the assortment that we offer customers of type j. The set of feasible assortments that we can offer to customers of type j is given by $\mathcal{F}^j$. As in Assumption 2.1, we assume that if $A \in \mathcal{F}^j$, then $S \in \mathcal{F}^j$ for all $S \subseteq A$. Similarly, the choice model $\{\phi^j_i(S) : S \subseteq \mathcal{N}\}$ that drives the choices of customers of type j satisfies $\phi^j_i(S \cup \{k\}) \leq \phi^j_i(S)$ for all $S \subseteq \mathcal{N}$, $k \in \mathcal{N}$ and $i \in S$. Given that the state is $\boldsymbol{q} \in \mathcal{Q}$ at the current time period, if there is no purchase, then the state at the next time period is given by the random vector $\boldsymbol{X}(\boldsymbol{q}) = \big(X_{i,0}(\boldsymbol{q}), X^j_{i,\ell}(\boldsymbol{q}) : i \in \mathcal{N},\ j \in \mathcal{M},\ \ell \geq 1\big)$, where we have

$$\begin{aligned} X_{i,0}(\boldsymbol{q}) \ &= \ q_{i,0} + \sum_{j \in \mathcal{M}} \sum_{s=1}^{\infty} \mathsf{Bin}\left(q^j_{i,s}, \rho^j_{i,s}\right), \\ X^j_{i,\ell}(\boldsymbol{q}) \ &= \ \begin{cases} 0 & \text{if } \ell = 1, \\ q^j_{i,\ell-1} - \ \mathsf{Bin}(q^j_{i,\ell-1}, \rho^j_{i,\ell-1}) & \text{if } \ell \geq 2. \end{cases} \end{aligned}$$

The transition dynamics above are similar to the one in (1). The only difference is that we need to keep track of the types of the customers using the units. Let $J^t(\boldsymbol{q})$ denote the maximum total expected revenue over the time periods $t, t+1, \ldots, T$, given that the system is in state $\boldsymbol{q}$ at time period t. We can compute $\{J^t : t \in \mathcal{T}\}$ by solving the dynamic program

$$\begin{aligned} J^t(\boldsymbol{q}) = &\sum_{i \in \mathcal{N}} \sum_{j \in \mathcal{M}} \pi^j_i \sum_{\ell=1}^{\infty} q^j_{i,\ell} \\ &+ \max_{(S^1, \ldots, S^m) \in \mathcal{F}^1 \times \ldots \times \mathcal{F}^m} \Bigg\{ \sum_{j \in \mathcal{M}} p^{t,j} \sum_{i \in \mathcal{N}} \mathbb{1}_{\{q_{i,0} \geq 1\}} \, \phi^j_i(S^j) \times \\ &\qquad \Bigg(r^j_i + \pi^j_i + \mathbb{E}\Big\{ \mathsf{Z}(\rho^j_{i,0}) \, J^{t+1}(\boldsymbol{X}(\boldsymbol{q})) + (1 - \mathsf{Z}(\rho^j_{i,0})) \, J^{t+1}\left(\boldsymbol{X}(\boldsymbol{q}) - \boldsymbol{e}_{i,0} + \boldsymbol{e}^j_{i,1}\right) \Big\} \Bigg) \\ &\qquad + \Big(1 - \sum_{j \in \mathcal{M}} p^{t,j} \sum_{i \in \mathcal{N}} \mathbb{1}_{\{q_{i,0} \geq 1\}} \phi^j_i(S^j)\Big) \mathbb{E}\Big\{ J^{t+1}(\boldsymbol{X}(\boldsymbol{q})) \Big\} \Bigg\}, \end{aligned}$$

with the boundary condition that $J^{T+1}(\cdot) = 0$. Here, $\boldsymbol{e}^j_{i,1}$ is the unit vector with a one in the $(i,1)$-th coordinate associated with a customer of type j and zero everywhere else. Note that the dynamic program above is very similar to the dynamic program in (2). As in Section 2, we can write the dynamic program above equivalently as

$$J^t(\boldsymbol{q}) = \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}} \pi_i^j \sum_{\ell=1}^{\infty} q_{i,\ell}^j \;+\; \mathbb{E}\Big\{J^{t+1}(\boldsymbol{X}(\boldsymbol{q}))\Big\}$$
$$+\sum_{j\in\mathcal{M}} p^{t,j} \max_{S^j\in\mathcal{F}^j}\left\{\sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\geq 1\}}\phi_i^j(S^j)\left(r_i^j+\pi_i^j-(1-\rho_{i,0}^j)\,\mathbb{E}\Big\{J^{t+1}(\boldsymbol{X}(\boldsymbol{q}))-J^{t+1}\left(\boldsymbol{X}(\boldsymbol{q})-\boldsymbol{e}_{i,0}+\boldsymbol{e}_{i,1}^j\right)\Big\}\right)\right\}. \tag{11}$$

Once we observe that the maximization problem in our initial dynamic programming formulation decomposes by the customer types, the way we obtain the dynamic program above from the initial one is similar to the way we obtain the dynamic program in (3) from (2). Next, we construct an approximation to the optimal value function and bound the optimal total expected revenue.

Assuming that we have no products in use at the beginning of the selling horizon, the initial state of the system is given by $\boldsymbol{q}^1=((q_{i,0}^1,q_{i,\ell}^{j,1}):i\in\mathcal{N},\ j\in\mathcal{M},\ \ell\geq 1)=((C_i,0):i\in\mathcal{N},\ j\in\mathcal{M},\ \ell\geq 1)$, so that the optimal total expected revenue is $J^1(\boldsymbol{q}^1)$. We use a value function approximation of the form $\hat{J}^t(\boldsymbol{q})=\hat{\beta}^t+\sum_{i\in\mathcal{N}}\hat{\theta}_i^t\,q_{i,0}+\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}}\sum_{\ell=1}^{\infty}\hat{\nu}_{i,\ell}^{t,j}\,q_{i,\ell}^j,$, where we compute $\hat{\beta}^t$, $\hat{\theta}_i^t$ and $\hat{\nu}_{i,\ell}^{t,j}$ as discussed in Section 5.1. We consider the greedy policy with respect to the value function approximations $\{\hat{J}^t:t\in\mathcal{T}\}$. If the system is in state $\boldsymbol{q}$ at time period t, then this policy offers the assortment $\hat{S}^{t,j}(\boldsymbol{q})$ to a customer of type j, which is given by

$$\begin{aligned}\hat{S}^{t,j}(\boldsymbol{q}) &= \arg\max_{S\in\mathcal{F}^j}\left\{\sum_{i\in\mathcal{N}}\mathbb{1}_{\{q_{i,0}\geq 1\}}\phi_i^j(S)\left(r_i^j+\pi_i^j-(1-\rho_{i,0}^j)\mathbb{E}\left\{\hat{J}^{t+1}(\boldsymbol{X}(\boldsymbol{q}))-\hat{J}^{t+1}\left(\boldsymbol{X}(\boldsymbol{q})-\boldsymbol{e}_{i,0}+\boldsymbol{e}_{i,1}^j\right)\right\}\right)\right\}\\ &= \arg\max_{S\in\mathcal{F}^j}\left\{\sum_{i=1}^{n}\mathbb{1}_{\{q_{i,0}\geq 1\}}\phi_i^j(S)\left(r_i^j+\pi_i^j-(1-\rho_{i,0}^j)\left[\hat{\theta}_i^{t+1}-\hat{\nu}_{i,1}^{t+1,j}\right]\right)\right\},\end{aligned}$$

where the second equality follows from the definition of $\hat{J}^t$. Our main result under heterogeneous customer types is stated in the following theorem.

Theorem D.1 (Performance of the Greedy Policy) *Under heterogeneous customer types, the total expected revenue of the greedy policy with respect to the value function approximations* $\left\{\hat{J}^t\,:\,t\in\mathcal{T}\right\}$ *is at least 50% of the optimal total expected revenue.*

To show Theorem D.1, we use the next lemma. Note that $\boldsymbol{q}^1=((q_{i,0}^1,q_{i,\ell}^{j,1}):i\in\mathcal{N},\ j\in\mathcal{M},\ \ell\geq 1)=((C_i,0):i\in\mathcal{N},\ j\in\mathcal{M},\ \ell\geq 1)$ is the initial state of the system.

Lemma D.2 $J^1(\boldsymbol{q}^1)\leq 2\sum_{i\in\mathcal{N}}\hat{\theta}_i^1\,C_i$.

Proof: We can obtain an upper bound on the optimal total expected revenue by using the objective value provided by any feasible solution to the linear program

$$\begin{aligned}
\min \quad & \tilde{J}^1\left(\boldsymbol{q}^1\right) \\
\text{s.t.} \quad & \tilde{J}^t(\boldsymbol{q}) \;\geq\; \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}} \pi_i^j \sum_{\ell=1}^{\infty} q_{i,\ell}^j \;+\; \mathbb{E}\Big\{\tilde{J}^{t+1}(\boldsymbol{X}(\boldsymbol{q}))\Big\} \\
& \quad + \sum_{j\in\mathcal{M}} p^{t,j} \sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\,\geq\,1\}}\, \phi_i^j(S^j)\bigg(r_i^j + \pi_i^j - (1-\rho_{i,0}^j)\,\mathbb{E}\Big\{\tilde{J}^{t+1}(\boldsymbol{X}(\boldsymbol{q})) - \tilde{J}^{t+1}\left(\boldsymbol{X}(\boldsymbol{q}) - \boldsymbol{e}_{i,0} + \boldsymbol{e}_{i,1}^j\right)\Big\}\bigg) \\
& \qquad\qquad \forall\, \boldsymbol{q}\in\mathcal{Q},\; (S^1,\ldots,S^m)\in\mathcal{F}^1\times\cdots\times\mathcal{F}^m,\; t\in\mathcal{T},
\end{aligned}$$

where the decision variables in the linear program above are $\{\tilde{J}^t(\boldsymbol{q}) : \boldsymbol{q}\in\mathcal{Q},\; t\in\mathcal{T}\}$ and we follow the convention that $\tilde{J}^{T+1}=0$. For any $\boldsymbol{q}\in\mathcal{Q},\; t\in\mathcal{T}$, let $\hat{J}^t(\boldsymbol{q}) = \hat{\beta}^t + \sum_{i\in\mathcal{N}}\hat{\theta}_i^t\, q_{i,0} + \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}}\sum_{\ell=1}^{\infty}\hat{\nu}_{i,\ell}^{t,j}\, q_{i,\ell}^j$, where we compute $\hat{\beta}^t$, $\hat{\theta}_i^t$ and $\hat{\nu}_{i,\ell}^{t,j}$ as discussed in Section 5.1. We claim that $\{\hat{J}^t(\boldsymbol{q}) : \boldsymbol{q}\in\mathcal{Q},\; t\in\mathcal{T}\}$ is a feasible solution to the linear program above. To establish the claim, noting that $\hat{J}^{t+1}(\boldsymbol{q})$ is linear in $\boldsymbol{q}$, by the definition of $\boldsymbol{X}(\boldsymbol{q})$, we get

$$\begin{aligned}
\mathbb{E}\Big\{\hat{J}^{t+1}(\boldsymbol{X}(\boldsymbol{q}))\Big\} &= \hat{\beta}^{t+1} + \sum_{i\in\mathcal{N}}\left\{\hat{\theta}_i^{t+1}\Big[q_{i,0} + \sum_{j\in\mathcal{M}}\sum_{\ell=1}^{\infty}\rho_{i,\ell}^j q_{i,\ell}^j\Big] + \sum_{j\in\mathcal{M}}\sum_{\ell=1}^{\infty}\hat{\nu}_{i,\ell+1}^{t+1,j}\,[q_{i,\ell}^j - \rho_{i,s}^j q_{i,\ell}^j]\right\} \\
&= \hat{\beta}^{t+1} + \sum_{i\in\mathcal{N}}\left\{q_{i,0}\,\hat{\theta}_i^{t+1} + \sum_{j\in\mathcal{M}}\sum_{\ell=1}^{\infty} q_{i,\ell}^j\,[\rho_{i,\ell}^j\,\hat{\theta}_i^{t+1} + (1-\rho_{i,\ell}^j)\,\hat{\nu}_{i,\ell+1}^{t+1,j}]\right\} \\
&= \hat{\beta}^{t+1} + \sum_{i\in\mathcal{N}}\left\{q_{i,0}\,\hat{\theta}_i^{t+1} + \sum_{j\in\mathcal{M}}\sum_{\ell=1}^{\infty} q_{i,\ell}^j\,[\hat{\nu}_{i,\ell}^{t,j} - \pi_i^j]\right\},
\end{aligned}$$

where the last equality follows from the way we compute $\hat{\nu}_{i,\ell}^{t+1,j}$ in Section 5.1. Similarly, using the fact that $\hat{J}^{t+1}(\boldsymbol{q})$ is linear in $\boldsymbol{q}$, we also have $\mathbb{E}\Big\{\hat{J}^{t+1}(\boldsymbol{X}(\boldsymbol{q})) - \hat{J}^{t+1}(\boldsymbol{X}(\boldsymbol{q}) - \boldsymbol{e}_{i,0} + \boldsymbol{e}_{i,1}^j)\Big\} = \hat{\theta}_i^{t+1} - \hat{\nu}_{i,1}^{t+1,j}$ by the definitions of $\hat{J}^{t+1}$ and $X(\boldsymbol{q})$. Thus, if we evaluate the right side of the constraint in the linear program above at $\{\hat{J}^t(\boldsymbol{q}) : \boldsymbol{q}\in\mathcal{Q},\; t\in\mathcal{T}\}$, then we get

$$\begin{aligned}
& \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}} \pi_i^j \sum_{\ell=1}^{\infty} q_{i,\ell}^j \;+\; \mathbb{E}\Big\{\tilde{J}^{t+1}(\boldsymbol{X}(\boldsymbol{q}))\Big\} \\
& \quad + \sum_{j\in\mathcal{M}} p^{t,j} \sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\,\geq\,1\}}\, \phi_i^j(S^j)\bigg(r_i^j + \pi_i^j - (1-\rho_{i,0}^j)\,\mathbb{E}\Big\{\tilde{J}^{t+1}(\boldsymbol{X}(\boldsymbol{q})) - \tilde{J}^{t+1}\left(\boldsymbol{X}(\boldsymbol{q}) - \boldsymbol{e}_{i,0} + \boldsymbol{e}_{i,1}^j\right)\Big\}\bigg) \\
&= \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}} \pi_i^j \sum_{\ell=1}^{\infty} q_{i,\ell}^j \;+\; \hat{\beta}^{t+1} + \sum_{i\in\mathcal{N}}\left\{q_{i,0}\,\hat{\theta}_i^{t+1} + \sum_{j\in\mathcal{M}}\sum_{\ell=1}^{\infty} q_{i,\ell}^j\,[\hat{\nu}_{i,\ell}^{t,j} - \pi_i^j]\right\} \\
& \quad + \sum_{j\in\mathcal{M}} p^{t,j} \sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\,\geq\,1\}}\, \phi_i^j(S^j)\Big(r_i^j + \pi_i^j - (1-\rho_{i,0}^j)\,(\hat{\theta}_i^{t+1} - \hat{\nu}_{i,1}^{t+1,j})\Big) \\
&= \sum_{i\in\mathcal{N}}\hat{\theta}_i^{t+1}\,C_i + \sum_{i\in\mathcal{N}}\left\{q_{i,0}\,\hat{\theta}_i^{t+1} + \sum_{j\in\mathcal{M}}\sum_{\ell=1}^{\infty} q_{i,\ell}^j\,\hat{\nu}_{i,\ell}^{t,j}\right\} \\
& \quad + \sum_{j\in\mathcal{M}} p^{t,j} \sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\,\geq\,1\}}\, \phi_i^j(S^j)\Big(r_i^j + \pi_i^j - (1-\rho_{i,0}^j)\,(\hat{\theta}_i^{t+1} - \hat{\nu}_{i,1}^{t+1,j})\Big),
\end{aligned}$$

where the second equality uses the definition of $\hat\beta^{t+1}$ in Section 5.1. Using the same argument in proof of Lemma 3.1, we can show that $\hat\theta_i^t \geq \hat\theta_i^{t+1}$ under heterogeneous customer types. Furthermore, using the same argument in the proof of Lemma A.1, we can show that $\sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\geq 1\}}\, \phi_i^j(S^j)\left[r_i^j+\pi_i^j-(1-\rho_{i,0}^j)\,(\hat\theta_i^{t+1}-\hat\nu_{i,1}^{t+1,j})\right] \leq \sum_{i\in\mathcal{N}} \phi_i(\hat A^{t,j}) \times \left[r_i^j+\pi_i^j-(1-\rho_{i,0}^j)\,(\hat\theta_i^{t+1}-\hat\nu_{i,1}^{t+1,j})\right]$ for all $S^j\in\mathcal{F}^j$. By the chain of equalities above, we can bound the right side of the constraint in the linear program as

$$\begin{aligned}
&\sum_{i\in\mathcal{N}} \hat\theta_i^{t+1} C_i + \sum_{i\in\mathcal{N}}\left\{q_{i,0}\,\hat\theta_i^{t+1} + \sum_{j\in\mathcal{M}}\sum_{\ell=1}^{\infty} q_{i,\ell}^j\,\hat\nu_{i,\ell}^{t,j}\right\}\\
&\qquad + \sum_{j\in\mathcal{M}} p^{t,j}\sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\geq 1\}}\,\phi_i^j(S^j)\left(r_i^j+\pi_i^j-(1-\rho_{i,0}^j)\,(\hat\theta_i^{t+1}-\hat\nu_{i,1}^{t+1,j})\right)\\
&\leq \sum_{i\in\mathcal{N}} \hat\theta_i^{t+1} C_i + \sum_{i\in\mathcal{N}}\left\{q_{i,0}\,\hat\theta_i^{t} + \sum_{j\in\mathcal{M}}\sum_{\ell=1}^{\infty} q_{i,\ell}^j\,\hat\nu_{i,\ell}^{t,j}\right\}\\
&\qquad + \sum_{j\in\mathcal{M}} p^{t,j}\sum_{i\in\mathcal{N}} \phi_i(\hat A^{t,j})\left(r_i^j+\pi_i^j-(1-\rho_{i,0}^j)\,(\hat\theta_i^{t+1}-\hat\nu_{i,1}^{t+1,j})\right)\\
&= \sum_{i\in\mathcal{N}} \hat\theta_i^{t+1} C_i + \sum_{i\in\mathcal{N}}\left\{q_{i,0}\,\hat\theta_i^{t} + \sum_{j\in\mathcal{M}}\sum_{\ell=1}^{\infty} q_{i,\ell}^j\,\hat\nu_{i,\ell}^{t,j}\right\} + \sum_{i\in\mathcal{N}} C_i\,(\hat\theta_i^t-\hat\theta_i^{t+1})\\
&= \sum_{i\in\mathcal{N}} \hat\theta_i^{t} C_i + \sum_{i\in\mathcal{N}}\left\{q_{i,0}\,\hat\theta_i^{t} + \sum_{j\in\mathcal{M}}\sum_{\ell=1}^{\infty} q_{i,\ell}^j\,\hat\nu_{i,\ell}^{t,j}\right\} = \hat\beta^t + \sum_{i\in\mathcal{N}}\left\{q_{i,0}\,\hat\theta_i^{t} + \sum_{j\in\mathcal{M}}\sum_{\ell=1}^{\infty} q_{i,\ell}^j\,\hat\nu_{i,\ell}^{t,j}\right\} = \hat J^t(\boldsymbol{q}),
\end{aligned}$$

where the first equality follows from the way we compute $\hat\theta_i^t$ in Section 5.1. By the chain of inequalities above, for any $\boldsymbol{q}\in\mathcal{Q}$, $(S^1,\ldots,S^m)\in\mathcal{F}^1\times\ldots\times\mathcal{F}^m$, $t\in\mathcal{T}$, if we evaluate the right side of the constraint in the linear program at $\{\hat J(\boldsymbol{q}):\boldsymbol{q}\in\mathcal{Q},\ t\in\mathcal{T}\}$, then the right side of the constraint is upper bounded by $\hat J^t(\boldsymbol{q})$. Therefore, the solution $\{\hat J(\boldsymbol{q}):\boldsymbol{q}\in\mathcal{Q},\ t\in\mathcal{T}\}$ is feasible to the linear program, in which case, the objective value provided by this solution is an upper bound on the optimal objective value of the linear program. Noting the definition of $\boldsymbol{q}^1$, the objective value provided by the solution $\{\hat J(\boldsymbol{q}):\boldsymbol{q}\in\mathcal{Q},\ t\in\mathcal{T}\}$ is $\hat J^1(\boldsymbol{q}^1)=\hat\beta^1+\sum_{i\in\mathcal{N}} C_i\,\hat\theta_i^1 = 2\sum_{i\in\mathcal{N}} C_i\,\hat\theta_i^1$. Thus, $2\sum_{i\in\mathcal{N}} C_i\,\hat\theta_i^1$ is an upper bound on the optimal total expected revenue. ■

If we are in state $\boldsymbol{q}$ at time period t, then the greedy policy offers the assortment $\hat S^{t,j}(\boldsymbol{q})$ to a customer of type j. Let $U^t(\boldsymbol{q})$ be the total expected revenue obtained by the greedy policy over the time periods $t,\ldots,T$, given that we are in state $\boldsymbol{q}$ at time period t. Using an argument similar to the one right before the proof of Theorem 3.2 and noting our dynamic programming formulation under heterogeneous customer types in (11), we can compute $\{U^t : t\in\mathcal{T}\}$ by using the recursion

$$\begin{aligned}
U^t(\boldsymbol{q}) &= \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}} \pi_i^j \sum_{\ell=1}^{\infty} q_{i,\ell}^j \;+\; \mathbb{E}\Big\{U^{t+1}(\boldsymbol{X}(\boldsymbol{q}))\Big\}\\
&\quad + \sum_{j\in\mathcal{M}} p^{t,j}\sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\geq 1\}}\,\phi_i^j(\hat S^{t,j}(\boldsymbol{q}))\left(r_i^j+\pi_i^j-(1-\rho_{i,0}^j)\,\mathbb{E}\Big\{U^{t+1}(\boldsymbol{X}(\boldsymbol{q}))-U^{t+1}\left(\boldsymbol{X}(\boldsymbol{q})-\boldsymbol{e}_{i,0}+\boldsymbol{e}_{i,1}^j\right)\Big\}\right),
\end{aligned}$$

with the boundary condition that $U^{T+1}=0$. The coefficient of $\mathbb{E}\{U^{t+1}(\boldsymbol{X}(\boldsymbol{q}))\}$ above is $1-\sum_{j\in\mathcal{M}} p^{t,j}\sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\geq 1\}}\,\phi_i^j(\hat{S}^{t,j}(\boldsymbol{q}))(1-\rho_{i,0}^j)$. Since $\sum_{j\in\mathcal{M}} p^{t,j}=1$ and $\sum_{i\in\mathcal{N}}\phi_i^j(\hat{S}^{t,j}(\boldsymbol{q}))\leq 1$, this coefficient is positive. The coefficient of $E\Big\{U^{t+1}\left(\boldsymbol{X}(\boldsymbol{q})-\boldsymbol{e}_{i,0}+\boldsymbol{e}_{i,1}^j\right)\Big\}$ is positive as well. Thus, if we replace the U^{t+1} on the right side above with a function H^{t+1} that satisfies $U^{t+1}(\boldsymbol{q})\geq H^{t+1}(\boldsymbol{q})$, then the right side of the expression above gets smaller.

Here is the proof of Theorem D.1.

Proof of Theorem D.1: Let $H^t(\boldsymbol{q})=\sum_{i\in\mathcal{N}}\Big\{\hat{\theta}_i^t\,q_{i,0}+\sum_{j\in\mathcal{M}}\sum_{\ell=1}^{\infty}\hat{\nu}_{i,\ell}^{t,j}\,q_{i,\ell}^j\Big\}$ for all $t\in\mathcal{T}$ and $\boldsymbol{q}\in\mathcal{Q}^{\mathcal{M}}$. We will use induction over the time periods to show that $U^t(\boldsymbol{q})\geq H^t(\boldsymbol{q})$ for all $\boldsymbol{q}\in\mathcal{Q}$ and $t\in\mathcal{T}$. In this case, noting that the initial state of the system is given by $\boldsymbol{q}^1$, we obtain $U^1(\boldsymbol{q}^1)\geq H^1(\boldsymbol{q}^1)=\sum_{i\in\mathcal{N}}\hat{\theta}_i^1\,C_i\geq\frac{1}{2}J^1(\boldsymbol{q}^1)$, where the equality follows from the definition of $\boldsymbol{q}^1$ and the second inequality follows from Lemma D.2. Thus, the desired result follows. We proceed to using induction over the time periods to show that $U^t(\boldsymbol{q})\geq H^t(\boldsymbol{q})$. Since $\hat{\nu}_i^{T+1}=0$ and $\hat{\nu}_{i,\ell}^{T+1,j}=0$ for all $i\in\mathcal{N}$, $j\in\mathcal{M}$, $\ell=1,2,\ldots$, we have $H^{T+1}=0$. We have $U^{T+1}=0$ as well. Therefore, the result holds at time period $T+1$. Assuming that $U^{t+1}(\boldsymbol{q})\geq H^{t+1}(\boldsymbol{q})$ for all $\boldsymbol{q}\in\mathcal{Q}$, we proceed to showing that $U^t(\boldsymbol{q})\geq H^t(\boldsymbol{q})$ for all $\boldsymbol{q}\in\mathcal{Q}$ as well. Since H^{t+1} is linear, by the same argument in the proof of Lemma D.2, we have

$$
\begin{aligned}
\mathbb{E}\Big\{H^{t+1}(\boldsymbol{X}(\boldsymbol{q}))\Big\} &= \sum_{i\in\mathcal{N}}\bigg\{\hat{\theta}_i^{t+1}\Big[q_{i,0}+\sum_{j\in\mathcal{M}}\sum_{\ell=1}^{\infty}\rho_{i,\ell}^j q_{i,\ell}^j\Big]+\sum_{j\in\mathcal{M}}\sum_{\ell=1}^{\infty}\hat{\nu}_{i,\ell+1}^{t+1,j}\,[q_{i,\ell}^j-\rho_{i,s}^j q_{i,\ell}^j]\bigg\}\\
&= \sum_{i\in\mathcal{N}}\bigg\{q_{i,0}\,\hat{\theta}_i^{t+1}+\sum_{j\in\mathcal{M}}\sum_{\ell=1}^{\infty}q_{i,\ell}^j\,[\rho_{i,\ell}^j\,\hat{\theta}_i^{t+1}+(1-\rho_{i,\ell}^j)\,\hat{\nu}_{i,\ell+1}^{t+1,j}]\bigg\}\\
&= \sum_{i\in\mathcal{N}}\bigg\{q_{i,0}\,\hat{\theta}_i^{t+1}+\sum_{j\in\mathcal{M}}\sum_{\ell=1}^{\infty}q_{i,\ell}^j\,[\hat{\nu}_{i,\ell}^{t,j}-\pi_i^j]\bigg\}.
\end{aligned}
$$

By the discussion in the same proof, we have $\mathbb{E}\Big\{H^{t+1}(\boldsymbol{X}(\boldsymbol{q}))-H^{t+1}(\boldsymbol{X}(\boldsymbol{q})-\boldsymbol{e}_{i,0}+\boldsymbol{e}_{i,1}^j)\Big\}=\hat{\theta}_i^{t+1}-\hat{\nu}_{i,1}^{t+1,j}$ as well. So, by the inductive hypothesis and the recursion that defines $U^t(\boldsymbol{q})$, we get

$$
\begin{aligned}
U^t(\boldsymbol{q}) \;\geq\; & \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}}\pi_i^j\sum_{\ell=1}^{\infty}q_{i,\ell}^j \;+\; \mathbb{E}\Big\{H^{t+1}(\boldsymbol{X}(\boldsymbol{q}))\Big\}\\
&+\sum_{j\in\mathcal{M}}p^{t,j}\sum_{i\in\mathcal{N}}\mathbb{1}_{\{q_{i,0}\geq 1\}}\,\phi_i^j(\hat{S}^{t,j}(\boldsymbol{q}))\bigg(r_i^j+\pi_i^j-(1-\rho_{i,0}^j)\,\mathbb{E}\Big\{H^{t+1}(\boldsymbol{X}(\boldsymbol{q}))-H^{t+1}\left(\boldsymbol{X}(\boldsymbol{q})-\boldsymbol{e}_{i,0}+\boldsymbol{e}_{i,1}^j\right)\Big\}\bigg)\\
=\; & \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}}\pi_i^j\sum_{\ell=1}^{\infty}q_{i,\ell}^j \;+\; \sum_{i\in\mathcal{N}}\bigg\{q_{i,0}\,\hat{\theta}_i^{t+1}+\sum_{j\in\mathcal{M}}\sum_{\ell=1}^{\infty}q_{i,\ell}^j\,[\hat{\nu}_{i,\ell}^{t,j}-\pi_i^j]\bigg\}\\
&+\sum_{j\in\mathcal{M}}p^{t,j}\sum_{i\in\mathcal{N}}\mathbb{1}_{\{q_{i,0}\geq 1\}}\,\phi_i^j(\hat{S}^{t,j}(\boldsymbol{q}))\bigg(r_i^j+\pi_i^j-(1-\rho_{i,0}^j)\,(\hat{\theta}_i^{t+1}-\hat{\nu}_{i,1}^{t+1,j})\bigg)\\
=\; & \sum_{i\in\mathcal{N}}\bigg\{q_{i,0}\,\hat{\theta}_i^{t+1}+\sum_{j\in\mathcal{M}}\sum_{\ell=1}^{\infty}q_{i,\ell}^j\,\hat{\nu}_{i,\ell}^{t,j}\bigg\}
\end{aligned}
$$

$$+\sum_{j\in\mathcal{M}} p^{t,j} \max_{S^j\in\mathcal{F}^j}\left\{\sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\geq 1\}}\,\phi_i^j(S^j)\left(r_i^j+\pi_i^j-(1-\rho_{i,0}^j)\,(\hat{\theta}_i^{t+1}-\hat{\nu}_{i,1}^{t+1,j})\right)\right\},$$

where the last equality is by the definition of $\hat{S}^{t,j}(\boldsymbol{q})$. Noting that $\hat{A}^{t,j}$ is a feasible, but not necessarily an optimal, solution to the last maximization problem, we have

$$\begin{aligned}
&\sum_{i\in\mathcal{N}}\left\{q_{i,0}\,\hat{\theta}_i^{t+1}+\sum_{j\in\mathcal{M}}\sum_{\ell=1}^{\infty} q_{i,\ell}^j\,\hat{\nu}_{i,\ell}^{t,j}\right\}\\
&\qquad\qquad+\sum_{j\in\mathcal{M}} p^{t,j}\max_{S^j\in\mathcal{F}^j}\left\{\sum_{i\in\mathcal{N}}\mathbb{1}_{\{q_{i,0}\geq 1\}}\,\phi_i^j(S^j)\left(r_i^j+\pi_i^j-(1-\rho_{i,0}^j)\,(\hat{\theta}_i^{t+1}-\hat{\nu}_{i,1}^{t+1,j})\right)\right\}\\
&\geq \sum_{i\in\mathcal{N}}\left\{q_{i,0}\,\hat{\theta}_i^{t+1}+\sum_{j\in\mathcal{M}}\sum_{\ell=1}^{\infty} q_{i,\ell}^j\,\hat{\nu}_{i,\ell}^{t,j}\right\}+\sum_{j\in\mathcal{M}} p^{t,j}\sum_{i\in\mathcal{N}}\mathbb{1}_{\{q_{i,0}\geq 1\}}\,\phi_i^j(\hat{A}^{t,j})\left(r_i^j+\pi_i^j-(1-\rho_{i,0}^j)\,(\hat{\theta}_i^{t+1}-\hat{\nu}_{i,1}^{t+1,j})\right)\\
&\geq \sum_{i\in\mathcal{N}}\left\{q_{i,0}\,\hat{\theta}_i^{t+1}+\sum_{j\in\mathcal{M}}\sum_{\ell=1}^{\infty} q_{i,\ell}^j\,\hat{\nu}_{i,\ell}^{t,j}\right\}+\sum_{j\in\mathcal{M}} p^{t,j}\sum_{i\in\mathcal{N}}\frac{q_{i,0}}{C_i}\,\phi_i^j(\hat{A}^{t,j})\left(r_i^j+\pi_i^j-(1-\rho_{i,0}^j)\,(\hat{\theta}_i^{t+1}-\hat{\nu}_{i,1}^{t+1,j})\right)\\
&= \sum_{i\in\mathcal{N}}\left\{q_{i,0}\,\hat{\theta}_i^{t+1}+\sum_{j\in\mathcal{M}}\sum_{\ell=1}^{\infty} q_{i,\ell}^j\,\hat{\nu}_{i,\ell}^{t,j}\right\}+\sum_{j\in\mathcal{M}} p^{t,j}\sum_{i\in\mathcal{N}} q_{i,0}\,(\hat{\theta}_i^t-\hat{\theta}_i^{t+1})\\
&= \sum_{i\in\mathcal{N}}\left\{\hat{\theta}_i^t\,q_{i,0}+\sum_{j\in\mathcal{M}}\sum_{\ell=1}^{\infty}\hat{\nu}_{i,\ell}^{t,j}\,q_{i,\ell}^j\right\}=H^t(\boldsymbol{q}).
\end{aligned}$$

In the second inequality above, we can use an argument similar to the one in the proof of Lemma 3.1 to show that $\phi_i^j(\hat{A}^{t,j})(r_i^j+\pi_i^j-(1-\rho_{i,0}^j)\,(\hat{\theta}_i^{t+1}-\hat{\nu}_{i,1}^{t+1,j}))\geq 0$ for all $i\in\mathcal{N}$, in which case, the second inequality follows by the fact that $\mathbb{1}_{\{q_{i,0}\geq 1\}}\geq q_{i,0}/C_i$ for any $q_{i,0}\leq C_i$. The first equality follows from the way we compute $\hat{\theta}_i^t$. The two chains of inequalities show that $U^t(\boldsymbol{q})\geq H^t(\boldsymbol{q})$, establishing the desired claim. ∎

Appendix E: Solving the Assortment Problem Approximately

We consider the case where we solve problem (4) only approximately. Assume that we have an FPTAS such that for any $\epsilon>0$, the FPTAS finds an assortment $\hat{A}^t$ that satisfies

$$(1+\epsilon)\sum_{i\in\mathcal{N}}\phi_i(\hat{A}^t)\Big[r_i+\pi_i-(1-\rho_{i,0})\left(\hat{\nu}_{i,0}^{t+1}-\hat{\nu}_{i,1}^{t+1}\right)\Big]\geq\max_{S\in\mathcal{F}}\sum_{i\in\mathcal{N}}\phi_i(S)\Big[r_i+\pi_i-(1-\rho_{i,0})\left(\hat{\nu}_{i,0}^{t+1}-\hat{\nu}_{i,1}^{t+1}\right)\Big]$$

in running time that is polynomial in n and $1/\epsilon$. We compute $\hat{\nu}_{i,\ell}^t$ and $\hat{\beta}^t$ for all $i\in\mathcal{N}$, $\ell\geq 0$, $t\in\mathcal{T}$ as in (5), but the assortment $\hat{A}^t$ satisfies the inequality above, rather than being an optimal solution to problem (4). In other words, the assortment $\hat{A}^t$ is a $1/(1+\epsilon)$-approximate solution to problem (4). In the next lemma, we generalize Lemma 3.3.

Lemma E.1 *Assume that $\hat{A}^t$ is an $1/(1+\epsilon)$-approximate solution to problem (4) for all $t\in\mathcal{T}$. Then, we have $J^1\left(\sum_{i\in\mathcal{N}}C_i\,\boldsymbol{e}_{i,0}\right)\leq 2\,(1+\epsilon)^T\sum_{i\in\mathcal{N}}C_i\,\hat{\nu}_{i,0}^1$.*

Proof: Assume that $\hat{\nu}^t_{i,\ell}$ and $\hat{\beta}^t$ for all $i \in \mathcal{N}$, $\ell \geq 0$, $t \in \mathcal{T}$ are computed as in (5), but $\hat{A}^t$ is a $1/(1+\epsilon)$-approximate solution to problem (4). Defining the value function approximation $\hat{J}^t(\boldsymbol{q}) = \sum_{i\in\mathcal{N}} \sum_{\ell=0}^{\infty} \hat{\nu}^t_{i,\ell}\, q_{i,\ell}$, we claim that $\{(1+\epsilon)^{T-t+1} \hat{J}^t(\boldsymbol{q}) : \boldsymbol{q} \in \mathcal{Q},\ t \in \mathcal{T}\}$ is a feasible solution to the linear program in the proof of Lemma 3.3. To show the claim, from the discussion in the proof of Lemma 3.3, recall that $\mathbb{E}\{\hat{J}^{t+1}(\boldsymbol{X}(\boldsymbol{q})) - \hat{J}^{t+1}(\boldsymbol{X}(\boldsymbol{q}) - \boldsymbol{e}_{i,0} + \boldsymbol{e}_{i,1})\} = \hat{\nu}^{t+1}_{i,0} - \hat{\nu}^{t+1}_{i,1}$ and $\mathbb{E}\{\hat{J}^{t+1}(\boldsymbol{X}(\boldsymbol{q}))\} = \hat{\beta}^{t+1} + \sum_{i\in\mathcal{N}} \{q_{i,0}\, \hat{\nu}^{t+1}_{i,0} + \sum_{\ell=1}^{\infty} q_{i,\ell}\, [\rho_{i,\ell}\, \hat{\nu}^{t+1}_{i,0} + (1-\rho_{i,\ell})\, \hat{\nu}^{t+1}_{i,\ell+1}]\}$. In this case, if we evaluate the right side of the constraint in the linear program in the proof of Lemma 3.3 at the solution $\{(1+\epsilon)^{T-t+1} \hat{J}^t(\boldsymbol{q}) : \boldsymbol{q} \in \mathcal{Q},\ t \in \mathcal{T}\}$, then we obtain

$$
\begin{aligned}
&\sum_{i\in\mathcal{N}} \pi_i \sum_{\ell=1}^{\infty} q_{i,\ell} \;+\; (1+\epsilon)^{T-t} \mathbb{E}\left\{\hat{J}^{t+1}(\boldsymbol{X}(\boldsymbol{q}))\right\} \\
&\qquad + \sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\geq 1\}}\, \phi_i(S) \left[r_i + \pi_i - (1-\rho_{i,0})\,(1+\epsilon)^{T-t}\, \mathbb{E}\left\{ \hat{J}^{t+1}(\boldsymbol{X}(\boldsymbol{q})) - \hat{J}^{t+1}(\boldsymbol{X}(\boldsymbol{q}) - \boldsymbol{e}_{i,0} + \boldsymbol{e}_{i,1}) \right\} \right] \\
&= \sum_{i\in\mathcal{N}} \pi_i \sum_{\ell=1}^{\infty} q_{i,\ell} + (1+\epsilon)^{T-t}\, \hat{\beta}^{t+1} + (1+\epsilon)^{T-t} \sum_{i\in\mathcal{N}} \left\{ q_{i,0}\, \hat{\nu}^{t+1}_{i,0} + \sum_{\ell=1}^{\infty} q_{i,\ell} \left[\rho_{i,\ell}\, \hat{\nu}^{t+1}_{i,0} + (1-\rho_{i,\ell})\, \hat{\nu}^{t+1}_{i,\ell+1} \right] \right\} \\
&\qquad + \sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\geq 1\}}\, \phi_i(S) \left[r_i + \pi_i - (1+\epsilon)^{T-t}\,(1-\rho_{i,0})\,(\hat{\nu}^{t+1}_{i,0} - \hat{\nu}^{t+1}_{i,1}) \right] \\
&\leq (1+\epsilon)^{T-t} \sum_{i\in\mathcal{N}} \hat{\nu}^{t+1}_{i,0}\, C_i + (1+\epsilon)^{T-t} \sum_{i\in\mathcal{N}} \left\{ q_{i,0}\, \hat{\nu}^{t+1}_{i,0} + \sum_{\ell=1}^{\infty} q_{i,\ell}\, \hat{\nu}^t_{i,\ell} \right\} \\
&\qquad + \sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\geq 1\}}\, \phi_i(S) \left[r_i + \pi_i - (1-\rho_{i,0})\,(1+\epsilon)^{T-t}\,(\hat{\nu}^{t+1}_{i,0} - \hat{\nu}^{t+1}_{i,1}) \right],
\end{aligned}
$$

where the inequality above uses the fact that $\hat{\beta}^{t+1} = \sum_{i\in\mathcal{N}} \hat{\nu}^{t+1}_{i,0}\, C_i$ and $\hat{\nu}^t_{i,\ell} = \pi_i + \rho_{i,\ell}\, \hat{\nu}^{t+1}_{i,0} + (1-\rho_{i,\ell})\, \hat{\nu}^{t+1}_{i,\ell+1}$. By Lemma A.1, we have $\sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\geq 1\}}\, \phi_i(S) \left[r_i + \pi_i - (1-\rho_{i,0})\,(\hat{\nu}^{t+1}_{i,0} - \hat{\nu}^{t+1}_{i,1}) \right] \leq \max_{A\in\mathcal{F}} \sum_{i\in\mathcal{N}} \phi_i(A) \left[r_i + \pi_i - (1-\rho_{i,0})\,(\hat{\nu}^{t+1}_{i,0} - \hat{\nu}^{t+1}_{i,1}) \right]$ for all $S \in \mathcal{F}$, since this lemma assumes that $\hat{A}^t$ is chosen as the optimal solution to the last maximization problem. Therefore, we can continue the chain of inequalities above as

$$
\begin{aligned}
&(1+\epsilon)^{T-t} \sum_{i\in\mathcal{N}} \hat{\nu}^{t+1}_{i,0}\, C_i + (1+\epsilon)^{T-t} \sum_{i\in\mathcal{N}} \left\{ q_{i,0}\, \hat{\nu}^{t+1}_{i,0} + \sum_{\ell=1}^{\infty} q_{i,\ell} \hat{\nu}^t_{i,\ell} \right\} \\
&\qquad + \sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\geq 1\}}\, \phi_i(S) \left[r_i + \pi_i - (1-\rho_{i,0})\,(1+\epsilon)^{T-t}\,(\hat{\nu}^{t+1}_{i,0} - \hat{\nu}^{t+1}_{i,1}) \right] \\
&\leq (1+\epsilon)^{T-t} \sum_{i\in\mathcal{N}} \hat{\nu}^{t+1}_{i,0}\, C_i + (1+\epsilon)^{T-t} \sum_{i\in\mathcal{N}} \left\{ q_{i,0}\, \hat{\nu}^{t+1}_{i,0} + \sum_{\ell=1}^{\infty} q_{i,\ell} \hat{\nu}^t_{i,\ell} \right\} \\
&\qquad + (1+\epsilon)^{T-t} \sum_{i\in\mathcal{N}} \mathbb{1}_{\{q_{i,0}\geq 1\}}\, \phi_i(S) \left[r_i + \pi_i - (1-\rho_{i,0})\,(\hat{\nu}^{t+1}_{i,0} - \hat{\nu}^{t+1}_{i,1}) \right] \\
&\leq (1+\epsilon)^{T-t} \sum_{i\in\mathcal{N}} \hat{\nu}^{t+1}_{i,0}\, C_i + (1+\epsilon)^{T-t} \sum_{i\in\mathcal{N}} \left\{ q_{i,0}\, \hat{\nu}^{t+1}_{i,0} + \sum_{\ell=1}^{\infty} q_{i,\ell} \hat{\nu}^t_{i,\ell} \right\} \\
&\qquad + (1+\epsilon)^{T-t+1} \frac{1}{1+\epsilon} \max_{A\in\mathcal{F}} \left\{ \sum_{i\in\mathcal{N}} \phi_i(A) \left[r_i + \pi_i - (1-\rho_{i,0})\,(\hat{\nu}^{t+1}_{i,0} - \hat{\nu}^{t+1}_{i,1}) \right] \right\}
\end{aligned}
$$

$$\begin{aligned}
&\leq \ (1+\epsilon)^{T-t}\sum_{i\in\mathcal{N}}\hat{\nu}_{i,0}^{t+1}\,C_i+(1+\epsilon)^{T-t}\sum_{i\in\mathcal{N}}\left\{q_{i,0}\,\hat{\nu}_{i,0}^{t+1}+\sum_{\ell=1}^{\infty}q_{i,\ell}\hat{\nu}_{i,\ell}^{t}\right\}\\
&\qquad\qquad +\ (1+\epsilon)^{T-t+1}\sum_{i\in\mathcal{N}}\phi_i(\hat{A}^t)\left[r_i+\pi_i-(1-\rho_{i,0})\,(\hat{\nu}_{i,0}^{t+1}-\hat{\nu}_{i,1}^{t+1})\right],\\
&= \ (1+\epsilon)^{T-t}\sum_{i\in\mathcal{N}}\hat{\nu}_{i,0}^{t+1}\,C_i+(1+\epsilon)^{T-t}\sum_{i\in\mathcal{N}}\left\{q_{i,0}\,\hat{\nu}_{i,0}^{t+1}+\sum_{\ell=1}^{\infty}q_{i,\ell}\hat{\nu}_{i,\ell}^{t}\right\}\\
&\qquad\qquad +\ (1+\epsilon)^{T-t+1}\sum_{i\in\mathcal{N}}C_i\,(\hat{\nu}_{i,0}^{t}-\hat{\nu}_{i,0}^{t+1}),
\end{aligned}$$

where the last inequality is by the fact that $\hat{A}^t$ is $1/(1+\epsilon)$-approximate solution to problem (4) and the last equality follows the fact that $\hat{\nu}_{i,0}^t = \hat{\nu}_{i,0}^{t+1}+\frac{1}{C_i}\phi_i(\hat{A}^t)[r_i+\pi_i-(1-\rho_{i,0})\left(\hat{\nu}_{i,0}^{t+1}-\hat{\nu}_{i,1}^{t+1}\right)]$. Even if we choose $\hat{A}^t$ as an approximate solution to problem (4), we can follow precisely the same reasoning in the proof of Lemma 3.1 to show that we can drop each product i with $\phi_i(\hat{A}^t)\left(r_i+\pi_i-(1-\rho_{i,0})\left(\hat{\nu}_{i,0}^{t+1}-\hat{\nu}_{i,1}^{t+1}\right)\right)\leq 0$ from $\hat{A}^t$ without deteriorating the objective value of problem (4) provided by the solution $\hat{A}^t$. In this case, by the reasoning in the proof of Lemma 3.1, it follows that we can assume $\hat{\nu}_{i,0}^t\geq\hat{\nu}_{i,0}^{t+1}$. So, we continue the last chain of inequalities as

$$\begin{aligned}
&(1+\epsilon)^{T-t}\sum_{i\in\mathcal{N}}\hat{\nu}_{i,0}^{t+1}\,C_i+(1+\epsilon)^{T-t}\sum_{i\in\mathcal{N}}\left\{q_{i,0}\,\hat{\nu}_{i,0}^{t+1}+\sum_{\ell=1}^{\infty}q_{i,\ell}\hat{\nu}_{i,\ell}^{t}\right\}+(1+\epsilon)^{T-t+1}\sum_{i\in\mathcal{N}}C_i\,(\hat{\nu}_{i,0}^{t}-\hat{\nu}_{i,0}^{t+1})\\
&\quad\leq\ (1+\epsilon)^{T-t+1}\sum_{i\in\mathcal{N}}\hat{\nu}_{i,0}^{t+1}\,C_i+(1+\epsilon)^{T-t+1}\sum_{i\in\mathcal{N}}\left\{q_{i,0}\,\hat{\nu}_{i,0}^{t}+\sum_{\ell=1}^{\infty}q_{i,\ell}\hat{\nu}_{i,\ell}^{t}\right\}+(1+\epsilon)^{T-t+1}\sum_{i\in\mathcal{N}}C_i\,(\hat{\nu}_{i,0}^{t}-\hat{\nu}_{i,0}^{t+1})\\
&\quad=\ (1+\epsilon)^{T-t+1}\sum_{i\in\mathcal{N}}\left\{\hat{\nu}_{i,0}^{t}\,C_i+\sum_{\ell=0}^{\infty}q_{i,\ell}\hat{\nu}_{i,\ell}^{t}\right\}\ =\ (1+\epsilon)^{T-t+1}\left\{\hat{\beta}^t+\sum_{i\in\mathcal{N}}\sum_{\ell=0}^{\infty}q_{i,\ell}\hat{\nu}_{i,\ell}^{t}\right\}\ =\ (1+\epsilon)^{T-t+1}\,\hat{J}^t(\boldsymbol{q}).
\end{aligned}$$

By the discussion so far, for any $\boldsymbol{q}\in\mathcal{Q}$, $S\in\mathcal{F}$, $t\in\mathcal{T}$, if we evaluate the right side of the constraint at $\{(1+\epsilon)^{T-t+1}\,\hat{J}^t(\boldsymbol{q}):\boldsymbol{q}\in\mathcal{Q},\ t\in\mathcal{T}\}$, then the right side of the constraint is upper bounded by $(1+\epsilon)^{T-t+1}\hat{J}^t(\boldsymbol{q})$. Thus, the solution $\{(1+\epsilon)^{T-t+1}\hat{J}^t(\boldsymbol{q}):\boldsymbol{q}\in\mathcal{Q},\ t\in\mathcal{T}\}$ is feasible to the linear program, which implies that the objective value of the linear program at this solution is an upper bound on the optimal total expected revenue. The desired result follows by noting that the objective value of the linear program at the solution $\{(1+\epsilon)^{T-t+1}\hat{J}^t(\boldsymbol{q}):\boldsymbol{q}\in\mathcal{Q},\ t\in\mathcal{T}\}$ is $(1+\epsilon)^T\,\hat{J}^1\left(\sum_{i\in\mathcal{N}}C_i\,\boldsymbol{e}_{i,0}\right)=(1+\epsilon)^T\hat{\beta}^1+(1+\epsilon)^T\sum_{i\in\mathcal{N}}\hat{\nu}_{i,0}^1\,C_i=2\,(1+\epsilon)^T\sum_{i\in\mathcal{N}}\hat{\nu}_{i,0}^1\,C_i$. ■

Consider a greedy policy with respect to the value function approximations $\{\hat{J}^t:t\in\mathcal{T}\}$. To compute the decision of this policy, we need to solve the combinatorial optimization problem in (6), which has the same structure as the one in (4). Therefore, we assume that we can obtain only an approximate solution to this problem (6). In particular, if the state of the system at time period t is $\boldsymbol{q}$, then the greedy policy offers the assortment $\hat{S}^t(\boldsymbol{q})$ such that

$$(1+\epsilon)\,\sum_{i=1}^{n}\mathbb{1}_{\{q_{i,0}\geq 1\}}\phi_i(\hat{S}^t(\boldsymbol{q}))\Big[r_i+\pi_i-(1-\rho_{i,0})(\hat{\nu}_{i,0}^{t+1}-\hat{\nu}_{i,1}^{t+1})\Big]$$

$$\geq \max_{S \in \mathcal{F}} \sum_{i=1}^{n} \mathbb{1}_{\{q_{i,0} \geq 1\}} \phi_i(S) \Big[r_i + \pi_i - (1-\rho_{i,0})(\hat{\nu}_{i,0}^{t+1} - \hat{\nu}_{i,1}^{t+1}) \Big].$$

We can compute the total expected revenue obtained by this greedy policy through the recursion in (7). The only difference is that the assortment $\hat{S}^t(\boldsymbol{q})$ is a $1/(1+\epsilon)$-approximate solution to problem (6), rather than the optimal solution. We let $U^t(\boldsymbol{q})$ be the expected revenue obtained by the greedy policy over the time periods $t, \ldots, T$, given that the system is in state $\boldsymbol{q}$ at time period t. We have the following lemma for the total expected revenue of the greedy policy.

Lemma E.2 *For all $\boldsymbol{q} \in \mathcal{Q}$ and $t \in \mathcal{T}$, we have $(1+\epsilon)^{T-t+1} U^t(\boldsymbol{q}) \geq \sum_{i \in \mathcal{N}} \sum_{\ell=0}^{\infty} \hat{\nu}_{i,\ell}^t \, q_{i,\ell}$.*

The proof of this lemma is omitted and it follows by using induction over the time periods and using the ideas in the proofs of Theorem 3.2 and Lemma E.1. Here is the proof of Theorem 5.1.

Proof of Theorem 5.1: For any $\delta > 0$, we will show that we can obtain a $1/(2\,(1+\delta))$-approximate policy and the running time to obtain and execute the approximate policy is polynomial in n, $1/\delta$ and T. Assume for the moment that $\delta \leq 1$. Given such δ, set $\epsilon = \delta/(4T)$, choose $\hat{A}^t$ as a $1/(1+\epsilon)$-approximate solution to problem (4) and choose $\hat{S}^t(\boldsymbol{q})$ as a $1/(1+\epsilon)$-approximate solution to problem (6). Since we can obtain these approximate solutions in running times polynomial in n and $1/\epsilon$, the running times involved are polynomial in n and T/δ, which are, in turn, polynomial in n, $1/\delta$ and T, establishing the desired running time. Also, by Lemmas E.1 and E.2, the expected revenue from the greedy policy satisfies

$$2\,(1+\epsilon)^{2T}\, U^1\Bigg(\sum_{i \in \mathcal{N}} C_i\, \boldsymbol{e}_{i,0}\Bigg) \geq 2\,(1+\epsilon)^T \sum_{i \in \mathcal{N}} \hat{\nu}_{i,0}^1\, C_i \geq J^1\Bigg(\sum_{i \in \mathcal{N}} C_i\, \boldsymbol{e}_{i,0}\Bigg).$$

Letting Z^* be the optimal total expected revenue and G be the total expected revenue from the greedy policy, noting that $\epsilon = \delta/(4T)$, the chain of inequalities above yields $Z^* \leq 2\,(1+\frac{\delta}{4T})^{2T}\, G \leq 2\,\exp(\delta/2)\, G \leq 2\,(1+\delta)\, G$, where the last inequality follows from the fact that $\exp(\delta/2) \leq 1+\delta$ for all $\delta \in [0,1]$. The last chain of inequalities shows that the greedy policy is a $1/(2\,(1+\delta))$-approximate policy. Lastly, if $\delta > 1$, then we can simply choose $\epsilon = 1/(4T)$. ∎

Appendix F: Upper Bound on the Optimal Total Expected Revenue

In this section, we give a proof for Proposition 6.1.

Proof of Proposition 6.1: Under the optimal policy, we use $Q_{i,\ell}^t$ to denote the number of units of product i that have been in use for ℓ time periods at time period t. Also, under the optimal policy, we let $Z^t(A) = 1$ if we offer assortment A at time period t; otherwise, we have $Z^t(A) = 0$. Lastly, under the optimal policy, we let $\Phi_i^t = 1$ if the customer arriving at time period t chooses product

i; otherwise, we have $\Phi_i^t = 0$. Note that $Q_{i,\ell}^t$, $Z^t(A)$ and Φ_i^t are random variables. Furthermore, $\Pr\{\Phi_i^t = 1 \,|\, Z^t(A) = 1\} = \phi_i(A)$. Using the vector $\boldsymbol{Q}^t = (Q_{i,\ell}^t : i \in \mathcal{N},\ \ell \geq 0)$, by the transition dynamics of our dynamic assortment problem, we have

$$\begin{aligned}\boldsymbol{Q}^{t+1} = &\sum_{i\in\mathcal{N}} \Phi_i^t \Big[\mathsf{Z}(\rho_{i,0})\, \boldsymbol{X}(\boldsymbol{Q}^t) + (1-\mathsf{Z}(\rho_{i,0}))\,(\boldsymbol{X}(\boldsymbol{Q}^t) - \boldsymbol{e}_{i,0} + \boldsymbol{e}_{i,1})\Big] + \left\{1 - \sum_{i\in\mathcal{N}} \Phi_i^t\right\} \boldsymbol{X}(\boldsymbol{Q}^t) \\ = &\ \boldsymbol{X}(\boldsymbol{Q}^t) - \sum_{i\in\mathcal{N}} \Phi_i^t\,(1-\mathsf{Z}(\rho_{i,0}))\,(\boldsymbol{e}_{i,0} - \boldsymbol{e}_{i,1}),\end{aligned}$$

where the first equality uses an argument similar to the one that we use to justify the first constraint in problem (10). Since $\Pr\{\Phi_i^t = 1 \,|\, Z^t(A) = 1\} = \phi_i(A)$, we have $\mathbb{E}\{\Phi_i^t\} = \sum_{A\in\mathcal{F}} \Pr\{Z^t(A)=1\}\, \Pr\{\Phi_i^t = 1 \,|\, Z^t(A)=1\} = \sum_{A\in\mathcal{F}} \phi_i(A)\, \Pr\{Z^t(A)=1\}$. In this case, letting $\bar{q}_{i,\ell}^t = \mathbb{E}\{Q_{i,\ell}^t\}$ and $\bar{z}^t(A) = \mathbb{E}\{Z^t(A)\}$, taking expectations in the chain of equalities above and noting that $\mathbb{E}\{\boldsymbol{X}(\boldsymbol{q})\}$ is linear in $\boldsymbol{q}$, it follows that the solution $(\bar{z}^t(A) : A \in \mathcal{F},\ t \in \mathcal{T})$ and $(\bar{q}_{i,\ell}^t : i \in \mathcal{N},\ \ell \geq 0,\ t \in \mathcal{T})$ satisfies the first constraint in problem (10). Under the optimal policy, we start with the initial state $\sum_{i\in\mathcal{N}} C_i\, \boldsymbol{e}_{i,0}$ and offer one assortment at each time period, so $\boldsymbol{Q}^1 = \sum_{i\in\mathcal{N}} C_i\, \boldsymbol{e}_{i,0}$ and $\sum_{A\in\mathcal{F}} Z^t(A) = 1$. Taking expectations in the last two equalities indicates that the solution $(\bar{z}^t(A) : A \in \mathcal{F},\ t \in \mathcal{T})$ and $(\bar{q}_{i,\ell}^t : i \in \mathcal{N},\ \ell \leq 0,\ t \in \mathcal{T})$ satisfies the second and third constraints in problem (10) as well. Therefore, this solution is feasible to problem (10). Furthermore, noting that $\mathbb{E}\{\Phi_i^t\} = \sum_{A\in\mathcal{F}} \Pr\{Z^t(A)=1\}\, \Pr\{\Phi_i^t = 1 \,|\, Z^t(A)=1\} = \sum_{A\in\mathcal{F}} \phi_i(A)\, \Pr\{Z^t(A)=1\} = \sum_{A\in\mathcal{F}} \phi_i(A)\, \bar{z}^t(A)$, the total expected revenue under the optimal policy is

$$\begin{aligned}J^1\left(\sum_{i\in\mathcal{N}} C_i\, \boldsymbol{e}_{i,0}\right) &= \mathbb{E}\left\{\sum_{t\in\mathcal{T}}\sum_{i\in\mathcal{N}} r_i\, \Phi_i^t + \sum_{t\in\mathcal{T}}\sum_{i\in\mathcal{N}} \pi_i \sum_{\ell=1}^{\infty} Q_{i,\ell}^t\right\} \\ &= \sum_{t\in\mathcal{T}}\sum_{i\in\mathcal{N}} r_i \sum_{A\in\mathcal{F}} \phi_i(A)\, \bar{z}^t(A) + \sum_{t\in\mathcal{T}}\sum_{i\in\mathcal{N}} \pi_i \sum_{\ell=1}^{\infty} \bar{q}_{i,\ell}^t,\end{aligned}$$

which is the objective value that the solution $(\bar{z}^t(A) : A \in \mathcal{F},\ t \in \mathcal{T})$ and $(\bar{q}_{i,\ell}^t : i \in \mathcal{N},\ \ell \leq 0,\ t \in \mathcal{T})$ provides for problem (10). So, there exists a feasible solution to problem (10) that provides an objective value of $J^1\left(\sum_{i\in\mathcal{N}} C_i\, \boldsymbol{e}_{i,0}\right)$. Therefore, the optimal objective value of problem (10) must be at least $J^1\left(\sum_{i\in\mathcal{N}} C_i\, \boldsymbol{e}_{i,0}\right)$. ■

Appendix G: Data and Experimental Setup for Street Parking Pricing in the City of Seattle

As discussed in Section 6.3, we augmented the data provided by the Open Data Program in Seattle to ensure that we have an intended parking locale for each driver. In this case, each transaction record gives the start time, duration, intended locale, actual parked local, and per hour rate for each parking event. Note that the intended and the actual parked locales may be the same. The parking duration in the data reflects the duration of time for which each driver made a payment, but the driver may not occupy the parking space for this whole duration. Nevertheless, we assume that a driver indeed occupies the parking space for the whole duration of time for which she made a payment. Payments can be made by using a smart phone application that allows extending a parking session remotely. Therefore, it is reasonable to treat the usage duration of a parking space as a random quantity not known to the system operator at the time a driver parks.

When estimating the distribution of the parking space usage durations, we observed that the city of Seattle imposes parking time limits that prevent drivers from creating transactions with a duration greater than the maximum time limit. Such time limits result in an abnormally large fraction of transactions with durations that are exactly equal to the time limit, which created difficulties when estimating the parking duration distributions. Thus, we eliminated the transaction records whose durations are exactly equal to the time limit. After eliminating these transactions, a negative binomial distribution with parameters (s_i, η_i) with $s_i = 2$ gave reasonable fits. After eliminating the transactions, the load in the system was small enough that taking the future driver arrivals into consideration did not make an impact and simple policies performed remarkably well. To alleviate this problem, we artificially multiplied the arrival rates estimated from the data by a constant factor and decreased the number of parking spaces by another constant factor to obtain a reasonably large load. The multipliers that we use are given in Section 6.3.

We assume that the drivers with intended locale of j arrive into the system according to a Poisson process with the arrival rate function $\{\Lambda^{\tau,j} : \tau \geq 0\}$, where the time τ is measured in seconds. Recall that each time period in our model corresponds to a time interval of 30 seconds. In this case, the probability that a driver with intended locale of j arrives at time period t approximately with probability $p^{t,j} = 30 \times \Lambda^{f(t),j}$, where $f(t)$ is the time in the day corresponding to time period t in the selling horizon of our model. For estimation purposes, we assume that the arrival rate function $\{\Lambda^{\tau,j} : \tau \geq 0\}$ is constant over each 15 minute time interval. When a drive with intended locale j arrives into the system, we offer a price menu for the five locales that are closest to her intended locale. The driver makes a choice among these locales or decides to leave the system without parking. The latter decision may correspond to using a parking space that is not street parking. In our choice model, if we offer the assortment S, then a driver whose intended locale is j chooses to park in locale i with probability

$$\phi_i^j(S) = \frac{e^{\alpha^j + \beta\, \pi_{i,h}}}{1 + \sum_{(\ell,g)\in S} e^{\alpha^j + \beta\, \pi_{\ell,g}}}$$

as long as $(i,h) \in S$. The parameter β is the price sensitivity of the drivers and it is constant over across all drivers. This assumption helps us keep the number of parameters that we need to estimate manageable. We estimate the parameters β, $(\alpha^j : j \in \mathcal{M})$ and $(p^{t,j} : t \in \mathcal{T},\ j \in \mathcal{M})$ through maximum likelihood. The likelihood function that we use for this purpose closely mirrors the one used by Vulcano et al. (2012).

As discussed in Section 3.3 of Vulcano et al. (2012), when estimating the parameters of the choice model and the arrival rates, there is a continuum of choices for the parameters that yield the same value for the likelihood function. Therefore, we fix the no purchase probability of each driver. In particular, we focus on the time period 11AM to 4PM in our numerical study. The per hour parking rate for each locale in the data is fixed during this time period, but each locale has a different rate. Fixing the no purchase probability at 0.1, the no purchase probability for a driver with intended locale j needs to satisfy $1/(1+\sum_{(\ell,g)\in S^j} e^{\alpha^j+\beta\,\pi_{\ell,g}})=0.1$, where S^j is the set of locale and rate combinations offered to a driver with intended locale j. If we fix the parameter β, then the value of the parameter α^j is fixed by the last equality. Therefore, we estimate β and $(p^{t,j} : t\in\mathcal{T},\ j\in\mathcal{M})$ through maximum likelihood and determine the values of the parameters $(\alpha^j : j\in\mathcal{M})$ by the last equality. We estimated the parameters of the choice model and the arrival rates by using the data from 15 weekdays of June 2017. Using the data from the remaining five days of June 2017, we checked the percent deviation in the expected number of parkings according to our demand model and the number of parkings in the data over each hourly interval in each locale. The average absolute percent deviation was 22.26%.

When we estimated the parameters of the choice model through the data, the price sensitivity parameter estimate came out to be $\beta=-0.191$ with a standard error of 0.008. Following the magnitudes of the fares in the data, we allow the price of a parking space to take values \$2, \$4 or \$6 per hour. With these settings, the price sensitivity parameter turned out to be so small that changing the price of a parking space did not make discernible difference in the choices of the drivers. Therefore, we bumped the prices sensitivity parameter to $\beta=-0.5$ in all of our computational experiments.