

Constrained Assortment Optimization for the Nested Logit Model

Guillermo Gallego

Department of Industrial Engineering and Operations Research
Columbia University, New York, New York 10027, USA
gmg2@columbia.edu

Huseyin Topaloglu

School of Operations Research and Information Engineering,
Cornell University, Ithaca, New York 14853, USA
topaloglu@orie.cornell.edu

September 11, 2013

Abstract

We study assortment optimization problems where customer choices are governed by the nested logit model and there are constraints on the set of products offered in each nest. Under the nested logit model, the products are organized in nests. Each product in each nest has a fixed revenue associated with it. The goal is to find a feasible set of products, i.e. a feasible assortment, to maximize the expected revenue per customer. We consider cardinality and space constraints on the offered assortment, which respectively limit the number of products and the total space consumption of the products offered in each nest. We show that the optimal assortment under cardinality constraints can be obtained efficiently by solving a linear program. The assortment optimization problem under space constraints is NP-hard. We show how to obtain an assortment with a performance guarantee of two under space constraints. This assortment also provides a performance guarantee of $1/(1 - \epsilon)$ when the space requirement of each product is at most a fraction ϵ of the space availability in each nest. Building on our results for constrained assortment optimization, we show that we can efficiently solve joint assortment optimization and pricing problems under the nested logit model, where we choose the assortment of products to offer to customers, as well as the prices of the offered products.

Discrete choice models have long been used to describe how customers choose among a set of products that differ in attributes such as price and quality. Specifically, discrete choice models represent the demand for a particular product through the attributes of all products that are in the offered assortment, capturing substitution possibilities and complementary relationships between the products. To pursue this thought, different discrete choice models have been proposed in the literature. Some of these models are based on axioms as in Luce (1959), resulting in the basic attraction model, whereas some others are based on random utility theory as in McFadden (1974), resulting in the multinomial logit model. A popular extension to the multinomial logit model is the nested logit model introduced by Williams (1977). Under the nested logit model, the products are organized in nests. The choice process of a customer proceeds in such a way that the customer first selects a nest, and then a product within the selected nest.

In this paper, we study constrained assortment optimization problems when customers choose according to the nested logit model. There is a fixed revenue contribution associated with each product. The goal is to find an assortment of products to offer so as to maximize the expected revenue per customer subject to a constraint on the assortment offered in each nest. We consider two types of constraints, which we refer to as cardinality and space constraints. Cardinality constraints limit the number of products in the assortment offered in each nest. We show that the optimal assortment under cardinality constraints can be obtained by solving a linear program. Under space constraints, each product occupies a certain amount of space and we limit the total space consumption of the products offered in each nest. The assortment optimization problem under space constraints is NP-hard, but we show that we can solve a tractable linear program to obtain an assortment with a certain performance guarantee. These results establish that we can obtain provably good assortments under cardinality or space constraints.

In addition, we consider joint assortment optimization and pricing problems under the nested logit model. In the joint assortment optimization and pricing problem, the goal is to decide which assortment of products to offer and set the prices of the offered products. Customers choose among the offered products according to the nested logit model and the price of a product affects its attractiveness in the sense that if we set the price of a product higher, then it becomes less attractive to customers. Building on our results for constrained assortment optimization problems, we show that an optimal solution to the joint assortment optimization and pricing problem can be obtained efficiently by solving a linear program. Therefore, our results are not only useful for solving constrained assortment optimization problems, but they are also useful for pricing.

Main Contributions. In assortment optimization problems, we consider a setting with m nests, each including n products that we can offer to customers. Under cardinality constraints, we show that we can solve a linear program with $1 + m$ decision variables and $O(mn^2)$ constraints to obtain the optimal assortment. Under space constraints, we show that we can solve a linear program of the same size to obtain an assortment whose expected revenue deviates from the optimal expected revenue by at most a factor of two. Also, if each product consumes at most a fraction $\epsilon \in [0, 1)$

of the space availability in a nest, then we show that the expected revenue from this assortment deviates from the optimal by at most a factor of $1/(1 - \epsilon)$. So, we can obtain particularly good assortments when each product, by itself, occupies a small fraction of the space availability.

Our approach for dealing with space constraints is flexible enough that we can refine it to obtain assortments with arbitrarily good performance guarantees as long as we are willing to increase the computational effort. In particular, it is possible to show that if we want to obtain an assortment with an arbitrary performance guarantee of $\alpha > 1$ under space constraints, then we can solve a linear program with $1 + m$ decision variables and $O(m \lceil \alpha/(\alpha - 1) \rceil n^{\lceil \alpha/(\alpha - 1) \rceil + 2})$ constraints, where $\lceil \cdot \rceil$ is the round up function. This result is similar to a polynomial time approximation scheme and it shows that we can obtain arbitrarily good performance guarantees by choosing α close to one. The approach that we use to obtain the performance guarantee of two mentioned in the paragraph above serves as a crucial building block to obtain this more general approximation scheme. Furthermore, the approach for obtaining the performance guarantee of two demonstrates that the assortment optimization problem under space constraints is particularly simple to approximate when each item occupies a small fraction of the capacity available in a nest, but such an intuition does not emerge from the more general approximation scheme. Thus, while this more general approximation scheme is certainly useful, we defer its full development to the online supplement.

Comparing our results with the earlier work, Rusmevichientong, Shen and Shmoys (2010) show how to obtain the optimal assortment when customers choose according to the multinomial logit model and there are cardinality constraints on the offered assortment. They exploit the fact that the expected revenue under the multinomial logit model is a fraction of two linear functions. In contrast, the expected revenue under the nested logit model is a fraction of two nonlinear functions and we have to resort to a new argument to deal with the nonlinearity. Also, since the multinomial logit model is a special case of the nested logit model with a single nest, our results are naturally more general than their corresponding assortment optimization results. Rusmevichientong et al. (2009) consider assortment problems under the nested logit model with space constraints. The crucial difference between our space constraints and theirs is that the space constraints in Rusmevichientong et al. (2009) limit the total amount of space consumed by the products offered in all nests, whereas our space constraints separately limit the amount of space consumed by the products offered in each nest. Rusmevichientong et al. (2009) give an algorithm for assortment optimization under a space constraint covering all nests, but their algorithm does not scale well with the number of nests. If we want to obtain an assortment with a performance guarantee of two and there are m nests each including n products, then their algorithm has a running time of at least $O(m^{8m+1} n^{8m} \log^m(mn))$. In contrast, if we have a space constraint in each nest separately, then we can obtain an assortment with a performance guarantee of two by solving a linear program with $1 + m$ decision variables and $O(mn^2)$ constraints. Also, since the multinomial logit model is a special case of the nested logit model with a single nest, both our work and the work in Rusmevichientong et al. (2009) can be used to solve the assortment optimization problem under the multinomial logit model with a space constraint over all products. Rusmevichientong et al. (2009) need $O(n^8 \log n)$

operations to get a performance guarantee of two, whereas we need to solve a linear program with two decision variables and $O(n^2)$ constraints. Except for the case with a single nest, one should not directly compare the approaches in our paper and in Rusmevichientong et al. (2009), since while we impose a space constraint on the assortment offered in each nest separately, Rusmevichientong et al. (2009) impose a space constraint over all nests. Both type of constraints can have important applications and neither one of them can be seen as a special case of the other. It is also useful to note that imposing constraints in each nest separately, as we do in this paper, requires that the constraint structure imposed on the assortment is aligned with the nest structure of the choice model. Nevertheless, there are indeed settings where it is necessary to impose a constraint on the assortment offered in each nest separately. We describe several such settings in Section 1.

In addition to assortment optimization problems under cardinality and space constraints, we focus on a joint assortment optimization and pricing problem. We consider the setting where there are m nests, each including p products that we can offer to customers. Each product can be offered at one of b different price levels. The price of a product affects its attractiveness and customers choose according to the nested logit model. The goal is to find an assortment of products to offer and their corresponding prices so as to maximize the expected revenue per customer. Building on our results for constrained assortment optimization problems, we show that we can obtain an optimal solution to the joint assortment optimization and pricing problem by solving a linear program with $1 + m$ decision variables and $O(mp b^2)$ constraints. Compared with recent work on pricing, our result provides useful advantages. Li and Huh (2011) and Gallego and Wang (2011) study pricing problems under the nested logit model and show how to obtain optimal prices under the assumption that there is a specific parametric relationship between the attractiveness of a product and its price. Our result does not make use of a parametric relationship and works irrespective of how the attractiveness of a product depends on its price. Furthermore, we can explicitly restrict and bound the prices that can be chosen by the decision maker, since we work with a finite number of possible prices. For example, we can bound the prices within a desired range or ensure that prices are chosen in full increments of a dollar. As far as we are aware, incorporating such restrictions is not possible within the framework of Li and Huh (2011) and Gallego and Wang (2011).

Related Literature. We focus our literature review on papers that model customer choices through variants of the multinomial logit model and refer the reader to Kok et al. (2008), Farias et al. (2011) and Farias et al. (2012) for assortment optimization under other choice models. The multinomial logit model dates back to Luce (1959) and McFadden (1974). Since then, this model has been studied extensively and it is known to be compatible with random utility maximization, where each customer associates a random utility with each product and chooses the one providing the largest utility. However, a shortcoming of the multinomial logit model is that if a product is added to the offered assortment, then the multinomial logit model predicts that the market share of each product in the offered assortment decreases by the same relative amount. This phenomenon is referred to as the independence of irrelevant alternatives property and it naturally should not occur when different products cannibalize on each other to different extents; see Ben-Akiva and

Lerman (1994). One option to avoid the independence of irrelevant alternatives property is to use a mixture of multinomial logit models, where there are multiple customer types and customers of different types choose according to different multinomial logit models. Another option is to use the nested logit model. McFadden (1980), Borsch-Supan (1990), McFadden and Train (2000) and Train (2003) discuss that both of these options yield choice models that are consistent with random utility maximization, while avoiding the independence of irrelevant alternatives property.

In this paper, we use the nested logit model to describe customer preferences. This model has been used to characterize demand in numerous retail settings. Anderson and de Palma (1992) study the price competition among multiple firms by using the nested logit model, where each nest includes the products offered by a different firm. Richards (2007) builds on their model to analyze the pricing and discount decisions of supermarket retailers, comparing model predictions with empirical data. Bucklin and Gupta (1992) use the nested logit model to capture how customers make timing and brand choice decisions when they make purchases. They apply their work to predict laundry detergent purchase behavior. Ansari et al. (1995) and Guadagni and Little (1998) use the nested logit model to describe the coffee choice process of a customer and calibrate their models using sales data. Baltas et al. (1997) study the choice between manufacturer and retailer brands through the nested logit model. Bell and Lattin (1998) use the nested logit model to understand the choice between a store that offers low prices every day and a store that offers temporary deep discounts. Siriwardena et al. (2012) analyze the choice between environment friendly and standard cars through the nested logit model.

One stream of literature closely related to our work is on assortment optimization models that maximize the expected revenue per customer when customers choose according to a particular choice model. If customers choose according to the multinomial logit model and there are no constraints on the offered assortment, then Talluri and van Ryzin (2004) show that the optimal assortment includes a number of products with the largest revenues. In this case, the optimal solution can be obtained by adding products greedily into the offered assortment in the order of decreasing revenues. Bront et al. (2009) and Rusmevichientong, Shmoys and Topaloglu (2010) study assortment problems when customers choose according to a mixture of multinomial logit models. The former shows that the problem is NP-hard in the strong sense and investigates integer programming formulations and heuristics, whereas the latter shows that the problem is NP-hard in the weak sense and investigates approximation methods. Thus, assortment optimization under a mixture of multinomial logit models is computationally intractable even without constraints on the offered assortment. Mendez-Diaz et al. (2010) study valid cuts for the integer programming formulation of the assortment problem under a mixture of multinomial logit models and their work remains applicable when there are cardinality constraints. As described earlier, Rusmevichientong et al. (2009) study assortment problems under the nested logit model with a space constraint, whereas Rusmevichientong, Shen and Shmoys (2010) work on assortment problems under the multinomial logit model with a cardinality constraint. Davis et al. (2011) study assortment problems under the nested logit model without constraints on the offered assortment. They find that the

problem is tractable when the dissimilarity parameters of the nests, which characterize the degree of dissimilarity between the products in a nest, do not exceed one. Fortunately, this happens to be the case that is compatible with random utility maximization; see McFadden (1980).

Another related stream of work is on pricing problems that incorporate choice models. In the pricing setting, the decision maker sets the prices of the products and the prices of all products jointly determine the probability that a customer purchases a particular product. The goal is to maximize the expected revenue per customer. For the pricing problem, Hanson and Martin (1996) observe that the expected revenue function is not concave in prices under the multinomial logit model, but Song and Xue (2007) and Dong et al. (2009) solve the pricing problem by noting that the expected revenue function remains concave in the market shares of the products. Li and Huh (2011) extend the concavity result to the nested logit model under the assumption that the price sensitivities of the products are constant within each nest and the nest dissimilarity parameters are all less than one. Gallego and Wang (2011) relax both assumptions in Li and Huh (2011) and characterize the structure of the optimal prices.

Finally, revenue management models with customer choice is related to our work. Talluri and van Ryzin (2004) consider a revenue management model with a single flight leg. Customers choose among the fare classes available for purchase. The goal is to dynamically adjust the set of available fare classes so as to maximize the expected revenue. Gallego et al. (2004), Liu and van Ryzin (2008), Kunnumkal and Topaloglu (2008), Zhang and Adelman (2009), Bront et al. (2009), Talluri (2011) and Meissner et al. (2012) extend this model to a flight network. The idea in these papers is to develop a variety of linear programming approximations, where the decision variables correspond to the number of time periods during which a particular subset of itinerary fare class combinations is offered. Since there is one decision variable for every possible subset of itinerary fare class combinations, the number of decision variables is quite large. Thus, the linear programs are solved by using column generation. Such column generation subproblems exactly correspond to our assortment problem when customers choose according to the nested logit model.

Organization. In Section 1, we formulate the constrained assortment optimization problem under the nested logit model. In Section 2, we show how to use a linear program to find the best assortment to offer in each nest out of a given collection of candidate assortments. In Section 3, we give a general result that shows how to find a collection of candidate assortments for each nest such that if we restrict our attention only to these candidate assortments, then we obtain an assortment with a certain performance guarantee. In Sections 4 and 5, we leverage our result in Section 3 for cardinality and space constraints, respectively. In this way, we show how to obtain the optimal assortment under cardinality constraints and how to obtain an assortment with a performance guarantee under space constraints. In Section 6, we show how to obtain the optimal solution to the joint assortment optimization and pricing problem under the nested logit model. In Section 7, we give computational results. In Section 8, we provide concluding remarks, describe some possible extensions of our work and point out future research directions.

1 Problem Formulation

In this section, we describe the nested logit model and formulate the assortment optimization problem. There are m nests indexed by $M = \{1, \dots, m\}$. Depending on the application, each nest may correspond to a different category of products, a different sales channel or a different retail store. In each nest, there are n products that we can choose to offer to customers and we index the products by $N = \{1, \dots, n\}$. Under the nested logit model, an arriving customer decides either to make a purchase in one of the nests or to leave the system without purchasing anything. If the customer decides to make a purchase in one of the nests, then the customer must choose one of the products offered in this nest. We use the vector $S_i = (S_{i1}, \dots, S_{in}) \in \{0, 1\}^n$ to denote the assortment of products that we offer in nest i , where $S_{ij} = 1$ if we offer product j in nest i and $S_{ij} = 0$ otherwise. Throughout the paper, we refer to the vector $S_i \in \{0, 1\}^n$ as the assortment offered in nest i . We let v_{ij} be the preference weight of product j in nest i and use $V_i(S_i)$ to denote the total preference weight of all offered products in nest i when we offer the assortment S_i of products in this nest. More precisely, we have $V_i(S_i) = \sum_{j \in N} v_{ij} S_{ij}$. Under the nested logit model, given that a customer decides to make a purchase in nest i , if the assortment S_i is offered in this nest, then the probability that the customer chooses product j is $v_{ij} S_{ij} / V_i(S_i)$. From this expression, we observe that if product j in nest i is not included in the assortment S_i so that $S_{ij} = 0$, then the probability that the customer chooses product j is zero, as expected. We let r_{ij} be the revenue of product j in nest i . In this case, if we offer the assortment S_i in nest i and a customer decides to make a purchase in this nest, then the expected revenue that we obtain from this customer is given by

$$R_i(S_i) = \sum_{j \in N} \frac{v_{ij} S_{ij}}{V_i(S_i)} r_{ij} = \frac{\sum_{j \in N} v_{ij} r_{ij} S_{ij}}{V_i(S_i)}.$$

Letting $\bar{0} = (0, \dots, 0) \in \mathbb{R}_+^n$, if $S_i = \bar{0}$, then $R_i(S_i) = 0$ and we assume that $0/0 = 0$ in the expression above. Our notation thus far implies that the number of possible products is the same in each nest, but this assumption is only for notational brevity and all of our results in the paper continue to hold when each nest includes a different number of products.

The preference weight of not making a purchase is v_0 . Each nest i has a dissimilarity parameter γ_i associated with it, characterizing the degree of dissimilarity of the products that can be offered in this nest. In this case, if we offer the assortment (S_1, \dots, S_m) over all nests with $S_i \in \{0, 1\}^n$ for all $i \in M$, then the probability that a customer decides to make a purchase in nest i is given by

$$Q_i(S_1, \dots, S_m) = \frac{V_i(S_i)^{\gamma_i}}{v_0 + \sum_{l \in M} V_l(S_l)^{\gamma_l}}.$$

This expression determines the probability that a customer decides to make a purchase in a particular nest as a function of the assortment offered over all nests. The form of the choice probabilities above can be derived by using a random utility maximization principle; see McFadden (1974) and Train (2003). In particular, consider the case where each customer associates a random

utility U_{ij} with product j in nest i and a random utility U_0 with the option of not making a purchase. The customer, being a utility maximizer, follows the option providing the largest utility. In this case, it is possible to show that if the random utilities $\{U_{ij} : i \in M, j \in N\} \cup \{U_0\}$ have a certain multi-dimensional generalized extreme value distribution, then the probability of choosing a particular nest and the probability of choosing a particular product within a nest have precisely the same forms specified by the nested logit model above. The preference weights $\{v_{ij} : i \in M, j \in N\} \cup \{v_0\}$ characterize the means of the random utilities and a larger value of v_{ij} corresponds to a larger mean of U_{ij} . The random utilities have unit variance and the dissimilarity parameters $\{\gamma_i : i \in M\}$ characterize the correlation structure between the random utilities. To ensure that the dissimilarity parameters characterize a valid correlation structure, the nested logit model assumes that $\gamma_i \in (0, 1]$ for all $i \in M$, which is a sufficient condition to guarantee compatibility with utility maximizing behavior; see McFadden (1978).

If we offer the assortment (S_1, \dots, S_m) over all nests, then the expected revenue obtained from each customer can be written as

$$\Pi(S_1, \dots, S_m) = \sum_{i \in M} Q_i(S_1, \dots, S_m) R_i(S_i) = \frac{1}{v_0 + \sum_{l \in M} V_l(S_l)^{\gamma_l}} \sum_{i \in M} V_i(S_i)^{\gamma_i} R_i(S_i),$$

where the second equality uses the definition of $Q_i(S_1, \dots, S_m)$. In this paper, we consider the problem of finding an assortment (S_1, \dots, S_m) to offer over all nests so as to maximize the expected revenue $\Pi(S_1, \dots, S_m)$ per customer while making sure that the assortment offered in each nest satisfies a certain feasibility constraint. We work with two types of feasibility constraints on the offered assortment in each nest. The first type of constraints correspond to *cardinality constraints*, limiting the number of products in the assortment offered in each nest. So, the feasible assortments in nest i are given by $\mathcal{C}_i = \{S_i \in \{0, 1\}^n : \sum_{j \in N} S_{ij} \leq c_i\}$, where c_i is the maximum number of products that we can offer in nest i . The second type of constraints correspond to *space constraints*, where there is a space requirement for each product and each nest provides a limited amount of available space. The total amount of space consumed by the products offered in a nest cannot exceed the space availability of the nest. In this case, using w_{ij} to denote the space requirement of product j in nest i and c_i to denote the amount of space available in nest i , the feasible assortments in nest i are given by $\mathcal{C}_i = \{S_i \in \{0, 1\}^n : \sum_{j \in N} w_{ij} S_{ij} \leq c_i\}$. We assume that $w_{ij} \leq c_i$ for all $i \in M, j \in N$, implying that it is feasible to offer each product by itself. Our goal is to find an assortment over all nests so that we maximize the expected revenue obtained from each customer while making sure that the assortment offered in each nest satisfies the cardinality or space constraint, yielding the problem

$$Z^* = \max_{(S_1, \dots, S_m) \in \mathcal{C}_1 \times \dots \times \mathcal{C}_m} \Pi(S_1, \dots, S_m), \quad (1)$$

where the feasible assortments $(\mathcal{C}_1, \dots, \mathcal{C}_m)$ may correspond to cardinality or space constraints. We focus on finding a solution to problem (1) in a tractable fashion throughout the paper.

The cardinality or space constraints in problem (1) separately limit the cardinality or space consumption of the assortment offered in each nest. There are a variety of situations where it

may be necessary to impose such constraints. As an example of a case where space constraints arise, consider the situation where each nest corresponds to a different retail store. Each retail store provides a certain amount of display space to show the assortment of products offered to customers. The display space is used to show one floor sample for each offered product. We note that it is common to display one floor sample for each offered product when selling electronic equipment and household appliances, such as computers, stereos, refrigerators, dishwashers and air conditioners. The floor sample of each product, if offered, occupies a fixed amount of space, irrespective of what other products are offered. Due to limited display space, it may be necessary to limit the total amount of space consumed by the floor samples of the products offered in each retail store. Similarly, if each nest corresponds to a different product category, then it may be necessary to limit the cardinality or total space consumption of the floor samples of the products offered in each product category. For example, a product category may represent the television set models with a particular screen size. A retail store may display one floor sample for each offered television set model. To ensure that the product offering is not dominated by television set models with a particular screen size, the retail store may limit the cardinality or total space consumption of the floor samples of the offered television set models with each possible screen size.

As another example, consider the case where a company offers its products in multiple sales channels, such as an online channel and a conventional retail store. In this case, each nest may correspond a different sales channel and customers may be substituting between the products in the same sales channel as well as between the products in different sales channels. Each product, if offered, occupies a fixed amount of space in the online sales channel, which corresponds to the website real estate used for the offered product. Noting that customers are likely to pay attention to top results in online searches, there is essentially limited website real estate for displaying the products offered to customers. Thus, the necessity to separately impose constraints on the cardinality or space consumption of the assortment offered in each nest arises in numerous settings. In our examples, the application area naturally calls for imposing constraints on the assortment offered in each nest, rather than imposing one monolithic constraint spanning the assortment offered over all nests.

We observe that our formulation of the nested logit model assumes that the preference weights $\{v_{ij} : i \in M, j \in N\} \cup \{v_0\}$ and the dissimilarity parameters $\{\gamma_i : i \in M\}$ are ~~fixed parameters, not depending on the~~ offered assortment of products. This assumption originates from the first principles of the nested logit model. The nested logit model is based on the assumption that a customer associates a random utility with each product in each nest. The utility U_{ij} associated with product j in nest i is of the form $U_{ij} = \bar{u}_{ij} + \varepsilon_{ij}$, where \bar{u}_{ij} is the deterministic mean utility and ε_{ij} is the random idiosyncratic utility with mean zero. The idiosyncratic utilities $\{\varepsilon_{ij} : i \in M, j \in N\}$ have a multi-dimensional generalized extreme value distribution. The idiosyncratic utilities for the products in different nests are independent of each other, whereas the idiosyncratic utilities for the products in nest i have a fixed correlation coefficient of $1 - \gamma_i$. Once a customer associates random utilities with all of the products based on this mean and correlation structure, the customer chooses

the available product that provides the largest utility. The probability of choosing a particular product under this utility maximization interpretation has precisely the same form specified by the nested logit model, as long as we set the preference weight of product j in nest i as $v_{ij} = e^{\bar{u}_{ij}/\gamma_i}$. Under this setup, we impose a fixed correlation structure on the utilities of all products, which does not depend on what products are available. This approach has some similarity to what is done in other fields, such as portfolio theory, where one estimates the correlation structure between the returns of different assets and assumes that the correlation structure does not change based on what assets are included in the portfolio.

A behavioral way of justifying the constant preference weights and dissimilarity parameters is to assume that a customer associates a random utility with each product by comparing the features of the product against a set of desirable features, without paying attention to what products are or will eventually be available. For example, if customers evaluate the products by using consumer reports or product reviews, then it is reasonable for them to attach utilities with the products without paying attention to what products will be available when they visit the retail store. Through this evaluation process, customers attach utilities with all of the products that can potentially be offered, but once they visit the retail store and observe the offered assortment, they choose the available product that provides the largest utility. If the utilities of the products have the multi-dimensional generalized extreme value distribution described above, then this choice process precisely corresponds to the choice probabilities captured by our nested logit model.

Finally, our formulation of the nested logit model assumes that if a customer decides to make a purchase in one of the nests, then the customer must choose one of the products offered in this nest. We can relax this assumption to model the possibility that a customer may leave without purchasing anything even after choosing a particular nest. All of our results continue to hold under this relaxation. In particular, we can assume that there is a no preference weight v_{i0} associated with nest i . In this case, using $\mathbf{1}(\cdot)$ to denote the indicator function, if we offer the assortment $S_i \in \{0, 1\}^n$ in nest i , then the total preference weight of all offered options in nest i is given by $V_i(S_i) = v_{i0} \mathbf{1}(S_i \neq \bar{0}) + \sum_{j \in N} v_{ij} S_{ij}$. Given that a customer decides to make a purchase in nest i , if the assortment S_i is offered in this nest, then the probability that the customer chooses product j is $v_{ij} S_{ij} / V_i(S_i)$. Therefore, if $S_i \neq \bar{0}$, then the customer leaves nest i without making a purchase with probability $v_{i0} / V_i(S_i)$. Due to the indicator function in the definition of $V_i(S_i)$ in this paragraph, we have $V_i(\bar{0}) = 0$ so that customers are not attracted to nest i at all when the empty assortment is offered in this nest. We can verify that all of our results continue to hold when we allow customers to leave a nest without purchasing anything by defining $V_i(S_i) = v_{i0} \mathbf{1}(S_i \neq \bar{0}) + \sum_{j \in N} v_{ij} S_{ij}$. We come back this extension at appropriate places in the paper.

2 Combining Candidate Assortments

In this section, we answer a fundamental question that arises when constructing an algorithm to solve problem (1). Assume that we are given a collection of candidate assortments $\{A_i^t : t \in \mathcal{T}_i\}$ to

offer in nest i . All of the assortments in the collection are feasible in the sense that $A_i^t \in \mathcal{C}_i$ for all $t \in \mathcal{T}_i$ and the set of feasible assortments \mathcal{C}_i may correspond to cardinality or space constraints. For each nest i , we want to find an assortment $\hat{S}_i \in \{A_i^t : t \in \mathcal{T}_i\}$ such that the combined assortment $(\hat{S}_1, \dots, \hat{S}_m)$ provides the largest expected revenue among all assortments of the form (S_1, \dots, S_m) with $S_i \in \{A_i^t : t \in \mathcal{T}_i\}$. In other words, we want to use the collections of candidate assortments $\{A_i^t : t \in \mathcal{T}_i\}$ for all $i \in M$ to stitch together the best assortment to offer over all nests. Finding an answer to this question by brute force is computationally difficult since there are $|\mathcal{T}_1| \times \dots \times |\mathcal{T}_m|$ possible combinations of assortments to choose over all nests.

We can solve a linear program to obtain an assortment that provides the largest expected revenue among all assortments of the form (S_1, \dots, S_m) with $S_i \in \{A_i^t : t \in \mathcal{T}_i\}$. The number of decision variables and the number of constraints in this linear program increase linearly with the number of nests and the number of assortments in the candidate collection, rendering the linear program practical to use even when the number of nests and the number of candidate assortments are large. To formulate this linear program, we begin with the following lemma that shows how we can stitch together the best assortment to offer over all nests by using the collections of candidate assortments $\{A_i^t : t \in \mathcal{T}_i\}$ for all $i \in M$.

Lemma 1 *For all $i \in M$, assume that we have a collection of assortments $\{A_i^t : t \in \mathcal{T}_i\}$ with $A_i^t \in \mathcal{C}_i$ for all $t \in \mathcal{T}_i$. If we let \hat{z} be the value of z that satisfies*

$$v_0 z = \sum_{i \in M} \max_{S_i \in \{A_i^t : t \in \mathcal{T}_i\}} \left\{ V_i(S_i)^{\gamma_i} (R_i(S_i) - z) \right\}, \quad (2)$$

then \hat{z} corresponds to the largest expected revenue that can be obtained by using assortments of the form (S_1, \dots, S_m) with $S_i \in \{A_i^t : t \in \mathcal{T}_i\}$. Furthermore, if we let $\hat{S}_i, i \in M$ be an optimal solution to the problem

$$\max_{S_i \in \{A_i^t : t \in \mathcal{T}_i\}} \left\{ V_i(S_i)^{\gamma_i} (R_i(S_i) - \hat{z}) \right\}, \quad (3)$$

then the assortment $(\hat{S}_1, \dots, \hat{S}_m)$ provides the largest expected revenue among all assortments of the form (S_1, \dots, S_m) with $S_i \in \{A_i^t : t \in \mathcal{T}_i\}$.

Proof. Letting \hat{z} and \hat{S}_i be as defined in the lemma, it is enough to show that \hat{z} is at least as large as the expected revenue provided by any assortment of the form (S_1, \dots, S_m) with $S_i \in \{A_i^t : t \in \mathcal{T}_i\}$ and the expected revenue provided by the assortment is $(\hat{S}_1, \dots, \hat{S}_m)$ is \hat{z} . To show the second statement, since \hat{z} is the value of z satisfying (2) and \hat{S}_i is an optimal solution to problem (3), we get $v_0 \hat{z} = \sum_{i \in M} V_i(\hat{S}_i)^{\gamma_i} (R_i(\hat{S}_i) - \hat{z})$. Solving for \hat{z} in the last expression, we obtain $\hat{z} = \sum_{i \in M} V_i(\hat{S}_i)^{\gamma_i} R_i(\hat{S}_i) / (v_0 + \sum_{i \in M} V_i(\hat{S}_i)^{\gamma_i}) = \Pi(\hat{S}_1, \dots, \hat{S}_m)$, where the last equality follows from the definition of $\Pi(S_1, \dots, S_m)$. Thus, the expected revenue provided by the assortment $(\hat{S}_1, \dots, \hat{S}_m)$ is \hat{z} , establishing the second statement. To show the first statement, for any assortment $(\tilde{S}_1, \dots, \tilde{S}_m)$ with $\tilde{S}_i \in \{A_i^t : t \in \mathcal{T}_i\}$, \tilde{S}_i provides a feasible but not necessarily

an optimal solution to the maximization problem on the right side of (2), in which case, we obtain $v_0 \hat{z} \geq \sum_{i \in M} V_i(\tilde{S}_i)^{\gamma_i} (R_i(\tilde{S}_i) - \hat{z})$. Solving for \hat{z} in the last inequality and using the definition of $\Pi(S_1, \dots, S_m)$ once more, we get $\hat{z} \geq \Pi(\tilde{S}_1, \dots, \tilde{S}_m)$. Thus, for any assortment $(\tilde{S}_1, \dots, \tilde{S}_m)$ with $\tilde{S}_i \in \{A_i^t : t \in \mathcal{T}_i\}$, \hat{z} is at least as large as the expected revenue provided by this assortment. Since our choice of \tilde{S}_i is arbitrary, \hat{z} is at least as large as the expected revenue from any assortment of the form (S_1, \dots, S_m) with $S_i \in \{A_i^t : t \in \mathcal{T}_i\}$, establishing the first statement. \square

Thus, Lemma 1 shows that if we use \hat{z} to denote the value of z satisfying (2) and let \hat{S}_i be an optimal solution to problem (3), then the assortment $(\hat{S}_1, \dots, \hat{S}_m)$ is the best assortment among all assortments of the form (S_1, \dots, S_m) with $S_i \in \{A_i^t : t \in \mathcal{T}_i\}$. This result provides an answer to the question that we pose at the beginning of this section. One remaining question, however, is that how we can find a value of z satisfying (2). Noting that the left side of (2) is strictly increasing in z and the right side is decreasing in z , there always exists a unique value of z satisfying (2). To construct a simple approach for obtaining the value of z that satisfies (2), we observe that this value of z corresponds to the optimal objective value of the optimization problem $\min\{z : v_0 z \geq \sum_{i \in M} \max_{S_i \in \{A_i^t : t \in \mathcal{T}_i\}} \{V_i(S_i)^{\gamma_i} (R_i(S_i) - z)\}\}$. To linearize the constraint in this problem, we define the decision variables $y = (y_1, \dots, y_m)$ as $y_i = \max_{S_i \in \{A_i^t : t \in \mathcal{T}_i\}} \{V_i(S_i)^{\gamma_i} (R_i(S_i) - z)\}$ and write the last optimization problem as

$$\min \left\{ z : v_0 z \geq \sum_{i \in M} y_i, y_i \geq V_i(S_i)^{\gamma_i} (R_i(S_i) - z) \quad \forall S_i \in \{A_i^t : t \in \mathcal{T}_i\}, i \in M \right\}, \quad (4)$$

where the decision variables are (z, y) . The problem above is a linear program with $1 + m$ decision variables and $1 + \sum_{i \in M} |\mathcal{T}_i|$ constraints, which is tractable as long as the number of candidate assortments is not too large. The next theorem follows from the discussion in this paragraph.

Theorem 2 *Let \hat{z} be the optimal objective value of the linear program in (4) and \hat{S}_i be an optimal solution to problem (3). Then, the assortment $(\hat{S}_1, \dots, \hat{S}_m)$ provides the largest expected revenue among all assortments of the form (S_1, \dots, S_m) with $S_i \in \{A_i^t : t \in \mathcal{T}_i\}$.*

If we are given collections of candidate assortments $\{A_i^t : t \in \mathcal{T}_i\}$ for all $i \in M$, then Theorem 2 provides a systematic approach for stitching together the best assortment over all nests. All we need to do is to solve the linear program in (4) to obtain its optimal objective value. Using \hat{z} to denote the optimal objective value of this linear program and \hat{S}_i to denote the optimal solution to problem (3), the assortment $(\hat{S}_1, \dots, \hat{S}_m)$ is the best assortment we can obtain by using the collection of candidate assortments $\{A_i^t : t \in \mathcal{T}_i\}$ for nest i . In the next section, we consider the question of how we can come up with a good collection of candidate assortments. Closing this section, we observe that the proofs of Lemma 1 and Theorem 2 do not use a particular structure for the form of $V_i(S_i)$. Therefore, these results continue to hold when we allow a customer to leave a nest without purchasing anything by letting $V_i(S_i) = v_{i0} \mathbf{1}(S_i \neq \bar{0}) + \sum_{j \in N} v_{ij} S_{ij}$.

3 Obtaining Candidate Assortments

In this section, we show how to obtain a collection of candidate assortments for each nest so that the best assortment that we can stitch together by focusing on these candidate assortments provides a certain performance guarantee. In the next lemma, we give characterization of good assortments for each nest.

Lemma 3 *Let (S_1^*, \dots, S_m^*) be an optimal solution to problem (1) with the objective value Z^* and set $u_i^* = \max\{Z^*, \gamma_i Z^* + (1 - \gamma_i) R_i(S_i^*)\}$ for all $i \in M$. If the assortments \hat{S}_i , $i \in M$ satisfy*

$$\alpha V_i(\hat{S}_i) (R_i(\hat{S}_i) - u_i^*) \geq \max_{S_i \in \mathcal{C}_i} \left\{ V_i(S_i) (R_i(S_i) - u_i^*) \right\} \quad (5)$$

for some $\alpha \geq 1$, then $\alpha \Pi(\hat{S}_1, \dots, \hat{S}_m) \geq Z^*$.

Proof. For notational brevity, we let $\hat{V}_i = V_i(\hat{S}_i)$, $V_i^* = V_i(S_i^*)$, $\hat{R}_i = R_i(\hat{S}_i)$ and $R_i^* = R_i(S_i^*)$ throughout the proof. We claim that $\alpha \hat{V}_i^{\gamma_i} (\hat{R}_i - Z^*) \geq (V_i^*)^{\gamma_i} (R_i^* - Z^*)$ for all $i \in M$. We proceed to showing this claim by separately considering a nest i satisfying $R_i^* > Z^*$ and a nest i satisfying $R_i^* \leq Z^*$. First, consider a nest i satisfying $R_i^* > Z^*$. For this nest, we have $u_i^* = \gamma_i Z^* + (1 - \gamma_i) R_i^*$ by the definition of u_i^* . If we evaluate the objective function of the maximization problem on the right side of (5) with $S_i = S_i^*$, then since S_i^* is a feasible but not necessarily an optimal solution to this maximization problem, we get $\alpha \hat{V}_i (\hat{R}_i - \gamma_i Z^* - (1 - \gamma_i) R_i^*) \geq V_i^* (R_i^* - \gamma_i Z^* - (1 - \gamma_i) R_i^*)$. Arranging the terms in this inequality, it follows that

$$\alpha \hat{V}_i (\hat{R}_i - Z^*) \geq (\gamma_i V_i^* + \alpha (1 - \gamma_i) \hat{V}_i) (R_i^* - Z^*). \quad (6)$$

Since $\gamma_i \in (0, 1]$, u^{γ_i} is a concave function of u and it satisfies the subgradient inequality $u^{\gamma_i} \leq \hat{u}^{\gamma_i} + \gamma_i \hat{u}^{\gamma_i - 1} (u - \hat{u}) = \hat{u}^{\gamma_i - 1} (\gamma_i u + (1 - \gamma_i) \hat{u}) \leq \hat{u}^{\gamma_i - 1} (\gamma_i u + \alpha (1 - \gamma_i) \hat{u})$ for all $u, \hat{u} \in \mathfrak{R}_+$, where the last inequality uses the fact that $\alpha \geq 1$ and $\gamma_i \leq 1$. Using this subgradient inequality with $u = V_i^*$ and $\hat{u} = \hat{V}_i$, we get $(V_i^*)^{\gamma_i} \leq \hat{V}_i^{\gamma_i - 1} (\gamma_i V_i^* + \alpha (1 - \gamma_i) \hat{V}_i)$. Since $R_i(S_i^*) = R_i^* > Z^* \geq 0$, we have $S_i^* \neq \bar{0}$, showing that $V_i^* > 0$, in which case, (6) implies that $\hat{V}_i > 0$. In this case, if we multiply both sides of (6) by $\hat{V}_i^{\gamma_i - 1}$, then we obtain

$$\alpha \hat{V}_i^{\gamma_i} (\hat{R}_i - Z^*) \geq \hat{V}_i^{\gamma_i - 1} (\gamma_i V_i^* + \alpha (1 - \gamma_i) \hat{V}_i) (R_i^* - Z^*),$$

but noting that $(V_i^*)^{\gamma_i} \leq \hat{V}_i^{\gamma_i - 1} (\gamma_i V_i^* + \alpha (1 - \gamma_i) \hat{V}_i)$ and $R_i^* > Z^*$, the inequality above yields $\alpha \hat{V}_i^{\gamma_i} (\hat{R}_i - Z^*) \geq (V_i^*)^{\gamma_i} (R_i^* - Z^*)$. Thus, our claim holds for all $i \in M$ satisfying $R_i^* > Z^*$. Second, consider a nest i satisfying $R_i^* \leq Z^*$. For this nest, we have $u_i^* = Z^*$ by the definition of u_i^* . If we evaluate the objective function of the maximization problem on the right side of (5) with $S_i = \bar{0}$, then since $\bar{0}$ is a feasible but not necessarily an optimal solution to this maximization problem and $V_i(\bar{0}) = 0$, we obtain $\alpha \hat{V}_i (\hat{R}_i - Z^*) \geq V_i(\bar{0}) (R_i(\bar{0}) - Z^*) = 0$, indicating that $\alpha \hat{V}_i (\hat{R}_i - Z^*) \geq 0$. The last inequality implies that either $\hat{V}_i = 0$, or $\hat{V}_i > 0$ and $\hat{R}_i - Z^* \geq 0$. In either case, since $R_i^* \leq Z^*$,

we obtain $\alpha \hat{V}_i^{\gamma_i} (\hat{R}_i - Z^*) \geq 0 \geq (V_i^*)^{\gamma_i} (R_i^* - Z^*)$, which establishes our claim for all $i \in M$ satisfying $R_i^* \leq Z^*$. Thus, our claim holds for all of the nests

By the claim established above, we have $\alpha V_i(\hat{S}_i)^{\gamma_i} (R_i(\hat{S}_i) - Z^*) \geq V_i(S_i^*)^{\gamma_i} (R_i(S_i^*) - Z^*)$ for all $i \in M$. Adding this inequality over all $i \in M$, we obtain $\sum_{i \in M} V_i(S_i^*)^{\gamma_i} (R_i(S_i^*) - Z^*) \leq \sum_{i \in M} \alpha V_i(\hat{S}_i)^{\gamma_i} (R_i(\hat{S}_i) - Z^*) \leq \sum_{i \in M} V_i(\hat{S}_i)^{\gamma_i} (\alpha R_i(\hat{S}_i) - Z^*)$, where the second inequality uses the fact that $\alpha \geq 1$ and $Z^* \geq 0$. On the other hand, since (S_1^*, \dots, S_m^*) is an optimal solution to problem (1) with the objective value Z^* , we have

$$Z^* = \Pi(S_1^*, \dots, S_m^*) = \frac{\sum_{i \in M} V_i(S_i^*)^{\gamma_i} R_i(S_i^*)}{v_0 + \sum_{i \in M} V_i(S_i^*)^{\gamma_i}}.$$

Focusing on the first and last terms in the chain of equalities above and arranging the terms, we obtain $v_0 Z^* = \sum_{i \in M} V_i(S_i^*)^{\gamma_i} (R_i(S_i^*) - Z^*)$. If we use this equality in the last chain of inequalities in this paragraph, then it follows that $v_0 Z^* \leq \sum_{i \in M} V_i(\hat{S}_i)^{\gamma_i} (\alpha R_i(\hat{S}_i) - Z^*)$ and solving for Z^* in this inequality yields $Z^* \leq \alpha \sum_{i \in M} V_i(\hat{S}_i)^{\gamma_i} R_i(\hat{S}_i) / (v_0 + \sum_{i \in M} V_i(\hat{S}_i)^{\gamma_i}) = \alpha \Pi(\hat{S}_1, \dots, \hat{S}_m)$, establishing the desired result. \square

Lemma 3 implies that if we use \hat{S}_i^α to denote an α -approximate solution to the maximization problem on the right side of (5), then the assortment $(\hat{S}_1^\alpha, \dots, \hat{S}_m^\alpha)$ is an α -approximate solution to problem (1). On the surface, this lemma is not immediately useful for obtaining good solutions to problem (1) since obtaining an α -approximate solution to the maximization problem on the right side of (5) requires knowing u_i^* , which, in turn, requires knowing Z^* and S_i^* , both of which are unknown to us without knowing the optimal solution to problem (1). To resolve this difficulty, the important observation is that the quantity u_i^* in the maximization problem on the right side of (5) is a constant. To exploit this observation, for any $u \in \mathfrak{R}_+$, we use $\hat{S}_i^\alpha(u)$ to denote an α -approximate solution to the problem

$$\max_{S_i \in \mathcal{C}_i} \left\{ V_i(S_i) (R_i(S_i) - u) \right\}. \quad (7)$$

In this case, we note that the collection of assortments $\{\hat{S}_i^\alpha(u) : u \in \mathfrak{R}_+\}$ includes an α -approximate solution to the maximization problem on the right side of (5), since problem (7) with $u = u_i^*$ is precisely the same as the maximization problem on the right side of (5). Therefore, it follows that the collection of assortments $\{\hat{S}_i^\alpha(u) : u \in \mathfrak{R}_+\}$ includes an α -approximate solution to the maximization problem on the right side of (5). This observation, in view of Lemma 3, implies that if we use $\{\hat{S}_i^\alpha(u) : u \in \mathfrak{R}_+\}$ as the collection of candidate assortments for each nest i , then the best assortment that we can stitch together by focusing only on these collections of candidate assortments provides an expected revenue that deviates from the optimal expected revenue by no more than a factor of α .

To summarize the discussion in the paragraph above, assume that we can come up with a collection of assortments $\{A_i^t : t \in \mathcal{T}_i\}$ such that this collection always includes an α -approximate solution to problem (7) for any $u \in \mathfrak{R}_+$. In other words, $\{A_i^t : t \in \mathcal{T}_i\} \supset \{\hat{S}_i^\alpha(u) : u \in \mathfrak{R}_+\}$. In this

case, the best assortment of the form $(\hat{S}_1, \dots, \hat{S}_m)$ with $\hat{S}_i \in \{A_i^t : t \in \mathcal{T}_i\}$ provides an expected revenue that deviates from the optimal expected revenue for problem (1) by no more than a factor of α . We record this observation in the next theorem.

Theorem 4 *Assume that the collection of assortments $\{A_i^t : t \in \mathcal{T}_i\}$ includes an α -approximate solution to problem (7) for any $u \in \mathfrak{R}_+$. Then, there exists an assortment $(\hat{S}_1, \dots, \hat{S}_m)$ with $\hat{S}_i \in \{A_i^t : t \in \mathcal{T}_i\}$ such that $\alpha \Pi(\hat{S}_1, \dots, \hat{S}_m) \geq Z^*$.*

Theorems 2 and 4 play a key role to construct algorithms for solving problem (1). In particular, if we can come up with a reasonably small collection of assortments $\{A_i^t : t \in \mathcal{T}_i\}$ such that this collection always includes an α -approximate solution to problem (7) for any $u \in \mathfrak{R}_+$, then Theorem 4 implies that the expected revenue from the best assortment $(\hat{S}_1, \dots, \hat{S}_m)$ with $\hat{S}_i \in \{A_i^t : t \in \mathcal{T}_i\}$ deviates from the optimal expected revenue by at most a factor of α . On the other hand, Theorem 2 implies that the best assortment $(\hat{S}_1, \dots, \hat{S}_m)$ with $\hat{S}_i \in \{A_i^t : t \in \mathcal{T}_i\}$ can be obtained by solving a linear program with $1 + m$ decision variables and $1 + \sum_{i \in M} |\mathcal{T}_i|$ constraints. These observations show that if, for some $\alpha \geq 1$, we can come up with a reasonably small collection of assortments that includes an α -approximate solution to problem (7) for any $u \in \mathfrak{R}_+$, then we can solve a linear program with small numbers of decision variables and constraints to obtain an assortment whose expected revenue deviates from the optimal expected revenue by at most a factor of α . This discussion essentially reduces the job of obtaining good solutions to problem (1) to the job of obtaining good solutions to problem (7) for any $u \in \mathfrak{R}_+$.

In general, it may not be possible to come up with a reasonably small collection of assortments $\{A_i^t : t \in \mathcal{T}_i\}$ that includes an α -approximate solution to problem (7) for any $u \in \mathfrak{R}_+$. However, the useful feature of problem (7) is that its objective function is given by

$$V_i(S_i) (R_i(S_i) - u) = V_i(S_i) \left[\frac{\sum_{j \in N} v_{ij} r_{ij} S_{ij}}{V_i(S_i)} - u \right] = \sum_{j \in N} v_{ij} (r_{ij} - u) S_{ij}, \quad (8)$$

where the last equality follows from the fact that $V_i(S_i) = \sum_{j \in N} v_{ij} S_{ij}$. Therefore, the objective function of problem (7) is linear in S_i . In the next two sections, we exploit this linearity to show that if the feasible assortments are given by cardinality or space constraints, then we can indeed come up with a reasonably small collection of assortments that includes an α -approximate solution to problem (7) for any $u \in \mathfrak{R}_+$. Closing this section, we emphasize that as far as the particular form of $V_i(S_i)$ is concerned, the proof of Lemma 3 only uses the fact that $V_i(\bar{0}) = 0$, which continues to hold when we define $V_i(S_i)$ as $V_i(S_i) = v_{i0} \mathbf{1}(S_i \neq \bar{0}) + \sum_{j \in N} v_{ij} S_{ij}$. Thus, this lemma holds even when we allow a customer to leave a nest without purchasing anything by letting $V_i(S_i) = v_{i0} \mathbf{1}(S_i \neq \bar{0}) + \sum_{j \in N} v_{ij} S_{ij}$. Once Lemma 3 continues to hold, Theorem 4 immediately follows from this lemma. Our results in the rest of the paper build on Theorems 2 and 4. Since these theorems continue to hold when customers are allowed to leave a nest without making a purchase, our subsequent results continue to hold under this extension as well.

4 Cardinality Constraints

In this section, we consider the case where the feasible assortments in nest i are captured by $\mathcal{C}_i = \{S_i \in \{0, 1\}^n : \sum_{j \in N} S_{ij} \leq c_i\}$, which corresponds to constraining the cardinality of the assortment offered in nest i to c_i . For this case, we show that we can come up with a collection of assortments $\{A_i^t : t \in \mathcal{T}_i\}$ such that this collection always includes an optimal, or 1-approximate, solution to problem (7) for any $u \in \mathfrak{R}_+$. Furthermore, we show that the collection $\{A_i^t : t \in \mathcal{T}_i\}$ includes $O(n^2)$ assortments and each one of the assortments in this collection can be identified in a tractable fashion. These results, in view of the discussion that follows Theorem 4, imply that we can find the optimal solution to problem (1) under cardinality constraints simply by solving a linear program with $1 + m$ decision variables and $O(mn^2)$ constraints.

To characterize the optimal solution to problem (7) for any $u \in \mathfrak{R}_+$, we note that the objective function of problem (7) can be written as in (8). In this case, we can use the decision variables $x_i = (x_{i1}, \dots, x_{in}) \in \{0, 1\}^n$ to write problem (7) under cardinality constraints as

$$\max \left\{ \sum_{j \in N} v_{ij} (r_{ij} - u) x_{ij} : \sum_{j \in N} x_{ij} \leq c_i, x_{ij} \in \{0, 1\} \forall j \in N \right\}. \quad (9)$$

Problem (9) is a knapsack problem where the utility of product j is $v_{ij} (r_{ij} - u)$ and each product consumes one unit of space. We can obtain an optimal solution to this knapsack problem by ordering the products with respect to their utilities and filling the knapsack starting from the product with the largest utility, as long as the utility of the product exceeds zero. Therefore, the optimal solution to problem (9) depends only on the ordering and signs of the utilities of the products. To exploit this observation, we define the linear functions $f_{ij}(u) = v_{ij} (r_{ij} - u)$ for $j \in N$ and $f_{i0}(u) = 0$, in which case, the function $f_{ij}(u)$ corresponds to the utility of product j in the knapsack problem above. The $n + 1$ linear functions $\{f_{ij}(\cdot) : j \in N \cup \{0\}\}$ intersect at $O(n^2)$ points, which can be obtained by solving the equation $f_{ij}(u) = f_{ik}(u)$ for u for all distinct $j, k \in N \cup \{0\}$. We use $\{\bar{u}_i^t : t \in \mathcal{T}_i\}$ with $|\mathcal{T}_i| = O(n^2)$ to denote the set of intersection points obtained in this fashion. Since we are interested in the optimal solution to problem (9) for $u \in \mathfrak{R}_+$, we add the value zero into the set $\{\bar{u}_i^t : t \in \mathcal{T}_i\}$. In this case, the points in the set $\{\bar{u}_i^t : t \in \mathcal{T}_i\}$ partition the positive real line into $O(|\mathcal{T}_i|)$ intervals. We denote these intervals by $\{\mathcal{I}_i^t : t \in \mathcal{T}_i\}$. In Figure 1, we show the linear functions $\{f_{ij}(\cdot) : j \in N \cup \{0\}\}$, the points $\{\bar{u}_i^t : t \in \mathcal{T}_i\}$ and the intervals $\{\mathcal{I}_i^t : t \in \mathcal{T}_i\}$ for a possible case with $N = \{1, 2, 3\}$. The solid lines show the functions $\{f_{ij}(\cdot) : j \in N \cup \{0\}\}$, the circles on the horizontal axis show the points $\{\bar{u}_i^t : t \in \mathcal{T}_i\}$ and the braces on the horizontal axis show the intervals $\{\mathcal{I}_i^t : t \in \mathcal{T}_i\}$. We observe that the ordering of the values of the linear functions $\{f_{ij}(u) : j \in N \cup \{0\}\}$ does not change as long as u takes values in one of the intervals $\{\mathcal{I}_i^t : t \in \mathcal{T}_i\}$. For example, we have $f_{i2}(u) \geq f_{i3}(u) \geq f_{i0}(u) \geq f_{i1}(u)$ for all $u \in \mathcal{I}_i^d$ and this ordering of $\{f_{ij}(u) : j \in N \cup \{0\}\}$ does not change as long as u is in \mathcal{I}_i^d .

So, the key observation is that the ordering and signs of the utilities of the products in the knapsack problem in (9) do not change as long as u takes values in one of the intervals

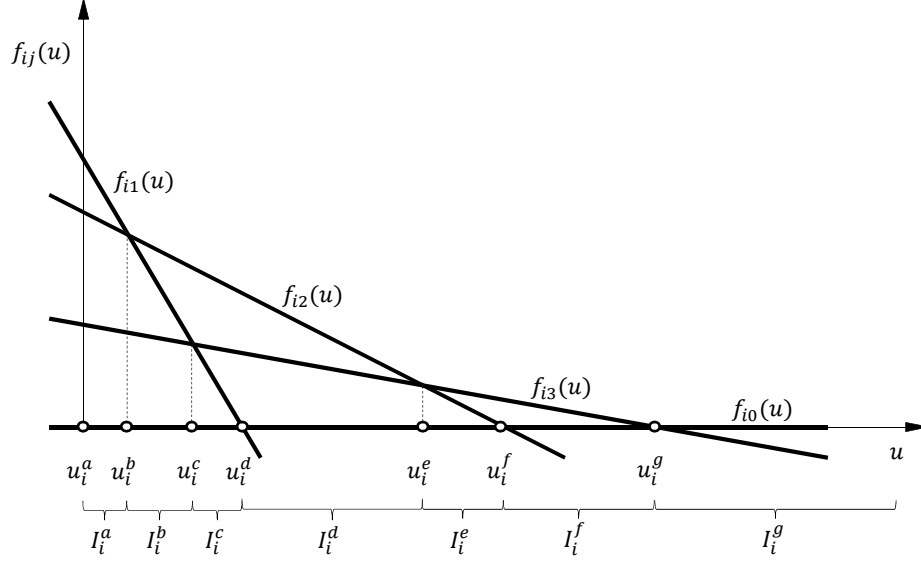


Figure 1: The linear functions $\{f_{ij}(\cdot) : j \in N \cup \{0\}\}$, the points $\{\bar{u}_i^t : t \in \mathcal{T}_i\}$ and the intervals $\{\mathcal{I}_i^t : t \in \mathcal{T}_i\}$ for a possible case with $N = \{1, 2, 3\}$.

$\{\mathcal{I}_i^t : t \in \mathcal{T}_i\}$. Since the optimal solution to the knapsack problem depends only on the ordering and signs of the utilities of the products, the optimal solution does not change either as long as u takes values in one of these intervals. Therefore, by checking the ordering and signs of the utilities of the products in each one of the intervals $\{\mathcal{I}_i^t : t \in \mathcal{T}_i\}$, we can come up with the optimal solution to the knapsack problem for any $u \in \mathfrak{R}_+$. Since there are $O(n^2)$ intervals in $\{\mathcal{I}_i^t : t \in \mathcal{T}_i\}$, we can come up with a collection of assortments $\{A_i^t : t \in \mathcal{T}_i\}$ with $|\mathcal{T}_i| = O(n^2)$ such that this collection always includes an optimal solution to problem (7) for any $u \in \mathfrak{R}_+$. The next theorem collects our observations in this section.

Theorem 5 *Under cardinality constraints, there exists a collection of assortments $\{A_i^t : t \in \mathcal{T}_i\}$ with $|\mathcal{T}_i| = O(n^2)$ that includes an optimal solution to problem (7) for any $u \in \mathfrak{R}_+$.*

In this case, by Theorem 4, the best assortment $(\hat{S}_1, \dots, \hat{S}_m)$ with $\hat{S}_i \in \{A_i^t : t \in \mathcal{T}_i\}$ is an optimal solution to problem (1). By Theorem 2, on the other hand, we can find the best assortment $(\hat{S}_1, \dots, \hat{S}_m)$ with $\hat{S}_i \in \{A_i^t : t \in \mathcal{T}_i\}$ by solving a linear program with $1 + m$ decision variables and $O(mn^2)$ constraints.

It is useful to make some observations regarding the structure of the optimal assortment to offer in each nest. The discussion in this section shows that if we solve problem (9) for all $u \in \mathfrak{R}_+$, then one of these solutions for some $u \in \mathfrak{R}_+$ corresponds to the optimal assortment to offer in nest i . Since problem (9) is a knapsack problem with each product consuming one unit of space, we include the products with the largest objective function coefficients in the optimal solution to problem (9). Therefore, when deciding which products to offer in nest i , the products should be prioritized according to the coefficients $\{v_{ij}(r_{ij} - u) : j \in N\}$ for some $u \in \mathfrak{R}_+$ and the value of u is common to all products in this nest. Intuitively, we can visualize u as the imputed

revenue adjustment for offering a product in nest i . In this case, we prioritize the products in the nest according to the product of their preference weights and their revenues, after modifying the revenues by the imputed adjustment. Products with larger preference weights and larger adjusted revenues get higher priority when choosing the products to include in the optimal assortment.

Interestingly, we can generate counterexamples to show that the optimal expected revenue is not necessarily a concave function of the capacities in the nests. Therefore, additional units of capacity may not yield decreasing marginal returns. To understand why this situation occurs, consider a problem instance with two nests, $\gamma_1 = \gamma_2 = 0.37$ and $v_0 = 0.85$. The first nest has one product with $r_{11} = 9$ and $v_{11} = 0.06$, whereas the second nest has three products with $(r_{21}, r_{22}, r_{23}) = (9, 7, 6.5)$ and $(v_{21}, v_{22}, v_{23}) = (0.75, 2.3, 10)$. So, the first products in both nests have the largest revenues, but the second and third products in the second nest have relatively small revenues. The capacity available in the first nest is $c_1 = 1$. We use Π^{c_2} to denote the optimal expected revenue over all nests as a function of the capacity c_2 available in the second nest. For this problem instance, we can check that $\Pi^1 = \Pi^2 \approx 5.36$ and $\Pi^3 \approx 5.43$. So, we get $\Pi^2 - \Pi^1 \leq \Pi^3 - \Pi^2$, indicating that we do not have decreasing marginal returns from additional units of capacity in the second nest. Furthermore, letting $S_2^{c_2} = (S_{21}^{c_2}, S_{22}^{c_2}, S_{23}^{c_2})$ be the optimal assortment in the second nest as a function of c_2 , we can verify that $S_2^1 = S_2^2 = (1, 0, 0)$ and $S_2^3 = (1, 1, 1)$.

The crucial tradeoff is between diluting the expected revenue from the second nest and decreasing the no purchase probability by adding products into the second nest. In particular, we observe that it is always optimal to offer the product with the largest revenue in the second nest, irrespective of the capacity available in this nest. If we have two units of capacity in the second nest, then it is not optimal to add any of the products with small revenues to the offered assortment. If we add any of these products, then a customer choosing the second nest may end up purchasing the product with small revenue and this possibility degrades the overall expected revenue. In other words, offering a product with small revenue dilutes the expected revenue from the second nest too much. However, if we have three units of capacity in the second nest, then it is optimal to add both of the products with small revenues to the offered assortment. By offering both of the products with small revenues, we increase the total preference weight of the products in the second nest, which, in turn, decreases the probability that a customer leaves without purchasing anything. Offering the two products with small revenues still dilutes the expected revenue from the assortment in the second nest, but we decrease the no purchase probability to such an extent that it becomes worthwhile to dilute the expected revenue from the second nest.

Rusmevichientong, Shen and Shmoys (2010) solve assortment problems under the multinomial logit model with a cardinality constraint by checking the ordering between $n + 1$ linear functions and by constructing a number of candidate assortments. They show that if the number of products in the offered assortment is limited to C and there are n available products, then they need $O(nC)$ candidate assortments. Following their argument, we can refine the discussion in this section to show that we need $O(nc_i)$ candidate assortments in nest i , rather than $O(n^2)$.

5 Space Constraints

In this section, we consider the case where product j in nest i consumes w_{ij} units of space and the space consumption of the assortment offered in nest i is limited to c_i . Therefore, the feasible assortments in nest i are $\mathcal{C}_i = \{S_i \in \{0, 1\}^n : \sum_{j \in N} w_{ij} S_{ij} \leq c_i\}$. Lemma 2.1 in Rusmevichientong et al. (2009) implies that problem (1) is NP-hard under such space constraints even when there is a single nest. Under space constraints, we show that we can come up with a collection of assortments $\{A_i^t : t \in \mathcal{T}_i\}$ that includes a 2-approximate solution to problem (7) for any $u \in \mathfrak{R}_+$. Furthermore, this collection includes $O(n^2)$ assortments. These results, together with the discussion that follows Theorem 4, imply that we can solve a linear program with $1 + m$ decision variables and $O(mn^2)$ constraints to obtain an assortment whose expected revenue deviates from the optimal expected revenue for problem (1) by at most a factor of two, when there are space constraints.

5.1 Performance Guarantee of Two

In this section, our goal is to come up with a collection of assortments $\{A_i^t : t \in \mathcal{T}_i\}$ that includes a 2-approximate solution to problem (7) for any $u \in \mathfrak{R}_+$. The line of reasoning we follow in this section is similar to the one in Section 4, but we work with the linear programming relaxation of a knapsack problem. Noting (8) and using the decision variables $x_i = (x_{i1}, \dots, x_{in}) \in \{0, 1\}^n$, we write problem (7) under space constraints as

$$\max \left\{ \sum_{j \in N} v_{ij} (r_{ij} - u) x_{ij} : \sum_{j \in N} w_{ij} x_{ij} \leq c_i, x_{ij} \in \{0, 1\} \forall j \in N \right\}. \quad (10)$$

The problem above is a knapsack problem where the utility of product j is $v_{ij}(r_{ij} - u)$ and the space consumption of product j is w_{ij} . We can solve the linear programming relaxation of this knapsack problem by ordering the products with respect to their utility to space consumption ratios and filling the knapsack starting from the product with the largest utility to space consumption ratio, as long as the utility of the product exceeds zero. Therefore, the optimal solution to the linear programming relaxation of the knapsack problem above depends only the ordering and signs of the utility to space consumption ratios of the products. Also, it is useful to observe that the optimal solution to the linear programming relaxation obtained in this fashion includes at most one fractional decision variable.

Following an argument similar to the one in Section 4, we define the linear functions $f_{ij}(u) = v_{ij}(r_{ij} - u)/w_{ij}$ for $j \in N$ and $f_{i0}(u) = 0$, in which case, $f_{ij}(u)$ corresponds to the utility to space consumption ratio of product j in the knapsack problem above. The $n + 1$ linear functions $\{f_{ij}(\cdot) : j \in N \cup \{0\}\}$ intersect at $O(n^2)$ points, which can be obtained by solving $f_{ij}(u) = f_{ik}(u)$ for u for all distinct $j, k \in N$. We use $\{\bar{u}_i^g : g \in \mathcal{G}_i\}$ with $|\mathcal{G}_i| = O(n^2)$ to denote the set of intersection points of the $n + 1$ linear functions $\{f_{ij}(\cdot) : j \in N \cup \{0\}\}$ obtained in this fashion. The points $\{\bar{u}_i^g : g \in \mathcal{G}_i\}$ partition the positive real line into $O(|\mathcal{G}_i|)$ intervals. Denoting these intervals by $\{\mathcal{I}_i^g : g \in \mathcal{G}_i\}$, the crucial observation is that the ordering of the values of the linear functions

$\{f_{ij}(u) : j \in N \cup \{0\}\}$ does not change as long as u takes values in one of these intervals. Thus, the ordering and signs of the utility to space consumption ratios of the products in problem (10) do not change as long as u takes values in one of the intervals $\{\mathcal{I}_i^g : g \in \mathcal{G}_i\}$. Noting that the optimal solution to the linear programming relaxation of problem (10) depends only on the ordering and signs of the utility to space consumption ratios of the products, the optimal solution to the linear programming relaxation of the knapsack problem above does not change either when u takes values in one of the intervals $\{\mathcal{I}_i^g : g \in \mathcal{G}_i\}$. We use $x_i^g = (x_{i1}^g, \dots, x_{in}^g) \in [0, 1]^n$ to denote the optimal solution to the linear programming relaxation of the knapsack problem in (10) when u takes values in the interval \mathcal{I}_i^g .

Using the solutions $\{x_i^g : g \in \mathcal{G}_i\}$, we define the collection of assortments $\{S_i^g : g \in \mathcal{G}_i\}$ such that each assortment $S_i^g = (S_{i1}^g, \dots, S_{in}^g) \in \{0, 1\}^n$ in this collection is obtained by setting $S_{ij}^g = \lfloor x_{ij}^g \rfloor$ for all $j \in N$, where $\lfloor \cdot \rfloor$ is the round down function. That is, the assortment S_i^g includes the products that take value one in the solution x_i^g . Augmenting the collection $\{S_i^g : g \in \mathcal{G}_i\}$ with the singleton assortments $\{\{j\} : j \in N\}$, we can show that the collection of assortments $\{S_i^g : g \in \mathcal{G}_i\} \cup \{\{j\} : j \in N\}$ always includes a 2-approximate solution to the knapsack problem in (10) for any $u \in \mathfrak{R}_+$. To see this result, assume that we solve the knapsack problem for some $u \in \mathfrak{R}_+$. By the discussion in the paragraph above, if the point u lies in the interval \mathcal{I}_i^g , then the optimal solution to the linear programming relaxation of the knapsack problem is given by x_i^g . In this case, using $z^*(u)$ to denote the optimal objective value of the knapsack problem above and j^g to denote the fractional component of the solution x_i^g when there is one, we obtain

$$\begin{aligned} z^*(u) &\leq \sum_{j \in N} v_{ij} (r_{ij} - u) x_{ij}^g \leq \sum_{j \in N} v_{ij} (r_{ij} - u) S_{ij}^g + v_{ij^g} (r_{ij^g} - u) \\ &\leq 2 \max \left\{ \sum_{j \in N} v_{ij} (r_{ij} - u) S_{ij}^g, v_{ij^g} (r_{ij^g} - u) \right\}, \quad (11) \end{aligned}$$

where the first inequality follows from the fact that the linear programming relaxation provides an upper bound on the optimal objective value of the knapsack problem and the second inequality follows from the fact that the products in the assortment S_i^g and the product j^g collectively include all components of the solution x_i^g that take strictly positive values. The chain of inequalities above imply that either one of the assortments S_i^g and $\{j^g\}$ is a 2-approximate solution to the knapsack problem in (10) and the desired result follows. Thus, the intervals $\{\mathcal{I}_i^g : g \in \mathcal{G}_i\}$ can be constructed by computing the intersection points of $n + 1$ linear functions. By checking the ordering and signs of the utility to space consumption ratios of the products in each one of the intervals $\{\mathcal{I}_i^g : g \in \mathcal{G}_i\}$, we can come up with all possible solutions $\{x_i^g : g \in \mathcal{G}_i\}$ to the linear programming relaxation of the knapsack problem in (10) for any $u \in \mathfrak{R}_+$. Using these solutions, we can construct the assortments $\{S_i^g : g \in \mathcal{G}_i\}$. In this case, if we use $\{A_i^t : t \in \mathcal{T}_i\}$ to denote the collection of assortments $\{S_i^g : g \in \mathcal{G}_i\} \cup \{\{j\} : j \in N\}$, then this collection of assortments satisfies $|\mathcal{T}_i| = O(|\mathcal{G}_i|) = O(n^2)$ and the chain of inequalities in (11) shows that this collection always includes a 2-approximate solution to problem (7) for any $u \in \mathfrak{R}_+$. We record this result in the next theorem.

Theorem 6 *Under space constraints, there exists a collection of assortments $\{A_i^t : t \in \mathcal{T}_i\}$ with $|\mathcal{T}_i| = O(n^2)$ that includes a 2-approximate solution to problem (7) for any $u \in \mathfrak{R}_+$.*

Therefore, by Theorem 4, the expected revenue from the best assortment $(\hat{S}_1, \dots, \hat{S}_m)$ with $\hat{S}_i \in \{A_i^t : t \in \mathcal{T}_i\}$ deviates from the optimal expected revenue by at most a factor of two. By Theorem 2, we can find the best assortment $(\hat{S}_1, \dots, \hat{S}_m)$ with $\hat{S}_i \in \{A_i^t : t \in \mathcal{T}_i\}$ by solving a linear program with $1 + m$ decision variables and $O(mn^2)$ constraints.

5.2 Refining the Performance Guarantee of Two

We can refine the performance guarantee of two by making use of the problem data. In particular, we assume that each product consumes a small fraction of the space available in its nest in the sense that $w_{ij} \leq \epsilon c_i$ for some $\epsilon \in [0, 1)$. In this case, it is possible to show that the collection of assortments $\{A_i^t : t \in \mathcal{T}_i\}$ constructed by using the approach in Section 5.1 always includes a $1/(1 - \epsilon)$ -approximate solution to problem (7) for any $u \in \mathfrak{R}_+$. Noting that $1/(1 - \epsilon) \rightarrow 1$ as $\epsilon \rightarrow 0$, this result implies that the assortments obtained by our approach perform particularly well when each product, by itself, does not consume too much of the space available in its nest. For example, if each product consumes less than 10% of the space available in a nest, then we obtain a performance guarantee of 10/9 instead of two.

To see that the collection of assortments $\{A_i^t : t \in \mathcal{T}_i\}$ constructed in Section 5.1 includes a $1/(1 - \epsilon)$ -approximate solution to problem (7) for any $u \in \mathfrak{R}_+$, assume that we solve the knapsack problem in (10) for some $u \in \mathfrak{R}_+$. If the point u lies in the interval \mathcal{I}_i^g , then the optimal solution to the linear programming relaxation of the knapsack problem is given by x_i^g . If the solution x_i^g does not have any fractional components, then the assortment $S_i^g \in \{A_i^t : t \in \mathcal{T}_i\}$ is an optimal solution to the knapsack problem and the desired result follows immediately. Otherwise, if the solution x_i^g has a fractional component, then it must be the case that the capacity of the knapsack in problem (10) is totally consumed by the solution x_i^g . Therefore, using j^g to denote the fractional component of the solution x_i^g and noting the definition of S_i^g , we obtain $\sum_{j \in N} w_{ij} S_{ij}^g + w_{ij^g} \geq c_i$. On the other hand, noting that all of the products in the assortment S_i^g take value one in the optimal solution to the linear programming relaxation of the knapsack problem, but product j^g takes a fractional value, the utility to space consumption ratios of the products in the assortment S_i^g must be at least as large as the utility to space consumption ratio of product j^g , which implies that

$$\sum_{j \in N} v_{ij} (r_{ij} - u) S_{ij}^g = \sum_{j \in N} \frac{v_{ij} (r_{ij} - u)}{w_{ij}} w_{ij} S_{ij}^g \geq \frac{v_{ij^g} (r_{ij^g} - u)}{w_{ij^g}} \sum_{j \in N} w_{ij} S_{ij}^g,$$

where the inequality follows from the fact that if product j is in assortment S_i^g satisfying $S_{ij}^g = 1$, then its utility to space consumption ratio $v_{ij} (r_{ij} - u)/w_{ij}$ should be at least as large as the utility to space consumption ratio of product j^g . In this case, if we use $z^*(u)$ to denote the optimal objective value of the knapsack problem in (10) and note that the optimal objective value of the

linear programming relaxation of problem (10) yields an upper bound on $z^*(u)$, then we obtain the chain of inequalities

$$\begin{aligned}
z^*(u) &\leq \sum_{j \in N} v_{ij} (r_{ij} - u) S_{ij}^g + v_{ij^g} (r_{ij^g} - u) \\
&= \left\{ 1 + \frac{v_{ij^g} (r_{ij^g} - u)}{\sum_{j \in N} v_{ij} (r_{ij} - u) S_{ij}^g} \right\} \sum_{j \in N} v_{ij} (r_{ij} - u) S_{ij}^g \leq \left\{ 1 + \frac{w_{ij^g}}{\sum_{j \in N} w_{ij} S_{ij}^g} \right\} \sum_{j \in N} v_{ij} (r_{ij} - u) S_{ij}^g \\
&\leq \left\{ 1 + \frac{w_{ij^g}}{c_i - w_{ij^g}} \right\} \sum_{j \in N} v_{ij} (r_{ij} - u) S_{ij}^g \leq \frac{1}{1 - \epsilon} \sum_{j \in N} v_{ij} (r_{ij} - u) S_{ij}^g,
\end{aligned}$$

where the first inequality is identical to the second inequality in (11), the second inequality uses the fact that $\sum_{j \in N} v_{ij} (r_{ij} - u) S_{ij}^g \geq [v_{ij^g} (r_{ij^g} - u)/w_{ij^g}] \sum_{j \in N} w_{ij} S_{ij}^g$ shown above, the third inequality follows by noting that $\sum_{j \in N} w_{ij} S_{ij}^g + w_{ij^g} \geq c_i$ and the fourth inequality follows by recalling the assumption that $w_{ij} \leq \epsilon c_i$. The chain of inequalities above shows that S_i^g is a $1/(1 - \epsilon)$ -approximate solution to problem (7). So, the collection of assortments $\{A_i^t : t \in \mathcal{T}_i\}$ as defined in Section 5.1 includes a $1/(1 - \epsilon)$ -approximate solution to problem (7) for any $u \in \mathfrak{R}_+$, as desired. Therefore, the collection of assortments $\{A_i^t : t \in \mathcal{T}_i\}$ constructed by using the approach in Section 5.1 not only includes a 2-approximate solution to problem (7) for any $u \in \mathfrak{R}_+$, but if each product consumes no more than a fraction ϵ of the capacity available in a nest, then this collection includes a $1/(1 - \epsilon)$ -approximate solution to problem (7) for any $u \in \mathfrak{R}_+$. In this case, by Theorem 4, the expected revenue from the best assortment $(\hat{S}_1, \dots, \hat{S}_m)$ with $\hat{S}_i \in \{A_i^t : t \in \mathcal{T}_i\}$ deviates from the optimal expected revenue by at most a factor of $\min\{2, 1/(1 - \epsilon)\}$.

To gain some insight into the structure of the assortments in the collection $\{A_i^t : t \in \mathcal{T}_i\} = \{S_i^g : g \in \mathcal{G}_i\} \cup \{\{j\} : j \in N\}$, we recall that each one of the assortments in $\{S_i^g : g \in \mathcal{G}_i\}$ is obtained by rounding down the solution to the linear programming relaxation of the knapsack problem in (10) for some $u \in \mathfrak{R}_+$. Since we can solve the linear programming relaxation of this knapsack problem by ordering the products with respect to their utility to space consumption ratios, our results in this section suggest that we can obtain good assortments by prioritizing the products in nest i according to the coefficients $\{v_{ij} (r_{ij} - u)/w_{ij} : j \in N\}$ for some $u \in \mathfrak{R}_+$ and the value of u is common to all products in this nest. We can visualize u as the imputed revenue adjustment for offering a product in nest i , in which case, $(r_{ij} - u)/w_{ij}$ can be interpreted as the adjusted revenue of product j in nest i per unit of space consumed. So, our results in this section suggest that it is sensible to prioritize the products in a nest according to the product of their preference weights and their adjusted revenues per unit of space consumed.

The approach in Section 4 yields an optimal assortment under cardinality constraints, whereas the approach in this section yields only an approximate solution under space constraints. This distinction is due to the fact that problem (9) is a knapsack problem where each item occupies one unit of space and we are able to solve such a knapsack problem in a tractable fashion. In contrast, problem (10) is a knapsack problem with general space consumptions, which is NP-hard. So, we have to be content with an approximate solution under space constraints.

6 Joint Assortment Optimization and Pricing

In this section, we consider a joint assortment optimization and pricing problem under the nested logit model, where we choose the assortment of products to offer in each nest and set the prices of the offered products. Customers choose among the offered products according to the nested logit model and the price of a product affects its preference weight, with higher prices resulting in lower preference weights. Our goal is to choose the assortment of products to offer in each nest and set the prices of the offered products to maximize the expected revenue obtained from each customer. The crucial idea that we use in this section is to transform the joint assortment optimization and pricing problem into a constrained assortment optimization problem. To achieve this transformation, we create multiple copies of each product, where each copy of a product corresponds to offering the product at a different price level. In this case, the joint assortment optimization and pricing problem reduces to the problem of finding which copy of a product, if any, should be offered. Once we transform the joint assortment optimization and pricing problem into a constrained assortment optimization problem, we are able to build on our earlier results to solve the joint assortment optimization and pricing problem to optimality.

We have m nests indexed by $M = \{1, \dots, m\}$. In each nest, there are p products and we index the products by $P = \{1, \dots, p\}$. Each product can be offered at b different price levels, which are indexed by $B = \{1, \dots, b\}$. Our notation implies that the number of possible price levels for each product is the same, but relaxing this assumption is straightforward. We use ρ_{ik}^l to denote the price associated with price level l of product k in nest i and ν_{ik}^l to denote the preference weight of product k in nest i when we offer this product at price level l . As mentioned above, we can formulate the joint assortment offering and pricing problem as an assortment optimization problem with constraints on the offered assortment. In particular, we create b copies of each product corresponding to different price levels. Thus, we have a total of $n = pb$ product copies in each nest. We refer to each one of these product copies as a virtual product so that each virtual product corresponds to offering a certain product at a certain price level. We index the virtual products in each nest by $N = \{1, \dots, n\}$. We use N_k to denote the set of virtual products corresponding to product $k \in P$. Thus, each one of the virtual products in N_k corresponds to offering product $k \in P$ at a certain price level and we have $|N_k| = b$ and $N_k \cap N_{k'} = \emptyset$ for all distinct $k, k' \in P$. In this case, the joint assortment optimization and pricing problem becomes that of finding an assortment of virtual products to offer in each nest subject to the constraint that we offer at most one virtual product corresponding to each product in a nest. By offering at most one virtual product corresponding to each product in a nest, we ensure that we choose at most one price level for each product. In this way, we can formulate the joint assortment optimization and pricing problem as a constrained assortment optimization problem involving the virtual products.

We use the vector $S_i = (S_{i1}, \dots, S_{in}) \in \{0, 1\}^n$ to denote the assortment of virtual products that we offer in nest i , where $S_{ij} = 1$ if we offer virtual product j in nest i and $S_{ij} = 0$ otherwise. If virtual product j corresponds to offering product k at price level l , then offering virtual product

j means that we offer product k at price level l . The feasible assortments in nest i are given by $\mathcal{C}_i = \{S_i \in \{0, 1\}^n : \sum_{j \in N_k} S_{ij} \leq 1 \forall k \in P\}$, which ensures that we offer at most one virtual product for each product $k \in P$. Therefore, if a product is offered, then it is offered at one price level. We use r_{ij} to denote the revenue associated with virtual product j in nest i . Specifically, if virtual product j corresponds to offering product k at price level l , then we have $r_{ij} = \rho_{ik}^l$. Similarly, we use v_{ij} to denote the preference weight associated with virtual product j in nest i . If virtual product j corresponds to offering product k at price level l , then we have $v_{ij} = \nu_{ik}^l$.

Once we formulate the problem as a constrained assortment optimization problem involving the virtual products, we can follow an approach similar to the one in Section 4 to come up with a collection of assortments $\{A_i^t : t \in \mathcal{T}_i\}$ with $|\mathcal{T}_i| = O(pb^2)$ such that this collection includes an optimal solution to problem (7) for any $u \in \mathfrak{R}_+$. In this case, we can obtain the optimal assortment by solving a linear program with $1 + m$ decision variables and $O(mp b^2)$ constraints. To characterize the optimal solution to problem (7) with the definition of \mathcal{C}_i given above, we use the decision variables $x_i = (x_{i1}, \dots, x_{in}) \in \{0, 1\}^n$ to write problem (7) as

$$\max \left\{ \sum_{j \in N} v_{ij} (r_{ij} - u) x_{ij} : \sum_{j \in N_k} x_{ij} \leq 1 \forall k \in P, x_{ij} \in \{0, 1\} \forall j \in N \right\}. \quad (12)$$

The constraints above ensure that at most one virtual product is offered corresponding to any product in P , implying that we choose at most one price level for each product. For any $k \in P$, we can find the optimal values of the decision variables $\{x_{ij} : j \in N_k\}$ in the problem above as follows. Noting the constraint $\sum_{j \in N_k} x_{ij} \leq 1$, among the decision variables $\{x_{ij} : j \in N_k\}$, we find the one with the largest objective function coefficient. If this objective function coefficient is positive, then we set the value of this decision variable to one and the values of the other decision variables in $\{x_{ij} : j \in N_k\}$ to zero. Otherwise, we set the values of all decision variables in $\{x_{ij} : j \in N_k\}$ to zero. Therefore, the optimal values of the decision variables $\{x_{ij} : j \in N_k\}$ in the problem above depend only on the ordering and signs of their objective function coefficients. To exploit this observation, defining the linear functions $f_{ij}(u) = v_{ij} (r_{ij} - u)$ for $j \in N$ and $f_{i0}(u) = 0$, we find the intersection points of the $b + 1$ linear functions $\{f_{ij}(\cdot) : j \in N_k \cup \{0\}\}$ by solving $f_{ij}(u) = f_{ij'}(u)$ for u for all distinct $j, j' \in N_k \cup \{0\}$. We let $\{\bar{u}_{ik}^g : g \in \mathcal{G}_{ik}\}$ with $|\mathcal{G}_{ik}| = O(b^2)$ be the set of intersection points obtained in this fashion. Thus, the ordering between the values of the linear functions $\{f_{ij}(u) : j \in N_k \cup \{0\}\}$ does not change as long as u takes values between two successive points in the set $\{\bar{u}_{ik}^g : g \in \mathcal{G}_{ik}\}$. Repeating the same argument for all $k \in P$, we obtain the points $\{\bar{u}_{ik}^g : k \in P, g \in \mathcal{G}_{ik}\}$ and noting that $|\mathcal{G}_{ik}| = O(b^2)$, there are a total of $O(pb^2)$ points in the set $\{\bar{u}_{ik}^g : k \in P, g \in \mathcal{G}_{ik}\}$. In this case, the points $\{\bar{u}_{ik}^g : k \in P, g \in \mathcal{G}_{ik}\}$ partition the positive real line into $O(pb^2)$ intervals. We denote these intervals by $\{\mathcal{I}_i^t : t \in \mathcal{T}_i\}$ with $|\mathcal{T}_i| = O(pb^2)$.

The key observation is that the ordering and signs of the objective function coefficients in problem (12) do not change as long as u takes values in one of the intervals $\{\mathcal{I}_i^t : t \in \mathcal{T}_i\}$. So, the optimal solution does not change either as long as u takes values in one of these intervals. Therefore, by checking the ordering and signs of the objective function coefficients in each one of the intervals

$\{\mathcal{I}_i^t : t \in \mathcal{T}_i\}$, we can come up with the optimal solution to problem (12) for any $u \in \mathfrak{R}_+$. Noting that there are $O(pb^2)$ intervals in $\{\mathcal{I}_i^t : t \in \mathcal{T}_i\}$, we can come up with a collection of assortments $\{A_i^t : t \in \mathcal{T}_i\}$ with $|\mathcal{T}_i| = O(pb^2)$ such that this collection always includes an optimal solution to problem (7) for any $u \in \mathfrak{R}_+$. These observations, in view of Theorems 2 and 4, imply that we can solve a linear program with $1 + m$ decision variables and $O(mp b^2)$ constraints to obtain an optimal solution to the joint assortment optimization and pricing problem.

Li and Huh (2011) and Gallego and Wang (2011) consider pricing problems under the nested logit model. Our approach for the joint assortment optimization and pricing problem provides some unique features when compared with their work. Both Li and Huh (2011) and Gallego and Wang (2011) assume that there exists a parametric relationship between the price of a product and its preference weight. In particular, they assume that if we set the price of product k in nest i as p_{ik} , then the preference weight of this product is given by $e^{\alpha_{ik} - \beta_{ik} p_{ik}}$, where α_{ik} and β_{ik} are fixed parameters of their model. In contrast, we assume that if we use the price level l for product k in nest i , then the price of this product is given by ρ_{ik}^l and the preference weight of this product is given by ν_{ik}^l and the relationship between ρ_{ik}^l and ν_{ik}^l can be completely arbitrary. Thus, our approach does not rely on a parametric relationship between the price and preference weight of a product. Furthermore, since we work with a finite number of possible price levels, we can explicitly restrict and bound the prices that can be chosen by our model, providing considerable control in our pricing decisions. For example, if we do not want to use prices outside a certain range, then we can simply define the set of possible prices $\{\rho_{ik}^l : l \in B\}$ of a product within the desired range, in which case, our model would never choose a price outside the desired range. Similarly, if we are interested in prices that lie on a certain grid, then we can define the set of possible prices $\{\rho_{ik}^l : l \in B\}$ by focusing only on the prices that lie on this grid. In certain retail applications, for example, it may be necessary to use prices that lie on the grid $\{\$49.99, \$59.99, \$69.99, \dots\}$ and our approach allows us to easily set the prices of the products on this grid. To our knowledge, the approach followed by Li and Huh (2011) and Gallego and Wang (2011) is not able to impose such explicit restrictions on prices.

7 Computational Experiments

Section 5 shows that we can solve a linear program with $1 + m$ decision variables and $O(mn^2)$ constraints to obtain a $\min\{2, 1/(1 - \epsilon)\}$ -approximate solution to problem (1) under space constraints. Throughout this section, we refer to this approach as CFP, standing for constant factor performance guarantee. In this section, our goal is to give computational experiments that test the performance of the assortments obtained by CFP under space constraints. Noting that we can obtain the optimal assortment under cardinality constraints, we do not give computational experiments under cardinality constraints. Similarly, since it is possible to obtain the optimal solution to the joint assortment optimization and pricing problem in a tractable fashion, we do not give computational experiments for this problem setting either.

7.1 Experimental Setup

The performance guarantee of $\min\{2, 1/(1 - \epsilon)\}$ for CFP can be assuring since this performance guarantee shows that CFP never performs arbitrarily poorly, always obtaining at least half of the optimal expected revenue. However, a guarantee of obtaining half of the optimal expected revenue may not be thoroughly satisfying from a practical perspective and a natural question is whether we can come up with a way to assess the performance of CFP for an individual problem instance. It turns out that we can solve a linear program to obtain an upper bound on the optimal expected revenue Z^* in problem (1). In particular, it is possible to show that the optimal objective value of the linear program

$$\min \left\{ z : v_0 z \geq \sum_{i \in M} y_i, y_i \geq \left(\sum_{j \in N} v_{ij} x_{ij}^g \right)^{\gamma_i} \left\{ \frac{\sum_{j \in N} v_{ij} r_{ij} x_{ij}^g}{\sum_{j \in N} v_{ij} x_{ij}^g} - z \right\} \forall g \in \mathcal{G}_i, i \in M \right\}, \quad (13)$$

provides an upper bound on the optimal expected revenue Z^* in problem (1) when there are space constraints on the offered assortment. We show this fact in Proposition 7 in Online Supplement A. In the linear program above, the decision variables are (z, y) . The quantity x_i^g , as defined in Section 5, corresponds to the optimal solution to the linear programming relaxation of the knapsack problem in (10) when u takes values in the interval \mathcal{I}_i^g . Thus, we can get a feel for the optimality gap of the assortment obtained by CFP simply by comparing the expected revenue from this assortment with the upper bound on the optimal expected revenue provided by the optimal objective value of the linear program in (13).

In our computational experiments, we generate a large number of problem instances. For each problem instance, we compute the assortment obtained by CFP and solve the linear program in (13). In this case, by comparing the expected revenue from the assortment obtained by CFP with the upper bound on the optimal expected revenue provided by the linear program in (13), we can bound the optimality gap of the assortment obtained by CFP. In all of our problem instances, the number of nests is $m = 5$. The number of products in each nest is either $n = 15$ or $n = 30$. To generate the dissimilarity parameter γ_i of each nest i , we sample γ_i from the uniform distribution over $[0.25, 0.75]$. To come up with the revenues and preference weights of the products, we generate U_{ij} from the uniform distribution over $[0, 1]$ and generate X_{ij} and Y_{ij} from the uniform distribution over $[0.75, 1.25]$. We set the revenue of product j in nest i as $r_{ij} = 10 \times U_{ij}^2 \times X_{ij}$ and set the preference weight of product j in nest i as $v_{ij} = 10 \times (1 - U_{ij}) \times Y_{ij}$. In these expressions for r_{ij} and v_{ij} , the role of the parameter U_{ij} is to introduce negative correlation between the revenues and the preference weights so that the more expensive products tend to have smaller preference weights. On the other hand, the role of the parameters X_{ij} and Y_{ij} is to introduce some idiosyncratic noise in the revenues and the preference weights so that it is not always the case that expensive products have small preference weights. The exponent of two in U_{ij}^2 skews the distribution of the revenues so that we end up having a large number of products with small revenues, but a small number of products with large revenues. To come up with the preference weight v_0 of the no purchase option, after generating the dissimilarity parameters of all of the nests and the preference weights of all of

the products, we calibrate the preference weight v_0 of the no purchase option so that a customer leaves without making any purchase with probability P_0 even when all of the products in all of the nests are offered. We use either $P_0 = 0.2$ or $P_0 = 0.4$ in our computational experiments.

To come up with the space consumptions of the products, we generate w_{ij} from the uniform distribution over $[1, 10]$. We set the space availability c_i of nest i such that the space availability of the nest is a fraction β of the total space consumption of the products in this nest. In particular, we set the space availability of nest i as $c_i = \beta \sum_{j \in N} w_{ij}$, but if c_i comes out to be smaller than any of the space consumptions $\{w_{ij} : j \in N\}$ of the products in nest i , then we bump c_i up to $\max\{w_{ij} : j \in N\}$. In this way, we ensure that each product fits into the space availability of its nest, satisfying $w_{ij} \leq c_i$ for all $i \in M, j \in N$. We use $\beta = 0.1$ or $\beta = 0.2$ or $\beta = 0.3$.

In our computational experiments, we use $n \in \{15, 30\}$, $P_0 \in \{0.2, 0.4\}$ and $\beta \in \{0.1, 0.2, 0.3\}$, yielding a total of 12 parameter combinations. For each parameter combination, we randomly generate 10,000 individual problem instances as described in the previous two paragraphs. For each problem instance, we compute the assortment obtained by CFP and we solve the linear program in (13) to obtain an upper bound on the optimal expected revenue.

7.2 Computational Results

Table 1 gives our computational results. The first column in this table shows the parameter combination of the test problems by using the triplet (n, P_0, β) . We recall that there are 10,000 problem instances generated in each parameter combination. For each problem instance, we compute the upper bound on the optimal expected revenue provided by the optimal objective value of problem (13). We let UB^k be this upper bound for problem instance k . We use CFP to obtain an assortment with $\min\{2, 1/(1-\epsilon)\}$ -approximation guarantee for each problem instance. For problem instance k , we let $RCFP^k$ be the expected revenue from the assortment obtained by CFP. The second column in Table 1 shows the average percent gap between UB^k and $RCFP^k$ over the problem instances in a particular parameter combination. The third column shows the 95th percentile of the percent gaps between UB^k and $RCFP^k$. In other words, the second and third columns respectively give the average and 95th percentile of the data $\{100(UB^k - RCFP^k)/UB^k : k = 1, \dots, 10,000\}$. The fourth column in Table 1 shows the number of problem instances where the percent gap between UB^k and $RCFP^k$ is less than 1%. The interpretations of the fifth to ninth columns are similar to that of the fourth column, but these five columns respectively give the number of problem instances where the percent gap between UB^k and $RCFP^k$ is less than 2%, 3%, 4%, 5% and 10%. The tenth column in Table 1 shows the average ratio between the space requirement of a product and the space availability of its nest, averaged over all products in all nests and over all problem instances in a parameter combination. If each item occupies a large fraction of the capacity availability in its nest, then the value of ϵ in the performance guarantee $\min\{2, 1/(1-\epsilon)\}$ becomes large and we expect the performance of CFP to deteriorate. Finally, the goal of the last column in Table 1 is to give a feel for the binding nature of the space constraints. In particular, this column shows the

Param. Combin. (n, P_0, β)	UB ^k and RCFP ^k % Gap		No. Probs. Given UB ^k and RCFP ^k % Gap						Avg. Cons. to. Cap. Rat.	No. of Const. Prob.
	Avg.	95th	1%	2%	3%	4%	5%	10%		
(15, 0.2, 0.3)	0.79	2.08	6,879	9,433	9,941	9,994	9,999	10,000	0.22	9,487
(15, 0.2, 0.2)	2.01	4.22	2,044	5,539	8,138	9,343	9,813	10,000	0.33	9,978
(15, 0.2, 0.1)	4.34	8.29	237	1,221	2,929	4,921	6,655	9,856	0.58	9,999
(15, 0.4, 0.3)	1.50	3.14	3,286	7,389	9,375	9,870	9,979	10,000	0.22	9,962
(15, 0.4, 0.2)	3.01	5.59	499	2,565	5,424	7,799	9,090	9,998	0.33	10,000
(15, 0.4, 0.1)	5.59	9.97	37	360	1,279	2,772	4,467	9,510	0.58	10,000
(30, 0.2, 0.3)	0.29	0.79	9,844	10,000	10,000	10,000	10,000	10,000	0.11	9,283
(30, 0.2, 0.2)	0.85	1.71	6,663	9,808	9,994	10,000	10,000	10,000	0.17	9,976
(30, 0.2, 0.1)	2.32	4.10	645	4,146	7,771	9,403	9,892	10,000	0.33	10,000
(30, 0.4, 0.3)	0.65	1.33	8,261	9,981	10,000	10,000	10,000	10,000	0.11	9,963
(30, 0.4, 0.2)	1.30	2.34	3,252	8,844	9,930	9,999	10,000	10,000	0.17	10,000
(30, 0.4, 0.1)	2.80	4.73	194	2,403	6,123	8,688	9,676	10,000	0.33	10,000
Average	2.12	4.02								

Table 1: Performance of CFP.

number of problem instances for which the unconstrained optimal solution to problem (1) violates the space constraint for at least one of the nests.

The results in Table 1 show that the assortments obtained by CFP perform quite well. The average optimality gap of these assortments is no larger than 2.12%. For nine out of 12 parameter combinations, the 95th percentile of the optimality gaps comes out to be smaller than 5%, which shows that the expected revenue loss from using CFP can be bounded by 5% for an overwhelming majority of our problem instances. The parameter combinations for which CFP provides the smallest optimality gaps correspond to the cases with $\beta = 0.3$. When the value of β is large, the space availability in each nest tends to be large and each product occupies a small fraction of the space available in a nest. This trend can also be observed from the second to last column in Table 1, which shows that the average ratio between the space consumption of the products and the space availability of the nests is small when $\beta = 0.3$. For these cases, the value of ϵ in the performance guarantee of $\min\{2, 1/(1 - \epsilon)\}$ comes out to be small and this performance guarantee becomes close to one. Therefore, it is not too surprising that CFP performs well when β is on the large side. The most problematic parameter combination (15, 0.4, 0.1) corresponds to a small value of β with $\beta = 0.1$, but the average optimality gap of the assortments provided by CFP is still no larger than 5.59% and the optimality gaps are no larger than 9.97% in more than 95% of the problem instances. For this parameter combination, the second to last column shows that each item occupies more than half of the space availability in a nest on average, indicating that we would not be able to offer more than two or three products in each nest. Even in such a drastic case, CFP continues to perform within 10% of the upper bound on the optimal expected revenue for more than 95% of the problem instances.

Over all problem instances, in 109,571 out of 120,000 cases, the optimality gaps turn out to be less than 5%. We note that the attractive performance of CFP in Table 1 is not only due to the

fact that this approach finds good assortments, but also due to the fact that the linear program in (13) provides good upper bounds. Even if we use an arbitrary heuristic to find a solution to problem (1), we can still use the linear program in (13) to obtain an upper bound on the optimal expected revenue and we can assess the optimality gap of the heuristic by comparing this upper bound with the expected revenue from the assortment obtained by the heuristic. If the gap turns out to be small, then there is no need to look for better assortments.

8 Conclusions and Extensions

In this paper, we study assortment problems under the nested logit model with constraints on the assortment offered in each nest. We obtain the optimal assortment under cardinality constraints and an assortment with a performance guarantee under space constraints. Building on our results for assortment optimization, we can solve a joint assortment optimization and pricing problem to optimality. Theorems 2 and 4 are pivotal in developing our approach. Given a collection of feasible assortments for each nest, Theorem 2 shows that we can solve the linear program in (4) to stitch together the best assortment over all nests. It is worthwhile to note that the form of this linear program is linked to the nonlinear fractional form of the expected revenue function. In particular, noting that Z^* is the optimal objective value of problem (1), we have $Z^* \geq \Pi(S_1, \dots, S_m)$ for all $(S_1, \dots, S_m) \in \mathcal{C}_1 \times \dots \times \mathcal{C}_m$ and the inequality holds as an equality at the optimal assortment. By the definition of $\Pi(S_1, \dots, S_m)$, we can write the last inequality as $v_0 Z^* \geq \sum_{i \in M} V_i(S_i)^{\gamma_i} (R_i(S_i) - Z^*)$, which implies that the optimal assortment solves the problem

$$\sum_{i \in M} \max_{S_i \in \mathcal{C}_i} \left\{ V_i(S_i)^{\gamma_i} (R_i(S_i) - Z^*) \right\}.$$

Theorem 4, on the other hand, shows that if we can come up with a reasonably small collection of assortments that includes an α -approximate solution to problem (7) for any $u \in \mathfrak{R}_+$, then we can use these assortments as candidate assortments to obtain an α -approximate solution to the assortment optimization problem.

We observe that the objective function of problem (7) reduces to $\sum_{j \in N} v_{ij} (r_{ij} - u) S_{ij}$, which is linear in S_i . Exploiting this linearity, if the constraint $S_i \in \mathcal{C}_i$ in problem (7) corresponds to cardinality or space constraints, then we obtain good solutions to this problem simply by checking all possible orderings of the objective function coefficients. The same approach is also useful to solve the joint assortment optimization and pricing problem. There may be other constraints under which we can obtain good solutions to problem (7) by checking all possible orderings of the objective function coefficients $\{v_{ij} (r_{ij} - u) : j \in N\}$ for all $u \in \mathfrak{R}_+$. To make this point concrete, in Online Supplement B, we give two additional types of constraints under which we can obtain assortments with performance guarantees. Under the first type of constraints, each product is designated either as a parent product or as a child product with an associated parent. A child product cannot be offered unless its parent product is offered. For this setting, we show how to obtain the optimal assortment. Under the second type of constraints, we have

both cardinality and space constraints on the offered assortment and we show how to obtain a 2-approximate solution. Furthermore, we can build on our approach to develop approximation algorithms that provide better performance guarantees as we increase the computational effort. In Online Supplement C, we give a concrete example of this possibility by showing that if we have space constraints in problem (1), then given any $\alpha > 1$, we can solve a linear program with $1 + m$ decision variables and $O(m \lceil \alpha / (\alpha - 1) \rceil n^{\lceil \alpha / (\alpha - 1) \rceil + 2})$ constraints to find an assortment whose expected revenue deviates from the optimal by at most a factor of α . This result holds for any $\alpha > 1$. Thus, if we are willing to increase the number of constraints in the linear program, then by choosing α closer to one, we can obtain a performance guarantee closer to one. Finally, using $\hat{S}_i(u)$ to denote an optimal solution to problem (7) and applying Theorem 4 with $\alpha = 1$ implies that if we use $\{\hat{S}_i(u) : u \in \mathfrak{R}_+\}$ as a collection of candidate assortments for each nest i , then we can stitch together the optimal assortment for problem (1) by using these collections. We can show that the collection $\{\hat{S}_i(u) : u \in \mathfrak{R}_+\}$ also includes an assortment that maximizes the expected revenue only from nest i . Thus, we can use the same collection of assortments to maximize the expected revenue over all nests or to maximize the expected revenue only from nest i , but which one of these assortments we end up picking to maximize the expected revenue over all nests can be entirely different from which one we end up picking to maximize the expected revenue only from nest i . We dwell on this connection in Online Supplement D.

Considering extensions of our work, Davis et al. (2011) use a version of the nested logit model where v_{i0} is the preference weight of the no purchase option in nest i . The total preference weight of the products offered in nest i is $V_i(S_i) = v_{i0} + \sum_{j \in N} v_{ij} S_{ij}$ so that a customer can leave without purchasing anything even after choosing a nest. In this case, the assortment problem becomes NP-hard even without constraints on the offered assortment. To get an approximation algorithm for the assortment problem without any constraints, the authors relate the problem to a knapsack problem and use existing approximation algorithms for the knapsack problem. If we incorporate cardinality or space constraints into their approach, then their knapsack problem turns into a two-dimensional knapsack problem and we can use existing approximation algorithms for two-dimensional knapsack problems. The same authors also study the case with the dissimilarity parameters $\{\gamma_i : i \in M\}$ exceeding one. The assortment problem even without any constraints becomes NP-hard in case as well. Since Lemma 3 uses the fact that $V_i(S_i)^{\gamma_i}$ is concave in $V_i(S_i)$ when $\gamma_i \in (0, 1]$, we need a new approach when the dissimilarity parameters exceed one. Finally, we assume that the preference weights and the dissimilarity parameters are fixed, but we can try to allow these parameters to depend on the offered assortment. In this paper, we are able to formulate problem (7) as a linear integer program, such as the ones in (9) and (10), but we lose the linear structure when the preference weights depend on the offered assortment. In addition, Lemma 3 uses the fact that $V_i(S_i)^{\gamma_i}$ is concave in $V_i(S_i)$ when $\gamma_i \in (0, 1]$, but it is not clear how to define the corresponding concavity property when the dissimilarity parameters depend on the offered assortment. Thus, we need a new approach to deal with assortment dependent preference weights and dissimilarity parameters.

References

- Anderson, S. P. and de Palma, A. (1992), ‘Multiproduct firms: A nested logit approach’, *The Journal of Industrial Economics* **40**(3), 261–276.
- Ansari, A., Bawa, K. and Ghosh, A. (1995), ‘A nested logit model of brand choice incorporating variety-seeking and marketing-mix variables’, *Marketing Letters* **6**(3), 199–210.
- Baltas, G., Doyle, P. and Dyson, P. (1997), ‘A model of consumer choice for national vs. private label brands’, *The Journal of the Operational Research Society* **48**(10), 988–995.
- Bell, D. R. and Lattin, J. M. (1998), ‘Shopping behavior and consumer preference for store price format: Why “large basket” shoppers prefer EDLP’, *Marketing Science* **17**(1), 66–88.
- Ben-Akiva, M. and Lerman, S. (1994), *Discrete Choice Analysis: Theory and Application to Travel Demand*, MIT Press in Transportation Studies, Cambridge, MA.
- Borsch-Supan, A. (1990), ‘On the compatibility of nested logit models with utility maximization’, *Journal of Econometrics* **43**, 373–388.
- Bront, J. J. M., Mendez-Diaz, I. and Vulcano, G. (2009), ‘A column generation algorithm for choice-based network revenue management’, *Operations Research* **57**(3), 769–784.
- Bucklin, R. E. and Gupta, S. (1992), ‘Brand choice, purchase incidence, and segmentation: An integrated modeling approach’, *Journal of Marketing Research* **29**(2), 201–215.
- Davis, J., Gallego, G. and Topaloglu, H. (2011), Assortment optimization under variants of the nested logit model, Technical report, Cornell University, School of Operations Research and Information Engineering.
- Dong, L., Kouvelis, P. and Tian, Z. (2009), ‘Dynamic pricing and inventory control of substitute products’, *Manufacturing & Service Operations Management* **11**(2), 317–339.
- Farias, V. F., Jagabathula, S. and Shah, D. (2011), ‘A non-parametric approach to modeling choice with limited data’, *Management Science* (to appear).
- Farias, V. F., Jagabathula, S. and Shah, D. (2012), Sparse choice models, Technical report, MIT, Cambridge, MA.
- Frieze, A. M. and Clarke, M. R. B. (1984), ‘Approximation algorithms for the m -dimensional 0-1 knapsack problem: Worst-case and probabilistic analyses’, *European Journal of Operational Research* **15**, 100–109.
- Gallego, G., Iyengar, G., Phillips, R. and Dubey, A. (2004), Managing flexible products on a network, Technical report, Columbia University.
- Gallego, G. and Wang, R. (2011), Multi-product price optimization and competition under the nested attraction model, Technical report, Columbia University.
- Guadagni, P. M. and Little, J. D. C. (1998), ‘When and what to buy: A nested logit model of coffee purchase’, *Journal of Forecasting* **17**(3–4), 303–326.
- Hanson, W. and Martin, K. (1996), ‘Optimizing multinomial logit profit functions’, *Management Science* **42**(7), 992–1003.
- Kok, A. G., Fisher, M. L. and Vaidyanathan, R. (2008), Assortment planning: Review of literature and industry practice, in ‘Retail Supply Chain Management’, Kluwer Publishers.
- Kunnumkal, S. and Topaloglu, H. (2008), ‘A refined deterministic linear program for the network revenue management problem with customer choice behavior’, *Naval Research Logistics Quarterly* **55**(6), 563–580.

- Li, H. and Huh, W. T. (2011), ‘Pricing multiple products with the multinomial logit and nested models: Concavity and implications’, *Manufacturing & Service Operations Management* **13**(4), 549–563.
- Liu, Q. and van Ryzin, G. (2008), ‘On the choice-based linear programming model for network revenue management’, *Manufacturing & Service Operations Management* **10**(2), 288–310.
- Luce, R. D. (1959), *Individual Choice Behavior: A Theoretical Analysis*, Wiley, New York, NY.
- McFadden, D. (1974), Conditional logit analysis of qualitative choice behavior, in P. Zarembka, ed., ‘*Frontiers in Economics*’, Academic Press, pp. 105–142.
- McFadden, D. (1978), Modeling the choice of residential location, in A. Karlqvist, L. Lundqvist, F. Snickars and J. Weibull, eds, ‘*Spatial Interaction Theory and Planning Models*’, North-Holland, Amsterdam, pp. 75–96.
- McFadden, D. (1980), ‘Econometric models for probabilistic choice among products’, *The Journal of Business* **53**(3), S13–29.
- McFadden, D. and Train, K. (2000), ‘Mixed MNL models for discrete response’, *Journal of Applied Economics* **15**, 447–470.
- Meissner, J., Strauss, A. and Talluri, K. (2012), ‘An enhanced concave program relaxation for choice network revenue management’, *Production and Operations Management* (to appear).
- Mendez-Diaz, I., Bront, J. J. M., Vulcano, G. and Zabala, P. (2010), ‘A branch-and-cut algorithm for the latent-class logit assortment problem’, *Discrete Applied Mathematics* **36**, 383–390.
- Richards, T. J. (2007), ‘A nested logit model of strategic promotion’, *Quantitative Marketing and Economics* **5**, 63–91.
- Rusmevichientong, P., Shen, Z.-J. M. and Shmoys, D. B. (2009), ‘A PTAS for capacitated sum-of-ratios optimization’, *Operations Research Letters* **37**(4), 230–238.
- Rusmevichientong, P., Shen, Z.-J. M. and Shmoys, D. B. (2010), ‘Dynamic assortment optimization with a multinomial logit choice model and capacity constraint’, *Operations Research* **58**(6), 1666–1680.
- Rusmevichientong, P., Shmoys, D. B. and Topaloglu, H. (2010), Assortment optimization with mixtures of logits, Technical report, Cornell University, School of Operations Research and Information Engineering.
- Siriwardena, S., Hunt, G., Teisl, M. F. and Noblet, C. L. (2012), ‘Effective environmental marketing of green cars: A nested-logit approach’, *Transportation Research Part D* **17**, 237–242.
- Song, J.-S. and Xue, Z. (2007), Demand management and inventory control for substitutable products, Technical report, Duke University, Durham, NC.
- Talluri, K. (2011), A randomized concave programming method for choice network revenue management, Technical report, Universitat Pompeu Fabra, Barcelona, Spain.
- Talluri, K. and van Ryzin, G. (2004), ‘Revenue management under a general discrete choice model of consumer behavior’, *Management Science* **50**(1), 15–33.
- Train, K. (2003), *Discrete Choice Methods with Simulation*, Cambridge University Press, Cambridge, UK.
- Williams, H. C. W. L. (1977), ‘On the formation of travel demand models and economic evaluation measures of user benefit’, *Environment and Planning* **9**(3), 285–344.
- Zhang, D. and Adelman, D. (2009), ‘An approximate dynamic programming approach to network revenue management with customer choice’, *Transportation Science* **42**(3), 381–394.

A Online Supplement: Proposition 7

Proposition 7 is used in Section 7 and it shows that the optimal objective value of problem (13) is an upper bound on the optimal expected revenue when we have space constraints.

Proposition 7 *If we use \hat{z} to denote the optimal objective value of the linear program in (13), then we have $\hat{z} \geq Z^*$.*

Proof. We let (\hat{z}, \hat{y}) be an optimal solution to problem (13) and (S_1^*, \dots, S_m^*) be an optimal solution to problem (1). We claim that $\hat{y}_i \geq V_i(S_i^*)^{\gamma_i} (R_i(S_i^*) - \hat{z})$ for all $i \in M$. To establish this claim, we note that if u is large enough, then the optimal solution to the linear programming relaxation of problem (10) is zero. Thus, letting \mathcal{I}_i^g and x_i^g be as defined in Section 5, there exists an interval \mathcal{I}_i^g such that the solution x_i^g to the linear programming relaxation of problem (10) is zero when u takes values in this interval. In this case, the second set of constraints in problem (13) implies that $\hat{y}_i \geq 0$ for all $i \in M$. Therefore, if $S_i^* = \bar{0}$ or $R_i(S_i^*) \leq \hat{z}$, then we have $V_i(S_i^*)^{\gamma_i} (R_i(S_i^*) - \hat{z}) \leq 0 \leq \hat{y}_i$ and our claim trivially holds when $S_i^* = \bar{0}$ or $R_i(S_i^*) \leq \hat{z}$. So, it is enough to establish our claim with $S_i^* \neq \bar{0}$ and $R_i(S_i^*) > \hat{z}$.

We let $\hat{u} = \gamma_i \hat{z} + (1 - \gamma_i) R_i(S_i^*)$. Letting g be the index of the interval \mathcal{I}_i^g that includes \hat{u} , by definition, x_i^g is the optimal solution to the linear programming relaxation of problem (10) when we solve this problem with $u = \hat{u}$. Therefore, we have $\sum_{j \in N} v_{ij} (r_{ij} - \hat{u}) x_{ij}^g \geq \sum_{j \in N} v_{ij} (r_{ij} - \hat{u}) S_{ij}^*$, where we use the fact that x_i^g is the optimal solution to the linear programming relaxation of problem (10) when we solve this problem with $u = \hat{u}$ but offering the products in S_i^* provides a feasible, but not necessarily an optimal, solution to this problem. First, we assume that $x_i^g \neq \bar{0}$. In this case, slightly abusing the notation to let $V_i(x_i^g) = \sum_{j \in N} v_{ij} x_{ij}^g$ and $R_i(x_i^g) = \sum_{j \in N} v_{ij} r_{ij} x_{ij}^g / V_i(x_i^g)$, the last inequality can equivalently be written as $V_i(x_i^g) (R_i(x_i^g) - \hat{u}) \geq V_i(S_i^*) (R_i(S_i^*) - \hat{u})$.

Noting that (\hat{z}, \hat{y}) is a feasible solution to problem (13), this solution satisfies the second set of constraints in problem (13) for the index g defined at the beginning of the previous paragraph, in which case, we can write this constraint in a compact fashion as $\hat{y}_i \geq V_i(x_i^g)^{\gamma_i} (R_i(x_i^g) - \hat{z})$. Also, using the fact that u^{γ_i} is a concave function of u , the subgradient inequality yields $V_i(S_i^*)^{\gamma_i} \leq V_i(x_i^g)^{\gamma_i} + \gamma_i V_i(x_i^g)^{\gamma_i - 1} (V_i(S_i^*) - V_i(x_i^g)) = \gamma_i V_i(x_i^g)^{\gamma_i - 1} V_i(S_i^*) + (1 - \gamma_i) V_i(x_i^g)^{\gamma_i}$. Using these observations, we have the chain of inequalities

$$\begin{aligned} \hat{y}_i &\geq V_i(x_i^g)^{\gamma_i} (R_i(x_i^g) - \hat{z}) = V_i(x_i^g)^{\gamma_i} (R_i(x_i^g) - \hat{u}) + (1 - \gamma_i) V_i(x_i^g)^{\gamma_i} (R_i(S_i^*) - \hat{z}) \\ &\geq V_i(x_i^g)^{\gamma_i - 1} V_i(S_i^*) (R_i(S_i^*) - \hat{u}) + (1 - \gamma_i) V_i(x_i^g)^{\gamma_i} (R_i(S_i^*) - \hat{z}) \\ &= \gamma_i V_i(x_i^g)^{\gamma_i - 1} V_i(S_i^*) (R_i(S_i^*) - \hat{z}) + (1 - \gamma_i) V_i(x_i^g)^{\gamma_i} (R_i(S_i^*) - \hat{z}) \\ &\geq V_i(S_i^*)^{\gamma_i} (R_i(S_i^*) - \hat{z}), \end{aligned}$$

where the first inequality follows from the inequality we establish at the beginning of this paragraph, the first equality follows by using the definition of \hat{u} and arranging the terms, the second inequality

follows from the fact that $V_i(x_i^g) (R_i(x_i^g) - \hat{u}) \geq V_i(S_i^*) (R_i(S_i^*) - \hat{u})$, which is established in the previous paragraph, the second equality follows by using the definition of \hat{u} and the third inequality follows by noting that $V_i(S_i^*)^{\gamma_i} \leq \gamma_i V_i(x_i^g)^{\gamma_i-1} V_i(S_i^*) + (1 - \gamma_i) V_i(x_i^g)^{\gamma_i}$, which is established above by using the subgradient inequality. So, our claim holds when $x_i^g \neq \bar{0}$.

Second, we assume that $x_i^g = \bar{0}$ so that the optimal solution to the linear programming relaxation of problem (10) is zero when this problem is solved with $u = \hat{u}$. Thus, the utility of each product in this problem should be negative, yielding $r_{ij} \leq \hat{u} = \gamma_i \hat{z} + (1 - \gamma_i) R_i(S_i^*)$ for all $j \in N$. Noting $R_i(S_i^*) > \hat{z}$, the last inequality gives $r_{ij} < R_i(S_i^*)$ for all $j \in N$. However, since $R_i(S_i^*) = \sum_{j \in N} v_{ij} r_{ij} S_{ij}^* / \sum_{j \in N} v_{ij} S_{ij}^*$ by definition, $R_i(S_i^*)$ is a weighted average of the product revenues $\{r_{ij} : j \in N\}$ in the assortment S_i^* . So, we cannot have $r_{ij} < R_i(S_i^*)$ for all $j \in N$, indicating that the case $x_i^g = \bar{0}$ cannot occur. Thus, our claim is established and we have $\hat{y}_i \geq V_i(S_i^*)^{\gamma_i} (R_i(S_i^*) - \hat{z})$ for all $i \in M$. Adding these inequalities over all $i \in M$ and noting that $v_0 \hat{z} \geq \sum_{i \in M} \hat{y}_i$ by the first constraint in problem (13), we obtain $v_0 \hat{z} \geq \sum_{i \in M} V_i(S_i^*)^{\gamma_i} (R_i(S_i^*) - \hat{z})$, in which case, solving for \hat{z} in this inequality and noting the definition of $\Pi(S_1, \dots, S_m)$, we get $\hat{z} \geq \Pi(S_1^*, \dots, S_m^*)$. So, \hat{z} is an upper bound on the expected revenue from the optimal assortment, as desired. \square

B Online Supplement: Extensions to Other Constraints

In this section, we show how to extend our approach to additional types of constraints.

B.1 Parent Product Constraints

We consider the case where certain products in a nest are designated as parent products. Each parent product has a set of child products associated with it and if a parent product is not offered, then none of its child products can be offered. We refer to this type of constraints as *parent product constraints*. Parent product constraints arise when company policy or law requires offering certain products before offering others. For example, a company may be required to offer the generic version of a drug before it can offer the brand name versions, in which case, the generic version acts as the parent product, the brand name versions are the child products of the generic version and if the generic version is not offered, then none of the brand name versions can be offered. To capture parent product constraints, we use P_i to denote the set of parent products in nest i . A parent product j in nest i has the set of child products C_{ij} . If product j is neither a parent product nor a child product, then we assume that product j is a parent product with an empty set of child products. Also, we assume that $C_{ij} \cap C_{ik} = \emptyset$ for all distinct $j, k \in P_i$ so that two parent products have different sets of child products. By the last two assumptions, the sets of products P_i and $\{C_{ij} : j \in P_i\}$ collectively partition N . So, the feasible assortments in nest i under parent constraints are given by $\mathcal{C}_i = \{S_i \in \{0, 1\}^N : S_{ik} \leq S_{ij} \forall j \in P_i, k \in C_{ij}\}$, ensuring that if we do not offer a parent product, then none of its child products can be offered.

Under parent product constraints, we show that we can come up with a collection of assortments

$\{A_i^t : t \in \mathcal{T}_i\}$ that includes an optimal solution to problem (7) for any $u \in \mathfrak{R}_+$. Furthermore, this collection of assortments includes $O(n)$ assortments. These results, together with Theorems 2 and 4, imply that we can solve a linear program with $1 + m$ decision variables and $O(mn)$ constraints to obtain the optimal assortment under parent product constraints. To characterize the optimal solution to problem (7) for any $u \in \mathfrak{R}_+$, we use the decision variables $x_i = (x_{i1}, \dots, x_{in}) \in \{0, 1\}^n$ to write problem (7) under parent product constraints as

$$\max \left\{ \sum_{j \in N} v_{ij} (r_{ij} - u) x_{ij} : x_{ik} \leq x_{ij} \forall j \in P_i, k \in C_{ij}, x_{ij} \in \{0, 1\} \forall j \in N \right\}. \quad (14)$$

Assume that parent product j is offered in the optimal solution to problem (14). In this case, we are free to offer any of the child products of parent product j . Since it is optimal to offer one of these child products when their objective function coefficient is positive, if parent product j is offered, then the total contribution of parent product j and all of its child products to the objective function of the problem above is given by $f_{ij}(u) = v_{ij} (r_{ij} - u) + \sum_{k \in C_{ij}} v_{ik} [r_{ik} - u]^+$, where we use $[\cdot]^+ = \max\{\cdot, 0\}$. On the other hand, if parent product j is not offered, then none of its child products can be offered, in which case, parent product j and all of its child products make a contribution of zero to the objective function of the problem above. Therefore, it is optimal to offer parent product j as long as $f_{ij}(u) > 0$. The function $f_{ij}(\cdot)$ is decreasing and piecewise linear with points of nondifferentiability occurring at $\{r_{ik} : k \in C_{ij}\}$. Thus, we can find a value of \bar{u}_{ij} such that $f_{ij}(u) > 0$ for any $u < \bar{u}_{ij}$ and $f_{ij}(u) \leq 0$ for any $u \geq \bar{u}_{ij}$, in which case, we offer parent product j in the optimal solution to the problem above when $u < \bar{u}_{ij}$ and we do not offer parent product j when $u \geq \bar{u}_{ij}$. If it is optimal not to offer parent product j , then none of its child products are offered, whereas if it is optimal to offer parent product j , then its child product k is offered when $u < r_{ik}$. Therefore, by comparing the value of u with \bar{u}_{ij} and $\{r_{ik} : k \in C_{ij}\}$, we can decide whether it is optimal to offer parent product j and any of its child products in the optimal solution to problem (14). Furthermore, it is straightforward to obtain the point \bar{u}_{ij} . Repeating the same reasoning for all of the parent products, we obtain the collections of points $\{\bar{u}_{ij} : j \in P_i\}$ and $\{r_{ik} : j \in P_i, k \in C_{ij}\}$. Since P_i and $\{C_{ij} : j \in P_i\}$ partition N , there are a total of n points in the collections $\{\bar{u}_{ij} : j \in P_i\}$ and $\{r_{ik} : j \in P_i, k \in C_{ij}\}$. These points completely characterize the optimal solution to problem (14) since we can compare u with \bar{u}_{ij} to decide whether it is optimal to offer parent product j . If this is the case, then we can decide whether it is optimal to offer its child product k by comparing u with r_{ik} .

Since there are a total of n points in the collections $\{\bar{u}_{ij} : j \in P_i\}$ and $\{r_{ik} : j \in P_i, k \in C_{ij}\}$, these points partition the positive real line into $O(n)$ intervals and we denote these intervals by $\{\mathcal{I}_i^t : t \in \mathcal{T}_i\}$ with $|\mathcal{T}_i| = O(n)$. We observe that as long as u takes values in one of the intervals $\{\mathcal{I}_i^t : t \in \mathcal{T}_i\}$, the ordering between u and any of the points in the collections $\{\bar{u}_{ij} : j \in P_i\}$ and $\{r_{ik} : j \in P_i, k \in C_{ij}\}$ does not change. This observation, in view of the discussion in the paragraph above, implies that the optimal solution to problem (14) does not change as long as u takes values in one of the intervals $\{\mathcal{I}_i^t : t \in \mathcal{T}_i\}$. Therefore, by comparing the value of u with $\{\bar{u}_{ij} : j \in P_i\}$ and $\{r_{ik} : j \in P_i, k \in C_{ij}\}$ in each one of the intervals $\{\mathcal{I}_i^t : t \in \mathcal{T}_i\}$, we can come up with a collection

of assortments $\{A_i^t : t \in \mathcal{T}_i\}$ with $|\mathcal{T}_i| = O(n)$ such that this collection always includes an optimal solution to problem (7) for any $u \in \mathfrak{R}_+$, as desired. In this case, Theorem 4 with $\alpha = 1$ implies that the best assortment $(\hat{S}_1, \dots, \hat{S}_m)$ with $\hat{S}_i \in \{A_i^t : t \in \mathcal{T}_i\}$ is the optimal solution to problem (1). By Theorem 2, we can find this best assortment by solving a linear program with $1 + m$ decision variables and $O(mn)$ constraints.

B.2 Cardinality and Space Constraints

In this section, we consider the case where we have both cardinality and space constraints on the assortment offered in each nest. In particular, if we use b_i to denote the limit on the cardinality of the assortment offered in nest i , c_i to denote the space availability in nest i and w_{ij} to denote the space requirement of product j in nest i , then the feasible assortments in nest i are given by $\mathcal{C}_i = \{S_i \in \{0, 1\}^n : \sum_{j \in N} S_{ij} \leq b_i, \sum_{j \in N} w_{ij} S_{ij} \leq c_i\}$. For this case, we begin by showing that we can come up with a collection of assortments $\{A_i^t : t \in \mathcal{T}_i\}$ with $|\mathcal{T}_i| = O(n^2)$ such that this collection includes a 3-approximate solution to problem (7) for any $u \in \mathfrak{R}_+$. Thus, we can solve a linear program with $1 + m$ decision variables and $O(mn^2)$ constraints to obtain a solution to the assortment optimization problem whose expected revenue deviates from the optimal expected revenue by no more than a factor of three. Later in this section, we refine our analysis to show that we can improve this performance guarantee from three to two.

The discussion in this section follows the development in Sections 4 and 5 closely. So, we mostly focus on the main points. Using the decision variables $x_i = (x_{i1}, \dots, x_{in}) \in \{0, 1\}^n$, we write problem (7) under cardinality and space constraints as

$$\max \left\{ \sum_{j \in N} v_{ij} (r_{ij} - u) x_{ij} : \sum_{j \in N} x_{ij} \leq b_i, \sum_{j \in N} w_{ij} x_{ij} \leq c_i, x_{ij} \in \{0, 1\} \forall j \in N \right\}, \quad (15)$$

which is a cardinality constrained knapsack problem. A basic exercise in duality theory shows that there are at most n^2 possible optimal bases to the linear programming relaxation of problem (15). Naturally, for any $u \in \mathfrak{R}_+$, the optimal solution to the linear programming relaxation of problem (15) must correspond to one of these optimal bases. In other words, for any $u \in \mathfrak{R}_+$, the optimal solution to the linear programming relaxation of problem (15) is one of n^2 solutions. By using the parametric simplex method over $u \in \mathfrak{R}_+$, we can generate all of these n^2 solutions. We use $\{x_i^g : g \in \mathcal{G}_i\}$ with $|\mathcal{G}_i| = O(n^2)$ to denote all possible solutions to the linear programming relaxation of problem (15). Thus, for any $u \in \mathfrak{R}_+$, there exists some x_i^g with $g \in \mathcal{G}_i$ such that x_i^g is the optimal solution to the linear programming relaxation of problem (15).

Implicitly treating the upper bounds $0 \leq x_{ij} \leq 1$ for all $j \in N$ in the linear programming relaxation of problem (15), we observe that there must be two basic decision variables in any basic optimal solution and the other decision variables have integer values. Thus, the solution x_i^g has at most two fractional components. Using the solution x_i^g , we define the assortment $S_i^g = (S_{i1}^g, \dots, S_{in}^g) \in \{0, 1\}^n$ such that $S_{ij}^g = \lfloor x_{ij}^g \rfloor$ for all $j \in N$. In other words, the assortment S_i^g

includes the products that take value one in the solution x_i^g . In this case, augmenting the collection of assortments $\{S_i^g : g \in \mathcal{G}_i\}$ with the collection of singleton assortments $\{\{j\} : j \in N\}$, it is possible to show that the collection of assortments $\{S_i^g : g \in \mathcal{G}_i\} \cup \{\{j\} : j \in N\}$ always includes a 3-approximate solution to problem (15) for any $u \in \mathfrak{R}_+$.

To see this result, assume that we solve problem (15) for some $u \in \mathfrak{R}_+$ and let g be such that x_i^g is the optimal solution to the linear programming relaxation of problem (15) when we solve this problem with the value of u in consideration. By the discussion at the beginning of the paragraph above, the solution x_i^g has at most two fractional components. We use j_1^g to denote the first fractional component of x_i^g when there is one. Similarly, we use j_2^g to denote the second fractional component of x_i^g when there is one. In this case, if we let $z^*(u)$ be the optimal objective value of problem (15), then noting that the optimal objective value of the linear programming relaxation provides an upper bound on $z^*(u)$, we obtain the chain of inequalities

$$\begin{aligned} z^*(u) &\leq \sum_{j \in N} v_{ij} (r_{ij} - u) x_{ij}^g \leq \sum_{j \in N} v_{ij} (r_{ij} - u) S_{ij}^g + v_{ij_1^g} (r_{ij_1^g} - u) + v_{ij_2^g} (r_{ij_2^g} - u) \\ &\leq 3 \max \left\{ \sum_{j \in N} v_{ij} (r_{ij} - u) S_{ij}^g, v_{ij_1^g} (r_{ij_1^g} - u), v_{ij_2^g} (r_{ij_2^g} - u) \right\}, \end{aligned} \quad (16)$$

where the second inequality follows from the fact that the assortment S_i^g , together with j_1^g and j_2^g , includes all components of the solution x_i^g that take strictly positive values. If the solution x_i^g has fewer than two fractional components, then the inequalities above continue to hold if we ignore the terms that involve j_1^g or j_2^g . From (16), we observe that either one of the assortments S_i^g , $\{j_1^g\}$ and $\{j_2^g\}$ is a 3-approximate solution to problem (15). Therefore, the collection of assortments $\{S_i^g : g \in \mathcal{G}_i\} \cup \{\{j\} : j \in N\}$ includes a 3-approximate solution to problem (15) for any $u \in \mathfrak{R}_+$ and there are $O(n^2)$ assortments in this collection, establishing the desired result.

We can tighten the approximation guarantee from three to two by using a somewhat more involved definition of the assortment S_i^g . If the solution x_i^g has zero or one fractional component, then we continue defining $S_i^g = (S_{i1}^g, \dots, S_{in}^g) \in \{0, 1\}^n$ such that $S_{ij}^g = \lfloor x_{ij}^g \rfloor$ for all $j \in N$. However, if the solution x_i^g has two fractional components, then using j_1^g and j_2^g to denote these fractional components with the convention that $w_{ij_1^g} \leq w_{ij_2^g}$, we define the assortment $S_i^g = (S_{i1}^g, \dots, S_{in}^g) \in \{0, 1\}^n$ as $S_{ij}^g = 1$ when $x_{ij}^g = 1$ or $j = j_1^g$, otherwise $S_{ij}^g = 0$. So, the assortment S_i^g includes all products that take value one and the product with the smaller space requirement that takes a fractional value in the solution x_i^g . In this case, we can show that the collection of assortments $\{S_i^g : g \in \mathcal{G}_i\} \cup \{\{j\} : j \in N\}$ includes a 2-approximate solution to problem (15) for any $u \in \mathfrak{R}_+$. To see this result, we can follow another basic exercise in duality theory to show that if there are two fractional components in a basic optimal solution to the linear programming relaxation of problem (15), then both constraints must be satisfied as equality. Thus, if x_i^g has two fractional components j_1^g and j_2^g , then $x_{ij_1^g}^g + x_{ij_2^g}^g = 1$. The last expression, together with the fact that $w_{ij_1^g} \leq w_{ij_2^g}$ and x_i^g is a feasible solution to the linear programming relaxation of problem (15), yields $c_i \geq \sum_{j \in N} w_{ij} x_{ij}^g = \sum_{j \in N} w_{ij} \mathbf{1}(x_{ij}^g = 1) + w_{ij_1^g} x_{ij_1^g}^g + w_{ij_2^g} x_{ij_2^g}^g \geq \sum_{j \in N} w_{ij} \mathbf{1}(x_{ij}^g = 1) + w_{ij_1^g}$. So,

the assortment S_i^g , which includes the products $\{j \in N : x_{ij}^g = 1\} \cup \{j_1^g\}$, is feasible to problem (15). In this case, we can use the same line of reasoning in (16) to get

$$\begin{aligned} z^*(u) &\leq \sum_{j \in N} v_{ij} (r_{ij} - u) x_{ij}^g \leq \sum_{j \in N} v_{ij} (r_{ij} - u) S_{ij}^g + v_{ij_2^g} (r_{ij_2^g} - u) \\ &\leq 2 \max \left\{ \sum_{j \in N} v_{ij} (r_{ij} - u) S_{ij}^g, v_{ij_2^g} (r_{ij_2^g} - u) \right\}, \end{aligned}$$

where the second inequality holds since the assortment S_i^g includes all strictly positive components of x_i^g except for j_2^g . The chain of inequalities above shows that either S_i^g or $\{j_2^g\}$ is a 2-approximate solution to problem (15), as desired.

C Online Supplement: Better Performance Guarantees under Space Constraints

In Section 5, we describe an approach to obtain a $\min\{2, 1/(1-\epsilon)\}$ -approximate solution to problem (1) under space constraints. The smallest possible value of ϵ that we can use in this performance guarantee is $\bar{\epsilon} = \max\{w_{ij}/c_i : i \in M, j \in N\}$, indicating that the best performance guarantee from the approach described in Section 5 is given by $\min\{2, 1/(1-\bar{\epsilon})\}$. In particular, even if we are willing to increase the computational effort, the approach described in Section 5 does not provide any guidance as to how we can improve this performance guarantee. In this section, our goal is to show how we can obtain better performance guarantees under space constraints as long as we are willing to increase the computational effort.

C.1 Improving the Performance Guarantee

The starting point for our discussion is problem (10), which is equivalent to problem (7) under space constraints. We recall that if we can come up with a collection of assortments $\{A_i^t : t \in \mathcal{T}_i\}$ such that this collection includes an α -approximate solution to problem (10) for any $u \in \mathfrak{R}_+$, then Theorem 4 implies that the best assortment $(\hat{S}_1, \dots, \hat{S}_m)$ with $\hat{S}_i \in \{A_i^t : t \in \mathcal{T}_i\}$ is an α -approximate solution to problem (1). Furthermore, by Theorem 2, we can find this best assortment by solving a linear program with $1 + m$ decision variables and $1 + \sum_{i \in M} |\mathcal{T}_i|$ constraints. In this section, we show that if we are given any $\alpha > 1$, then we can come up with a collection of assortments $\{A_i^t : t \in \mathcal{T}_i\}$ with $|\mathcal{T}_i| = O(\lceil \alpha/(\alpha-1) \rceil n^{\lceil \alpha/(\alpha-1) \rceil + 2})$ such that this collection always includes an α -approximate solution to problem (10) for any $u \in \mathfrak{R}_+$. In this case, by Theorem 4, the best assortment $(\hat{S}_1, \dots, \hat{S}_m)$ with $\hat{S}_i \in \{A_i^t : t \in \mathcal{T}_i\}$ is an α -approximate solution to problem (1). By Theorem 2, we can find this best assortment by solving a linear program with $1 + m$ decision variables and $O(m \lceil \alpha/(\alpha-1) \rceil n^{\lceil \alpha/(\alpha-1) \rceil + 2})$ constraints. Thus, by choosing α closer to one, we can obtain a performance guarantee that is closer to one as long as we are willing to increase the number of constraints in the linear program.

To characterize approximate solutions to problem (10), we use a special linear programming relaxation to this problem. Using the decision variables $x_i = (x_{i1}, \dots, x_{in}) \in [0, 1]^n$, for any given

$J \subset N$, we consider the problem

$$\max \left\{ \sum_{j \in N} v_{ij} (r_{ij} - u) x_{ij} : \sum_{j \in N} w_{ij} x_{ij} \leq c_i, \right. \\ \left. x_{ij} = 1 \forall j \in J, \quad 0 \leq x_{ik} \leq \mathbf{1} \left(v_{ik} (r_{ik} - u) \leq \min_{j \in J} \{v_{ij} (r_{ij} - u)\} \right) \forall k \in N \setminus J \right\}. \quad (17)$$

We can interpret the problem above as the linear programming relaxation of a knapsack problem after fixing the values of some of the decision variables at zero or one. In particular, we fix the values of the decision variables corresponding to the products in J at one. For the remaining decision variables, if the utility of a product corresponding to one of these decision variables exceeds the minimum of the utilities of the products in J , then we fix the value of this decision variable at zero. The role of the indicator function in the third set of constraints is to drop the products with utilities exceeding the minimum of the utilities of the products in J from consideration. Similar constraints appear in Frieze and Clarke (1984). The problem above may be infeasible for a certain J , in which case, we set the values of all decision variables to zero by convention. Problem (17) is the linear programming relaxation of a knapsack problem, where the utility of product j is $v_{ij} (r_{ij} - u)$ and the capacity consumption of product j is w_{ij} . So, we can solve this problem by using the following procedure. We put all of the products in J into the knapsack and drop these products from consideration. We order the other products with respect to their utilities. If there are any products whose utilities exceed the smallest of the utilities of the products in J , then we drop these products from consideration as well. Considering the remaining products, we fill the knapsack starting from the product with the largest utility to space consumption ratio, as long as the utility of the product exceeds zero. This procedure implies that the optimal solution to problem (17) does not change as long as the ordering of the utilities, ordering of the utility to space consumption ratios and signs of the utilities of the products do not change. Also, there is at most one fractional decision variable in the optimal solution to problem (17) obtained by using this procedure.

To exploit the fact that the optimal solution to problem (17) does not change as long as the ordering of the utilities, ordering of the utility to space consumption ratios and signs of the utilities of the products do not change, we define the linear functions $h_{ij}(u) = v_{ij} (r_{ij} - u)$ and $f_{ij}(u) = v_{ij} (r_{ij} - u)/w_{ij}$ for $j \in N$ and $h_{i0}(u) = 0$, $f_{i0}(u) = 0$. In this case, $h_{ij}(u)$ and $f_{ij}(u)$ respectively capture the utility and utility to space consumption ratio of product j in problem (17). We use $\{\bar{w}_i^g : g \in \mathcal{H}_i\}$ to denote the set of intersection points of the $n + 1$ linear functions $\{h_{ij}(\cdot) : j \in N \cup \{0\}\}$ and $\{\bar{u}_i^g : g \in \mathcal{H}_i\}$ to denote the set of intersection points of the $n + 1$ linear functions $\{f_{ij}(\cdot) : j \in N \cup \{0\}\}$. Thus, there are $|\mathcal{H}_i| = O(n^2)$ points in each one of these two sets of intersection points. Collecting the points in the two sets $\{\bar{w}_i^g : g \in \mathcal{H}_i\}$ and $\{\bar{u}_i^g : g \in \mathcal{H}_i\}$ together, we observe that the points $\{\bar{w}_i^g : g \in \mathcal{H}_i\} \cup \{\bar{u}_i^g : g \in \mathcal{H}_i\}$ partition the positive real line into $O(2|\mathcal{H}_i|) = O(n^2)$ intervals. We use $\{\mathcal{I}_i^g : g \in \mathcal{G}_i\}$ with $\mathcal{G}_i = O(n^2)$ to denote these intervals, in which case, the ordering of the utilities, ordering of the utility to space consumption ratios and signs of the utilities of the products in problem (17) do not change as long as u takes values in

one of these intervals. Since the optimal solution to problem (17) depends only on the ordering of the utilities, ordering of the utility to space consumption ratios and signs of the utilities of the products, the optimal solution to problem (17) does not change either when u takes values in one of these intervals. We use $x_i^g(J)$ to denote the optimal solution to problem (17) when u takes values in the interval \mathcal{I}_i^g . Our notation for $x_i^g(J)$ reflects the fact that the optimal solution to problem (17) depends on the choice of J .

Using the solution $x_i^g(J)$, we define the assortment $S_i^g(J) = (S_{i1}^g(J), \dots, S_{in}^g(J)) \in \{0, 1\}^n$ as $S_{ij}^g(J) = \lfloor x_{ij}^g(J) \rfloor$ for all $j \in N$. Therefore, the assortment $S_i^g(J)$ includes the products taking value one in the solution $x_i^g(J)$. In this case, using \wp_q to denote the set of subsets of N with cardinality not exceeding q , we propose using the collection of assortments $\{S_i^g(J) : J \in \wp_q, g \in \mathcal{G}_i\}$ as a collection of possibly good solutions to problem (10). Noting that $|\wp_q| = O(qn^q)$ and $|\mathcal{G}_i| = O(n^2)$, there are $O(qn^{q+2})$ assortments in the collection $\{S_i^g(J) : J \in \wp_q, g \in \mathcal{G}_i\}$, which can be manageable when q is not too large. The next lemma shows that this collection always includes a $q/(q-1)$ -approximate solution to problem (10) for any $u \in \mathfrak{R}_+$. To keep the focus on our main result, we defer the proof of this lemma to Section C.2.

Lemma 8 *Letting $S_i^g(J)$ be as given above, the collection of assortments $\{S_i^g(J) : J \in \wp_q, g \in \mathcal{G}_i\}$ includes a $q/(q-1)$ -approximate solution to problem (10) for any $u \in \mathfrak{R}_+$.*

For any desired performance guarantee $\alpha > 1$, setting $\alpha = q/(q-1)$ and solving for q , we obtain $q = \alpha/(\alpha-1)$. Thus, if we choose $q = \lceil \alpha/(\alpha-1) \rceil$ in the lemma above, then the collection of assortments $\{S_i^g(J) : J \in \wp_{\lceil \alpha/(\alpha-1) \rceil}, g \in \mathcal{G}_i\}$ includes an α -approximate solution to problem (10) for any $u \in \mathfrak{R}_+$. To come up with this collection of assortments, we compute the intervals $\{\mathcal{I}_i^g : g \in \mathcal{G}_i\}$ by finding the intersection points of the linear functions $\{h_{ij}(\cdot) : j \in N \cup \{0\}\}$ and $\{f_{ij}(\cdot) : j \in N \cup \{0\}\}$. In this case, the ordering of the utilities, ordering of the utility to space consumption ratios and signs of the utilities of the products in problem (17) do not change when u takes values in one of the intervals $\{\mathcal{I}_i^g : g \in \mathcal{G}_i\}$. Once these intervals are computed, we focus on each one of them one by one. For each interval \mathcal{I}_i^g and for each $J \in \wp_{\lceil \alpha/(\alpha-1) \rceil}$, we solve problem (17) to get the optimal solution $x_i^g(J)$ and define the assortment $S_i^g(J)$ as above. Since $|\wp_{\lceil \alpha/(\alpha-1) \rceil}| = O(\lceil \alpha/(\alpha-1) \rceil n^{\lceil \alpha/(\alpha-1) \rceil})$ and $|\mathcal{G}_i| = O(n^2)$, there are $O(\lceil \alpha/(\alpha-1) \rceil n^{\lceil \alpha/(\alpha-1) \rceil + 2})$ assortments in the collection $\{S_i^g(J) : J \in \wp_{\lceil \alpha/(\alpha-1) \rceil}, g \in \mathcal{G}_i\}$. The next theorem collects our observations.

Theorem 9 *Under space constraints, for any $\alpha > 1$, there exists a collection of assortments $\{A_i^t : t \in \mathcal{T}_i\}$ with $|\mathcal{T}_i| = O(\lceil \alpha/(\alpha-1) \rceil n^{\lceil \alpha/(\alpha-1) \rceil + 2})$ such that this collection includes an α -approximate solution to problem (7) for any $u \in \mathfrak{R}_+$.*

Thus, Theorem 4 implies that the best assortment $(\hat{S}_1, \dots, \hat{S}_m)$ with $\hat{S}_i \in \{A_i^t : t \in \mathcal{T}_i\}$ provides a performance guarantee of α for problem (1) under space constraints. Noting Theorem 2, this best assortment can be obtained by solving a linear program with $1 + m$ decision variables and

$O(m\lceil\alpha/(\alpha-1)\rceil n^{\lceil\alpha/(\alpha-1)\rceil+2})$ constraints. So, for any desired performance guarantee $\alpha > 1$, finding an assortment that provides this performance guarantee amounts to solving a linear program with $1 + m$ decision variables and $O(m\lceil\alpha/(\alpha-1)\rceil n^{\lceil\alpha/(\alpha-1)\rceil+2})$ constraints. This result demonstrates how we can improve the performance guarantee by increasing the number of constraints in the linear program. This approach naturally becomes computationally intractable when α gets too close to one, but if, for example, we want a performance guarantee of $\alpha = 3/2$, then the number of constraints we need comes out to be $O(mn^5)$.

The development in this section builds on Frieze and Clarke (1984), where the authors develop polynomial time approximation schemes for multi-dimensional knapsack problems. However, the focus of Frieze and Clarke (1984) is on solving a single instance of a multi-dimensional knapsack problem, but we are interested in finding good solutions to problem (10) for all $u \in \mathfrak{R}_+$.

C.2 Proof of Lemma 8

In this section, we give a proof for Lemma 8. Throughout this section, we denote an assortment offered in nest i by using a subset $S_i \subset N$. This is a slight deviation from our earlier notation where we use a vector $S_i \in \{0, 1\}^n$ to denote an assortment offered in nest i , but given a vector $S_i \in \{0, 1\}^n$, we can define the corresponding subset as $\{j \in N : S_{ij} = 1\}$. Using a subset $S_i \subset N$ to denote an assortment offered in nest i considerably simplifies our notation in this section.

Fixing u at an arbitrary $\hat{u} \in \mathfrak{R}_+$, we use x_i^* to denote the optimal solution to problem (10) when solved with $u = \hat{u}$. We have the assortment $S_i^* = \{j \in N : x_{ij}^* = 1\} \subset N$ corresponding to this optimal solution. Throughout the proof, we let \hat{g} be such that \hat{u} takes a value in the interval $\mathcal{I}_i^{\hat{g}}$, where the intervals $\{\mathcal{I}_i^g : g \in \mathcal{G}_i\}$ are as defined in Section C.1 of Online Supplement C. We begin by considering the case $|S_i^*| \leq q$ so that there are q or fewer products in the assortment S_i^* . If $|S_i^*| \leq q$, then we have $S_i^* \in \wp_q$. In this case, noting that $J \subset S_i^g(J)$ by the definitions of $x_i^g(J)$ and $S_i^g(J)$, we have $S_i^* \subset S_i^{\hat{g}}(S_i^*)$, which implies that $\sum_{j \in S_i^*} v_{ij} (r_{ij} - \hat{u}) \leq \sum_{j \in S_i^{\hat{g}}(S_i^*)} v_{ij} (r_{ij} - \hat{u})$. Thus, the assortment $S_i^{\hat{g}}(S_i^*)$ provides a better objective value for problem (10) than the assortment S_i^* when this problem is solved with $u = \hat{u}$. So, the assortment $S_i^{\hat{g}}(S_i^*)$ is also optimal to problem (10) when solved with $u = \hat{u}$. Also, since $S_i^* \in \wp_q$, we have $S_i^{\hat{g}}(S_i^*) \in \{S_i^g(J) : J \in \wp_q, g \in \mathcal{G}_i\}$, showing that the collection of assortments $\{S_i^g(J) : J \in \wp_q, g \in \mathcal{G}_i\}$ includes an optimal solution to problem (10) when this problem is solved with $u = \hat{u}$. Since \hat{u} is arbitrary, we are done.

In the rest of the proof, we assume that $|S_i^*| > q$. We let J_i^* be the subset of S_i^* that includes the q elements of S_i^* with the largest utilities in problem (10) when this problem is solved with $u = \hat{u}$. Since $|S_i^*| > q$, J_i^* is well defined. Consider the optimal solution $x_i^{\hat{g}}(J_i^*)$ to problem (17) when this problem is solved with $u = \hat{u}$ and $J = J_i^*$. If this solution has a fractional component j' , then the third set of constraints in problem (17) implies that product j' satisfies $v_{ij'} (r_{ij'} - \hat{u}) \leq v_{ij} (r_{ij} - \hat{u})$ for all $j \in J_i^*$. Thus, using $z^*(u)$ to denote the optimal objective value of problem (10), we have $z^*(\hat{u}) = \sum_{j \in S_i^*} v_{ij} (r_{ij} - \hat{u}) = \sum_{j \in J_i^*} v_{ij} (r_{ij} - \hat{u}) + \sum_{j \in S_i^* \setminus J_i^*} v_{ij} (r_{ij} - \hat{u}) \geq q v_{ij'} (r_{ij'} - \hat{u})$, where

the inequality follows by the fact that $|J_i^*| = q$ and $v_{ij'}(r_{ij'} - \hat{u}) \leq v_{ij}(r_{ij} - \hat{u})$ for all $j \in J_i^*$. Thus, the last chain of inequalities yields $v_{ij'}(r_{ij'} - \hat{u}) \leq z^*(\hat{u})/q$.

To finish the proof, let $z^*(u)$ and $\zeta^*(u, J)$ respectively be the optimal objective values of problems (10) and (17). We claim that $z^*(\hat{u}) \leq \zeta^*(\hat{u}, J_i^*)$. To see this claim, we note that J_i^* includes the q products with the largest utilities among the products taking value one in the optimal solution to problem (10) when this problem is solved with $u = \hat{u}$. The products in $S_i^* \setminus J_i^*$ also take value one in the optimal solution to problem (10), but by the definition of J_i^* , these products satisfy $v_{ik}(r_{ik} - \hat{u}) \leq \min_{j \in J_i^*} \{v_{ij}(r_{ij} - \hat{u})\}$ for all $k \in S_i^* \setminus J_i^*$. This inequality, together with the third set of constraints in problem (17), implies that if we solve problem (17) with $u = \hat{u}$ and $J = J_i^*$, then we fix the values of the decision variables corresponding to the products in J_i^* at one, but the values of the decision variables corresponding to the products in $S_i^* \setminus J_i^*$ are free between zero and one. Furthermore, problem (17) does not require the decision variables to take binary values. Thus, the solution S_i^* is optimal to problem (10) and feasible to problem (17) when we solve the first problem with $u = \hat{u}$ and the second problem with $u = \hat{u}$ and $J = J_i^*$. So, our claim holds and we have $z^*(\hat{u}) \leq \zeta^*(\hat{u}, J_i^*)$. In this case, if we use, as above, $x_i^{\hat{g}}(J_i^*)$ to denote the optimal solution to problem (17) when this problem is solved with $u = \hat{u}$ and $J = J_i^*$, then letting j' be the fractional component of $x_i^{\hat{g}}(J_i^*)$ when there is one, we obtain

$$\begin{aligned} z^*(\hat{u}) \leq \zeta^*(\hat{u}, J_i^*) &= \sum_{j \in N} v_{ij}(r_{ij} - \hat{u}) x_{ij}^{\hat{g}}(J_i^*) \\ &\leq \sum_{j \in S_i^{\hat{g}}(J_i^*)} v_{ij}(r_{ij} - \hat{u}) + v_{ij'}(r_{ij'} - \hat{u}) \leq \sum_{j \in S_i^{\hat{g}}(J_i^*)} v_{ij}(r_{ij} - \hat{u}) + z^*(\hat{u})/q, \end{aligned}$$

where the second inequality is by noting that $S_i^{\hat{g}}(J_i^*)$ includes all strictly positive and integer valued components of $x_i^{\hat{g}}(J_i^*)$ and the only possibly fractional component is j' and the third inequality is by the fact that $v_{ij'}(r_{ij'} - \hat{u}) \leq z^*(\hat{u})/q$, which is shown in the paragraph above. Focusing on the first and last terms in the chain of inequalities above, we have $z^*(\hat{u}) \leq (q/(q-1)) \sum_{j \in S_i^{\hat{g}}(J_i^*)} v_{ij}(r_{ij} - \hat{u})$, showing that the assortment $S_i^{\hat{g}}(J_i^*)$ is a $q/(q-1)$ -approximate solution to problem (10) when solved with $u = \hat{u}$. Since $|J_i^*| = q$, we have $S_i^{\hat{g}}(J_i^*) \in \{S_i^g(J) : J \in \wp_q, g \in \mathcal{G}_i\}$. Noting that the choice of \hat{u} is arbitrary, we conclude that the collection of assortments $\{S_i^g(J) : J \in \wp_q, g \in \mathcal{G}_i\}$ includes a $q/(q-1)$ -approximate solution to problem (10) for any $u \in \mathfrak{R}_+$, as desired.

D Online Supplement: Maximizing the Expected Revenue from a Single Nest

Problem (1) finds an assortment (S_1, \dots, S_m) that maximizes the expected revenue over all nests. In this section, we elaborate on the connections of this problem to the problem of maximizing the expected revenue only from a single nest. To that end, as a function of $u \in \mathfrak{R}_+$, we let $\hat{S}_i(u)$ be an optimal solution to the problem

$$\max_{S_i \in \mathcal{C}_i} \left\{ V_i(S_i)(R_i(S_i) - u) \right\} = \max_{S_i \in \mathcal{C}_i} \left\{ \sum_{j \in N} v_{ij}(r_{ij} - u) S_{ij} \right\}, \quad (18)$$

where the equality follows from (8). Applying Theorem 4 with $\alpha = 1$ implies that if we obtain an optimal solution to the problem above for all $u \in \mathfrak{R}_+$ and use $\{\hat{S}_i(u) : u \in \mathfrak{R}_+\}$ as a collection of candidate assortments for nest i , then the best assortment $(\hat{S}_1, \dots, \hat{S}_m)$ with $\hat{S}_i \in \{\hat{S}_i(u) : u \in \mathfrak{R}_+\}$ is an optimal solution to problem (1). Therefore, if we use $\{\hat{S}_i(u) : u \in \mathfrak{R}_+\}$ for all $i \in M$ as collections of candidate assortments for the different nests, then we can stitch together the optimal solution to problem (1) by using an assortment from each one of these collections. We proceed to showing that the collection $\{\hat{S}_i(u) : u \in \mathfrak{R}_+\}$ does not only allow us to stitch together an assortment that maximizes the expected revenue over all nests, but it also includes an assortment solving the problem $\max_{S_i \in \mathcal{C}_i} R_i(S_i)$. In other words, the collection of assortments $\{\hat{S}_i(u) : u \in \mathfrak{R}_+\}$ also includes an assortment that maximizes the expected revenue only from nest i .

To see that the collection of assortments $\{\hat{S}_i(u) : u \in \mathfrak{R}_+\}$ includes an optimal solution to the problem $\max_{S_i \in \mathcal{C}_i} R_i(S_i)$, we let $z^* = \max_{S_i \in \mathcal{C}_i} R_i(S_i)$. So, we have $R_i(S_i) \leq z^*$ for all $S_i \in \mathcal{C}_i$ and the inequality holds as equality at the assortment that maximizes the expected revenue from nest i . Noting that $R_i(S_i) = \sum_{j \in N} v_{ij} r_{ij} S_{ij} / \sum_{j \in N} v_{ij} S_{ij}$, the inequality $R_i(S_i) \leq z^*$ can be written as $\sum_{j \in N} v_{ij} (r_{ij} - z^*) S_{ij} \leq 0$. In this case, we have $\sum_{j \in N} v_{ij} (r_{ij} - z^*) S_{ij} \leq 0$ for all $S_i \in \mathcal{C}_i$ and the inequality holds as equality at the assortment that maximizes the expected revenue from nest i . Therefore, an optimal solution to the problem $\max_{S_i \in \mathcal{C}_i} \sum_{j \in N} v_{ij} (r_{ij} - z^*) S_{ij}$ is also an optimal solution to the problem $\max_{S_i \in \mathcal{C}_i} R_i(S_i)$. Since an optimal solution to the problem $\max_{S_i \in \mathcal{C}_i} \sum_{j \in N} v_{ij} (r_{ij} - z^*) S_{ij}$ is given by $\hat{S}_i(z^*)$, it follows that $\hat{S}_i(z^*)$ is an optimal solution to the problem $\max_{S_i \in \mathcal{C}_i} R_i(S_i)$. Noting that $\hat{S}_i(z^*) \in \{\hat{S}_i(u) : u \in \mathfrak{R}_+\}$, the collection of assortments $\{\hat{S}_i(u) : u \in \mathfrak{R}_+\}$ indeed includes an optimal solution to the problem $\max_{S_i \in \mathcal{C}_i} R_i(S_i)$.

We can construct the collection of assortments $\{\hat{S}_i(u) : u \in \mathfrak{R}_+\}$ by obtaining an optimal solution to problem (18) for all $u \in \mathfrak{R}_+$. In this case, if we use $\{\hat{S}_i(u) : u \in \mathfrak{R}_+\}$ for all $i \in M$ as collections of candidate assortments for the different nests, then the best assortment $(\hat{S}_1, \dots, \hat{S}_m)$ with $\hat{S}_i \in \{\hat{S}_i(u) : u \in \mathfrak{R}_+\}$ is an optimal solution to problem (1). Furthermore, the collection of assortments $\{\hat{S}_i(u) : u \in \mathfrak{R}_+\}$ also includes an optimal solution to the problem $\max_{S_i \in \mathcal{C}_i} R_i(S_i)$ and if we want to maximize the expected revenue only from nest i , then it is enough to focus on the assortments in the collection $\{\hat{S}_i(u) : u \in \mathfrak{R}_+\}$. Rusmevichientong, Shen and Shmoys (2010) consider the problem of maximizing the expected revenue under cardinality constraints when customers choose according to the multinomial logit model. Since the multinomial logit model can be viewed as the nested logit model with a single nest, their problem is similar to the problem $\max_{S_i \in \mathcal{C}_i} R_i(S_i)$ with \mathcal{C}_i corresponding to a cardinality constraint. As a result, they construct their collection of candidate assortments by solving a problem similar to problem (18).

The discussion in this section shows that we can use the collection $\{\hat{S}_i(u) : u \in \mathfrak{R}_+\}$ as candidate assortments for nest i both to maximize the expected revenue over all nests and to maximize the expected revenue only from nest i . However, the particular one of the assortments in this collection that we end up using to maximize the expected revenue over all nests can be entirely different from the particular one that we end up using to maximize the expected revenue only from nest

i. Indeed, it is possible to generate examples such that if we solve the problem $\max_{S_i \in \mathcal{C}_i} R_i(S_i)$ to find an assortment \tilde{S}_i that maximizes the expected revenue only from nest i and use the assortment $(\tilde{S}_1, \dots, \tilde{S}_m)$ as a possible solution to problem (1), then the assortment $(\tilde{S}_1, \dots, \tilde{S}_m)$ performs arbitrarily poorly for problem (1).