

Pricing Problems under the Nested Logit Model with a Quality Consistency Constraint

James M. Davis, Huseyin Topaloglu, David P. Williamson¹

April 28, 2015

Abstract

We consider pricing problems when customers choose among the products according to the nested logit model and there is a quality consistency constraint on the prices charged for the products. We consider two types of quality consistency constraints. In the first type of constraint, there is an inherent ordering between the qualities of the products in a particular nest and the price for the product of a higher quality level should be larger. In the second type of constraint, different nests correspond to different quality levels and the price of any product that is in a nest corresponding to a higher quality level should be larger than the price of any product that is in a nest corresponding to a lower quality level. The prices of the products are chosen within a finite set of possible prices. We develop algorithms to find the prices to charge for the products to maximize the expected revenue obtained from a customer, while adhering to a quality consistency constraint. Our algorithms are based on solving linear programs whose sizes scale polynomially with the number of nests, number of products and number of possible prices for the products. Numerical experiments indicate that our algorithms can effectively compute the optimal prices even when there is a large number of products in consideration.

¹Authors are affiliated with School of Operations Research and Information Engineering, Cornell University, Ithaca, New York 14853, USA. Their email addresses are jmd388@cornell.edu, topaloglu@orie.cornell.edu and dpw@cs.cornell.edu.

In many retail environments, there are multiple substitutable products that can serve the needs of a customer and customers make a choice among the available products by comparing them with respect to attributes such as price, quality, richness of features and durability. When such substitution possibilities are present, the demand for a product depends not only on its own attributes, but also on the attributes of the other products, creating interactions between the demands for different products. Discrete choice models become useful to capture such demand interactions, since they represent the demand for a particular product as a joint function of the attributes of all available products. Capturing the interactions between the demands for the products has recently become more important than ever, as online retailers and travel agencies bring a large variety of options to customers. Nevertheless, optimization models that try to find the right prices to charge for the products quickly become complicated when one uses sophisticated choice models to capture the interaction between the demands for the products. These optimization models become even more complicated when one tries to impose operational constraints on the prices charged for the products.

In this paper, we consider pricing problems when customers choose according to the nested logit model and there is a quality consistency constraint on the prices charged for the products. Under the nested logit model, the products are grouped into nests. The choice process of the customer proceeds in two stages. In the first stage, the customer decides either to make a purchase in one of the nests or to leave the system without making a purchase. In the second stage, if the customer decides to make a purchase in one of the nests, then the customer chooses one of the products in the chosen nest. This choice process is shown in Figure 1.a. The customer starts from the root node of the tree. In the first stage, she chooses one of the nests or the no purchase option. In the second stage, if she has chosen one of the nests in the first stage, then she selects one of the products in the chosen nest. In the quality consistency constraint that we impose on the prices, there is an intrinsic ordering between the qualities of the products. The quality consistency constraint ensures that the prices charged for the products of higher quality are also larger. The goal is to find the prices to charge for the products to maximize the expected revenue obtained from a customer, while making sure that the prices satisfy the quality consistency constraint.

We consider two types of quality consistency constraint. In the first type of constraint, there is an intrinsic ordering between the qualities of the products in each nest. We refer to this quality consistency constraint as *price ladders inside nests*. Figure 1.b illustrates this quality consistency constraint with three products in each nest and the price of product j in nest i is denoted by p_{ij} . The products in each nest are indexed such that the third product is of higher quality than the second product in the same nest, which is, in turn, of higher quality than the first product. Therefore, the price of the third product should be larger than the price of the second product, which should, in turn, be larger than the price of the first product. There is no dictated ordering between the qualities or prices of the products in different nest. In the second type of constraint, there is an intrinsic ordering between the qualities of the nests, but there is no clear ordering between the qualities of the products in the same nest. We refer to this quality consistency constraint as *price*

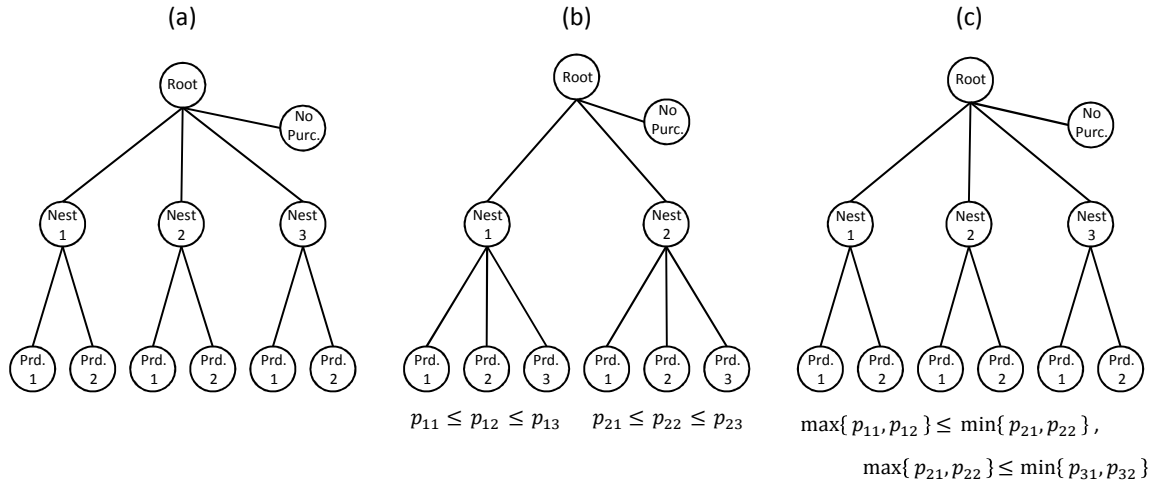


Figure 1: Nested logit model, price ladders inside nests and price ladders between nests.

ladders between nests. Figure 1.c illustrates this quality consistency constraint with three nests. The nests are indexed such that the third nest corresponds to a higher quality level than the second nest, which, in turn, corresponds to a higher quality level than the first nest. So, the price for any product in the third nest should be larger than the price for any product in the second nest, which should, in turn, be larger than the price for any product in the first nest. At the end of this section, we discuss practical situations where such price constraints naturally arise.

Main Results and Contributions. In this paper, we give algorithms to find the optimal prices to charge under both types of quality consistency constraints. In our setting, there are m nests and n products in each nest. The price of each product is chosen within a finite set of possible prices and there are q possible prices for each product. Therefore, the vector of prices charged for the products in a nest takes values in \mathfrak{R}^n and each component of this vector takes one of the q possible values, which implies that there are $O(q^n)$ possible price vectors that we can charge for the products in a nest. Under price ladders within nests, we show that the optimal price vector to charge in a nest is one of at most nq candidate price vectors and all of these candidate price vectors can be constructed by solving a linear program through the parametric simplex method. The linear program that we use to come up with the candidate price vectors has $O(nq)$ decision variables and $O(nq^2)$ constraints. This result reduces the number of possible price vectors to consider for each nest from $O(q^n)$ to $O(nq)$. However, although the optimal price vector to charge in each nest is one of $O(nq)$ candidate price vectors, computing the optimal prices to charge over all nests can still be challenging, since there are $O((nq)^m)$ different ways of combining nq candidate price vectors from m nests. To deal with this difficulty, we give a linear program with $O(m)$ decision variables and $O(mnq)$ constraints that finds the optimal combination of price vectors to charge in different nests. Thus, we solve a linear program with $O(nq)$ decision variables and $O(nq^2)$ constraints by

using the parametric simplex method to come up with $O(nq)$ candidate price vectors for each nest. We find the optimal combination of candidate price vectors to charge in different nests by solving another linear program with $O(m)$ decision variables and $O(mnq)$ constraints.

Pricing problems under price ladders between nests are considerably more difficult than the ones under price ladders inside nests since price ladders between nests create interactions between the prices charged for the products in different nests. Under price ladders between nests, we show that the optimal price vector to charge in a nest is one of at most nq^3 candidate price vectors and all of these candidate price vectors can be constructed by solving a linear program through the parametric simplex method. The linear program that we use to come up with the candidate price vectors has $O(nq)$ decision variables and $O(n)$ constraints. To find the optimal combination of price vectors to charge in different nests, we give a linear program with $O(mq)$ decision variables and $O(mnq^4)$ constraints. In our numerical experiments, we consider test problems with as many as $m = 6$ nests, $n = 30$ products in each nest and $q = 30$ possible prices for each product, yielding a total of 180 products. Under price ladders inside nests, we can compute the optimal prices to charge for the products in about two seconds, whereas under price ladders between nests, we can compute the optimal prices in about two minutes.

In addition to providing solution algorithms that find the optimal prices under price ladders inside nests and between nests, we make practically useful contributions through our formulation of the pricing problem. In our formulation, the price of each product is chosen within a finite set of possible prices and the set of possible prices for a product is defined by the modeler. The modeler can design the set of possible prices for a product to correspond to the prices that are commonly used in retail, such as prices that end in 99 cents or prices that are in increments of 10 dollars. Furthermore, the nested logit model commonly assumes that there is a parametric relationship between the attractiveness of a product and its price. For example, it is common to assume that if the price charged for product j in nest i is p_{ij} , then the attractiveness of this product is given by $\exp(\alpha_{ij} - \beta_{ij} p_{ij})$, where α_{ij} and β_{ij} are fixed parameters; see Li and Huh (2011) and Gallego and Wang (2014). Our formulation of the pricing problem does not assume a parametric relationship between the attractiveness of a product and its price, allowing the attractiveness of a product to depend on its price in an arbitrary fashion.

Related Literature. There is a significant amount of work on solving pricing problems under variants of the multinomial and nested logit models. Under the multinomial logit model, Hanson and Martin (1996) observe that the expected revenue is not a concave function of the prices for the products. Song and Xue (2007) and Dong et al. (2009) solve the pricing problem by expressing the expected revenue as a function of the market shares of the products and showing that the expected revenue is a concave function of the market shares. Chen and Hausman (2000) and Wang (2012) give tractable solution methods for joint assortment optimization and pricing problems under the multinomial logit model, where the set of products offered to customers, as well as the prices of the offered products, are decision variables. Zhang and Lu (2013) discuss revenue

management problems, where the prices charged for the products are dynamically adjusted over time as a function of the remaining time in the selling horizon and the remaining inventory for the products. Davis et al. (2013) show that pricing problems under the multinomial logit model with a finite set of possible prices can be formulated as a linear program. Keller et al. (2014) study pricing problems, where the attractiveness of a product depends on its price in a general fashion and there are constraints on the expected number of sales for the products.

The literature on solving pricing problems under the nested logit model has recently started growing. Li and Huh (2011) study pricing problems under the assumption that the products in the same nest have the same price sensitivity. They show that if the products in the same nest have the same price sensitivity, then the pricing problem can be formulated as a convex program. Gallego and Wang (2014) consider the case where the products in the same nest do not necessarily have the same price sensitivity. They show that the expected revenue function can have multiple local maxima and show how to find a local maximum of the expected revenue function. The authors also give sufficient conditions that eliminate multiple local maxima. Rayfield et al. (2013) develop algorithms for computing solutions with a specified performance guarantee even when there are multiple local maxima of the expected revenue function. Li and Huh (2013) and Li et al. (2014) consider pricing problems under the nested logit model, where the choice process proceeds in more than two stages. The earlier work on pricing problems under the nested logit model does not consider quality consistency constraints.

Charging quality consistent prices is practically important since such prices convey a sense of fairness to customers. Rusmevchientong et al. (2006) consider quality consistent pricing problems. Their work is motivated by a pricing problem in the automobile industry, where the prices of the automobiles with richer features should also be larger. They use a nonparametric choice model, show that the corresponding pricing problem with a quality consistency constraint is NP hard and provide an approximation algorithm. Gallego and Stefanescu (2009) discuss fairness issues when providing upgrades to customers. In the context of airline industry, they point out that if the customers need to be upgraded, then the customers with low fare class reservations should be upgraded to a relatively lower fare class, when compared with the customers with high fare class reservations. They develop fluid models for revenue management problems with upgrade possibilities. They show that if the prices satisfy a certain quality consistency constraint, then their model upgrades customers with low fare class reservations to a relatively lower fare class, when compared with the customers with high fare class reservations.

We work with two different types of quality consistency constraint. In the first type of quality consistency constraint, there is an intrinsic ordering between the qualities of the products in the same nest and the prices for the products of higher quality should also be larger. As an example of a situation where this type of quality consistency constraint becomes relevant, we consider the case where the nests correspond to different brands and the products within a particular nest correspond to the variants of a particular brand with different qualities. There is a verifiable

ordering between the qualities of the different variants of a particular brand and the customers expect that the prices for the variants of higher quality should also be larger. On the other hand, it is difficult to compare the variants of different brands in terms of quality and there is no reason for the customers to expect a particular ordering between the prices for the variants of different brands. In the second type of quality consistency constraint, different nests correspond to different quality levels and there is an intrinsic ordering between the quality levels of the nests. The price for any product in a nest corresponding to a higher quality level should be larger than the price for any product in a nest corresponding to a lower quality level. As an example of a situation where this type of quality consistency constraint becomes relevant, we consider the case where the nests correspond to different quality levels and the products within a particular nest correspond to products that differ in cosmetic or personal features, such as color. The customers expect that the prices for the products in a nest corresponding to a higher quality level are larger than the prices for the products in a nest corresponding to a lower quality level, but there is no reason for the customers to expect a particular ordering between the prices for the products in a particular nest since these products differ in cosmetic or personal features. Although the two types of quality consistency constraints cover a variety of useful situations, we observe that there can be other quality consistency constraints that are not covered by our results. In our conclusions, we discuss some possible extensions of our quality consistency constraints.

A useful approach for solving optimization problems under the nested logit model is to construct a small collection of candidate solutions for each nest and to solve a linear program to combine the candidate solutions for the different nests. Gallego and Topaloglu (2014) and Feldman and Topaloglu (2014) follow this approach for assortment optimization problems, where the prices of the products are fixed and the goal is to find a set of products to offer to customers to maximize the expected revenue obtained from a customer. The development in this paper is based on this general approach as well, but there are two important challenges that need to be overcome when using this approach for pricing problems. First, constructing a small collection of candidate price vectors to charge in each nest carefully exploits the structure of the pricing problem. In particular, we make use of the fact that the attractiveness of a product is decreasing in its price and it is not clear how to construct a small collection of candidate price vectors when the attractiveness of a product is not necessarily decreasing in its price. Second, under price ladders between nests, the prices charged in different nests interact with each other since the price for any product in a nest corresponding to a higher quality level should be larger than the price for any product in a nest corresponding to a lower quality level. Due to the interactions between the prices charged in different nests, finding the optimal combination of price vectors to choose in each nest becomes difficult. We address this difficulty by using the linear programming formulation of a dynamic program that finds the best combination of the candidate price vectors for the different nests.

The rest of the paper is organized as follows. In Section 1, we study the pricing problem under price ladders inside nests. In Section 2, we study the pricing problem under price ladders between nests. In Section 3, we provide numerical experiments. In Section 4, we conclude.

1 Price Ladders Inside Nests

In this section, we consider the case with price ladders inside nests. In this setting, there is an intrinsic ordering between the qualities of the products in the same nest and the prices for the products of higher quality should also be larger. There is no intrinsic ordering between the qualities or the prices of the products in different nests.

1.1 Problem Formulation

There are m nests and we index the nests by $M = \{1, \dots, m\}$. In each nest, there are n products and we index the products in each nest by $N = \{1, \dots, n\}$. For each product, there are q possible prices. The possible prices for a product are given by $\Theta = \{\theta^1, \dots, \theta^q\}$. Without loss of generality, we index the possible prices so that $0 < \theta^1 < \theta^2 < \dots < \theta^q$. We use $p_{ij} \in \Theta$ to denote the price that we charge for product j in nest i . If we charge price p_{ij} for product j in nest i , then the preference weight of this product is given by $v_{ij}(p_{ij})$. If we charge a larger price for a product, then its preference weight becomes smaller, implying that $v_{ij}(\theta^1) > v_{ij}(\theta^2) > \dots > v_{ij}(\theta^q) > 0$. Our notation so far implies that the number of products in each nest is the same and the set of possible prices that we can charge for each product is the same. However, this assumption is only for notational brevity and our results in the paper continue to hold with straightforward modifications when there are different numbers of products in different nests and the sets of possible prices for the different products are different.

We use $p_i = (p_{i1}, \dots, p_{in}) \in \Theta^n$ to capture the price vector charged in nest i . As a function of the price vector p_i charged in nest i , we use $V_i(p_i)$ to denote the total preference weight of the products in nest i , so that $V_i(p_i) = \sum_{j \in N} v_{ij}(p_{ij})$. Under the nested logit model, if we charge the price vector p_i in nest i , then a customer that has already decided to make a purchase in nest i chooses product j in this nest with probability $v_{ij}(p_{ij})/V_i(p_i)$. In this case, if we charge the price vector p_i in nest i and a customer has already decided to make a purchase in this nest, then the expected revenue obtained from this customer is given by

$$R_i(p_i) = \sum_{j \in N} p_{ij} \frac{v_{ij}(p_{ij})}{V_i(p_i)} = \frac{\sum_{j \in N} p_{ij} v_{ij}(p_{ij})}{V_i(p_i)}. \quad (1)$$

For each nest i , the nested logit model has a parameter $\gamma_i \in [0, 1]$ characterizing the degree of dissimilarity between the products in this nest. We use v_0 to denote the preference weight of the no purchase option. Under the nested logit model, if we charge the price vectors $(p_1, \dots, p_m) \in \Theta^{m \times n}$ over all nests, then a customer decides to make a purchase in nest i with probability $Q_i(p_1, \dots, p_m) = V_i(p_i)^{\gamma_i} / (v_0 + \sum_{l \in M} V_l(p_l)^{\gamma_l})$. The last expression provides the probability that a customer chooses nest i as a function of the prices charged for all products in all nests. The parameter γ_i magnifies or dampens the preference weights of the products in nest i .

According to the nested logit model, if we charge the price vectors (p_1, \dots, p_m) over all nests, then a customer decides to make a purchase in nest i with probability $Q_i(p_1, \dots, p_m) =$

$V_i(p_i)^{\gamma_i}/(v_0 + \sum_{l \in M} V_l(p_l)^{\gamma_l})$. If the customer decides to make a purchase in nest i , then the expected revenue obtained from this customer is $R_i(p_i)$. Thus, if we charge the price vectors (p_1, \dots, p_m) over all nests, then the expected revenue from a customer is given by

$$\Pi(p_1, \dots, p_m) = \sum_{i \in M} Q_i(p_1, \dots, p_m) R_i(p_i) = \frac{\sum_{i \in M} V_i(p_i)^{\gamma_i} R_i(p_i)}{v_0 + \sum_{i \in M} V_i(p_i)^{\gamma_i}}. \quad (2)$$

Our goal is to find the price vectors (p_1, \dots, p_m) to charge over all nests to maximize the expected revenue above subject to the constraint that the price vector charged in each nest satisfies a price ladder constraint. To formulate the price ladder constraint, without loss of generality, we index the products in each nest such that products with larger indices are of higher quality. In other words, the products $N = \{1, \dots, n\}$ in each nest are indexed in the order of increasing quality. The price ladder constraint ensures that the price for a product of higher quality is larger. That is, the price ladder constraint in nest i ensures that $p_{i1} \leq p_{i2} \leq \dots \leq p_{in}$. Thus, the set of feasible price vectors in nest i can be written as $\mathcal{F}_i = \{p_i \in \Theta^n : p_{ij} \geq p_{i,j-1} \forall j \in N \setminus \{1\}\}$. We want to find the price vectors to charge over all nests to maximize the expected revenue from a customer while satisfying the price ladder constraint, yielding the problem

$$z^* = \max_{\substack{(p_1, \dots, p_m) \in \Theta^{m \times n} \\ p_i \in \mathcal{F}_i \forall i \in M}} \left\{ \Pi(p_1, \dots, p_m) \right\}. \quad (3)$$

In the problem above, the price of each product takes values in the discrete set Θ . Furthermore, the objective function depends on the prices of the products in a nonlinear fashion. Thus, this problem is a nonlinear combinatorial optimization problem.

We emphasize two useful advantages of our formulation of problem (3). First, since the price for each product is chosen among a set of possible prices given by Θ and we can design Θ in any way we want, our formulation allows choosing the prices of the products among the prices that are commonly used in retail, such as prices that end in 99 cents or prices that are in increments of 10 dollars. Second, the nested logit model commonly assumes a fixed functional relationship between the price of a product and its preference weight. For example, as a function of the price p_{ij} of product j in nest i , it is common to assume that the preference weight $v_{ij}(p_{ij})$ of this product is given by $v_{ij}(p_{ij}) = \exp(\alpha_{ij} - \beta_{ij} p_{ij})$, where α_{ij} and β_{ij} are fixed parameters. In contrast, our formulation of problem (3) does not rely on such a fixed functional relationship and we allow the dependence between $v_{ij}(p_{ij})$ and p_{ij} to be arbitrary, as long as $v_{ij}(p_{ij})$ is decreasing in p_{ij} .

1.2 Connection to a Fixed Point Representation

In this section, we answer a question that becomes critical when developing a tractable solution approach for problem (3). Assume that we have a collection of candidate price vectors $\mathcal{P}_i = \{p_i^t : t \in \mathcal{T}_i\}$ to charge in nest i and all of the price vectors in the collection \mathcal{P}_i satisfy the price ladder constraint in the sense that $p_i^t \in \mathcal{F}_i$ for all $t \in \mathcal{T}_i$. We know that we can stitch together an optimal solution to problem (3) by picking one price vector from each one of the candidate

collections $\mathcal{P}_1, \dots, \mathcal{P}_m$. In other words, we know that there exists an optimal solution (p_1^*, \dots, p_m^*) to problem (3) that satisfies $p_i^* \in \mathcal{P}_i$ for all $i \in M$. The question that we want to answer is how we can pick a price vector p_i^* from the collection \mathcal{P}_i for each nest i such that the solution (p_1^*, \dots, p_m^*) is indeed optimal to problem (3). It is difficult to answer this question through complete enumeration since complete enumeration requires checking the expected revenues from $|\mathcal{P}_1| \times \dots \times |\mathcal{P}_m|$ possible solutions, which quickly gets intractable when the number of nests is large. To answer this question, we relate problem (3) to the problem of computing the fixed point of an appropriately defined function. In particular, for $z \in \mathfrak{R}_+$, we define $f(z)$ as

$$f(z) = \sum_{i \in M} \max_{p_i \in \mathcal{P}_i} \left\{ V_i(p_i)^{\gamma_i} (R_i(p_i) - z) \right\}. \quad (4)$$

The value of \hat{z} satisfying $v_0 \hat{z} = f(\hat{z})$ is the fixed point of the function $f(\cdot)/v_0$. Since $v_0 z$ is increasing and $f(z)$ is decreasing in z with $f(0) \geq 0$, there exists \hat{z} satisfying $v_0 \hat{z} = f(\hat{z})$. In the next theorem, we show that we can use this value of \hat{z} to construct an optimal solution to problem (3). In this theorem, we recall that z^* corresponds to the optimal objective value of problem (3).

Theorem 1 *Assume that we have a collection of candidate price vectors \mathcal{P}_i for each nest i such that we can stitch together an optimal solution to problem (3) by picking one price vector from each one of the collections $\mathcal{P}_1, \dots, \mathcal{P}_m$. Let the value of \hat{z} be such that $v_0 \hat{z} = f(\hat{z})$ and \hat{p}_i be an optimal solution to the problem*

$$\max_{p_i \in \mathcal{P}_i} \left\{ V_i(p_i)^{\gamma_i} (R_i(p_i) - \hat{z}) \right\}. \quad (5)$$

Then, we have $\Pi(\hat{p}_1, \dots, \hat{p}_m) \geq z^$.*

Proof. We use (p_1^*, \dots, p_m^*) to denote an optimal solution to problem (3). By our assumption, we can stitch together an optimal solution to problem (3) by picking one price vector from each one of the collections $\mathcal{P}_1, \dots, \mathcal{P}_m$. Thus, we can assume that $p_i^* \in \mathcal{P}_i$ for all $i \in M$, which implies that solution p_i^* is feasible to the problem on the right side of (4) and we get $f(\hat{z}) \geq \sum_{i \in M} V_i(p_i^*)^{\gamma_i} (R_i(p_i^*) - \hat{z})$. In this case, noting the fact that $v_0 \hat{z} = f(\hat{z})$, we have $v_0 \hat{z} \geq \sum_{i \in M} V_i(p_i^*)^{\gamma_i} (R_i(p_i^*) - \hat{z})$. Solving for \hat{z} in the last inequality, we obtain $\hat{z} \geq \sum_{i \in M} V_i(p_i^*)^{\gamma_i} R_i(p_i^*) / (v_0 + \sum_{i \in M} V_i(p_i^*)^{\gamma_i})$. Noting that $z^* = \Pi(p_1^*, \dots, p_m^*) = \sum_{i \in M} V_i(p_i^*)^{\gamma_i} R_i(p_i^*) / (v_0 + \sum_{i \in M} V_i(p_i^*)^{\gamma_i})$ by the definition of $\Pi(p_1, \dots, p_m)$ in (2), the last inequality implies that $\hat{z} \geq z^*$. Thus, to finish the proof, it is enough to show that $\Pi(\hat{p}_1, \dots, \hat{p}_m) = \hat{z}$. Since \hat{p}_i is an optimal solution to problem (5), by the definition of $f(z)$ in (4) and the fact that $v_0 \hat{z} = f(\hat{z})$, we have $v_0 \hat{z} = f(\hat{z}) = \sum_{i \in M} V_i(\hat{p}_i)^{\gamma_i} (R_i(\hat{p}_i) - \hat{z})$. In this case, focusing on the first and last expressions in the last chain of equalities and solving for \hat{z} , we obtain $\hat{z} = \sum_{i \in M} V_i(\hat{p}_i)^{\gamma_i} R_i(\hat{p}_i) / (v_0 + \sum_{i \in M} V_i(\hat{p}_i)^{\gamma_i})$ and the desired result follows by noting that $\Pi(\hat{p}_1, \dots, \hat{p}_m) = \sum_{i \in M} V_i(\hat{p}_i)^{\gamma_i} R_i(\hat{p}_i) / (v_0 + \sum_{i \in M} V_i(\hat{p}_i)^{\gamma_i})$. \square

Theorem 1 suggests the following approach to obtain an optimal solution to problem (3). Assume that we have a collection of candidate price vectors $\mathcal{P}_i = \{p_i^t : t \in \mathcal{T}_i\}$ for each nest i such that

we can stitch together an optimal solution to problem (3) by picking one price vector from each one of the collections $\mathcal{P}_1, \dots, \mathcal{P}_m$. Furthermore, assume that each one of the price vectors in the candidate collection $\mathcal{P}_i = \{p_i^t : t \in \mathcal{T}_i\}$ satisfies the price ladder constraint in the sense that $p_i^t \in \mathcal{F}_i$ for all $p_i^t \in \mathcal{P}_i$. To obtain an optimal solution to problem (3), we find the value of \hat{z} that satisfies $v_0 \hat{z} = f(\hat{z})$. In this case, if we let \hat{p}_i be an optimal solution to problem (5), then it follows from Theorem 1 that $\Pi(\hat{p}_1, \dots, \hat{p}_m) \geq z^*$. Furthermore, since $p_i^t \in \mathcal{F}_i$ for all $p_i^t \in \mathcal{P}_i$, the solution $(\hat{p}_1, \dots, \hat{p}_m)$ is feasible to problem (3). Therefore, $(\hat{p}_1, \dots, \hat{p}_m)$ is a feasible solution to problem (3) and provides an objective value to problem (3) that is at least as large as the optimal objective value of this problem, which implies that $(\hat{p}_1, \dots, \hat{p}_m)$ is an optimal solution to problem (3), as desired. We observe that the discussion in this paragraph also provides an answer to the question that we ask at the beginning of this section. In particular, if we know that we can stitch together an optimal solution to problem (3) by picking one price vector from each one of the collections $\mathcal{P}_1, \dots, \mathcal{P}_m$, then we can use Theorem 1 to obtain an optimal solution to problem (3).

One remaining question is how we can find the value of \hat{z} that satisfies $v_0 \hat{z} = f(\hat{z})$ in a tractable fashion. Noting that $v_0 z$ is increasing and $f(z)$ is decreasing in z , we can find the value of \hat{z} that satisfies $v_0 \hat{z} = f(\hat{z})$ by solving the problem $\min\{z : v_0 z \geq \sum_{i \in M} \max_{p_i \in \mathcal{P}_i} V_i(p_i)^{\gamma_i} (R_i(p_i) - z)\}$, where the decision variable is z . The constraint in this problem is nonlinear in z , but we can linearize the constraint by using the additional decision variables $y = (y_1, \dots, y_m)$ with the interpretation that $y_i = \max_{p_i \in \mathcal{P}_i} V_i(p_i)^{\gamma_i} (R_i(p_i) - z)$. In this case, we can find the value of \hat{z} that satisfies $v_0 \hat{z} = f(\hat{z})$ by solving the problem

$$\min \left\{ z : v_0 z \geq \sum_{i \in M} y_i, y_i \geq V_i(p_i)^{\gamma_i} (R_i(p_i) - z) \quad \forall p_i \in \mathcal{P}_i, i \in M \right\}, \quad (6)$$

where the decision variables are (z, y) . The problem above is a linear program with $O(m)$ decision variables and $\sum_{i \in M} O(|\mathcal{P}_i|)$ constraints, which is tractable as long as the numbers of price vectors in the collections $\mathcal{P}_1, \dots, \mathcal{P}_m$ are relatively small. In the rest of our discussion, we focus on how to construct a small collection of candidate price vectors \mathcal{P}_i for each nest i such that we can stitch together an optimal solution to problem (3) by picking one price vector from each one of the collections $\mathcal{P}_1, \dots, \mathcal{P}_m$. Once we have these collections, we can solve problem (6) to find \hat{z} satisfying $v_0 \hat{z} = f(\hat{z})$ and we can use Theorem 1 to obtain an optimal solution to problem (3).

1.3 Characterizing Candidate Price Vectors

In this section, we give a characterization of the optimal price vector to charge in each nest. This characterization ultimately becomes useful to construct a collection of candidate price vectors \mathcal{P}_i for each nest i such that we can stitch together an optimal solution to problem (3) by picking one price vector from each one of the collections $\mathcal{P}_1, \dots, \mathcal{P}_m$. In the next lemma, we begin by giving a simple condition for a solution to provide an objective value for problem (3) that is at least as large as the optimal objective value. Subsequent to this lemma, we build on this condition to give a characterization of the optimal price vector to charge in each nest.

Lemma 2 Let (p_1^*, \dots, p_m^*) be an optimal solution to problem (3) providing the objective value z^* . If $(\hat{p}_1, \dots, \hat{p}_m)$ satisfies

$$V_i(\hat{p}_i)^{\gamma_i} (R_i(\hat{p}_i) - z^*) \geq V_i(p_i^*)^{\gamma_i} (R_i(p_i^*) - z^*)$$

for all $i \in M$, then we have $\Pi(\hat{p}_1, \dots, \hat{p}_m) \geq z^*$.

Proof. Adding the inequality in the lemma over all $i \in M$ yields $\sum_{i \in M} V_i(\hat{p}_i)^{\gamma_i} (R_i(\hat{p}_i) - z^*) \geq \sum_{i \in M} V_i(p_i^*)^{\gamma_i} (R_i(p_i^*) - z^*)$. Since (p_1^*, \dots, p_m^*) is an optimal solution to problem (3), we have $z^* = \sum_{i \in M} V_i(p_i^*)^{\gamma_i} R_i(p_i^*) / (v_0 + \sum_{i \in M} V_i(p_i^*)^{\gamma_i})$. Arranging the terms in this equality, we obtain $v_0 z^* = \sum_{i \in M} V_i(p_i^*)^{\gamma_i} (R_i(p_i^*) - z^*)$, in which case, it follows that $\sum_{i \in M} V_i(\hat{p}_i)^{\gamma_i} (R_i(\hat{p}_i) - z^*) \geq \sum_{i \in M} V_i(p_i^*)^{\gamma_i} (R_i(p_i^*) - z^*) = v_0 z^*$. Focusing on the first and last terms in this chain of inequalities and solving for z^* , we get $\sum_{i \in M} V_i(\hat{p}_i)^{\gamma_i} R_i(\hat{p}_i) / (v_0 + \sum_{i \in M} V_i(\hat{p}_i)^{\gamma_i}) \geq z^*$ and the desired result follows by noting that $\Pi(\hat{p}_1, \dots, \hat{p}_m) = \sum_{i \in M} V_i(\hat{p}_i)^{\gamma_i} R_i(\hat{p}_i) / (v_0 + \sum_{i \in M} V_i(\hat{p}_i)^{\gamma_i})$. \square

In the next theorem, we build on the condition given in Lemma 2 to give a characterization of the optimal price vector to charge in each nest.

Theorem 3 Let (p_1^*, \dots, p_m^*) be an optimal solution to problem (3) providing the objective value z^* and set $u_i^* = \max\{\gamma_i z^* + (1 - \gamma_i) R_i(p_i^*), z^*\}$. If \hat{p}_i is an optimal solution to the problem

$$\max_{p_i \in \mathcal{F}_i} \left\{ V_i(p_i) (R_i(p_i) - u_i^*) \right\}, \quad (7)$$

then $(\hat{p}_1, \dots, \hat{p}_m)$ is an optimal solution to problem (3).

Proof. For notational brevity, we let $R_i^* = R_i(p_i^*)$, $V_i^* = V_i(p_i^*)$, $\hat{R}_i = R_i(\hat{p}_i)$ and $\hat{V}_i = V_i(\hat{p}_i)$. We claim that $\hat{V}_i^{\gamma_i} (\hat{R}_i - z^*) \geq (V_i^*)^{\gamma_i} (R_i^* - z^*)$ for all $i \in M$. To see this claim, we consider a nest i that satisfies $R_i^* \geq z^*$. Since p_i^* is a feasible but not necessarily an optimal solution to problem (7), we have $\hat{V}_i (\hat{R}_i - u_i^*) \geq V_i^* (R_i^* - u_i^*)$. Since $R_i^* \geq z^*$, we have $u_i^* = \gamma_i z^* + (1 - \gamma_i) R_i^*$ by the definition of u_i^* and plugging this value of u_i^* into the last inequality yields $\hat{V}_i (\hat{R}_i - z^*) - (1 - \gamma_i) \hat{V}_i (R_i^* - z^*) \geq \gamma_i V_i^* (R_i^* - z^*)$. Arranging the terms in the last inequality, we obtain

$$\hat{R}_i - z^* \geq \left[\gamma_i \frac{V_i^*}{\hat{V}_i} + (1 - \gamma_i) \right] (R_i^* - z^*). \quad (8)$$

Noting that the dissimilarity parameter for nest i satisfies $\gamma_i \in [0, 1]$, the function x^{γ_i} is concave in x and its derivative at point 1 is γ_i . Therefore, the subgradient inequality at point 1 yields $x^{\gamma_i} \leq 1 + \gamma_i(x - 1) = \gamma_i x + (1 - \gamma_i)$ for all $x \in \mathfrak{R}_+$. Using the subgradient inequality with $x = V_i^* / \hat{V}_i$, it follows that $(V_i^* / \hat{V}_i)^{\gamma_i} \leq \gamma_i (V_i^* / \hat{V}_i) + 1 - \gamma_i$. In this case, since $R_i^* \geq z^*$, (8) implies that $\hat{R}_i - z^* \geq (V_i^* / \hat{V}_i)^{\gamma_i} (R_i^* - z^*)$ and arranging the terms in this inequality yields $\hat{V}_i^{\gamma_i} (\hat{R}_i - z^*) \geq (V_i^*)^{\gamma_i} (R_i^* - z^*)$. Therefore, the claim holds for each nest i that satisfies $R_i^* \geq z^*$.

We consider a nest i that satisfies $R_i^* < z^*$. Since θ^q is the largest possible price for a product, the optimal expected revenue in problem (3) does not exceed θ^q and we obtain $z^* \leq \theta^q$. We

define the solution $\tilde{p}_i = (\tilde{p}_{i1}, \dots, \tilde{p}_{in})$ to problem (7) as $\tilde{p}_{ij} = \theta^q$ for all $j \in N$, which is feasible to this problem. Furthermore, (1) implies that $R_i(\tilde{p}_i) = \sum_{j \in N} \theta^q v_{ij}(\tilde{p}_{ij}) / \sum_{j \in N} v_{ij}(\tilde{p}_{ij}) = \theta^q$. Since $R_i^* < z^*$, we have $u_i^* = z^*$ by the definition of u_i^* and we obtain $\hat{V}_i(\hat{R}_i - z^*) = \hat{V}_i(\hat{R}_i - u_i^*) \geq V_i(\tilde{p}_i)(R_i(\tilde{p}_i) - u_i^*) = V_i(\tilde{p}_i)(R_i(\tilde{p}_i) - z^*) \geq 0 > V_i^*(R_i^* - z^*)$, where the first inequality uses the fact that \tilde{p}_i is a feasible but not necessarily an optimal solution to problem (7), the second inequality uses the fact that $R_i(\tilde{p}_i) = \theta^q \geq z^*$ and the third inequality uses the fact that $R_i^* < z^*$. Thus, we have $\hat{V}_i(\hat{R}_i - z^*) \geq 0 > V_i^*(R_i^* - z^*)$, which implies that $\hat{V}_i^{\gamma_i}(\hat{R}_i - z^*) \geq 0 > (V_i^*)^{\gamma_i}(R_i^* - z^*)$, establishing the claim for each nest i that satisfies $R_i^* < z^*$.

The discussion in the previous two paragraphs shows that our claim holds and we obtain $\hat{V}_i^{\gamma_i}(\hat{R}_i - z^*) \geq (V_i^*)^{\gamma_i}(R_i^* - z^*)$ for all $i \in M$. In this case, the solution $(\hat{p}_1, \dots, \hat{p}_m)$ satisfies the assumption of Lemma 2 and it follows from this lemma that $\Pi(\hat{p}_1, \dots, \hat{p}_m) \geq z^*$. On the other hand, noting that \hat{p}_i is a feasible solution to problem (7), we have $\hat{p}_i \in \mathcal{F}_i$ for all $i \in M$, which indicates that $(\hat{p}_1, \dots, \hat{p}_m)$ is a feasible solution to problem (3). The solution $(\hat{p}_1, \dots, \hat{p}_m)$ is feasible to problem (3) and it provides an objective value for problem (3) that is at least as large as the optimal objective value of this problem. Therefore, we conclude that $(\hat{p}_1, \dots, \hat{p}_m)$ is an optimal solution to problem (3), as desired. \square

By Theorem 3, we can recover an optimal solution to problem (3) by solving problem (7) for all $i \in M$. Thus, if we let \hat{p}_i be an optimal solution to problem (7) and use the singleton $\mathcal{P}_i = \{\hat{p}_i\}$ as the collection of candidate price vectors to charge in nest i , then we can stitch together an optimal solution to problem (3) by picking one price vector from each one of the collections $\mathcal{P}_1, \dots, \mathcal{P}_m$. However, this approach is not immediately useful for constructing a collection of candidate price vectors, since solving problem (7) requires the knowledge of u_i^* , which, in turn, requires the knowledge of an optimal solution to problem (3). To get around this difficulty, as a function of $u_i \in \mathfrak{R}_+$, we use $\hat{p}_i(u_i)$ to denote an optimal solution to the problem

$$\max_{p_i \in \mathcal{F}_i} \left\{ V_i(p_i)(R_i(p_i) - u_i) \right\}. \quad (9)$$

In this case, we observe that if we use the collection of price vectors $\mathcal{P}_i = \{\hat{p}_i(u_i) : u_i \in \mathfrak{R}_+\}$ as the collection of candidate price vectors for nest i , then we can stitch together an optimal solution to problem (3) by picking one price vector from each one of the collections $\mathcal{P}_1, \dots, \mathcal{P}_m$. To see this result, letting u_i^* be as defined in Theorem 3, we note that $\hat{p}_i(u_i^*) \in \{\hat{p}_i(u_i) : u_i \in \mathfrak{R}_+\}$ for all $i \in M$. Furthermore, since problem (9) with $u_i = u_i^*$ is identical to problem (7), by Theorem 3, the solution $(\hat{p}_1(u_1^*), \dots, \hat{p}_m(u_m^*))$ is optimal to problem (3). Therefore, for each nest i , the solution $(\hat{p}_1(u_1^*), \dots, \hat{p}_m(u_m^*))$ uses one price vector from the collection of candidate price vectors $\mathcal{P}_i = \{\hat{p}_i(u_i) : u_i \in \mathfrak{R}_+\}$ and this solution is optimal to problem (3).

We propose using $\{\hat{p}_i(u_i) : u_i \in \mathfrak{R}_+\}$ as the collection of candidate price vectors for nest i , which is the collection of optimal solutions to problem (9) for any value of $u_i \in \mathfrak{R}_+$. In the subsequent sections, we show that the collection $\{\hat{p}_i(u_i) : u_i \in \mathfrak{R}_+\}$ includes a reasonably small number of price vectors and we can find these price vectors in a tractable fashion.

1.4 Counting Candidate Price Vectors

In this section, we show that there exists a collection of price vectors $\mathcal{P}_i = \{p_i^t : t \in \mathcal{T}_i\}$ such that this collection includes an optimal solution to problem (9) for any value of $u_i \in \mathfrak{R}_+$ and there are at most nq price vectors in this collection, where n is the number of products in a nest and q is the number of possible price levels. The fact that the objective function of problem (9) has a simple form plays an especially important role in this result. In particular, using the definitions of $V_i(p_i)$ and $R_i(p_i)$, we can write problem (9) equivalently as

$$\max_{p_i \in \mathcal{F}_i} \left\{ \sum_{j \in N} v_{ij}(p_{ij}) \left[\frac{\sum_{j \in N} p_{ij} v_{ij}(p_{ij})}{\sum_{j \in N} v_{ij}(p_{ij})} - u_i \right] \right\} = \max_{p_i \in \mathcal{F}_i} \left\{ \sum_{j \in N} (p_{ij} - u_i) v_{ij}(p_{ij}) \right\}. \quad (10)$$

In the next lemma, we begin by showing that as the value of u_i in problem (10) gets larger, then the optimal price for each product also gets larger.

Lemma 4 *Using $\hat{p}_i(u_i) = (\hat{p}_{i1}(u_i), \dots, \hat{p}_{in}(u_i))$ to denote an optimal solution to problem (10) as a function of u_i , if we have $u_i^- < u_i^+$, then it holds that $\hat{p}_{ij}(u_i^-) \leq \hat{p}_{ij}(u_i^+)$ for all $j \in N$.*

Proof. To get a contradiction, assume that $u_i^- < u_i^+$ but we have $\hat{p}_{ij}(u_i^-) > \hat{p}_{ij}(u_i^+)$ for some $j \in N$. For notational brevity, we let $\hat{p}_i^- = \hat{p}_i(u_i^-)$ and $\hat{p}_i^+ = \hat{p}_i(u_i^+)$. Since the solutions \hat{p}_i^- and \hat{p}_i^+ are optimal to problem (10) when this problem is solved at particular values of u_i , we have $\hat{p}_i^- \in \mathcal{F}_i$ and $\hat{p}_i^+ \in \mathcal{F}_i$, which is to say that $\hat{p}_{i1}^- \leq \hat{p}_{i2}^- \leq \dots \leq \hat{p}_{in}^-$ and $\hat{p}_{i1}^+ \leq \hat{p}_{i2}^+ \leq \dots \leq \hat{p}_{in}^+$. We let $J = \{j \in N : \hat{p}_{ij}^- > \hat{p}_{ij}^+\}$, which is nonempty by the assumption that $\hat{p}_{ij}^- > \hat{p}_{ij}^+$ for some $j \in N$.

We define the solution $\tilde{p}_i = (\tilde{p}_{i1}, \dots, \tilde{p}_{in})$ to problem (10) as $\tilde{p}_{ij} = \hat{p}_{ij}^- \vee \hat{p}_{ij}^+$ for all $j \in N$, where we use $a \vee b = \max\{a, b\}$. If $f(j)$ and $g(j)$ are both increasing functions of $j \in N$, then $f(j) \vee g(j)$ is also an increasing function of $j \in N$. By the discussion at the end of the previous paragraph, \hat{p}_{ij}^- and \hat{p}_{ij}^+ are both increasing functions of $j \in N$. Thus, $\tilde{p}_{ij} = \hat{p}_{ij}^- \vee \hat{p}_{ij}^+$ is also an increasing function of $j \in N$, which implies that $\tilde{p}_{i1} \leq \tilde{p}_{i2} \leq \dots \leq \tilde{p}_{in}$. Therefore, we have $\tilde{p}_i \in \mathcal{F}_i$, indicating that \tilde{p}_i is a feasible solution to problem (10). In this case, since \hat{p}_i^+ is an optimal solution to problem (10) when we solve this problem with $u_i = u_i^+$, we have $\sum_{j \in N} (\hat{p}_{ij}^+ - u_i^+) v_{ij}(\hat{p}_{ij}^+) \geq \sum_{j \in N} (\tilde{p}_{ij} - u_i^+) v_{ij}(\tilde{p}_{ij})$. By the definitions of J and \tilde{p}_i , we have $\tilde{p}_{ij} = \hat{p}_{ij}^-$ for all $j \in J$ and $\tilde{p}_{ij} = \hat{p}_{ij}^+$ for all $j \notin J$. Thus, the last inequality can be written as $\sum_{j \in N} (\hat{p}_{ij}^+ - u_i^+) v_{ij}(\hat{p}_{ij}^+) \geq \sum_{j \in J} (\hat{p}_{ij}^- - u_i^+) v_{ij}(\hat{p}_{ij}^-) + \sum_{j \notin J} (\hat{p}_{ij}^+ - u_i^+) v_{ij}(\hat{p}_{ij}^+)$, in which case, canceling the common terms on the two sides of the inequality, we have $\sum_{j \in J} (\hat{p}_{ij}^+ - u_i^+) v_{ij}(\hat{p}_{ij}^+) \geq \sum_{j \in J} (\hat{p}_{ij}^- - u_i^+) v_{ij}(\hat{p}_{ij}^-)$.

We define the solution $\bar{p}_i = (\bar{p}_{i1}, \dots, \bar{p}_{in})$ to problem (10) as $\bar{p}_{ij} = \hat{p}_{ij}^+ \wedge \hat{p}_{ij}^-$ for all $j \in N$, where we use $a \wedge b = \min\{a, b\}$. We note that if $f(j)$ and $g(j)$ are both increasing functions of $j \in N$, then $f(j) \wedge g(j)$ is also an increasing function of $j \in N$. In this case, using the same approach in the previous paragraph, we can show that $\bar{p}_i \in \mathcal{F}_i$. Thus, since \hat{p}_i^- is an optimal solution to problem (10) when we solve this problem with $u_i = u_i^-$, we have $\sum_{j \in N} (\hat{p}_{ij}^- - u_i^-) v_{ij}(\hat{p}_{ij}^-) \geq \sum_{j \in N} (\bar{p}_{ij} - u_i^-) v_{ij}(\bar{p}_{ij})$. Noting the definitions of J and \bar{p}_i , the last inequality can equivalently be

written as $\sum_{j \in N} (\hat{p}_{ij}^- - u_i^-) v_{ij}(\hat{p}_{ij}^-) \geq \sum_{j \in J} (\hat{p}_{ij}^+ - u_i^-) v_{ij}(\hat{p}_{ij}^+) + \sum_{j \notin J} (\hat{p}_{ij}^- - u_i^-) v_{ij}(\hat{p}_{ij}^-)$, in which case, canceling the common terms on the two sides of the inequality yields $\sum_{j \in J} (\hat{p}_{ij}^- - u_i^-) v_{ij}(\hat{p}_{ij}^-) \geq \sum_{j \in J} (\hat{p}_{ij}^+ - u_i^-) v_{ij}(\hat{p}_{ij}^+)$. From the previous paragraph, we also have $\sum_{j \in J} (\hat{p}_{ij}^+ - u_i^+) v_{ij}(\hat{p}_{ij}^+) \geq \sum_{j \in J} (\hat{p}_{ij}^- - u_i^+) v_{ij}(\hat{p}_{ij}^-)$. Adding the last two inequalities and canceling the common terms yield $u_i^- \sum_{j \in J} (v_{ij}(\hat{p}_{ij}^+) - v_{ij}(\hat{p}_{ij}^-)) \geq u_i^+ \sum_{j \in J} (v_{ij}(\hat{p}_{ij}^+) - v_{ij}(\hat{p}_{ij}^-))$.

By the definition of J , we have $\hat{p}_{ij}^- > \hat{p}_{ij}^+$ for all $j \in J$. Noting that the preference weight of a product gets larger as we charge a smaller price for the product, we have $v_{ij}(\hat{p}_{ij}^-) < v_{ij}(\hat{p}_{ij}^+)$ for all $j \in J$. Thus, we have $\sum_{j \in J} (v_{ij}(\hat{p}_{ij}^+) - v_{ij}(\hat{p}_{ij}^-)) > 0$, in which case, by the inequality at the end of the previous paragraph, we obtain $u_i^- \geq u_i^+$, which is a contradiction. \square

We observe that the last step in the proof of Lemma 4 critically depends on the assumption that $v_{ij}(p_{ij})$ is a decreasing function of p_{ij} . Also, it is worthwhile to note that Lemma 4 holds even when there are multiple optimal solutions to problem (10) and we choose $\hat{p}_i(u_i)$ as any one of these solutions. In the next theorem, we use Lemma 4 to show that there exists a collection of at most nq price vectors such that this collection includes an optimal solution to problem (10) for any value of $u_i \in \mathfrak{R}_+$. The intuition behind this result is that if we increase the value of u_i in problem (10), then Lemma 4 implies that the price of a product in an optimal solution either does not change or becomes larger. Since there are q possible prices for a product, the price of a product will no longer change after a small number of price changes.

Theorem 5 *There exists a collection of at most nq price vectors such that this collection includes an optimal solution to problem (10) for any value of $u_i \in \mathfrak{R}_+$.*

Proof. Assume that there are K distinct values of $u_i \in \mathfrak{R}_+$ such that if we solve problem (10) with each one of these values, then we obtain a distinct optimal solution. We use $\{\hat{u}_i^k : k = 1, \dots, K\}$ to denote these values of $u_i \in \mathfrak{R}_+$ and use \hat{p}_i^k to denote an optimal solution to problem (10) when we solve this problem with $u_i = \hat{u}_i^k$. By our assumption, none of price vectors in $\{\hat{p}_i^k : k = 1, \dots, K\}$ are equal to each other. To get a contradiction, assume that $K > nq$. Without loss of generality, we index the values $\{\hat{u}_i^k : k = 1, \dots, K\}$ such that $\hat{u}_i^1 < \hat{u}_i^2 < \dots < \hat{u}_i^K$, in which case, Lemma 4 implies that $\hat{p}_{ij}^1 \leq \hat{p}_{ij}^2 \leq \dots \leq \hat{p}_{ij}^K$ for all $j \in N$. Since the price vectors $\{\hat{p}_i^k : k = 1, \dots, K\}$ are distinct, using $\mathbf{1}(\cdot)$ to denote the indicator function, we have $\sum_{j \in N} \mathbf{1}(\hat{p}_{ij}^k < \hat{p}_{ij}^{k+1}) > 1$, indicating that there is at least one different price in the price vectors \hat{p}_i^k and \hat{p}_i^{k+1} . Adding the last inequality over all $k = 1, \dots, K-1$ and noting that $K > nq$, we obtain $\sum_{j \in N} \sum_{k=1}^{K-1} \mathbf{1}(\hat{p}_{ij}^k < \hat{p}_{ij}^{k+1}) > K-1 \geq nq$. Focusing on the first and last terms in the last chain of inequalities, since $|N| = n$, it must be the case that $\sum_{k=1}^{K-1} \mathbf{1}(\hat{p}_{ij}^k < \hat{p}_{ij}^{k+1}) > q$ for some $j \in N$, which implies that more than q of the inequalities $\hat{p}_{ij}^1 \leq \hat{p}_{ij}^2 \leq \dots \leq \hat{p}_{ij}^K$ are strict, but since there are q possible values for the price of a product, more than q of these inequalities cannot be strict and we obtain a contradiction. \square

Thus, there exists a reasonably small collection of price vectors that includes an optimal solution to problem (10) for any $u_i \in \mathfrak{R}_+$. In the next section, we show how to construct this collection.

1.5 Constructing Candidate Price Vectors

In the previous section, we show that there exists a collection of price vectors with at most nq price vectors in it such that this collection includes an optimal solution to problem (10) for any value of $u_i \in \mathfrak{R}_+$. In this section, we show how to come up with this collection in a tractable fashion. In problem (10), if we charge the price p_{ij} for product j in nest i , then we obtain a contribution of $(p_{ij} - u_i)v_{ij}(p_{ij})$. By the constraint $p_i \in \mathcal{F}_i$, the price charged for product j should be at least as large as the price charged for product $j - 1$. Problem (10) finds the prices to charge for the products in nest i to maximize the total contribution. So, we can solve problem (10) by using a dynamic program. The decision epochs are the products in nest i . When making the decision for product j in nest i , the state variable is the price for product $j - 1$. Thus, for a fixed value of $u_i \in \mathfrak{R}_+$, we can solve problem (10) by using the dynamic program

$$\Phi_{ij}(p_{i,j-1} | u_i) = \max_{\substack{p_{ij} \in \Theta : \\ p_{ij} \geq p_{i,j-1}}} \left\{ (p_{ij} - u_i)v_{ij}(p_{ij}) + \Phi_{i,j+1}(p_{ij} | u_i) \right\}, \quad (11)$$

with the boundary condition that $\Phi_{i,n+1}(\cdot | u_i) = 0$. The optimal objective value of problem (10) is given by $\Phi_{i1}(\theta^1 | u_i)$, where the value functions $\{\Phi_{ij}(p_{i,j-1} | u_i) : p_{i,j-1} \in \Theta, j \in N\}$ are obtained through the dynamic program in (11). By Theorem 5, there are most nq solutions from the dynamic program in (11) such that the solution from this dynamic program for any value of $u_i \in \mathfrak{R}_+$ is one of these nq solutions. The question is how to come up with these solutions.

To answer this question, we use the linear programming formulation of the dynamic program in (11). Dynamic programs with finite state and action spaces have equivalent linear programming formulations; see Puterman (1994). In these linear programs, there is one decision variable for each state and decision epoch and there is one constraint for each state, action and decision epoch. The linear program corresponding to the dynamic program in (11) is given by

$$\begin{aligned} \min \quad & \phi_{i1}(\theta^1) \\ \text{s.t.} \quad & \phi_{ij}(p_{i,j-1}) \geq (p_{ij} - u_i)v_{ij}(p_{ij}) + \phi_{i,j+1}(p_{ij}) \quad \forall p_{i,j-1} \in \Theta, p_{ij} \in \mathcal{L}(p_{i,j-1}), j \in N, \end{aligned} \quad (12)$$

where the decision variables are $\{\phi_{ij}(p_{i,j-1}) : p_{i,j-1} \in \Theta, j \in N\}$ and we follow the convention that $\phi_{i,n+1}(p_{in}) = 0$ for all $p_{in} \in \Theta$. The set $\mathcal{L}(p_{i,j-1})$ is the set of feasible prices for product j given that the price for product $j - 1$ is $p_{i,j-1}$, which is given by $\mathcal{L}(p_{i,j-1}) = \{p_{ij} \in \Theta : p_{ij} \geq p_{i,j-1}\}$. If we solve the linear program in (12), then the optimal value of the decision variable $\phi_{i1}(\theta^1)$ is equal to $\Phi_{i1}(\theta^1 | u_i)$ obtained through the dynamic program in (11), which is, in turn, equal to the optimal objective value of problem (10). The critical observation is that the value of $u_i \in \mathfrak{R}_+$ only affects the right hand side coefficients of the constraints in problem (12). Therefore, we can vary $u_i \in \mathfrak{R}_+$ parametrically and solve problem (12) by using the parametric simplex method to generate the possible optimal solutions to this problem for all values of $u_i \in \mathfrak{R}_+$. These solutions provide the solutions to the dynamic program in (11) for all values of $u_i \in \mathfrak{R}_+$.

Since there are q possible prices for a product and there are n products in a nest, the linear program in (12) has $O(nq)$ decision variables and $O(nq^2)$ constraints. Putting all of the discussion so

far together, we solve problem (12) by using the parametric simplex method to generate the optimal solutions to this problem for all values of $u_i \in \mathfrak{R}_+$. These solutions correspond to the optimal solutions to problem (10) for all values of $u_i \in \mathfrak{R}_+$. By the discussion that follows Theorem 3, we can use the optimal solutions to problem (10) for all values of $u_i \in \mathfrak{R}_+$ as the collection of candidate price vectors \mathcal{P}_i for nest i . Once we have the collection of candidate price vectors for each nest, we can solve the linear program in (6) to obtain the value of \hat{z} that satisfies $v_0 \hat{z} = f(\hat{z})$. Since there are at most nq price vectors in each one of the collections $\mathcal{P}_1, \dots, \mathcal{P}_m$, there are $O(m)$ decision variables and $O(mnq)$ constraints in the linear program in (6). In this case, by Theorem 1, we can solve problem (5) for all $i \in M$ to obtain an optimal solution to problem (3).

In Section 4, we provide numerical experiments and demonstrate that the approach described above can obtain an optimal solution to problem (3) quite fast. For problem instances with six nests, 30 products in each nest and 30 possible prices for each product, we can obtain an optimal solution to problem (3) in no more than two seconds.

2 Price Ladders Between Nests

In this section, we consider the case with price ladders between nests. In this setting, there is an intrinsic ordering between the qualities of the nests and the prices charged in a nest corresponding to a higher quality level should also be larger. There is no intrinsic ordering between the qualities or the prices of the products in the same nest.

2.1 Problem Formulation

Our problem formulation is similar to the one in Section 1.1. There are m nests indexed by $M = \{1, \dots, m\}$. In each nest, there are n products indexed by $N = \{1, \dots, n\}$. For each product, there are q possible prices given by $\Theta = \{\theta^1, \dots, \theta^q\}$. The possible prices for a product are indexed such that $0 < \theta^1 < \theta^2 < \dots < \theta^q$. We use $p_{ij} \in \Theta$ to denote the price that we charge for product j in nest i . If we charge the price p_{ij} for product j in nest i , then the preference weight of this product is given by $v_{ij}(p_{ij})$. If we charge a larger price for a product, then its preference weight becomes smaller, implying that $v_{ij}(\theta^1) > v_{ij}(\theta^2) > \dots > v_{ij}(\theta^q) > 0$. Customers follow the same choice process described in Section 1.1. Thus, if we use $p_i = (p_{i1}, \dots, p_{in}) \in \Theta^n$ to denote the price vector charged in nest i , then the expected revenue obtained from a customer that has already decided to make a purchase in nest i is given by $R_i(p_i)$, where $R_i(p_i)$ is as in (1). If we charge the price vectors $(p_1, \dots, p_m) \in \Theta^{m \times n}$ over all nests, then the expected revenue obtained from a customer is given by $\Pi(p_1, \dots, p_m)$, where $\Pi(p_1, \dots, p_m)$ is as in (2).

Our goal is to find the price vectors (p_1, \dots, p_m) to maximize the expected revenue $\Pi(p_1, \dots, p_m)$ subject to the constraint that the price vectors charged in the different nests are consistent with the quality level that each nest represents. In other words, if nest i corresponds to a higher quality level than nest l , then the prices of the products in nest i should be larger than the prices of

the products in nest l . This constraint can be interpreted as a price ladder constraint between the nests. To formulate the price ladder constraint, without loss of generality, we index the nests such that a nests with a larger index represents a higher quality level. In other words, the nests $M = \{1, \dots, m\}$ are indexed in the order of increasing quality levels. Thus, the price ladder constraint ensures that the price vectors (p_1, \dots, p_m) charged over all nests satisfy $\max_{j \in N} p_{1j} \leq \min_{j \in N} p_{2j}$, $\max_{j \in N} p_{2j} \leq \min_{j \in N} p_{3j}$, \dots , $\max_{j \in N} p_{m-1,j} \leq \min_{j \in N} p_{mj}$. As a function of the price vector p_{i-1} charged in nest $i - 1$, the set of feasible price vectors in nest i is $\mathcal{G}_i(p_{i-1}) = \{p_i \in \Theta^n : \min_{j \in N} p_{ij} \geq \max_{j \in N} p_{i-1,j}\}$. We want to find the price vectors to charge over all nests to maximize the expected revenue from a customer, yielding the problem

$$z^* = \max_{\substack{(p_1, \dots, p_m) \in \Theta^{m \times n} : \\ p_i \in \mathcal{G}_i(p_{i-1}) \forall i \in M \setminus \{1\}}} \left\{ \Pi(p_1, \dots, p_m) \right\}. \quad (13)$$

Problem (13) is significantly more difficult than problem (3) since the constraints link the price vectors charged in different nests. The broad outline of our approach for problem (13) is similar to the one for problem (3). We relate problem (13) to the problem of computing the fixed point of a function. Assuming that we have a collection of candidate price vectors for each nest such that we can stitch together an optimal solution to problem (13) by picking one price vector from each one of the collections, we show how to obtain an optimal solution to problem (13). Finally, we show how to come up with the collections of candidate price vectors. Although the broad outline of our approach for problem (13) is similar to the one for problem (3), the details are quite different as problem (13) is significantly more difficult than problem (3).

2.2 Connection to a Fixed Point Representation

Assume that we have a collection of candidate price vectors $\mathcal{P}_i = \{p_i^t : t \in \mathcal{T}_i\}$ for each nest i such that we can stitch together an optimal solution to problem (13) by picking one price vector from each one of the collections $\mathcal{P}_1, \dots, \mathcal{P}_m$. In other words, there exists an optimal solution (p_1^*, \dots, p_m^*) to problem (13) such that $p_i^* \in \mathcal{P}_i$ for all $i \in M$. The question is how we can pick a price vector p_i^* from the collection \mathcal{P}_i for each nest i such that the solution (p_1^*, \dots, p_m^*) is indeed optimal to problem (13). To answer this question, for any $z \in \mathfrak{R}_+$, we define $g(z)$ as

$$g(z) = \max_{\substack{(p_1, \dots, p_m) \in \mathcal{P}_1 \times \dots \times \mathcal{P}_m : \\ p_i \in \mathcal{G}_i(p_{i-1}) \forall i \in M \setminus \{1\}}} \left\{ \sum_{i \in M} V_i(p_i)^{\gamma_i} (R_i(p_i) - z) \right\}. \quad (14)$$

Since $v_0 z$ is increasing and $g(z)$ is decreasing in z with $g(0) \geq 0$, there exists a value of \hat{z} that satisfies $v_0 \hat{z} = g(\hat{z})$, which corresponds to the fixed point of the function $g(\cdot)/v_0$. In the next theorem, we show that the value of \hat{z} that satisfies $v_0 \hat{z} = g(\hat{z})$ can be used to construct an optimal solution to problem (13). The proof of this theorem follows from an outline that is similar to the proof of Theorem 1 and we omit the proof. In the theorem, we recall that z^* corresponds to the optimal objective value of problem (13).

Theorem 6 *Assume that we have a collection of candidate price vectors \mathcal{P}_i for each nest i such that we can stitch together an optimal solution to problem (13) by picking one price vector from each one of the collections $\mathcal{P}_1, \dots, \mathcal{P}_m$. Let \hat{z} be such that $v_0 \hat{z} = g(\hat{z})$ and $(\hat{p}_1, \dots, \hat{p}_m)$ be an optimal solution to problem (14) when we solve this problem with $z = \hat{z}$. Then, we have $\Pi(\hat{p}_1, \dots, \hat{p}_m) \geq z^*$.*

Building on Theorem 6, we can obtain an optimal solution to problem (13) as follows. Assume that we have a collection of candidate price vectors $\mathcal{P}_i = \{p_i^t : t \in \mathcal{T}_i\}$ for each nest i such that we can stitch together an optimal solution to problem (13) by picking one price vector from each one of the collections $\mathcal{P}_1, \dots, \mathcal{P}_m$. We find the value of \hat{z} that satisfies $v_0 \hat{z} = g(\hat{z})$. If we let $(\hat{p}_1, \dots, \hat{p}_m)$ be an optimal solution to problem (14) when we solve this problem with $z = \hat{z}$, then by Theorem 6, we have $\Pi(\hat{p}_1, \dots, \hat{p}_m) \geq z^*$. Since $(\hat{p}_1, \dots, \hat{p}_m)$ is a feasible solution to problem (14), we also have $\hat{p}_i \in \mathcal{G}_i(p_{i-1})$ for all $i \in M \setminus \{1\}$. Thus, the solution $(\hat{p}_1, \dots, \hat{p}_m)$ is feasible to problem (13) and provides an objective value for problem (13) that is at least as large as the optimal objective value of this problem, indicating that $(\hat{p}_1, \dots, \hat{p}_m)$ is an optimal solution to problem (13).

One important question is how we can find the value of \hat{z} that satisfies $v_0 \hat{z} = g(\hat{z})$. In problem (14), we observe that if we charge the price vector p_i in nest i , then we obtain a contribution of $V_i(p_i)^{\gamma_i} (R_i(p_i) - z)$. By the constraints $\hat{p}_i \in \mathcal{G}_i(p_{i-1})$ for all $i \in M \setminus \{1\}$, the smallest price charged in nest i should be at least as large as the largest price charged in nest $i - 1$. Problem (14) finds the price vectors to charge for the nests to maximize the total contribution. So, for a fixed value of $z \in \mathfrak{R}_+$, we can solve problem (14) by using a dynamic program. The decision epochs are the nests. When making the decision for nest i , the state variable is the largest price charged for the products in nest $i - 1$. Thus, for a fixed value of $z \in \mathfrak{R}_+$, we can obtain an optimal solution to problem (14) by solving the dynamic program

$$\Psi_i(w_{i-1} | z) = \max_{\substack{p_i \in \mathcal{P}_i : \\ p_{ij} \geq w_{i-1} \forall j \in N}} \left\{ V_i(p_i)^{\gamma_i} (R_i(p_i) - z) + \Psi_{i+1}(\max_{j \in N} p_{ij} | z) \right\}, \quad (15)$$

with the boundary condition that $\Psi_{m+1}(\cdot | z) = 0$. The optimal objective value of problem (14) is given by $\Psi_1(\theta^1 | z)$, where the value functions $\{\Psi_i(w_{i-1} | z) : w_{i-1} \in \Theta, i \in M\}$ are obtained through the dynamic program in (15). Since there are q possible prices for a product, we can solve the dynamic program above in $O(q \sum_{i \in M} |\mathcal{P}_i|)$ operations, which is reasonable as long as the numbers of price vectors in the collections $\mathcal{P}_1, \dots, \mathcal{P}_m$ are not too large. In Section 1.2, we recall that we find the value of \hat{z} satisfying $v_0 \hat{z} = f(\hat{z})$ by using the linear program in (6), but this linear program is not useful to find the value of \hat{z} satisfying $v_0 \hat{z} = g(\hat{z})$ since problem (14) does not decompose by the nests due to the constraints $p_i \in \mathcal{G}_i(p_{i-1})$ for all $i \in M \setminus \{1\}$. Instead, we show how to use the linear programming formulation of the dynamic program in (15) to find the value of \hat{z} that satisfies $v_0 \hat{z} = g(\hat{z})$.

As mentioned in Section 1.5, dynamic programs with finite state and action spaces have equivalent linear programming formulations. In these linear programs, there is one decision variable for each state and decision epoch and there is one constraint for each state, action and decision

epoch. Therefore, building on the linear programming formulation corresponding to the dynamic program in (15), we propose solving the linear program

$$\begin{aligned}
\min \quad & \psi_1(\theta^1) \\
\text{s.t.} \quad & \psi_i(w_{i-1}) \geq V_i(p_i)^{\gamma_i} (R_i(p_i) - z) + \psi_{i+1}(\max_{j \in N} p_{ij}) \quad \forall w_{i-1} \in \Theta, p_i \in \mathcal{M}_i(w_{i-1}), i \in M \\
& v_0 z = \psi_1(\theta^1)
\end{aligned} \tag{16}$$

to find the value of \hat{z} that satisfies $v_0 \hat{z} = g(\hat{z})$. In the linear program above, the decision variables are z and $\psi = \{\psi_i(w_{i-1}) : w_{i-1} \in \Theta, i \in M\}$. We follow the convention that $\psi_{m+1}(w_m) = 0$ for all $w_m \in \Theta$. The set $\mathcal{M}_i(w_{i-1})$ is the set of feasible price vectors in nest i given that the largest price charged in nest $i-1$ is w_{i-1} , which is given by $\mathcal{M}_i(w_{i-1}) = \{p_i \in \mathcal{P}_i : p_{ij} \geq w_{i-1} \forall j \in N\}$. If we drop the second constraint in problem (16) and solve this problem for a fixed value of $z \in \mathbb{R}_+$, then this problem corresponds to the linear programming formulation for the dynamic program in (15). Therefore, the optimal value of the decision variable $\psi_1(\theta^1)$ would correspond to $\Psi_1(\theta^1 | z)$ obtained through the dynamic program in (15), which is equal to the optimal objective value of problem (14) for a fixed value of z . On the other hand, it turns out that if we solve problem (16) as formulated, then the optimal value of the decision variable z corresponds to the value of \hat{z} that satisfies $v_0 \hat{z} = g(\hat{z})$. We show this result in the next theorem.

Theorem 7 *Using $(\hat{z}, \hat{\psi})$ to denote an optimal solution to problem (16), we have $v_0 \hat{z} = g(\hat{z})$.*

Proof. Let \tilde{z} satisfy $v_0 \tilde{z} = g(\tilde{z})$. We want to show that $\tilde{z} = \hat{z}$. We solve the dynamic program in (15) with $z = \tilde{z}$ to obtain the value functions $\Psi(\tilde{z}) = \{\Psi_i(w_{i-1} | \tilde{z}) : w_{i-1} \in \Theta, i \in M\}$. Due to the way these value functions are computed in the dynamic program in (15), we have $\Psi_i(w_{i-1} | \tilde{z}) \geq V_i(p_i)^{\gamma_i} (R_i(p_i) - \tilde{z}) + \Psi_{i+1}(\max_{j \in N} p_{ij} | \tilde{z})$ for all $w_{i-1} \in \Theta, p_i \in \mathcal{M}(w_{i-1})$ and $i \in M$. Thus, the solution $(\hat{z}, \Psi(\hat{z}))$ satisfies the first set of constraints in problem (16). By the discussion that follows the dynamic program in (15), $\Psi_1(\theta^1 | \tilde{z})$ provides the optimal objective value of problem (14) when we solve this problem with $z = \tilde{z}$, yielding $\Psi_1(\theta^1 | \tilde{z}) = g(\tilde{z}) = v_0 \tilde{z}$. Thus, the solution $(\hat{z}, \Psi(\hat{z}))$ satisfies the second constraint in problem (16) as well. Since the solution $(\hat{z}, \Psi(\hat{z}))$ is feasible to problem (16), the objective value provided by this solution is at least as large as the optimal objective value, yielding $\Psi_1(\theta^1 | \tilde{z}) \geq \hat{\psi}_1(\theta^1)$. Thus, we obtain $v_0 \tilde{z} = g(\tilde{z}) = \Psi_1(\theta^1 | \tilde{z}) \geq \hat{\psi}_1(\theta^1) = v_0 \hat{z}$, where the last equality holds since $(\hat{z}, \hat{\psi})$ is a feasible solution to problem (16).

The last chain of inequalities in the previous paragraph shows that $\tilde{z} \geq \hat{z}$. To show that $\tilde{z} = \hat{z}$, we solve problem (14) with $z = \hat{z}$ to obtain an optimal solution $(\hat{p}_1, \dots, \hat{p}_m)$. Therefore, we have $g(\hat{z}) = \sum_{i \in M} V_i(\hat{p}_i)^{\gamma_i} (R_i(\hat{p}_i) - \hat{z})$. For all $i \in M$, we let $\hat{w}_i = \max_{j \in N} \hat{p}_{ij}$ with the convention that $\hat{w}_0 = \theta^1$. Since the solution $(\hat{p}_1, \dots, \hat{p}_m)$ is feasible to problem (14), we have $\hat{p}_i \in \mathcal{G}_i(\hat{p}_{i-1})$ for all $i \in M \setminus \{1\}$ and $\hat{p}_i \in \mathcal{P}_i$ for all $i \in M$, which is equivalent to having $\hat{p}_i \in \mathcal{M}(\hat{w}_{i-1})$ for all $i \in M$. In this case, using the fact that the solution $(\hat{z}, \hat{\psi})$ is feasible to problem (16), we have $\hat{\psi}_i(\hat{w}_{i-1}) \geq V_i(\hat{p}_i)^{\gamma_i} (R_i(\hat{p}_i) - \hat{z}) + \hat{\psi}_{i+1}(\hat{w}_i)$ for all $i \in M$. Adding these inequalities over all $i \in M$

and noting that $\hat{w}_0 = \theta^1$, we obtain $\hat{\psi}_1(\theta^1) \geq \sum_{i \in M} V_i(\hat{p}_i)^{\gamma_i} (R_i(\hat{p}_i) - \hat{z}) = g(\hat{z})$, where the last equality uses the definition of $(\hat{p}_1, \dots, \hat{p}_m)$. This chain of inequalities shows that $\hat{\psi}_1(\theta^1) \geq g(\hat{z})$. As mentioned at the beginning of this paragraph, we have $\tilde{z} \geq \hat{z}$. Noting that $g(z)$ is decreasing in z , we obtain $g(\hat{z}) \geq g(\tilde{z})$. In this case, we have $\hat{\psi}_1(\theta^1) \geq g(\hat{z}) \geq g(\tilde{z}) = v_0 \tilde{z} \geq v_0 \hat{z} = \hat{\psi}_1(\theta^1)$, where the first equality uses the definition of \tilde{z} , the third inequality is by the fact that $\tilde{z} \geq \hat{z}$ and the second equality uses the fact that the solution $(\hat{z}, \hat{\psi})$ is feasible to problem (16) so that it satisfies the second constraint in this problem. Thus, all of the inequalities in the last chain of inequalities hold as equality and we obtain $g(\hat{z}) = g(\tilde{z}) = v_0 \tilde{z} = v_0 \hat{z}$, establishing that $\tilde{z} = \hat{z}$. \square

By Theorem 7, we can solve problem (16) to find the value of \hat{z} that satisfies $v_0 \hat{z} = g(\hat{z})$. Problem (16) is a linear program with $O(mq)$ decision variables and $\sum_{i \in M} O(q|\mathcal{P}_i|)$ constraints, which is tractable as long as the numbers of price vectors in the collections $\mathcal{P}_1, \dots, \mathcal{P}_m$ are reasonably small. In the rest of our discussion, we focus on how to construct a reasonably small collection of candidate price vectors \mathcal{P}_i for each nest i such that we can stitch together an optimal solution to problem (13) by picking one price vector from each one of the collections $\mathcal{P}_1, \dots, \mathcal{P}_m$. Once we have these collections, we can solve problem (16) to find \hat{z} satisfying $v_0 \hat{z} = g(\hat{z})$ and we can use Theorem 6 to obtain an optimal solution to problem (13).

2.3 Characterizing Candidate Price Vectors

In this section, we provide an alternative characterization of the optimal price vector to charge in each nest. This characterization ultimately becomes useful when we construct a collection of candidate price vectors \mathcal{P}_i for each nest i such that we can stitch together an optimal solution to problem (13) by picking one price vector from each one of the collections $\mathcal{P}_1, \dots, \mathcal{P}_m$. In the next theorem, we provide our alternative characterization of the optimal price vector to charge in each nest. This theorem is analogous to Theorem 3, but its proof is substantially more involved due to the constraints in problem (13) that link the price vectors charged in different nests. We defer the proof to the appendix.

Theorem 8 *Let (p_1^*, \dots, p_m^*) be an optimal solution to problem (13) providing the objective value z^* and set $u_i^* = \max\{\gamma_i z^* + (1 - \gamma_i) R_i(p_i^*), z^*\}$, $\ell_i^* = \min_{j \in N} p_{ij}^*$ and $w_i^* = \max_{j \in N} p_{ij}^*$. If \hat{p}_i is an optimal solution to the problem*

$$\max_{\substack{p_i \in \Theta : \\ \ell_i^* \leq p_{ij} \leq w_i^* \forall j \in N}} \left\{ V_i(p_i) (R_i(p_i) - u_i^*) \right\}, \quad (17)$$

then $(\hat{p}_1, \dots, \hat{p}_m)$ is an optimal solution to problem (13).

Theorem 8 implies that we can recover an optimal solution to problem (13) by solving problem (17) for all $i \in M$. Therefore, if we let \hat{p}_i be an optimal solution to problem (17) and use $\mathcal{P}_i = \{\hat{p}_i\}$ as the collection of candidate price vectors to charge in nest i , then we can stitch together an

optimal solution to problem (13) by picking one price vector from each one of the collections $\mathcal{P}_1, \dots, \mathcal{P}_m$. However, this approach is not immediately useful for constructing a collection of candidate price vectors, since solving problem (17) requires the knowledge of u_i^* , ℓ_i^* and w_i^* , all of which, in turn, require the knowledge of an optimal solution to problem (13). To deal with this difficulty, as a function of $u_i \in \mathfrak{R}_+$, $\ell_i \in \Theta$ and $w_i \in \Theta$, we use $\hat{p}_i(u_i, \ell_i, w_i)$ to denote an optimal solution to the problem

$$\max_{\substack{p_i \in \Theta : \\ \ell_i \leq p_{ij} \leq w_i \forall j \in N}} \left\{ V_i(p_i) (R_i(p_i) - u_i) \right\}. \quad (18)$$

In this case, if we use the collection $\mathcal{P}_i = \{\hat{p}_i(u_i, \ell_i, w_i) : u_i \in \mathfrak{R}_+, \ell_i \in \Theta, w_i \in \Theta\}$ as the collection of candidate price vectors for nest i , then we can stitch together an optimal solution to problem (13) by picking one price vector from each one of the collections $\mathcal{P}_1, \dots, \mathcal{P}_m$. To see this result, letting u_i^* , ℓ_i^* and w_i^* be as defined in Theorem 8, we have $\hat{p}_i(u_i^*, \ell_i^*, w_i^*) \in \{\hat{p}_i(u_i, \ell_i, w_i) : u_i \in \mathfrak{R}_+, \ell_i \in \Theta, w_i \in \Theta\}$ for all $i \in M$. Furthermore, since problem (18) with $u_i = u_i^*$, $\ell_i = \ell_i^*$ and $w_i = w_i^*$ is identical to problem (17), Theorem 8 implies that $(\hat{p}_1(u_1^*, \ell_1^*, w_1^*), \dots, \hat{p}_m(u_m^*, \ell_m^*, w_m^*))$ is an optimal solution to problem (13). Therefore, if we use the collection $\mathcal{P}_i = \{\hat{p}_i(u_i, \ell_i, w_i) : u_i \in \mathfrak{R}_+, \ell_i \in \Theta, w_i \in \Theta\}$ as the collection of candidate price vectors for nest i , then we can stitch together an optimal solution to problem (13) by picking one price vector from each one of the collections $\mathcal{P}_1, \dots, \mathcal{P}_m$.

Noting the discussion above, we can use $\{\hat{p}_i(u_i, \ell_i, w_i) : u_i \in \mathfrak{R}_+, \ell_i \in \Theta, w_i \in \Theta\}$ as the collection of candidate price vectors to charge in nest i . In the subsequent sections, we show that for a given $\ell_i \in \Theta$ and $w_i \in \Theta$, the collection $\{\hat{p}_i(u_i, \ell_i, w_i) : u_i \in \mathfrak{R}_+\}$ includes at most nq price vectors and we can find these price vectors in a tractable fashion. Therefore, since there are q possible values for each of ℓ_i and w_i , the collection $\{\hat{p}_i(u_i, \ell_i, w_i) : u_i \in \mathfrak{R}_+, \ell_i \in \Theta, w_i \in \Theta\}$ includes at most nq^3 price vectors.

2.4 Counting Candidate Price Vectors

In this section, we consider problem (18) for fixed values of $\ell_i \in \Theta$ and $w_i \in \Theta$. We show that there exists a collection of price vectors $\mathcal{P}_i = \{p_i^t : t \in \mathcal{T}_i\}$ such that this collection includes an optimal solution to problem (18) for any value of $u_i \in \mathfrak{R}_+$ and there are at most nq price vectors in this collection. To show this result, we write problem (18) as

$$\max_{\substack{p_i \in \Theta : \\ \ell_i \leq p_{ij} \leq w_i \forall j \in N}} \left\{ \sum_{j \in N} v_{ij}(p_{ij}) \left[\frac{\sum_{j \in N} p_{ij} v_{ij}(p_{ij})}{\sum_{j \in N} v_{ij}(p_{ij})} - u_i \right] \right\} = \max_{\substack{p_i \in \Theta : \\ \ell_i \leq p_{ij} \leq w_i \forall j \in N}} \left\{ \sum_{j \in N} (p_{ij} - u_i) v_{ij}(p_{ij}) \right\}. \quad (19)$$

In the next lemma, we begin by showing that as the value of u_i in problem (19) gets larger, then the optimal price for each product also gets larger. This lemma is similar to Lemma 4 but its proof is significantly simpler than that of Lemma 4 since the prices in problem (19) has only upper and lower bound constraints, rather than a price ladder constraint.

Lemma 9 *Using $\hat{p}_i(u_i) = (\hat{p}_{i1}(u_i), \dots, \hat{p}_{in}(u_i))$ to denote an optimal solution to problem (19) as a function of u_i , if we have $u_i^- < u_i^+$, then it holds that $\hat{p}_{ij}(u_i^-) \leq \hat{p}_{ij}(u_i^+)$ for all $j \in N$.*

Proof. To get a contradiction, assume that $u_i^- < u_i^+$, but we have $\hat{p}_{ij}(u_i^-) > \hat{p}_{ij}(u_i^+)$ for some $j \in N$. For notational brevity, we let $\hat{p}_i^- = \hat{p}_i(u_i^-)$ and $\hat{p}_i^+ = \hat{p}_i(u_i^+)$. Noting that $\hat{p}_{ij}^- > \hat{p}_{ij}^+$ and using the fact that the preference weight of a product gets larger as we charge a smaller price for the product, we obtain $v_{ij}(\hat{p}_{ij}^-) < v_{ij}(\hat{p}_{ij}^+)$. In problem (19), if we charge the price p_{ij} for product j , then this product makes a contribution of $(p_{ij} - u_i) v_{ij}(p_{ij})$ to the objective function. We note that \hat{p}_i^+ is an optimal solution to problem (19) when we solve this problem with $u_i = u_i^+$. Therefore, if we solve problem (19) with $u_i = u_i^+$, then the contribution of product j when we charge the price p_{ij}^+ for this product should be at least as large as the contribution when we charge the price p_{ij}^- . Otherwise, it would not be optimal to charge the price p_{ij}^+ for product j when we solve problem (19) with $u_i = u_i^+$. Thus, we obtain $(p_{ij}^+ - u_i^+) v_{ij}(p_{ij}^+) \geq (p_{ij}^- - u_i^+) v_{ij}(p_{ij}^-)$. Similarly, \hat{p}_i^- is an optimal solution to problem (19) when we solve this problem with $u_i = u_i^-$. Therefore, following an argument similar to the preceding one, it holds that $(p_{ij}^- - u_i^-) v_{ij}(p_{ij}^-) \geq (p_{ij}^+ - u_i^-) v_{ij}(p_{ij}^+)$. Adding the last two inequalities and canceling the common terms, we obtain $u_i^- (v_{ij}(\hat{p}_{ij}^+) - v_{ij}(\hat{p}_{ij}^-)) \geq u_i^+ (v_{ij}(\hat{p}_{ij}^+) - v_{ij}(\hat{p}_{ij}^-))$. Noting that $v_{ij}(\hat{p}_{ij}^-) < v_{ij}(\hat{p}_{ij}^+)$ by the discussion at the beginning of the proof, the last inequality implies that $u_i^- \geq u_i^+$, which is a contradiction. \square

In the next theorem, we build on the lemma above show that there exists a collection of at most nq price vectors such that this collection includes an optimal solution to problem (19) for any value of $u_i \in \mathfrak{R}_+$. The proof of this theorem uses Lemma 9 and it follows from an outline that is identical to that of Theorem 5. Thus, we omit the proof.

Theorem 10 *There exists a collection of at most nq price vectors such that this collection includes an optimal solution to problem (19) for any value of $u_i \in \mathfrak{R}_+$.*

By Theorem 10, for fixed values of $\ell_i \in \Theta$ and $w_i \in \Theta$, there exists a collection of at most nq price vectors such that this collection includes an optimal solution to problem (19) for any value of $u_i \in \mathfrak{R}_+$. In the next section, we show how to construct this collection. Since there are q possible values for each of ℓ_i and w_i , repeating our approach for all possible values of ℓ_i and w_i , it follows that there exists a collection of at most nq^3 price vectors such that this collection includes an optimal solution to problem (19) for any value of $u_i \in \mathfrak{R}_+$, $\ell_i \in \Theta$ and $w_i \in \Theta$.

2.5 Constructing Candidate Price Vectors

In the previous section, we consider problem (19) for fixed values of $\ell_i \in \Theta$ and $w_i \in \Theta$. We show that there exists a collection of at most nq price vectors such that this collection includes an optimal solution to problem (19) for any value of $u_i \in \mathfrak{R}_+$. In this section, we show how to come up with this collection in a tractable fashion. Our approach builds on a linear programming

formulation of problem (19). To give this linear programming formulation, we use the decision variables $\{x_{ij}(p_{ij}) : p_{ij} \in \Theta, j \in N\}$, where $x_{ij}(p_{ij}) = 1$ if we charge price p_{ij} for product j in nest i , otherwise $x_{ij}(p_{ij}) = 0$. In this case, problem (19) can be written as

$$\begin{aligned}
\max \quad & \sum_{j \in N} \sum_{p_{ij} \in \Theta} (p_{ij} - u_i) v_{ij}(p_{ij}) x_{ij}(p_{ij}) \\
\text{s.t.} \quad & \sum_{p_{ij} \in \Theta} x_{ij}(p_{ij}) = 1 \quad \forall j \in N \\
& x_{ij}(p_{ij}) = 0 \quad \forall p_{ij} \notin \{\ell_i, \dots, w_i\}, j \in N \\
& x_{ij}(p_{ij}) \in \{0, 1\} \quad \forall p_{ij} \in \Theta, j \in N.
\end{aligned} \tag{20}$$

In the problem above, the first set of constraints ensures that we choose one price for each product, whereas the second set of constraints ensures that the price of each product is between ℓ_i and w_i . Using the second set of constraints, we can set the values of the decision variables $\{x_{ij}(p_{ij}) : p_{ij} \notin \{\ell_i, \dots, w_i\}, j \in N\}$ to zero and drop these decision variables from problem (20). On the other hand, each row of the constraint matrix corresponding to the first set of constraints includes consecutive ones. Such a matrix is called an interval matrix and interval matrices are totally unimodular; see Nemhauser and Wolsey (1988). Therefore, we can obtain an optimal solution to problem (20) by solving its linear programming relaxation. Also, we observe that the value of $u_i \in \mathfrak{R}_+$ only affects the objective function coefficients in problem (20). Thus, we can vary $u_i \in \mathfrak{R}_+$ parametrically and solve problem (20) by using the parametric simplex method to generate the optimal solutions to this problem for all values of $u_i \in \mathfrak{R}_+$. These solutions correspond to the optimal solutions to problem (19) for all values of $u_i \in \mathfrak{R}_+$.

Therefore, for fixed values of $\ell_i \in \Theta$ and $w_i \in \Theta$, we solve problem (20) by using the parametric simplex method to generate the optimal solutions to this problem for all values of $u_i \in \mathfrak{R}_+$. Repeating this approach for all possible values of ℓ_i and w_i , we obtain the optimal solutions to problem (20) for all values of $u_i \in \mathfrak{R}_+$, $\ell_i \in \Theta$ and $w_i \in \Theta$. By the discussion that follows Theorem 8, we can use the optimal solutions to problem (20) for all values of $u_i \in \mathfrak{R}_+$, $\ell_i \in \Theta$ and $w_i \in \Theta$ as the collection of candidate price vectors \mathcal{P}_i in nest i . Once we have the collection of candidate price vectors in each nest, we can solve the linear program in (16) to find the value of \hat{z} that satisfies $v_0 \hat{z} = g(\hat{z})$. Since there are at most nq^3 price vectors in each one of the collections $\mathcal{P}_1, \dots, \mathcal{P}_m$, we have $|\mathcal{M}(w_{i-1})| = O(nq^3)$, which implies that the linear program in (16) has $O(mq)$ decision variables and $O(mnq^4)$ constraints. In this case, by Theorem 6, we can solve problem (14) with $z = \hat{z}$ to obtain an optimal solution to problem (13). To solve problem (14) with $z = \hat{z}$, we can simply solve the dynamic program in (15) with $z = \hat{z}$.

3 Numerical Experiments

In this section, we provide numerical experiments to show that the approaches in Sections 1 and 2 can obtain the optimal solutions to problems (3) and (13) reasonably fast. We also investigate the number of candidate price vectors that we construct to obtain the optimal solutions.

3.1 Price Ladders Inside Nests

In this section, we consider problem instances with price ladders inside nests. In our numerical experiments, we vary the number of nests over $m \in \{2, 4, 6\}$, the number of products in each nest over $n \in \{10, 20, 30\}$ and the number of possible prices for each product over $q \in \{10, 20, 30\}$. This setup provides 27 parameter combinations for (m, n, q) . In each parameter combination, we generate 10 individual problem instances by using the following approach. The possible prices for each product take values over the interval $[1, 10]$ and we obtain the prices $\{\theta^1, \dots, \theta^q\}$ by dividing the interval $[1, 10]$ into q equal pieces. To come up with the preference weights, we sample α_{ij} and β_{ij} from the uniform distribution over the interval $[0, 2]$ for all $i \in M, j \in N$. The preference weight of product j in nest i corresponding to the price p_{ij} is given by $\exp(\alpha_{ij} - \beta_{ij} p_{ij})$. The nested logit model has a random utility maximization interpretation, where a customer associates random utilities with the products and the no purchase option, choosing the option with the largest utility. In the random utility maximization setup, α_{ij} captures the nominal mean utility of product j in nest i and β_{ij} captures how the mean utility of product j in nest i changes as a function of its price; see McFadden (1974). We sample the dissimilarity parameter γ_i for each nest i from the uniform distribution over the interval $[0.25, 1]$. For each problem instance, we use the approach described at the end of Section 1.5 to obtain an optimal solution to problem (3).

We summarize our numerical results in Table 1. The first column in this table shows the parameter configurations for our test problems. We recall that we generate 10 individual problem instances in each parameter configuration. The second column shows the average CPU seconds to obtain an optimal solution to problem (3), where the average is computed over 10 problem instances that we generate for a particular parameter combination. The third and fourth columns respectively show the maximum and minimum CPU seconds over 10 problem instances. Similar to the average, the maximum and minimum are computed over 10 problem instances that we generate for a particular parameter combination. There are two main steps in obtaining an optimal solution to problem (3). First, we construct the collections of candidate price vectors for each nest, which requires solving problem (12) by using the parametric simplex method to generate the possible optimal solutions to this problem for all values of $u_i \in \mathfrak{R}_+$. Second, we solve problem (6) to stitch together an optimal solution by using the collection of candidate price vectors for each nest. The fifth column in Table 1 shows what percent of the CPU seconds is spent on generating the collections of candidate price vectors. The remaining portion of the CPU seconds is spent on stitching together an optimal solution. The sixth column shows the average number of price vectors in the collection that we generate for each nest, where the average is computed over all nests in a problem instance and over 10 problem instances that we generate for a particular parameter combination. The seventh and eighth columns respectively show the maximum and minimum number of price vectors in the collection that we generate for each nest.

To demonstrate the potential importance of generating the collections of candidate price vectors carefully, we also find the best solution to problem (3) that charges a constant price in each

Param. Comb. (m, n, q)	Total CPU Secs.			Perc. Time Const. Cand.	No. of Cand. Price Vectors per Nest			Perc. Opt. Gap of Const. Price in Nests		
	Avg.	Max.	Min.		Avg.	Max.	Min.	Avg.	Max.	Min.
(2, 10, 10)	0.01	0.01	0.00	91.89	23	34	10	9.05	22.98	0.01
(2, 10, 20)	0.05	0.06	0.03	96.92	52	64	34	9.24	19.45	0.01
(2, 10, 30)	0.12	0.18	0.06	98.00	73	97	35	11.40	18.90	0.00
(2, 20, 10)	0.02	0.02	0.01	95.35	29	40	12	10.62	20.04	0.01
(2, 20, 20)	0.10	0.13	0.03	98.64	55	74	14	12.38	20.84	0.01
(2, 20, 30)	0.26	0.38	0.14	99.17	78	117	41	11.17	20.65	0.01
(2, 30, 10)	0.03	0.04	0.01	96.50	30	41	13	8.72	18.37	0.02
(2, 30, 20)	0.14	0.24	0.06	98.89	52	98	22	6.33	16.28	0.01
(2, 30, 30)	0.49	0.78	0.26	99.47	98	163	48	8.71	17.45	0.03
(4, 10, 10)	0.01	0.02	0.01	92.48	22	30	11	6.86	12.87	0.00
(4, 10, 20)	0.08	0.09	0.06	97.10	43	55	34	7.01	11.57	0.04
(4, 10, 30)	0.23	0.32	0.17	98.58	66	95	48	7.20	13.43	0.02
(4, 20, 10)	0.03	0.04	0.02	96.31	25	31	15	5.24	11.19	0.01
(4, 20, 20)	0.17	0.23	0.10	98.36	50	69	29	4.18	10.51	0.02
(4, 20, 30)	0.53	0.77	0.30	99.24	77	115	43	5.65	10.85	0.01
(4, 30, 10)	0.05	0.07	0.04	97.12	30	41	22	6.88	10.46	0.01
(4, 30, 20)	0.26	0.31	0.20	98.90	51	62	38	5.97	10.19	0.02
(4, 30, 30)	0.85	1.03	0.50	99.48	84	103	47	6.99	11.00	0.02
(6, 10, 10)	0.02	0.03	0.01	91.71	21	31	10	5.17	10.93	0.00
(6, 10, 20)	0.12	0.15	0.05	97.39	42	59	20	6.27	9.88	0.00
(6, 10, 30)	0.34	0.43	0.23	98.32	65	84	45	5.90	10.79	0.02
(6, 20, 10)	0.05	0.07	0.04	95.51	28	40	19	5.96	10.11	0.01
(6, 20, 20)	0.28	0.38	0.15	98.58	53	74	29	5.55	9.40	0.00
(6, 20, 30)	0.79	1.12	0.49	99.29	78	113	45	5.11	9.64	0.01
(6, 30, 10)	0.07	0.10	0.05	97.41	27	35	21	3.98	10.83	0.01
(6, 30, 20)	0.47	0.61	0.38	99.04	59	80	47	5.81	10.74	0.00
(6, 30, 30)	1.40	1.89	1.12	99.51	89	118	72	7.26	10.93	0.02

Table 1: Computational results for test problems with price ladders inside nests.

nest. Letting $e \in \mathbb{R}^n$ be the vector of all ones, this solution can be obtained by using the collection $\mathcal{P}_i = \{\theta^1 e, \dots, \theta^q e\}$ as the collection of candidate price vectors for each nest i and solving problem (6) by using this collection for each nest i . (Li and Huh (2011) show that if $\beta_{ij} = \beta_{ik}$ for all $j, k \in N$ and $i \in M$, then it is optimal to charge a constant price in each nest.) The ninth column in Table 1 shows the average percent gap between the optimal objective value of problem (3) and the best expected revenue obtained by charging a constant price in each nest, where the average is computed over 10 problem instances that we generate for a particular parameter combination. In other words, using Opt^k to denote the optimal expected revenue for problem instance k that we generate for a particular parameter combination and ConP^k to denote the best expected revenue obtained by charging a constant price in each nest, the ninth column shows the average of the data $\{100 \times (\text{Opt}^k - \text{ConP}^k) / \text{Opt}^k : k = 1, \dots, 10\}$. The tenth and eleventh columns show the maximum and minimum percent gaps between the optimal objective value of problem (3) and the best expected revenue obtained by charging a constant price in each nest.

Our computational results indicate that we can obtain an optimal solution to problem (3) rather fast. For the largest problem instances with $m = 6$, $n = 30$ and $q = 30$, we can obtain an optimal

solution in less than two seconds and the average CPU seconds for these problem instances is about 1.4. Naturally, the CPU seconds have an increasing trend as the number of nests, products or possible prices increases. We observe that almost all of the CPU seconds are spent on constructing the collections of candidate price vectors. In Section 1.4, we show that we need to construct at most nq candidate price vectors in each nest, but our numerical results demonstrate that the number of candidate price vectors that we actually end up constructing can be substantially smaller than the upper bound of nq . For example, for the problem instances with $n = 30$ and $q = 30$, we have $nq = 900$, but the average number of candidate price vectors that we construct for each nest is about 90 and the number of candidate price vectors that we construct for each nest does not exceed 163. Charging a constant price in each nest can provide a good solution for some problem instances, but it is not too surprising to see that this approach is generally not reliable. There are numerous problem instances in Table 1 where the optimal objective value exceeds the best expected revenue obtained by charging a constant price in each nest by more than 20%.

3.2 Price Ladders Between Nests

In this section, we consider problem instances with price ladders between nests. We generate our problem instances by using the same approach that we use for generating the problem instances with price ladders inside nests. For each problem instance, we use the approach described at the end of Section 2.5 to obtain an optimal solution to problem (13). In particular, to construct the collection of candidate price vectors for each nest, we solve problem (20) through the parametric simplex method to generate the possible optimal solutions to this problem for all values of $u_i \in \mathbb{R}_+$. In this case, we solve problem (16) to find the value of \hat{z} satisfying $v_0 \hat{z} = g(\hat{z})$ and to stitch together an optimal solution by using the collection of candidate price vectors for each nest.

We summarize our numerical results in Table 2. The layout of this table is identical to that of Table 1. For the problem instances with 10 possible prices for each product, we can obtain an optimal solution in 1.34 seconds. For the largest problem instances with $m = 6$, $n = 30$ and $q = 30$, the CPU seconds are below two minutes. On average, about half of the CPU seconds is spent on constructing the collections of candidate price vectors. In Section 2.4, we show that we need to construct at most nq^3 price vectors in each nest, but we actually end up generating substantially fewer candidate price vectors. For example, for the problem instances with $n = 30$ and $q = 30$, we have $nq^3 = 810,000$, but the average number of candidate price vectors that we construct for each nest is about 25,000. Finally, the best expected revenue obtained by charging a constant price in each nest can deviate significantly from the optimal expected revenue and there are problem instances where this approach suffers optimality gaps that exceed 20%.

Param. Comb. (m, n, q)	Total CPU Secs.			Perc. Time Const. Cand.	No. of Cand. Price Vectors per Nest			Perc. Opt. Gap of Const. Price in Nests		
	Avg.	Max.	Min.		Avg.	Max.	Min.	Avg.	Max.	Min.
(2, 10, 10)	0.06	0.08	0.05	42.90	357	463	262	9.01	22.79	0.01
(2, 10, 20)	0.77	1.00	0.59	58.00	1,798	2,447	882	9.35	19.59	0.02
(2, 10, 30)	3.60	4.43	3.03	69.65	5,845	8,191	3,914	11.37	19.19	0.01
(2, 20, 10)	0.19	0.20	0.18	47.24	924	1,030	674	9.97	18.84	0.01
(2, 20, 20)	2.75	2.99	2.54	58.24	4,833	6,276	3,481	11.54	18.93	0.01
(2, 20, 30)	14.05	15.76	11.56	69.99	15,797	21,829	12,189	10.54	19.87	0.01
(2, 30, 10)	0.37	0.40	0.33	53.34	1,332	1,476	1,061	10.52	22.18	0.02
(2, 30, 20)	5.55	5.90	5.09	65.54	8,432	9,921	6,873	7.65	19.69	0.01
(2, 30, 30)	32.01	35.95	29.40	72.53	21,821	26,983	14,980	10.53	21.42	0.02
(4, 10, 10)	0.15	0.16	0.13	40.00	284	374	206	7.50	13.97	0.00
(4, 10, 20)	1.87	2.57	1.37	49.60	1,981	2,980	899	7.69	13.22	0.02
(4, 10, 30)	9.52	12.54	6.71	56.84	5,246	7,580	3,373	7.95	15.14	0.02
(4, 20, 10)	0.46	0.54	0.40	41.59	853	954	688	5.89	12.60	0.01
(4, 20, 20)	6.28	7.45	5.25	52.35	5,233	6,390	4,451	4.61	11.82	0.02
(4, 20, 30)	30.65	33.15	28.63	60.86	15,433	19,105	13,145	6.33	12.12	0.02
(4, 30, 10)	0.81	0.83	0.75	46.37	1,368	1,450	1,219	7.97	12.12	0.01
(4, 30, 20)	11.62	12.34	10.38	60.00	7,753	9,109	5,289	6.92	11.82	0.02
(4, 30, 30)	68.79	71.38	64.05	63.61	24,193	27,144	21,519	8.18	12.89	0.02
(6, 10, 10)	0.20	0.25	0.18	36.71	301	365	223	4.85	10.23	0.00
(6, 10, 20)	2.54	2.90	2.08	49.89	1,733	2,202	1,070	5.93	9.39	0.00
(6, 10, 30)	13.86	16.93	11.86	52.79	5,928	8,913	4,201	5.50	10.05	0.01
(6, 20, 10)	0.66	0.74	0.59	39.98	863	984	737	5.39	9.33	0.01
(6, 20, 20)	8.92	9.48	7.81	52.88	4,899	5,513	3,992	5.02	8.52	0.00
(6, 20, 30)	48.62	51.89	44.69	57.91	15,677	18,043	12,110	4.58	8.64	0.00
(6, 30, 10)	1.25	1.34	1.23	44.98	1,335	1,446	1,259	3.72	9.91	0.01
(6, 30, 20)	18.31	19.17	17.29	56.94	8,049	9,222	7,078	5.54	10.78	0.00
(6, 30, 30)	108.72	116.77	102.09	60.95	24,930	27,364	21,331	6.92	10.93	0.01

Table 2: Computational results for test problems with price ladders between nests.

4 Conclusions

We provided algorithms to solve pricing problems under the nested logit model when there are price ladders inside nests or between nests. There are several interesting extensions of our work. It is possible to extend our approach to the case where there are price ladders both inside nests and between nests. In this setting, if we index both the nests $M = \{1, \dots, m\}$ and the products $N = \{1, \dots, n\}$ in the order of increasing quality levels, then the prices charged for the products should satisfy $p_{11} \leq p_{12} \leq \dots \leq p_{1n} \leq p_{21} \leq p_{22} \leq \dots \leq p_{2n} \leq \dots \leq p_{m1} \leq p_{m2} \leq \dots \leq p_{mn}$. To make this extension, we observe that if we know the smallest and the largest optimal prices charged in nest i to be ℓ_i^* and w_i^* , then we can use a dynamic program similar to the one in (11) to construct the collection of candidate price vectors for nest i . Since we do not know ℓ_i^* and w_i^* , we can solve a dynamic program similar to the one in (11) for all $\ell_i \in \Theta$ and $w_i \in \Theta$, to construct the collection of candidate price vectors for nest i . To find the optimal combination of candidate price vectors for the different nests, we can use a dynamic program similar to the one in (15), where we impose the constraint that the smallest price charged in nest i should be at least as large as the largest price

charged in nest $i - 1$. Also, in our pricing problems, the set of products offered to the customers is fixed and the goal is to find the prices to charge for the products, but our results easily extend to joint assortment optimization and pricing problems, where we decide which products to offer to the customers, as well as the prices of the offered products.

Considering other extensions, as described in the introduction section, there can be quality consistency constraints that are more general than the ones considered in this paper. It is important to investigate how to construct collections of candidate price vectors under other quality consistency constraints. We can also study quality consistency constraints for pricing problems under the nested logit model with more than two stages. The linear programs that we use to combine the candidate price vectors for the different nests do not work when there are multiple stages in the nested logit model and extensions in this direction seem nontrivial. Finally, under the price ladders inside nests and between nests, we respectively have the upper bounds of nq and nq^3 for the numbers of candidate price vectors in each nest. These upper bounds demonstrate that the number of candidate price vectors that we need scales polynomially with the numbers of products and possible prices, but we can investigate whether it is possible to obtain tighter upper bounds.

References

- Chen, K. D. and Hausman, W. H. (2000), ‘Technical note: Mathematical properties of the optimal product line selection problem using choice-based conjoint analysis’, *Management Science* **46**(2), 327–332.
- Davis, J., Gallego, G. and Topaloglu, H. (2013), Assortment planning under the multinomial logit model with totally unimodular constraint structures, Technical report, Cornell University, School of Operations Research and Information Engineering.
Available at <http://legacy.orie.cornell.edu/~huseyin/publications/publications.html>.
- Dong, L., Kouvelis, P. and Tian, Z. (2009), ‘Dynamic pricing and inventory control of substitute products’, *Manufacturing & Service Operations Management* **11**(2), 317–339.
- Feldman, J. B. and Topaloglu, H. (2014), Capacity constraints across nests in assortment optimization under the nested logit model, Technical report, Cornell University, School of Operations Research and Information Engineering.
Available at <http://legacy.orie.cornell.edu/~huseyin/publications/publications.html>.
- Gallego, G. and Stefanescu, C. (2009), Upgrades, upsells and pricing in revenue management, Technical report, Columbia University.
- Gallego, G. and Topaloglu, H. (2014), ‘Constrained assortment optimization for the nested logit model’, *Management Science* **60**(10), 2583–2601.
- Gallego, G. and Wang, R. (2014), ‘Multi-product price optimization and competition under the nested attraction model’, *Operations Research* **62**(2), 450–461.
- Hanson, W. and Martin, K. (1996), ‘Optimizing multinomial logit profit functions’, *Management Science* **42**(7), 992–1003.
- Keller, P. W., Levi, R. and Perakis, G. (2014), ‘Efficient formulations for pricing under attraction demand models’, *Math. Programming A.* **145**, 223–261.
- Li, G., Rusmevichientong, P. and Topaloglu, H. (2014), ‘The d -level nested logit model: Assortment and price optimization problems’, *Operations Research* (to appear).

- 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.
- Li, H. and Huh, W. T. (2013), Pricing under the nested attraction model with a multi-stage choice structure, in ‘INFORMS Annual Meeting, Minneapolis, MN’.
- McFadden, D. (1974), Conditional logit analysis of qualitative choice behavior, in P. Zarembka, ed., ‘*Frontiers in Economics*’, Academic Press, New York, NY, pp. 105–142.
- Nemhauser, G. and Wolsey, L. (1988), *Integer and Combinatorial Optimization*, John Wiley & Sons, Inc., Chichester.
- Puterman, M. L. (1994), *Markov Decision Processes*, John Wiley and Sons, Inc., New York.
- Rayfield, Z., Rusmevichientong, P. and Topaloglu, H. (2013), ‘Approximation methods for pricing problems under the nested logit model with price bounds’, *INFORMS Journal on Computing* (to appear).
- Rusmevichientong, P., Van Roy, B. and Glynn, P. (2006), ‘A nonparametric approach to multiproduct pricing’, *Operations Research* **54**(1), 82–98.
- Song, J.-S. and Xue, Z. (2007), Demand management and inventory control for substitutable products, Technical report, Duke University, Durham, NC.
- Wang, R. (2012), ‘Capacitated assortment and price optimization under the multinomial logit model’, *Operations Research Letters* **40**, 492–497.
- Zhang, D. and Lu, Z. (2013), ‘Assessing the value of dynamic pricing in network revenue management’, *INFORMS Journal on Computing* **25**(1), 102–115.

Appendix: Proof of Theorem 8

In this section, we show Theorem 8. We need the two intermediate lemmas to show Theorem 8. In the next lemma, we show an ordering between the optimal expected revenues from a customer that has already decided to make a purchase in different nests.

Lemma 11 *If (p_1^*, \dots, p_m^*) is an optimal solution to problem (13), then we have $R_1(p_1^*) \leq R_2(p_2^*) \leq \dots \leq R_m(p_m^*)$.*

Proof. Since (p_1^*, \dots, p_m^*) is a feasible solution to problem (13), we have $p_i \in \mathcal{G}_i(p_{i-1})$, which implies that $\max_{j \in N} p_{i-1,j}^* \leq \min_{j \in N} p_{i,j}^*$. Therefore, the largest price in the price vector p_{i-1}^* is smaller than the smallest price in the price vector p_i^* . By (1), we observe that $R_{i-1}(p_{i-1}^*)$ is a convex combination of the prices in the price vector p_{i-1}^* , whereas $R_i(p_i^*)$ is a convex combination of the prices in the price vector p_i^* . Since the largest price in the price vector p_{i-1}^* is smaller than the smallest price in the price vector p_i^* , we obtain $R_{i-1}(p_{i-1}^*) \leq R_i(p_i^*)$. \square

In the next lemma, we show that if the optimal expected revenue from a customer that has already decided to make a purchase in a particular nest does not exceed the optimal expected revenue, then the smallest price in the next nest does not exceed the optimal expected revenue.

Lemma 12 *If (p_1^*, \dots, p_m^*) is an optimal solution to problem (13) providing the objective value z^* and $R_i(p_i^*) < z^*$ for some $i \in M$, then we have $i \in M \setminus \{m\}$ and $\min_{j \in N} p_{i+1,j}^* < z^*$.*

Proof. First, we show that if $R_i(p_i^*) < z^*$ for some $i \in M$, then we have $i \in M \setminus \{m\}$. To get a contradiction, assume that $R_m(p_m^*) < z^*$. By Lemma 11, we have $R_1(p_1^*) \leq R_2(p_2^*) \leq \dots \leq R_m(p_m^*) < z^*$. Thus, we obtain $\Pi(p_1^*, \dots, p_m^*) = \sum_{i \in M} V_i(p_i^*)^{\gamma_i} R_i(p_i^*) / (v_0 + \sum_{i \in M} V_i(p_i^*)^{\gamma_i}) < \sum_{i \in M} V_i(p_i^*)^{\gamma_i} z^* / (v_0 + \sum_{i \in M} V_i(p_i^*)^{\gamma_i}) < z^*$, which contradicts the fact that (p_1^*, \dots, p_m^*) is an optimal solution to problem (13).

Second, we show that if $R_i(p_i^*) < z^*$ for some $i \in M$, then $\min_{j \in N} p_{i+1,j}^* < z^*$. To get a contradiction assume that there exists a nest k such that $R_k(p_k^*) < z^*$ and $\min_{j \in N} p_{k+1,j}^* \geq z^*$. For notational brevity, we let $\ell_{k+1}^* = \min_{j \in N} p_{k+1,j}^*$. By our assumption, we have $\ell_{k+1}^* \geq z^*$. We define a solution $(\hat{p}_1, \dots, \hat{p}_m)$ to problem (13) as $\hat{p}_i = p_i^*$ for all $i \in M \setminus \{k\}$ and $\hat{p}_{kj} = \ell_{k+1}^*$ for all $j \in N$. Since the solutions (p_1^*, \dots, p_m^*) and $(\hat{p}_1, \dots, \hat{p}_m)$ charge the same prices in all nests other than nest k , we have $V_i(p_i^*)^{\gamma_i} (R_i(p_i^*) - z^*) = V_i(\hat{p}_i)^{\gamma_i} (R_i(\hat{p}_i) - z^*)$ for all $i \in M \setminus \{k\}$. For nest k , we have $R_k(p_k^*) < z^*$, but $R_k(\hat{p}_k) = \sum_{j \in N} \hat{p}_{kj} v_{kj}(\hat{p}_{kj}) / \sum_{j \in N} v_{kj}(\hat{p}_{kj}) = \sum_{j \in N} \ell_{k+1}^* v_{kj}(\hat{p}_{kj}) / \sum_{j \in N} v_{kj}(\hat{p}_{kj}) = \ell_{k+1}^* \geq z^*$. Thus, we obtain $V_k(p_k^*)^{\gamma_k} (R_k(p_k^*) - z^*) < 0 \leq V_k(\hat{p}_k)^{\gamma_k} (R_k(\hat{p}_k) - z^*)$. The discussion so far in this paragraph shows that $V_i(p_i^*)^{\gamma_i} (R_i(p_i^*) - z^*) \leq V_i(\hat{p}_i)^{\gamma_i} (R_i(\hat{p}_i) - z^*)$ for all $i \in M$ and the inequality holds as strict inequality for nest k . Adding this inequality over all $i \in M$, we have $\sum_{i \in M} V_i(p_i^*)^{\gamma_i} (R_i(p_i^*) - z^*) < \sum_{i \in M} V_i(\hat{p}_i)^{\gamma_i} (R_i(\hat{p}_i) - z^*)$. On the other hand, since (p_1^*, \dots, p_m^*) is

an optimal solution to problem (13), we have $z^* = \sum_{i \in M} V_i(p_i^*) R_i(p_i^*) / (v_0 + \sum_{i \in M} V_i(p_i^*)^{\gamma_i})$ and arranging the terms in this equality yields $v_0 z^* = \sum_{i \in M} V_i(p_i^*)^{\gamma_i} (R_i(p_i^*) - z^*)$. In this case, having $\sum_{i \in M} V_i(p_i^*)^{\gamma_i} (R_i(p_i^*) - z^*) < \sum_{i \in M} V_i(\hat{p}_i)^{\gamma_i} (R_i(\hat{p}_i) - z^*)$ and $v_0 z^* = \sum_{i \in M} V_i(p_i^*)^{\gamma_i} (R_i(p_i^*) - z^*)$ yields $v_0 z^* < \sum_{i \in M} V_i(\hat{p}_i)^{\gamma_i} (R_i(\hat{p}_i) - z^*)$. Solving for z^* in this inequality, we obtain $z^* < \sum_{i \in M} V_i(\hat{p}_i) R_i(\hat{p}_i) / (v_0 + \sum_{i \in M} V_i(\hat{p}_i)^{\gamma_i}) = \Pi(\hat{p}_1, \dots, \hat{p}_m)$. Thus, the solution $(\hat{p}_1, \dots, \hat{p}_m)$ provides an objective value for problem (13) that is strictly larger than the optimal objective value. In the rest of the proof, we show that $(\hat{p}_1, \dots, \hat{p}_m)$ is a feasible solution to problem (13), which yields a contradiction and the desired result follows.

We have $\min_{j \in N} \hat{p}_{kj} = \ell_{k+1}^* = \min_{j \in N} p_{k+1,j}^* \geq \max_{j \in N} p_{kj}^* \geq \min_{j \in N} p_{kj}^* \geq \max_{j \in N} p_{k-1,j}^* = \max_{j \in N} \hat{p}_{k-1,j}$, where the first equality uses the definition of \hat{p}_k , the second equality uses the definition of ℓ_{k+1}^* , the first and third inequalities use the fact that (p_1^*, \dots, p_m^*) is a feasible solution to problem (13) so that $p_{k+1}^* \in \mathcal{G}_{k+1}(p_k)$ and $p_k^* \in \mathcal{G}_k(p_{k-1}^*)$ and the last equality is by the definition of \hat{p}_{k-1} . Thus, this chain of inequalities shows that $\hat{p}_k \in \mathcal{G}_k(\hat{p}_{k-1})$. Similarly, we have $\min_{j \in N} \hat{p}_{k+1,j} = \min_{j \in N} p_{k+1,j}^* = \ell_{k+1}^* = \max_{j \in N} \hat{p}_{kj}$, where the first and third equalities use the definitions of \hat{p}_{k+1} and \hat{p}_k , whereas the second equality uses the definition of ℓ_{k+1}^* . Thus, this chain of equalities shows that $\hat{p}_{k+1} \in \mathcal{G}_{k+1}(\hat{p}_k)$. Since the solutions (p_1^*, \dots, p_m^*) and $(\hat{p}_1, \dots, \hat{p}_m)$ charge the same prices in all nests other than nest k and (p_1^*, \dots, p_m^*) is a feasible solution to problem (13), we have $\hat{p}_i \in \mathcal{G}_i(\hat{p}_{i-1})$ for all $i \in M \setminus \{1, k, k+1\}$ as well. Therefore, we have $\hat{p}_i \in \mathcal{G}_i(\hat{p}_{i-1})$ for all $i \in M \setminus \{1\}$, which indicates that $(\hat{p}_1, \dots, \hat{p}_m)$ is a feasible solution to problem (13). \square

In the rest of this section, we show Theorem 8.

For notational brevity, we let $R_i^* = R_i(p_i^*)$, $V_i^* = V_i(p_i^*)$, $\hat{R}_i = R_i(\hat{p}_i)$ and $\hat{V}_i = V_i(\hat{p}_i)$. First, we consider a nest i that satisfies $R_i^* < z^*$. By Lemma 12, we observe that $i \in M \setminus \{m\}$. Since \hat{p}_i is a feasible solution to problem (17), we have $\hat{p}_{ij} \leq w_i^*$ for all $j \in N$, where w_i^* is as defined in Theorem 8. We claim that $\hat{p}_{ij} = w_i^*$ for all $j \in N$. To get a contradiction, assume that $\hat{p}_{ij} < w_i^*$ for some $j \in N$. Since (p_1^*, \dots, p_m^*) is a feasible solution to problem (13), we have $p_{i+1}^* \in \mathcal{G}_{i+1}(p_i^*)$, which implies that $\min_{j \in N} p_{i+1,j}^* \geq \max_{j \in N} p_{ij}^* = w_i^*$, where the equality is by the definition of w_i^* given in Theorem 8. On the other hand, since $R_i^* < z^*$, Lemma 12 implies that $\min_{j \in N} p_{i+1,j}^* < z^*$. Therefore, we obtain $w_i^* = \max_{j \in N} p_{ij}^* \leq \min_{j \in N} p_{i+1,j}^* < z^*$. We define a solution $\tilde{p}_i = (\tilde{p}_{i1}, \dots, \tilde{p}_{in})$ to problem (17) as $\tilde{p}_{ij} = w_i^*$ for all $j \in N$. This solution is clearly feasible to problem (17) and satisfies $R_i(\tilde{p}_i) = \sum_{j \in N} w_i^* v_{ij}(p_{ij}^*) / \sum_{j \in N} v_{ij}(p_{ij}^*) = w_i^* < z^*$. Furthermore, we have $R_i(\hat{p}_i) = \sum_{j \in N} \hat{p}_{ij} v_{ij}(\hat{p}_{ij}) / \sum_{j \in N} v_{ij}(\hat{p}_{ij}) \leq \sum_{j \in N} w_i^* v_{ij}(\hat{p}_{ij}) / \sum_{j \in N} v_{ij}(\hat{p}_{ij}) = w_i^* = R_i(\tilde{p}_i)$. By the last two chains of inequalities, we get $z^* - R_i(\hat{p}_i) \geq z^* - R_i(\tilde{p}_i) = z^* - w_i^* > 0$. Noting that the preference weight of a product becomes smaller as we charge a larger price, since $\hat{p}_{ij} \leq w_i^* = \tilde{p}_{ij}$ for all $j \in N$ and the inequality is strict for some $j \in N$, it holds that $v_{ij}(\hat{p}_{ij}) \geq v_{ij}(\tilde{p}_{ij})$ for all $j \in N$ and the inequality is strict for some $j \in N$. Thus, adding the last inequality over all $j \in N$, we obtain $V_i(\hat{p}_i) > V_i(\tilde{p}_i)$. In this case, having $z^* - R_i(\hat{p}_i) \geq z^* - R_i(\tilde{p}_i) > 0$ and $V_i(\hat{p}_i) > V_i(\tilde{p}_i)$ implies that $V_i(\hat{p}_i)(z^* - R_i(\hat{p}_i)) > V_i(\tilde{p}_i)(z^* - R_i(\tilde{p}_i))$. Since $R_i^* < z^*$, we have $u_i^* = z^*$ by the definition of u_i^* , in which case, the last inequality can equivalently be written as

$V_i(\hat{p}_i) (R_i(\hat{p}_i) - u_i^*) < V_i(\tilde{p}_i) (R_i(\tilde{p}_i) - u_i^*)$, which contradicts the fact that \hat{p}_i is an optimal solution to problem (17). Thus, our claim holds and we have $\hat{p}_{ij} = w_i^*$ for all $j \in N$.

By the claim established in the previous paragraph, we have $\hat{p}_{ij} = w_i^*$ for all $j \in N$. Noting that $w_i^* = \max_{j \in N} p_{ij}^*$ by the definition of w_i^* , we have $\hat{p}_{ij} = w_i^* \geq p_{ij}^*$ for all $j \in N$. Since the preference weight of a product becomes smaller as we charge a larger price, the last inequality implies that $v_{ij}(\hat{p}_{ij}) \leq v_{ij}(p_{ij}^*)$ for all $j \in N$. Adding this inequality over all $j \in N$, we obtain $V_i(\hat{p}_i) \leq V_i(p_i^*)$. Furthermore, we have $R_i(\hat{p}_i) = \sum_{j \in N} w_i^* v_{ij}(\hat{p}_{ij}) / \sum_{j \in N} v_{ij}(\hat{p}_{ij}) = w_i^* = \max_{j \in N} p_{ij}^* \leq \min_{j \in N} p_{i+1,j}^* < z^*$, where the first inequality is by the fact that (p_1^*, \dots, p_m^*) is a feasible solution to problem (13) so that $p_{i+1}^* \in \mathcal{G}_{i+1}(p_i^*)$ and the second inequality follows by noting that $R_i^* < z^*$ and using Lemma 12. Since $w_i^* \geq p_{ij}^*$ for all $j \in N$, we have $R_i(p_i^*) \leq \sum_{j \in N} w_i^* v_{ij}(\hat{p}_{ij}) / \sum_{j \in N} v_{ij}(\hat{p}_{ij}) = w_i^* = R_i(\hat{p}_i)$ as well. The last two chains of inequalities show that $z^* - R_i(p_i^*) \geq z^* - R_i(\hat{p}_i) = z^* - w_i^* > 0$. In this case, having $V_i(\hat{p}_i) \leq V_i(p_i^*)$ and $z^* - R_i(p_i^*) > z^* - R_i(\hat{p}_i) > 0$ yields $V_i(p_i^*)^{\gamma_i} (z^* - R_i(p_i^*)) > V_i(\hat{p}_i)^{\gamma_i} (z^* - R_i(\hat{p}_i))$. The last inequality shows that $(V_i^*)^{\gamma_i} (R_i^* - z^*) < \hat{V}_i^{\gamma_i} (\hat{R}_i - z^*)$ for each nest i that satisfies $R_i^* < z^*$.

Second, we consider a nest i that satisfies $R_i^* \geq z^*$. In this case, we can follow the same argument at the beginning of the proof of Theorem 3 to show that $(V_i^*)^{\gamma_i} (R_i^* - z^*) \leq \hat{V}_i^{\gamma_i} (\hat{R}_i - z^*)$ for each nest i that satisfies $R_i^* \geq z^*$. Therefore, we obtain $(V_i^*)^{\gamma_i} (R_i^* - z^*) \leq \hat{V}_i^{\gamma_i} (\hat{R}_i - z^*)$ for all $i \in M$. Adding this inequality over all $i \in M$, we have $\sum_{i \in M} (V_i^*)^{\gamma_i} (R_i^* - z^*) \leq \sum_{i \in M} \hat{V}_i^{\gamma_i} (\hat{R}_i - z^*)$. Since (p_1^*, \dots, p_m^*) is an optimal solution to problem (13), we have $z^* = \sum_{i \in M} (V_i^*)^{\gamma_i} R_i^* / (v_0 + \sum_{i \in M} (V_i^*)^{\gamma_i})$. Arranging the terms in this equality, it follows that $v_0 z^* = \sum_{i \in M} (V_i^*)^{\gamma_i} (R_i^* - z^*)$, in which case, we have $v_0 z^* = \sum_{i \in M} (V_i^*)^{\gamma_i} (R_i^* - z^*) \leq \sum_{i \in M} \hat{V}_i^{\gamma_i} (\hat{R}_i - z^*)$. Focusing on the first and last terms in this chain of inequalities and solving for z^* , we get $z^* \leq \sum_{i \in M} \hat{V}_i^{\gamma_i} \hat{R}_i / (v_0 + \sum_{i \in M} \hat{V}_i^{\gamma_i}) = \Pi(\hat{p}_1, \dots, \hat{p}_m)$. Thus, the solution $(\hat{p}_1, \dots, \hat{p}_m)$ provides an expected revenue that is at least as large as the optimal objective value of problem (13). In the rest of the proof, we show that $(\hat{p}_1, \dots, \hat{p}_m)$ is a feasible solution to problem (13), which establishes that $(\hat{p}_1, \dots, \hat{p}_m)$ is an optimal solution to problem (13).

Consider nest $i \in M \setminus \{1\}$. Noting that \hat{p}_i is a feasible solution to problem (17), we obtain $\hat{p}_{ij} \geq \ell_i^*$ and $\hat{p}_{i-1,j} \leq w_{i-1}^*$ for all $j \in N$, which imply that $\min_{j \in N} \hat{p}_{ij} \geq \ell_i^*$ and $\max_{j \in N} \hat{p}_{i-1,j} \leq w_{i-1}^*$. Since (p_1^*, \dots, p_m^*) is a feasible solution to problem (13), we also have $p_i^* \in \mathcal{G}_i(p_{i-1}^*)$, which implies that $w_{i-1}^* = \max_{j \in N} p_{i-1,j}^* \leq \min_{j \in N} p_{ij}^* = \ell_i^*$. Therefore, we obtain $\max_{j \in N} \hat{p}_{i-1,j} \leq w_{i-1}^* \leq \ell_i^* \leq \min_{j \in N} \hat{p}_{ij}$. The last inequality shows that $\hat{p}_i \in \mathcal{G}_i(\hat{p}_{i-1})$. Since our choice of nest i is arbitrary, we have $\hat{p}_i \in \mathcal{G}_i(\hat{p}_{i-1})$ for all $i \in M \setminus \{1\}$.