Assortment Optimization with Replacement Options for Retail Platforms with Stockout Risk

Dmitry Mitrofanov[1], Huseyin Topaloglu[2], Yuheng Wang[2]
[1]Carroll School of Management, Boston College, Chestnut Hill, MA 02467, USA
[2]School of Operations Research and Information Engineering, Cornell Tech, New York, NY 10044, USA
mitrofan@bc.edu, topaloglu@orie.cornell.edu, yw634@cornell.edu

May 15, 2024

There has been a notable surge of online platforms that operate as market facilitators without having direct control over the inventory of the products they sell. When an online platform operates as a market facilitator, the products offered to a customer may have a significant probability of being out of stock when the order placed by the customer is being fulfilled. Therefore, such online platforms often prompt the customers to pick not only a preferred option, but also a replacement option, in case their preferred option is out of stock when fulfilling the order. We study assortment optimization problems when the customers are offered assortments of products among which they pick their preferred and replacement options. If the preferred option is available, then they are provided with the preferred option. If the preferred option is not available, but the replacement option is, then they are provided with the replacement option. Otherwise, the customers are provided with no products. We use a version of the multinomial logit model to capture the choice process of the customers. We consider two variants of the problem. In the non-adaptive variant, we compute one assortment of products among which the customers pick their both preferred and replacement options. In the adaptive variant, the assortment of products among which the customer chooses her replacement option depends on the preferred option she picked. We show that both variants are NP-hard. Our main technical contributions are a polynomial-time approximation scheme for the adaptive variant and a fully polynomial-time approximation scheme for the non-adaptive variant. Using data from Instacart, we demonstrate that explicitly modeling the presence of a replacement option and using an adaptive strategy to offer assortments for the replacement option can provide significant revenue benefits.

Key words : online platforms, retail, assortment optimization, stockout risk, replacement options.

1. Introduction

Online platforms that primarily function as delivery service providers do not have direct control over the inventory of the products they sell. Therefore, the product availability at the time when a customer places an order on these online platforms may be different from the product availability at the time when the order is fulfilled, creating a significant probability of being out of stock for the products in the order. In this case, the online platform risks the possibility of offering a product to a customer that it ultimately cannot fulfill. To mitigate this possibility, it is becoming common for online platforms to ask their customers to specify a replacement option along with their preferred option so that the replacement option can be used to fulfill the customer's order in case the preferred option is out of stock at the time of order fulfillment. The practice of asking for replacement options primarily originated from selling groceries, but it has also been adopted for other product categories. In Figure 1, we show shopping carts on Instacart, Amazon Fresh,

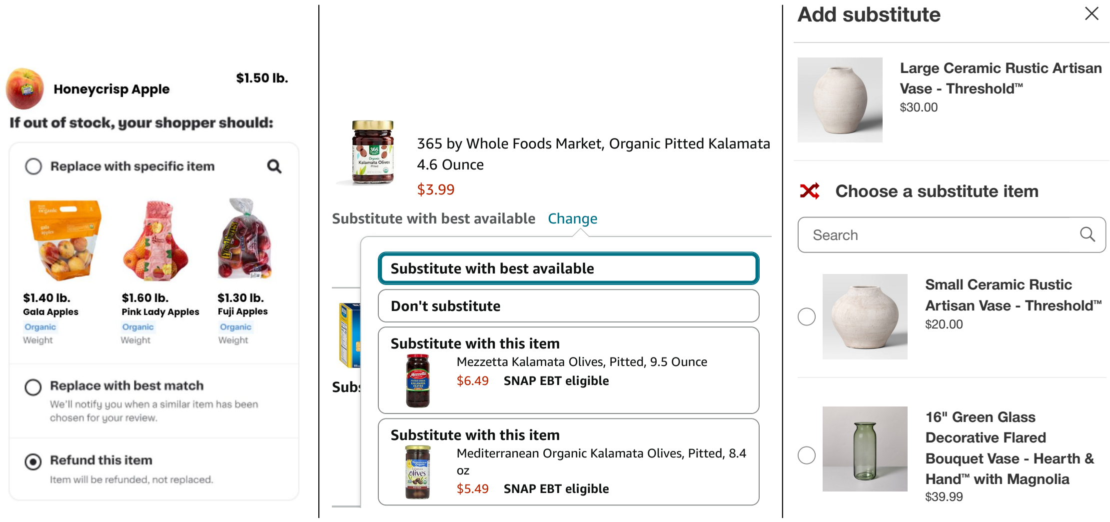

Figure 1 Shopping carts with replacement options at Instacart (left), Amazon Fresh (middle) and Target.com (right). Images for illustrative purpose only.

and Target.com, all of which allow specifying a replacement option. The carts on Instacart and Amazon Fresh include groceries, but the cart on Target.com includes household goods.

We consider assortment optimization problems when there is a risk that the products are out of stock at the time of order fulfillment, so the customers are asked to specify replacement options. In our model, the customers are offered two assortments. They specify their preferred option within the first assortment, whereas they specify their replacement option within the second assortment. We impose the constraint that the second assortment is included in the first assortment so that a product that is a candidate for the replacement option must have been a candidate for the preferred option, to begin with. We use a variant of the multinomial logit model to capture the choice process of the customers. A customer associates Gumbel distributed utilities with the products and the no-purchase option. Among the products in the first assortment, she specifies the one with the largest utility as the preferred option. Among the products in the second assortment, the customer specifies the one with the largest utility as the replacement option. Noting that the first assortment includes the second one, if the utility of the no-purchase option is larger than the utilities of all products in the first assortment, then the customer leaves without a purchase, whereas if the utility of the no-purchase option is smaller than the utility of a product in the first assortment but larger than the utilities of all products in the second assortment, then the customer does not specify a replacement option. At the order fulfillment time, if the preferred option is available, then we fulfill the preferred option. If the preferred option is not available, but the replacement option is, then we fulfill the replacement option. We collect the revenue corresponding to the fulfilled option.

We consider two variants of our assortment optimization problem. In the non-adaptive variant, we compute one assortment of products among which the customer picks her both preferred and replacement options. In the adaptive variant, the assortment of products among which the customer chooses her replacement option depends on the preferred option she picked. In both variants, our goal is to maximize the expected revenue from a customer. We give an expression for the expected revenue as a function of the assortments offered for the choices of preferred and replacement options. We show that both the adaptive and non-adaptive variants are NP-hard. We characterize the revenue gap between adaptive and non-adaptive variants. We give a polynomial-time approximation scheme (PTAS) for the adaptive variant. We give a fully polynomial-time approximation scheme (FPTAS) for the non-adaptive variant. Using data from a large grocery delivery platform, we demonstrate the importance of explicitly incorporating the possibility of replacement options. Computing the revenue-maximizing assortment without modeling the replacement options and offering this assortment as potential possibilities for both preferred and replacement options results in an average expected revenue loss of more than 3.31% across all product categories. Moreover, switching from the non-adaptive to the adaptive approach yields an average expected revenue improvement of about 0.94%, the improvement exceeding 4% for some product categories.

Main Contributions: We discuss our modeling and algorithmic contributions, along with the insights from our case study on data from a grocery delivery platform.

Assortment Optimization with Replacement Options and Hardness. We formulate assortment optimization problems when the customers specify replacement options next to their preferred options so that we use the replacement option to fulfill the customer order whenever the preferred option is not available. To our knowledge, our formulation represents the first work to study replacement options when the products are subject to stockout risk. We derive the expected revenue as a function of the assortments offered as possible candidates for the preferred and replacement options. The form of the expected revenue function utilizes the properties of the Gumbel distributed utilities. For both the adaptive and non-adaptive variants of our assortment optimization problem, we show that the problem of finding the revenue-maximizing assortment is NP-hard.

Comparing Adaptive and Non-Adaptive Solutions. We show that computing the assortment for the replacement option as a function of the preferred option of a customer can increase the expected revenue by at most a factor of two. In other words, we establish an upper bound of two on the value of using an adaptive solution, as opposed to a non-adaptive one. We give a problem instance for which the adaptive solution improves the expected revenue from the non-adaptive solution by a factor that is arbitrarily close to 4/3, yielding a lower bound on the value of using an adaptive solution. This problem instance leaves a gap between the upper and lower bounds on the value of using an adaptive solution, which we were not able to close, but it demonstrates that the value of an adaptive solution can be significant. Our computational experiments with a grocery delivery platform also confirm this observation. In the process of establishing an upper bound of two on the value of using an adaptive solution, we show that we can obtain a very simple $\frac{1}{2}$-approximate solution for both the adaptive and non-adaptive variants by ignoring the replacement option and solving an assortment optimization problem under the standard multinomial logit model. However, the practical performance of such a naive approach is quite poor, so we turn our attention to developing more sophisticated approximation schemes.

PTAS for the Adaptive Variant. Our main technical contribution is a PTAS for the adaptive variant of our assortment optimization problem. We can interpret the adaptive variant as an assortment optimization problem with recourse. In particular, we offer two assortments sequentially, among which a customer picks her preferred and replacement options, but the second assortment, among which the customer picks her replacement option can depend on the preferred option of the customer. The presence of such a recourse possibility brings unique challenges, so there are several novel ideas in the design of our PTAS. We refer to the product of the revenue, availability probability, and preference weight of a product as its weight. In our PTAS, given $\epsilon \in (0,1]$, we guess the set of $\lceil 1/\epsilon \rceil$ products with the largest weights in the optimal assortment offered for the preferred option. We refer to this set of products as the critical set. We formulate a surrogate problem, where the assortment offered for the replacement option depends on the preferred option of the customer only when the preferred option is in the critical set. If the preferred option of the customer is not in the critical set, then the assortment offered for the replacement option does not depend on the preferred option of the customer. We show that the surrogate problem approximates the adaptive variant within a factor $1/(1+\epsilon)$. Furthermore, we show that we can formulate the surrogate problem as a multi-dimensional knapsack problem with $\lceil \frac{1}{\epsilon} \rceil + 3$ constraints. Solving this knapsack problem is difficult, but we use a dynamic program to obtain a super-optimal solution, which is to say that we obtain a solution that provides at least the optimal objective value of the knapsack problem but allows violations of the constraints by a factor of $1+\epsilon$. The dynamic program to obtain a super-optimal solution uses the ideas pioneered by Desire et al. (2020), but the construction of the surrogate problem and its reformulation as a knapsack problem are novel. Our PTAS obtains a $(1-\epsilon)$-approximate solution in a running time of $O((\frac{n}{\epsilon})^{O(1/\epsilon)})$.

FPTAS for the Non-Adaptive Variant. We give an FPTAS for the non-adaptive variant. The development of the FPTAS for the non-adaptive variant is more direct. In the non-adaptive variant, we compute one assortment of products, among which the customers choose both their preferred and replacement options. Consequently, we guess the total weight of the products in the optimal solution, in which case, the problem of finding the revenue-maximizing assortment that is consistent with the guess can be formulated as a multi-dimensional knapsack problem with two constraints.

We use a dynamic program to obtain a super-optimal solution to this knapsack problem. Our FPTAS obtains a $(1-\epsilon)$-approximate solution in a running time of $O(\frac{n^7(\log n)^2}{\epsilon^4})$.

Case Study on Data from a Grocery Delivery Platform. Carrying out a case study on data from a large grocery delivery platform, we demonstrate that explicitly modeling the availability of replacement options provides an average expected revenue improvement of 3.31%. Moreover, the expected revenue benefit of using adaptive solutions, as opposed to non-adaptive ones, is another 0.94% over all product categories. Not so surprisingly, we find that solving a model that plans the assortments for preferred and replacement options also increases the fulfillment rates, boosting the likelihood that customers are not left empty-handed by as much as 3.69% for some product categories. The improvements in expected revenues and fulfillment rates are particularly noticeable for product categories with larger stockout risks. Lastly, we test the numerical performance of our PTAS on randomly generated problem instances with favorable outcomes.

Related Literature: There is a large body of work on assortment optimization under the multinomial logit model. In this setting, each product has a fixed revenue. Customers choose among the products according to the multinomial logit model. We want to find a revenue-maximizing assortment. Talluri and van Ryzin (2004) show that the optimal assortment is revenue-ordered in the sense that it includes a certain number of products with the largest revenues. Thus, we can find the revenue-maximizing assortment by checking the expected revenue from each revenue-ordered assortment. Rusmevichientong et al. (2010) study the cardinality-constrained variant with an upper bound on the number of products that can be offered and give a polynomial-time algorithm. Sumida et al. (2021) work with constraints that can be encoded using a totally unimodular matrix and formulate the assortment optimization problem as a linear program.

A natural extension of the multinomial logit model is the mixture of multinomial logit models, where there are multiple customer types and customers of different types choose according to different multinomial logit models. Bront et al. (2009) show that the assortment problem under a mixture of multinomial logit models is NP-hard and develop heuristics. Rusmevichientong et al. (2014) show that the problem is NP-hard even with two customer types and give guarantees for revenue-ordered assortments. Berbeglia and Joret (2020) generalize some of these guarantees to general choice models that are driven by the random utility maximization principle. Desir et al. (2022) show that the problem is hard to approximate within a factor of $1/m$, where m is the number of customer types. By relating the assortment problem to a multi-dimensional knapsack problem, the authors give an FPTAS when the number of customer types is fixed.

There are other enhancements for the multinomial logit model. Feldman and Topaloglu (2018) incorporate nested consideration sets, where customers eliminate products based on one feature such as price. Aouad et al. (2018) give a PTAS when each product is included in the consideration set independently. Jagabathula and Vulcano (2018) use a model where customers eliminate products using a precedence graph and choose among the remaining ones according to a choice model. Jagabathula et al. (2022) operationalize this model by focusing on the case where the choice model is the multinomial logit model. Wagner and Martinez-de-Albeniz (2020) incorporate product exchanges. Chen and Mitrofanov (2023) use general consideration sets, but the choice within a consideration set is according to the multinomial logit model with equal preference weights.

We offer two assortments sequentially, where the first assortment includes the second one. Flores et al. (2019) consider a problem, where the assortment is gradually revealed in two stages, so the two assortments are non-overlapping and the customer decides whether to stop the search at each stage. Liu et al. (2020) generalize this model to multiple stages. Gao et al. (2021) give a variant of the same problem that is consistent with utility maximization. Feldman and Segev (2022) give sharpened performance guarantees for the same problem. Bai et al. (2023) allow multiple purchases within an assortment. These papers do not involve a recourse decision, as each assortment in the sequence is pre-fixed. El Housni and Topaloglu (2023) focus on a setting with multiple customer types, where one chooses an assortment to stock under a cardinality constraint and specializes this assortment further based on the type of the arriving customer. The structure of their problem is different from ours, as their problem is trivial without a cardinality constraint in the first stage.

There are empirical studies on the effect of stockouts on customer satisfaction and ways of mitigating the negative effects of stockouts. Fitzsimons (2000) carries out laboratory experiments to verify that customers are negatively impacted by the unavailability, especially for products with personal commitments. Anderson et al. (2006) use data to demonstrate that customers cut back the consumption of items with frequent unavailability and companies can mitigate the impact of stockouts through more effective communication strategies. Musalem et al. (2010) introduce a choice model that incorporates product availability as a factor and delve into the problem of model estimation, but not assortment optimization. Knight and Mitrofanov (2022) carry out a field experiment at Instacart and find that disclosing low product availability results in increased customer satisfaction. The preceding work is empirical in nature, whereas we focus on algorithmic strategies to make effective assortment optimization decisions under product unavailability.

Organization: In Section 2, we formulate the adaptive and non-adaptive variants and characterize the hardness of both variants. In Section 3, we compare the expected revenues for the two variants. In Section 4, we develop the PTAS for the adaptive variant by constructing the surrogate problem and relating it to the adaptive variant. In Section 5, we show how to approximate the surrogate problem. In Section 6, we develop the FPTAS for the non-adaptive variant. In Section 7, we give our case study on the data from a grocery delivery platform. In Section 8, we test the numerical performance of our PTAS. In Section 9, we conclude.

2. Problem Formulation

We have n products indexed by $N = \{1, 2, \ldots, n\}$. The revenue of product i is f_i. The availability probability of product i is q_i. Customers follow the random utility maximization principle to choose among the products. We use the random variable U_i to capture the utility of product i, which has the Gumbel distribution with location and scale parameters $(\mu_i, 1)$. Letting $v_i = \exp(\mu_i)$, we refer to v_i as the preference weight of product i. For any subset of products $S \subseteq N$, we use $V(S) = \sum_{i \in S} v_i$ to capture the total preference weights of the products in the subset. We use the random variable U_0 to capture the utility of the no-purchase option, which has the Gumbel distribution with location and scale parameters $(0, 1)$. Following the standard assumption for the multinomial logit model, the utilities of the products and the no-purchase option are all independent. Furthermore, the availabilities of the products and utilities of the products are all independent as well.

The choice process of a customer proceeds in two stages. The customer associates utilities with the products and no-purchase option, all sampled from their corresponding distributions. In the first stage, the customer is offered an assortment of products $S_0 \subseteq N$. Among the products in the offered assortment and the no-purchase option, the customer chooses the alternative with the largest utility. If the customer chooses a product in the first stage, then this product is her preferred option. In the second stage, if the preferred option of the customer was product i, then the customer is offered an assortment of products $S_i \subseteq S_0$ with $i \notin S_i$. Among the products in the offered assortment and the no-purchase option, the customer chooses the alternative with the largest utility. If the customer chooses a product in the second stage, then this product is her replacement option. In this way, the customer forms her preferred and replacement options.

At fulfillment time, if the preferred option is available, then the customer receives and pays for her preferred choice. If the preferred option is not available but the replacement option is available, then the customer receives and pays for her replacement option. Our goal is to pick assortments to offer in the first and second stages to maximize the expected revenue from a customer. We make two observations regarding the choice process in the previous paragraph. First, the assortment offered in the second stage is a subset of the assortment offered in the first stage. In this way, we ensure that a product cannot be offered as a replacement option unless it is offered as a preferred option. Thus, there are no "surprise" products in the second stage. Second, the assortment offered in the second stage can depend on the preferred option of the customer. Therefore, we can adapt the assortment offered in the second stage based on the preferred option of the customer.

If the customer chooses the no-purchase option in the first stage, then the utility of the no-purchase option is larger than the utilities of all products offered in the first stage. In this case, because the assortment offered in the second stage is a subset of the assortment offered in the first stage, the utility of the no-purchase option is larger than the utilities of all products that could be offered in the second stage as well. Thus, if a customer chooses the no-purchase option in the first stage, then there is no need to pick an assortment to offer in the second stage. We use two properties of Gumbel random variables to compute the expected revenue from a customer. We consider a sequence of independent Gumbel random variables $\{X_1, X_2, \ldots, X_n\}$, where X_i has the Gumbel distribution with location and scale parameters $(\mu_i, 1)$.

Maximum of Gumbels: We have $\mathbb{P}\{X_1 \geq X_2\} = e^{\mu_1}/(e^{\mu_1} + e^{\mu_2})$ and $\max\{X_1, X_2, \ldots, X_n\}$ has the Gumbel distribution with the location and scale parameters $(\log(e^{\mu_1} + \ldots, e^{\mu_n}), 1)$, so

$$\mathbb{P}\Big\{X_1 \geq \max\{X_2, X_3, \ldots, X_n\}\Big\} = \frac{e^{\mu_1}}{e^{\mu_1} + \ldots + e^{\mu_2}}. \tag{1}$$

Conditional Ranking: Given that X_1 is the largest among $\{X_1, X_2, \ldots, X_n\}$, the conditional ranking of $\{X_2, X_3, \ldots, X_n\}$ has the same distribution as the unconditional ranking, so

$$\mathbb{P}\Big\{X_2 \geq X_3 \geq \ldots \geq X_n \;\Big|\; X_1 \geq \max\{X_2, X_3, \ldots, X_n\}\Big\} = \mathbb{P}\Big\{X_2 \geq X_3 \geq \ldots \geq X_n\Big\}. \tag{2}$$

By (1), noting that $v_i = e^{\mu_i}$, if we offer the assortment $S_0 \subseteq N$ in the first stage, then the customer picks product $i \in S_0$ with probability $v_i/1 + V(S_0)$. By (2), given the customer picks product i in the first stage as her preferred option, if we offer the assortment $S_i \subseteq N$ with $i \notin S_i$ in the second stage, then the conditional choice process of the customer in the second stage is identical to the unconditional choice process. Therefore, if the customer picks product i in the first stage as her preferred option and we offer the assortment $S_i \subseteq N$ with $i \notin S_i$ in the second stage, then the customer picks product $j \in S_i$ as her replacement option with probability $v_j/1 + V(S_i)$.

We use $(S_0, S_1, \ldots, S_n) \in 2^N \times 2^N \times \ldots \times 2^N$ to capture our assortment decisions, where S_0 is the assortment that we offer in the first stage and S_i is the assortment that we offer in the second stage to a customer with product i for her preferred option. Our assortment decisions satisfy $S_i \subseteq S_0 \setminus \{i\}$ for all $i \in N$, so the assortment in the second stage is a subset of the assortment in the first stage and we do not offer product i in the second stage to a customer with product i for her preferred option. We compute the expected revenue from a customer when our assortment decisions are given by $(S_0, S_1, \ldots, S_n)$. By the discussion in the previous paragraph, under these assortment decisions, the customer picks product $i \in S_0$ for per preferred option and product $j \in S_i$ for her replacement option with probability $\frac{v_i}{1+V(S_0)} \times \frac{v_j}{1+V(S_i)}$. In this case, if product i is available, then we obtain a revenue of f_i. If product i is not available, but product j is available, then we obtain a revenue of f_j. Similarly, the customer picks product $i \in S_0$ for per primary option and the no-purchase option for her replacement option with probability $\frac{v_i}{1+V(S_0)} \times \frac{1}{1+V(S_i)}$. In this case, if product i

is available, then we obtain a revenue of f_i. Therefore, if our assortment decisions are given by $(S_0, S_1, \ldots, S_n)$, then the expected revenue that we obtain from a customer is

$$\begin{aligned} \mathrm{R}(S_0, S_1, \ldots, S_n) &= \sum_{i \in S_0} \sum_{j \in S_i} \frac{v_i}{1+V(S_0)} \, \frac{v_j}{1+V(S_i)} \left[q_i \, f_i + (1-q_i) \, q_j \, f_j \right] + \sum_{i \in S_0} \frac{v_i}{1+V(S_0)} \, \frac{1}{1+V(S_i)} \, q_i \, f_i \\ &= \sum_{i \in S_0} \frac{v_i}{1+V(S_0)} \left(q_i \, f_i + (1-q_i) \sum_{j \in S_i} \frac{v_j}{1+V(S_i)} \, q_j \, f_j \right), \end{aligned} \tag{3}$$

where the first equality follows from the two cases discussed earlier in this paragraph and the second equality follows by arranging the terms.

We consider two variants of our problem that differ in the set of feasible decisions. We refer to the first variant as the adaptive one. In the adaptive variant, we can adjust the assortment offered in the second stage as a function of the preferred option of the customer in the first stage. Therefore, the assortment decisions $(S_0, S_1, \ldots, S_n)$ have to satisfy $S_i \subseteq S_0 \setminus \{i\}$ for all $i \in N$. We refer to the second variant as the non-adaptive one. In the non-adaptive variant, we do not adjust the assortment offered in the second stage as a function of the preferred option of the customer in the first stage. If we offer the assortment S_0 in the first stage and the customer picks product i in the first stage, then the assortment offered in the second stage is simply $S_0 \setminus \{i\}$. Therefore, the assortment decisions $(S_0, S_1, \ldots, S_n)$ have to satisfy $S_i = S_0 \setminus \{i\}$ for all $i \in N$. In the adaptive variant, we adjust the assortment offered in the second stage based on the choices in the first stage. By doing so, we can increase the expected revenue obtained from the customer, but implementing the solution for the adaptive variant in practice requires more work. Furthermore, the products offered in the first stage as alternatives for the preferred option are not necessarily all available in the second stage as alternatives for the replacement option, which may confuse some customers. In the non-adaptive variant, all products offered in the first stage as alternatives for the preferred option are available in the second stage as alternatives for the replacement option, but such inflexibility comes at a loss of expected revenue. To succinctly capture both variants, we impose the constraint $S_i \in \mathcal{F}_i(S_0)$ for all $i \in N$ on our assortment decisions. Setting $\mathcal{F}_i(S_0) = \{S \subseteq N : S \subseteq S_0 \setminus \{i\}\}$ yields the adaptive variant, whereas setting $\mathcal{F}_i(S_0) = \{S_0 \setminus \{i\}\}$ yields the non-adaptive variant. To maximize the expected revenue from a customer, we need to solve

$$\max_{\substack{(S_0, S_1, \ldots, S_n) \in 2^N \times 2^N \times \ldots \times 2^N, \\ S_i \in \mathcal{F}_i(S_0) \; \forall \, i \in N}} \mathrm{R}(S_0, S_1, \ldots, S_n), \tag{4}$$

where the set of feasible solutions may capture the adaptive or non-adaptive variant. We will study solution algorithms for both variants.

In Appendix A, we show that problem (4) is NP-hard for the adaptive and non-adaptive variants by using a reduction from the partition problem. Thus, we focus on approximation strategies.

3. Comparison of the Adaptive and Non-Adaptive Variants

The adaptive variant allows adjusting the assortment offered in the second stage as a function of the preferred option of the customer, whereas the non-adaptive variant does not allow such flexibility. Thus, the expected revenue for the adaptive variant is at least as large as the expected revenue for the non-adaptive one. We show that the expected revenue for the adaptive variant cannot exceed the expected revenue for the non-adaptive one by more than a factor of two. In other words, the value of adaptivity is at most two. Directly building on the approach that we use to show this result, we can also give an approach to get quick $\frac{1}{2}$-approximate solutions for both the adaptive and non-adaptive variants. While this approach has a constant-factor performance guarantee, it is rather naive and does not perform well, so we spend the rest of the paper on more sophisticated approximation schemes. For notational brevity, throughout the paper, we set $r_i = q_i\, f_i$, which can be interpreted as the adjusted revenue from product i when we take its availability into consideration. Furthermore, we set $p_i = 1 - q_i$, which corresponds to the unavailability probability of product i. In this case, the expected revenue function in (3) becomes

$$\mathrm{R}(S_0, S_1, \ldots, S_n) = \sum_{i \in S_0} \frac{v_i}{1+V(S_0)} \left(r_i + p_i \frac{\sum_{j \in S_i} v_j\, r_j}{1+V(S_i)} \right). \tag{5}$$

We will need the expected revenue function above when all products are available with $p_i = 0$ for all $i \in N$, so we let $\overline{\mathrm{R}}(S_0) = \frac{\sum_{i\in S_0} v_i\, r_i}{1+V(S_0)}$. In the next proposition, we bound the value of adaptivity.

Proposition 3.1 *Letting Z^*_{AD} and Z^*_{NA} be the optimal objective values of problem* (4)*, respectively, for the adaptive and non-adaptive variants, we have $Z^*_{\mathsf{NA}} \le Z^*_{\mathsf{AD}} \le 2\, Z^*_{\mathsf{NA}}$.*

Proof: We express (5) equivalently as $\mathrm{R}(S_0, S_1, \ldots, S_n) = \overline{\mathrm{R}}(S_0) + \sum_{i\in S_0} \frac{v_i}{1+V(S_0)}\, p_i\, \overline{\mathrm{R}}(S_i)$. Letting $S^* = \arg\max_{S \subseteq N} \overline{\mathrm{R}}(S)$, we make three observations. First, noting the equivalent expression for (5) along with the definition of S^*, for any assortment decisions $(S_0, S_1, \ldots, S_n)$, we have $\mathrm{R}(S_0, S_1, \ldots, S_n) \le \overline{\mathrm{R}}(S^*) + \sum_{i \in S_0} \frac{v_i}{1+V(S_0)}\, p_i\, \overline{\mathrm{R}}(S^*) \le 2\overline{\mathrm{R}}(S^*)$. Second, noting the equivalent expression for (5) once more, we have $\mathrm{R}(S_0, S_1, \ldots, S_n) \ge \overline{\mathrm{R}}(S_0)$ for any assortment decisions $(S_0, S_1, \ldots, S_n)$. Third, the solution $(S^*, S^* \setminus \{1\}, \ldots, S^* \setminus \{n\})$ is feasible for the non-adaptive variant. Letting $(\widehat{S}_0, \widehat{S}_1, \ldots, \widehat{S}_n)$ be an optimal solution to the adaptive variant, we get

$$Z^*_{\mathsf{AD}} \;=\; \mathrm{R}(\widehat{S}_0, \widehat{S}_1, \ldots, \widehat{S}_n) \;\overset{(a)}{\le}\; 2\, \overline{\mathrm{R}}(S^*) \;\overset{(b)}{\le}\; 2\, \mathrm{R}(S^*, S^* \setminus \{1\}, \ldots, S^* \setminus \{n\}) \;\overset{(c)}{\le}\; 2\, Z^*_{\mathsf{NA}}, \tag{6}$$

where (a), (b), and (c) use the first, second, and third observations, establishing the second inequality in the proposition. The first inequality in the proposition holds trivially. ■

By the proposition above, by switching from non-adaptive assortments to adaptive assortments, we can increase the expected revenue by at most a factor of two. Furthermore, the proof of the proposition provides quick $\frac{1}{2}$-approximate solutions for both the adaptive and non-adaptive variants. Letting $S^* = \arg\max_{S \subseteq N} \overline{\mathrm{R}}(S)$, by (6), we have $\mathrm{R}(S^*, S^* \setminus \{1\}, \ldots, S^* \setminus \{n\}) \geq \frac{1}{2} Z^*_{\mathsf{AD}}$. Thus, the solution $(S^*, S^* \setminus \{1\}, \ldots, S^* \setminus \{n\})$ yields at least half of the optimal expected revenue for the adaptive variant. Noting that the solution $(S^*, S^* \setminus \{1\}, \ldots, S^* \setminus \{n\})$ is feasible for the adaptive variant, this solution is a $\frac{1}{2}$-approximation for the adaptive variant. On the other hand, because the optimal expected revenue for the adaptive variant is at least as large as the one for the non-adaptive variant, we have $\mathrm{R}(S^*, S^* \setminus \{1\}, \ldots, S^* \setminus \{n\}) \geq \frac{1}{2} Z^*_{\mathsf{AD}} \geq \frac{1}{2} Z^*_{\mathsf{NA}}$, in which case, noting that the solution $(S^*, S^* \setminus \{1\}, \ldots, S^* \setminus \{n\})$ is feasible for the non-adaptive variant, this solution is a $\frac{1}{2}$-approximation for the non-adaptive variant as well. We can compute the assortment S^* efficiently. In particular, noting the definition of $\overline{\mathrm{R}}(S)$, the assortment S^* maximizes the expected revenue obtained from only one stage. Talluri and van Ryzin (2004) show that the assortment S^* includes a certain number of products with the largest adjusted revenues. In other words, indexing the products such that $r_1 \geq r_2 \geq \ldots \geq r_n$, the assortment S^* is of the form $\{1, \ldots, i\}$ for some $i \in N$. Therefore, we can find S^* by checking the expected revenue $\overline{\mathrm{R}}(\{1, \ldots, i\})$ for each $i \in N$ and choosing the assortment of the form $\{1, \ldots, i\}$ that provides the largest expected revenue.

According to the discussion above, the optimal expected revenues for the adaptive and non-adaptive variants differ by at most a factor or two. Thus, the value of adaptivity is upper bounded by two. To complement this result, we give a problem instance such that the optimal expected revenues for the adaptive and non-adaptive variants differ by a factor arbitrarily close to $4/3$, yielding a lower bound on the value of adaptivity. This problem instance does not establish that our characterization of the value of adaptivity is tight, but it shows that the value of switching to an adaptive solution from a non-adaptive one can be significant. The challenge in giving a tight characterization of the value of adaptivity appears to be that the proof of Proposition 3.1 is, intuitively speaking, based on fully foregoing the expected revenue in either of the stages under the non-adaptive solution, but it is difficult to give a problem instance where the non-adaptive solution incurs such a loss. Despite our best efforts, we are not able to tighten our bound on the value of adaptivity in either direction and we have to leave a tight characterization open. Nevertheless, our problem instance demonstrates that the value of adaptivity can be significant. We consider a problem instance with $n+1$ products. For each of the first n products, we have $r_i = \frac{1}{4}$, $v_i = \epsilon$ and $p_i = 1 - \epsilon$ for $i = 1, \ldots, n$. For the last product, we have $r_{n+1} = 1$, $v_{n+1} = 1$ and $p_{n+1} = 0$.

To lower bound the optimal expected revenue for the adaptive variant, consider offering all products in the first stage. If the preferred option of the customer is one of the first n products, then we offer only product $n+1$ in the second stage. If the preferred option of the customer is product $n+1$, then we offer the empty assortment in the second stage. Thus, this solution corresponds to the assortment decisions $(\widehat{S}_0, \widehat{S}_1, \ldots, \widehat{S}_{n+1})$ with $\widehat{S}_0 = \{1, \ldots, n+1\}$, $\widehat{S}_i = \{n+1\}$ for all $i = 1, \ldots, n$ and $\widehat{S}_{n+1} = \varnothing$. Using (5), the expected revenue from the solution $(\widehat{S}_0, \widehat{S}_1, \ldots, \widehat{S}_{n+1})$ satisfies

$$\begin{aligned} \mathrm{R}(\widehat{S}_0, \widehat{S}_1, \ldots, \widehat{S}_{n+1}) &= n \, \frac{\epsilon}{1 + n\epsilon + 1} \left(\frac{1}{4} + (1-\epsilon) \, \frac{1}{1+1} \right) + \frac{1}{1 + n\epsilon + 1} \\ &= \frac{n\epsilon}{2 + n\epsilon} \left(\frac{3}{4} - \frac{\epsilon}{2} \right) + \frac{1}{2 + n\epsilon} \ \geq \ \frac{n\epsilon}{2 + n\epsilon} \left(\frac{3}{4} - \frac{\epsilon}{2} \right). \end{aligned}$$

Thus, the optimal expected revenue for the adaptive variant is at least $\frac{n\epsilon}{2+n\epsilon}\,(\frac{3}{4} - \frac{\epsilon}{2})$. To upper bound the optimal expected revenue for the non-adaptive variant, consider any solution that does not offer product $n+1$ in the first stage. Because the revenues of the first n products are $\frac{1}{4}$, even if we obtain revenue from the preferred and replacement options of the customer, the expected revenue from a customer cannot exceed $\frac{1}{2}$. Thus, the expected revenue from any solution that does not offer product $n+1$ in the first stage is at most $\frac{1}{2}$. On the other hand, consider a solution that offers product $n+1$ in the first stage. In this case, the only question for the non-adaptive variant is how many of the first n products, along with product $n+1$, we offer in the first stage. Consider offering k of the first n products, along with product $n+1$, in the first stage. If the customer picks product $i \in \{1, \ldots, k, n+1\}$ as her preferred option, then she chooses among the products $\{1, \ldots, k, n+1\} \setminus \{i\}$ in the second stage. Therefore, this solution corresponds to the assortment decisions $(\widetilde{S}_0, \widetilde{S}_1, \ldots, \widetilde{S}_{n+1})$ with $\widetilde{S}_0 = \{1, \ldots, k, n+1\}$, $\widetilde{S}_i = \{1, \ldots, k, n+1\} \setminus \{i\}$ for $i = 1, \ldots, k, n+1$ and $\widetilde{S}_i = \varnothing$ for $i = k+1, \ldots, n$. Note that $\frac{x}{2+x}$ is increasing in $x \geq 0$. Also, by cross-multiplying the numerators and denominators of the fractions, we can check that $\frac{1-\epsilon}{2+(k-1)\epsilon} \leq \frac{1}{2+k\epsilon}$. In this case, using (5), the expected revenue from the solution $(\widetilde{S}_0, \widetilde{S}_1, \ldots, \widetilde{S}_{n+1})$ satisfies

$$\begin{aligned} \mathrm{R}(\widetilde{S}_0, \widetilde{S}_1, \ldots, \widetilde{S}_{n+1}) &= k \, \frac{\epsilon}{1 + k\epsilon + 1} \left(\frac{1}{4} + (1-\epsilon) \, \frac{(k-1)\epsilon \frac{1}{4} + 1}{1 + (k-1)\epsilon + 1} \right) + \frac{1}{1 + k\epsilon + 1} \\ &= \frac{k\epsilon}{2 + k\epsilon} \left(\frac{1}{4} + (1-\epsilon) \, \frac{\frac{1}{4}\,(k-1)\epsilon}{2 + (k-1)\epsilon} + \frac{1-\epsilon}{2 + (k-1)\epsilon} \right) + \frac{1}{2 + k\epsilon} \\ &\leq \frac{k\epsilon}{2 + k\epsilon} \left(\frac{1}{4} + \frac{\frac{1}{4}\,k\epsilon}{2 + k\epsilon} + \frac{1}{2 + k\epsilon} \right) + \frac{1}{2 + k\epsilon} \end{aligned}$$

Consider the function $g(x) = \frac{x}{2+x}(\frac{1}{4} + \frac{\frac{1}{4}x}{2+x} + \frac{1}{2+x}) + \frac{1}{2+x}$. We can check that $g'(x) = \frac{2-x}{2\,(2+x)^3}$, so $g(x)$ attains its maximum at $x = 2$. Therefore, we have $\mathrm{R}(\widetilde{S}_0, \widetilde{S}_1, \ldots, \widetilde{S}_{n+1}) \leq g(2) = \frac{9}{16}$.

Using Z^*_{AD} and Z^*_{NA} to, respectively, denote the optimal expected revenues of the adaptive and non-adaptive variants, we have $Z^*_{\mathsf{AD}} \geq \frac{n\epsilon}{2+n\epsilon}\,(\frac{3}{4} - \frac{\epsilon}{2})$ and $Z^*_{\mathsf{NA}} \leq \max\{\frac{1}{2}, \frac{9}{16}\} = \frac{9}{16}$. Choosing $\epsilon = \frac{1}{\sqrt{n}}$, we get $Z^*_{\mathsf{AD}}/Z^*_{\mathsf{NA}} \geq \frac{\sqrt{n}}{2+\sqrt{n}}\,(\frac{3}{4} - \frac{1}{2\sqrt{n}})\big/\frac{9}{16}$. If the number of products is arbitrarily large, then the right side of the inequality becomes arbitrarily close to $\frac{4}{3}$, yielding a problem instance for which the optimal expected revenues for the adaptive and non-adaptive variants differ by a factor arbitrarily close to $\frac{4}{3}$. In our computational experiments, we use a realistic dataset to demonstrate that switching from non-adaptive to adaptive assortments can significantly increase the expected revenue in practice as well. Closing this section, setting $S^* = \arg\max_{S \subseteq N} \overline{\mathrm{R}}(S)$, the proof of Proposition 3.1 also implies that the solution $(S^*, S^* \setminus \{1\}, \ldots, S^* \setminus \{n\})$ provides at least half of the optimal expected revenue for the adaptive and non-adaptive variants. We give a problem instance for which the expected revenue from this solution differs from the optimal expected revenue by a factor arbitrarily close to $\frac{1}{2}$. We have two products with $r_1 = 1$, $v_1 = 1$, $p_1 = 0$ and $r_2 = \frac{1}{2} - \epsilon$, $v_2 = \frac{1}{\epsilon}$, $p_2 = 1 - \epsilon$. To compute S^*, recall that it is enough to focus on assortments of the form $\{1, i\}$ for $i = 1, 2$. We have $\overline{\mathrm{R}}(\{1\}) = \frac{1}{1+1} = \frac{1}{2}$ and $\overline{\mathrm{R}}(\{1,2\}) = \frac{1 + \frac{1}{\epsilon}(\frac{1}{2} - \epsilon)}{1 + 1 + \frac{1}{\epsilon}} < \frac{1}{2}$, which implies that $S^* = \{1\}$. In this case, the expected revenue provided by the solution $(S^*, S^* \setminus \{1\}, S^* \setminus \{2\})$ is $\mathrm{R}(\{1\}, \varnothing, \varnothing) = \frac{1}{1+1} = \frac{1}{2}$. On the other hand, consider the solution $(\{1,2\}, \{2\}, \{1\})$ for the adaptive and non-adaptive variants. The expected revenue from this solution is

$$\begin{aligned}
\mathrm{R}(\{1,2\},\{2\},\{1\}) \; &= \; \frac{1}{1+1+1/\epsilon} + \frac{1/\epsilon}{1+1+1/\epsilon}\left(\frac{1}{2} - \epsilon + (1-\epsilon)\,\frac{1}{1+1}\right) \\
&= \; \frac{1}{2+1/\epsilon} + \frac{1}{1+2\epsilon}\left(1 - \frac{3}{2}\epsilon\right) \; \geq \; \frac{1}{1+2\epsilon}\left(1 - \frac{3}{2}\epsilon\right).
\end{aligned}$$

Thus, we have $\frac{\mathrm{R}(S^*, S^* \setminus \{1\}, S^* \setminus \{2\})}{Z^*_{\mathsf{AD}}} \leq \frac{\mathrm{R}(S^*, S^* \setminus \{1\}, S^* \setminus \{2\})}{Z^*_{\mathsf{NA}}} \leq \frac{\mathrm{R}(S^*, S^* \setminus \{1\}, S^* \setminus \{2\})}{\mathrm{R}(\{1,2\},\{2\},\{1\})} \leq \frac{1}{2} \Big/ \frac{1}{1+2\epsilon}(1 - \frac{3}{2}\epsilon)$. If we choose ϵ arbitrarily close to zero, then the last expression becomes arbitrarily close to $\frac{1}{2}$.

4. Approximation Scheme for the Adaptive Variant

We give a PTAS for the adaptive variant. For any $\epsilon > 0$, our PTAS yields a $(1-\epsilon)$-approximate solution to the adaptive variant in a running time of $O((\frac{n}{\epsilon})^{O(1/\epsilon)})$. We go through a number of steps to construct our PTAS. We construct an approximate version of the expected revenue function that differs from the expected revenue function by a factor of $1+\epsilon$. Maximizing the approximate objective function is still difficult, but we use the structure of the approximate objective function to maximize the approximate objective function through a multi-dimensional knapsack problem. We find an approximate solution to the multi-dimensional function that violates the constraints of the multi-dimensional knapsack problem by at most a factor of $1+\epsilon$. We use this solution to construct an approximate solution to the adaptive variant. To pursue this outline, we start with an alternative expression for the expected revenue function. Recalling that $\overline{\mathrm{R}}(S) = \frac{\sum_{i \in S} v_i r_i}{1 + V(S)}$ is the expected revenue from the assortment $S \subseteq N$ when all products are available, we write the expected revenue function in (5) as $\mathrm{R}(S_0, S_1, \ldots, S_n) = \sum_{i \in S_0} \frac{v_i}{1+V(S_0)}\,(r_i + p_i\,\overline{\mathrm{R}}(S_i))$. For notational brevity, we also define $\overline{\mathrm{F}}(S) = \max_{Q \subseteq S} \overline{\mathrm{R}}(Q)$, which is the optimal expected revenue when all products are available and the universe of products is S. In the adaptive variant, the assortment decisions $(S_0, S_1, \ldots, S_n)$ have to satisfy $S_i \subseteq S_0 \setminus \{i\}$. In this case, the adaptive variant is given by

$$\begin{aligned} Z^*_{\mathsf{AD}} &= \max_{\substack{(S_0, S_1, \ldots, S_n) \in 2^N \times 2^N \times \ldots \times 2^N, \\ S_i \subseteq S_0 \setminus \{i\} \; \forall i \in N}} \left\{ \sum_{i \in S_0} \frac{v_i}{1 + V(S_0)} \Big(r_i + p_i \, \overline{\mathrm{R}}(S_i) \Big) \right\} \\ &= \max_{S_0 \subseteq N} \left\{ \sum_{i \in S_0} \frac{v_i}{1 + V(S_0)} \left(r_i \; + \; p_i \max_{S_i \subseteq S_0 \setminus \{i\}} \overline{\mathrm{R}}(S_i) \right) \right\} \\ &= \max_{S \subseteq N} \left\{ \sum_{i \in S} \frac{v_i}{1 + V(S)} \Big(r_i + p_i \, \overline{\mathrm{F}}(S \setminus \{i\}) \Big) \right\}. \end{aligned} \tag{7}$$

In this way, we formulate the adaptive variant by using a single assortment $S \subseteq N$ as the decision variable, as opposed to $n+1$ assortments $(S_0, S_1, \ldots, S_n)$ in our original formulation.

Considering the sum in the objective function of problem (7), the subset S can have as many as n products, so this sum can involve as many as n terms. For any $\epsilon > 0$, we construct an approximate version of the objective function with $O(1/\epsilon)$ terms in the analogue of this sum. This approximate version of the objective function will differ from the objective function of problem (7) by at most a factor of $1 + \epsilon$. In this case, to obtain an approximate solution to problem (7), we formulate a surrogate problem that maximizes the approximate version of the objective function. The surrogate problem is still difficult to solve, but using the fact that the sum in the objective function of the surrogate problem involves $O(1/\epsilon)$ terms, it turns out that we can formulate the surrogate problem as a multi-dimensional knapsack problem with $O(1/\epsilon)$ constraints. In this case, we use a dynamic program to find an approximate solution to the multi-dimensional knapsack problem, while allowing violations of the constraints of the multi-dimensional knapsack problem by a factor of $1+\epsilon$. This approximate solution to the multi-dimensional knapsack problem is enough to construct an approximate solution to the surrogate problem, in which case, we use the approximate solution to the surrogate problem to construct an approximate solution to the adaptive variant. Following this outline, we obtain the next theorem. The proof of this theorem uses the development in this and next section, so we give the proof of the theorem at the end of the next section.

Theorem 4.1 (PTAS for Adaptive Variant) *There exists an algorithm that provides a $(1-\epsilon)$-approximate solution to the adaptive variant in a running time of $O((\frac{n}{\epsilon})^{O(1/\epsilon)})$.*

We proceed to give an approximate version of the objective function of problem (7), which allows us to formulate the surrogate problem.

Constructing the Surrogate Problem:

Corresponding to the approximation guarantee that we target in our PTAS, we fix $\epsilon > 0$. To construct our approximate objective function, we define three sets of products. We use the set S^* to denote an optimal solution to the adaptive variant in (7). Letting $z^* = \max_{S \subseteq S^*} \overline{\mathrm{R}}(S)$, we partition the set S^* into the sets $S^*_{\mathsf{H}} = \{i \in S^* : r_i \geq z^*\}$ and $S^*_{\mathsf{L}} = \{i \in S^* : r_i < z^*\}$. *As our first set*, we define $U = \{i \in N : r_i \geq z^*\}$, so we can use U to capture the set of potential products that could be in S^*_{H}. Because we do not know the value of z^*, we do not know the set U either, but noting that U includes a certain number of products with the largest adjusted revenues, there are n possibilities for U, so we can try each possibility. Therefore, we proceed with the understanding that we know the set U. By the definitions of the sets S^*_{H} and U, we have $S^*_{\mathsf{H}} \subseteq U$.

Using the set U, we define two other sets. We refer to $v_i r_i$ as the weight of product i. Using $\lceil \cdot \rceil$ to denote the roundup function, *as our second set*, we define M_{H} as the set of $\lceil 1/\epsilon \rceil$ products in S^*_{H} that has the largest weights. Thus, we have $\min_{i \in M_{\mathsf{H}}} v_i r_i \geq \max_{i \in S^*_{\mathsf{H}} \setminus M_{\mathsf{H}}} v_i r_i$. Because we do not know the set S^*_{H}, we do not know the set M_{H} either, but noting that $M_{\mathsf{H}} \subseteq S^*_{\mathsf{H}} \subseteq U$, M_{H} is a subset of U with cardinality $\lceil 1/\epsilon \rceil$. Because we proceed with the understanding that we know the set U, there are $O(n^{\lceil 1/\epsilon \rceil})$ possibilities for M_{H}, so we can try each possibility. Thus, we proceed with the understanding that we know the set M_{H} as well. Lastly, the set M_{H} includes the $\lceil 1/\epsilon \rceil$ products in S^*_{H} that have the largest weights. To capture the set of potential remaining products that could be in S^*_{H}, *as our third set*, we define the set $Q_{\mathsf{H}} = \{i \in U \setminus M_{\mathsf{H}} : v_i r_i \leq \min_{j \in M_{\mathsf{H}}} v_j r_j\}$. We proceed with the understanding that we know the sets U and M_{H}, so we know the set Q_{H}.

Note that we have $S^*_{\mathsf{H}} \subseteq M_{\mathsf{H}} \cup Q_{\mathsf{H}}$ by our construction. To see this inclusion, assume on the contrary that there exists some product i such that $i \in S^*_{\mathsf{H}}$ and $i \notin M_{\mathsf{H}} \cup Q_{\mathsf{H}}$. Because $i \in S^*_{\mathsf{H}}$, $i \notin M_{\mathsf{H}}$ and $S^*_{\mathsf{H}} \subseteq U$, we must have $i \in U \setminus M^*_{\mathsf{H}}$. In this case, because $i \notin Q_{\mathsf{H}}$, by the definition of Q_{H}, we must have $v_i r_i > \min_{j \in M_{\mathsf{H}}} v_j r_j$. Having $i \in S^*_{\mathsf{H}}$ and $v_i r_i > \min_{j \in M_{\mathsf{H}}} v_j r_j$ contradicts the fact that M_{H} includes a certain number of products with the largest weights in S^*_{H}, so the inclusion holds. Thus, letting $N_{\mathsf{H}} = M_{\mathsf{H}} \cup Q_{\mathsf{H}}$, we have $S^*_{\mathsf{H}} \subseteq N_{\mathsf{H}}$. On the other hand, letting $N_{\mathsf{L}} = \{i \in N : r_i < z^*\}$, we have $S^*_{\mathsf{L}} \subseteq N_{\mathsf{L}}$. Noting that $N_{\mathsf{L}} = N \setminus U$, because we proceed with the understanding that we know the set U, we know the set N_{L}. The preceding discussion implicitly assumes that there are at least $\lceil 1/\epsilon \rceil$ products in the set S^*_{H}, but if there are fewer than $\lceil 1/\epsilon \rceil$ products in the set S^*_{H}, then all of our development continues to hold with no modifications by setting $N_{\mathsf{L}} = \varnothing$.

To recap the essential elements of the preceding construction that will be critical in our subsequent development, we constructed the sets N_{H} and N_{L} such that $S^*_{\mathsf{H}} \subseteq N_{\mathsf{H}}$ and $S^*_{\mathsf{L}} \subseteq N_{\mathsf{L}}$. We can express the set N_{H} as $N_{\mathsf{H}} = M_{\mathsf{H}} \cup Q_{\mathsf{H}}$, where the set M_{H} includes the $\lceil 1/\epsilon \rceil$ products in S^*_{H} that have the largest weights. In general, we do not know the sets M_{H}, Q_{H} and N_{L}, but we can come up with $n \times O(n^{\lceil 1/\epsilon \rceil}) = O(n^{O(1/\epsilon)})$ possibilities such that one of these possibilities provides all of the sets M_{H}, Q_{H} and N_{L}. Therefore, throughout this section, we proceed with the understanding that we know the sets M_{H}, Q_{H}, and N_{L}. When implementing our PTAS, we will try each possibility.

We use the sets M_{H}, Q_{H}, and N_{L} to give an approximate version of (7), which we refer to as the surrogate problem. In particular, the surrogate problem is given by

$$Z^*_{\mathsf{SR}} = \max_{\substack{S \subseteq N:\\ M_{\mathsf{H}} \subseteq S \subseteq N_{\mathsf{H}} \cup N_{\mathsf{L}}}} \left\{ \sum_{i \in M_{\mathsf{H}}} \frac{v_i}{1+V(S)} \Big(r_i + p_i \,\overline{\mathrm{F}}(S \setminus \{i\}) \Big) + \sum_{i \in S \setminus M_{\mathsf{H}}} \frac{v_i}{1+V(S)} \Big(r_i + p_i \,\overline{\mathrm{R}}(S \cap N_{\mathsf{H}}) \Big) \right\}. \quad (8)$$

We can interpret the surrogate problem in (8) as an approximation to the adaptive variant in (7), where we use a mixture of adaptive and non-adaptive solutions. In particular, noting the first sum in (8), if a customer picks a product in M_{H} as her preferred option in the first stage, then we adapt the assortment offered in the second stage to her preferred option. We offer the assortment $\arg\max_{Q \subseteq S\setminus\{i\}} \overline{\mathrm{R}}(Q)$ in the second stage to a customer with preferred option $i \in M_{\mathsf{H}}$, in which case, we obtain an expected revenue of $\overline{\mathrm{F}}(S \setminus \{i\}) = \max_{Q\subseteq S\setminus\{i\}} \overline{\mathrm{R}}(Q)$. On the other hand, noting the second sum in (8), if a customer picks a product in $S \setminus M_{\mathsf{H}}$ as her preferred option in the first stage, then we do not adapt the assortment offered in the second stage to her preferred option. We offer the assortment $S \cap N_{\mathsf{H}}$ in the second stage to a customer with preferred option $i \in S \setminus M_{\mathsf{H}}$, in which case, we obtain an expected revenue of $\overline{\mathrm{R}}(S \cap N_{\mathsf{H}})$. Note that the assortment $S \cap N_{\mathsf{H}}$ does not depend on the preferred option of the customer. For product $i \in M_{\mathsf{H}}$, noting that we have $M_{\mathsf{H}} \subseteq S^*_{\mathsf{H}} = \{j \in S^* : r_j \geq z^*\}$ by our construction, we have $r_i \geq z^*$. Furthermore, by the definition of M_{H}, the weight of the product is one of the largest $\lceil 1/\epsilon \rceil$ weights among the products in S^*_{H}. Thus, the products in M_{H} have both favorable adjusted revenues and weights. Making adaptive assortment decisions for these products turns out to be enough to obtain a performance guarantee. Lastly, because $|M_{\mathsf{H}}| = \lceil 1/\epsilon \rceil$, there are $O(1/\epsilon)$ terms in the first sum in (8). We give two lemmas that compare the objective functions of problems (7) and (8).

In the next lemma, we show that the optimal objective value problem (8) upper bounds the optimal objective value of problem (7).

Lemma 4.2 (Upper Bound from Surrogate) *Noting that Z^*_{AD} and Z^*_{SR} are, respectively, the optimal objective values of problems* (7) *and* (8)*, we have $Z^*_{\mathsf{SR}} \geq Z^*_{\mathsf{AD}}$.*

Proof: Let S^* be an optimal solution to (7). We partition S^* into S^*_{H} and S^*_{L} as done earlier. By our construction of the sets M^*_{H}, N_{H} and N_{L}, we have $M_{\mathsf{H}} \subseteq S^*_{\mathsf{H}}$, $S^*_{\mathsf{H}} \subseteq N_{\mathsf{H}}$ and $S^*_{\mathsf{L}} \subseteq N_{\mathsf{L}}$, so $M^*_{\mathsf{H}} \subseteq S^*_{\mathsf{H}} \subseteq S^*$ and $S^* = S^*_{\mathsf{H}} \cup S^*_{\mathsf{L}} \subseteq N_{\mathsf{H}} \cup N_{\mathsf{L}}$. Therefore, the solution S^* is feasible to (8). Also, because $S^* = S^*_{\mathsf{H}} \cup S^*_{\mathsf{L}}$, $S^*_{\mathsf{H}} \subseteq N_{\mathsf{H}}$, $S^*_{\mathsf{L}} \subseteq N_{\mathsf{L}}$ and $N_{\mathsf{H}} \cap N_{\mathsf{L}} = \varnothing$, we have $S^* \cap N_{\mathsf{H}} = S^*_{\mathsf{H}}$. Lastly, by a standard result for assortment optimization under the multinomial logit model, setting $\widehat{z} = \max_{Q \subseteq S} \overline{\mathrm{R}}(Q)$, an optimal solution to the problem $\max_{Q\subseteq S} \overline{\mathrm{R}}(Q)$ is given by $\{i \in S : r_i \geq \widehat{z}\}$; see Theorem 5.1 in Gallego and Topaloglu (2019). Thus, noting our definition $z^* = \max_{S \subseteq S^*} \overline{\mathrm{R}}(S)$ and $S^*_{\mathsf{H}} = \{i \in S^* : r_i \geq z^*\}$, we have $\overline{\mathrm{R}}(S^*_{\mathsf{H}}) = \max_{S\subseteq S^*} \overline{\mathrm{R}}(S)$, so we get $\overline{\mathrm{R}}(S^*_{\mathsf{H}}) = \max_{S\subseteq S^*} \overline{\mathrm{R}}(S) \geq \max_{S\subseteq S^*\setminus\{i\}} \overline{\mathrm{R}}(S) = \overline{\mathrm{F}}(S^* \setminus \{i\})$. In this case, because the solution S^* is feasible to problem (8), we get

$$\begin{aligned} Z^*_{\mathsf{SR}} \;\geq\; & \sum_{i\in M_{\mathsf{H}}} \frac{v_i}{1+V(S^*)} \Big(r_i + p_i\,\overline{\mathrm{F}}(S^*\setminus\{i\})\Big) + \sum_{i\in S^*\setminus M_{\mathsf{H}}} \frac{v_i}{1+V(S^*)} \Big(r_i + p_i\,\overline{\mathrm{R}}(S^*\cap N_{\mathsf{H}})\Big) \\ & \geq \sum_{i\in S^*} \frac{v_i}{1+V(S^*)} \Big(r_i + p_i\,\overline{\mathrm{F}}(S^*\setminus\{i\})\Big) \;=\; Z^*_{\mathsf{AD}}, \end{aligned}$$

where the second inequality holds because $S^*\cap N_{\mathsf{H}} = S^*_{\mathsf{H}}$ and $\overline{\mathrm{R}}(S^*_{\mathsf{H}}) \geq \overline{\mathrm{F}}(S^*\setminus\{i\})$ and the equality holds because S^* is an optimal solution to problem (8).

In the next lemma, we complement the previous result by showing that $\frac{1}{1+\epsilon}$ fraction of the objective function of problem (8) lower bounds the objective function of problem (7).

Lemma 4.3 (Lower Bound from Surrogate) *Letting $\mathcal{R}_{\mathsf{AD}}(S)$ and $\mathcal{R}_{\mathsf{SR}}(S)$, respectively, be the objective functions of problems* (7) *and* (8)*, for any $S \subseteq N$ that satisfies $M_{\mathsf{H}} \subseteq S \subseteq N_{\mathsf{H}} \cup N_{\mathsf{L}}$, we have $\mathcal{R}_{\mathsf{AD}}(S) \geq \frac{1}{1+\epsilon}\,\mathcal{R}_{\mathsf{SR}}(S)$.*

Proof: Consider an assortment $S \subseteq N$ that satisfies $M_{\mathsf{H}} \subseteq S \subseteq N_{\mathsf{H}} \cup N_{\mathsf{L}}$. For any $i \in S\setminus M_{\mathsf{H}}$, we claim that we have $\overline{\mathrm{F}}(S\setminus\{i\}) \geq \frac{1}{1+\epsilon}\,\overline{\mathrm{R}}(S\cap N_{\mathsf{H}})$. Because $S \subseteq N_{\mathsf{H}}\cup N_{\mathsf{L}}$, if $i \in S\setminus M_{\mathsf{H}}$, then we have either $i \in (S\cap N_{\mathsf{H}})\setminus M_{\mathsf{H}}$ or $i \in (S\cap N_{\mathsf{L}})\setminus M_{\mathsf{H}}$. We consider the two cases separately. First, consider the case $i \in (S\cap N_{\mathsf{H}})\setminus M_{\mathsf{H}}$. Because $N_{\mathsf{H}} = M_{\mathsf{H}} \cup Q_{\mathsf{H}}$, having $i \in (S\cap N_{\mathsf{H}})\setminus M_{\mathsf{H}}$ implies that $i \in S\cap Q_{\mathsf{H}}$, in which case, by the definition of Q_{H}, we have $v_i\,r_i \leq v_j\,r_j$ for all $j \in M_{\mathsf{H}}$. Furthermore, noting that $M_{\mathsf{H}} \subseteq S$ and $M_{\mathsf{H}} \subseteq N_{\mathsf{H}}$, we get $M_{\mathsf{H}} \subseteq S\cap N_{\mathsf{H}}$, but because $i \in S\cap N_{\mathsf{H}}$ and $i \notin M_{\mathsf{H}}$, we obtain $M_{\mathsf{H}}\cup\{i\} \subseteq S\cap N_{\mathsf{H}}$. In this case, letting $\kappa = \lceil 1/\epsilon\rceil$ for notational brevity, noting that $|M_{\mathsf{H}}| = \kappa$, we obtain $(\kappa+1)\,v_i\,r_i \leq \sum_{j\in M_{\mathsf{H}}} v_j\,r_j + v_i\,r_i \leq \sum_{j\in S\cap N_{\mathsf{H}}} v_j\,r_j$. Focusing on the first and last terms in the last chain of inequalities and arranging the terms, we get $\sum_{j\in S\cap N_{\mathsf{H}}} v_j\,r_j - v_i\,r_i \geq \frac{\kappa}{1+\kappa}\sum_{j\in S\cap N_{\mathsf{H}}} v_j\,r_j$. Therefore, we obtain the chain of inequalities

$$\begin{aligned} \overline{\mathrm{F}}(S\setminus\{i\}) \;&=\max_{Q\subseteq S\setminus\{i\}} \overline{\mathrm{R}}(Q) \;\geq \max_{Q\subseteq (S\cap N_{\mathsf{H}})\setminus\{i\}} \overline{\mathrm{R}}(Q) \\ &\geq\; \overline{\mathrm{R}}((S\cap N_{\mathsf{H}})\setminus\{i\}) \;=\; \frac{\sum_{j\in S\cap N_{\mathsf{H}}} v_j\,r_j - r_i\,v_i}{1+V(S\cap N_{\mathsf{H}}) - v_i} \;\geq\; \frac{\frac{\kappa}{1+\kappa}\sum_{j\in S\cap N_{\mathsf{H}}} v_j\,r_j}{1+V(S\cap N_{\mathsf{H}})} \\ &=\; \frac{\kappa}{1+\kappa}\overline{\mathrm{R}}(S\cap N_{\mathsf{H}}) \;\geq\; \frac{1/\epsilon}{1+1/\epsilon}\overline{\mathrm{R}}(S\cap N_{\mathsf{H}}) \;=\; \frac{1}{1+\epsilon}\overline{\mathrm{R}}(S\cap N_{\mathsf{H}}), \end{aligned}$$

where the last inequality holds because we have $\kappa = \lceil 1/\epsilon\rceil$ and $x/(1+x)$ is increasing in x. Thus, the claim holds in the first case.

Second, consider the case $i \in (S\cap N_{\mathsf{L}})\setminus M_{\mathsf{H}}$. We have $i \in N_{\mathsf{L}}$, in which case, because N_{H} and N_{L} are disjoint, we get $i \notin N_{\mathsf{H}}$. Therefore, we get $S\cap N_{\mathsf{H}} \subseteq S\setminus\{i\}$. In this case, we obtain the chain of inequalities $\overline{\mathrm{F}}(S\setminus\{i\}) = \max_{Q\subseteq S\setminus\{i\}} \overline{\mathrm{R}}(Q) \geq \overline{\mathrm{R}}(S\cap N_{\mathsf{H}})$, where the inequality holds because we have $S\cap N_{\mathsf{H}} \subseteq S\setminus\{i\}$, so the solution $S\cap N_{\mathsf{H}}$ is feasible to the last maximization problem. Therefore, the claim holds in the second case as well. By the claim, we have $\overline{\mathrm{F}}(S\setminus\{i\}) \geq \frac{1}{1+\epsilon}\,\overline{\mathrm{R}}(S\cap N_{\mathsf{H}})$ for each $i\in S\setminus M_{\mathsf{H}}$. In this case, considering the objective function of problem (7), if we replace $\overline{\mathrm{F}}(S\setminus\{i\})$ with $\frac{1}{1+\epsilon}\,\overline{\mathrm{F}}(S\setminus\{i\})$ for each $i\in M_{\mathsf{H}}$ and replace $\overline{\mathrm{F}}(S\setminus\{i\})$ with $\frac{1}{1+\epsilon}\,\overline{\mathrm{R}}(S\cap N_{\mathsf{H}})$ for each $i\in S\setminus M_{\mathsf{H}}$, then the objective function of problem (7) gets smaller, but the smaller objective function corresponds to the $\frac{1}{1+\epsilon}$ fraction of the objective function of problem (8). ■

The two lemmas allow us to approximate the adaptive variant in (7). For $\alpha\in(0,1]$, let S_{APP} be an α-approximate solution to the surrogate problem in (8). Recalling that Z^*_{AD} and Z^*_{SR} are the optimal objective values, whereas $\mathcal{R}_{\mathsf{AD}}(S)$ and $\mathcal{R}_{\mathsf{SR}}(S)$ are the objective functions, of problems (7) and (8), we get $\mathcal{R}_{\mathsf{AD}}(S_{\mathsf{APP}}) \geq \frac{1}{1+\epsilon}\mathcal{R}_{\mathsf{SR}}(S_{\mathsf{APP}}) \geq \frac{\alpha}{1+\epsilon}\,Z^*_{\mathsf{SR}} \geq \frac{\alpha}{1+\epsilon}\,Z^*_{\mathsf{AD}}$, where the first inequality is by Lemma 4.3, the second inequality holds because S_{APP} is an α-approximate solution to the surrogate problem and the third inequality is by Lemma 4.2. Thus, the expected revenue provided by the solution S_{APP} for the adaptive variant is at least $\frac{\alpha}{1+\epsilon}$ fraction of the optimal expected revenue.

Thus, to obtain an approximate solution to the adaptive variant, it is enough to obtain an approximate solution to the surrogate problem. We focus on the latter task.

5. Approximating the Surrogate Problem

By the discussion at the end of the previous section, we focus on finding an approximate solution to the surrogate problem in (8). Recalling that we proceed with the understanding that we know the sets M_{H}, N_{H}, N_{L}, and the decision variable in this problem is the assortment S. For a given subset $S\subseteq N$, the objective function in (8) depends on S only through the quantities $\overline{\mathrm{F}}(S\setminus\{i\})$ for $i\in M_{\mathsf{H}}$, $V(S)$ and $\overline{\mathrm{R}}(S\cap N_{\mathsf{H}})$. Thus, noting that $|M_{\mathsf{H}}| = \lceil 1/\epsilon\rceil$, to evaluate the expected revenue from the assortment S, we need to know $\lceil 1/\epsilon\rceil + 2$ quantities. Building on this observation, to find an approximate solution to the surrogate problem, we guess the values of these $\lceil 1/\epsilon\rceil + 2$ quantities in an optimal solution to the surrogate problem. We solve an optimization problem that finds an assortment that maximizes the expected revenue while making sure that the assortment that we find is consistent with the guesses of the $\lceil 1/\epsilon\rceil + 2$ quantities. This optimization problem takes the form of a multi-dimensional knapsack problem with $\lceil 1/\epsilon\rceil + 3$ constraints, in which case, we can solve a dynamic program to obtain an approximate solution to the multi-dimensional knapsack problem while allowing violations of the constraints by a factor of $1+\epsilon$. It turns out that this approximate solution yields an approximate solution to the surrogate problem.

We turn to the question of finding an assortment S that is consistent with the guesses of the quantities $\overline{\mathrm{F}}(S\setminus\{i\})$ for $i\in M_{\mathsf{H}}$, $V(S)$ and $\overline{\mathrm{R}}(S\cap N_{\mathsf{H}})$. Because the objective function in (8) is increasing in $\overline{\mathrm{F}}(S\setminus\{i\})$ and $\overline{\mathrm{R}}(S\cap N_{\mathsf{H}})$, but decreasing in $V(S)$, it will be enough to ensure consistency with lower bound guesses on $\overline{\mathrm{F}}(S\setminus\{i\})$ and $\overline{\mathrm{R}}(S\cap N_{\mathsf{H}})$, but an upper bound guess on $V(S)$. For fixed $f_i\in\mathbb{R}_+$, noting that $\overline{\mathrm{F}}(S\setminus\{i\})=\max_{Q\subseteq S\setminus\{i\}}\overline{\mathrm{R}}(Q)$, we have $\overline{\mathrm{F}}(S\setminus\{i\})\geq f_i$ as long as there exists a subset $Q\subseteq S\setminus\{i\}$ such that $\overline{\mathrm{R}}(Q)\geq f_i$. Using the fact that $\overline{\mathrm{R}}(Q)=\frac{\sum_{j\in Q}v_j\,r_j}{1+\sum_{j\in Q}v_j}$, the inequality $\overline{\mathrm{R}}(Q)\geq f_i$ is equivalent to $\sum_{j\in Q}v_j\,(r_j-f_i)\geq f_i$. Thus, if there exists a subset $Q\subseteq S\setminus\{i\}$ such that $\sum_{j\in Q}v_j\,(r_j-f_i)\geq f_i$, then we have $\overline{\mathrm{F}}(S\setminus\{i\})\geq f_i$. Having a subset $Q\subseteq S\setminus\{i\}$ such that $\sum_{j\in Q}v_j\,(r_j-f_i)\geq f_i$ is equivalent to having $\max_{Q\subseteq S\setminus\{i\}}\sum_{j\in Q}v_j\,(r_j-f_i)\geq f_i$, but the optimal objective value of the last maximization problem is simply $\sum_{j\in S\setminus\{i\}}v_j\,[r_j-f_i]^+$. Therefore, by the discussion so far, if $\sum_{j\in S\setminus\{i\}}v_j\,[r_j-f_i]^+\geq f_i$, then we have $\overline{\mathrm{F}}(S\setminus\{i\})\geq f_i$. Interpreting f_i as the lower bound guess on $\overline{\mathrm{F}}(S\setminus\{i\})$, imposing the constraint $\sum_{j\in S\setminus\{i\}}v_j\,[r_j-f_i]^+\geq f_i$ is enough to ensure that $\overline{\mathrm{F}}(S\setminus\{i\})\geq f_i$. Moving on to the upper bound guess on $V(S)$, for fixed $g,h\in\mathbb{R}_+$, if we have $\sum_{j\in S\cap N_{\mathsf{H}}}v_j\leq g$ and $\sum_{j\in S\cap N_{\mathsf{L}}}v_j\leq h$, then we have $V(S)=\sum_{j\in S\cap N_{\mathsf{H}}}v_j+\sum_{j\in S\cap N_{\mathsf{L}}}v_j\leq g+h$. Considering the lower bound guess for $\overline{\mathrm{R}}(S\cap N_{\mathsf{H}})$, for fixed $g,w\in\mathbb{R}_+$, if we have $\sum_{j\in S\cap N_{\mathsf{H}}}v_j\leq g$ and $\sum_{j\in S\cap N_{\mathsf{H}}}v_j\,r_j\geq w$, then also we have $\overline{\mathrm{R}}(S\cap N_{\mathsf{H}})=\frac{\sum_{j\in S\cap N_{\mathsf{H}}}v_j\,r_j}{1+\sum_{j\in S\cap N_{\mathsf{H}}}v_j}\geq\frac{w}{1+g}$. In this case, using f_i and w to denote lower bound guesses on $\overline{\mathrm{F}}(S\setminus\{i\})$ and $\sum_{j\in S\cap N_{\mathsf{H}}}v_j\,r_j$, whereas g and h to denote upper bound guesses on $\sum_{j\in S\cap N_{\mathsf{H}}}v_j$ and $\sum_{j\in S\cap N_{\mathsf{L}}}v_j$, using the vector $\boldsymbol{f}=(f_i:i\in M_{\mathsf{H}})\in\mathbb{R}_+^{\lceil 1/\epsilon\rceil}$, to find an assortment S that maximizes the expected revenue in (8), while remaining consistent with the upper and lower bound guesses, we solve the problem

$$
\begin{aligned}
Z^*_{\mathsf{KS}}(\boldsymbol{f},g,h,w) \;=\; \max_{\substack{S\subseteq N:\\ M_{\mathsf{H}}\subseteq S\subseteq N_{\mathsf{H}}\cup N_{\mathsf{L}}}} \Bigg\{ & \sum_{i\in M_{\mathsf{H}}}\frac{v_i}{1+g+h}\,(r_i+p_i\,f_i)+\sum_{i\in S\setminus M_{\mathsf{H}}}\frac{v_i}{1+g+h}\Big(r_i+p_i\,\frac{w}{1+g}\Big)\;:\\
& \sum_{j\in S\setminus\{i\}}v_j\,[r_j-f_i]^+\geq f_i\quad\forall\, i\in M_{\mathsf{H}},\\
& \sum_{j\in S\cap N_{\mathsf{H}}}v_j\leq g,\quad \sum_{j\in S\cap N_{\mathsf{L}}}v_j\leq h,\quad \sum_{j\in S\cap N_{\mathsf{H}}}v_j\,r_j\geq w\Bigg\},
\end{aligned}\tag{9}
$$

where $(\boldsymbol{f},g,h,w)\in\mathbb{R}_+^{\lceil 1/\epsilon\rceil+3}$ is a vector of fixed constants, so the only decision variable in the problem above is the subset $S\subseteq N$.

Given fixed $(\boldsymbol{f},g,h,w)$, the four constraints in (9) ensure that any feasible solution $S\subseteq N$ to (9) satisfies $\overline{\mathrm{F}}(S\setminus\{i\})\geq f_i$ for all $i\in M_{\mathsf{H}}$, $V(S\cap N_{\mathsf{H}})\leq g$, $V(S\cap N_{\mathsf{L}})\leq h$ and $\sum_{j\in S\cap N_{\mathsf{H}}}v_j\,r_j\geq w$, in which case, we have $V(S)\leq g+h$ and $\overline{\mathrm{R}}(S\cap N_{\mathsf{H}})\geq\frac{w}{1+g}$ as well. Therefore, if we compute the objective function of problem (9) at any feasible solution $S\subseteq N$ to this problem, then we obtain a lower bound on the objective function of problem (8) evaluated at the same solution. We view problem (9) as a multi-dimensional knapsack problem with $\lceil 1/\epsilon\rceil+3$ constraints. Some of the constraints in problem (9) are covering constraints, rather than packing constraints, but we continue referring to this problem as a knapsack problem for convenience of terminology. Considering, for example, the first $\lceil 1/\epsilon \rceil$ constraints, which are indexed by the elements of M_{H}, if $j \neq i$, then the capacity consumption of product j in constraint $i \in M_{\mathsf{H}}$ is $v_j\,[r_j - f_i]^+$. If $j = i$, then the capacity consumption of product j in constraint $i \in M_{\mathsf{H}}$ is zero. Considering the last constraint, on the other hand, if $j \in N_{\mathsf{H}}$, then the capacity consumption of product j in this constraint is $v_j\, r_j$, otherwise the capacity consumption of product j in this constraint is zero. We consider the question of how we can use the knapsack problem in (9) to obtain an approximate solution to the surrogate problem in (8). We refer to the assortment $S \subseteq N$ as a <u>*super-optimal solution*</u> to problem (9) if this solution violates the constraints by at most a factor of $1+\epsilon$ and provides at least the optimal objective value. In other words, $S \subseteq N$ is a super-optimal solution to (9) if it satisfies $M_{\mathsf{H}} \subseteq S \subseteq N_{\mathsf{H}} \cup N_{\mathsf{L}}$,

$$\sum_{j \in S \setminus \{i\}} v_j\,[r_j - f_i]^+ \;\geq\; \frac{1}{1+\epsilon}\, f_i \quad \forall\, i \in M_{\mathsf{H}}, \tag{10}$$

$$\sum_{j \in S \cap N_{\mathsf{H}}} v_j \;\leq\; (1+\epsilon)\, g, \qquad \sum_{j \in S \cap N_{\mathsf{L}}} v_j \;\leq\; (1+\epsilon)\, h, \qquad \sum_{j \in S \cap N_{\mathsf{H}}} v_j\, r_j \;\geq\; \frac{1}{1+\epsilon}\, w,$$

$$\sum_{i \in M_{\mathsf{H}}} \frac{v_i}{1+g+h}\, \big(r_i + p_i\, f_i\big) + \sum_{i \in S \setminus M_{\mathsf{H}}} \frac{v_i}{1+g+h} \Big(r_i + p_i\, \frac{w}{1+g}\Big) \;\geq\; Z^*_{\mathsf{KS}}(\boldsymbol{f}, g, h, w).$$

In the next lemma, we show that we can use a super-optimal solution at particular guesses to get an approximate solution to the surrogate problem. In this lemma, we set $W(S) = \sum_{i \in S} v_i\, r_i$.

Lemma 5.1 (Knapsack to Surrogate) *Letting S^* be an optimal solution to the surrogate problem in* (8)*, consider the guesses $(\widehat{\boldsymbol{f}}, \widehat{g}, \widehat{h}, \widehat{w})$ that satisfy*

$$\widehat{f}_i \;\leq\; \overline{\mathrm{F}}(S^* \setminus \{i\}) \;\leq\; (1+\epsilon)\, \widehat{f}_i \quad \forall\, i \in M_{\mathsf{H}},$$

$$\frac{1}{1+\epsilon}\, \widehat{g} \;\leq\; V(S^* \cap N_{\mathsf{H}}) \;\leq\; \widehat{g}, \qquad \frac{1}{1+\epsilon}\, \widehat{h} \;\leq\; V(S^* \cap N_{\mathsf{L}}) \;\leq\; \widehat{h},$$

$$\widehat{w} \;\leq\; W(S^* \cap N_{\mathsf{H}}) \;\leq\; (1+\epsilon)\, \widehat{w}.$$

If the solution $\widehat{S}$ is super-optimal to the knapsack problem in (9) *with $(\boldsymbol{f}, g, h, w) = (\widehat{\boldsymbol{f}}, \widehat{g}, \widehat{h}, \widehat{w})$, then $\widehat{S}$ is a $(1-6\epsilon)$-approximate solution to the surrogate problem in* (8)*.*

Proof: We make two observations. First, because the solution $\widehat{S}$ is super-optimal to problem (9) with $(\boldsymbol{f}, g, h, w) = (\widehat{\boldsymbol{f}}, \widehat{g}, \widehat{h}, \widehat{w})$, by the first condition in (10) for a super-optimal solution, we obtain $\frac{1}{1+\epsilon}\, \widehat{f}_i \leq \sum_{j \in \widehat{S} \setminus \{i\}} v_j\, [r_j - \widehat{f}_i]^+ \leq \sum_{j \in \widehat{S} \setminus \{i\}} v_j\, [r_j - \frac{1}{1+\epsilon} \widehat{f}_i]^+$, where the last inequality holds because $[a-x]^+$ is decreasing in x. In this case, following the same sequence of steps just before our construction of problem (9), we can show that having $\frac{1}{1+\epsilon}\, \widehat{f}_i \leq \sum_{j \in \widehat{S} \setminus \{i\}} v_j\, [r_j - \frac{1}{1+\epsilon} \widehat{f}_i]^+$ is equivalent to having $\overline{\mathrm{F}}(\widehat{S} \setminus \{i\}) \geq \frac{1}{1+\epsilon}\, \widehat{f}_i$. Furthermore, by the second, third and fourth conditions in (10) for a super-optimal solution, we obtain $V(\widehat{S} \cap N_{\mathsf{H}}) \leq (1+\epsilon)\, \widehat{g}$, $V(\widehat{S} \cap N_{\mathsf{L}}) \leq (1+\epsilon)\, \widehat{h}$ and $\overline{\mathrm{R}}(\widehat{S} \cap N_{\mathsf{H}}) = \frac{W(\widehat{S} \cap N_{\mathsf{H}})}{1 + V(\widehat{S} \cap N_{\mathsf{H}})} \geq \frac{\widehat{w}/(1+\epsilon)}{1 + (1+\epsilon)\, \widehat{g}} \geq \frac{1}{(1+\epsilon)^2}\, \frac{\widehat{w}}{1+\widehat{g}}$. Second, by the first chain of inequalities in the statement of the lemma, we have $\overline{\mathrm{F}}(S^* \setminus \{i\}) \geq \widehat{f}_i$, in which case, following the same sequence of steps just before our construction of problem (9) once more, we can show that having $\overline{\mathrm{F}}(S^* \setminus \{i\}) \geq \widehat{f}_i$ is equivalent to having $\sum_{j \in S^* \setminus \{i\}} v_j \, [r_j - \widehat{f}_i]^+ \geq \widehat{f}_i$. Furthermore, by the second, third, and fourth chains of inequalities in the statement of the lemma, we have $V(S^* \cap N_{\mathsf{H}}) \leq \widehat{g}$, $V(S^* \cap N_{\mathsf{L}}) \leq \widehat{h}$ and $W(S^* \cap N_{\mathsf{H}}) \geq \widehat{w}$. Because the solution S^* is optimal to problem (8), we have $M_{\mathsf{H}} \subseteq S^* \subseteq N_{\mathsf{H}} \cup N_{\mathsf{L}}$ as well. Thus, the solution S^* is feasible to problem (9) when we solve this problem with the guesses $(\boldsymbol{f}, g, h, w) = (\widehat{\boldsymbol{f}}, \widehat{g}, \widehat{h}, \widehat{w})$, whereas the solution $\widehat{S}$ is super-optimal. In this case, we get

$$
\begin{aligned}
&\sum_{i \in M_{\mathsf{H}}} \frac{v_i}{1 + V(\widehat{S})} \Big(r_i + p_i \, \overline{\mathrm{F}}(\widehat{S} \setminus \{i\}) \Big) + \sum_{i \in \widehat{S} \setminus M_{\mathsf{H}}} \frac{v_i}{1 + V(\widehat{S})} \Big(r_i + p_i \, \overline{\mathrm{R}}(\widehat{S} \cap N_{\mathsf{H}}) \Big) \\
&\quad \overset{(a)}{\geq} \sum_{i \in M_{\mathsf{H}}} \frac{v_i}{1 + (1+\epsilon)\,(\widehat{g} + \widehat{h})} \left(r_i + p_i \, \frac{1}{1+\epsilon} \, \widehat{f}_i \right) + \sum_{i \in \widehat{S} \setminus M_{\mathsf{H}}} \frac{v_i}{1 + (1+\epsilon)(\widehat{g} + \widehat{h})} \left(r_i + p_i \, \frac{1}{(1+\epsilon)^2} \, \frac{\widehat{w}}{1 + \widehat{g}} \right) \\
&\quad \geq \frac{1}{(1+\epsilon)^3} \left\{ \sum_{i \in M_{\mathsf{H}}} \frac{v_i}{1 + \widehat{g} + \widehat{h}} \Big(r_i + p_i \, \widehat{f}_i \Big) + \sum_{i \in \widehat{S} \setminus M_{\mathsf{H}}} \frac{v_i}{1 + \widehat{g} + \widehat{h}} \left(r_i + p_i \, \frac{\widehat{w}}{1 + \widehat{g}} \right) \right\} \\
&\quad \overset{(b)}{\geq} \frac{1}{(1+\epsilon)^3} \left\{ \sum_{i \in M_{\mathsf{H}}} \frac{v_i}{1 + \widehat{g} + \widehat{h}} \Big(r_i + p_i \, \widehat{f}_i \Big) + \sum_{i \in S^* \setminus M_{\mathsf{H}}} \frac{v_i}{1 + \widehat{g} + \widehat{h}} \left(r_i + p_i \, \frac{\widehat{w}}{1 + \widehat{g}} \right) \right\} \\
&\quad \overset{(c)}{\geq} \frac{1}{(1+\epsilon)^3} \left\{ \sum_{i \in M_{\mathsf{H}}} \frac{v_i}{1 + (1+\epsilon)\, V(S^*)} \left(r_i + p_i \, \frac{1}{1+\epsilon} \, \overline{\mathrm{F}}(S^* \setminus \{i\}) \right) \right. \\
&\qquad\qquad \left. + \sum_{i \in \widehat{S} \setminus M_{\mathsf{H}}} \frac{v_i}{1 + (1+\epsilon)\, V(S^*)} \left(r_i + p_i \, \frac{W(S^* \cap N_{\mathsf{H}})/(1+\epsilon)}{1 + (1+\epsilon)\, V(S^* \cap N_{\mathsf{H}})} \right) \right\} \\
&\quad \geq \frac{1}{(1+\epsilon)^6} \left\{ \sum_{i \in M_{\mathsf{H}}} \frac{v_i}{1 + V(S^*)} \Big(r_i + p_i \, \overline{\mathrm{F}}(S^* \setminus \{i\}) \Big) + \sum_{i \in \widehat{S} \setminus M_{\mathsf{H}}} \frac{v_i}{1 + V(S^*)} \Big(r_i + p_i \, \overline{\mathrm{R}}(S^* \cap N_{\mathsf{H}}) \Big) \right\} \\
&\quad \overset{(d)}{=} \frac{1}{(1+\epsilon)^6} Z^*_{\mathsf{SR}} \; \geq \; (1 - 6\epsilon)\, Z^*_{\mathsf{SR}},
\end{aligned}
$$

where (a) uses the first observation, (b) uses the second observation, (c) follows from the four chains of inequalities in the statement of the lemma and (d) holds because S^* is optimal to (8).

Thus, $\widehat{S}$ provides $(1 - 6\epsilon)$ fraction of the optimal objective value of problem (8). Also, because $\widehat{S}$ is super-optimal to (9), we have $M_{\mathsf{H}} \subseteq \widehat{S} \subseteq N_{\mathsf{H}} \cup N_{\mathsf{L}}$, so the solution $\widehat{S}$ is feasible to (8). ■

By Lemma 5.1, if we have the guesses $(\widehat{\boldsymbol{f}}, \widehat{g}, \widehat{h}, \widehat{w})$ that satisfy the inequalities in Lemma 5.1, then a super-optimal solution to problem (9) at these guesses is a $(1-6\epsilon)$-approximate solution to the surrogate problem. To use Lemma 5.1, we need to answer two questions. First, how do we come up with the guesses $(\widehat{\boldsymbol{f}}, \widehat{g}, \widehat{h}, \widehat{w})$ that satisfy the inequalities in Lemma 5.1, given that we do not know the optimal solution S^* to the surrogate problem? Second, how do we compute a super-optimal solution to (9), even if we have the guesses $(\widehat{\boldsymbol{f}}, \widehat{g}, \widehat{h}, \widehat{w})$ that satisfy the inequalities in Lemma 5.1? For the first question, we come up with $O((\frac{n}{\epsilon})^{O(1/\epsilon)})$ possibilities for the guesses $(\widehat{\boldsymbol{f}}, \widehat{g}, \widehat{h}, \widehat{w})$ such that one of these possibilities satisfy all of the inequalities in Lemma 5.1. For the second question, we use a dynamic program. We take on each of these tasks one by one.

Constructing the Possibilities for Guesses:

We construct $O((\frac{n}{\epsilon})^{O(1/\epsilon)})$ possibilities for the guesses $(\widehat{\boldsymbol{f}}, \widehat{g}, \widehat{h}, \widehat{w})$ such that one of these possibilities satisfies all inequalities in Lemma 5.1. Letting S^* be an optimal solution to problem (8), we begin by coming up with possibilities for the guess $\widehat{g}$ such that one of these possibilities satisfies $\frac{1}{1+\epsilon}\,\widehat{g} \leq V(S^* \cap N_{\mathsf{H}}) \leq \widehat{g}$ without knowing the solution S^*. Set $\overline{V} = \max_{i \in S^* \cap N_{\mathsf{H}}} v_i$. Because we do not know S^*, we do not know $\overline{V}$ either, but there are $O(n)$ possibilities for $\overline{V}$ given by $\{v_i : i \in N\}$, so we can try each possibility. By the definition of $\overline{V}$, we have $V(S^* \cap N_{\mathsf{H}}) \geq \overline{V}$ and $V(S^* \cap N_{\mathsf{H}}) \leq n\overline{V}$, so $V(S^* \cap N_{\mathsf{H}})$ lies in the interval $[\overline{V}, n\overline{V}]$. We construct a geometric grid to cover the interval $[\overline{V}, n\overline{V}]$. So, considering the grid points $\{(1+\epsilon)^k\, \overline{V} : k = 0, 1, \ldots, \lceil \log n / \log(1+\epsilon) \rceil\}$, we have $(1+\epsilon)^k\, \overline{V} \leq V(S^* \cap N_{\mathsf{H}}) \leq (1+\epsilon)^{k+1}\, \overline{V}$ for some $k = 0, 1, \ldots, \lceil \log n / \log(1+\epsilon) \rceil$. We have $O(\frac{\log n}{\log(1+\epsilon)}) = O(\frac{\log n}{\epsilon})$ grid points. Thus, if we knew the value of $\overline{V}$, then we could construct a collection of $O(\frac{\log n}{\epsilon})$ grid points such that successive grid points differ by a factor of $1+\epsilon$ and $V(S^* \cap N_{\mathsf{H}})$ lies between a pair of successive grid points. Because we do not know the value of $\overline{V}$, we can construct a collection of grid points for each of the n possibilities for $\overline{V}$, in which case, we end up with a collection of $O(\frac{n \log n}{\epsilon})$ grid points such that successive grid points differ by at most a factor of $1+\epsilon$ and $V(S^* \cap N_{\mathsf{H}})$ lies between a pair of successive grid points.

We can use the same approach above to come up with $O(\frac{n \log n}{\epsilon})$ possibilities for the guess $\widehat{h}$ such that one of these possibilities satisfies $\frac{1}{1+\epsilon}\,\widehat{h} \leq V(S^* \cap N_{\mathsf{L}}) \leq \widehat{h}$. Similarly, we can construct $O(\frac{n \log n}{\epsilon})$ possibilities for the guess $\widehat{w}$ such that one of these possibilities satisfies $\widehat{w} \leq W(S^* \cap N_{\mathsf{H}}) \leq (1+\epsilon)\,\widehat{w}$. Lastly, we focus on coming up with possibilities for the guess $\widehat{f}_i$ such that one of these possibilities satisfies $\widehat{f}_i \leq \overline{\mathrm{F}}(S^* \setminus \{i\}) \leq (1+\epsilon)\,\widehat{f}_i$. We follow the approach in the previous paragraph, but finding an interval that includes $\overline{\mathrm{F}}(S^* \setminus \{i\})$ requires a bit more care. Set $\overline{z}_i = \max_{j \in S^* \setminus \{i\}} \overline{\mathrm{R}}(\{j\})$. Because we do not know S^*, we do not know $\overline{z}_i$ either, but there are only $O(n)$ possibilities for $\overline{z}_i$ given by $\{\overline{\mathrm{R}}(\{j\}) : j \in N \setminus \{i\}\}$, so we can try each possibility. In this case, we have the chain of inequalities $\overline{\mathrm{F}}(S^* \setminus \{i\}) = \max_{Q \subseteq S^* \setminus \{i\}} \overline{\mathrm{R}}(Q) \geq \max_{j \in S^* \setminus \{i\}} \overline{\mathrm{R}}(\{j\}) = \overline{z}_i$, which lower bounds $\overline{\mathrm{F}}(S^* \setminus \{i\})$. To upper bound $\overline{\mathrm{F}}(S^* \setminus \{i\})$, letting $\widehat{S}_i = \arg\max_{Q \subseteq S^* \setminus \{i\}} \overline{\mathrm{R}}(Q)$, we have $\overline{\mathrm{F}}(S^* \setminus \{i\}) = \overline{\mathrm{R}}(\widehat{S}_i)$ and $\widehat{S}_i \subseteq S^* \setminus \{i\}$. Thus, we have $\overline{\mathrm{F}}(S^* \setminus \{i\}) = \overline{\mathrm{R}}(\widehat{S}_i) = \frac{\sum_{j \in \widehat{S}_i} v_j\, r_j}{1 + \sum_{j \in \widehat{S}_i} v_j} \leq \sum_{j \in \widehat{S}_i} \frac{v_j\, r_j}{1 + v_j} = \sum_{j \in \widehat{S}_i} \overline{\mathrm{R}}(\{j\}) \leq n\, \overline{z}_i$, where the last inequality holds because $\widehat{S}_i \subseteq S^* \setminus \{i\}$ and $\overline{z}_i = \max_{j \in S^* \setminus \{i\}} \overline{\mathrm{R}}(\{j\})$. By the discussion so far, $\overline{\mathrm{F}}(S^* \setminus \{i\})$ lies in the interval $[\overline{z}_i, n\overline{z}_i]$, in which case, we follow our earlier approach to construct a collection of $O(\frac{n \log n}{\epsilon})$ grid points such that successive grid points differ by at most a factor of $1+\epsilon$ and $\overline{\mathrm{F}}(S^* \setminus \{i\})$ lies between a pair of successive grid points.

There are $\lceil 1/\epsilon \rceil + 3$ components in the vector of guesses $(\widehat{\boldsymbol{f}}, \widehat{g}, \widehat{h}, \widehat{w})$. By the discussion in the previous two paragraphs, for each one of these $\lceil 1/\epsilon \rceil + 3$ components, we can come up with

$O(\frac{n\log n}{\epsilon})$ possibilities such that one of these possibilities satisfies the corresponding inequality in Lemma 5.1. Thus, considering all of the components simultaneously, we can come up with $O((\frac{n\log n}{\epsilon})^{\lceil 1/\epsilon\rceil+3}) = O((\frac{n}{\epsilon})^{O(1/\epsilon)})$ possibilities for the vector of guesses $(\widehat{\boldsymbol{f}}, \widehat{g}, \widehat{h}, \widehat{w})$ such that one of these possibilities satisfies all of the inequalities in Lemma 5.1, as desired. We proceed to discussing how we can obtain a super-optimal solution to the knapsack problem in (9).

Obtaining a Super-Optimal Solution:

We can solve the knapsack problem in (9) using a dynamic program. In the dynamic program, the epochs correspond to the products. The action in the epoch for product i corresponds to whether we include product i in the solution. Noting that there are $\lceil 1/\epsilon\rceil + 3$ constraints in problem (9), the state variable at a particular epoch keeps track of the value of the left side of each of these constraints as a function of the products that we include in the solution up to the current epoch. The state variable in such a dynamic program takes continuous values, so we cannot solve the dynamic program in polynomial time. To get around this difficulty, we use an approach pioneered by Desire et al. (2020). We scale each constraint in problem (9) by its right side so the right side of each constraint is normalized to one. After normalizing each constraint, we multiply each constraint by n/ϵ and round each constraint coefficient on the left side to an integer. In this case, the right side of each constraint becomes n/ϵ. In this way, we achieve three goals. First, the capacity consumption of each product on the left side of each constraint becomes an integer quantity. Second, if ϵ is small, then n/ϵ is large, so the constraint coefficients on the left side are large numbers, in which case, intuitively speaking, rounding them to an integer does not result in too much error. Third, because the right side of each constraint is n/ϵ, we do not need to consider the values of the state variable with a component exceeding n/ϵ. In this case, using the fact that the capacity consumption of each product on the left side of each constraint in an integer quantity and we only focus on the values of the state variable with components not exceeding n/ϵ, we can solve a dynamic program in $O(n(\frac{n}{\epsilon})^{\lceil 1/\epsilon\rceil+3}) = O((\frac{n}{\epsilon})^{O(1/\epsilon)})$ operations to obtain an optimal solution to the knapsack problem with scaled constraints. This optimal solution turns out to be a super-optimal solution to the knapsack problem in (9). The preceding discussion gives an overview of our approach for obtaining a super-optimal solution. In Appendix B, we give the full details of our approach.

Proof of Theorem 4.1:

We use our results to give a proof for Theorem 4.1. By the discussion in Section 4, there are $O(n^{O(1/\epsilon)})$ possibilities for the sets M_{H}, Q_{H} and N_{L}. We execute our PTAS for each possibility. By the discussion in this section, there are $O((\frac{n}{\epsilon})^{O(1/\epsilon)})$ possibilities for the vector of guesses $(\widehat{\boldsymbol{f}}, \widehat{g}, \widehat{h}, \widehat{w})$ such that one of these possibilities satisfies the inequalities in Lemma 5.1. For each possible vector of guesses, we solve a dynamic program in $O((\frac{n}{\epsilon})^{O(1/\epsilon)})$ operations to obtain a super-optimal solution to the knapsack problem in (9). By Lemma 5.1, one of these super-optimal solutions is a $(1-6\epsilon)$-approximate solution to the surrogate problem in (8), in which case, by the discussion that follows Lemma 4.3, this approximate solution to the surrogate problem is a $\frac{1-6\epsilon}{1+\epsilon}$-approximate solution to the adaptive variant. Noting that $\frac{1-6\epsilon}{1+\epsilon} \geq 1-7\epsilon$, we get a $(1-7\epsilon)$-approximate solution to the adaptive variant in a running time of $O(n^{O(1/\epsilon)}) \times O((\frac{n}{\epsilon})^{O(1/\epsilon)}) \times O((\frac{n}{\epsilon})^{O(1/\epsilon)}) = O((\frac{n}{\epsilon})^{O(1/\epsilon)})$. For given $\delta \in (0,1)$, to obtain a $(1-\delta)$-approximate solution to the adaptive variant, we can execute our approach with $\epsilon = \delta/7$, which yields a running time of $O((\frac{n}{\delta})^{O(1/\delta)})$. ■

6. Approximation Scheme for the Non-Adaptive Variant

We give an FPTAS for the non-adaptive variant. In particular, for any $\epsilon > 0$, our FPTAS yields a $(1-\epsilon)$-approximate solution to the non-adaptive variant in a running time of $O(\frac{n^7 (\log n)^2}{\epsilon^4})$. The FPTAS for the non-adaptive variant is a bit more direct than the PTAS for the adaptive variant. In the non-adaptive variant, the assortment decisions $(S_0, S_1, \ldots, S_n)$ satisfy $S_i = S_0 \setminus \{i\}$ for all $i \in N$. Therefore, we can represent our assortment decisions as $(S_0, S_0 \setminus \{1\}, \ldots, S_0 \setminus \{n\})$, so we only need to find the assortment S_0 offered in the first stage. Noting the expected revenue from the assortment decisions $(S_0, S_1, \ldots, S_n)$ given by (5), the non-adaptive variant is

$$Z^*_{\mathsf{NA}} \;=\; \max_{S \subseteq N} \left\{ \sum_{i \in S} \frac{v_i}{1+V(S)} \left(r_i + p_i \frac{\sum_{j \in S \setminus \{i\}} v_j \, r_j}{1+V(S \setminus \{i\})} \right) \right\}. \tag{11}$$

Throughout this section, we use S^* to denote an optimal solution to the non-adaptive variant in (11). Also, recall that $W(S) = \sum_{i \in S} v_i \, r_i$. We proceed to constructing our FPTAS.

We define two specific products. Let $i^* = \arg\max_{i \in S^*} v_i$ and $j^* = \arg\max_{i \in S^*} v_i \; r_i$. We do not know i^* or j^*, but there are $O(n)$ possibilities for each product, so we can try each possibility. Thus, we proceed with the understanding that we know i^* and j^*. To capture the other products that can possibly be in S^*, we define the set $N_{\mathsf{L}} = \{i \in N \setminus \{i^*, j^*\} : v_i \leq v_{i^*} \text{ and } v_i \, r_i \leq v_{j^*} \, r_{j^*}\}$. In this case, $S^* \subseteq \{i^*, j^*\} \cup N_{\mathsf{L}}$. Guessing the quantities $V(S^* \setminus \{i^*\})$ and $W(S^* \setminus \{j^*\})$, we find an assortment that is consistent with these guesses while maximizing the expected revenue. Using g and w to, respectively, denote the guesses of $V(S^* \setminus \{i^*\})$ and $W(S^* \setminus \{j^*\})$, we consider the problem

$$\begin{aligned} Z^*_{\mathsf{KS}}(g,w) \quad = \quad \max_{\substack{S \subseteq N: \\ \{i^*, j^*\} \subseteq S \subseteq \{i^*, j^*\} \cup N_{\mathsf{L}}}} & \left\{ \sum_{i \in S} \frac{v_i}{1+g+v_{i^*}} \left(r_i + p_i \frac{w + v_{j^*} \, r_{j^*} - v_i \, r_i}{1+g+v_{i^*} - v_i} \right) \; : \right. \\ & \left. \sum_{i \in S \setminus \{i^*\}} v_i \leq g, \qquad \sum_{i \in S \setminus \{j^*\}} v_i \, r_i \geq w \right\}. \end{aligned} \tag{12}$$

In problem (11), the objective function is decreasing in $V(S)$ and $V(S \setminus \{i\})$, but increasing in $W(S \setminus \{i\})$. Therefore, it will be enough to impose an upper bound on $V(S \setminus \{i^*\})$ and a lower bound on $W(S\setminus\{j^*\})$ in problem (12). Given fixed $(g,w)\in\mathbb{R}^2_+$, the two constraints in (12) ensure that any feasible solution $S\subseteq N$ to problem (12) satisfies $V(S\setminus\{i^*\})\le g$ and $W(S\setminus\{j^*\})\ge w$, in which case, we have $V(S)\le g+v_{i^*}$, $W(S\setminus\{i\})\ge w+v_{j^*}r_{j^*}-v_i\,r_i$ and $V(S\setminus\{i\})\le g+v_{i^*}-v_i$. Therefore, if we compute the objective function of problem (12) at any feasible solution $S\subseteq N$ to this problem, then we obtain a lower bound on the objective function of problem (11) evaluated at the same solution. For fixed (g,w), we view problem (12) as a two-dimensional knapsack problem, even though the first constraint is a packing, whereas the second constraint is a covering constraint. The outline of our FPTAS for the non-adaptive variable follows an outline similar to that of our PTAS for the adaptive variant, but the development of our FPTAS will be simpler. In our FPTAS for the non-adaptive variant, we will relate super-optimal solutions to the knapsack problem in (12) directly to the non-adaptive variant in (11). In our PTAS for the adaptive variant, we relate super-optimal solutions to the knapsack problem in (9) to the surrogate problem in (8) and approximate solutions to the surrogate problem to the adaptive variant in (7). We refer to solution $S\subseteq N$ as a super-optimal solution to problem (12) if it satisfies $\{i^*,j^*\}\subseteq S\subseteq\{i^*,j^*\}\cup N_{\mathsf{L}}$,

$$\sum_{i\in S\setminus\{i^*\}} v_i \le (1+\epsilon)\,g, \qquad \sum_{i\in S\setminus\{j^*\}} v_i\,r_i \ge \frac{1}{1+\epsilon}\,w,$$
$$\sum_{i\in S}\frac{v_i}{1+g+v_{i^*}}\left(r_i+p_i\frac{w+v_{j^*}\,r_{j^*}-v_i\,r_i}{1+g+v_{i^*}-v_i}\right) \ge Z^*_{\mathsf{KS}}(g,w).$$

In the next lemma, we relate a super-optimal solution to the knapsack problem in (12) at particular guesses to an approximate solution to the non-adaptive variant in (11).

Lemma 6.1 (Knapsack to Non-Adaptive) *Letting S^* be an optimal solution to the non-adaptive variant in* (11)*, consider the guesses $(\widehat{g},\widehat{w})$ that satisfy*

$$\frac{1}{1+\epsilon}\,\widehat{g} \le V(S^*\setminus\{i^*\}) \le \widehat{g}, \qquad \widehat{w} \le W(S^*\setminus\{j^*\}) \le (1+\epsilon)\,\widehat{w}.$$

If the solution $\widehat{S}$ is super-optimal to the knapsack problem in (12) *with $(g,w)=(\widehat{g},\widehat{w})$, then $\widehat{S}$ is a $(1-4\epsilon)$-approximate solution to the non-adaptive variant in* (11).

Lemma 6.1 is the analogue of Lemma 5.1 and its proof uses an argument that is similar to one in the proof of Lemma 5.1, so we skip the proof of Lemma 6.1. To use Lemma 6.1, we need to come up with guesses $(\widehat{g},\widehat{w})$ that satisfy the inequalities in Lemma 6.1 and compute a super-optimal solution to problem (12). We address both of these questions using an approach that closely follows the one that we used for the adaptive variant. First, to come up with guesses $(\widehat{g},\widehat{w})$ that satisfy the inequalities in Lemma 6.1, we construct $O((\frac{n\log n}{\epsilon})^2)$ possibilities for the guesses $(\widehat{g},\widehat{w})$ such that one of these possibilities satisfies all of the inequalities in Lemma 6.1. We construct these possibilities using precisely the same approach that we used for the adaptive variant. To obtain a super-optimal solution to the knapsack problem in (12), note that problems (9) and (12) are both multi-dimensional knapsack problems. Problem (9) has $\lceil 1/\epsilon \rceil + 3$ constraints. In Appendix B, we solve a dynamic program in $O(n(\frac{n}{\epsilon})^{\lceil 1/\epsilon \rceil + 3})$ operations to obtain a super-optimal solution to problem (9). On the other hand, problem (12) has two constraints. In this case, we can follow precisely the same approach in Appendix B to solve a dynamic program in $O(n(\frac{n}{\epsilon})^2)$ operations to obtain a super-optimal solution to problem (12). Thus, we can come up with guesses $(\widehat{g}, \widehat{w})$ that satisfy the inequalities in Lemma 6.1 and compute a super-optimal solution to problem (12). In the next theorem, we use these results to give our FPTAS for the non-adaptive variant.

Theorem 6.2 (FPTAS for Non-Adaptive Variant) *There exists an algorithm that provides a $(1-\epsilon)$-approximate solution to the non-adaptive variant in a running time of $O(\frac{n^7(\log n)^2}{\epsilon^4})$.*

Proof: There are $O(n^2)$ possibilities for the products i^* and j^*. We execute our FPTAS for each possibility. There are $O((\frac{n \log n}{\epsilon})^2)$ possibilities for the pair of guesses $(\widehat{g}, \widehat{w})$ such that that one of these possibilities satisfies the inequalities in Lemma 6.1. For each possible pair of guesses, we solve a dynamic program in $O(n(\frac{n}{\epsilon})^2)$ operations to obtain a super-optimal solution to the knapsack problem in (12). By Lemma 6.1, one of these super-optimal solutions is a $(1-4\epsilon)$-approximate solution to the non-adaptive variant in (11). Thus, we obtain a $(1-4\epsilon)$-approximate solution to the non-adaptive variant in a running time of $O(n^2) \times O((\frac{n \log n}{\epsilon})^2) \times O(n(\frac{n}{\epsilon})^2) = O(\frac{n^7(\log n)^2}{\epsilon^4})$. For given $\delta \in (0,1)$, to obtain a $(1-\delta)$-approximate solution to the non-adaptive variant, we can execute our approach with $\epsilon = \delta/4$, which yields a running time of $O(\frac{n^7(\log n)^2}{\delta^4})$. ■

We discuss the findings from our case study on a dataset from a grocery delivery platform and numerically test the performance of our PTAS for the adaptive variant.

7. Case Study

Using a dataset from Instacart, a large grocery delivery platform, we demonstrate the benefits from explicitly incorporating replacement possibilities and using adaptive solutions.

7.1 Sale Transaction Dataset and Model Estimation

We obtain our data from our industry partner Instacart. A sample of 11 million orders was taken over 21 weeks from select retailers. Customers arriving at the platform choose the retailer that they would like to receive delivery from, at which point, the platform offers assortments of products in different product categories from the retailer chosen by the customer. Each row of our sale transaction data provides information on the ID number, product category and price of the sold product, transaction date, ID number of the retailer making the sale, and availability index in the interval $[0, 1]$, representing the estimated probability that the sold product will be in stock. The platform offers assortments of products in different product categories from the retailer chosen by the customer, so we consider the sale transactions from each product category and retailer combination as a different dataset. We focus on 12 product categories with the largest number of sale transactions, dropping others from consideration. For each product category and retailer combination, we calculate the total number of sale transactions for each product. Sorting these numbers of sale transactions in decreasing order, we keep the top 90% of the sale transactions and label the remaining sale transactions as a no-purchase sale transaction. Thus, for each product category and retailer combination, we focus on the products with the largest sale volumes that provide 90% of the total sale volume in the product category. In this way, we fix the number of products in each product category offered by each retailer. Lastly, we focus on the product category and retailer combinations that include 10 to 60 products, so that we can solve the model estimation and assortment optimization problems relatively fast.

The dataset does not give the assortment of products offered to a customer during a sale transaction. We construct this assortment by assuming that all products for which we observed a sale during the week of the sale transaction are available in the offered assortment. This approach is commonly used to construct offered assortments; see Jagabathula and Rusmevichientong (2019). Using the preceding discussion, we associate an offered assortment with each transaction, in which case, for product category c and retailer r, our the dataset provides $\{(S_t, i_t, f_t, q_t) : t = 1, \ldots, T_{(c,r)}\}$, where $T_{(c,r)}$ is the number of sale transactions for the product category and retailer combination (c, r), S_t is the assortment offered, i_t is the product, if any, sold, f_t is the price of the sold product and q_t is the estimated availability of the product sold in transaction t. A sale transaction may correspond to a no-purchase, in which case, we set $i_t = \varnothing$. In Table 1, we give summary statistics for the dataset. The first column shows the product category. The second column shows the number of retailers making sales in the product category. The third column shows the average number of sale transactions in the product category, where the average is computed over the retailers. The fourth column shows the average number of products in the product category. The fifth column shows the average number of products in the assortments offered in the product category.

The dataset provides the price of the product involved in each sale transaction. For each product i, we set the revenue f_i as the average of the prices in the sale transactions in which the product is sold. Adopting a pessimistic viewpoint, for each product i, we set the availability probability q_i as the minimum of the availability probabilities in the sale transactions in which the

Category	Rets.	Trans.	Prods.	Offs.	Category	Rets.	Trans.	Prods.	Offs.
Apples	34	3455	17.44	15.57	Leafy Vegetables	39	5743	36.18	30.63
Berries	26	6709	16.00	14.17	Plain Milk	39	3735	24.62	22.74
Candy	11	1163	25.73	14.41	Seasonings	13	1143	32.69	19.91
Chicken	48	3238	35.98	28.50	Sparkling Water	12	2212	33.15	24.85
Chips	12	2796	28.67	23.27	Tomatoes	27	3765	14.48	13.77
Greek Yoghurt	8	1811	36.75	25.61	Whole Eggs	23	3292	13.48	12.95

Table 1 Summary statistics for the sale transaction data for different product categories.

product is sold. Using the dataset $\{(S_t, i_t) : t = 1, \ldots, T_{(c,r)}\}$, we fit a multinomial logit model to characterize the choice process among the products offered by retailer r in product category c. We use the sales transactions in the first 16 weeks as the training data to fit the model, but use the sales transactions in the last five weeks as the testing data to check the out-of-sample performance. We use standard maximum likelihood estimation to fit the multinomial logit model; see Vulcano et al. (2012). In this way, we obtain the preference weight v_i for each product i.

For the multinomial logit models that we fit for the different product category and retailer combinations, we provide the out-of-sample statistics in Table 2. The first column gives the product categories. The second column gives the average number of products in the offered assortments, where the average is computed over the retailers making sales in the product category. In the next three columns, we give the 1-hit, 2-hit, and 3-hit rates, each averaged over all retailers. We define the k-hit rate as the percentage of time that the sold product is one of the top k products with the largest choice probabilities within the offered assortment. In particular, using $\mathcal{T}_{(c,r)}$ to denote the set of sale transactions in the testing data for product category and retailer combination (c,r) and $S_t(k)$ to denote the set of k products with the largest choice probabilities within the assortment S_t under the fitted multinomial logit model, the k-hit rate for product category and retailer combination is given by $100 \times \frac{\sum_{t \in \mathcal{T}_{(c,r)}} \mathbf{1}(i_t \in S_t(k))}{\sum_{t \in \mathcal{T}_{(c,r)}} \mathbf{1}(i_t \neq \emptyset)}$. In the last column, we give the mean absolute error between the expected and observed fractions of purchases for each product. Using $N_{(c,r)}$ to denote the set of products in product category and retailer combination (c,r) and $\phi_i(S)$ to denote the choice probability of product i within assortment S under the fitted multinomial logit model, the mean absolute error for product category and retailer combination (c,r) is $100 \times \frac{1}{|N_{(c,r)}|} \sum_{i \in N_{(c,r)}} \left| \frac{\sum_{t \in \mathcal{T}_{(c,r)}} \mathbf{1}(i = i_t)}{|\mathcal{T}_{(c,r)}|} - \frac{\sum_{t \in \mathcal{T}_{(c,r)}} \phi_i(S_t)}{|\mathcal{T}_{(c,r)}|} \right|$, where the two expressions in the absolute value are the observed and expected fractions of purchases for product i.

The results in Table 2 indicate that the fitted multinomial logit model does a good job of predicting the purchases in the sale transactions in the training dataset. The average 3-hit rate is 27.53%, which implies that the purchased product in 27.53% of the transactions is one of the top three products with the largest choice probability under the fitted choice model. Considering the fact that the average number of products in the offered assortments is 20.59, such a 3-hit rate is quite satisfactory. The average mean absolute error in the fractions of purchases for the products

Category	Offs.	1-hit	2-hit	3-hit.	Mean abs. err.
Apples	15.57	21.14	34.14	42.91	1.33
Berries	14.17	18.30	32.90	43.91	1.32
Candy	14.41	20.55	32.03	39.10	1.33
Chicken	28.50	11.99	20.98	29.37	0.76
Chips	23.27	12.97	23.65	30.80	0.81
Greek Yoghurt	25.61	14.76	24.07	31.18	0.58
Avg.	20.26	16.62	27.96	36.21	1.02

Category	Offs.	1-hit	2-hit	3-hit.	Mean abs. err.
Leafy Vegetables	30.63	15.09	25.92	32.92	0.58
Plain Milk	22.74	14.71	25.23	33.73	0.67
Seasonings	19.91	9.88	18.75	25.88	0.81
Sparkling Water	24.85	10.95	19.89	28.02	0.65
Tomatoes	13.77	19.56	33.04	44.36	1.09
Whole Eggs	12.95	25.67	39.81	51.59	0.99
Avg.	20.91	15.98	27.11	36.08	0.80

Table 2 Out of sample performance measures for the fitted multinomial logit model.

is 0.91%. To put this mean absolute error in perspective, on average, there are 20.59 products in the offered assortments, so if we were to proceed under the, admittedly strong, assumption that the customers choose uniformly among the offered products, then each product would command 4.86% of the purchases. We estimate such magnitude of fractions with 0.91% error.

7.2 Performance of Naive, Non-Adaptive and Adaptive Assortments

We compare the performance of three solutions to our assortment optimization problem with increasing levels of sophistication. In the first solution, we compute the revenue-maximizing assortment without modeling the replacement options and using the standard multinomial logit model to capture the choice process. We refer to this solution as the naive solution. In particular, noting the definition of $\overline{\mathrm{R}}(S)$ in Section 3, we compute the naive solution as $\widehat{S}_{\mathsf{ML}} = \arg\max_{S\subseteq N} \overline{\mathrm{R}}(S)$, where the subscript emphasizes that we compute this solution under the standard multinomial logit model. In the second solution, we use the non-adaptive solution. Letting $\widehat{S}_{\mathsf{NA}}$ be an optimal solution to problem (11), the non-adaptive solution is $\widehat{S}_{\mathsf{NA}}$. In the third solution, we use the adaptive solution. Letting $\widehat{S}_{\mathsf{AD}}$ be an optimal solution to problem (7), the adaptive solution is $\widehat{S}_{\mathsf{AD}}$. In the adaptive solution, noting (7), given that a customer picked product i as her preferred choice, we compute her second stage assortment as $\widehat{S}_i = \arg\max_{S\subseteq \widehat{S}_{\mathsf{AD}}\setminus\{i\}} \overline{\mathrm{R}}(S)$. We compute the expected revenue provided by each of these three solutions by using the same objective function. Using the expected revenue function $\mathrm{R}(S_0, S_1, \ldots, S_n)$ in (3), we compute the expected revenue from the three solutions as $\mathsf{Rev}_{\mathsf{ML}} = \mathrm{R}(\widehat{S}_{\mathsf{ML}}, \widehat{S}_{\mathsf{ML}} \setminus \{1\}, \ldots, \widehat{S}_{\mathsf{ML}} \setminus \{n\})$, $\mathsf{Rev}_{\mathsf{NA}} = \mathrm{R}(\widehat{S}_{\mathsf{NA}}, \widehat{S}_{\mathsf{NA}} \setminus \{1\}, \ldots, \widehat{S}_{\mathsf{NA}} \setminus \{n\})$ and $\mathsf{Rev}_{\mathsf{AD}} = \mathrm{R}(\widehat{S}_{\mathsf{AD}}, \widehat{S}_1, \ldots, \widehat{S}_n)$. The solutions $\widehat{S}_{\mathsf{ML}}$, $\widehat{S}_{\mathsf{NA}}$, and $\widehat{S}_{\mathsf{AD}}$ are all computed under the assumption that the fitted multinomial logit model is the ground choice model. This assumption does not put any of the solutions at a disadvantage. We are simply interested in how the three solutions perform under problem parameters driven by a realistic dataset. Note that we compute the solutions $\widehat{S}_{\mathsf{ML}}$, $\widehat{S}_{\mathsf{NA}}$ and $\widehat{S}_{\mathsf{AD}}$ for each product category and retailer combination using the multinomial logit model fitted to the dataset for each product category and retailer combination.

Our goal is to compare the expected revenue performance of the three solutions, rather than testing the robustness of our approximation schemes. To shield our results from the approximation errors incurred by our approximation schemes, we compute the solutions $\widehat{S}_{\mathsf{NA}}$ and $\widehat{S}_{\mathsf{AD}}$ using mixed integer programs. We give mixed integer programs for the adaptive and non-adaptive variants in Appendix C. Mixed integer programs do not yield solutions in polynomial time, but they isolate potential errors from the approximation schemes. We explore the robustness of our approximate schemes in the next section. In addition to the expected revenues provided by the three solutions, we compute performance measures related to the market share. Full market share is the fraction of customers that pick some choice, instead of leaving without a purchase. Net market share is the fraction of customers that pick some preferred and replacement choices and receive one of the picked options. The refund rate is the fraction of customers that pick some choice but receive nothing due to unavailability. We can compute the net market share by using the expected revenue function in (3) with $f_i = 1$ for all $i \in N$, so that each sale brings unit revenue. In this case, corresponding to the solution $(S_0, S_1, \ldots, S_n)$, using $\mathsf{FMS}(S_0)$, $\mathsf{NMS}(S_0, S_1, \ldots, S_n)$ and $\mathsf{RFR}(S_0, S_1, \ldots, S_n)$ to, respectively, denote the full market share, net market share and refund rate, we have

$$\begin{aligned}\mathsf{FMS}(S_0) &= 100 \times \frac{\sum_{i \in S_0} v_i}{1 + V(S_0)}\\ \mathsf{NMS}(S_0, S_1, \ldots, S_n) &= 100 \times \sum_{i \in S_0} \frac{v_i}{1+V(S_0)} \left(q_i + (1 - q_i) \sum_{j \in S_i} \frac{v_j}{1+V(S_i)}\, q_j \right)\\ \mathsf{RFR}(S_0, S_1, \ldots, S_n) &= \mathsf{FMS}(S_0) - \mathsf{NMS}(S_0, S_1, \ldots, S_n).\end{aligned}$$

For the naive solution $\widehat{S}_{\mathsf{ML}}$, we set $\mathsf{Full}_{\mathsf{ML}} = \mathsf{FMS}(\widehat{S}_{\mathsf{ML}})$, $\mathsf{Net}_{\mathsf{ML}} = \mathsf{NMS}(\widehat{S}_{\mathsf{ML}}, \widehat{S}_{\mathsf{ML}} \setminus \{1\}, \ldots, \widehat{S}_{\mathsf{ML}} \setminus \{n\})$ and $\mathsf{Ref}_{\mathsf{ML}} = \mathsf{RFR}(\widehat{S}_{\mathsf{ML}}, \widehat{S}_{\mathsf{ML}} \setminus \{1\}, \ldots, \widehat{S}_{\mathsf{ML}} \setminus \{n\})$. We define similar quantities for $\widehat{S}_{\mathsf{NA}}$ and $\widehat{S}_{\mathsf{AD}}$.

Value of Incorporating Replacement Options: To understand the benefit from explicitly incorporating the replacement options, we compare the performance of the solutions $\widehat{S}_{\mathsf{ML}}$ and $\widehat{S}_{\mathsf{NA}}$. These solutions use a non-adaptive strategy in the second stage, but we do not explicitly incorporate the replacement options when computing the solution $\widehat{S}_{\mathsf{ML}}$. We give our results in Table 3. The first column gives the product category. The second and third columns give the number of products in the solutions $\widehat{S}_{\mathsf{ML}}$ and $\widehat{S}_{\mathsf{NA}}$, averaged over the retailers in the product category. For each product category and retailer combination, we compute the percent expected revenue gap as $100 \times \frac{\mathsf{Rev}_{\mathsf{NA}} - \mathsf{Rev}_{\mathsf{ML}}}{\mathsf{Rev}_{\mathsf{NA}}}$. The third, fourth and fifth columns give the average, 95th percentile, and maximum percent expected revenue gap, where these three statistics are computed over all retailers in a product category. For each product category and retailer combination, we compute the gap between the full market shares as $\mathsf{Full}_{\mathsf{NA}} - \mathsf{Full}_{\mathsf{ML}}$. In the sixth column, we give the average gap between the full market shares, where the average is computed over all retailers. In the seventh and eighth columns, we give the average gaps between the net market shares and refund rates.

The results in Table 3 indicate that there is a significant expected revenue benefit from incorporating the replacement options explicitly when we compute the assortments to offer to the

	Off. size		Diff. in Exp. Rev.			Diff. in Mkt. Share		
Category	$\widehat{S}_{\mathsf{ML}}$	$\widehat{S}_{\mathsf{NA}}$	Avg.	95-th	Max	Full	Net	Rfn.
Apples	6.06	7.88	4.02	10.36	13.79	6.29	0.75	5.54
Berries	5.35	7.58	3.32	8.08	17.21	6.84	1.68	5.17
Candy	6.82	7.64	3.87	7.19	8.23	3.51	-1.62	5.14
Chicken	10.38	13.90	3.15	7.70	9.70	6.77	2.50	4.27
Chips	7.42	8.92	3.90	9.98	10.81	4.88	-0.43	5.30
Greek Yoghurt	8.75	11.63	3.05	5.40	5.79	4.93	1.68	3.25
Leafy Vegetables	11.59	16.62	2.91	5.19	8.15	6.16	2.15	4.01
Plain Milk	7.95	10.87	3.58	7.41	10.32	4.95	-0.21	5.16
Seasonings	9.85	11.85	2.02	3.82	3.93	3.77	1.88	1.90
Sparkling Water	8.77	11.62	2.59	4.63	5.11	6.98	3.12	3.86
Tomatoes	6.30	8.52	3.65	8.96	15.76	5.86	0.84	5.02
Whole Eggs	5.55	7.27	3.72	11.80	13.18	8.44	3.69	4.75
Avg.	7.90	10.36	3.31	7.54	10.16	5.78	1.33	4.45

Table 3 Benefit from explicitly incorporating the replacement options.

customers. Over all product categories, the non-adaptive solution provides an average expected revenue improvement of 3.31% over the naive solution. There are product category and retailer combinations for which the expected revenue improvements exceed 17%. The assortments offered as options for the preferred choice under the non-adaptive solution are larger than those under the naive solution, although it is not possible to show theoretically that the former assortments include the latter. Because the non-adaptive solution tends to offer larger assortments, not so surprisingly, the market share under the non-adaptive solution is larger. More importantly, perhaps, the net market share, which captures the faction of customers with fulfilled orders through their preferred or replacement options, tends to be larger under the non-adaptive solution. Over all product categories, the non-adaptive solution provides an average improvement of 1.33% on the net market share of the naive solution, even though we do not observe a larger market share for the non-adaptive solution uniformly over all product categories.

Value of Using Adaptive Solutions: We compare the performance of the solutions $\widehat{S}_{\mathsf{NA}}$ and $\widehat{S}_{\mathsf{AD}}$ to understand the benefit from adapting the assortment in the second stage to the preferred choice of the customer. We give our results in Table 4. The format of this table is identical to that of Table 3. The difference is that we compare the solutions $\widehat{S}_{\mathsf{NA}}$ and $\widehat{S}_{\mathsf{AD}}$, as opposed to the solutions $\widehat{S}_{\mathsf{ML}}$ and $\widehat{S}_{\mathsf{NA}}$. The adaptive solution provides noticeable improvements in expected revenue when compared with the non-adaptive solution. Over all product categories, the average improvement in the expected revenue is 0.94%. There are product category and retailer combinations, where the improvement in the expected revenue provided by the adaptive solution can exceed 4%. A few percent improvements in the expected revenue translate into much larger percent improvements in the expected profit. The number of products in the adaptive solution is generally larger than those in the non-adaptive solution. If the adaptive solution offers the assortment S_0 in the first stage, then it can offer any assortment $S_i \subseteq S_0 \setminus \{i\}$ in the second stage to a customer with a preferred

	Off. size		Diff. in Exp. Rev.			Diff. in Mkt. Share		
Category	$\widehat{S}_{\mathsf{NA}}$	$\widehat{S}_{\mathsf{AD}}$	Avg.	95-th	Max	Full	Net	Rfn.
Apples	7.88	8.94	0.94	2.25	2.78	1.74	0.78	0.96
Berries	7.58	9.35	1.21	2.03	3.30	3.15	0.76	2.39
Candy	7.64	9.00	1.06	1.65	2.46	3.75	0.94	2.81
Chicken	13.90	15.77	0.98	2.41	2.44	2.07	0.69	1.37
Chips	8.92	10.33	0.97	1.59	3.56	3.54	1.38	2.15
Greek Yoghurt	11.63	13.62	0.83	1.46	2.35	2.74	0.46	2.28
Leafy Vegetables	16.62	18.82	0.91	1.63	2.84	2.03	0.64	1.39
Plain Milk	10.87	12.95	1.23	2.26	3.04	2.49	0.48	2.01
Seasonings	11.85	12.15	0.41	1.02	1.40	0.34	0.27	0.08
Sparkling Water	11.62	13.38	1.04	1.65	2.14	2.67	0.60	2.07
Tomatoes	8.52	9.59	1.03	2.43	4.12	1.97	0.86	1.11
Whole Eggs	7.27	7.73	0.62	1.59	1.95	1.49	0.11	1.38
Avg.	10.36	11.80	0.94	1.83	2.70	2.33	0.66	1.67

Table 4 Benefit from using an adaptive solution.

choice of product i, whereas if the non-adaptive solution offers the assortment S_0 in the first stage, then it has to offer the assortment $S_0 \setminus \{i\}$. Because the adaptive solution, intuitively speaking, has the flexibility to restrict the assortment in the second stage, it appears that it tends to start with larger assortments in the first stage. The full and net market shares for the adaptive solution are larger than their counterparts for the non-adaptive solution. In light of the fact that the adaptive solutions offer larger assortments, it is not surprising that they realize a larger full market share, but the adaptive solutions also improve the net market share by 0.66% on average.

The adaptive solution potentially offers different assortments than the non-adaptive solution both in the first and second stages. A sensible thought experiment is to consider the case where we offer the assortment $\overline{S}$ in the first stage and compare an adaptive solution that offers the assortment $\arg\max_{S \subseteq \overline{S} \setminus \{i\}} \overline{\mathrm{R}}(S)$ to a customer with preferred choice of product i with a non-adaptive solution that offers the assortment $\overline{S} \setminus \{i\}$ to a customer with preferred choice of product i. We have five weeks in the training dataset, which implies that we have five assortments for each product category and retailer combination. We refer to each one of these assortments as a historical assortment. For each historical assortment $\overline{S}_{\mathsf{HS}}$, we compute $\overline{S}_i = \arg\max_{S \subseteq \overline{S}_{\mathsf{HS}} \setminus \{i\}} \overline{\mathrm{R}}(S)$ and compare the expected revenues $\mathrm{R}(\overline{S}_{\mathsf{HS}}, \overline{S}_1, \ldots, \overline{S}_n)$ and $\mathrm{R}(\overline{S}_{\mathsf{HS}}, \overline{S}_{\mathsf{HS}} \setminus \{1\}, \ldots, \overline{S}_{\mathsf{HS}} \setminus \{n\})$. The former expected revenue is obtained from an adaptive solution, whereas the latter is obtained from a non-adaptive one.

We give our results in Table 5. The first column gives the product categories. The second, third, and fourth columns give the average, 95-th percentile, and maximum gap expected revenue gap between the adaptive and non-adaptive solutions described in the previous paragraph, where these statistics are computed over all retailers in a product category and their five historical assortments. The average improvement in expected revenue by using an adaptive solution is 9.26%. To be perfectly fair, if the assortment offered in the first stage is a poorly constructed assortment, then an adaptive solution can compensate for the poorly constructed assortment in the

Category	Diff. in Exp. Rev. Avg.	95-th	Max
Apples	10.38	19.58	24.02
Berries	11.69	20.48	24.68
Candy	9.46	14.26	15.65
Chicken	9.18	15.21	16.67
Chips	10.11	12.62	18.05
Greek Yoghurt	7.96	10.79	12.55
Avg.	9.80	15.49	18.60

Category	Diff. in Exp. Rev. Avg.	95-th	Max
Leafy Vegetables	11.55	18.79	24.42
Plain Milk	8.69	14.90	16.25
Seasonings	5.21	11.07	21.12
Sparkling Water	8.98	12.64	22.11
Tomatoes	8.32	11.69	11.83
Whole Eggs	9.61	17.71	20.20
Avg.	8.73	14.47	19.32

Table 5 Benefit from using an adaptive solution when the assortment in the first stage is fixed by the history.

second stage, whereas a non-adaptive solution does not have too much flexibility to compensate for the poorly constructed assortment. Nevertheless, proceeding with the understanding that our fitted multinomial logit model captures the choice process of the customers well, our results indicate that there is substantial potential to improve the expected revenue by carefully choosing the assortment offered in the second stage, even if we leave the assortment in the first stage unchanged.

To close this section, we discuss the drivers of expected revenue improvements provided by adaptive solutions. In Figure 2, we give scatter plots of average expected revenue improvements provided by adaptive solutions and the availability probability. In both charts in the figure, we have one data point for each product category. The horizontal axis gives the average product availability probability for the product category. The vertical axis gives the average expected revenue improvement provided by the adaptive solution when compared with the non-adaptive solution, where the average is computed over all retailers. The chart on the left focuses on the expected revenue gaps between the adaptive and non-adaptive solutions $(\widehat{S}_{\mathsf{AD}}, \widehat{S}_1, \ldots, \widehat{S}_n)$ and $(\widehat{S}_{\mathsf{NA}}, \widehat{S}_{\mathsf{NA}} \setminus \{1\}, \ldots, \widehat{S}_{\mathsf{NA}} \setminus \{n\})$, which we compared in Table 4, whereas the chart on the right focuses on the expected revenue gaps between the adaptive and non-adaptive solutions $(\overline{S}_{\mathsf{HS}}, \overline{S}_1, \ldots, \overline{S}_n)$ and $(\overline{S}_{\mathsf{HS}}, \overline{S}_{\mathsf{HS}} \setminus \{1\}, \ldots, \overline{S}_{\mathsf{HS}} \setminus \{n\})$, which we compared in Table 5. The dotted lines in both charts are the linear regression fits. In both charts, perhaps not so surprisingly, the expected revenue improvement provided by an adaptive solution over a non-adaptive one is more pronounced when there is a larger probability that the products are unavailable. Naturally, when all products are available, for example, the expected revenue gap between the two solutions is zero.

8. Performance of the Approximation Scheme

We test the practical performance of the PTAS for the adaptive variant by using synthetically generated test problems. We explain our experimental setup and give our numerical results.

Experimental Setup: We randomly generate a large number of instances of our adaptive variant. We use our PTAS to compute an approximate solution for each problem instance. We

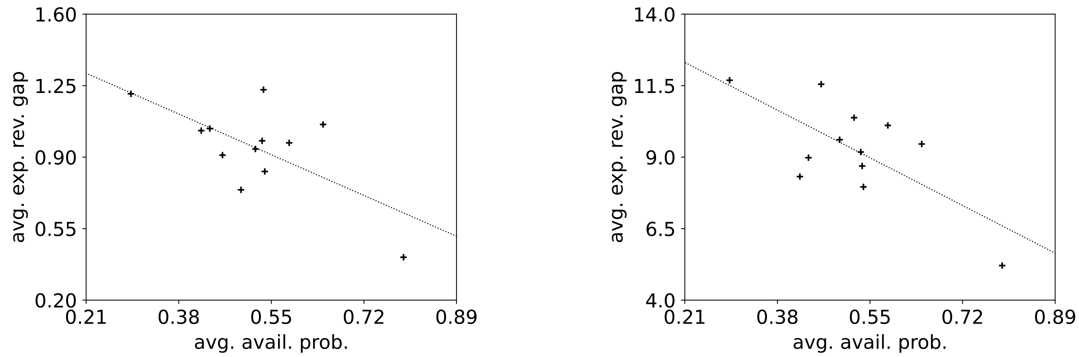

Figure 2 Expected revenue improvements from adaptive solutions and product availability probabilities.

use the linear programming relaxation of the mixed integer program in Appendix C to efficiently compute an upper bound on the optimal expected revenue for each problem instance. We compare the expected revenue provided by our approximate solution with the upper bound on the optimal expected revenue. We generate our problem instances as follows. The number of products is fixed at $n=20$. We sample the adjusted revenue r_i of each product i from the uniform distribution over $[0,1]$. For product i, recall that the adjusted revenue r_i is the product of the revenue f_i and availability probability q_i. To come up with the preference weights, for each product i, we sample γ_i from the uniform distribution over $[\alpha, 1-\alpha]$, where α is a parameter that we vary to control the variability of the preference weights. We set the preference weight product i as $v_i = \frac{(1-P_0)\,\gamma_i}{P_0 \sum_{j\in N}\gamma_j}$, where P_0 is another parameter that we vary. In this way, if we offer all products, then a customer makes a purchase with probability $\frac{\sum_{i\in N} v_i}{1+\sum_{i\in N} v_i} = \frac{(1-P_0)/P_0}{1+(1-P_0)/P_0} = 1-P_0$, so the parameter P_0 controls the probability that a customer makes a purchase. To come up with the availability probabilities, we designate each product as a high availability product with probability θ and as a low availability product with probability $1-\theta$. If product i is high availability, then we sample the availability probability q_i from the uniform distribution over $[0.8,1]$. Otherwise, we sample from the uniform distribution over $[0.2,0.8]$. We vary the parameter θ, which controls the fraction of high-availability products. Once we generate the problem parameters through the approach discussed so far, we post-process them in two different ways. First, we leave them unchanged. Second, we re-order the adjusted revenues and unavailability probabilities so that $r_1 \geq r_2 \geq \ldots \geq r_n$ and $p_1 \leq p_2 \leq \ldots \leq p_n$. In this way, the less expensive products tend to be less available.

We use the tuple (θ, T, P_0, α) to capture the parameter configuration for our test problems, where θ, P_0 and α are as in the previous paragraph and we have $T \in \{U, O\}$, where $T=U$ corresponds to leaving the adjusted revenues and preference weights unordered, whereas $T=O$ corresponds to ordering them. Varying $\theta \in \{0.5, 0.75\}$, $T \in \{U, O\}$, $P_0 \in \{0.1, 0.3\}$ and $\alpha \in \{0, 0.2, 0.4\}$, we get 24 parameter configurations. In each parameter configuration, we generate 30 problem instances.

In Appendix D, we give a slightly tighter version of the surrogate problem in (8) that yields the same approximation guarantee with a bit smaller computational effort. This surrogate problem is less interpretable than the one in (8), but if we define the set M_{H} as the set of $\lceil 1/\epsilon \rceil$ products in S^*_{H} with the largest weights, then this surrogate problem approximates the expected revenue from below within a factor of $\frac{2+\epsilon}{2+2\epsilon}$. By Lemma 4.3, the surrogate problem in (8) approximates the expected revenue from below within a factor of $\frac{1}{1+\epsilon}$. Because $\frac{2+\epsilon}{2+2\epsilon}/\frac{1}{1+\epsilon} = 1 + \frac{\epsilon}{2}$, we get a slightly tighter lower bound. In our numerical experiments, to construct the set M_{H}, we guess one product with the largest weight in S^*_{H}. We use $\epsilon = 0.1$ to construct our super-optimal solutions. In this way, we get a 0.3-approximate solution to the adaptive variant. This approximation guarantee is loose, but our PTAS with such an approximation guarantee performs remarkably well.

Experimental Results: We give our results in Table 6. The first column gives the parameter configuration for our problem instances by using the tuple (θ, T, P_0, α). Recall that we generate 30 problem instances in each parameter configuration. For each problem instance, we find a solution using our PTAS and compute an upper bound on the optimal expected revenue. The second column gives the average percent gap between the upper bound and the expected revenue from the solution obtained by our PTAS, where the average is computed over the problem instances in a particular parameter configuration. In other words, using Rev^k to denote the expected revenue from the solution obtained by our PTAS and UB^k to denote the upper bound for problem instance k, the second column gives the average of the data $\{100 \times \frac{\mathsf{UB}^k - \mathsf{Rev}^k}{\mathsf{UB}^k} : k = 1, \ldots, 30\}$. The third and fourth columns, respectively, give the standard deviation and maximum of the same data.

The results in Table 6 indicate that the practical performance of our PTAS is quite good. The average optimality gap over all parameter configurations is below 1%. The largest optimality gap that we observe over all problem instances is 3.07%. To put these gaps into perspective, we emphasize that we compare the expected revenue from the solutions obtained by our PTAS with an upper bound on the optimal expected revenue, rather than the expected revenue itself. Naturally, unless we compute the optimal solutions, it is difficult to tell what fraction of the gaps is due to the fact that our PTAS potentially yields suboptimal solutions and what fraction of the gaps is due to the fact that we do not compare against the optimal expected revenue. Nevertheless, even the upper bounds on the optimality gaps are rather small. Moreover, the gaps are stable across different parameter configurations.

9. Conclusions

We studied assortment optimization problems for online platforms that do not have full visibility into product availabilities. We considered both non-adaptive and adaptive variants, where the

Param. Conf. (θ, T, P_0, α)	Avg.	Std. dev.	Max
$(0.5, U, 0.3, 0.0)$	0.66	0.47	1.78
$(0.5, U, 0.3, 0.2)$	0.73	0.52	1.94
$(0.5, U, 0.3, 0.4)$	0.91	0.65	2.55
$(0.5, U, 0.1, 0.0)$	0.81	0.51	2.51
$(0.5, U, 0.1, 0.2)$	1.06	0.62	2.23
$(0.5, U, 0.1, 0.4)$	0.88	0.54	2.21
$(0.5, O, 0.3, 0.0)$	0.41	0.31	1.36
$(0.5, O, 0.3, 0.2)$	0.54	0.42	1.63
$(0.5, O, 0.3, 0.4)$	0.62	0.49	2.09
$(0.5, O, 0.1, 0.0)$	0.43	0.36	1.68
$(0.5, O, 0.1, 0.2)$	0.76	0.57	2.63
$(0.5, O, 0.1, 0.4)$	0.54	0.46	1.73
Avg.	0.70	0.49	2.03

Param. Conf. (θ, T, P_0, α)	Avg.	Std. dev.	Max
$(0.75, U, 0.3, 0.0)$	0.68	0.48	1.94
$(0.75, U, 0.3, 0.2)$	0.56	0.42	1.41
$(0.75, U, 0.3, 0.4)$	0.83	0.79	2.95
$(0.75, U, 0.1, 0.0)$	0.69	0.46	1.77
$(0.75, U, 0.1, 0.2)$	0.95	0.72	3.07
$(0.75, U, 0.1, 0.4)$	0.72	0.68	2.81
$(0.75, O, 0.3, 0.0)$	0.47	0.36	1.54
$(0.75, O, 0.3, 0.2)$	0.48	0.44	1.77
$(0.75, O, 0.3, 0.4)$	0.54	0.38	1.47
$(0.75, O, 0.1, 0.0)$	0.47	0.35	1.65
$(0.75, O, 0.1, 0.2)$	0.81	0.72	2.78
$(0.75, O, 0.1, 0.4)$	0.66	0.50	1.84
Avg.	0.66	0.53	2.08

Table 6 Performance of our PTAS for the adaptive variant.

non-adaptive variant pre-computes the assortments offered as possibilities for the preferred and replacement options, whereas the adaptive variant takes the preferred option of the customer into consideration when offering the assortment for the replacement option. We characterized the gap in the optimal expected revenues for the non-adaptive and adaptive solutions. Both variants are NP-hard. We developed approximation schemes. Our case study on a dataset from a grocery delivery platform indicated that explicitly incorporating replacement options and switching from non-adaptive to adaptive solutions can have a significant revenue impact. There are several research directions to pursue. We allow specifying one replacement option. This approach is consistent with virtually every platform with replacement options, but one can, at least in theory, allow customers to specify more than one replacement option. Although we can construct an expression for the expected revenue function under multiple replacement options, the structure of the expected revenue function is rather different and the extension of our approximation schemes to multiple replacement options appears to be difficult. Also, we built on the multinomial logit model to develop our assortment optimization problems. Conceptually, one can build on any choice model that uses the random utility maximization principle, but it appears to be difficult to obtain reasonably simple expressions for the expected revenue even under close relatives of the multinomial logit model. One can study analogues of our model under other choice models. Lastly, one can consider incorporating operational constraints that limit what kind of assortments we can offer to the customers.

References

Anderson, E. T., G. J. Fitzsimons, D. Simester. 2006. Measuring and mitigating the costs of stockouts. *Management Science* **52**(11) 1751–1763.

Aouad, A., J. Feldman, D. Segev, D. J. Zhang. 2018. Click-based MNL: Algorithmic frameworks for modeling click data in assortment optimization. Tech. rep., Washington University, St. Louis, MO.

Bai, Y., J. Feldman, D. Segev, H. Topaloglu, L. Wagner. 2023. Assortment optimization under the multi-purchase multinomial logit choice model. *Operations Research* (forthcoming).

Berbeglia, G., G. Joret. 2020. Assortment optimisation under a general discrete choice model: A tight analysis of revenue-ordered assortments. *Algorithmica* **82**(4) 681–720.

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

Chen, Y.-C., D. Mitrofanov. 2023. A nonparametric stochastic set model: Identification, optimization, and prediction. Tech. rep., Boston College, Boston, MA. Available at arXiv:2302.04354.

Desir, A., V. Goyal, J. Zhang. 2022. Capacitated assortment optimization: Hardness and approximation. *Operations Research* **70**(2) 893–904.

El Housni, O., H. Topaloglu. 2023. Joint assortment optimization and customization under a mixture of multinomial logit models: On the value of personalized assortments. *Operations Research* **71**(4) 1197–1215.

Feldman, J., D. Segev. 2022. Technical note – The multinomial logit model with sequential offerings: Algorithmic frameworks for product recommendation displays. *Operations Research* **70**(4) 2162–2184.

Feldman, J., H. Topaloglu. 2018. Technical note: Capacitated assortment optimization under the multinomial logit model with nested consideration sets. *Operations Research* **66**(2) 380–391.

Fitzsimons, G. J. 2000. Consumer response to stockouts. *Journal of Consumer Research* **27**(2) 249–266.

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

Gallego, G., H. Topaloglu. 2019. *Revenue Management and Pricing Analytics*. International Series in Operations Research & Management Science, Springer, New York, NY.

Gao, P., Y. Ma, N. Chen, G. Gallego, A. Li, P. Rusmevichientong, H. Topaloglu. 2021. Assortment optimization and pricing under the multinomial logit model with impatient customers: Sequential recommendation and selection. *Operations Research* **69**(5) 1509–1532.

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

Jagabathula, S., D. Mitrofanov, G. Vulcano. 2022. Personalized retail promotions through a directed acyclic graph-based representation of customer preferences. *Operations Research* **70**(2) 641–665.

Jagabathula, S., G. Vulcano. 2018. A partial-order-based model to estimate individual preferences using panel data. *Management Science* **64**(4) 1609–1628.

Jagabathula, Srikanth, Paat Rusmevichientong. 2019. The limit of rationality in choice modeling: Formulation, computation, and implications. *Management Science* **65**(5) 2196–2215.

Knight, B., D. Mitrofanov. 2022. Disclosing low product availability: An online retailer's strategy for mitigating stockout risk. Tech. rep., Boston College, Boston, MA. Available at SSRN 4137758.

Liu, N., Y. Ma, H. Topaloglu. 2020. Assortment optimization under the multinomial logit model with sequential offerings. *INFORMS Journal on Computing* **32**(3) 835–853.

Musalem, A., M. Olivares, E. T. Bradlow, C. Terwiesch, D. Corsten. 2010. Structural estimation of the effect of out-of-stocks. *Management Science* **56**(7) 1180–1197.

Rusmevichientong, P., Z.-J. M. Shen, D. B. Shmoys. 2010. Dynamic assortment optimization with a multinomial logit choice model and capacity constraint. *Operations research* **58**(6) 1666–1680.

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

Sumida, M., G. Gallego, P. Rusmevichientong, H. Topaloglu, J. Davis. 2021. Revenue-utility tradeoff in assortment optimization under the multinomial logit model with totally unimodular constraints. *Management Science* **67**(5) 2845–2869.

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

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

Wagner, L., V. Martinez-de-Albeniz. 2020. Pricing and assortment strategies with product exchanges. *Operations Research* **68**(2) 453–466.

Electronic Companion: Assortment Optimization with Replacement Options for Retail Platforms with Stockout Risk

Appendix A: Computational Complexity

We start by showing that the adaptive variant of problem (4) is NP-hard. Noting (3), as well as the fact that we have $S_i \subseteq S_0 \setminus \{i\}$ for all $i \in N$ in the adaptive variant, the adaptive variant is

$$\max_{\substack{(S_0, S_1, \ldots, S_n) \in 2^N \times 2^N \times \ldots \times 2^N, \\ S_i \subseteq S_0 \setminus \{i\} \; \forall \, i \in N}} \left\{ \sum_{i \in S_0} \frac{v_i}{1+V(S_0)} \left(q_i \, f_i + (1-q_i) \sum_{j \in S_i} \frac{v_j}{1+V(S_i)} \, q_j \, f_j \right) \right\}$$
$$= \max_{S_0 \subseteq N} \left\{ \sum_{i \in S_0} \frac{v_i}{1+V(S_0)} \left(q_i \, f_i + (1-q_i) \max_{S_i \subseteq S_0 \setminus \{i\}} \left\{ \sum_{j \in S_i} \frac{v_j}{1+V(S_i)} \, q_j \, f_j \right\} \right) \right\}. \quad (13)$$

Focusing on the objective function of the outer maximization problem on the right side above, as a function of S_0, we use $\mathcal{R}_{\mathsf{AD}}(S_0)$ to denote the objective function of this maximization problem. Thus, we can express the adaptive variant as $\max_{S_0 \subseteq N} \mathcal{R}_{\mathsf{AD}}(S_0)$. The inner maximization problem on the right side above finds an assortment $S_i \subseteq S_0 \setminus \{i\}$ that maximizes the expected revenue under the standard multinomial logit model. We can solve this inner maximization problem in polynomial time. Thus, for given S_0, we can compute $\mathcal{R}_{\mathsf{AD}}(S_0)$ in polynomial time. To characterize the computational complexity of the adaptive variant, we consider the feasibility version of this problem. In the feasibility version of the adaptive variant, given an expected revenue threshold K, we ask whether there exists $S_0 \subseteq N$ such that $\mathcal{R}_{\mathsf{AD}}(S_0) \geq K$. To show that the feasibility version of the adaptive variant is NP-complete, we use a reduction from the partition problem, which is a well know NP-complete problem; see Garey and Johnson (1979). The partition problem is defined as follows. We are given a set of items $N = \{1, \ldots, n\}$. The size of item i is $c_i \in \mathbb{Z}_+$. Assuming that the total size of the items is even, set $\sum_{i \in N} c_i = 2T$ for some $T \in \mathbb{Z}_+$. In the partition problem, we ask whether there exists $S \subseteq N$ such that $\sum_{i \in S} c_i = T$.

In the next theorem, we use a reduction from the partition problem to show that the feasibility version of the adaptive variant is NP-complete.

Theorem A.1 (Complexity for Adaptive Variant) *The feasibility version of the adaptive variant is NP-complete.*

Proof: We are given an arbitrary instance of the partition problem with the set of items $N = \{1, \ldots, n\}$. The size of item i is c_i. The total size of the items is $\sum_{i \in N} c_i = 2T$. Corresponding to this instance of the partition problem, we define an instance of the feasibility version of the adaptive variant as follows. There are $n+1$ products. For each $i=1,\dots,n$, the revenue, availability probability and preference weight of product i are $f_i=1$, $q_i=1$ and $v_i=c_i$. For product $n+1$, its revenue, availability probability and preference weight are $f_{n+1}=\frac{2+4(T+1)^2-T^2}{(T+1)^2}$, $q_{n+1}=\frac{1}{2}$ and $v_{n+1}=2(T+1)^2$. The expected revenue threshold is $K=2$. We argue that there exists an assortment $S_0\subseteq\{1,\dots,n+1\}$ that provides an expected revenue of two or more if and only if there exists a subset $S\subseteq N$ such that $\sum_{i\in N}c_i=T$. Noting that $f_i=1$ for all $i=1,\dots,n$, if we do not offer product $n+1$, then the objective function in (13) cannot exceed one. We can check that $\frac{v_{n+1}}{1+v_{n+1}}\,q_{n+1}\,f_{n+1}<2$. Thus, if we offer only product $n+1$, then the objective function in (13) cannot exceed two. Thus, if we want to obtain an expected revenue of two or more, then we need to offer product $n+1$ along with a subset of the first n products. We compute the objective function of the outer maximization problem in (13) at an assortment of the form $S_0=S\cup\{n+1\}$ with $S\subseteq\{1,\dots,n\}$. For all $i=1,\dots,n$, we have $f_i=1$ and $q_i=1$, so if $S\subseteq\{1,\dots,n\}$, then $\max_{Q\subseteq S}\frac{\sum_{j\in Q}v_j\,q_j\,f_j}{1+V(Q)}=\frac{V(S)}{1+V(S)}$. Thus, once again noting that $f_i=1$ and $q_i=1$ for all $i=1,\dots,n$ and $q_{n+1}=\frac{1}{2}$, the objective function of the outer maximization problem in (13) at $S\cup\{n+1\}$ for some $S\subseteq\{1,\dots,n\}$ is

$$\begin{aligned}\mathcal{R}_{\mathsf{AD}}(S\cup\{n+1\}) &= \sum_{i\in S}\frac{v_i}{1+V(S)+v_{n+1}}+\frac{v_{n+1}}{1+V(S)+v_{n+1}}\left(\frac{1}{2}\,f_{n+1}+\frac{1}{2}\times\frac{V(S)}{1+V(S)}\right)\\ &= \frac{V(S)}{1+V(S)+2(T+1)^2}+\frac{2(T+1)^2}{1+V(S)+2(T+1)^2}\left(\frac{2+4(T+1)^2-T^2}{2(T+1)^2}+\frac{1}{2}\times\frac{V(S)}{1+V(S)}\right)\\ &= \frac{\big(1+V(S)\big)\,V(S)+\big(1+V(S)\big)\,\big(2+4(T+1)^2-T^2\big)+(T+1)^2\,V(S)}{\big(1+V(S)\big)\,\big(1+V(S)+2(T+1)^2\big)},\end{aligned}\tag{14}$$

where the last equality simply follows by arranging the terms. We argue that there exists $S\subseteq\{1,\dots,n\}$ with $\mathcal{R}_{\mathsf{AD}}(S\cup\{n+1\})\ge 2$ if and only if there exists $S\subseteq N$ with $\sum_{i\in N}c_i=T$.

For notational brevity, we fix $V(S)=\eta$, in which case, by the chain of equalities above, we have $\mathcal{R}_{\mathsf{AD}}(S\cup\{n+1\})\ge 2$ if and only if $\frac{(1+\eta)\,\eta+(1+\eta)(2+4(T+1)^2-T^2)+(T+1)^2\eta}{(1+\eta)\,(1+\eta+2(T+1)^2)}\ge 2$. Multiplying the last inequality by $(1+\eta)\,(1+\eta+2(T+1)^2)$ and collecting all of the terms that involve the common multiplier $1+\eta$ to the right side of the inequality, we write the the last inequality equivalently as $(T+1)^2\,\eta\ge(1+\eta)\,(2+2\eta+4(T+1)^2-\eta-2-4(T+1)^2+T^2)$. The right side of the last inequality is equal to $(1+\eta)\,(\eta+T^2)$, so $\mathcal{R}_{\mathsf{AD}}(S\cup\{n+1\})\ge 2$ if and only if $(T+1)^2\,\eta\ge(1+\eta)\,(\eta+T^2)$. The last inequality simplifies to $(\eta-T)^2\le 0$, so replacing η with $V(S)$, we have $\mathcal{R}_{\mathsf{AD}}(S\cup\{n+1\})\ge 2$ if and only if $(V(S)-T)^2\le 0$, but the last inequality holds if and only if $T=V(S)$. Thus, noting that $V(S)=\sum_{i\in S}v_i=\sum_{i\in S}c_i$, there exists a subset $S\subseteq\{1,\dots,n\}$ such that $\mathcal{R}_{\mathsf{AD}}(S\cup\{n+1\})\ge 2$ if and only if there exists a subset $S\subseteq\{1,\dots,n\}$ such that $\sum_{i\in S}c_i=T$. ■

We turn to showing that the non-adaptive variant of problem (4) is NP-hard. This result will follow as a corollary to the theorem above. Noting that $S_i=S_0\setminus\{i\}$ for all $i\in N$ in the non-adaptive variant, we can formulate the non-adaptive variant using the assortment in the first stage as the only decision variable. In particular, using (3), the non-adaptive variant is given by

$$\max_{S_0 \subseteq N} \left\{ \sum_{i \in S_0} \frac{v_i}{1+V(S_0)} \left(q_i\, f_i + (1-q_i)\, \frac{\sum_{j \in S_0 \setminus \{i\}} v_j\, q_j\, f_j}{1+V(S_0 \setminus \{i\})} \right) \right\}. \tag{15}$$

We use $\mathcal{R}_{\mathsf{NA}}(S_0)$ to denote the objective function in (15). We define the feasibility version of the non-adaptive variant similar to that of the adaptive variant. We have the following result.

Corollary A.2 (Complexity for Non-Adaptive Variant) *The feasibility version of the non-adaptive variant is NP-complete.*

Proof: Given an arbitrary instance of the partition problem, we define an instance of the feasibility version of the non-adaptive variant precisely as in the proof of Theorem A.1. Recall that the set of products in this instance of the feasibility version of the non-adaptive variant is $\{1,\ldots,n+1\}$ and the expected revenue threshold is $K=2$. By following the same argument in the proof of Theorem A.1 but using the objective function in (15), we can argue that if we offer the assortment $S \subseteq \{1,\ldots,n\}$, then $\mathcal{R}_{\mathsf{NA}}(S) < 1$, whereas if we offer only product $n+1$, then $\mathcal{R}_{\mathsf{NA}}(\{n+1\}) < 2$. Thus, if we want to obtain an expected revenue of two or more, then we need to offer an assortment of the form $S \cup \{n+1\}$ with $S \subseteq \{1,\ldots,n\}$. Considering an assortment of the form $S \cup \{n+1\}$ with $S \subseteq \{1,\ldots,n\}$, using the objective function in (15), the expected revenue $\mathcal{R}_{\mathsf{NA}}(S \cup \{n+1\})$ evaluates to precisely the same expression in (14). In this case, we can follow the same argument in the proof of Theorem A.1 to argue that there exists a subset $S \subseteq \{1,\ldots,n\}$ such that $\mathcal{R}_{\mathsf{NA}}(S \cup \{n+1\}) \geq 2$ if and only if there exists a subset $S \subseteq \{1,\ldots,n\}$ such that $\sum_{i \in S} c_i = T$. Therefore, given an arbitrary instance of the partition problem, we can reduce this instance to an instance of the feasibility version of the non-adaptive variant. ■

Next, we consider obtaining a super-optimal solution to the multi-dimensional knapsack problem given in problem (9).

Appendix B: Obtaining a Super-Optimal Solution

Given a ground set $N = \{1,\ldots,n\}$ and the subsets M and U with $M \subseteq U \subseteq N$, as well as two generic sets of constraint indices G and H, consider the multi-dimensional knapsack problem

$$Z^* \;=\; \max_{\substack{S \subseteq N:\\ M \subseteq S \subseteq U}} \left\{ \sum_{i \in S} c_i \;:\; \sum_{i \in S} u_{ij} \leq g_j \quad \forall\, j \in G, \quad \sum_{i \in S} v_{ij} \geq h_j \quad \forall\, j \in H \right\}. \tag{16}$$

In the multi-dimensional knapsack problem in (9), some of the constraints are of the packing type, whereas some of the constraints are of the covering type. Furthermore, noting the constraint

$M_{\mathsf{H}} \subseteq S \subseteq N_{\mathsf{H}} \cup N_{\mathsf{L}}$ in this problem, the products in M_{H} have to be included, whereas the products in $N \setminus (N_{\mathsf{H}} \cup N_{\mathsf{L}})$ have to be excluded. We view problem (16) as a generic version of problem (9), where the first $|G|$ constraints are of packing type, whereas the last $|H|$ constraints are of covering type. Due to the constraint $M \subseteq S \subseteq U$, the items in M have to be included, whereas the items in $N \setminus U$ have to be excluded. Thus, we focus on obtaining a super-optimal solution to problem (16). We refer to S as a super-optimal solution to problem (16) if it satisfies $M \subseteq S \subseteq U$, $\sum_{i \in S} u_{ij} \leq (1+\epsilon)\, g_j$ for all $j \in G$, $\sum_{i \in S} v_{ij} \geq \frac{1}{1+\epsilon}\, h_j$ for all $j \in H$ and $\sum_{i \in S} c_i \geq Z^*$. To obtain a super-optimal solution, we express the constraint $\sum_{i \in S} u_{ij} \leq g_j$ equivalently as $\sum_{i \in S} \frac{4n\, u_{ij}}{\epsilon\, g_j} \leq \frac{4n}{\epsilon}$, in which case, using $\lfloor \cdot \rfloor$ to denote the round down function, a relaxation of the last constraint is given by $\sum_{i \in S} \lfloor \frac{4n\, u_{ij}}{\epsilon\, g_j} \rfloor \leq \lceil \frac{4n}{\epsilon} \rceil$. Similarly, a relaxation of the constraint $\sum_{i \in S} v_{ij} \geq h_j$ is given by $\sum_{i \in S} \lceil \frac{4n\, v_{ij}}{\epsilon\, h_j} \rceil \geq \lfloor \frac{4n}{\epsilon} \rfloor$. In this case, letting $\overline{u}_{ij} = \lfloor \frac{4n\, u_{ij}}{\epsilon\, g_j} \rfloor$ and $\overline{v}_{ij} = \lceil \frac{4n\, v_{ij}}{\epsilon\, h_j} \rceil$, to obtain a relaxation of problem (16), we consider the multi-dimensional knapsack problem

$$\max_{\substack{S \subseteq N:\\ M \subseteq S \subseteq U}} \left\{ \sum_{i \in S} c_i \ : \ \sum_{i \in S} \overline{u}_{ij} \leq \left\lceil \frac{4n}{\epsilon} \right\rceil \quad \forall\, j \in G, \quad \sum_{i \in S} \overline{v}_{ij} \geq \left\lfloor \frac{4n}{\epsilon} \right\rfloor \quad \forall\, j \in H \right\}. \tag{17}$$

In the next proposition, we show that an optimal solution to problem (17) corresponds to a super-optimal solution to problem (16).

Proposition B.1 (Super-Optimal to Knapsack) *An optimal solution to problem* (17) *is a super-optimal solution to problem* (16)*.*

Proof: Let $\widehat{S}$ be an optimal solution to problem (17). Thus, we have $M \subseteq \widehat{S} \subseteq U$. Furthermore, using the fact that $x \leq \lfloor x \rfloor + 1$ for $x \geq 0$, we have the chain of inequalities

$$\begin{aligned}
\sum_{i \in \widehat{S}} u_{ij} \ &= \ \frac{\epsilon\, g_j}{4n} \sum_{i \in \widehat{S}} \frac{4n\, u_{ij}}{\epsilon\, g_j} \ \leq \ \frac{\epsilon\, g_j}{4n} \sum_{i \in \widehat{S}} \left(\left\lfloor \frac{4n\, u_{ij}}{\epsilon\, g_j} \right\rfloor + 1 \right) \\
&\stackrel{(a)}{=} \ \frac{\epsilon\, g_j}{4n} \sum_{i \in \widehat{S}} (\overline{u}_{ij} + 1) \ \stackrel{(b)}{\leq} \ \frac{\epsilon\, g_j}{4n} \left(\left\lceil \frac{4n}{\epsilon} \right\rceil + n \right) \ \stackrel{(c)}{\leq} \ \frac{\epsilon\, g_j}{4n} \left(\frac{4n}{\epsilon} + 1 + n \right) \\
&= \ g_j \left(1 + \frac{\epsilon}{4n} + \frac{\epsilon}{4} \right) \ \leq \ g_j \left(1 + \frac{\epsilon}{2} \right) \ \leq \ g_j\, (1+\epsilon),
\end{aligned}$$

where (a) uses the definition of $\overline{u}_{ij}$, (b) holds because $\widehat{S}$ satisfies the first constraint in (17), as well as noting that $|\widehat{S}| \leq n$ and (c) follows by the fact that $\lceil x \rceil \leq x + 1$ for $x \geq 0$. Using a symmetric argument, we can also show that $\sum_{i \in \widehat{S}} v_{ij} \geq h_j\, (1 - \frac{\epsilon}{2})$, but noting that $\epsilon \in (0,1)$, we have $(1 - \frac{\epsilon}{2})\,(1+\epsilon) = 1 + \frac{\epsilon}{2} - \frac{\epsilon^2}{2} \geq 1$, so $1 - \frac{\epsilon}{2} \geq \frac{1}{1+\epsilon}$. In this case, having $\sum_{i \in \widehat{S}} v_{ij} \geq h_j\, (1 - \frac{\epsilon}{2})$ implies that $\sum_{i \in \widehat{S}} v_{ij} \geq h_j\, \frac{1}{1+\epsilon}$. Finally, because problem (17) is a relaxation of problem (16), as discussed just before the proposition, the optimal objective value of (17) is at least as large as that of problem (16), so $\sum_{i\in\widehat{S}} c_i \ge Z^*$. The inequalities established so far in the proof show that $\widehat{S}$ is a super-optimal solution to problem (16). ■

By Proposition B.1, we can solve problem (17) to obtain a super-optimal solution for problem (16). We can solve problem (17) using a dynamic program. In the dynamic program, the epochs correspond to the items in the set N. The action in the epoch for item i corresponds to whether we include item i in the solution. Noting that there are $|G|+|H|$ constraints in problem (17), the state variable at a particular epoch keeps track of the value of the left side of each of these constraints as a function of the items that we include in the solution up to the current epoch. The important point is that $\overline{u}_{ij}$ and $\overline{v}_{ij}$ are integers, so each component of the state variable takes integer values. In particular, using the vector $(\boldsymbol{x},\boldsymbol{y})\in\mathbb{Z}_+^{|G|+|H|}$ as the state variable, using $\boldsymbol{\delta}_j\in\{0,1\}^{|G|}$ to denote the unit vector with a one in component $j\in G$, using $\boldsymbol{\eta}_j\in\{0,1\}^{|H|}$ to denote the unit vector with a one in component $j\in H$ and letting $\mathbf{1}(\cdot)$ be the indicator function, we can obtain an optimal solution to problem (17) through the dynamic program

$$J_i(\boldsymbol{x},\boldsymbol{y}) \;=\; \max_{\substack{z_i\,\in\,\{0,1\}:\\ \mathbf{1}(i\in M)\,\le\, z_i\,\le\,\mathbf{1}(i\in U)}} \left\{ c_i\, z_i + J_{i+1}\left(\boldsymbol{x} + z_i \sum_{j\in G} \overline{u}_{ij}\,\boldsymbol{\delta}_j\,,\,\boldsymbol{y} - z_i \sum_{j\in H} \overline{v}_{ij}\,\boldsymbol{\eta}_j\right)\right\}. \tag{18}$$

The boundary condition in the dynamic program is $J_{n+1}(\boldsymbol{x},\boldsymbol{y})=-\infty$ if $x_j>\lceil 4n/\epsilon\rceil$ for some $j\in G$ or $y_j<0$ for some $j\in H$. Otherwise, we have $J_{n+1}(\boldsymbol{x},\boldsymbol{y})=0$. For $i=1,\ldots,n$, we also set $J_i(\boldsymbol{x},\boldsymbol{y})=-\infty$ if $x_j>\lceil 4n/\epsilon\rceil$ for some $j\in G$ or $y_j<0$ for some $j\in H$. The decision variable z_i takes value one if we include item i in the solution. The constraint $\mathbf{1}(i\in M)\le z_i\le \mathbf{1}(i\in U)$ ensures that an item in M must be included in the solution, whereas an item not in U must not be included in the solution. Once we decide whether we include item i, we update the value of the left side of the constraints as a function of the items included up to the current epoch. Because the first $|G|$ constraints are packing, but the last $|H|$ constraints are covering, we count up the left side of the first $|G|$ constraints, but count down the left side of the last $|H|$ constraints. The vector $\sum_{j\in H}\lfloor 4n\epsilon\rfloor\,\boldsymbol{\eta}_j\in\mathbb{Z}_+^{|H|}$ has components of all equal to $\lfloor 4n\epsilon\rfloor$, so using $\mathbf{0}\in\mathbb{Z}_+^{|G|}$ to denote a vector of all zeros, the optimal objective value of problem (17) is $J_1(\mathbf{0},\sum_{j\in H}\lfloor 4n/\epsilon\rfloor\,\boldsymbol{\eta}_j)$.

In the next proposition, we calculate the running time to solve the dynamic program in (18) to give an algorithm to obtain a super-optimal solution to problem (16).

Proposition B.2 (Dynamic Program) *There exists an algorithm to obtain a super-optimal solution to problem* (16) *in a running time of* $O(n\,(\frac{n}{\epsilon})^{|G|+|H|})$.

Proof: By the boundary condition of the dynamic program in (18), we have $J_i(\boldsymbol{x},\boldsymbol{y})=-\infty$ if $x_j>\lceil 4n/\epsilon\rceil$ for some $j\in G$ or $y_j<0$ for some $j\in H$. Furthermore, considering the state variable

$(\boldsymbol{x}, \boldsymbol{y})$ in epoch i, the value of the state variable x_j can only go up, whereas the value of the state variable y_j can only go down at the next epoch. Lastly, we need to compute $J_1(\mathbf{0}, \sum_{j \in H} \lfloor 4n\epsilon \rfloor \boldsymbol{\eta}_j)$ through the dynamic program. Therefore, we need to compute the value function $J_i(\boldsymbol{x}, \boldsymbol{y})$ for the values of the state variable that satisfy $0 \le x_j \le \lceil 4n\epsilon \rceil$ for all $j \in G$ and $0 \le y_j \le \lfloor 4n\epsilon \rfloor$ for all $j \in G$, which implies that we have $O((\frac{n}{\epsilon})^{|G|+|H|})$ values of the state variable at each epoch. For each value of the state variable, we can solve the maximization problem in (18) in $O(1)$ operations. There are n epochs. Thus, we can solve the dynamic program in (18) in a running time of $O(n\,(\frac{n}{\epsilon})^{|G|+|H|})$, which, in turn, yields a super-optimal solution to problem (16). ∎

Appendix C: Mixed Integer Programming Formulations

We give mixed integer programs to get near-optimal solutions with desired accuracy for the adaptive and non-adaptive variants. We also use these mixed integer programs to compute upper bounds.

Mixed Integer Program for the Adaptive Variant:

We can solve the assortment optimization problem under the multinomial logit model through a linear program. In particular, using the decision variables $\boldsymbol{x} = (x_1, \ldots, x_n)$ and x_0, we have

$$\begin{aligned}\overline{\mathrm{F}}(S) &= \max_{Q \subseteq S} \overline{\mathrm{R}}(S) \\ &= \max_{(\boldsymbol{x}, x_0) \in [0,1]^{n+1}} \left\{ \sum_{i \in N} r_i\, x_i \;:\; \sum_{i \in N} x_i + x_0 \le 1,\; \frac{x_i}{v_i} \le x_0 \;\forall\, i \in N, \quad x_i \le \mathbf{1}(i \in S) \quad \forall\, i \in N \right\}. \quad (19)\end{aligned}$$

In the linear program, we interpret x_i as the probability that a customer purchases product i, whereas we interpret x_0 as the probability that a customer leaves without a purchase. The objective function accounts for the expected revenue from a customer. The first constraint ensures that a customer can either purchase a product or leave without a purchase. The second constraint, intuitively speaking, ensures that the choices of the customers are aligned with the multinomial logit model. The third constraint ensures that a customer cannot purchase a product that is not in the set S because such a product cannot be offered. Solving the assortment optimization problem under the multinomial logit model through a linear program goes back to Sumida et al. (2021).

Turning to the adaptive variant, letting $v_{\min} = \min_{i \in N} v_i$ and $v_{\max} = \max_{i \in N} v_i$, for any non-empty assortment $S \subseteq N$, we have $v_{\min} \le V(S) \le n\, v_{\max}$. We divide the interval $[v_{\min}, n\, v_{\max}]$ into K subintervals by using the grid points $v_{\min} = \nu^0 < \nu^1 < \ldots < \nu^{K-1} < \nu^K = n\, v_{\max}$. We shortly specify how to choose the grid points. In our mixed integer program, considering the adaptive variant in (7), letting S^* be an optimal solution to this problem, we guess the value of $V(S^*)$ to be one of the grid points. We find a solution that maximizes the expected revenue while staying consistent with the guess. We consider the decision variables $\boldsymbol{y} = (y_j : j \in N) \in \{0,1\}^n$, $\boldsymbol{x} = (x_{ij} : i, j \in N) \in [0,1]^{n^2}$ and $\boldsymbol{x}_0 = (x_{0j} : j \in N) \in [0,1]^n$, where we use the decision variables $\boldsymbol{y}$ to capture which products are included in an optimal solution to (7), whereas we use the decision variables $\boldsymbol{x}$ and $\boldsymbol{x}_0$ to compute the value of $\overline{\mathrm{F}}(S \setminus \{i\})$ for each $i \in N$. We consider the mixed integer program

$$Z^k = \max_{(\boldsymbol{y},\boldsymbol{x},\boldsymbol{x}_0)\in\{0,1\}^n\times[0,1]^{n^2+n}} \Bigg\{ \sum_{j\in N} \frac{v_j}{1+\nu^{k-1}} \Big(r_j\, y_j + p_j \sum_{i\in N} r_i\, x_{ij} \Big) \;:\; \sum_{j\in N} v_j\, y_j \le \nu^k,$$

$$\sum_{i\in N} x_{ij} + x_{0j} \le 1 \quad \forall\, j\in N, \quad \frac{x_{ij}}{v_i} \le x_{0j} \quad \forall\, i,j\in N,$$

$$x_{ij} \le \mathbf{1}(i\ne j)\, y_i \quad \forall\, i,j\in N, \quad x_{ij}\le y_j \quad \forall\, i,j\in N \Bigg\}. \tag{20}$$

In the problem above, the decision variable y_j takes value one if we include product j in an optimal solution to problem (7). Letting S^* be an optimal solution to problem (7), we use the decision variables $(x_{ij} : i \in N)$ and x_{0j} to compute the value of $\overline{\mathrm{F}}(S^* \setminus \{j\})$. In the objective function, we compute the expected revenue of the adaptive variant by using our guess of $V(S^*)$ and the linear program in (19) to compute $\overline{\mathrm{F}}(S^* \setminus \{j\})$. The first constraint ensures that our guess for $V(S^*)$ is consistent with the products included in an optimal solution. The second, third, and fourth constraints are from problem (19), which, along with the objective function, allow us to compute $\overline{\mathrm{F}}(S^* \setminus \{j\})$. In the fourth constraint, we use the fact that if the preferred option of a customer is j, then her replacement option cannot be product j. In the fifth constraint, we ensure that if we do not offer a particular product, then we do not compute the expected revenue from a customer choosing that product as her preferred option. We use the upper bound ν^k of the interval in the first constraint, whereas the lower bound ν^{k-1} of the interval in the objective function, so that problem (20) yields an upper bound on the optimal expected revenue in the adaptive variant.

In the next lemma, we show that we can use problem (20) to indeed get an upper bound on the optimal expected revenue no matter how we choose the grid points $\{\nu^k : k = 0, 1, \ldots, K\}$.

Lemma C.1 (Upper Bound for Adaptive Variant) *Noting that Z^*_{AD} and Z^k are, respectively, the optimal objective values of problems* (7) *and* (20)*, we have* $\max_{k=1,\ldots,K} Z^k \ge Z^*_{\mathsf{AD}}$.

Proof: Letting S^* be an optimal solution to the adaptive variant in (7), we choose the grid point κ such that $\nu^{\kappa-1} \le V(S^*) \le \nu^\kappa$. Let $\widehat{y}_j = \mathbf{1}(j \in S^*)$ for all $j \in N$. For each $j \in S^*$, let $(\widehat{x}_{ij} : i \in N)$ and $\widehat{x}_{0j}$ be an optimal solution to problem (19) when we solve this problem with $S = S^* \setminus \{j\}$. Lastly, for each $j \notin S^*$, let $\widehat{x}_{ij} = 0$ for all $i \in N$ and $\widehat{x}_{0j} = 0$. Letting $\widehat{\boldsymbol{x}} = (\widehat{x}_{ij} : i, j \in N)$ and $\widehat{\boldsymbol{x}}_0 = (\widehat{x}_{0j} : j \in N)$ we claim that the solution $(\widehat{\boldsymbol{y}}, \widehat{\boldsymbol{x}}, \widehat{\boldsymbol{x}}_0)$ is feasible to problem (20) with $k = \kappa$. By the choice of the grid point κ and the definition of $\widehat{y}_j$, we have $\sum_{j\in N} v_j\, \widehat{y}_j = \sum_{j\in S^*} v_j = V(S^*) \le \nu^\kappa$. Furthermore, for each $j \notin S^*$, we have $\widehat{x}_{ij} = 0 \le \widehat{y}_j$, whereas for each $j \in S^*$, we have $\widehat{x}_{ij} \le 1 = \mathbf{1}(j \in S^*) = \widehat{y}_j$. Thus, the solution $(\widehat{\boldsymbol{y}}, \widehat{\boldsymbol{x}}, \widehat{\boldsymbol{x}}_0)$ satisfies the first and fifth constraints in (20). For each $j \notin S^*$, because

$\widehat{x}_{ij}=0$ and $\widehat{x}_{0j}=0$, the left sides of the second and third constraints in (20) evaluate to zero, these constraints are automatically satisfied by the solution $(\widehat{\boldsymbol{y}},\widehat{\boldsymbol{x}},\widehat{\boldsymbol{x}}_0)$. On the other hand, for each $j\in S^*$, because $(\widehat{x}_{ij}:i\in N)$ and $\widehat{x}_{0j}$ are an optimal solution to problem (19) with $S=S^*\setminus\{j\}$, by the first and second constraints in this problem, the second and third constraints in (20) are satisfied by the solution $(\widehat{\boldsymbol{y}},\widehat{\boldsymbol{x}},\widehat{\boldsymbol{x}}_0)$ as well. Thus, the solution $(\widehat{\boldsymbol{y}},\widehat{\boldsymbol{x}},\widehat{\boldsymbol{x}}_0)$ satisfies the second and third constraints in (20). Lastly, for each $j\notin S^*$, the left side of the fourth constraint evaluates to zero, whereas for each $j\in S^*$, noting the third constraint in (19), along with the definition of $\widehat{y}_j$, we get $\widehat{x}_{ij}\le \mathbf{1}(i\in S^*\setminus\{j\})=\mathbf{1}(i\neq j,\ i\in S^*)=\mathbf{1}(i\neq j)\,\widehat{y}_i$. Thus, the solution $(\widehat{\boldsymbol{y}},\widehat{\boldsymbol{x}},\widehat{\boldsymbol{x}}_0)$ satisfies the fourth constraint in (20), so the claim follows. In this case, because the solution $(\widehat{\boldsymbol{y}},\widehat{\boldsymbol{x}},\widehat{\boldsymbol{x}}_0)$ is feasible to problem (20) with $k=\kappa$, we obtain

$$
\begin{aligned}
Z^\kappa &\ge \sum_{j\in N}\frac{v_j}{1+\nu^{\kappa-1}}\left(r_j\,\widehat{y}_j+p_j\sum_{i\in N}r_i\,\widehat{x}_{ij}\right)\overset{(a)}{=}\sum_{j\in N}\frac{v_j}{1+\nu^{\kappa-1}}\Big(r_j\,\mathbf{1}(j\in S^*)+p_j\,\mathbf{1}(j\in S^*)\,\overline{\mathrm{F}}(S^*\setminus\{j\})\Big)\\
&\overset{(b)}{\ge}\sum_{j\in N}\frac{v_j}{1+V(S^*)}\,\mathbf{1}(j\in S^*)\Big(r_j+p_j\,\overline{\mathrm{F}}(S^*\setminus\{j\})\Big)=\sum_{j\in S^*}\frac{v_j}{1+V(S^*)}\Big(r_j+p_j\,\overline{\mathrm{F}}(S^*\setminus\{j\})\Big)\overset{(c)}{=}Z^*_{\mathsf{AD}},
\end{aligned}
$$

where (a) holds because if $j\notin S^*$, then $\widehat{x}_{ij}=0$ for all $i\in N$, but if $j\in S^*$, then $(\widehat{x}_{ij}:i\in N)$ and $\widehat{x}_{0j}$ is optimal to (19) with $S=S^*\setminus\{j\}$, (b) holds because $V(S^*)\ge\nu^{\kappa-1}$ and (c) is by (7). ■

In the next lemma, we show that we can use the mixed integer program in (20) to get an approximate solution to the adaptive variant.

Lemma C.2 (Approximation to Adaptive Variant) *Fixing $\kappa^*=\arg\max_{k=1,\dots,K}Z^k$, using $(\boldsymbol{y}^*,\boldsymbol{x}^*,\boldsymbol{x}_0^*)$ to denote an optimal solution to problem (20) with $k=\kappa^*$, if we set $\widehat{S}=\{j\in N: y_j^*=1\}$, then $\widehat{S}$ is a $\left(\frac{\nu^{\kappa^*-1}}{\nu^{\kappa^*}}\right)$-approximate solution to the adaptive variant in (7).*

Proof: We make two observations. First, because the solution $(\boldsymbol{y}^*,\boldsymbol{x}^*,\boldsymbol{x}_0^*)$ is feasible to problem (20), by the fourth constraint in problem (20), if $j\in\widehat{S}$, then we have the chain of inequalities $x^*_{ij}\le\mathbf{1}(i\neq j)\,y^*_i=\mathbf{1}(i\neq j)\,\mathbf{1}(i\in\widehat{S})=\mathbf{1}(i\in\widehat{S}\setminus\{j\})$, where the first equality uses the definition of $\widehat{S}$. Furthermore, noting that the solution $(\boldsymbol{y}^*,\boldsymbol{x}^*,\boldsymbol{x}_0^*)$ satisfies the second and third constraints in problem (20), it follows that if $j\in\widehat{S}$, then the solution $(x^*_{ij}:i\in N)$ and x_{0j} satisfies all of the constraints in problem (19) when we solve this problem with $S=\widehat{S}\setminus\{j\}$. Therefore, we have $\overline{\mathrm{F}}(\widehat{S}\setminus\{j\})\ge\sum_{i\in N}r_i\,x^*_{ij}$ as long as $j\in\widehat{S}$. On the other hand, if $j\notin\widehat{S}$, then we have $y^*_j=0$ by the definition of $\widehat{S}$, in which case, by the last constraint in problem (20), we have $x^*_{ij}=0$ for all $i\in N$. Therefore, using the fact that $\overline{\mathrm{F}}(\widehat{S}\setminus\{j\})\ge\sum_{i\in N}r_i\,x^*_{ij}$ for all $j\in\widehat{S}$ as we just established, we get $\mathbf{1}(j\in\widehat{S})\,\overline{\mathrm{F}}(\widehat{S}\setminus\{j\})\ge\sum_{i\in N}r_i\,x^*_{ij}$ for all $j\in N$. Second, because the solution $(\boldsymbol{y}^*,\boldsymbol{x}^*,\boldsymbol{x}_0^*)$ satisfies the first constraint in problem (20), we get $V(\widehat{S}) = \sum_{j\in\widehat{S}} v_j = \sum_{j\in N} v_j\,\widehat{y}_j \le \nu^{\kappa^*}$. In this case, computing the objective function of problem (7) at the solution $\widehat{S}$, we obtain

$$\begin{aligned}
\sum_{j\in\widehat{S}} \frac{v_j}{1+V(\widehat{S})}\Big(r_j + p_j\,\overline{\mathrm{F}}(\widehat{S}\setminus\{j\})\Big) \;&=\; \sum_{j\in N} \frac{v_j}{1+V(\widehat{S})}\,\mathbf{1}(j\in\widehat{S})\Big(r_j + p_j\,\overline{\mathrm{F}}(\widehat{S}\setminus\{j\})\Big)\\
&\overset{(a)}{=} \sum_{j\in N} \frac{v_j}{1+V(\widehat{S})}\Big(r_j\,y_j^* + p_j\,\mathbf{1}(j\in\widehat{S})\,\overline{\mathrm{F}}(\widehat{S}\setminus\{j\})\Big) \overset{(b)}{\ge} \sum_{j\in N} \frac{v_j}{1+\nu^{\kappa^*}}\left(r_j\,y_j^* + p_j \sum_{i\in N} r_i\,x_{ij}^*\right)\\
&\ge \sum_{j\in N} \frac{v_j}{\frac{\nu^{\kappa^*}}{\nu^{\kappa^*-1}}\,(1+\nu^{\kappa^*-1})}\left(r_j\,y_j^* + p_j \sum_{i\in N} r_i\,x_{ij}^*\right) \overset{(c)}{=} \frac{\nu^{\kappa^*-1}}{\nu^{\kappa^*}} \max_{k=1,\dots,K} Z^k \overset{(d)}{\ge} \frac{\nu^{\kappa^*-1}}{\nu^{\kappa^*}}\,Z^*_{\mathsf{AD}},
\end{aligned}$$

where (a) uses the definition of $\widehat{S}$, (b) uses the two observations at the beginning of the proof, (c) uses the definitions of κ^* and $(\boldsymbol{y}^*, \boldsymbol{x}^*, \boldsymbol{x}_0^*)$ and (d) is by Lemma C.1. ∎

To use the lemma above to obtain an approximate solution to the adaptive variant, for fixed $\epsilon > 0$, we can choose the grid points as $\nu^k = v_{\min}\,(1+\epsilon)^k$ for $k = 0, 1, \dots, \lceil \log(\frac{n v_{\max}}{v_{\min}})/\log(1+\epsilon)\rceil$. In this case, we have $\frac{\nu^{k-1}}{\nu^k} = \frac{1}{1+\epsilon}$ for all $k = 1, \dots, K$. Noting that $\frac{1}{1+\epsilon} > 1-\epsilon$, by Lemma C.2, we obtain a $(1-\epsilon)$-approximate solution to the adaptive variant. The number of grid points is $O(\frac{\log(n\,v_{\max}/v_{\min})}{\log(1+\epsilon)}) = O(\frac{\log(n\,v_{\max}/v_{\min})}{\epsilon})$, so we need to solve $O(\frac{\log(n\,v_{\max}/v_{\min})}{\epsilon})$ mixed integer programs of the form in (20) to obtain a $(1-\epsilon)$-approximate solution to the adaptive variant. In Section 7, we use this approach with $\epsilon = 0.01$ to obtain near-optimal solutions to the adaptive variant. Letting $\widetilde{Z}^k$ be the optimal objective value of the linear programming relaxation of the mixed integer program in (20), we naturally have $\widetilde{Z}^k \ge Z^k$, so by Lemma C.1, we can use $\max_{k=1,\dots,K} \widetilde{Z}^k$ as an upper bound on the optimal objective value of the adaptive variant. In Section 8, we use $\epsilon = 0.001$, along with the linear programming relaxation of (20), to obtain an upper bound on the optimal objective value of the adaptive variant. Because we do not solve mixed integer programs to obtain the upper bound, we can afford choosing a significantly smaller value for ϵ.

Mixed Integer Program for the Non-Adaptive Variant:

Our approach for formulating a mixed integer program for the non-adaptive variant is similar to that for formulating a mixed integer program for the adaptive variant, but the overall approach is a bit simpler. Using S^* to denote an optimal solution to the non-adaptive variant in (11), we guess the value of $V(S^*)$, as well as the product with the largest preference weight in the optimal solution. In this case, we solve an integer program to find an assortment of products to offer to maximize the expected revenue while ensuring that the assortment that we offer is consistent with the guesses. To implement this approach, letting $v_{\min} = \min_{i\in N} v_i$ and $v_{\max} = \max_{i\in N} v_i$, we divide the interval $[v_{\min}, n\,v_{\max}]$ into K subintervals by using the grid points $v_{\min} = \nu^0 < \nu^1 < \ldots < \nu^{K-1} < \nu^K = n\,v_{\max}$. We consider the decision variables $\boldsymbol{y} = (y_j : j \in N) \in \{0,1\}^n$ and $\boldsymbol{x} = (x_{ij} : i, j \in N) \in \{0,1\}^{n^2}$, where y_j takes value one if and only if we offer product j, whereas x_{ij} takes value one if and only if we offer both products i and j with $i \neq j$. In this case, guessing the product with the largest preference weight as ℓ and the value of $V(S^* \setminus \{\ell\})$ as the grid point ν^k, we consider the problem

$$Z_\ell^k = \max_{(\boldsymbol{y},\boldsymbol{x}) \in \{0,1\}^{n+n^2}} \left\{ \sum_{j \in N} \frac{v_j}{1+v_\ell+\nu^{k-1}} \left(r_j\, y_j + p_j \frac{\sum_{i \in N} r_i\, v_i\, x_{ij}}{1+v_\ell+\nu^{k-1}-v_j} \right) \; : \; \sum_{j \in N} v_j\, y_j \leq v_\ell + \nu^k, \right.$$

$$y_j \leq \mathbf{1}(v_j \leq v_\ell) \quad \forall\, j \in N, \quad y_\ell = 1,$$

$$\left. x_{ij} \leq \mathbf{1}(i \neq j)\, y_i \quad \forall\, i,j \in N, \quad x_{ij} \leq \mathbf{1}(i \neq j)\, y_j \quad \forall\, i,j \in N \right\}. \qquad (21)$$

Noting the objective function in (11), the objective function above computes the expected revenue from a customer. The first constraint ensures that the products that we offer are consistent with the guess of $V(S^*)$. The second constraint ensures that product ℓ is the one with the largest preference weight. The third constraint ensures that product ℓ is offered. Noting that the objective function coefficient of the decision variable x_{ij} is positive, the fourth and fifth constraints ensure that x_{ij} takes value one if and only if both products i and j with $i \neq j$ are offered. Similar to our approach in (20), we use the upper bound ν^k of the interval in the first constraint, whereas the lower bound ν^{k-1} of the interval in the objective function, which will allow us to use problem (21) to obtain an upper bound on the optimal expected revenue. In the next lemma, we show that we can indeed use problem (21) to obtain an upper bound on the optimal expected revenue. Recall that we use $\mathcal{R}_{\mathsf{NA}}(S)$ to denote the objective function of problem (11). We set $a \vee b = \max\{a, b\}$.

Lemma C.3 (Upper Bound for Non-Adaptive Variant) *Noting that Z^*_{NA} and Z^k_ℓ are, respectively, the optimal objective values of problems* (11) *and* (21)*, we have*

$$\left(\max_{k=1,\dots,K,\; \ell \in N} Z_\ell^k \right) \vee \left(\max_{\ell \in N} \mathcal{R}_{\mathsf{NA}}(\{\ell\}) \right) \;\geq\; Z^*_{\mathsf{NA}}.$$

Proof: Let S^* be an optimal solution to (11). If there is only one product in S^*, then the result immediately follows by noting the second term in the maximum operator on the left side of the inequality in the lemma. Therefore, we proceed with the understanding that there are two or more products in S^*. Let λ be the product with the largest preference weight in S^* and choose the grid point κ such that $\nu^{\kappa-1} \leq V(S^* \setminus \{\lambda\}) \leq \nu^\kappa$. Because there is at least one product in $S^* \setminus \{\lambda\}$, there exists such a grid point. We construct a solution $(\widehat{\boldsymbol{y}}, \widehat{\boldsymbol{x}})$ to problem (21) as follows. We set $\widehat{y}_j = \mathbf{1}(j \in S^*)$ for all $j \in N$ and $\widehat{x}_{ij} = \mathbf{1}(i, j \in S^*,\ i \neq j)$. We claim that this solution is feasible to problem (21) when we solve this problem with $k = \kappa$ and $\ell = \lambda$. Because product λ is the one with the largest preference weight in S^*, we have $\lambda \in S^*$, so $V(S^*) = v_\lambda + V(S^* \setminus \{\lambda\}) \leq v_\lambda + \nu^k$, so $\sum_{j \in N} v_j\, \widehat{y}_j = \sum_{j \in S^*} v_j \leq v_\lambda + \nu^\kappa$. Thus, the first constraint is satisfied. Because product λ is the one with the largest preference weight in S^*, the second and third constraints are satisfied. Finally, the fourth and fifth constraints are immediately satisfied by the definitions of $\widehat{y}_j$ and $\widehat{x}_{ij}$. Thus, the claim follows. In this case, we obtain the chain of inequalities

$$\begin{aligned}
Z^\kappa_\lambda \;\ge\; & \sum_{j\in N} \frac{v_j}{1+v_\lambda+\nu^{\kappa-1}}\left(r_j\,\widehat{y}_j + p_j\,\frac{\sum_{i\in N} v_i\,r_i\,\widehat{x}_{ij}}{1+v_\lambda+\nu^{\kappa-1}-v_j}\right)\\
&\overset{(a)}{=} \sum_{j\in N} \frac{v_j}{1+v_\lambda+\nu^{k-1}}\mathbf{1}(j\in S^*)\left(r_j + p_j\,\frac{\sum_{i\in N} v_i\,r_i\,\mathbf{1}(i\in S^*,\ i\neq j)}{1+v_\lambda+\nu^{\kappa-1}-v_j}\right)\\
&= \sum_{j\in S^*} \frac{v_j}{1+v_\lambda+\nu^{\kappa-1}}\left(r_j + p_j\,\frac{\sum_{i\in S^*\setminus\{j\}} v_i\,r_i}{1+v_\lambda+\nu^{k-1}-v_j}\right)\\
&\overset{(b)}{\ge} \sum_{j\in S^*} \frac{v_j}{1+V(S^*)}\left(r_j + p_j\,\frac{\sum_{i\in S^*\setminus\{j\}} v_i\,r_i}{1+V(S^*\setminus\{j\})}\right) \overset{(c)}{=} Z^*_{\mathsf{NA}},
\end{aligned}$$

where (a) holds by the definitions of $\widehat{y}_j$ and $\widehat{x}_{ij}$, (b) holds because $V(S^*\setminus\{\lambda\})\ge \nu^{\kappa-1}$ and $\lambda\in S^*$ and (c) follows by noting the objective function of problem (11). ■

In the next lemma, we give a result that will be useful to show that we can use problem (21) to get an approximate solution to the non-adaptive variant.

Lemma C.4 (Approximation to Non-Adaptive Variant) *Letting $(\boldsymbol{y}^*,\boldsymbol{x}^*)$ be an optimal solution to problem* (21)*, if we set $\widehat{S}=\{j\in N: y^*_j=1\}$, then the expected revenue provided by the solution $\widehat{S}$ for the non-adaptive variant is at least $(\frac{\nu^{k-1}}{\nu^k})^2\,Z^k_\ell$.*

Proof: Because the objective function coefficient of x_{ij} in (21) is positive, by the last two constraints, $x^*_{ij}=\mathbf{1}(i\neq j)\,y^*_i\,y^*_j$, so $y^*_j=\mathbf{1}(j\in\widehat{S})$ and $x^*_{ij}=\mathbf{1}(i\in\widehat{S},\ j\in\widehat{S},\ i\neq j)$. Thus, we get

$$\begin{aligned}
&\sum_{j\in\widehat{S}} \frac{v_j}{1+V(\widehat{S})}\left(r_j + p_j\frac{\sum_{i\in\widehat{S}\setminus\{j\}} v_i\,r_i}{1+V(\widehat{S}\setminus\{j\})}\right)\\
&\quad= \sum_{j\in N} \frac{v_j}{1+V(\widehat{S})}\,\mathbf{1}(j\in\widehat{S})\left(r_j + p_j\frac{\sum_{i\in N} v_i\,r_i\,\mathbf{1}(i\in\widehat{S},\ i\neq j)}{1+V(\widehat{S}\setminus\{j\})}\right)\\
&\quad\overset{(a)}{\ge} \sum_{j\in N} \frac{v_j}{1+v_\ell+\nu^k}\,\mathbf{1}(j\in\widehat{S})\left(r_j + p_j\frac{\sum_{i\in N} v_i\,r_i\,\mathbf{1}(i\in\widehat{S},\ i\neq j)}{1+v_\ell+\nu^k-v_j}\right)\\
&\quad\overset{(b)}{=} \sum_{j\in N} \frac{v_j}{1+v_\ell+\nu^k}\left(r_j\,y^*_j + p_j\frac{\sum_{i\in N} v_i\,r_i\,x^*_{ij}}{1+v_\ell+\nu^k-v_j}\right)\\
&\quad\overset{(c)}{\ge} \sum_{j\in N} \frac{v_j}{\frac{\nu^k}{\nu^{k-1}}(1+v_\ell+\nu^{k-1})}\left(r_j\,y^*_j + p_j\frac{\sum_{i\in N} v_i\,r_i\,x^*_{ij}}{\frac{\nu^k}{\nu^{k-1}}(1+v_\ell+\nu^{k-1}-v_j)}\right) \overset{(d)}{\ge} \left(\frac{\nu^{k-1}}{\nu^k}\right)^2 Z^k_\ell,
\end{aligned}$$

where (a) hold because $V(\widehat{S})=\sum_{j\in N} v_j\,\widehat{y}_j\le v_\ell+\nu^k$ by the first constraint in (21), (b) holds by the discussion at the beginning of the proof, (c) holds because if $x^*_{ij}=1$, then we must have $y^*_j=1$ by the last two constraints in (21), in which case, we must have $v_j \leq v_\ell$ by the second constraint in (21) and (d) holds by the objective function of problem (21). ■

Following the same approach that we followed for the adaptive variant, we can choose $O(\frac{\log(n\, v_{\max}/v_{\min})}{\epsilon})$ grid points such that $\frac{\nu^{k-1}}{\nu^k} = \frac{1}{1+\epsilon}$. In this case, for each $k = 1, \ldots, K$ and $\ell \in N$, using $\widehat{S}_\ell^k$ to denote the solution constructed in Lemma C.4, recalling that $\mathcal{R}_{\mathsf{NA}}(S)$ is the objective function of the non-adaptive variant in (11), we have $\mathcal{R}_{\mathsf{NA}}(\widehat{S}_\ell^k) \geq \frac{1}{(1+\epsilon)^2}\, Z_\ell^k \geq (1-\epsilon)^2\, Z_\ell^k \geq (1-2\epsilon)\; Z_\ell^k$ for all $k = 1, \ldots, K$ and $\ell \in N$. In addition to the solutions constructed in Lemma C.4, considering the solutions of the form $\{\ell\}$ for $\ell \in N$, we obtain $\max_{k=1,\ldots,K,\ \ell\in N} \mathcal{R}_{\mathsf{NA}}(\widehat{S}_\ell^k) \vee \max_{\ell\in N} \mathcal{R}_{\mathsf{NA}}(\{\ell\}) \geq (1-2\epsilon)\big(\max_{k=1,\ldots,K,\ \ell\in N} Z_\ell^k \vee \max_{\ell\in N} \mathcal{R}_{\mathsf{NA}}(\{\ell\})\big) \geq (1-2\epsilon)\, Z_{\mathsf{NA}}^*$, where the last inequality uses Lemma C.3. Therefore, if we check the expected revenue provided by the solutions $\widehat{S}_\ell^k$ for all $k = 1, \ldots, K$ and $\ell \in N$, as well as the solutions $\{\ell\}$ for all $\ell \in N$, then the best of these solutions provides a $(1-2\,\epsilon)$-approximate solution to the non-adaptive variant. To obtain this approximate solution, noting that we have $O(\frac{\log(n\, v_{\max}/v_{\min})}{\epsilon})$ grid points and n products, we need to solve $O(n \frac{\log(n\, v_{\max}/v_{\min})}{\epsilon})$ problems of the form in (21).

Appendix D: Tightening the Surrogate Problem

We give a slightly tighter version of the surrogate problem in (8). This surrogate problem is less interpretable than the one in (8). In particular, the tighter version of the surrogate problem involves a constraint. Nevertheless, if we continue defining the set M_{H} as the set of $\lceil 1/\epsilon \rceil$ products in S_{H}^* with the largest weights, then this surrogate problem approximates the expected revenue from below within a factor of $\frac{2+\epsilon}{2+2\epsilon}$. By Lemma 4.3, the surrogate problem in (8) approximates the expected revenue from below within a factor of $\frac{1}{1+\epsilon}$. We have $\frac{2+\epsilon}{2+2\epsilon} - \frac{1}{1+\epsilon} = \frac{\epsilon}{2+2\epsilon}$, so the surrogate problem that we give in this section is slightly tighter than the one in (8), but the advantage of the tighter surrogate disappears as ϵ gets smaller. When ϵ is large so that we guess a small number of products with the largest weights, the tighter surrogate problem may be useful from a computational viewpoint. We use the tighter surrogate in Section 8. We define the quantity z^* and the sets M_{H}, N_{H} and N_{L} exactly as in Section 4. For the tighter surrogate problem, we define the objective function as

$$\mathcal{R}_{\mathsf{SR}}(S) \;=\; \sum_{i\in M_{\mathsf{H}}} \frac{v_i}{1+V(S)} \Big(r_i + p_i\, \overline{\mathrm{F}}(S\setminus\{i\})\Big) + \sum_{i\in S\setminus M_{\mathsf{H}}} \frac{v_i}{1+V(S)}\,(r_i + p_i\, z^*). \tag{22}$$

Our surrogate problem is given by $Z_{\mathsf{SR}}^* = \max_{S\subseteq N}\big\{\mathcal{R}_{\mathsf{SR}}(S) : M_{\mathsf{H}} \subseteq S \subseteq N_{\mathsf{H}} \cup N_{\mathsf{L}},\ \overline{\mathrm{R}}(S\cap N_{\mathsf{H}}) \geq z^*\big\}$. Throughout this section, we refer to this problem as the surrogate problem.

We can follow the same reasoning in the proof of Lemma 4.2 to show that the optimal objective value of the surrogate problem in (22) is an upper bound on the optimal expected revenue in the adaptive variant. This result is the analogue of Lemma 4.2 for the tighter version of the surrogate problem. In the next lemma, we give an analogue of Lemma 4.3, where we lower bound the objective function of the adaptive variant by using the objective function of the surrogate problem.

Lemma D.1 (Lower Bound from Surrogate) *Letting $\mathcal{R}_{\mathsf{AD}}(S)$ and $\mathcal{R}_{\mathsf{SR}}(S)$, respectively, be the objective functions of the adaptive variant in* (7) *and the surrogate problem in* (22)*, for any $S \subseteq N$ that satisfies $M_{\mathsf{H}} \subseteq S \subseteq N_{\mathsf{H}} \cup N_{\mathsf{L}}$ and $\overline{\mathrm{R}}(S \cap N_{\mathsf{H}}) \geq z^*$, we have $\mathcal{R}_{\mathsf{AD}}(S) \geq \frac{2+\epsilon}{2+2\epsilon}\, \mathcal{R}_{\mathsf{SR}}(S)$.*

Proof: Noting that $S \subseteq N_{\mathsf{H}} \cup N_{\mathsf{L}}$, we partition $S \setminus M_{\mathsf{H}}$ into the two sets $(S \setminus M_{\mathsf{H}}) \cap N_{\mathsf{H}}$ and $(S \setminus M_{\mathsf{H}}) \cap N_{\mathsf{L}}$. We will lower bound $r_i + p_i\, \overline{\mathrm{F}}(S \setminus \{i\})$ by considering two cases. First, consider the case $i \in (S \setminus M_{\mathsf{H}}) \cap N_{\mathsf{L}}$. In this case, because $i \in N_{\mathsf{L}}$ and $N_{\mathsf{H}} \cap N_{\mathsf{L}} = \varnothing$, we have $S \cap N_{\mathsf{H}} \subseteq S \setminus \{i\}$, so we get $\overline{\mathrm{F}}(S \setminus \{i\}) = \max_{Q \subseteq S \setminus \{i\}} \overline{\mathrm{R}}(Q) \geq \overline{\mathrm{R}}(S \cap N_{\mathsf{H}}) \geq z^*$, where the first inequality holds because $S \cap N_{\mathsf{H}}$ is a feasible solution to the maximization problem on the left side of the inequality and the second inequality holds by the assumption in the lemma. Therefore, noting that $\frac{2+\epsilon}{2+2\epsilon} \leq 1$, we have $r_i + p_i\, \overline{\mathrm{F}}(S \setminus \{i\}) \geq \frac{2+\epsilon}{2+2\epsilon}(r_i + p_i\, z^*)$. Second, consider the case $i \in (S \setminus M_{\mathsf{H}}) \cap N_{\mathsf{H}}$. Letting $\kappa = \lceil 1/\epsilon \rceil$ for notational brevity, using precisely the same approach in the proof of Lemma 4.3, we have $\sum_{j \in S \cap N_{\mathsf{H}}} v_j\, r_j - v_i\, r_i \geq \frac{\kappa}{1+\kappa} \sum_{j \in S \cap N_{\mathsf{H}}} v_j\, r_j$. Also, by the definition of N_{H}, we have $r_j \geq z^*$ for all $j \in N_{\mathsf{H}}$, in which case, for all $j \in S \cap N_{\mathsf{H}}$, we get $r_j \geq \frac{1}{2} r_j + \frac{1}{2} p_j\, z^*$. Lastly, noting that $\kappa \geq 1/\epsilon$ and $(1+2x)/(2+2x)$ is increasing in x, we have $\frac{1+2\kappa}{2+2\,\kappa} \geq \frac{1+2/\epsilon}{2+2/\epsilon}$. In this case, we get

$$
\begin{aligned}
r_i + p_i\, \overline{\mathrm{F}}(S \setminus \{i\}) \;&=\; r_i + p_i \max_{Q \subseteq S \setminus \{i\}} \overline{\mathrm{R}}(Q) \;\overset{(a)}{\geq}\; r_i + p_i\, \overline{\mathrm{R}}((S \cap N_{\mathsf{H}}) \setminus \{i\}) \\
&=\; r_i + p_i \frac{\sum_{j \in S \cap N_{\mathsf{H}}} v_j\, r_j - v_i\, r_i}{1 + V(S \cap N_{\mathsf{H}}) - v_i} \;\overset{(b)}{\geq}\; r_i + \frac{\kappa}{1+\kappa}\, p_i \frac{\sum_{j \in S \cap N_{\mathsf{H}}} v_j\, r_j}{1 + V(S \cap N_{\mathsf{H}})} \;\overset{(c)}{\geq}\; r_i + \frac{\kappa}{1+\kappa}\, p_i\, z^* \\
&=\; \frac{\kappa}{1+\kappa}(r_i + p_i\, z^*) + \frac{1}{1+\kappa} r_i \;\overset{(d)}{\geq}\; \frac{\kappa}{1+\kappa}(r_i + p_i\, z^*) + \frac{1}{2+2\kappa}(r_i + p_i\, z^*) \\
&=\; \frac{1+2\kappa}{2+2\,\kappa}(r_i + p_i\, z^*) \;\geq\; \frac{1+2/\epsilon}{2+2/\epsilon}(r_i + p_i\, z^*) \;=\; \frac{2+\epsilon}{2+2\epsilon}(r_i + p_i\, z^*),
\end{aligned}
$$

where (a) uses $(S \cap N_{\mathsf{H}}) \setminus \{i\} \subseteq S \setminus \{i\}$, (b) is by $\sum_{j \in S \cap N_{\mathsf{H}}} v_j\, r_j - v_i\, r_i \geq \frac{\kappa}{1+\kappa} \sum_{j \in S \cap N_{\mathsf{H}}} v_j\, r_j$, (c) holds by the assumption in the lemma and (d) holds because $r_j \geq \frac{1}{2} r_j + \frac{1}{2} p_j\, z^*$ for all $j \in S \cap N_{\mathsf{H}}$.

By the two cases, we have $r_i + p_i\, \overline{\mathrm{F}}(S \setminus \{i\}) \geq \frac{2+\epsilon}{2+2\epsilon}(r_i + p_i\, z^*)$ for all $i \in S \setminus M_{\mathsf{H}}$. The result holds because if we replace $r_i + p_i\, \overline{\mathrm{F}}(S \setminus \{i\})$ in (7) with $r_i + p_i\, z^*$ for all $i \in S \setminus M_{\mathsf{H}}$, then we get (22). ■

We can use a multi-dimensional knapsack problem similar to the one in Section 5 to obtain an approximate solution to the tighter version of the surrogate problem with the objective function in (22) while satisfying the constraint $\overline{\mathrm{R}}(S \cap N_{\mathsf{L}}) \geq \frac{1}{1+\epsilon}\, z^*$. The proof of Lemma D.1 goes through with no modifications when the set S satisfies $\overline{\mathrm{R}}(S \cap N_{\mathsf{H}}) \geq \frac{1}{1+\epsilon}\, z^*$, but we obtain $\mathcal{R}_{\mathsf{AD}}(S) \geq \frac{2+\epsilon}{2(1+\epsilon)^2}\, \mathcal{R}_{\mathsf{SR}}(S)$. In this case, we can build on this result, along with super-optimal solutions to the multi-dimensional knapsack problem, to construct an approximate solution to the adaptive variant.