Should Only Popular Products Be Stocked? Warehouse Assortment Selection for E-Commerce Companies

Xiaobo Li
Department of Industrial Systems Engineering and Management, National University of Singapore, iselix@nus.edu.sg
Hongyuan Lin
Faculty of Business in SciTech and School of Management, University of Science and Technology of China, hylin@ustc.edu.cn
Fang Liu
Durham University Business School, fang.liu@durham.ac.uk

Problem definition: This paper studies the single-warehouse assortment selection problem that aims to minimize the order fulfillment cost under the cardinality constraint. We propose two fulfillment-related cost functions corresponding to spillover fulfillment and order-splitting, respectively. This problem includes the fill rate maximization problem as a special case. We show that although the objective function is submodular for a broad class of cost functions, the fill rate maximization problem with the largest order size being two is NP-hard. **Methodology**: To make the problem tractable to solve, we formulate the general warehouse assortment problem under the two types of cost functions as mixed integer linear programs (MILPs). We also provide a dynamic programming algorithm to solve the problem in polynomial time if orders are non-overlapping. Furthermore, we propose a simple heuristic called the marginal choice indexing (MCI) policy that allows the warehouse to store the most popular products. This policy is easy to compute and hence is scalable to large-size problems. Although the performance of MCI can be arbitrarily bad in some extreme scenarios, we find a general condition under which it is optimal. This condition is satisfied by many multi-purchase choice models. **Managerial implications**: Through extensive numerical experiments on a real-world dataset from RiRiShun Logistics, we find that the MCI policy is surprisingly near-optimal in all the settings we tested. Simply applying the MCI policy, the fill rate is estimated to improve by 9.18% on average compared to the current practice for the local transfer centers (LTCs) on the training data set. More surprisingly, the MCI policy outperforms the MILP optimal solution in 14 out of 25 cases on the test data set, illustrating its robustness against demand fluctuations.

Key words: warehouse assortment selection, demand choice models, submodular functions, marginal choice probability

1. Introduction

The growth of e-commerce has brought out the need for responsive delivery (Aryapadi et al. 2020). Many online retailers, such as Amazon and Alibaba, offer same-day or even two-hour delivery services. To achieve this, e-commerce companies often operate warehouses in urban areas to be closer to customers, which makes selecting assortments to stock in these warehouses challenging. Logistics companies (e.g., RiRiShun Logistics) that handle bulky and heavy items, such as furniture and appliances, also face similar challenges. If the products stored in such local warehouses are not carefully selected, significant transshipment and order fulfillment costs could drastically harm the company's profit.

The primary goal of a local warehouse is to select its product assortment to fulfill customer orders at the lowest possible fulfillment cost. Unlike assortment planning problems where companies maximize revenue by choosing an optimal product assortment, in warehouse assortment selection problems, a set of products is selected to store at a local warehouse to minimize fulfillment costs based on the customer demand distribution. Specifically, when a customer places an order, the cost of fulfilling this order is the least when it can be fully satisfied by the target local warehouse. However, if this local warehouse cannot fully cover the customer's order, the order is either fulfilled by a back-end warehouse through spillover fulfillment or split into several suborders fulfilled by other local warehouses. In either case, substantial additional shipment and operational costs may occur, resulting in increased fulfillment costs.

Warehouse assortment selection has become more challenging as e-commerce companies encourage customers to place larger orders by offering promotions such as reduced delivery fees and discounts for orders that exceed a certain amount. This practice significantly increases the order size, resulting in a higher probability that an order cannot be fulfilled by the local warehouse. In an ideal scenario, if the local warehouse has sufficient capacity, then it would store all of the SKUs and fulfill all the orders sent to this warehouse resulting in the most cost-efficient scenario. However, storing a wide variety of SKUs adds complexity to operations, leading to a reduction in

operational efficiency and an increase in both storage and retrieval costs (Wan and Dresner 2015). Due to space constraints and high handling costs, local warehouses usually have tight constraints on the number of SKUs they can store.

The primary focus of this work is on selecting the optimal warehouse assortment subject to the SKU capacity constraint. On the one hand, an increasing number of e-commerce companies (e.g., Alibaba, Shipbob) have begun to pay attention to the SKU capacity of individual warehouses within their distribution networks, aiming to enhance the logistical efficiency and improve the customer experience (Lopienski 2021, Alfaro and Corbett 2003). On the other hand, managing a large number of SKUs may incur significant operational and handling costs, whereas focusing on a limited number of SKUs simplifies operational complexity and reduces both handling and retrieval costs. Furthermore, the SKU capacity constraint is widely used in warehouse planning literature, such as Catalán and Fisher (2012), Wu et al. (2019), Zhu et al. (2021).

In this paper, we take the initiative in investigating the single-warehouse assortment selection problem, subject to the same SKU capacity constraint as in the literature, i.e., the cardinality constraint. The objective is to minimize the total fulfillment cost, equivalent to minimizing the expected additional cost incurred when a local warehouse cannot fully fulfill its orders. We propose two types of order-related additional fulfillment costs, reflecting spillover and order-splitting costs. Specifically, we consider the spillover cost as a function of the complete set of ordered items, if they cannot be fully fulfilled by the local warehouse, and the order-splitting cost as a function of the subset of ordered items not included in the assortment of the local warehouse. In particular, the order fill rate maximization problem is a special case of the warehouse assortment problem under either type of cost function. The contributions of our paper are as follows.

First, we show that the objective function is submodular for a broad class of cost functions, irrespective of the demand distribution considered. This result is absent from the literature, even for the order fill rate maximization problem. Hence, techniques used in minimizing submodular functions can be applied here. This result bridges the warehouse assortment problem with the extensive literature on submodular function minimization.

Second, we show that the order fill rate maximization problem is NP-hard even when the largest order size is two, and the greedy policy can perform arbitrarily badly. To our knowledge, these are the first theoretical results on the computational tractability of the single-warehouse assortment selection problem. We then formulate the warehouse assortment problem under both types of cost functions as two mixed integer linear programming (MILP) problems. Additionally, we provide a dynamic programming algorithm to exactly solve the warehouse assortment problem with type-I cost functions in polynomial time if the orders are non-overlapping.

Third, we propose a simple heuristic called the marginal choice indexing (MCI) policy to solve the warehouse assortment problem. The MCI policy selects the K products with the highest marginal choice probability to store at the warehouse. Although the MCI policy may perform arbitrarily badly, we identify a sufficient condition on the customers' demand distribution for the MCI policy to be optimal. This condition holds for the classic discrete choice model, independent choice model, and some recently proposed multi-purchase choice models in Tulabandhula et al. (2023), Bai et al. (2023), and Lin et al. (2022). To our knowledge, the MCI policy is the only approach in the literature that guarantees optimality under some non-trivial conditions.

Finally, we conduct extensive numerical experiments on a real-world logistic dataset from RiRiShun Logistics. We find that the proposed MILP formulations are tractable in a practical size, allowing us to solve the optimal solution and analyze the MCI policy. Additionally, the MCI policy is near-optimal in all the settings we tested. We also identify a performance index, partial fulfillment rate (PFR), which can be used to explain the differences between the MCI policy and the MILP optimal solution. Simply applying the MCI policy, it is estimated that we can improve the order fill rate by 9.18% on average compared to the current practice for the local transfer centers on the training data set. Surprisingly, the MCI policy outperforms the MILP optimal solution in 14 out of 25 cases on the test data set. This implies that the MCI policy is robust with respect to demand changes.

The rest of the paper is organized as follows. In Section 2, we provide a comprehensive literature review. In Section 3, we set up the model under two types of cost functions, and show that the

objective function is submodular under mild conditions. In Section 4, we establish the NP-hardness of the warehouse assortment selection problem and formulate two MILPs to address the problem under different cost functions. In addition, we introduce a dynamic programming method for the case where orders are non-overlapping. In Section 5, we introduce the MCI policy and discuss conditions under which this policy is optimal. In Section 6, we conduct a case study on the real-world data from RiRiShun Logistics. In Section 7, we conclude this work and provide some future research directions.

2. Literature Review

2.1. Assortment Optimization

This paper lands in the area of assortment optimization. In what follows, we first explain the difference between our problem and the well-studied assortment planning problems. Then, we discuss papers related to warehouse assortment selection.

Assortment Planning and Choice Models. In the revenue management literature, the problem of assortment planning has been widely studied since its introduction in the seminal work of Ryzin and Mahajan (1999). The modern development of revenue management heavily relies on modeling demand using discrete choice models (see, e.g., Gallego and Topaloglu 2019 for a complete review). Instead of independently modeling the demand for each product, the discrete choice model assumes that each customer picks his or her most preferred product among the set of products offered or leaves without a purchase. Discrete choice models capture customers' substitution patterns when the offered set or product prices change. However, these models are unsuitable for our warehouse assortment selection problem because they assume each customer chooses one product at most, which is unlikely in practice.

Some recent papers on the choice model also allow customers to purchase multiple products in a single order. These models include the well-known multivariate logit model (e.g., Cox 1972) and its variants, multivariate MNL model (e.g., Russell and Petersen 2000), bundle multivariate logit model (e.g., Tulabandhula et al. 2023), multiple discrete-continuous extreme value (MDCEV)

model (e.g., Bhat 2005), threshold utility model (TUM) (Gallego and Wang 2019), and multi-purchase random utility model (Bai et al. 2023, Lin et al. 2022). Among these choice models, the MDCEV model and its variants (e.g., the multiple discrete continuous (MDC) choice model proposed by Huh and Li 2022) allow the purchase of multiple units of the same item and the TUM is a demand model based on aggregate level demand. These demand models are beyond the scope of this paper. For the rest of the aforementioned demand models, we discuss the corresponding warehouse assortment and provide sufficient conditions under which MCI is optimal.

The main difference between the assortment planning problem and the warehouse assortment problem is that the former aims to maximize revenue by offering an appropriate assortment to customers, while the latter aims to minimize cost by selecting a warehouse assortment to cover as many customers' orders as possible. In assortment planning problems, the customers' substitution behavior is taken into account, and no penalty is incurred if not all customer's product needs are met. However, in warehouse assortment problems, a penalty is incurred if a customer's order is not fully fulfilled by the warehouse. Additionally, when customers make choices, their decisions are not affected by the assortment of the warehouse.

Warehouse Assortment Selection. Compared to assortment planning in revenue management, studies on warehouse assortment problems are scarce. Among them, most papers (e.g., Catalán and Fisher 2012, Zhu et al. 2021, Söylemez 2021) consider the multi-warehouse assortment allocation problem that aims to minimize the number of split orders. They formulate the problem as mixed integer programs and study the properties of the corresponding linear programming relaxations. They also propose several heuristics and numerically test their performance. On the theoretical side, Catalán and Fisher (2012) show that the problem is NP-hard for the two-location problem.

Söylemez (2021) briefly mention the case where there is only one warehouse. The corresponding formulation aligns with the order fill rate maximization problem, which is a specific case studied in our paper. The single-warehouse order fill rate maximization is also studied in Wu et al. (2019). They propose a model combining exponential smoothing and community detection to predict

future demand. Then, based on the estimated demand function, they propose a robust optimization formulation to deal with the demand uncertainty and solve it using a heuristic. To the best of our knowledge, no literature studies the theoretical properties of the order fill rate maximization problem. In contrast, we systematically study a more general version of the warehouse assortment problem. We obtain a series of theoretical properties, including submodularity and NP-harness. We also propose a simple heuristic and identify conditions under which it is optimal.

The order fill rate problem is widely studied in the inventory management literature. It is an important service measure in the industry that is first applied in Song (1998) to the inventory system. They consider the continuous-review multi-item inventory system and define the order fill rate as the probability that a complete order can be satisfied within a time window. Subsequent research works on this topic include Song and Yao (2002), Lu et al. (2003), Lu and Song (2005), etc. For these problems, the assortment is fixed, and the main decision is the inventory control policy. Instead, we focus on the assortment decision without considering the inventory constraints.

2.2. Submodular Function Optimization

Both submodular function maximization and minimization problems have been extensively studied in the literature. In general, even unconstrained submodular function maximization is NP-hard to solve (e.g., Nemhauser and Wolsey 1978). For monotone submodular function maximization problems under the cardinality constraint, Nemhauser et al. (1978) have demonstrated that the greedy algorithm has the $1 - 1/e$ performance guarantee. This guarantee is shown to be the best possible in Nemhauser and Wolsey (1978) for any algorithms that query the objective function at a polynomial number of sets. Additionally, various special cases of this problem have been proven to be NP-hard, including weighted coverage (Feige 1998) or mutual information (Krause and Guestrin 2012).

On the contrary, the unconstrained submodular function minimization problem is possible to be solved in polynomial time using the Lovász extension (e.g., Lovász 1983, Schrijver 2003). Additionally, several constant approximation bounds are available for various variants, as demonstrated

by Nemhauser et al. (1978), Lee et al. (2009), and Feige et al. (2011). However, for the monotone submodularity function minimization with cardinality lower bound (SMCL) (e.g., Svitkina and Fleischer 2011), it is NP-hard to obtain a constant approximation ratio. More precisely, Svitkina and Fleischer (2011) show that it is impossible to obtain a solution that is ρ-approximation to the objective and σ-feasible if $\frac{\rho}{\sigma} = o\left(\sqrt{\frac{n}{\log n}}\right)$ for (SMCL). The $O\left(\sqrt{\frac{n}{\log n}}\right)$-approximation can be achieved using their algorithm, matching the impossibility result. Other algorithms for solving (SMCL) problems are presented in subsequent papers, e.g., Nagano et al. (2011), Iyer and Bilmes (2013), Goemans et al. (2009). As will be shown in Section 3.1, our objective function is monotone and submodular under some mild conditions. Therefore, under certain conditions, our problem can be viewed as a special class of (SMCL). The solution approaches and insights developed for our problem may shed light on similar submodular function minimization problems.

3. Model Setting

In this section, we formally define our general problem and several variants. Consider N distinct products indexed $1, 2, \ldots, N$. Denote $\mathcal{N} = \{1, 2, \ldots, N\}$ as the universe product set. The demand function $\pi(\cdot)$ characterizes the probability of customers choosing any subset, i.e., for any subset $\mathcal{T} \subseteq \mathcal{N}$, $\pi(\mathcal{T})$ is the probability that a random customer buys the set $\mathcal{T}$. Then, $\pi(\mathcal{T}) \geq 0$ and $\sum_{\mathcal{T} \subseteq \mathcal{N}} \pi(\mathcal{T}) = 1$. Without loss of generality, we assume that all the products in $\mathcal{N}$ can be reached, i.e., for any $n \in \mathcal{N}$, there exists $\mathcal{T} \subseteq \mathcal{N}$ such that $n \in \mathcal{T}$ and $\pi(\mathcal{T}) > 0$.

For the assortment selection problem, an e-company selects a subset of products $\mathcal{S} \subseteq \mathcal{N}$ to store in the local warehouse. The set $\mathcal{S}$ contains at most K products. Given that $\mathcal{S}$ is stored, if a customer orders a set $\mathcal{T} \subseteq \mathcal{N}$, then the company incurs a basic cost if $\mathcal{T}$ can be completely fulfilled by $\mathcal{S}$. However, if $\mathcal{T}$ cannot be completely fulfilled by $\mathcal{S}$, then the company incurs an additional cost of $C(\mathcal{T}|\mathcal{S})$ to fulfill $\mathcal{T}$. In particular, we consider two types of order-dependent additional fulfillment cost functions: (i) *type-I cost function* $C(\mathcal{T}|\mathcal{S}) = G(\mathcal{T})$ if $\mathcal{T} \nsubseteq \mathcal{S}$ and 0 otherwise, (ii) *type-II cost function* $C(\mathcal{T}|\mathcal{S}) = G(\mathcal{T} \backslash \mathcal{S})$ if $\mathcal{T} \nsubseteq \mathcal{S}$ and 0 otherwise, where $G(\cdot)$ is a non-negative set function satisfying $G(\emptyset) = 0$. Intuitively, if $\mathcal{T} \subseteq \mathcal{S}$, there would not be any additional fulfillment

cost; otherwise, the type-I cost depends on $\mathcal{T}$, while the type-II cost depends on $\mathcal{T}\backslash\mathcal{S}$. When $\mathcal{T} \nsubseteq \mathcal{S}$, additional costs would incur. On the one hand, the type-I cost function is plausible if the company does not split the order and fulfill the entire order $\mathcal{T}$ from elsewhere (e.g., from the back-end warehouse). On the other hand, the type-II cost function is plausible if the company partially fulfills the order $\mathcal{T}$ using the products stored at the local warehouse and fulfills the remaining order from elsewhere, in which case the fulfillment cost would depend on those products that are not stored at the local warehouse. The expected additional fulfillment cost is $f(\mathcal{S}) = \sum_{\mathcal{T}\subseteq\mathcal{N}} \pi(\mathcal{T})C(\mathcal{T}|\mathcal{S})$.

The company's goal is to select $\mathcal{S} \subseteq \mathcal{N}$ to minimize its expected fulfillment cost. Since all orders must be fulfilled, minimizing the total fulfillment is equivalent to minimizing the additional fulfillment cost. Therefore, the company solves the following cardinality-constrained set function minimization problem:

$$(CP): \min_{\mathcal{S}\subseteq\mathcal{N},\ |\mathcal{S}|\leq K} f(\mathcal{S}),$$

where $|\mathcal{S}|$ refers to the number of products in $\mathcal{S}$. Note that different from the product assortment planning problems in revenue management literature, here all customers share the same offer set, the universe product set $\mathcal{N}$, and the warehouse assortment $\mathcal{S}$ is selected from the offer set. To facilitate the discussion, we define several common properties of a set function.

Definition 1. Consider a set function $G(\mathcal{T}), \mathcal{T} \subseteq \mathcal{N}$. (i) $G(\cdot)$ is *increasing(decreasing)* if $G(\mathcal{T}) \leq (\geq) G(\mathcal{T}')$ for all $\mathcal{T} \subseteq \mathcal{T}' \subseteq \mathcal{N}$; (ii) $G(\cdot)$ is *size-based* if there exist a function $c(\cdot)$ such that $G(\mathcal{T}) = c(|\mathcal{T}|)$ for all $\mathcal{T} \subseteq \mathcal{N}$; (iii) $G(\cdot)$ is *binary* if $G(\mathcal{T}) = 1$ for all $\mathcal{T} \neq \emptyset$, and $G(\emptyset) = 0$; (iv) $G(\cdot)$ is *submodular(supermodular)* if $G(\mathcal{T}) + G(\mathcal{T}') \geq (\leq) G(\mathcal{T} \cup \mathcal{T}') + G(\mathcal{T} \cap \mathcal{T}')$ for all $\mathcal{T}, \mathcal{T}' \subseteq \mathcal{N}$.

Throughout the paper, we refer to increasing and decreasing in the weak sense.

3.1. Monotonicity and Submodularity

If $K = N$, it is obvious that the optimal policy for the company is to store all of the products and incur zero additional fulfillment cost. If $K < N$ and the objective $f(\cdot)$ is decreasing, then offering an assortment $\mathcal{S}$ of size K is optimal. In the following proposition, we provide conditions under which $f(\cdot)$ is decreasing.

PROPOSITION 1. *The objective function $f(\cdot)$ is decreasing if either of the following conditions holds: (i) the cost function is type-I; (ii) the cost function is type-II and $G(\cdot)$ is increasing. Moreover, in this case, (CP) is equivalent to (CP') provided as follows:*

$$(CP'): \min_{\mathcal{S}\subseteq\mathcal{N},\ |\mathcal{S}|=K} f(\mathcal{S}).$$

All the proofs of statements can be found in Section EC.1. Note that the objective function is decreasing under the type-I cost function even if $G(\cdot)$ is not monotone. This is because if order $\mathcal{T}$ is not a subset of $\mathcal{S}$, then the additional fulfillment cost is not a function of $\mathcal{S}$. Since the goal of (CP) is to minimize the expected additional fulfillment cost, a larger $\mathcal{S}$ is clearly better.

We can also show that the objective function $f(\cdot)$ is submodular under some general conditions.

THEOREM 1. *The objective function $f(\cdot)$ is submodular if either of the following conditions holds: (i) the cost function is type-I; (ii) the cost function is type-II and $G(\cdot)$ is submodular. In particular, if $G(\cdot)$ is size-based, then it is submodular if and only if $c(\cdot)$ is concave.*

Nonetheless, $f(\mathcal{S})$ being monotone and submodular does not imply that (CP) can be easily solved. To see this, note that if $f(\mathcal{S})$ is a submodular function, then its complement function, $g(\mathcal{S}) \triangleq f(\mathcal{N}\backslash\mathcal{S})$, is also a submodular function. Since $f(\mathcal{S})$ is submodular implies that $g(\mathcal{S})+g(\mathcal{S}') = f(\mathcal{N}\backslash\mathcal{S})+f(\mathcal{N}\backslash\mathcal{S}') \geq f\left((\mathcal{N}\backslash\mathcal{S})\cup(\mathcal{N}\backslash\mathcal{S}')\right)+f\left((\mathcal{N}\backslash\mathcal{S})\cap(\mathcal{N}\backslash\mathcal{S}')\right) = f\left(\mathcal{N}\backslash(\mathcal{S}\cap\mathcal{S}')\right)+f\left(\mathcal{N}\backslash(\mathcal{S}\cup\mathcal{S}')\right) = g(\mathcal{S}\cap\mathcal{S}')+g(\mathcal{S}\cup\mathcal{S}')$, $g(\mathcal{S})$ is also submodular. Therefore, (CP) is equivalent to minimizing $g(\mathcal{S})$ subject to $|\mathcal{S}| \geq N-K$, which is a submodular minimization with cardinality lower bound (SMCL).

In the existing literature, a number of algorithms have been developed to approximately solve (SMCL) problems (e.g., Svitkina and Fleischer 2011, Nagano et al. 2011, Iyer and Bilmes 2013, Goemans et al. 2009). Note that the monotonicity and submodularity properties, as demonstrated in Proposition 1 and Theorem 1, hold for any demand function $\pi(\cdot)$. Consequently, (CP) can be addressed using existing techniques from the (SMCL) problem for any demand function $\pi(\cdot)$, provided the aforementioned conditions are met.

3.2. Special Case: Order Fill Rate Maximization

An important special case of the cost function is when $G(\cdot)$ is binary. In this case, both type-I and type-II cost functions reduce to: $C(\mathcal{T}|\mathcal{S})$ equals to 0 if $\mathcal{T} \subseteq \mathcal{S}$, and 1 otherwise. Let $\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_M$ be the M different types of orders with choice probabilities $\pi_1, \pi_2, \ldots, \pi_M > 0$ and $\sum_{m=1}^{M} \pi_m = 1$. Then, $f(\mathcal{S}) = \sum_{m=1}^{M} \pi_m \mathbb{I}(\mathcal{T}_m \nsubseteq \mathcal{S})$, where $\mathbb{I}(\mathcal{A})$ is the indicator function that event $\mathcal{A}$ occurs, is the probability that the local warehouse cannot completely fulfill an order. Thus, $\text{OFR}(\mathcal{S}) \triangleq \sum_{m=1}^{M} \pi_m \mathbb{I}(\mathcal{T}_m \subseteq \mathcal{S}) = 1 - f(\mathcal{S})$ is the *order fill rate* (OFR) of the local warehouse. Then, (CP) can be reformulated as an *order fill rate maximization problem* (OFRM) with the same optimal solution but $\text{OFR}(\cdot)$ as the objective function

$$(\text{OFRM}): \max_{\mathcal{S} \subseteq \mathcal{N},\ |\mathcal{S}| \leq K} \text{OFR}(\mathcal{S}). \tag{1}$$

This formulation is also seen in Wu et al. (2019) and Söylemez (2021). Its corresponding cost minimization formulation, i.e., the *order fill rate maximization problem in the cost form (OFRM-C) is as follows*

$$\textit{(OFRM-C)}: \min_{\mathcal{S} \subseteq \mathcal{N},\ |\mathcal{S}| \leq K} 1 - \text{OFR}(\mathcal{S}). \tag{2}$$

The following corollary follows directly from Proposition 1 and Theorem 1.

Corollary 1. *The objective function in (OFRM) is increasing and supermodular, while the objective function in (OFRM-C) is decreasing and submodular. Thus, (OFRM-C) belongs to (SMCL). The size constraints in both (OFRM-C) and (OFRM) are tight under optimality.*

To our knowledge, this is the first result that relates (OFRM) to the submodular minimization problem. Due to this result, existing methods for (SMCL) can be directly applied to (OFRM).

As demonstrated in Svitkina and Fleischer (2011), approximating (SMCL) to a factor of $o\left(\sqrt{\frac{n}{\log n}}\right)$ is unattainable even when the objective function is monotonic. However, this complexity result is not directly applicable to (OFRM-C) since it is a special case of (SMCL). On the other hand, although the algorithms for general submodular function minimization with cardinality constraint (e.g., Goemans et al. 2009, Svitkina and Fleischer 2011, Nagano et al. 2011, Iyer and

Bilmes 2013) can be applied to (CP) under mild conditions, the algorithms are usually very complex and the guaranteed performance ratio are usually not attractive for this special case. In light of these considerations, we shall dig deeper into the warehouse assortment selection problem in the subsequent subsections. Specifically, we show that even the order fill rate maximization problem, the simplest special case of our problem, is already NP-hard. Also, we introduce two mixed-integer linear programming formulations tailored to different cost functions, allowing us to find the exact solution of (CP). Besides, unlike the generally complicated algorithms proposed in the literature for approximating (SMCL), we present an easy-to-implement heuristic and prove its optimality under some mild conditions.

4. Problem Solving

In this section, we study the tractability of (CP). First, we show that even the (OFRM) with the largest order size being two is NP-hard, and the approximation ratio of the greedy algorithm can be arbitrarily bad. Second, we introduce two MILP formulations for solving (CP) with respect to two different types of cost functions. In addition, we propose a dynamic programming approach to find the optimal assortment under type-I cost functions when the orders are non-overlapping.

4.1. Computational Complexity and the Greedy Policy

To study the computational complexity of (CP), in this subsection, we focus on the (OFRM), which is a special case of (CP) for both types of cost functions. For ease of analysis, we denote $L \in [N]$ as the largest order size, where $[n]$ denotes $\{1, 2, \ldots, n\}$ for any $n \in \mathbb{N}_+ \triangleq \{1, 2, \ldots\}$. Without specifying a particular L, $L = N$ by default. We first show that (OFRM) is already NP-hard when $L = 2$ by a reduction from the Densest k-Subgraph Problem.

THEOREM 2. *Both (OFRM) and (OFRM-C) are NP-hard even when $L = 2$. Thus, (OFRM), (OFRM-C), and (CP) are NP-hard.*

To our knowledge, the only related result in the literature is the NP-hardness result shown in Catalán and Fisher (2012) for the two-warehouse split-order minimization problem, which is much more general than (OFRM). They show the hardness by reducing the problem to the graph

bisection problem. Instead, we show that the single-warehouse (OFRM) when $L = 2$ includes the Densest k-Subgraph Problem as a special case, and thus is NP-hard.

For the submodular optimization problems, the greedy policy is commonly discussed and studied. The greedy policy adds products to the assortment in a greedy manner, i.e., in each step, the policy adds one product that improves the objective the most. It is not hard to see that the greedy policy can perform arbitrarily badly even for small N.

EXAMPLE 1. [Greedy Policy] Assume $\mathcal{N} = \{1, 2, 3\}$ and $K = 2$. There are two possible orders $\mathcal{T}_1 = \{1\},\ \mathcal{T}_2 = \{2, 3\}$, with choice probabilities $\pi(\mathcal{T}_1) = \epsilon,\ \pi(\mathcal{T}_2) = 1 - \epsilon$ for some small $\epsilon > 0$. Then, the optimal solution is clearly $\mathcal{S} = \{2, 3\}$ with the fill rate $1 - \epsilon$. However, the greedy policy would add product 1 in the first step and thus yield a fill rate ϵ. As ϵ approaches 0, the approximation ratios for (OFRM) and (OFRM-C) are arbitrarily bad. □

Besides the arbitrarily bad approximation ratio in the worst case, the greedy policy has two shortfalls. First, it is highly possible that more than one product leads to the same maximum objective improvement. Then, a tie occurs, resulting in different assortment outputs for different tie-breaking policies. Second, the computational complexity of the greedy policy could be high. It takes $\mathcal{O}(KNM)$ operations to solve (OFRM) and (OFRM-C), which can be unacceptable for real-world applications (details will be shown in Section 6.1).

4.2. Mixed Integer Linear Programming Formulations

In this subsection, we derive two mixed integer linear programming (MILP) formulations to solve (CP) where the cost function is type-I or type-II size-based.

4.2.1. (CP) with type-I cost functions. Let $\boldsymbol{\xi} \in \{0, 1\}^N$ be the binary decision variable with $\xi_n = 1$ if product n is included in the assortment $\mathcal{S}$ and 0 otherwise. Let $\boldsymbol{\zeta} \in \{0, 1\}^M$ be the binary decision variable with $\zeta_m = 1$ if order m can be fulfilled by $\mathcal{S}$, i.e. $\mathcal{T}_m \subseteq \mathcal{S}$, and 0 otherwise. Then, for a given type-I cost function, the objective function can be written as $f(\mathcal{S}) = \sum_{m=1}^{M} \pi_m g_m (1 - \zeta_m) = \sum_{m=1}^{M} \pi_m g_m - \sum_{m=1}^{M} \pi_m g_m \zeta_m$, where $g_m = G(\mathcal{T}_m)$. Note that the first term in the objective is a constant, thus minimizing $f(\mathcal{S})$ is equivalent to maximizing $\sum_{m=1}^{M} \pi_m g_m \zeta_m$.

Now we provide the following MILP for (CP):

$$\begin{aligned}
\max_{\boldsymbol{\xi},\boldsymbol{\zeta}} \ & \sum_{m=1}^{M} \pi_m g_m \zeta_m, \\
\text{s.t.} \ & \sum_{n=1}^{N} \xi_n = K, \\
& \zeta_m \le \xi_n, \qquad \forall m \in [M],\ n \in \mathcal{T}_m, \\
& \xi_n \in \{0,1\}, \quad \forall n \in [N], \\
& \zeta_m \ge 0, \qquad \forall m \in [M],
\end{aligned} \tag{3}$$

In Problem (3), the first constraint implies $\mathcal{S}$ should consist of K products, which is a direct consequence of Proposition 1. The rest of the constraints imply that an order can be fulfilled only if all the products in the order are included. Since the formulation is separable in $\boldsymbol{\zeta}$, given that $\boldsymbol{\xi}$ is binary, ζ_m has to be binary for all m under optimality.

Compared to the MILP formulation in Wu et al. (2019), formulation (3) has two distinct features. First, we can solve (CP) with any type-I cost by solving Problem (3) whereas the MILP formulations in Wu et al. (2019) are only for solving (OFRM). Second, $\boldsymbol{\zeta}$ is not binary, while the existing formulations impose the binary assumption. This simple twist of the formulation greatly reduces the number of integer variables, from $M+N$ to N.

Given that M can be as large as 2^N, directly solving MILP (3) may be infeasible when N or M is substantially large. As a remedy, we introduce a method utilizing Benders decomposition to solve MILP (3) in Section EC.2.

4.2.2. (CP) with type-II size-based cost functions Let $\boldsymbol{\xi} \in \{0,1\}^N$ be the binary decision variable with $\xi_n = 1$ if product n is included in the assortment $\mathcal{S}$ and 0 otherwise and $\boldsymbol{\tau} = [\tau_{m,l}]_{M\times(N+1)} \in \{0,1\}^{M\times(N+1)}$ be the binary decision variable with $\tau_{m,l} = 1$ if order m has l products that cannot be fulfilled by $\mathcal{S}$, i.e. $|\mathcal{T}_m \backslash \mathcal{S}| = l$, and 0 otherwise. Note that l can take the value 0 for any $m \in [M]$ and $\tau_{m,l} = 0$ for $l > |\mathcal{T}_m|$. In addition, for a given $m \in [M]$, only one $\tau_{m,l}$, $l \in \{0,1,\ldots,|\mathcal{T}_m|\}$, can take the value 1 whereas the rest take the value 0. Then, for a given increasing type-II size-based cost function $G(\mathcal{T}) = c(|\mathcal{T}|)$, the objective function can be written

as $f(\mathcal{S}) = \sum_{m=1}^{M} \sum_{l=0}^{|\mathcal{T}_m|} \pi_m c_l \tau_{m,l}$, where $c_l = c(l)$. The following MILP formulation for (CP) with increasing type-II size-based cost functions:

$$
\begin{aligned}
\min_{\boldsymbol{\xi},\boldsymbol{\tau}} \ & \sum_{m=1}^{M} \sum_{l=1}^{|\mathcal{T}_m|} \pi_m c_l \tau_{m,l}, \\
\text{s.t.} \ & \sum_{n=1}^{N} \xi_n = K, \\
& \sum_{l=0}^{|\mathcal{T}_m|} \tau_{m,l} = 1, && \forall m \in [M], \\
& \sum_{n \in \mathcal{T}_m} \xi_n = \sum_{l=0}^{|\mathcal{T}_m|} \left(|\mathcal{T}_m| - l\right) \tau_{m,l}, && \forall m \in [M], \\
& \xi_n \in \{0,1\}, && \forall n \in [N], \\
& \tau_{m,l} \geq 0, && \forall m \in [M],\ l \in \{0,1,\ldots,|\mathcal{T}_m|\}, \\
& \tau_{m,l} = 0, && \forall m \in [M],\ l \in \{|\mathcal{T}_m|+1,\ldots,N\}.
\end{aligned}
\tag{4}
$$

In Problem (4), the fourth constraint ensures that for any $m \in [M]$, only $|\mathcal{T}_m| - l$ products in $\mathcal{T}_m$ is included in $\mathcal{S}$ where l refers to the only $\tau_{m,l}$ that equals 1. Similar to the MILP for (CP) with type-I cost functions, $\tau_{m,l}$ is forced to be binary at optimality provided $\boldsymbol{\xi}$ is binary, and formulation (4) also only involves N binary variables.

To our knowledge, formulation (4) is the first MILP formulation in the literature for solving (CP) with type-II cost functions. It is structurally different from the MILP formulations in Catalán and Fisher (2012) and Wu et al. (2019), yet all of them include fill rate maximization as special cases.

4.3. Optimal Algorithm for Non-Overlapping Orders

This subsection explores a specific instance of the (OFRM) that can be solved within polynomial time. Our focus rests on the scenario where there is no overlap between distinct orders, i.e., $\mathcal{T}_m \cap \mathcal{T}_{m'} = \emptyset$ for all $m, m' \in \tilde{M}$ and $m \neq m'$, where there are $\tilde{M}$ ($\tilde{M} \leq N$) possible orders denoted by $\mathcal{T}_1, \ldots, \mathcal{T}_{\tilde{M}}$ with corresponding sizes $s_1, \ldots, s_{\tilde{M}}$.

For (OFRM), if all potential orders are non-overlapping, then we can claim that: (i) introducing product j into $\mathcal{S}$ will not augment the expected order fill rate if $j \notin \mathcal{T}_m$ for all $m \in \tilde{M}$; (ii) the expected order fill rate is improved by π_m only if all the products within $\mathcal{T}_m$ are included in $\mathcal{S}$. Consequently, rather than determining which product to incorporate into the assortment, we can

consider each potential order as an integral unit. To facilitate analysis, we reindex the orders so that $\pi_1 \leq \pi_2 \leq \cdots \leq \pi_{\tilde{M}}$. We denote $\boldsymbol{x} = [x_1, \ldots, x_{\tilde{M}}]^\top \in \{0,1\}^{\tilde{M}}$ as the vector of the new decision variables, where $x_m = 1$ if $\mathcal{T}_m$ is included in $\mathcal{S}$ and 0 otherwise. Then, (OFRM) can be reformulated as

$$\begin{aligned} \max_{\boldsymbol{x}\in\{0,1\}^{\tilde{M}}} & \sum_{m=1}^{\tilde{M}} \pi_m x_m \\ \text{s.t.} & \sum_{i=1}^{\tilde{M}} s_m x_m \leq K. \end{aligned} \tag{5}$$

Problem (5) remains a nontrivial task to solve, as demonstrated by Example 1. One intuitive modification to the greedy policy is to select orders based on the profit-to-weight ratio π_i/s_i. The following example illustrates that even with this adjustment, the greedy policy can still yield arbitrarily bad results.

Example 2. [Modified Greedy Policy] Assume $\mathcal{N} = \{1, 2, \ldots, N\}$ and $K = N - 1$. There are two possible orders $\mathcal{S}_1 = \{1\}$ and $\mathcal{S}_2 = \{2, 3, \ldots, N\}$ with choice probabilities $\pi(\mathcal{S}_1) = \frac{1}{n} + \epsilon$ and $\pi(\mathcal{S}_2) = \frac{n-1}{n} - \epsilon$ for some small $\epsilon > 0$. The optimal solution is $\mathcal{S} = \{2, 3, \ldots, N\}$ with fill rate $\frac{n-1}{n} - \epsilon$. However, the modified greedy policy selects $\{1\}$ in the first step thus yields a fill rate of $\frac{1}{n} + \epsilon$. Let $\epsilon = 1/n^2$. As $n \to \infty$, the approximation ratio for Problem (5) is arbitrarily bad. □

However, the following proposition provides an efficient way to solve Problem (5).

Proposition 2. *Problem (5) can be solved in run time $O(NK)$ using dynamic programming.*

The key idea underlying the proof of Proposition 2 is to reduce Problem (5) to a $0-1$ knapsack problem. Although the $0-1$ knapsack problem is NP-complete (Williamson and Shmoys 2011), it can be effectively solved using a dynamic programming algorithm. Moreover, since s_m $\forall m \in [\tilde{M}]$ are integers and K is upper bounded by N, it turns out the algorithm becomes polynomial-time for Problem (5).

In light of Proposition 2, we can effectively solve (OFRM) to exact optimality without solving the MILP (3) when dealing with non-overlapping orders. It is also noteworthy that Proposition 2 can be easily extended to the cost minimization problem with arbitrarily type-I cost functions by replacing π_m with $\pi_m \cdot G(\mathcal{T}_m)$ in the analysis. In the following section, we will introduce a simple-to-implement heuristic and demonstrate its ability to achieve optimality under specific conditions.

5. The Marginal Choice Indexing Policy

In this section, we introduce a simple heuristic, referred to as the Marginal Choice Indexing (MCI) policy, which leverages the unique nature of the warehouse assortment selection problem. To define the MCI policy, we start by introducing an indexing rule. For a universe set $\mathcal{N}=\{1,2,\ldots,N\}$, an indexing rule $\mathcal{I}$ is a permutation of all elements in $\mathcal{N}$ with $\mathcal{I}(n)$ denoting the index of element n in $\mathcal{I}$ and $\mathcal{I}^{-1}(n)$ denoting the element indexed n.

Let $\omega_n=\sum_{\mathcal{T}\subseteq\mathcal{N}}\pi(\mathcal{T})\cdot\mathbb{I}(n\in\mathcal{T})$ be the marginal choice probability of product $n\in\mathcal{N}$. An indexing rule $\mathcal{I}$ is called marginal choice indexing (MCI) if all products are indexed in the descending order of their marginal choice probabilities, i.e. $\omega_{\mathcal{I}^{-1}(1)}\geq\omega_{\mathcal{I}^{-1}(2)}\geq\cdots\geq\omega_{\mathcal{I}^{-1}(N)}$. If more than two products have the same marginal choice probability, then any such indexing following a deterministic tie-breaking rule is an MCI. Then, the MCI policy chooses the first K products to be the assortment.

Intuitively, the MCI policy selects the most popular K products based on historical data. A salient feature of the MCI policy is that it only depends on the marginal choice probability of each product, which only involves N parameters. Thus, this policy is simple to calculate and implement. However, the MCI policy does not always guarantee a good solution. Here, we provide two examples to illustrate that the MCI policy can perform arbitrarily badly.

Example 3. [MCI Policy for (OFRM)] Assume $\mathcal{N}=\{1,2,3,4\}$ and $K=2$. There are four possible orders $\mathcal{T}_1=\{1,3\},\ \mathcal{T}_2=\{3,4\},\ \mathcal{T}_3=\{1,2\},\ \mathcal{T}_4=\{2,4\}$, with choice probabilities $\pi(\mathcal{T}_1)=0.5-\epsilon+\xi,\ \pi(\mathcal{T}_2)=\epsilon-\xi-\eta,\ \pi(\mathcal{T}_3)=\epsilon,\ \pi(\mathcal{T}_4)=0.5-\epsilon+\eta$ for some small $\epsilon,\xi,\eta>0$ such that $\xi>\eta$ and $\epsilon>\xi+\eta$. Then, we have $\omega_1=0.5+\xi,\ \omega_2=0.5+\eta,\ \omega_3=0.5-\eta,\ \omega_4=0.5-\xi$, and $\omega_1>\omega_2>\omega_3>\omega_4$. Apparently, the optimal solution is $\{1,3\}$, while the MCI policy gives $\{1,2\}$. For (2), we have $\lim_{\epsilon\to0^+}\frac{f(\{1,2\})}{f(\{1,3\})}=\lim_{\epsilon\to0^+}\frac{1-\epsilon}{1-(0.5-\epsilon+\xi)}=2$, which means selecting $\{1,2\}$ could incur twice the cost compared to selecting $\{1,3\}$. However, in terms of the order fill rate maximization Problem (1), we have $\lim_{\epsilon\to0^+}\frac{\mathrm{OFR}(\{1,2\})}{\mathrm{OFR}(\{1,3\})}=\lim_{\epsilon\to0^+}\frac{\pi(\{1,2\})}{\pi(\{1,3\})}=\lim_{\epsilon\to0^+}\frac{\epsilon}{0.5-\epsilon+\xi}=0$, which implies that the MCI policy can perform arbitrarily badly. □

Example 4. [MCI Policy for (OFRM-C)] Assume $N=2n$ for $n>1$. $\mathcal{N}=\{1,\ldots,2n\}$ and $K=n$. There are $n+1$ distinct orders $\mathcal{T}_1=\{1\},\ldots\ \mathcal{T}_n=\{n\},\ \mathcal{T}_{n+1}=\{n+1,\ldots,2n\}$, with choice

probabilities $\pi(\mathcal{T}_1) = \cdots = \pi(\mathcal{T}_n) = (n-1)/n^2,\ \pi(\mathcal{T}_{n+1}) = 1/n$. Then, we have $\omega_1 = \cdots = \omega_n = (n-1)/n^2 < \omega_{n+1} = \cdots = \omega_{2n} = 1/n$. The MCI policy gives solution $\{n+1, \ldots, 2n\}$, while the optimal policy is $\{1, \ldots, n\}$. For (2), we have $\lim_{n\to+\infty} \frac{f(\{n+1,\ldots,2n\})}{f(\{1,\ldots,n\})} = \lim_{n\to+\infty} \frac{1-1/n}{1-(n-1)/n} = \lim_{n\to+\infty} n - 1 = +\infty$. Thus, the MCI policy is arbitrarily bad. Note that as $n \to +\infty$, the fill rate under the optimal policy converges to 1 yet that of the MCI policy converges to 0. This implies that the MCI policy can perform arbitrarily badly, resulting in a low fill rate. □

Examples 3 and 4 indicate that the MCI policy may have arbitrarily bad performance compared to the optimal solution even for (OFRM). However, the MCI policy is simple and intuitive as it selects the products with the highest marginal choice probabilities. This motivates us to find conditions under which the MCI policy is optimal.

5.1. Optimality Condition of MCI Policy

We first make a relatively simple observation for the case where the cost is type-II linear size-based.

PROPOSITION 3. *If the cost function is type-II size-based and $c(|\mathcal{T}|)$ is linear in $|\mathcal{T}|$ for every subset $\mathcal{T} \subseteq \mathcal{N}$, then the MCI policy is optimal for (CP).*

The proof of Proposition 3 converts (CP) into a knapsack problem with the same weight. Thus, the greedy policy is optimal and equivalent to the MCI policy.

Proposition 3 can be easily extended if the cost function is of type-II and the additional fulfillment cost is the summation of the individual cost of the products that cannot be fulfilled by the target warehouse. Specifically, if the cost function is type-II with $G(\mathcal{T}) = \sum_{n\in\mathcal{T}} \kappa_n$, where $\kappa_n \geq 0$ represents the product-specific additional fulfillment cost associated with product $n \in \mathcal{N}$, for every subset $\mathcal{T} \subseteq \mathcal{N}$, and let $\omega_n\kappa_n$ be the choice-weighted fulfillment cost of product n, we define a modified MCI that ranks the products in descending sequence according to their choice-weighted fulfillment costs. Subsequently, the modified MCI policy selects the K products with the largest choice-weighted fulfillment costs to store. The following proposition establishes the optimality of this modified MCI policy.

PROPOSITION 4. *If the cost function is type-II with $G(\mathcal{T}) = \sum_{n\in\mathcal{T}} \kappa_n$ for every subset $\mathcal{T} \subseteq \mathcal{N}$, then a modified MCI Policy is optimal for (CP).*

When the product-specific additional fulfillment cost is the same across all products, Proposition 4 reduces to Proposition 3. However, these linear additional cost functions presented in Propositions 3 and 4 are rather restrictive and do not capture the non-linearity of the cost functions. In the remainder of this work, we focus on more general cost functions and primarily analyze the performance of the MCI policy.

To characterize the optimality conditions of the MCI policy, we define the *dominant indexing rule* as follows.

Definition 2 (Dominant Indexing Rule w.r.t. Demand Function π). For a universe set $\mathcal{N}=\{1,2,\ldots,N\}$ and a given demand function π, we call an indexing rule $\mathcal{I}$ is *dominant w.r.t demand function* π if $\pi(\mathcal{S})\geq\pi(\mathcal{T})$ holds for any pair of subsets $\mathcal{S}=\{s_i\}_{i=1}^k\subseteq\mathcal{N}$ and $\mathcal{T}=\{t_i\}_{i=1}^k\subseteq\mathcal{N}$ (of the same size $k\in[N]$) with $\mathcal{I}(s_1)<\cdots<\mathcal{I}(s_k)$, $\mathcal{I}(t_1)<\cdots<\mathcal{I}(t_k)$, and $\mathcal{I}(s_i)\leq\mathcal{I}(t_i)\ \forall i\in[k]$. Note that for a given demand function π, a dominant indexing rule may not always exist. Intuitively, if a demand function has a dominant indexing rule, then, given any two subsets of the same size, the subset with smaller indexed components has a higher choice probability than the other with larger indexed components. The following theorem identifies a general condition for the MCI policy to be optimal.

Theorem 3. *For a given demand function π, if an indexing rule is dominant w.r.t. π, then it must be an MCI, and the corresponding MCI policy is optimal, provided that the cost function satisfies one of the following conditions: (i) type-I size-based; (ii) increasing type-II size-based.*

Theorem 3 equips us with a simple sufficient condition to derive the optimal solution for (CP), i.e. finding a dominant indexing rule among the class of MCIs. If a dominant indexing rule exists, then it must be an MCI; however, the reverse may not always be true. For example, consider $\mathcal{N}=\{1,2,3\}$ and $\pi(\{1\})=0.4,\pi(\{2\})=0.1,\pi(\{3\})=0,\pi(\{1,2\})=0.1,\pi(\{1,3\})=0.1,\pi(\{2,3\})=0.3,\pi(\{1,2,3\})=0$. Since $\omega_1=\pi(\{1\})+\pi(\{1,2\})+\pi(\{1,3\})+\pi(\{1,2,3\})=0.6>\omega_2=\pi(\{2\})+\pi(\{1,2\})+\pi(\{2,3\})+\pi(\{1,2,3\})=0.5>\omega_3=\pi(\{3\})+\pi(\{1,3\})+\pi(\{2,3\})+\pi(\{1,2,3\})=0.4$, then the current index is the unique MCI. However, $\pi(\{1,3\})<\pi(\{2,3\})$ indicating that the current index is not dominant.

Next, we show that for a wide range of demand functions, the dominant indexing exists and can be easily found.

Single-Purchase Discrete Choice Models. When each customer purchases at most one product, the single-purchase Discrete Choice Models (DCMs) apply and $\omega_n = \pi_{DCM}(\{n\})\ \forall n \in \mathcal{N}$, i.e. the choice probability of each product is exactly its marginal choice probability. Obviously, any MCI is a dominant indexing rule in this special case. In addition, the two types of cost functions are the same and can be represented as $C(\{n\}|\mathcal{S}) = 0$ if $n \in \mathcal{S}$, and $C(\{n\}|\mathcal{S}) = G(\{n\})$ if $n \notin \mathcal{S}$. Note that the cost function is product-dependent. We have the following proposition stating the optimal assortment selection under the single-purchase DCMs.

PROPOSITION 5. *If each customer purchases at most one product, then the optimal assortment is to select the K products that have the largest $\pi_{DCM}(\{n\})G(\{n\})$. If the cost function is product-independent, i.e. $G(\cdot)$ is a constant independent of n $\forall n \in \mathcal{N}$, then the optimal assortment is to select the K products that have the largest $\pi_{DCM}(\{n\})$, i.e. the MCI policy is optimal.*

Proposition 5 implies that it is sufficient to consider whether a single product can be fulfilled by the local warehouse or not, and there is no partial fulfillment. Hence, the optimal solution to (CP) is to select the products with the largest expected cost of not being selected.

Independent Choice Models. Under the independent choice model (ICM), a customer selects each product independently of the other products (see e.g., Lin et al. 2022 for reference). We denote p_n as the probability of product $n \in \mathcal{N}$ being selected. Then, for any set $\mathcal{T} \subseteq \mathcal{N}$, the probability it is selected is $\pi_{ICM}(\mathcal{T}) = \prod_{i\in\mathcal{T}} p_i \prod_{j\in\mathcal{N}\setminus\mathcal{T}}(1-p_j)$. It is easy to verify that $\sum_{\mathcal{T}\subseteq\mathcal{N}} \pi_{ICM}(\mathcal{T}) = 1$. We index the products such that $p_1 \geq p_2 \geq \cdots \geq p_N \geq 0$. Since $\omega_n = \sum_{\mathcal{T}\subseteq\mathcal{N}} \pi_{ICM}(\mathcal{T}) \cdot \mathbb{I}(n \in \mathcal{T}) = p_n$, the decreasing order of $p_n\ \forall n \in \mathcal{N}$ is an MCI, and we have the following proposition.

PROPOSITION 6. *Under the ICM, indexing based on the decreasing order of $\boldsymbol{p} = \{p_1, .., p_N\}$, i.e. $p_1 \geq p_2 \geq \cdots \geq p_N$, is an MCI and is dominant. Hence, an optimal solution to (CP) is $\mathcal{S}^* = \{1, 2, \ldots, K\}$, provided the cost function is type-I size-based or increasing type-II size-based.*

Multi-Choice Random Independent Utility Models. One of the general multi-purchase choice models is the multi-choice random utility model (MC-RUM) proposed by Lin et al. (2022). In this model, a customer may purchase multiple products from the universe product set $\mathcal{N}$ with a random intended purchase quantity (IPQ) Q that can take a value of $\{0,1,\ldots,N\}$. Each product n has a utility U_n, $n \in \mathcal{N}$, where product 0 is the outside option. We let $\boldsymbol{U} = \{U_1,\ldots,U_N\}$, $\boldsymbol{U}^+ = \boldsymbol{U} \cup \{U_0\}$, and $\mathcal{N}^+ = \mathcal{N} \cup \{0\}$. In general, $\boldsymbol{U}^+$ and Q are jointly distributed. A customer will choose at most Q different products whose utilities are the highest and larger than the utility of the outside option. The probability that a customer purchases set $\mathcal{T} \subseteq \mathcal{N}$ is $\pi_{MC-RUM}(\mathcal{T}) = \mathbb{P}\left(Q = |\mathcal{T}|, \min_{i\in\mathcal{T}} U_i > \max_{j\in\mathcal{N}^+\setminus\mathcal{T}} U_j\right) + \mathbb{P}\left(Q > |\mathcal{T}|, \min_{i\in\mathcal{T}} U_i > U_0, U_0 > \max_{k\in\mathcal{N}\setminus\mathcal{T}} U_k\right)$. In particular, the ICM is a special case of MC-RUM by letting $\mathbb{P}(Q = N) = 1$.

Suppose $\boldsymbol{U}^+$ and Q are independent, $U_1, U_2, \ldots, U_N$ are independent and no assumption is posed on the distribution of U_0. We call this model the *Multi-Choice Random Independent Utility Model* (MC-RIUM). In order to characterize the structure of the optimal warehouse assortment under MC-RIUM, we introduce the definition of the first-order stochastic dominance (see, e.g., Section 6.B.1 of Shaked and Shanthikumar 2007 for reference).

Definition 3 (First-Order Stochastic Dominance (FSD)). For two random variables U' and U'' with cumulative distribution functions (CDFs) F' and F'', respectively, U' first-order stochastic dominates U'' (denoted as $U' \succeq_1 U''$) if and only if $F''(u) \geq F'(u)\ \forall u \in (-\infty, \infty)$.

Then, we have the following proposition.

Proposition 7. *Under the MC-RIUM, if products can be indexed such that utilities are in the descending FSD order, i.e., $U_1 \succeq_1 U_2 \succeq_1 \cdots \succeq_1 U_N$, then this indexing is an MCI and is dominant. Hence, the optimal solution for (CP) is $\mathcal{S}^* = \{1,2,\ldots,K\}$, provided the cost function is type-I size-based or increasing type-II size-based.*

Moreover, if $U_n = V_n + \epsilon_n\ \forall n \in \mathcal{N}^+$ for some deterministic utility $\boldsymbol{V}^+ = [V_0, V_1, \ldots, V_N]^\top$ and idiosyncratic noise $\boldsymbol{\epsilon} = [\epsilon_0, \epsilon_1, \ldots, \epsilon_N]^\top$, then $U_1, U_2, \ldots, U_N$ are independent is equivalent to $\epsilon_1, \ldots, \epsilon_N$ are independent. In this case, the multi-purchase multinomial logit (MP-MNL) model proposed by Bai

et al. (2023) is a special case of the MC-RIUM by setting $\boldsymbol{\epsilon} = [\epsilon_0, \epsilon_1, \ldots, \epsilon_N]^\top$ to be i.i.d. Gumbel. Furthermore, if $\epsilon_1, \ldots, \epsilon_N$ are independent and identically distributed, then the decreasing FSD order of the utility is equivalent to the decreasing order of the deterministic utility. Thus, we have the following corollary as a direct consequence of Proposition 7.

COROLLARY 2. *Under the MC-RIUM, if random utilities have the form* $U_n = V_n + \epsilon_n \ \forall n \in \mathcal{N}^+$ *and* $\epsilon_1, \ldots, \epsilon_N$ *are independent and identically distributed, then indexing based on the decreasing order of deterministic utility, i.e.,* $V_1 \geq V_2 \geq \cdots \geq V_N$*, is an MCI and is dominant.*

Bundle Multivariate Logit Models. The Bundle Multivariate Logit (BundleMVL) model is first proposed by Russell and Petersen (2000) to solve the market basket selection problem and is brought to solve the assortment optimization problem by Tulabandhula et al. (2023). In the BundleMVL model, for any given maximum purchase quantity $L \in [N]$ (a predetermined parameter), the conditional random utility of selecting product $n \in \mathcal{N}$ can be represented as $U(n|\{X_{n'} = x_{n'} \colon n' \in \mathcal{N}, n' \neq n\}) = (V_n + \sum_{n' \in \mathcal{N}, n' \neq n} \beta_{nn'} x_{n'} + \epsilon_n)\mathbb{I}(\sum_{j \in \mathcal{N}, n' \neq n} x_{n'} < L)$, where $X_{n'} \ \forall n' \in \mathcal{N}$ represent binary random variables that signify whether product n' is chosen or not ($x_{n'}$ are the corresponding realizations), V_n is the intrinsic utility and ϵ_n is the Gumbel distributed random noise of product n, and parameters $\beta_{nn'}$ capture interactions between product pairs n and n' ($\beta_{nn'} = \beta_{n'n}$). Then, the probability of choosing subset $\mathcal{T} \subseteq \mathcal{N}$ with $|\mathcal{T}| \leq L$ can be represented as $\pi_{BundleMVL-L}(\mathcal{T}) = \frac{V_\mathcal{T}}{1 + \sum_{\mathcal{T}' \subseteq \mathcal{N}, |\mathcal{T}'| \leq L} V_{\mathcal{T}'}}$, where $V_\mathcal{T} = \exp\left(\sum_{n \in \mathcal{N}} V_n x_n + \sum_{n \in \mathcal{N}} \sum_{n' \in \mathcal{N}, n' < n} \beta_{nn'} x_n x_{n'}\right)$, and $x_{n'} = 1$ if $n' \in \mathcal{T}$ and 0 otherwise. Note that positive $\beta_{nn'}$ implies complementary relations between product n and n', while negative $\beta_{nn'}$ implies substitution relations between product n and n'.

If $\beta_{ni} \geq \beta_{nj} \ \forall n \in \mathcal{N} \backslash \{i, j\}$ for any $i, j \in \mathcal{N}$ with $V_i \geq V_j$, then indexing according to the decreasing order of deterministic utilities is dominant. Formally, we have the following proposition.

PROPOSITION 8. *For any BundleMVL-L model with* $L \in [N]$*, if* $\beta_{ni} \geq \beta_{nj} \ \forall n \in \mathcal{N} \backslash \{i, j\}$ *for any* $i, j \in \mathcal{N}$ *with* $V_i \geq V_j$*, then indexing based on the decreasing order of the deterministic utilities, i.e.,* $V_1 \geq V_2 \geq \cdots \geq V_N$*, is an MCI and is dominant. Hence, the optimal solution for* (CP) *is* $\mathcal{S}^* = \{1, 2, \ldots, K\}$*, provided the cost function is type-I size-based or increasing type-II size-based.*

Note that the assumption on $\boldsymbol{\beta}$ is imposed on its relative values but not its absolute values. That is, given any $i, j \in \mathcal{N}$, $\beta_{ni} \geq \beta_{nj}$ for any $n \in \mathcal{N} \backslash \{i, j\}$ if $V_i \geq V_j$. Then, the utility of any subset containing product i is higher than that containing j provided the other products in the subset are the same. In other words, the products can be ranked consistently such that selecting a product with a smaller index would always result in a higher utility for any given subset of products. However, a product with a higher intrinsic utility does not indicate a higher level of complementarity (see Feng et al. 2018) with other products because $\beta_{nn'}$ could be negative for some n and n'.

6. A Case Study from RiRiShun Logistics

In this section, we conduct a case study on the real-world logistics operational-level data from RiRiShun (RRS) Logistics, who focuses on home appliance delivery and installation in China. The logistics network of RRS consists of 7 central distribution centers (CDCs), 26 regional distribution centers (RDCs), 100 local transfer centers (LTCs), and more than $6,000$ last-mile hubs (Guo et al. 2021). Due to the specialty of home appliances, which are usually bulky and heavy, their delivery services typically require special equipment and involve special installation procedures. According to Table 1, orders with more than one SKU consist of more than 23% of total orders. Thus, we can reasonably assume that any spillover order fulfillment or order splitting would incur non-negligible extra shipping and delivery costs. As a result, it is interesting to investigate whether this additional order fulfillment cost could be reduced by wisely planning the warehouse assortment.

Table 1 Proportion of Orders with Distinct Amounts of SKUs

Number of Distinct SKUs in Each Order	1	2	3	≥ 4
Proportion in Total Number of Orders	76.79%	22.98%	0.19%	0.04%

Since each DC also serves as a warehouse in RRS's logistics network (Guo et al. 2021) and no warehouse information is available, we do not differentiate DC and warehouse in the experiment. Since we focus on the strategic level of assortment selection for DCs, we assume that the inventory level of each SKU is large enough to satisfy all demand. Such an assumption would be plausible if RRS took care of the inventory control.

According to Guo et al. (2021), each order has an associated last-mile hub (LMH) that serves as the last stop before customers receive their orders. Hence, we treat each LMH as a demand zone.

For any LMH, we identify the dominant DC as the one that delivers the most orders to this LMH. We then allocate all LMHs to their dominant DC as service regions. Then, we have a set of DCs and their dedicated service regions. As we have no access to the actual fulfillment cost, we mainly focus on maximizing the OFR.

In the rest of this section, we undertake two sets of experiments. First, we evaluate the potential for improvement in the current assortment selections of the distribution centers and assess the performance of the proposed methods faced with demand changes. In addition, we test several distinct cost functions to gain further insights into the MCI policy. Second, we explore the assortment selection problem if we want to separate a front-end distribution center that only serves a chosen city. All the computation is done on a MacBook Pro 13-inch (2018) with 2.3 GHz quad-core Intel Core i5, and the MILPs are solved using Gurobi v9.0.2.

6.1. Experiments on Dominant Distribution Centers

From the data of the recorded year, we find 62 dominant DCs and their corresponding service regions. For any given dominant DC, the demand is determined by aggregating the demand of its service regions. In addition, we notice from the data that some SKUs stored in the DCs only ship to LMHs that are not part of their service regions. To ensure a fair comparison, we assume the current assortment of any given DC contains the SKUs that have been shipped from this DC to its service region. We set the size of the current assortment as the DC's *SKU capacity* K.

For the DCs whose SKU capacities are less than the total number of demanded SKUs, it is worth exploring if we can find a better assortment selection than the current assortment and improve the OFR. We compare OFR of the current DC assortment with that of the optimal (OPT) assortment computed using MILP in formulation (3) and the assortment derived by the MCI policy. We also tested the greedy algorithm, which iteratively selects the SKU such that it leads to the largest improvement of the OFR. However, due to $\mathcal{O}(KNM)$ complexity and the relatively large number of products, we find that for a case with $N = 5143$, $M = 5554$, and $K = 3000$, the greedy algorithm does not stop after several hours, which is much slower than solving the MILP (finishes within 3

seconds). For this reason, we do not bring the greedy policy into the comparison. We also compare the random pick (RP) policy that randomly selects an assortment of size K, and we denote $\mathcal{T}_{RP}$ as the randomly selected assortment. To reduce the variance of the performance of the RP policy while retaining a manageable run time, all the reported results of the RP policy are the average performance of 100 randomly selected assortments.

In the experiment, we do the train-test split based on the order date. The orders are recorded from May 30th, 2018 to Sept. 30th, 2019. However, there are 15 abnormal orders with dates prior to 2017 (1 in 2016 and 14 in 2000), which are subsequently omitted. Then, we set the first half of the orders whose order dates are before Feb. 1st, 2019 as the training set and the rest of the orders as the test set. Note that the demands, the DCs' current assortments, and the SKU capacity all depend on the training data. In the test set, all the SKUs that do not appear in the training data are eliminated. After this adjustment, 4 DCs end up with no valid orders, so we do not include them in our experiment.

We separate the results into three parts: LTC as dominant DC in Tables 2 and 3, RDC as dominant DC in Tables EC.1 and EC.2, and CDC as dominant DC in Tables EC.3 and EC.4. Since our main motivation is to study the warehouse assortment selection for local warehouses, the results for LTCs are our main focus. Due to space limitations, we only present the results for LTCs. The results for RDCs and CDCs can be found in Section EC.3.1.

In these tables, *# SKU* denotes the number of demanded SKUs in the service regions, *# Diff Orders* denotes the number of different orders, *OPT IMP Current* and *MCIP IMP Current* denote the improvement on the order fill rate of the MILP optimal assortment selection and the MCI policy selection over the current assortment, respectively. Solving the MILP, we found that: (i) LTCs have the potential for a 10.32% improvement in OFR on average over current assortments; (ii) RDCs have the potential for a 5.84% improvement in OFR on average over current assortments; (iii) CDCs have the potential for a 1.37% improvement in OFR on average over current assortments. LTCs have the most significant potential for improvement, as they have smaller capacity and

Table 2 In-Sample OFR Comparison Results for LTCs

DC Code	# SKUs	SKU Cap	# Diff Orders	Current OFR (%)	OPT OFR (%)	MCIP OFR (%)	Avg RP OFR (%)	OPT IMP Current (%)	MCIP IMP Current (%)	$\frac{ORF_{MCIP}}{ORF_{OPT}}$ (%)	MILP Time (s)
RRSZX005	672	367	609	71.6	87.01	86.51	51.94	21.52	20.82	99.42	0.26
RRSZX012	328	239	302	91.21	95.13	93.53	70.37	4.29	2.54	98.32	0.2
RRSZX021	28	13	15	75.51	83.67	83.67	21.14	10.81	10.81	100.0	0.17
RRSZX028	454	306	425	78.02	84.49	82.54	65.64	8.29	5.79	97.69	0.21
RRSZX034	36	27	19	85.11	89.36	85.11	57.77	5.0	0.0	95.24	0.14
RRSZX036	40	23	35	77.53	84.27	83.15	54.78	8.7	7.25	98.67	0.14
RRSZX037	3655	2013	3779	86.16	97.18	96.99	53.36	12.79	12.56	99.8	1.0
RRSZX047	828	577	784	91.05	95.1	94.79	68.72	4.45	4.11	99.67	0.58
RRSZX049	236	202	228	93.68	97.12	96.34	85.6	3.67	2.84	99.2	0.2
RRSZX050	3270	1380	3291	85.01	93.42	93.1	40.04	9.88	9.51	99.66	1.07
RRSZX055	41	24	41	58.73	73.02	73.02	58.4	24.32	24.32	100.0	0.34
RRSZX057	594	419	531	83.26	92.31	90.78	67.88	10.87	9.04	98.35	0.46
RRSZX061	1429	891	1279	86.49	93.67	93.63	58.78	8.31	8.26	99.96	0.68
RRSZX065	101	75	101	83.41	87.8	87.32	74.35	5.26	4.68	99.44	0.17
RRSZX066	202	149	177	84.86	90.0	85.95	71.75	6.05	1.27	95.5	0.21
RRSZX068	177	128	164	79.38	86.25	83.51	71.65	8.66	5.19	96.81	0.2
RRSZX075	182	147	185	92.28	93.57	93.57	80.46	1.39	1.39	100.0	0.18
RRSZX076	200	150	196	86.81	90.21	89.36	74.11	3.92	2.94	99.06	0.2
RRSZX077	626	335	605	73.15	84.05	83.81	52.89	14.89	14.56	99.72	0.42
RRSZX096	33	25	30	76.47	94.12	92.94	74.79	23.08	21.54	98.75	0.15
RRSZX105	2134	1087	1923	81.79	92.58	92.23	46.12	13.19	12.76	99.62	0.72
RRSZX106	2543	1195	2364	82.13	91.89	91.57	44.01	11.88	11.48	99.64	0.61
RRSZX107	951	503	837	77.44	89.88	89.59	48.84	16.06	15.69	99.68	0.3
RRSZX108	948	551	881	82.25	91.44	90.96	56.1	11.18	10.59	99.48	0.3
RRSZX109	39	28	42	78.75	86.25	86.25	68.57	9.52	9.52	100.0	0.18

can only store a limited number of products. This aligns with our focus on carefully selecting assortments for front-end distribution centers, which can significantly reduce fulfillment costs.

Surprisingly, although the MCI policy can perform poorly in some situations (e.g., Example 3), it performs extremely well in the experiments. We find that applying the MCI policy to select the assortments will have 9.18%, 5.59%, and 1.29% of improvement over the OFR over the current practice for LTCs, RDCs, and CDCs, respectively. This improvement ratio is very close to that of the MILP optimal selection, which is also revealed by $\frac{ORF_{MCIP}}{ORF_{OPT}}$ whose values all are nearly 100%. In contrast, the performance of the RP policy is rather poor, which implies the importance of optimizing the assortment for each warehouse or distribution center.

Besides examining the in-sample performance, we also apply the MILP optimal and the MCI assortments derived from the training set to the test set. Since we have no information on how the assortments are updated by the firm over time, we do not compare them with DCs' assortments in the test set. In these tables, *"HS OPT"* denotes the OFR calculated by the hindsight MILP optimal solution, i.e. deriving from solving the MILP with the underlying demand distribution.

Analyzing Table 3, Tables EC.2 and EC.4, we can see that the MILP optimal assortments computed from the training dataset are not necessarily optimal on the testing set. For LTCs, the MCI policy performs better in 14 out of 25 cases; for RDCs, the MCI policy performs better in 14 out of 26 cases; for CDCs, the MCI policy performs better in 3 out of 7 cases. Although the MCI

Table 3 Out-of-Sample OFR Comparison Results for LTCs

DC Code	# SKUs	SKU Cap	# Diff Orders	HS OPT OFR (%)	Test OPT OFR (%)	Test MCIP OFR (%)	Test MCIP IMP Over Test OPT (%)
RRSZX005	672	367	435	98.42	87.58	86.46	-1.28
RRSZX012	328	239	206	100.0	93.65	94.81	1.24
RRSZX021	28	13	8	100.0	46.71	51.5	10.26
RRSZX028	454	306	287	100.0	78.72	84.02	6.73
RRSZX034	36	27	15	100.0	96.17	97.12	1.0
RRSZX036	40	23	22	99.41	91.76	82.94	-9.62
RRSZX037	3655	2013	3380	99.01	93.07	93.45	0.41
RRSZX047	828	577	582	99.99	88.94	90.9	2.2
RRSZX049	236	202	150	100.0	98.5	97.01	-1.52
RRSZX050	3270	1380	2936	96.69	87.79	89.3	1.72
RRSZX055	41	24	15	100.0	90.0	85.0	-5.56
RRSZX057	594	419	333	100.0	81.7	85.51	4.66
RRSZX061	1429	891	934	99.37	92.76	92.68	-0.08
RRSZX065	101	75	54	100.0	89.94	85.85	-4.55
RRSZX066	202	149	97	100.0	86.83	86.52	-0.36
RRSZX068	177	128	78	100.0	77.01	80.46	4.48
RRSZX075	182	147	157	99.57	91.59	90.0	-1.74
RRSZX076	200	150	129	100.0	81.73	81.06	-0.81
RRSZX077	626	335	471	96.8	82.36	82.78	0.51
RRSZX096	33	25	16	100.0	97.78	98.33	0.57
RRSZX105	2134	1087	1578	97.89	88.94	88.95	0.02
RRSZX106	2543	1195	2079	96.77	87.78	88.61	0.95
RRSZX107	951	503	598	98.46	86.08	87.38	1.51
RRSZX108	948	551	619	99.35	91.44	91.27	-0.18
RRSZX109	39	28	33	100.0	94.08	87.57	-6.92

policy is not necessarily optimal, it outperforms the training-optimal solution for the majority of cases. This result implies that the MCI policy is robust to the change in demand.

Also, we carry out supplementary experiments in Section EC.3.3 to evaluate the performance of the MCI policy across various cost functions, as well as the modified MCI policy discussed in Proposition 4 when facing an additional fulfillment cost calculated as the sum of the costs for unfulfilled products. We find that, in comparison to current practices, the MCI policy yields substantial cost reductions regardless of which size-based cost function is tested. Furthermore, minor adjustments to the MCI policy can yield significant improvements, particularly when tailored to address specific type-II cost functions.

6.2. Experiments on the Most Popular Cities

In this section, we zoom into some popular cities or areas and investigate insights into warehouse assortment problems. We choose five popular cities that have the most orders during the recorded year. The details of the five cities are listed in Table EC.5 in Section EC.3. For each chosen city, we calculate the MILP optimal assortment (blue curve), the assortment derived from the MCI policy (red curve), and the assortment derived by the RP policy (green curve) with cardinality constraint being $\{1, 21, 41, 61, \dots\}$. Besides comparing the normal OFR, we also examine the partial fulfillment rate (PFR), i.e., the proportion of multi-purchase orders that are partially fulfilled (for single-purchase orders, as they can only be fully fulfilled or not fulfilled, they are out of consideration in this index). Specifically, PFR equals the proportion of multi-purchase orders that are partially fulfilled by the selected assortment divided by the total proportion of multi-purchase orders. Here, multi-purchase and single-purchase orders are referred to as orders with the purchase

of multiple distinct SKUs and a single SKU, respectively. A larger PFR indicates a more "inefficient" assortment selection. Moreover, to gain deeper insights into the MCI policy, we also plot the ratio of products that are both in the MCI assortment and the MILP optimal assortment. Because the structures of the results for all five cities are similar, we only present the results for the most popular city and the rest can be found in Section EC.3. In Figure 1, the red curves nearly merge

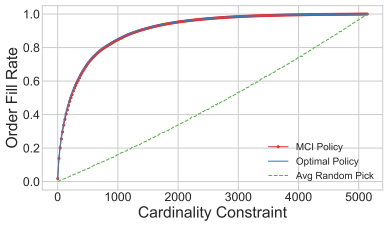

Figure 1 City 1 OFRs

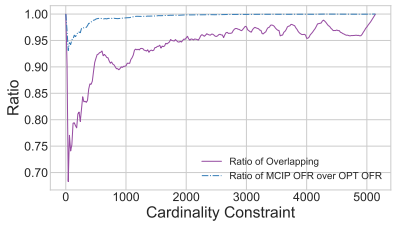

Figure 2 City 1 Ratios

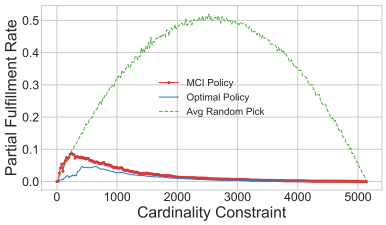

Figure 3 City 1 PFRs

with the blue curves implying that the MCI policy is near optimal for all values of the SKU capacity. Figure 2 implies that even if the MCI policy does not fully coincide with the MILP optimal solution, it still provides a near-optimal solution. Thus, when no further information about the customers' demands is available, the MCI policy can be a good choice.

Furthermore, a high OFR can be achieved with a relatively small SKU capacity since the OFR curve of the MILP optimal solution and MCI policy are concave with a steep slope at zero. For example, in Figure 1 with $N = 5143$, an assortment of around 2000 can achieve an OFR of over 90%. This observation suggests a way to design the assortment at warehouses in different levels of the logistics network: the front-end distribution centers should have a small SKU capacity and keep the most popular products to fulfill the majority of orders while the back-end distribution centers should connect several front-end distribution centers for fulfilling demands that cannot be completely satisfied by the front-end distribution centers.

We compare the performance of the RP policy, MCI policy, and MILP optimal solution with respect to PFR in Figure 3. We find that the MILP optimal solution results in the least PFR followed by the MCI policy. In contrast, the RP policy results in the highest PFR. Similar to the construction of Example 4, assortment selections with high PFR will result in a bad performance in improving OFR. These figures imply that the main difference between the MCI policy and the

MILP optimal solution is that the MCI policy does not reduce PFR as much as the MILP optimal solution. When the orders contain more items, the gap between these two policies may be larger.

7. Conclusion and Future Research Directions

In this study, we investigate the cardinality-constrained single warehouse assortment selection problem, which aims to minimize the fulfillment cost incurred at the local warehouse by selecting the optimal assortment. The problem includes the well-known order fill rate maximization problem, which we prove to be NP-hard. Indeed, we construct examples to illustrate that the greedy policy can perform arbitrarily badly. Although numerous well-established submodular minimization techniques from the literature may be employed when some trivial conditions are met, the general submodular minimization methodology does not account for the unique feature of the warehouse assortment selection problem. In response to this, we formulate two MILPs of the warehouse assortment selection problem for different cost functions. Moreover, leveraging the nature of the problem, we propose a simple heuristic, the MCI policy, which selects the K products with the highest marginal choice probabilities and shows that it can be optimal under mild conditions. Our numerical studies show that the MCI policy results in near-optimal performance and can improve the fill rate by 9.18%, on average, compared to the current practice for local transfer centers. Additionally, the MCI policy exhibits robustness in the face of demand fluctuations.

Our findings offer several promising avenues for future research. First, a natural extension to our work would be exploring the multi-warehouse assortment selection problem. The introduction of a multi-warehouse system considerably complicates the assortment selection problem, as various fulfillment policies must be considered. Besides, not only are the overall efficiency of the logistics network and cost reduction vital metrics, but the workload balance across the entire system also emerges as a potential concern. This complexity makes the multi-warehouse system a rich and fruitful direction for future investigation. Second, while our study primarily focuses on SKU capacity in warehouse planning, we recognize that inventory management remains a critical component of supply chain management. As such, a promising future direction lies in studying the joint warehouse assortment and inventory planning problem. By integrating these two aspects, researchers

may uncover novel insights and solutions that advance our understanding of optimal warehouse management and supply chain efficiency.

References

Alfaro JA, Corbett CJ (2003) The value of sku rationalization in practice (the pooling effect under suboptimal inventory policies and nonnormal demand). *Production and Operations Management* 12(1):12–29.

Aryapadi M, Ecker T, Spielvogel J (2020) Future of retail operations: Winning in a digital era. *McKinsey & Company* .

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

Bertsimas D, Mišić VV (2019) Exact first-choice product line optimization. *Operations Research* 67(3):651–670.

Bhat CR (2005) A multiple discrete–continuous extreme value model: formulation and application to discretionary time-use decisions. *Transportation Research Part B: Methodological* 39(8):679–707.

Catalán A, Fisher M (2012) Assortment allocation to distribution centers to minimize split customer orders. *Available at SSRN 2166687* .

Corneil DG, Perl Y (1984) Clustering and domination in perfect graphs. *Discrete Applied Mathematics* 9(1):27–39.

Cox DR (1972) The analysis of multivariate binary data. *Applied statistics* 113–120.

Feige U (1998) A threshold of ln n for approximating set cover. *Journal of the ACM (JACM)* 45(4):634–652.

Feige U, Mirrokni VS, Vondrák J (2011) Maximizing non-monotone submodular functions. *SIAM Journal on Computing* 40(4):1133–1153.

Feng G, Li X, Wang Z (2018) On substitutability and complementarity in discrete choice models. *Operations Research Letters* 46(1):141–146.

Gallego G, Topaloglu H (2019) *Revenue Management and Pricing Analytics*, volume 209 (Springer).

Gallego G, Wang R (2019) Threshold utility model with applications to retailing and discrete choice models. *Available at SSRN 3420155* .

Goemans MX, Harvey NJ, Iwata S, Mirrokni V (2009) Approximating submodular functions everywhere. *Proceedings of the twentieth annual ACM-SIAM symposium on Discrete algorithms*, 535–544 (SIAM).

Guo X, Yu Y, Allon G, Wang M, Zhang Z (2021) Ririshun logistics: Home appliance delivery data for the 2021 manufacturing & service operations management data-driven research challenge. *Manufacturing & Service Operations Management* .

Huh WT, Li H (2022) Optimal pricing under multiple-discrete customer choices and diminishing return of consumption. *Operations Research* 70(2):905–917.

Iyer RK, Bilmes JA (2013) Submodular optimization with submodular cover and submodular knapsack constraints. *Advances in neural information processing systems* 26.

Krause A, Guestrin CE (2012) Near-optimal nonmyopic value of information in graphical models. *arXiv preprint arXiv:1207.1394* .

Lee J, Mirrokni VS, Nagarajan V, Sviridenko M (2009) Non-monotone submodular maximization under matroid and knapsack constraints. *Proceedings of the forty-first annual ACM symposium on Theory of computing*, 323–332.

Lin H, Li X, Wu L (2022) Multi-choice preferences learning and assortment recommendation in e-commerce. *Available at SSRN 4035033* .

Lopienski K (2021) How can sku rationalization help you improve business performance. URL `https://www.shipbob.com/blog/sku-rationalization/`, accessed: 2023-08-16.

Lovász L (1983) Submodular functions and convexity. *Mathematical Programming The State of the Art* 235–257.

Lu Y, Song JS (2005) Order-based cost optimization in assemble-to-order systems. *Operations Research* 53(1):151–169.

Lu Y, Song JS, Yao DD (2003) Order fill rate, leadtime variability, and advance demand information in an assemble-to-order system. *Operations Research* 51(2):292–308.

Nagano K, Kawahara Y, Aihara K (2011) Size-constrained submodular minimization through minimum norm base. *ICML*.

Nemhauser GL, Wolsey LA (1978) Best algorithms for approximating the maximum of a submodular set function. *Mathematics of operations research* 3(3):177–188.

Nemhauser GL, Wolsey LA, Fisher ML (1978) An analysis of approximations for maximizing submodular set functions—i. *Mathematical programming* 14(1):265–294.

Russell GJ, Petersen A (2000) Analysis of cross category dependence in market basket selection. *Journal of Retailing* 76(3):367–392.

Ryzin Gv, Mahajan S (1999) On the relationship between inventory costs and variety benefits in retail assortments. *Management Science* 45(11):1496–1509.

Schrijver A (2003) *Combinatorial optimization: polyhedra and efficiency*, volume B (Springer), part IV, Chapters 39-49.

Shaked M, Shanthikumar JG (2007) *Stochastic orders* (Springer).

Song JS (1998) On the order fill rate in a multi-item, base-stock inventory system. *Operations research* 46(6):831–845.

Song JS, Yao DD (2002) Performance analysis and optimization of assemble-to-order systems with random lead times. *Operations Research* 50(5):889–903.

Söylemez D (2021) *Assortment Planning Considering Split Orders*. Ph.D. thesis, Bilkent Universitesi (Turkey).

Svitkina Z, Fleischer L (2011) Submodular approximation: Sampling-based algorithms and lower bounds. *SIAM Journal on Computing* 40(6):1715–1737.

Tulabandhula T, Sinha D, Karra SR, Patidar P (2023) Multi-purchase behavior: Modeling, estimation, and optimization. *Manufacturing & Service Operations Management* .

Wan X, Dresner ME (2015) Closing the loop: an empirical analysis of the dynamic decisions affecting product variety. *Decision Sciences* 46(6):1141–1164.

Williamson DP, Shmoys DB (2011) *The design of approximation algorithms* (Cambridge university press).

Wu T, Mao H, Li Y, Chen D (2019) Assortment selection for a frontend warehouse: A robust data-driven approach. *49th International Conference on Computers and Industrial Engineering (CIE 2019)*, 56–64.

Zhu S, Hu X, Huang K, Yuan Y (2021) Optimization of product category allocation in multiple warehouses to minimize splitting of online supermarket customer orders. *European journal of operational research* 290(2):556–571.

Electronic Companion for *Shall We Only Store Popular Products? Warehouse Assortment Selection for E-Companies*

EC.1. Proofs of Statements

Proof of Proposition 1 Proof of part (i): Consider any $\mathcal{S},\mathcal{S}'\subseteq\mathcal{N}$ with $\mathcal{S}\subseteq\mathcal{S}'$, we have

$$C(\mathcal{T}|\mathcal{S})=\begin{cases}0 & \text{if } \mathcal{T}\subseteq\mathcal{S}\\ G(\mathcal{T}) & \text{if } \mathcal{T}\nsubseteq\mathcal{S}\end{cases}=\begin{cases}0 & \text{if } \mathcal{T}\subseteq\mathcal{S}\\ G(\mathcal{T}) & \text{if } \mathcal{T}\nsubseteq\mathcal{S} \text{ and } \mathcal{T}\subseteq\mathcal{S}'\\ G(\mathcal{T}) & \text{if } \mathcal{T}\nsubseteq\mathcal{S}'\end{cases},$$

and

$$C(\mathcal{T}|\mathcal{S}')=\begin{cases}0 & \text{if } \mathcal{T}\subseteq\mathcal{S}\\ 0 & \text{if } \mathcal{T}\nsubseteq\mathcal{S} \text{ and } \mathcal{T}\subseteq\mathcal{S}'\\ G(\mathcal{T}) & \text{if } \mathcal{T}\nsubseteq\mathcal{S}'\end{cases}.$$

So, given $G(\mathcal{T})\geq 0\ \forall\mathcal{T}\subseteq\mathcal{N}$, we have $C(\mathcal{T}|\mathcal{S})\geq C(\mathcal{T}|\mathcal{S}')\ \forall\mathcal{T}\subseteq\mathcal{N}$.

Thus, given $\pi(\mathcal{T})\geq 0\ \forall\mathcal{T}\subseteq\mathcal{N}$, we have $f(\mathcal{S})=\sum_{\mathcal{T}\subseteq\mathcal{N}}\pi(\mathcal{T})C(\mathcal{T}|\mathcal{S})\geq\sum_{\mathcal{T}\subseteq\mathcal{N}}\pi(\mathcal{T})C(\mathcal{T}|\mathcal{S}')=f(\mathcal{S}')$. That is, the larger $\mathcal{S}$ is, the lower the total fulfillment costs. Since we want to minimize the cost, (CP') is equivalent to (CP).

Proof of part (ii): Consider any $\mathcal{S},\mathcal{S}'\subseteq\mathcal{N}$ with $\mathcal{S}\subseteq\mathcal{S}'$, we have

$$C(\mathcal{T}|\mathcal{S})=\begin{cases}0 & \text{if } \mathcal{T}\subseteq\mathcal{S}\\ G(\mathcal{T}\backslash\mathcal{S}) & \text{if } \mathcal{T}\nsubseteq\mathcal{S}\end{cases}=\begin{cases}0 & \text{if } \mathcal{T}\subseteq\mathcal{S}\\ G(\mathcal{T}\backslash\mathcal{S}) & \text{if } \mathcal{T}\nsubseteq\mathcal{S} \text{ and } \mathcal{T}\subseteq\mathcal{S}'\\ G(\mathcal{T}\backslash\mathcal{S}) & \text{if } \mathcal{T}\nsubseteq\mathcal{S}'\end{cases},$$

and

$$C(\mathcal{T}|\mathcal{S}')=\begin{cases}0 & \text{if } \mathcal{T}\subseteq\mathcal{S}\\ 0 & \text{if } \mathcal{T}\nsubseteq\mathcal{S} \text{ and } \mathcal{T}\subseteq\mathcal{S}'\\ G(\mathcal{T}\backslash\mathcal{S}') & \text{if } \mathcal{T}\nsubseteq\mathcal{S}'\end{cases}.$$

Since $(\mathcal{T}\backslash\mathcal{S}')\subseteq(\mathcal{T}\backslash\mathcal{S})\ \forall\mathcal{T}\subseteq\mathcal{N}$ and $G(X)\leq G(Y)\ \forall X\subseteq Y\subseteq\mathcal{N}$ (the type-II cost function is increasing), then we have $G(\mathcal{T}\backslash\mathcal{S})\geq G(\mathcal{T}\backslash\mathcal{S}')$. So, $C(\mathcal{T}|\mathcal{S})\geq C(\mathcal{T}|\mathcal{S}')\ \forall\mathcal{T}\subseteq\mathcal{N}$.

Thus, given $\pi(\mathcal{T})\geq 0\ \forall\mathcal{T}\subseteq\mathcal{N}$, we have $f(\mathcal{S})=\sum_{\mathcal{T}\subseteq\mathcal{N}}\pi(\mathcal{T})C(\mathcal{T}|\mathcal{S})\geq\sum_{\mathcal{T}\subseteq\mathcal{N}}\pi(\mathcal{T})C(\mathcal{T}|\mathcal{S}')=f(\mathcal{S}')$. That is, the larger $\mathcal{S}$ is, the lower the total fulfillment costs. Since we want to minimize the cost, problem (CP') is equivalent to problem (CP). □

Proof of Theorem 1 Proof of part (i): For type-I cost functions and any $\mathcal{S},\mathcal{S}'\subseteq\mathcal{N}$, take any set $\mathcal{T}\subseteq\mathcal{N}$. There are five cases: (a) $\mathcal{T}\subseteq\mathcal{S}'\cap\mathcal{S}$; (b) $\mathcal{T}\subseteq\mathcal{S}'$ but $\mathcal{T}\nsubseteq\mathcal{S}$; (c) $\mathcal{T}\nsubseteq\mathcal{S}'$ but $\mathcal{T}\subseteq\mathcal{S}$; (d) $\mathcal{T}\subseteq\mathcal{S}'\cup\mathcal{S}$ but $\mathcal{T}\nsubseteq\mathcal{S}'$ and $\mathcal{T}\nsubseteq\mathcal{S}$; and (e) $\mathcal{T}\nsubseteq\mathcal{S}'\cup\mathcal{S}$.

Under case (a) $C(\mathcal{T}|\mathcal{S}'\cap\mathcal{S})=C(\mathcal{T}|\mathcal{S}')=C(\mathcal{T}|\mathcal{S})=C(\mathcal{T}|\mathcal{S}'\cup\mathcal{S})=0$; under case (b) $C(\mathcal{T}|\mathcal{S}'\cap\mathcal{S})=C(\mathcal{T}|\mathcal{S})=G(\mathcal{T})$ and $C(\mathcal{T}|\mathcal{S}')=C(\mathcal{T}|\mathcal{S}'\cup\mathcal{S})=0$; under case (c) $C(\mathcal{T}|\mathcal{S}'\cap\mathcal{S})=C(\mathcal{T}|\mathcal{S}')=G(\mathcal{T})$ and $C(\mathcal{T}|\mathcal{S})=C(\mathcal{T}|\mathcal{S}'\cup\mathcal{S})=0$; under case (d) $C(\mathcal{T}|\mathcal{S}'\cap\mathcal{S})=C(\mathcal{T}|\mathcal{S}')=C(\mathcal{T}|\mathcal{S})=G(\mathcal{T})$ and $C(\mathcal{T}|\mathcal{S}'\cup\mathcal{S})=0$; and under case (e) $C(\mathcal{T}|\mathcal{S}'\cap\mathcal{S})=C(\mathcal{T}|\mathcal{S}')=C(\mathcal{T}|\mathcal{S})=C(\mathcal{T}|\mathcal{S}'\cup\mathcal{S})=G(\mathcal{T})$.

For cases (a)-(c) and (e), $C(\mathcal{T}|\mathcal{S}'\cap\mathcal{S})+C(\mathcal{T}|\mathcal{S}'\cup\mathcal{S})=C(\mathcal{T}|\mathcal{S}')+C(\mathcal{T}|\mathcal{S})$. For case (d) $C(\mathcal{T}|\mathcal{S}'\cap\mathcal{S})+C(\mathcal{T}|\mathcal{S}'\cap\mathcal{S})=G(\mathcal{T})\le 2G(\mathcal{T})=C(\mathcal{T}|\mathcal{S}')+C(\mathcal{T}|\mathcal{S})$. Thus, for any given $\mathcal{T}\subseteq\mathcal{N}$, $C(\mathcal{T}|\mathcal{S})$ is submodular in $\mathcal{S}$. Taking the weighted sum over all $\mathcal{T}$, we have $f(\mathcal{S})$ is submodular in $\mathcal{S}$.

Proof of part (ii): For type-II cost functions with $G(\cdot)$ being submodular and any $\mathcal{S},\mathcal{S}'\subseteq\mathcal{N}$, we have

$$\begin{aligned}
C(\mathcal{T}|\mathcal{S})+C(\mathcal{T}|\mathcal{S}') &= G(\mathcal{T}\backslash\mathcal{S})+G(\mathcal{T}\backslash\mathcal{S}')\\
&\ge G\left((\mathcal{T}\backslash\mathcal{S})\cap(\mathcal{T}\backslash\mathcal{S}')\right)+G\left((\mathcal{T}\backslash\mathcal{S})\cup(\mathcal{T}\backslash\mathcal{S}')\right)\\
&= G\left((\mathcal{T}\backslash(\mathcal{S}\cap\mathcal{S}')\right)+G\left((\mathcal{T}\backslash(\mathcal{S}\cup\mathcal{S}')\right)\\
&= C(\mathcal{T}|\mathcal{S}\cap\mathcal{S}')+C(\mathcal{T}|\mathcal{S}\cup\mathcal{S}').
\end{aligned}$$

Thus, $C(\mathcal{T}|\mathcal{S})$ is submodular in $\mathcal{S}$. Taking the weighted sum over all $\mathcal{T}$, we have $f(\mathcal{S})$ is submodular in $\mathcal{S}$.

If $G(\mathcal{S})$ is size-based, i.e., $G(\mathcal{S})=c(|\mathcal{S}|)\ \forall\mathcal{S}\subseteq\mathcal{N}$ for some function $c(\cdot)$, then $G(\mathcal{S}')+G(\mathcal{S})\ge G(\mathcal{S}'\cap\mathcal{S})+G(\mathcal{S}'\cup\mathcal{S})$ is equivalent to $c(|\mathcal{S}'|)+c(|\mathcal{S}|)\ge c(|\mathcal{S}'\cap\mathcal{S}|)+c(|\mathcal{S}'\cup\mathcal{S}|)$. Note that $|\mathcal{S}'|+|\mathcal{S}|=|\mathcal{S}'\cap\mathcal{S}|+|\mathcal{S}'\cup\mathcal{S}|$ and $|\mathcal{S}'\cap\mathcal{S}|\le|\mathcal{S}'|,|\mathcal{S}|\le|\mathcal{S}'\cup\mathcal{S}|$. Hence, $c(|\mathcal{S}'|)+c(|\mathcal{S}|)\ge c(|\mathcal{S}'\cap\mathcal{S}|)+c(|\mathcal{S}'\cup\mathcal{S}|)$ is equivalent to $c(\cdot)$ is concave. Thus, $G(\mathcal{S})$ is submodular is equivalent to $c(\cdot)$ is concave. □

Proof of Corollary 1 In terms of (OFRM-C), let $f(\mathcal{S})=\sum_{m=1}^{M}\pi_m\mathbb{I}(\mathcal{T}_m\nsubseteq\mathcal{S})$. It is obvious that the objective function in (OFRM-C) is decreasing and submodular. In terms of (OFRM), let $\mathrm{OFR}(\mathcal{S})=\sum_{m=1}^{M}\pi_m\mathbb{I}(\mathcal{T}_m\subseteq\mathcal{S})=1-f(\mathcal{S})$. On the one hand, for any $\mathcal{S},\mathcal{S}'\subseteq\mathcal{N}$ with $\mathcal{S}\subseteq\mathcal{S}'$, we have

$$C(\mathcal{T}|\mathcal{S})=\begin{cases}0 \text{ if } \mathcal{T}\subseteq\mathcal{S}\\ 1 \text{ if } \mathcal{T}\nsubseteq\mathcal{S}\end{cases}=\begin{cases}0 \text{ if } \mathcal{T}\subseteq\mathcal{S}\\ 1 \text{ if } \mathcal{T}\nsubseteq\mathcal{S} \text{ and } \mathcal{T}\subseteq\mathcal{S}'\\ 1 \text{ if } \mathcal{T}\nsubseteq\mathcal{S}'\end{cases},$$

and

$$C(\mathcal{T}|\mathcal{S}')=\begin{cases}0 \text{ if } \mathcal{T}\subseteq\mathcal{S}\\ 0 \text{ if } \mathcal{T}\nsubseteq\mathcal{S} \text{ and } \mathcal{T}\subseteq\mathcal{S}'\\ 1 \text{ if } \mathcal{T}\nsubseteq\mathcal{S}'\end{cases}.$$

Hence, we have $C(\mathcal{T}|\mathcal{S})\ge C(\mathcal{T}|\mathcal{S}')\ \forall\mathcal{T}\subseteq\mathcal{N}$. So, given $\pi(\mathcal{T})\ge 0\ \forall\mathcal{T}\subseteq\mathcal{N}$, we have $\mathrm{OFR}(\mathcal{S})=1-f(\mathcal{S})=1-\sum_{\mathcal{T}\subseteq\mathcal{N}}\pi(\mathcal{T})C(\mathcal{T}|\mathcal{S})\le 1-\sum_{\mathcal{T}\subseteq\mathcal{N}}\pi(\mathcal{T})C(\mathcal{T}|\mathcal{S}')=1-f(\mathcal{S}')=\mathrm{OFR}(\mathcal{S}')$. That is, $\mathrm{OFR}(\cdot)$ is increasing.

On the other hand, for any $\mathcal{S},\mathcal{S}'\subseteq\mathcal{N}$, take any set $\mathcal{T}\subseteq\mathcal{N}$. There are five cases: (a) $\mathcal{T}\subseteq\mathcal{S}'\cap\mathcal{S}$; (b) $\mathcal{T}\subseteq\mathcal{S}'$ but $\mathcal{T}\nsubseteq\mathcal{S}$; (c) $\mathcal{T}\nsubseteq\mathcal{S}'$ but $\mathcal{T}\subseteq\mathcal{S}$; (d) $\mathcal{T}\subseteq\mathcal{S}'\cup\mathcal{S}$ but $\mathcal{T}\nsubseteq\mathcal{S}'$ and $\mathcal{T}\nsubseteq\mathcal{S}$; and (e) $\mathcal{T}\nsubseteq\mathcal{S}'\cup\mathcal{S}$. Under case (a) $C(\mathcal{T}|\mathcal{S}'\cap\mathcal{S})=$

$C(\mathcal{T}|\mathcal{S}') = C(\mathcal{T}|\mathcal{S}) = C(\mathcal{T}|\mathcal{S}' \cup \mathcal{S}) = 0$; under case (b) $C(\mathcal{T}|\mathcal{S}' \cap \mathcal{S}) = C(\mathcal{T}|\mathcal{S}) = 1$ and $C(\mathcal{T}|\mathcal{S}') = C(\mathcal{T}|\mathcal{S}' \cup \mathcal{S}) = 0$; under case (c) $C(\mathcal{T}|\mathcal{S}' \cap \mathcal{S}) = C(\mathcal{T}|\mathcal{S}') = 1$ and $C(\mathcal{T}|\mathcal{S}) = C(\mathcal{T}|\mathcal{S}' \cup \mathcal{S}) = 0$; under case (d) $C(\mathcal{T}|\mathcal{S}' \cap \mathcal{S}) = C(\mathcal{T}|\mathcal{S}') = C(\mathcal{T}|\mathcal{S}) = 1$ and $C(\mathcal{T}|\mathcal{S}' \cup \mathcal{S}) = 0$; and under case (e) $C(\mathcal{T}|\mathcal{S}' \cap \mathcal{S}) = C(\mathcal{T}|\mathcal{S}') = C(\mathcal{T}|\mathcal{S}) = C(\mathcal{T}|\mathcal{S}' \cup \mathcal{S}) = 1$. For cases (a)-(c) and (e), $C(\mathcal{T}|\mathcal{S}' \cap \mathcal{S}) + C(\mathcal{T}|\mathcal{S}' \cup \mathcal{S}) = C(\mathcal{T}|\mathcal{S}') + C(\mathcal{T}|\mathcal{S})$. For case (d) $C(\mathcal{T}|\mathcal{S}' \cap \mathcal{S}) + C(\mathcal{T}|\mathcal{S}' \cap \mathcal{S}) = 1 \leq 2 = C(\mathcal{T}|\mathcal{S}') + C(\mathcal{T}|\mathcal{S})$. Hence, for any given $\mathcal{T} \subseteq \mathcal{N}$, $C(\mathcal{T}|\mathcal{S})$ is submodular in $\mathcal{S}$. So, $1 - C(\mathcal{T}|\mathcal{S})$ is supermodular in $\mathcal{S}$. Taking the weighted sum over all $\mathcal{T} \subseteq \mathcal{N}$, we have $\text{OFR}(\mathcal{S})$ is supermodular in $\mathcal{S}$. □

Proof of Theorem 2 We prove by a reduction from the Densest k-Subgraph (DkS) Problem, which is known to be NP-hard (Corneil and Perl 1984).

We start by presenting the DkS problem. Consider a undirected simple graph $G(V, E)$ with $V = \{1, 2, \ldots, N\}$, $E \subseteq \{\{i, j\} | i, j \in V, i \neq j\}$ (loops are permitted), and $|E| = m$. Given a parameter k, the goal of the DkS problem is to find a subgraph of G induced on k vertices that contains the largest number of edges.

Let A be the adjacency matrix of G, then the DkS problem can be formulated as follows

$$\begin{aligned} \max_{\boldsymbol{x}} \ & \boldsymbol{x}^\top A \boldsymbol{x} \\ \text{s.t.} \ & \textstyle\sum_{i=1}^N x_i = k \\ & x_i = \{0, 1\} \quad i \in \{1, 2, \ldots, N\}, \end{aligned} \tag{EC.1}$$

where x_i a binary decision variable which equals to 1 if vertex i is in the densest k-subgraph and 0 otherwise.

Now, we construct an instance of the (OFRM) problem with $L = 2$ and cardinality constraint k. Assume that there are N different items and let $\mathcal{N} = \{1, 2, \ldots, N\}$. For any $i, j \in \mathcal{N}$, subset $\{i, j\}$ has a probability of $\frac{1}{m}$ to be chosen if $\{i, j\} \in E$, and the rest subsets are never been chosen. Let $B = \{b_{i,j}\}_{N \times N}$ with $b_{i,j} = \frac{1}{2m}$ $\forall \{i, j\} \in E$, and 0 otherwise, then the (OFRM) problem with $L = 2$ for this instance can be formulated as follows.

$$\begin{aligned} \max_{\boldsymbol{x}} \ & \boldsymbol{x}^\top B \boldsymbol{x} \\ \text{s.t.} \ & \textstyle\sum_{i=1}^N x_i \leq K \\ & x_i = \{0, 1\} \quad i \in \mathcal{N}, \end{aligned} \tag{EC.2}$$

where x_i a binary decision variable which equals to 1 if product i is selected in assortment $\mathcal{S}$ and 0 otherwise. Moreover, according to Proposition 1, Problem (EC.2) is equivalent to

$$\begin{aligned} \max_{\boldsymbol{x}} \ & \tfrac{1}{2m} \cdot \boldsymbol{x}^\top A \boldsymbol{x} \\ \text{s.t.} \ & \textstyle\sum_{i=1}^N x_i = K \\ & x_i = \{0, 1\} \quad i \in \mathcal{N}. \end{aligned} \tag{EC.3}$$

Since when $K = k$, Problem (EC.3) is equivalent to Problem (EC.1). Thus, we conclude that the (OFRM) problem with $L = 2$ is NP-hard.

Furthermore, by the equivalence of the (OFRM) problem and the (OFRM-C) problem, we also conclude that the (OFRM-C) problem with $L = 2$ is NP-hard. □

Proof of Proposition 2 We will show that Problem (5) can be reduced to a knapsack problem, which can be solved using dynamic programming.

We start by presenting the common $0-1$ knapsack problem. Assume that there are n items with $v_1,\ldots,v_n$ being the associated item values and $w_1,\ldots,w_n$ being the associated item sizes, where item sizes are assumed to be positive integers. The goal of the knapsack problem is to find a subset of items such that the total value (of the subset of items) is maximized and the total size (of the subset of items) does not exceed the knapsack size $W\in\mathbb{N}^+$. We assume that the items are indexed such that $v_1\le v_2\le\cdots\le v_n$. Let $\boldsymbol{x}=[x_1,\ldots,x_n]^\mathcal{T}$ be the decision variable with $x_i=1$ if item i is included in the interested subset and 0 otherwise. Then, the knapsack problem is formulated as follows.

$$\begin{aligned}\max_{\boldsymbol{x}\in\{0,1\}^n}\ &\sum_{i=1}^{n} v_i x_i\\ \text{s.t.}\ &\sum_{i=1}^{n} w_i x_i\le W.\end{aligned}\tag{EC.4}$$

Now, we construct an instance of the knapsack problem with $n=\tilde{M}$ items. We let $v_i=\pi_i$ and $w_i=s_i\ \forall i\in[\tilde{M}]$ be the item value and item size, respectively. Additionally, we let the knapsack size $W=K$. So, compare both problem formulations, solving Problem (5) is equivalent to solve the constructed knapsack Problem (EC.4).

Generally, the $0-1$ knapsack problem is (binary) NP-complete (Williamson and Shmoys 2011) and can be solved via a dynamic programming algorithm (Algorithm 1).

Algorithm 1: Dynamic Programming Algorithm for $0-1$ Knapsack Problem

Input : $v_1,\ldots,v_n,\ w_1,\ldots,w_n,\ W$

Output: $m_{n,W}$

$m_{i,0}\leftarrow 0\ \forall i\in[n]$;
$m_{0,j}\leftarrow 0\ \forall j\in[W]$;
for $i\leftarrow 1$ *to* n *do* **do**
  for $j\leftarrow 1$ *to* W *do* **do**
    if $w_i\le j$ **then**
      $m_{i,j}=\max\{m_{i-1,j-w_i}+v_i,\ m_{i-1,j}\}$;
    else
      $m_{i,j}=m_{i-1,j}$;
    end
  end
end
return $m_{n,W}$

The computational complexity of Algorithm 1 is $\mathcal{O}(nW)$, which means the general $0-1$ knapsack problem can be solved in pseudo-polynomial time (given that W can be arbitrary large). However, in our special case, $n=\tilde{M}\leq N$ and $W=K\leq N$ is capped by the total number of products, the constructed $0-1$ knapsack problem can be solved in run time $\mathcal{O}(NK)$. Thus, Problem (5) can be solved in polynomial time using dynamic programming Algorithm 1.

□

Proof of Proposition 3 Without loss of generality, we assume $c(n)=n$. Then, problem (CP) is equivalent to the following integer programming

$$\begin{aligned}\max_{\boldsymbol{\xi}}\ & \textstyle\sum_{n=1}^{N}\left(\sum_{m:\mathcal{T}_m\ni n}\pi_m\right)\xi_n\\ \text{s.t.}\ & \sum_{n=1}^{N}\xi_n\leq K\\ & \xi_n\in\{0,1\} \qquad\qquad \forall n\in[N],\end{aligned} \tag{EC.5}$$

where $\boldsymbol{\xi}\in\{0,1\}^N$ denotes the binary decision variable with $\xi_n=1$ if product n is included in the assortment $\mathcal{S}$ and 0 otherwise. Problem (EC.5) is a knapsack problem with the same weight, as a result, the optimal policy is to select K ξ_ns with the largest $\sum_{m:\mathcal{T}_m\ni n}\pi_m$ among the N products, setting them to be 1. Note that $\sum_{m:\mathcal{T}_m\ni n}\pi_m$ is the marginal choice probability of product n. Thus, the MCI policy is optimal. □

Proof of Proposition 4 Considering type-II cost function with $G(\mathcal{T})=\sum_{n\in\mathcal{T}}\kappa_n$, (CP) can be reformulated as the following integer programming

$$\begin{aligned}\max_{\boldsymbol{\xi}}\ & \textstyle\sum_{n=1}^{N}\left(\sum_{m:\mathcal{T}_m\ni n}\pi_m\right)\kappa_n\xi_n\\ \text{s.t.}\ & \sum_{n=1}^{N}\xi_n\leq K\\ & \xi_n\in\{0,1\} \qquad\qquad \forall n\in[N],\end{aligned} \tag{EC.6}$$

where $\boldsymbol{\xi}\in\{0,1\}^N$ denotes the binary decision variable with $\xi_n=1$ if product n is included in the assortment $\mathcal{S}$ and 0 otherwise. Problem (EC.6) is a knapsack problem with the same weight, as a result, the optimal policy is to select K ξ_ns with the largest $\left(\sum_{m:\mathcal{T}_m\ni n}\pi_m\right)\kappa_n$ among the N products, setting them to be 1. Note that $\left(\sum_{m:\mathcal{T}_m\ni n}\pi_m\right)\kappa_n=\omega_n\kappa_n$ is the choice-weighted fulfillment cost of product n. Thus, the modified MCI policy described in Proposition 4 is optimal. □

Proof of Theorem 3 Without loss of generality, we assume the elements in $\mathcal{N}$ are already indexed according to $\mathcal{I}$. For the first half, we need to show that $\omega_i\geq\omega_j$ for all $i,j\in\mathcal{N}$ with $i<j$. Consider arbitrary $i,j\in\mathcal{N}$ with $i<j$. Since the elements in $\mathcal{N}$ are indexed according to $\mathcal{I}$, which is a dominant indexing rule w.r.t. the choice model π, then for any $\mathcal{T}\subseteq\mathcal{N}$, we have $\pi(\mathcal{T}\cup\{i\})\geq\pi(\mathcal{T}\cup\{j\})$. So,

$$\begin{aligned}\omega_i&=\sum_{\mathcal{T}\subseteq\mathcal{N}}\pi(\mathcal{T})\cdot\mathbb{I}(i\in\mathcal{T})=\sum_{\mathcal{T}\subseteq\mathcal{N}\setminus\{i\}}\pi(\mathcal{T}\cup\{i\})=\sum_{\mathcal{T}\subseteq\mathcal{N}\setminus\{i,j\}}\pi(\mathcal{T}\cup\{i\})+\pi(\mathcal{T}\cup\{i,j\})\\ &\geq\sum_{\mathcal{T}\subseteq\mathcal{N}\setminus\{i,j\}}\pi(\mathcal{T}\cup\{j\})+\pi(\mathcal{T}\cup\{i,j\})=\sum_{\mathcal{T}\subseteq\mathcal{N}\setminus\{j\}}\pi(\mathcal{T}\cup\{j\})=\omega_j.\end{aligned}$$

Thus, if a dominant indexing rule exists, then it is an MCI.

For the second half, we need to show that $\mathcal{S}^*=\{1,2,\ldots,K\}$ is the optimal solution.

(i) According to part 1 of Proposition 1, problem (CP') is equivalent to problem (CP) under type-I size-based cost functions for any choice model. So, we only have to prove $\mathcal{S}^*$ is an optimal solution for problem (CP').

We prove the proposition by constructing a contradiction. Assume that $\mathcal{S}^0 = \{1, 2, \ldots, l-1, m, l+1, \ldots, K\}$ such that $f(\mathcal{S}^0) < f(\mathcal{S}^*)$ for some $l \in \{1, \ldots, K\}$ and $m \in \{K+1, \ldots, N\}$.

Let $\tilde{\mathcal{S}} = \{1, \ldots, l-1, l+1, \ldots, K\}$. Then, we have $\mathcal{S}^* = \tilde{\mathcal{S}} \cup \{l\}$ and $\mathcal{S}^0 = \tilde{\mathcal{S}} \cup \{m\}$. So,

$$\begin{aligned} f(\mathcal{S}^*) &= \sum_{\mathcal{T} \subseteq \mathcal{N}} \pi(\mathcal{T}) C(\mathcal{T}|\mathcal{S}^*) = \sum_{\mathcal{T} \subseteq \mathcal{N}, \mathcal{T} \nsubseteq \mathcal{S}^*} \pi(\mathcal{T}) c(|\mathcal{T}|) = \sum_{n=1}^{N} \left(\sum_{\mathcal{T} \subseteq \mathcal{N}, \mathcal{T} \nsubseteq \mathcal{S}^*, |\mathcal{T}|=n} \pi(\mathcal{T}) c(n) \right) \\ &= \sum_{n=1}^{N} c(n) \left(1 - \sum_{\mathcal{T} \subseteq \mathcal{S}^*, |\mathcal{T}|=n} \pi(\mathcal{T}) \right) = \sum_{n=1}^{N} c(n) \left(1 - \sum_{\mathcal{T} \subseteq \tilde{\mathcal{S}}, |\mathcal{T}|=n} \pi(\mathcal{T}) - \sum_{\mathcal{T} \subseteq \tilde{\mathcal{S}}, |\mathcal{T}|=n-1} \pi(\mathcal{T} \cup \{l\}) \right), \end{aligned}$$

and

$$\begin{aligned} f(\mathcal{S}^0) &= \sum_{\mathcal{T} \subseteq \mathcal{N}} \pi(\mathcal{T}) C(\mathcal{T}|\mathcal{S}^0) = \sum_{\mathcal{T} \subseteq \mathcal{N}, \mathcal{T} \nsubseteq \mathcal{S}^0} \pi(\mathcal{T}) c(|\mathcal{T}|) = \sum_{n=1}^{N} c(n) \left(1 - \sum_{\mathcal{T} \subseteq \mathcal{S}^0, |\mathcal{T}|=n} \pi(\mathcal{T}) \right) \\ &= \sum_{n=1}^{N} c(n) \left(1 - \sum_{\mathcal{T} \subseteq \tilde{\mathcal{S}}, |\mathcal{T}|=n} \pi(\mathcal{T}) - \sum_{\mathcal{T} \subseteq \tilde{\mathcal{S}}, |\mathcal{T}|=n-1} \pi(\mathcal{T} \cup \{m\}) \right). \end{aligned}$$

Since $\mathcal{I}$ is a dominant indexing rule w.r.t. the choice model π, for any $\mathcal{T} \subseteq \tilde{\mathcal{S}}$, we have $\pi(\mathcal{T} \cup \{l\}) \geq \pi(\mathcal{T} \cup \{m\})$. So, $f(\mathcal{S}^0) \geq f(\mathcal{S}^*)$. It contradicts the assumption. In a similar sense, we can iteratively replace elements in $\{1, \ldots, K\}$ with elements in $\{K+1, \ldots, N\}$ and show that selecting $\mathcal{S}^* = \{1, \ldots, K\}$ has the lower cost compared to that of any other $\mathcal{S}^0 \subseteq \mathcal{N}$ with $|\mathcal{S}^0| = K$.

Thus, $f(\mathcal{S}^*) \leq f(\mathcal{S})$ $\forall \mathcal{S} \subseteq \mathcal{N}$ with $|\mathcal{S}| = K$. This proves that $\mathcal{S}^*$ is optimal.

(ii) According to part 2 of Proposition 1, problem (CP') is equivalent to problem (CP) under increasing type-II size-based cost functions for any choice model. So, we only have to prove $\mathcal{S}^*$ is an optimal solution for problem (CP').

We prove this by constructing a contradiction. Assume that $\mathcal{S}^0 = \{1, 2, \ldots, l-1, m, l+1, \ldots, K\}$ such that $f(\mathcal{S}^0) < f(\mathcal{S}^*)$ for some $l \in \{1, \ldots, K\}$ and $m \in \{K+1, \ldots, N\}$.

Let $\tilde{\mathcal{S}} = \{1, \ldots, l-1, l+1, \ldots, K\}$, $\bar{\mathcal{S}} = \{K+1, \ldots, m-1, m+1, \ldots, N\}$. Then, we have $\mathcal{S}^* = \tilde{\mathcal{S}} \cup \{l\}$ and $\mathcal{S}^0 = \tilde{\mathcal{S}} \cup \{m\}$. Additionally, we let $\bar{\mathcal{S}}^* = \bar{\mathcal{S}} \cup \{m\}$ and $\bar{\mathcal{S}}^0 = \bar{\mathcal{S}} \cup \{l\}$. So,

$$
\begin{aligned}
f(\mathcal{S}^*) &= \sum_{\mathcal{T}\subseteq\mathcal{N}} \pi(\mathcal{T}) C(\mathcal{T}|\mathcal{S}^*) = \sum_{\mathcal{T}\subseteq\mathcal{N}, \mathcal{T}\nsubseteq\mathcal{S}^*} \pi(\mathcal{T}) c(|\mathcal{T}\backslash\mathcal{S}^*|) \\
&= \sum_{n=1}^{N-K} \left(\sum_{\mathcal{T}\subseteq\mathcal{N}, \mathcal{T}\nsubseteq\mathcal{S}^*, |\mathcal{T}\backslash\mathcal{S}^*|=n} \pi(\mathcal{T}) c(n) \right) = \sum_{n=1}^{N-K} c(n) \left(\sum_{X\subseteq\mathcal{S}^*} \sum_{Y\subseteq\bar{\mathcal{S}}^*, |Y|=n} \pi(X\cup Y) \right) \\
&= \sum_{n=1}^{N-K} c(n) \left(\sum_{X\subseteq\tilde{\mathcal{S}}} \sum_{Y\subseteq\bar{\mathcal{S}}^*, |Y|=n} \pi(X\cup Y) + \sum_{X\subseteq\tilde{\mathcal{S}}} \sum_{Y\subseteq\bar{\mathcal{S}}^*, |Y|=n} \pi(X\cup Y\cup\{l\}) \right) \\
&= \sum_{n=1}^{N-K-1} c(n) \sum_{X\subseteq\tilde{\mathcal{S}}} \sum_{Y\subseteq\bar{\mathcal{S}}, |Y|=n} \pi(X\cup Y) + \sum_{n=1}^{N-K} c(n) \sum_{X\subseteq\tilde{\mathcal{S}}} \sum_{Y\subseteq\bar{\mathcal{S}}, |Y|=n-1} \pi(X\cup Y\cup\{m\}) \\
&\quad + \sum_{n=1}^{N-K-1} c(n) \sum_{X\subseteq\tilde{\mathcal{S}}} \sum_{Y\subseteq\bar{\mathcal{S}}, |Y|=n} \pi(X\cup Y\cup\{l\}) + \sum_{n=1}^{N-K} c(n) \sum_{X\subseteq\tilde{\mathcal{S}}} \sum_{Y\subseteq\bar{\mathcal{S}}, |Y|=n-1} \pi(X\cup Y\cup\{l\}\cup\{m\}) \\
&= \sum_{n=1}^{N-K-1} c(n) \sum_{X\subseteq\tilde{\mathcal{S}}} \sum_{Y\subseteq\bar{\mathcal{S}}, |Y|=n} \pi(X\cup Y) + \sum_{n=1}^{N-K} c(n) \sum_{X\subseteq\tilde{\mathcal{S}}} \sum_{Y\subseteq\bar{\mathcal{S}}, |Y|=n-1} \pi(X\cup Y\cup\{l\}\cup\{m\}) \\
&\quad + \sum_{n=1}^{N-K-1} c(n) \sum_{X\subseteq\tilde{\mathcal{S}}} \sum_{Y\subseteq\bar{\mathcal{S}}, |Y|=n} \pi(X\cup Y\cup\{l\}) + \sum_{n=0}^{N-K-1} c(n+1) \sum_{X\subseteq\tilde{\mathcal{S}}} \sum_{Y\subseteq\bar{\mathcal{S}}, |Y|=n} \pi(X\cup Y\cup\{m\}),
\end{aligned}
$$

and

$$
\begin{aligned}
f(\mathcal{S}^0) &= \sum_{\mathcal{T}\subseteq\mathcal{N}} \pi(\mathcal{T}) C(\mathcal{T}|\mathcal{S}^0) = \sum_{\mathcal{T}\subseteq\mathcal{N}, \mathcal{T}\nsubseteq\mathcal{S}^0} \pi(\mathcal{T}) c(|\mathcal{T}\backslash\mathcal{S}^0|) \\
&= \sum_{n=1}^{N-K} \left(\sum_{\mathcal{T}\subseteq\mathcal{N}, \mathcal{T}\nsubseteq\mathcal{S}^0, |\mathcal{T}\backslash\mathcal{S}^0|=n} \pi(\mathcal{T}) c(n) \right) = \sum_{n=1}^{N-K} c(n) \left(\sum_{X\subseteq\mathcal{S}^0} \sum_{Y\subseteq\bar{\mathcal{S}}^0, |Y|=n} \pi(X\cup Y) \right) \\
&= \sum_{n=1}^{N-K} c(n) \left(\sum_{X\subseteq\tilde{\mathcal{S}}} \sum_{Y\subseteq\bar{\mathcal{S}}^0, |Y|=n} \pi(X\cup Y) + \sum_{X\subseteq\tilde{\mathcal{S}}} \sum_{Y\subseteq\bar{\mathcal{S}}^0, |Y|=n} \pi(X\cup Y\cup\{m\}) \right) \\
&= \sum_{n=1}^{N-K-1} c(n) \sum_{X\subseteq\tilde{\mathcal{S}}} \sum_{Y\subseteq\bar{\mathcal{S}}, |Y|=n} \pi(X\cup Y) + \sum_{n=1}^{N-K} c(n) \sum_{X\subseteq\tilde{\mathcal{S}}} \sum_{Y\subseteq\bar{\mathcal{S}}, |Y|=n-1} \pi(X\cup Y\cup\{l\}) \\
&\quad + \sum_{n=1}^{N-K-1} c(n) \sum_{X\subseteq\tilde{\mathcal{S}}} \sum_{Y\subseteq\bar{\mathcal{S}}, |Y|=n} \pi(X\cup Y\cup\{m\}) + \sum_{n=1}^{N-K} c(n) \sum_{X\subseteq\tilde{\mathcal{S}}} \sum_{Y\subseteq\bar{\mathcal{S}}, |Y|=n-1} \pi(X\cup Y\cup\{m\}\cup\{l\}) \\
&= \sum_{n=1}^{N-K-1} c(n) \sum_{X\subseteq\tilde{\mathcal{S}}} \sum_{Y\subseteq\bar{\mathcal{S}}, |Y|=n} \pi(X\cup Y) + \sum_{n=1}^{N-K} c(n) \sum_{X\subseteq\tilde{\mathcal{S}}} \sum_{Y\subseteq\bar{\mathcal{S}}, |Y|=n-1} \pi(X\cup Y\cup\{m\}\cup\{l\}) \\
&\quad + \sum_{n=1}^{N-K-1} c(n) \sum_{X\subseteq\tilde{\mathcal{S}}} \sum_{Y\subseteq\bar{\mathcal{S}}, |Y|=n} \pi(X\cup Y\cup\{m\}) + \sum_{n=0}^{N-K-1} c(n+1) \sum_{X\subseteq\tilde{\mathcal{S}}} \sum_{Y\subseteq\bar{\mathcal{S}}, |Y|=n} \pi(X\cup Y\cup\{l\}).
\end{aligned}
$$

Since $\mathcal{I}$ is dominant w.r.t. π, then for any $X \subseteq \tilde{\mathcal{S}}$ and $Y \subseteq \bar{\mathcal{S}}$, we have $\pi(X \cup Y \cup \{l\}) \geq \pi(X \cup Y \cup \{m\})$. So,

$$
\begin{aligned}
& f(\mathcal{S}^*) - f(\mathcal{S}^0) \\
= & \sum_{n=1}^{N-K-1} c(n) \sum_{X \subseteq \tilde{\mathcal{S}}} \sum_{Y \subseteq \bar{\mathcal{S}}, |Y|=n} \pi(X \cup Y \cup \{l\}) + \sum_{n=0}^{N-K-1} c(n+1) \sum_{X \subseteq \tilde{\mathcal{S}}} \sum_{Y \subseteq \bar{\mathcal{S}}, |Y|=n} \pi(X \cup Y \cup \{m\}) \\
& - \sum_{n=1}^{N-K-1} c(n) \sum_{X \subseteq \tilde{\mathcal{S}}} \sum_{Y \subseteq \bar{\mathcal{S}}, |Y|=n} \pi(X \cup Y \cup \{m\}) - \sum_{n=0}^{N-K-1} c(n+1) \sum_{X \subseteq \tilde{\mathcal{S}}} \sum_{Y \subseteq \bar{\mathcal{S}}, |Y|=n} \pi(X \cup Y \cup \{l\}) \\
= & \sum_{n=0}^{N-K-1} c(n) \sum_{X \subseteq \tilde{\mathcal{S}}} \sum_{Y \subseteq \bar{\mathcal{S}}, |Y|=n} \pi(X \cup Y \cup \{l\}) + \sum_{n=0}^{N-K-1} c(n+1) \sum_{X \subseteq \tilde{\mathcal{S}}} \sum_{Y \subseteq \bar{\mathcal{S}}, |Y|=n} \pi(X \cup Y \cup \{m\}) \\
& - \sum_{n=0}^{N-K-1} c(n) \sum_{X \subseteq \tilde{\mathcal{S}}} \sum_{Y \subseteq \bar{\mathcal{S}}, |Y|=n} \pi(X \cup Y \cup \{m\}) - \sum_{n=0}^{N-K-1} c(n+1) \sum_{X \subseteq \tilde{\mathcal{S}}} \sum_{Y \subseteq \bar{\mathcal{S}}, |Y|=n} \pi(X \cup Y \cup \{l\}) \\
& - c(0) \sum_{X \subseteq \tilde{\mathcal{S}}} \sum_{Y \subseteq \bar{\mathcal{S}}, |Y|=0} \pi(X \cup Y \cup \{l\}) + c(0) \sum_{X \subseteq \tilde{\mathcal{S}}} \sum_{Y \subseteq \bar{\mathcal{S}}, |Y|=0} \pi(X \cup Y \cup \{m\}) \\
= & \sum_{n=0}^{N-K-1} [c(n) - c(n+1)] \left[\sum_{X \subseteq \tilde{\mathcal{S}}} \sum_{Y \subseteq \bar{\mathcal{S}}, |Y|=n} (\pi(X \cup Y \cup \{l\}) - \pi(X \cup Y \cup \{m\})) \right] \\
\leq & \ 0.
\end{aligned}
$$

It contradicts the assumption. In a similar sense, we can iteratively replace elements in $\{1, \ldots, K\}$ with elements in $\{K+1, \ldots, N\}$ and show that selecting $\mathcal{S}^* = \{1, \ldots, K\}$ has the lower cost compared to that of any other $\mathcal{S}^0 \subseteq \mathcal{N}$ with $|\mathcal{S}^0| = K$.

Thus, $f(\mathcal{S}^*) \leq f(\mathcal{S})$ $\forall \mathcal{S} \subseteq \mathcal{N}$ with $|\mathcal{S}| = K$. This proves that $\mathcal{S}^*$ is optimal. $\square$

Proof of Proposition 6 For the first half, we claim that indexing products such that $p_1 \geq p_2 \geq \cdots \geq p_N$ is dominant w.r.t. the ICM.

Since $\omega_n = \sum_{\mathcal{T} \subseteq \mathcal{N}} \pi_{ICM}(\mathcal{T}) \cdot \mathbb{I}(n \in \mathcal{T}) = p_n$ $\forall n \in \mathcal{N}$, then such indexing rule is exactly an MCI. For any subset $\mathcal{T} \subseteq \mathcal{N}$ and two distinct elements $l, m \in \mathcal{N} \backslash \mathcal{T}$ with $l < m$. Let $\mathcal{T}' = \mathcal{T} \cup \{l\}$ and $\mathcal{T}'' = \mathcal{T} \cup \{m\}$. Then, we have $p_l \geq p_m$, $1 - p_m \geq 1 - p_l$, and

$$
\begin{aligned}
\pi_{ICM}(\mathcal{T}') &= \pi_{ICM}(\mathcal{T} \cup \{l\}) \\
&= p_l \textstyle\prod_{i \in \mathcal{T}} p_i \prod_{j \in \mathcal{N} \backslash \mathcal{T}, j \neq l} (1 - p_j) \\
&= p_l (1 - p_m) \textstyle\prod_{i \in \mathcal{T}} p_i \prod_{j \in \mathcal{N} \backslash \mathcal{T}, j \neq l, j \neq m} (1 - p_j) \\
&\geq p_m (1 - p_l) \textstyle\prod_{i \in \mathcal{T}} p_i \prod_{j \in \mathcal{N} \backslash \mathcal{T}, j \neq l, j \neq m} (1 - p_j) \\
&= \pi_{ICM}(\mathcal{T} \cup \{m\}) = \pi_{ICM}(\mathcal{T}'').
\end{aligned}
$$

In a similar sense, we can iteratively pick a pair of elements (i_l, i_m) with $i_l, i_m \in \mathcal{N} \backslash \mathcal{T}$ and $i_l < i_m$, and verify that $\pi_{ICM}(\mathcal{T}' \cup \{i_l\}) \geq \pi_{ICM}(\mathcal{T}'' \cup \{i_m\})$. Thus, we conclude this indexing rule is dominant w.r.t the ICM.

For the second half, it holds directly from Theorem 3 as we find the MCI is dominant w.r.t. the ICM. $\square$

Proof of Proposition 7 For the first half, we claim that indexing products such that $U_1 \succeq_1 U_2 \succeq_1 \cdots \succeq_1 U_N$ is dominant w.r.t. the MC-RIUM.

The choice probability of a set $\mathcal{T} \subseteq \mathcal{N}$ under MC-RIUM can be represented as

$$\begin{aligned}
\pi_{MC-RIUM}(\mathcal{T}) &= \mathbb{P}\left(Q = |\mathcal{T}|, \min_{i\in\mathcal{T}} U_i > \max_{j\in\mathcal{N}^+\setminus\mathcal{T}} U_j\right) + \mathbb{P}\left(Q > |\mathcal{T}|, \min_{i\in\mathcal{T}} U_i > U_0, U_0 > \max_{k\in\mathcal{N}\setminus\mathcal{T}} U_k\right) \\
&= \mathbb{P}\left(Q = |\mathcal{T}|\right) \cdot \mathbb{P}\left(\min_{i\in\mathcal{T}} U_i > \max_{j\in\mathcal{N}^+\setminus\mathcal{T}} U_j\right) + \mathbb{P}\left(Q > |\mathcal{T}|\right) \cdot \mathbb{P}\left(\min_{i\in\mathcal{T}} U_i > U_0, U_0 > \max_{k\in\mathcal{N}\setminus\mathcal{T}} U_k\right) \\
&= \mathbb{P}(Q = |\mathcal{T}|) \cdot \mathbb{P}\left(\mathcal{T} = \operatorname*{arg\,max}_{\mathcal{S}\subseteq\mathcal{N}^+, |\mathcal{S}|=|\mathcal{T}|} \sum_{i\in\mathcal{S}} U_i\right) \\
&\quad + \mathbb{P}\left(Q > |\mathcal{T}|\right) \cdot \mathbb{P}\left(\mathcal{T} = \operatorname*{arg\,max}_{\mathcal{S}\subseteq\mathcal{N}^+, |\mathcal{S}|=|\mathcal{T}|} \sum_{i\in\mathcal{S}} U_i, \mathcal{T}\cup\{0\} = \operatorname*{arg\,max}_{\mathcal{S}'\subseteq\mathcal{N}^+, |\mathcal{S}'|=|\mathcal{T}|+1} \sum_{i\in\mathcal{S}'} U_i\right) \\
&= \mathbb{P}\left(Q = |\mathcal{T}|\right) \cdot \mathbb{P}\left(\sum_{t\in\mathcal{T}} U_t = \max_{\mathcal{S}\subseteq\mathcal{N}^+, |\mathcal{S}|=|\mathcal{T}|} \sum_{i\in\mathcal{S}} U_i\right) \\
&\quad + \mathbb{P}\left(Q > |\mathcal{T}|\right) \cdot \mathbb{P}\left(\sum_{t\in\mathcal{T}} U_t = \max_{\mathcal{S}\subseteq\mathcal{N}^+, |\mathcal{S}|=|\mathcal{T}|} \sum_{i\in\mathcal{S}} U_i, \sum_{t\in\mathcal{T}} U_t + U_0 = \max_{\mathcal{S}'\subseteq\mathcal{N}^+, |\mathcal{S}'|=|\mathcal{T}|+1} \sum_{i\in\mathcal{S}'} U_i\right).
\end{aligned}$$

Given that $U_1 \succeq_1 U_2 \succeq_1 \cdots \succeq_1 U_N$, if $U_l \succeq_1 U_m$, we have

$$\begin{aligned}
\omega_l &= \sum_{\mathcal{T}\subseteq\mathcal{N}} \pi_{MC-RIUM}(\mathcal{T}) \cdot \mathbb{I}(l \in \mathcal{T}) \\
&= \sum_{\mathcal{T}\subseteq\mathcal{N}, l\in\mathcal{T}} \left[\mathbb{P}\left(Q = |\mathcal{T}|\right) \cdot \mathbb{P}\left(\sum_{t\in\mathcal{T}} U_t = \max_{\mathcal{S}\subseteq\mathcal{N}^+, |\mathcal{S}|=|\mathcal{T}|} \sum_{i\in\mathcal{S}} U_i\right)\right. \\
&\quad \left. + \mathbb{P}\left(Q > |\mathcal{T}|\right) \cdot \mathbb{P}\left(\sum_{t\in\mathcal{T}} U_t = \max_{\mathcal{S}\subseteq\mathcal{N}^+, |\mathcal{S}|=|\mathcal{T}|} \sum_{i\in\mathcal{S}} U_i, \sum_{t\in\mathcal{T}} U_t + U_0 = \max_{\mathcal{S}'\subseteq\mathcal{N}^+, |\mathcal{S}'|=|\mathcal{T}|+1} \sum_{i\in\mathcal{S}'} U_i\right)\right] \\
&= \sum_{\mathcal{T}\subseteq\mathcal{N}\setminus\{l,m\}} \left[\mathbb{P}(Q = |\mathcal{T}|+1) \cdot \mathbb{P}\left(U_l + \sum_{t\in\mathcal{T}} U_t = \max_{\mathcal{S}\subseteq\mathcal{N}^+, |\mathcal{S}|=|\mathcal{T}|+1} \sum_{i\in\mathcal{S}} U_i\right)\right. \\
&\quad + \mathbb{P}(Q = |\mathcal{T}|+2) \cdot \mathbb{P}\left(U_l + U_m + \sum_{t\in\mathcal{T}} U_t = \max_{\mathcal{S}\subseteq\mathcal{N}^+, |\mathcal{S}|=|\mathcal{T}|+2} \sum_{i\in\mathcal{S}} U_i\right) \\
&\quad + \mathbb{P}(Q > |\mathcal{T}|+1) \cdot \mathbb{P}\left(\begin{array}{l} U_l + \sum_{t\in\mathcal{T}} U_t = \max_{\mathcal{S}\subseteq\mathcal{N}^+, |\mathcal{S}|=|\mathcal{T}|+1} \sum_{i\in\mathcal{S}} U_i, \\ U_l + \sum_{t\in\mathcal{T}} U_t + U_0 = \max_{\mathcal{S}'\subseteq\mathcal{N}^+, |\mathcal{S}'|=|\mathcal{T}|+2} \sum_{i\in\mathcal{S}'} U_i \end{array}\right) \\
&\quad \left. + \mathbb{P}(Q > |\mathcal{T}|+2) \cdot \mathbb{P}\left(\begin{array}{l} U_l + U_m + \sum_{t\in\mathcal{T}} U_t = \max_{\mathcal{S}\subseteq\mathcal{N}^+, |\mathcal{S}|=|\mathcal{T}|+2} \sum_{i\in\mathcal{S}} U_i, \\ U_l + U_m + \sum_{t\in\mathcal{T}} U_t + U_0 = \max_{\mathcal{S}'\subseteq\mathcal{N}^+, |\mathcal{S}'|=|\mathcal{T}|+3} \sum_{i\in\mathcal{S}'} U_i) \end{array}\right)\right] \\
&\geq \sum_{\mathcal{T}\subseteq\mathcal{N}\setminus\{l,m\}} \left[\mathbb{P}(Q = |\mathcal{T}|+1) \cdot \mathbb{P}\left(U_m + \sum_{t\in\mathcal{T}} U_t = \max_{\mathcal{S}\subseteq\mathcal{N}^+, |\mathcal{S}|=|\mathcal{T}|+1} \sum_{i\in\mathcal{S}} U_i\right)\right. \\
&\quad + \mathbb{P}(Q = |\mathcal{T}|+2) \cdot \mathbb{P}\left(U_l + U_m + \sum_{t\in\mathcal{T}} U_t = \max_{\mathcal{S}\subseteq\mathcal{N}^+, |\mathcal{S}|=|\mathcal{T}|+2} \sum_{i\in\mathcal{S}} U_i\right) \\
&\quad + \mathbb{P}(Q > |\mathcal{T}|+1) \cdot \mathbb{P}\left(\begin{array}{l} U_m + \sum_{t\in\mathcal{T}} U_t = \max_{\mathcal{S}\subseteq\mathcal{N}^+, |\mathcal{S}|=|\mathcal{T}|+1} \sum_{i\in\mathcal{S}} U_i, \\ U_m + \sum_{t\in\mathcal{T}} U_t + U_0 = \max_{\mathcal{S}'\subseteq\mathcal{N}^+, |\mathcal{S}'|=|\mathcal{T}|+2} \sum_{i\in\mathcal{S}'} U_i \end{array}\right) \\
&\quad \left. + \mathbb{P}(Q > |\mathcal{T}|+2) \cdot \mathbb{P}\left(\begin{array}{l} U_l + U_m + \sum_{t\in\mathcal{T}} U_t = \max_{\mathcal{S}\subseteq\mathcal{N}^+, |\mathcal{S}|=|\mathcal{T}|+2} \sum_{i\in\mathcal{S}} U_i, \\ U_l + U_m + \sum_{t\in\mathcal{T}} U_t + U_0 = \max_{\mathcal{S}'\subseteq\mathcal{N}^+, |\mathcal{S}'|=|\mathcal{T}|+3} \sum_{i\in\mathcal{S}'} U_i) \end{array}\right)\right] \\
&= \omega_m.
\end{aligned}$$

Hence, $U_l \succeq_1 U_m$ implies $\omega_l \geq \omega_m$ $\forall l, m \in \mathcal{N}$ and $l \neq m$. So, indexing products such that $U_1 \succeq_1 U_2 \succeq_1 \cdots \succeq_1 U_N$ is an MCI. For any subset $\mathcal{T} \subseteq \mathcal{N}$ with $|\mathcal{T}| = k$ for some non-negative integer k and two distinct elements $l, m \in \mathcal{N}\setminus\mathcal{T}$ with

$l<m$. Let $\mathcal{T}'=\mathcal{T}\cup\{l\}$ and $\mathcal{T}''=\mathcal{T}\cup\{m\}$. Then, we have $U_l\succeq_1 U_m$ and

$$\begin{aligned}
\pi_{MC-RIUM}(\mathcal{T}') &= \pi_{MC-RIUM}(\mathcal{T}\cup\{l\}) \\
&= \mathbb{P}(Q=k+1)\cdot\mathbb{P}\left(\sum_{t\in\mathcal{T}\cup\{l\}}U_t=\max_{\mathcal{S}\subseteq\mathcal{N}^+,|\mathcal{S}|=k+1}\sum_{i\in\mathcal{S}}U_i\right) \\
&\quad +\mathbb{P}(Q>k+1)\cdot\mathbb{P}\left(\begin{array}{l}\sum_{t\in\mathcal{T}\cup\{l\}}U_t=\max_{\mathcal{S}\subseteq\mathcal{N}^+,|\mathcal{S}|=k+1}\sum_{i\in\mathcal{S}}U_i, \\ \sum_{t\in\mathcal{T}\cup\{l\}}U_t+U_0=\max_{\mathcal{S}'\subseteq\mathcal{N}^+,|\mathcal{S}'|=k+2}\sum_{i\in\mathcal{S}'}U_i\end{array}\right) \\
&= \mathbb{P}(Q=k+1)\cdot\mathbb{P}\left(U_l+\sum_{t\in\mathcal{T}}U_t=\max_{\mathcal{S}\subseteq\mathcal{N}^+,|\mathcal{S}|=k+1}\sum_{i\in\mathcal{S}}U_i\right) \\
&\quad +\mathbb{P}(Q>k+1)\cdot\mathbb{P}\left(\begin{array}{l}U_l+\sum_{t\in\mathcal{T}}U_t=\max_{\mathcal{S}\subseteq\mathcal{N}^+,|\mathcal{S}|=k+1}\sum_{i\in\mathcal{S}}U_i, \\ U_l+\sum_{t\in\mathcal{T}}U_t+U_0=\max_{\mathcal{S}'\subseteq\mathcal{N}^+,|\mathcal{S}'|=k+2}\sum_{i\in\mathcal{S}'}U_i\end{array}\right) \\
&\geq \mathbb{P}(Q=k+1)\cdot\mathbb{P}\left(U_m+\sum_{t\in\mathcal{T}}U_t=\max_{\mathcal{S}\subseteq\mathcal{N}^+,|\mathcal{S}|=k+1}\sum_{i\in\mathcal{S}}U_i\right) \\
&\quad +\mathbb{P}(Q>k+1)\cdot\mathbb{P}\left(\begin{array}{l}U_m+\sum_{t\in\mathcal{T}}(V_t+\epsilon_t)=\max_{\mathcal{S}\subseteq\mathcal{N}^+,|\mathcal{S}|=k+1}\sum_{i\in\mathcal{S}}U_i, \\ U_m+\sum_{t\in\mathcal{T}}V_t+\epsilon_t+U_0=\max_{\mathcal{S}'\subseteq\mathcal{N}^+,|\mathcal{S}'|=k+2}\sum_{i\in\mathcal{S}'}U_i\end{array}\right) \\
&= \pi_{MC-RIUM}(\mathcal{T}\cup\{m\})=\pi_{MC-RIUM}(\mathcal{T}'').
\end{aligned}$$

In a similar sense, we can repeatedly pick a pair of elements (i_l,i_m) with $i_l,i_m\in\mathcal{N}\backslash\mathcal{T}$ and $i_l<i_m$, and verify that $\pi_{MC-RIUM}(\mathcal{T}'\cup\{i_l\})\geq\pi_{MC-RIUM}(\mathcal{T}''\cup\{i_m\})$. Thus, we conclude this indexing rule is dominant w.r.t the MC-RIUM.

For the second half, it holds directly from Theorem 3 as we find the MCI is dominant w.r.t. the MC-RIUM. □

Proof of Corollary 2 Assume that random utilities have the form $U_n=V_n+\epsilon_n\ \forall n\in\mathcal{N}^+$ and $\epsilon_1,\ldots,\epsilon_N$ are independent and identically distributed. For any $l,m\in\mathcal{N}$, if $V_l\geq V_m$, then we have

$$\mathbb{P}(U_l\leq u)=\mathbb{P}(V_l+\epsilon_l\leq u)\leq\mathbb{P}(V_m+\epsilon_l\leq u)=\mathbb{P}(U_m\leq u).$$

By the definition of first-order stochastic dominance, the decreasing order of the utility in the FSD sense is equivalent to the decreasing order of the deterministic utility.

Thus, according to Proposition 7, indexing products such that $V_1\geq V_2\geq\ldots V_N$ is an MCI and is dominant w.r.t. the MC-RIUM. □

Proof of Proposition 8 For the first half, if $\beta_{il}\geq\beta_{im}\ \forall i\in\mathcal{N}\backslash\{l,m\}$ for any $l,m\in\mathcal{N}$ with $V_l\geq V_m$, we claim that indexing products such that $V_1\geq V_2\geq\cdots\geq V_N$ is dominant w.r.t. the BundleMVL-L model.

The choice probability of a set $\mathcal{T}\subseteq\mathcal{N}$ under BundleMVL-L model can be represented as $\pi_{BundleMVL-L}(\mathcal{T})=\frac{V_\mathcal{T}}{1+\sum_{\mathcal{T}'\subseteq\mathcal{N},|\mathcal{T}'|\leq L}V_{\mathcal{T}'}}$, where $V_\mathcal{T}=\exp\left(\sum_{i\in\mathcal{T}}V_i+\sum_{i,j\in\mathcal{T},i<j}\beta_{ij}\right)$. Given $\beta_{il}\geq\beta_{im}\ \forall i\in\mathcal{N}\backslash\{l,m\}$ for any $l,m\in\mathcal{N}$

with $V_l \geq V_m$, if $V_l \geq V_m$, we have

$$\begin{aligned}
\omega_l &= \sum_{\mathcal{T}\subseteq\mathcal{N},|\mathcal{T}|\leq L} \pi_{BundleMVL-L}(\mathcal{T})\cdot\mathbb{I}(l\in\mathcal{T}) \\
&= \sum_{\mathcal{T}\subseteq\mathcal{N},|\mathcal{T}|\leq L,l\in\mathcal{T}} \frac{\exp\left(\sum_{i\in\mathcal{T}} V_i + \sum_{i,j\in\mathcal{T},i<j}\beta_{ij}\right)}{1+\sum_{\mathcal{T}'\subseteq\mathcal{N},|\mathcal{T}'|\leq L}\exp\left(\sum_{i\in\mathcal{T}'} V_i + \sum_{i,j\in\mathcal{T}',i<j}\beta_{ij}\right)} \\
&= \sum_{\mathcal{T}\subseteq\mathcal{N}\setminus\{l,m\},|\mathcal{T}|\leq L-1} \frac{\exp\left((V_l+\sum_{j\in\mathcal{T},j\neq l}\beta_{jl}) + \sum_{i\in\mathcal{T}} V_i + \sum_{i,j\in\mathcal{T},i<j}\beta_{ij}\right)}{1+\sum_{\mathcal{T}'\subseteq\mathcal{N},|\mathcal{T}'|\leq L}\exp\left(\sum_{i\in\mathcal{T}'} V_i + \sum_{i,j\in\mathcal{T}',i<j}\beta_{ij}\right)} \\
&\quad + \sum_{\mathcal{T}\subseteq\mathcal{N}\setminus\{l,m\},|\mathcal{T}|\leq L-2} \frac{\exp\left((V_l+\sum_{j\in\mathcal{T},j\neq l}\beta_{jl}+V_m+\sum_{j\in\mathcal{T},j\neq m}\beta_{jm}) + \sum_{i\in\mathcal{T}} V_i + \sum_{i,j\in\mathcal{T},i<j}\beta_{ij}\right)}{1+\sum_{\mathcal{T}'\subseteq\mathcal{N},|\mathcal{T}'|\leq L}\exp\left(\sum_{i\in\mathcal{T}'} V_i + \sum_{i,j\in\mathcal{T}',i<j}\beta_{ij}\right)} \\
&\geq \sum_{\mathcal{T}\subseteq\mathcal{N}\setminus\{l,m\},|\mathcal{T}|\leq L-1} \frac{\exp\left((V_m+\sum_{j\in\mathcal{T},j\neq m}\beta_{jm}) + \sum_{i\in\mathcal{T}} V_i + \sum_{i,j\in\mathcal{T},i<j}\beta_{ij}\right)}{1+\sum_{\mathcal{T}'\subseteq\mathcal{N},|\mathcal{T}'|\leq L}\exp\left(\sum_{i\in\mathcal{T}'} V_i + \sum_{i,j\in\mathcal{T}',i<j}\beta_{ij}\right)} \\
&\quad + \sum_{\mathcal{T}\subseteq\mathcal{N}\setminus\{l,m\},|\mathcal{T}|\leq L-2} \frac{\exp\left((V_l+\sum_{j\in\mathcal{T},j\neq l}\beta_{jl}+V_m+\sum_{j\in\mathcal{T},j\neq m}\beta_{jm}) + \sum_{i\in\mathcal{T}} V_i + \sum_{i,j\in\mathcal{T},i<j}\beta_{ij}\right)}{1+\sum_{\mathcal{T}'\subseteq\mathcal{N},|\mathcal{T}'|\leq L}\exp\left(\sum_{i\in\mathcal{T}'} V_i + \sum_{i,j\in\mathcal{T}',i<j}\beta_{ij}\right)} \\
&= \omega_m.
\end{aligned}$$

Hence, $V_l \geq V_m$ implies $\omega_l \geq \omega_m$ $\forall l,m\in\mathcal{N}$ and $l\neq m$. So, indexing products such that $V_1 \geq V_2 \geq \cdots \geq V_N$ is an MCI. For any subset $\mathcal{T}\subseteq\mathcal{N}$ with $|\mathcal{T}|=k\leq L-1$ for some non-negative integer k and two distinct elements $l,m\in\mathcal{N}\setminus\mathcal{T}$ with $l<m$. Let $\mathcal{T}'=\mathcal{T}\cup\{l\}$ and $\mathcal{T}''=\mathcal{T}\cup\{m\}$. Then, we have $V_l\geq V_m$ and

$$\begin{aligned}
\pi_{BundleMVL-L}(\mathcal{T}') = \pi_{BundleMVL-L}(\mathcal{T}\cup\{l\}) &= \frac{\exp\left((V_l+\sum_{j\in\mathcal{T},j\neq l}\beta_{jl}) + \sum_{i\in\mathcal{T}} V_i + \sum_{i,j\in\mathcal{T},i<j}\beta_{ij}\right)}{1+\sum_{\mathcal{R}\subseteq\mathcal{N},|\mathcal{R}|\leq L}\exp\left(\sum_{i\in\mathcal{R}} V_i + \sum_{i,j\in\mathcal{R},i<j}\beta_{ij}\right)} \\
&\geq \frac{\exp\left((V_m+\sum_{j\in\mathcal{T},j\neq m}\beta_{jm}) + \sum_{i\in\mathcal{T}} V_i + \sum_{i,j\in\mathcal{T},i<j}\beta_{ij}\right)}{1+\sum_{\mathcal{R}\subseteq\mathcal{N},|\mathcal{R}|\leq L}\exp\left(\sum_{i\in\mathcal{R}} V_i + \sum_{i,j\in\mathcal{R},i<j}\beta_{ij}\right)} \\
&= \pi_{BundleMVL-L}(\mathcal{T}\cup\{m\}) = \pi_{BundleMVL-L}(\mathcal{T}'').
\end{aligned}$$

In a similar sense, we can repeatedly pick a pair of elements (i_l,i_m) with $i_l,i_m\in\mathcal{N}\setminus\mathcal{T}$ and $i_l<i_m$, and verify that $\pi_{BundleMVL-L}(\mathcal{T}'\cup\{i_l\}) \geq \pi_{BundleMVL-L}(\mathcal{T}''\cup\{i_m\})$. Thus, we conclude this indexing rule is dominant w.r.t the BundleMVL-L model provided $\beta_{il}\geq\beta_{im}$ $\forall i\in\mathcal{N}\setminus\{l,m\}$ for any $l,m\in\mathcal{N}$ with $V_l\geq V_m$.

For the second half, it holds from Theorem 3 as we find the MCI is dominant w.r.t. the BundleMVL-L model provided $\beta_{il}\geq\beta_{im}$ $\forall i\in\mathcal{N}\setminus\{l,m\}$ for any $l,m\in\mathcal{N}$ with $V_l\geq V_m$. □

EC.2. A Method Utilizing Benders Decomposition for Solving (CP) with Type-I Cost Functions

First, we reformulate Problem (3) as follows

$$\begin{aligned}
\max_{\boldsymbol{\xi},\boldsymbol{\lambda}} \ & \sum_{m=1}^{M} \pi_m g_m \lambda_m, \\
\text{s.t. } & \lambda_m \leq h_m(\boldsymbol{\xi}), \quad \forall m\in[M], \\
& \sum_{n=1}^{N} \xi_n = K, \\
& \xi_n\in\{0,1\}, \qquad \forall n\in[N],
\end{aligned} \tag{EC.7}$$

where subproblem $h_m(\boldsymbol{\xi})$ is defined as follows

$$\begin{aligned} h_m(\boldsymbol{\xi}) = \max_{\zeta_m} \ & \zeta_m, \\ \text{s.t. } & \zeta_m \le \xi_n, \ \forall n \in \mathcal{T}_m, \\ & \zeta_m \ge 0. \end{aligned} \tag{EC.8}$$

Similar to the analysis of the MILP (3), when $\boldsymbol{\xi}$ is binary, the constraints in Problem (EC.8) automatically forces $\boldsymbol{\zeta}$ to be the correct binary value. Let $A^{(m)}(\boldsymbol{\xi}) = \{\zeta | \zeta \le \xi_n \ \forall n \in \mathcal{T}_m, \ \zeta \ge 0\}$ be the feasible set of ζ_m for the subProblem (EC.8), then we can rewrite the main Problem (EC.7) as

$$\begin{aligned} \max_{\boldsymbol{\xi},\boldsymbol{\lambda}} \ & \sum_{m=1}^{M} \pi_m g_m \lambda_m, \\ \text{s.t. } & \lambda_m \le \zeta_m, \qquad \forall m \in [M], \ \zeta_m \in A^m(\boldsymbol{\xi}), \\ & \sum_{n=1}^{N} \xi_n = K, \\ & \xi_n \in \{0,1\}, \qquad \forall n \in [N]. \end{aligned} \tag{EC.9}$$

According to the straightforward structure of subProblem (EC.8), we can easily derive an optimal solution. We present this formally in the following proposition.

Proposition EC.1. *Given $\boldsymbol{\xi} \in \{0,1\}^N$, the optimal solution of Problem (EC.8) for order $m \in [M]$ with the assortment represented by $\boldsymbol{\xi}$ is*

$$\zeta_m = \begin{cases} 0 \ \textit{if } \min_{j \in \mathcal{T}_m} \xi_j < 1, \\ 1 \ \textit{otherwise.} \end{cases}$$

Proposition EC.1 provides us with a simple approach to test whether an integer solution $(\boldsymbol{\xi}, \boldsymbol{\lambda})$ to Problem (EC.8) violates any constraint or not, i.e., we generate ζ_m according to Proposition EC.1 and compare it with λ_m for each $m \in [M]$. If λ_m is less or equal to ζ_m, then the mth set of constraints remains unviolated; otherwise, if λ_m is strictly larger than ζ_m, then we have identified a violated constraint and add it to the formulation. This procedure is inspired by Bertsimas and Mišić (2019), which proposes a Benders decomposition method to solve the product line design problem under the classic single-purchase rank list model. Note that the structure of Problem (3) is simpler than that presented in Bertsimas and Mišić (2019), so applying Benders decomposition to our problem is expected be more efficient.

EC.3. Supplementary Materials for Numerical Experiments

EC.3.1. Supplementary Materials for Section 6.1

In-sample and out-of-sample comparisons of order fill rates under different methods for RDCs are shown in Table EC.1 and Table EC.2. In-sample and out-of-sample comparisons of order fill rates under different methods for CDCs are shown in Table EC.3 and Table EC.4.

Table EC.1 In-Sample OFR Comparison Results for RDCs

DC Code	# SKUs	SKU Cap	# Diff Orders	Current OFR (%)	OPT OFR (%)	MCIP OFR (%)	Avg RP OFR (%)	OPT IMP Current (%)	MCIP IMP Current (%)	$\frac{ORF_{MCIP}}{ORF_{OPT}}$ (%)	MILP Time (s)
RRSZX001	3985	3601	3901	98.84	99.7	99.6	89.22	0.87	0.76	99.9	2.18
RRSZX002	2915	1932	2730	88.04	97.14	96.82	63.05	10.34	9.98	99.67	1.01
RRSZX003	3323	2441	3205	93.24	98.59	98.48	70.49	5.74	5.62	99.89	0.82
RRSZX004	3463	2525	3410	94.39	98.76	98.58	71.18	4.63	4.44	99.82	0.89
RRSZX020	3256	2580	3204	96.61	99.21	99.16	78.53	2.68	2.63	99.95	0.85
RRSZX022	47	31	26	66.1	86.44	84.75	46.02	30.77	28.21	98.04	0.14
RRSZX023	922	658	852	96.39	98.57	98.44	70.41	2.26	2.13	99.87	0.29
RRSZX033	2992	1905	2940	92.82	97.91	97.76	62.17	5.49	5.32	99.84	0.87
RRSZX038	3084	1911	3054	83.94	96.64	96.52	59.62	15.13	14.98	99.88	0.88
RRSZX039	1436	1041	1313	94.57	99.34	99.17	70.92	5.04	4.86	99.83	0.42
RRSZX040	1148	1049	1044	99.16	99.8	99.71	90.72	0.64	0.55	99.9	0.33
RRSZX041	4279	3054	4335	94.85	99.0	98.94	67.94	4.38	4.31	99.93	1.12
RRSZX043	4128	3102	4140	96.59	99.35	99.27	72.13	2.86	2.77	99.92	1.67
RRSZX044	3630	2720	3571	95.22	98.78	98.71	71.91	3.74	3.66	99.92	1.19
RRSZX045	3748	2792	3772	95.21	98.87	98.69	71.98	3.84	3.66	99.82	1.22
RRSZX058	4363	3230	4515	97.11	99.44	99.38	72.34	2.4	2.34	99.94	1.88
RRSZX059	4444	3410	4568	97.03	99.52	99.46	74.33	2.57	2.51	99.94	1.72
RRSZX060	4061	3018	4170	97.29	99.38	99.29	72.05	2.15	2.06	99.9	1.39
RRSZX073	3002	1986	3026	91.06	97.54	97.23	64.51	7.12	6.78	99.68	1.47
RRSZX074	3055	2070	3138	93.35	98.36	98.23	66.78	5.37	5.23	99.86	1.16
RRSZX082	3475	2298	3535	93.91	98.43	98.3	65.0	4.81	4.68	99.87	1.31
RRSZX083	3655	2568	3645	95.26	98.9	98.78	68.85	3.82	3.69	99.88	1.25
RRSZX084	3988	2960	3937	95.79	99.21	99.09	72.82	3.56	3.44	99.88	1.57
RRSZX094	1597	957	1569	84.46	94.86	94.5	58.49	12.32	11.9	99.62	0.6
RRSZX095	2879	1884	2817	92.21	97.67	97.39	64.51	5.93	5.63	99.72	0.92
RRSZX110	4108	3189	3911	95.72	98.93	98.88	74.96	3.36	3.31	99.95	1.04

Table EC.2 Out-of-Sample OFR Comparison Results for RDCs

DC Code	# SKUs	SKU Cap	# Diff Orders	HS OPT OFR (%)	Test OPT OFR (%)	Test MCIP OFR (%)	Test MCIP IMP Over Test OPT (%)
RRSZX001	3985	3601	3505	100.0	98.63	98.85	0.22
RRSZX002	2915	1932	2323	99.63	95.44	95.49	0.05
RRSZX003	3323	2441	2764	99.89	96.53	95.9	-0.65
RRSZX004	3463	2525	2868	99.91	96.83	97.21	0.4
RRSZX020	3256	2580	2685	100.0	97.95	98.0	0.04
RRSZX022	47	31	13	100.0	48.91	42.39	-13.33
RRSZX023	922	658	686	99.82	97.48	97.72	0.25
RRSZX033	2992	1905	2465	99.66	95.16	95.89	0.77
RRSZX038	3084	1911	2658	99.33	94.64	94.98	0.35
RRSZX039	1436	1041	1058	99.95	96.81	96.92	0.11
RRSZX040	1148	1049	904	100.0	98.33	98.66	0.34
RRSZX041	4279	3054	3986	99.88	97.78	97.52	-0.28
RRSZX043	4128	3102	3753	99.95	98.23	97.81	-0.44
RRSZX044	3630	2720	3287	99.92	96.58	96.53	-0.05
RRSZX045	3748	2792	3534	99.92	96.68	95.98	-0.73
RRSZX058	4363	3230	3846	99.98	98.42	98.26	-0.15
RRSZX059	4444	3410	3868	100.0	98.16	98.05	-0.11
RRSZX060	4061	3018	3595	99.98	98.03	97.97	-0.06
RRSZX073	3002	1986	2676	99.56	93.35	94.5	1.23
RRSZX074	3055	2070	2760	99.75	96.45	96.64	0.2
RRSZX082	3475	2298	3114	99.69	95.99	96.12	0.13
RRSZX083	3655	2568	3251	99.86	96.15	96.15	-0.0
RRSZX084	3988	2960	3422	99.95	97.72	97.38	-0.35
RRSZX094	1597	957	1264	99.22	93.17	93.4	0.25
RRSZX095	2879	1884	2415	99.7	96.56	96.29	-0.28
RRSZX110	4108	3189	3320	100.0	97.64	97.65	0.01

Table EC.3 In-Sample OFR Comparison Results for CDCs

DC Code	# SKUs	SKU Cap	# Diff Orders	Current OFR (%)	OPT OFR (%)	MCIP OFR (%)	Avg RP OFR (%)	OPT IMP Current (%)	MCIP IMP Current (%)	$\frac{ORF_{MCIP}}{ORF_{OPT}}$ (%)	MILP Time (s)
RRSZX031	4410	3973	4631	99.22	99.8	99.74	89.43	0.58	0.52	99.94	1.55
RRSZX042	3981	3680	4021	98.68	99.82	99.72	91.33	1.16	1.05	99.9	1.58
RRSZX056	4084	3722	4136	99.22	99.85	99.79	90.35	0.64	0.58	99.94	1.67
RRSZX072	3838	3420	3888	97.4	99.64	99.55	87.42	2.3	2.2	99.91	1.61
RRSZX081	4087	3870	4042	99.54	99.91	99.83	93.96	0.38	0.29	99.92	1.55
RRSZX093	4030	3659	4011	98.65	99.74	99.67	89.82	1.11	1.04	99.93	1.06
RRSZX104	3648	2754	3494	95.75	99.03	98.92	72.31	3.43	3.32	99.9	0.84

Table EC.4 Out-of-Sample OFR Comparison Results for CDCs

DC Code	# SKUs	SKU Cap	# Diff Orders	HS OPT OFR (%)	Test OPT OFR (%)	Test MCIP OFR (%)	Test MCIP IMP Over Test OPT (%)
RRSZX031	4410	3973	4144	100.0	98.25	98.1	-0.15
RRSZX042	3981	3680	3651	100.0	99.06	99.26	0.2
RRSZX056	4084	3722	3634	100.0	99.66	99.49	-0.17
RRSZX072	3838	3420	3577	100.0	98.17	98.07	-0.1
RRSZX081	4087	3870	3480	100.0	99.68	99.84	0.16
RRSZX093	4030	3659	3635	100.0	99.25	98.53	-0.73
RRSZX104	3648	2754	3103	99.94	96.35	96.8	0.47

EC.3.2. Supplementary Materials for Section 6.2

Detail of the chosen cities is listed in Table EC.5.

Table EC.5 Details of Chosen Cities

	City Code	# SKU	# Distinct Orders	Avg MILP Solving Time
City 1	b3bbd6bcd84bc84f8bb874d96cee51e6	5143	5554	1.54
City 2	e7e6252a02709c4f1bfab796ebd3efe2	5140	5663	1.60
City 3	2942fa707f340db57611c88fca53a211	4952	5264	1.47
City 4	08641489dbf16de3f0fbdb8095d8721d	4984	5268	1.47
City 5	b0cadecbe35d998f1758ff45f23dffbb	4841	4910	1.35

Figures of OFR comparisons, PFR comparisons, and the ratios comparisons for City 2 to 5 are shown in Figure EC.1 to Figure EC.12.

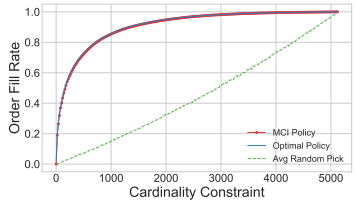

Figure EC.1 City 2 OFRs

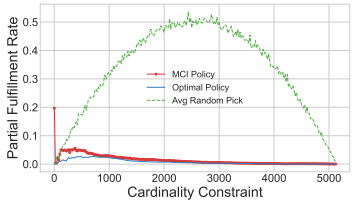

Figure EC.2 City 2 PFRs

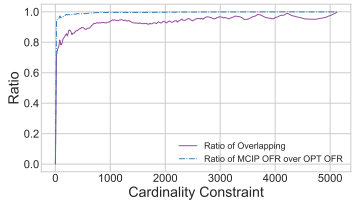

Figure EC.3 City 2 Ratios

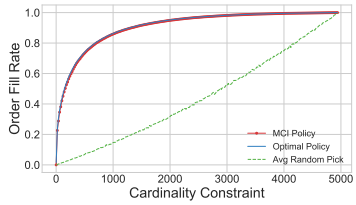

Figure EC.4 City 3 OFRs

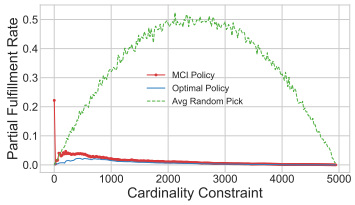

Figure EC.5 City 3 PFRs

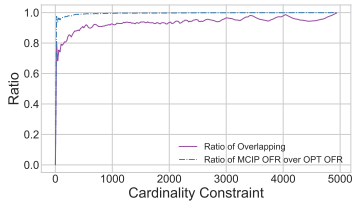

Figure EC.6 City 3 Ratios

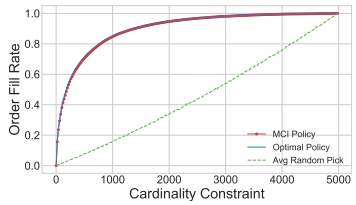

Figure EC.7 City 4 OFRs

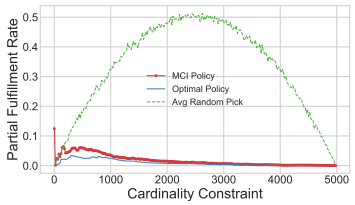

Figure EC.8 City 4 PFRs

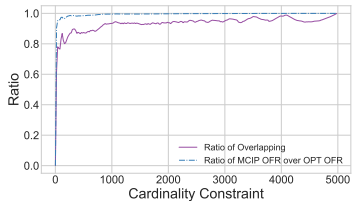

Figure EC.9 City 4 Ratios

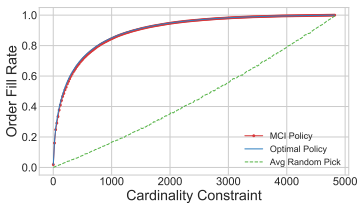

Figure EC.10 City 5 OFRs

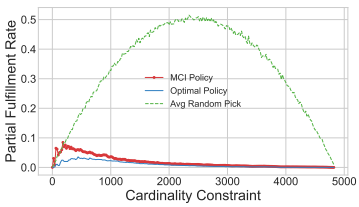

Figure EC.11 City 5 PFRs

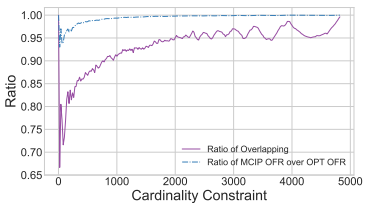

Figure EC.12 City 5 Ratios

EC.3.3. Experiments on General Cost Functions

We further test the MCI policy on some general cost functions. The experiment settings are the same as that in Section 6.1.

EC.3.3.1. Size-Based Cost Functions We fist consider both the type-I and type-II size-based cost functions with $c(n) = n^\alpha \ \forall n \in \{1, 2, \dots\}$, where $\alpha = 1/3, 1/2, 1, 2$. Table EC.6 reports the average cost reduction ratio of applying the MCI policy and the MILP optimal solution compared to the current policy at different distribution centers. It demonstrates the high potential for additional fulfillment cost reduction by carefully selecting the warehouse assortment for all (either concave, linear, or convex) cost functions. Specifically, the MCI policy results in a significant average cost reduction ranging from 33.76% to 80.44% under both types of cost functions. Indeed, as is shown in Proposition 3, when the cost function is type-II size-based with $\alpha = 1$, the MCI policy is optimal. Besides, we find that for most cases, as α becomes smaller (from 2 to 1/3), the average cost reduction by applying the MCI policy becomes larger under type-I size-based cost functions; while the reverse is true under type-II size-based cost functions. Note that the more different cost functions are from linear functions, the larger gap the MCI policy performs in comparison to the MILP optimal solution. However, although formulation (3) can be solved within seconds considering type-I cost functions, it takes much longer (more than half an hour) for solving (CP) with type-II cost functions through MILP formulation (4) when N and M are larger than 3000. In this regard, the MCI policy with near-optimal performance solved in milliseconds is quite acceptable.

Table EC.6 In-Sample Average Cost Reduction Comparison Results

DC Type	Cost Type	α	MCIP AVG Cost Reduction Over Current	OPT AVG Cost Reduction Over Current
LTC	Type I	0.33	38.04%	43.62%
		0.5	37.98%	43.26%
		1	37.43%	42.12%
		2	33.76%	48.35%
	Type II	0.33	39.18%	44.03%
		0.5	39.73%	43.41%
		1	41.23%	41.23%
		2	43.63%	49.24%
RDC	Type I	0.33	72.02%	74.49%
		0.5	71.74%	74.01%
		1	70.68%	72.97%
		2	66.86%	76.66%
	Type II	0.33	72.55%	75.20%
		0.5	72.59%	74.55%
		1	72.71%	72.71%
		2	72.98%	77.46%
CDC	Type I	0.33	72.81%	79.24%
		0.5	72.42%	78.37%
		1	71.05%	76.59%
		2	66.73%	83.59%
	Type II	0.33	74.15%	80.63%
		0.5	74.56%	79.50%
		1	76.09%	76.09%
		2	80.44%	82.69%

Moreover, since RDCs usually have more SKU capacities than LTCs and CDCs usually have more SKU capacities than RDCs, Table EC.6 implies that as the SKU capacities become larger, the average cost reduction of using the MCI policy becomes more evident. This finding complements the result in Section 6.1 where the MCI policy results in larger OFR improvement for facilities with smaller SKU capacities.

EC.3.3.2. Type-II Cost Functions with $G(\mathcal{T}) = \sum_{n\in\mathcal{T}} \kappa_n$ In this subsection, we specifically consider the case where the cost function is type-II with $G(\mathcal{T}) = \sum_{n\in\mathcal{T}} \kappa_n$, which is discussed in the beginning of Section 5.1. Specifically, we designate the product-specific additional fulfillment cost κ_n as the volume or weight of the product in our experiments. According to Proposition 4, a modified MCI policy (MMCIP), which selects the K products with the largest choice-weighted fulfillment costs to store, achieves optimality. In these experiments, we engage in a comparative analysis of the cost reduction achieved by applying both MMCIP and the standard MCI policy (MCIP). The results of the comparison are summarized in Table EC.7. We observe that although applying

Table EC.7 In-Sample Average Type-II Cost with $G(\mathcal{T}) = \sum_{n\in\mathcal{T}} \kappa_n$ Reduction Comparison Results

DC Type	Product-Specific Cost (κ_n) Based on	MCIP AVG Cost Reduction Over Current	MMCIP AVG Cost Reduction Over Current	MMCIP AVG IMP Over MCIP
LTC	volume	-5.38%	65.79%	48.12%
	weight	35.00%	66.25%	45.32%
RDC	volume	56.27%	81.99%	53.17%
	weight	67.27%	82.61%	44.87%
CDC	volume	48.47%	91.47%	74.77%
	weight	76.65%	90.95%	61.89%

the standard MCI policy leads to reduced additional fulfillment costs in most instances compared to the current practice, the modified MCI policy consistently delivers more substantial cost reductions across all test cases. These findings suggest that when the additional fulfillment cost exhibits a type-II structure and is intricately related to product-specific features, it is important to incorporate these features into the design of the assortment selection policy. Indeed, a minor adjustment to the standard MCI policy can significantly enhance its performance.

Additionally, recall from Table EC.6 that it is verified the MMCIP reduces to the MCIP and achieves optimality when the cost function is type-II with $G(\mathcal{T}) = \sum_{n\in\mathcal{T}} \kappa_n$ and the product-specific additional fulfillment cost is uniform across all products (type-II sized-based with $\alpha = 1$).

Owing to space constraints, all detailed results of the experiments conducted on different cost functions within this subsection are available in `https://docs.google.com/spreadsheets/d/1OUI8j4YfKRGoZqS_LHstto6SItuu0e8QMwClqgBUEZs/edit?usp=sharing`.