Joint Placement, Delivery Promise and Fulfillment in Online Retail

Yicheng Bai[1], Paat Rusmevichientong[2], Huseyin Topaloglu[1]
[1]School of Operations Research and Information Engineering, Cornell Tech, New York, NY 10044, USA
[2]Marshall School of Business, University of Southern California, Los Angeles, CA 90089, USA
yb279@cornell.edu, rusmevic@marshall.usc.edu, topaloglu@orie.cornell.edu
July 16, 2024

We consider the placement, delivery promise and fulfillment decisions of an online retailer. We have a set of products with given numbers of units to be placed at capacitated fulfillment centers. Once we make the placement decisions, we face demands for the products arriving from different demand regions randomly over time. In response to each demand, we pick a delivery promise to offer, which determines the probability that the demand converts into sale, as well as choose a fulfillment center to use to serve the demand. Our goal is to decide where to place the units at the beginning of the selling horizon and to find a policy to make delivery promise and fulfillment decisions over the selling horizon so that we maximize the total expected profit. We give an approximation strategy to obtain solutions with performance guarantees for this joint placement, delivery promise and fulfillment problem. In our approximation strategy, we construct a bounding function that upper bounds the total expected profit from the delivery promise and fulfillment policy when viewed as a function of the placement decisions. To make the placement decisions, we maximize the bounding function subject to the capacity constraints at the fulfillment centers. To make the delivery promise and fulfillment decisions, we construct a policy that obtains a constant fraction of the bounding function. Setting $a \wedge b = \min\{a, b\}$, letting $C_{\min}$ be the smallest number of units we need to place for a product and $U_{\min}$ be the smallest capacity of a fulfillment center, using our approximation strategy with an appropriate bounding function, for any $\epsilon > 0$, we obtain a $\max\left\{\frac{1}{4+\epsilon}, 1 - O\left(\frac{\sqrt{\log(C_{\min} \wedge U_{\min})}}{(C_{\min} \wedge U_{\min})^{1/3}}\right)\right\}$-approximate solution. Thus, we obtain a solution with a constant factor performance guarantee, but if the size of the system, measured by the numbers of units that we need to place and capacities of the fulfillment centers, is large, then we get an asymptotically optimal solution. We compare our approximation strategy with approaches that ignore the interactions between the placement, delivery promise and fulfillment decisions, as well as heuristics that are based on Lagrangian relaxation, demonstrating that our approximation strategy compares quite favorably.

1. Introduction

Placement, delivery promise and fulfillment are three important pieces of the logistics operations of an online retailer. Placement focuses on allocating the available units of the products to different fulfillment centers. Delivery promise involves committing to a timeline to get the products into the hands of the customer. Fulfillment figures out a fulfillment center to use to satisfy the customer while adhering to the given delivery promise. Placement decisions are made in advance, whereas delivery promise and fulfillment decisions are made in real time. While these decisions are made in different time scales, given that the locations of the units affect what delivery promises can be offered and what fulfillment centers can used for serving orders, online retailers need to make their placement decisions by considering their downstream impact on delivery promise and fulfillment. There are challenges in coordinating placement with delivery promise and fulfillment. In particular, delivery promise and fulfillment decisions are inherently made in an uncertain environment. It is a priori unknown where and in what quantities the demand will occur and whether the demand will convert into sale. Thus, it is difficult to characterize the total expected profit from the delivery promise and fulfillment decisions as a function of the placement decisions. Even if we can come up with an approximation to the total expected profit from the delivery promise and fulfillment decisions as a function of the placement decisions, such an approximation needs to be simple enough that we can solve an optimization problem that ultimately chooses the placement of the units.

In this paper, we consider the problem of coordinating the placement, delivery promise and fulfillment decisions for an online retailer. For each product, we have a fixed number of units to be placed at the capacitated fulfillment centers at the beginning of the selling horizon. Once we place these units, demands for the products arrive from different demand regions randomly over the selling horizon. In response to each demand, we pick a delivery promise to offer and a fulfillment center to use to serve the demand. The delivery promise refers to the number days within which the product will reach the hands of the customer and it determines the probability that the demand converts into sale. Our goal is to decide where to place the units at the beginning of the selling horizon and to find a policy to make the delivery promise and fulfillment decisions so that we maximize the total expected profit. In our setup, the total number of units for each product to be placed at the fulfillment centers at the beginning of the selling horizon is exogenously fixed, but we choose where to place the units. This approach is aligned with the practice of large online retailers that operate separate buying and placement groups. Buying groups choose the quantity of inventory to bring into the system. Placement groups take over once the inventory is brought into the system. Regional buys that bring inventory directly to fulfillment centers are uncommon. The common practice is to bring large quantities of inventory to cross docks through national buys, at which point, placement groups decide where to place the inventory. Buying groups calculate national target inventory positions, which dictate the total number of units for each product to be placed at different fulfillment centers. Placement groups choose the fulfillment centers to place these units, which have been fixed through a national target inventory position calculation.

In our model, we start by giving a dynamic program to characterize the optimal delivery promise and fulfillment policy over the selling horizon. In the dynamic program, the state variable keeps track of the numbers of remaining units of products at each fulfillment center, whereas the actions at each decision epoch correspond to the delivery promise that we offer to the demand arriving at the current decision epoch, along with the fulfillment center to use to serve the demand. The initial value function in the dynamic program captures the optimal total expected profit from the delivery promise and fulfillment decisions as a function of the initial placement quantities. To make the placement decisions, we maximize the initial value function by choosing the placement quantity for each product at each fulfillment center. In this way, we find placement decisions such that if we start with these initial placement decisions, then we maximize the optimal total expected profit from the delivery promise and fulfillment decisions over the selling horizon. Computing the initial value function requires solving a high-dimensional dynamic program, so we focus on giving an approximation strategy to obtain solutions with performance guarantees.

In our approximation strategy, we build a bounding function that upper bounds the optimal total expected profit from the delivery promise and fulfillment decisions when viewed as a function of the initial placement quantities. Once we build a bounding function, we proceed in two steps. First, we formulate a placement problem, where we choose the placement quantity for each product at each fulfillment center to maximize the bounding function. For some $\alpha \in (0,1]$, we show that we can obtain an α-approximate solution to the placement problem, yielding our approximate placement quantities. Second, we construct an approximate policy to make the delivery promise and fulfillment decisions. For some $\beta \in (0,1]$, we show that the total expected profit of the approximate policy starting with the approximate placement quantities is at least a β-fraction of the bounding function evaluated at the same placement quantities. In this case, making the placement decisions by following the approximate placement quantities and making the delivery promise and fulfillment decisions by following the approximate policy provides at least an $\alpha\beta$-fraction of the optimal total expected profit in the joint placement, delivery promise and fulfillment problem.

Contributions: Our main technical contribution is providing solutions with performance guarantees. We position our work and overview our technical contributions.

Coordinating Offline Resource Allocation with Online Rationing. Viewing placement as offline resource allocation decisions, whereas viewing delivery promise and fulfillment as online rationing decisions, we focus on coordinating offline resource allocation with online rationing. These two classes of decisions have traditionally been studied in isolation. There is only recent work on coordinating them. Considering our delivery promise and fulfillment decisions, these decisions are similar to capacity control and pricing decisions in revenue management, which choose the products whose demands should be accepted and the prices to be charged for these products in an online fashion. There is significant work on capacity control and pricing policies that are asymptotically optimal as the initial inventories and total expected demands for the products get large. Such asymptotic optimality results require that the inventories of all products simultaneously get large. Translating this requirement into our problem, to make asymptotically optimal delivery promise and fulfillment decisions, the inventories of all products at all fulfillment centers have to be large. In our problem setting, however, even if the total number of units to place for each product is large, once we place the units at the fulfillment centers, we cannot guarantee that the inventories of all products at all fulfillment centers will be large. On the other hand, considering our placement decisions, we make the placement decisions by solving a placement problem that chooses the placement quantity of each product at each fulfillment center to maximize the bounding function. Solving a continuous relaxation of this problem does not immediately yield useful placement decisions because the optimal solution to the continuous relaxation may place small fractional units at many fulfillment centers and it is not clear whether we can round such a solution while bounding the loss incurred by rounding. Therefore, there are difficulties in both making the placement decisions and finding a delivery promise and fulfillment policy.

Letting $a \wedge b = \min\{a, b\}$, using $C_{\min}$ to denote the smallest total number of units to be placed for a product and $U_{\min}$ to denote the smallest capacity of a fulfillment center, for any $\epsilon > 0$, we obtain a $\max\left\{\frac{1}{4+\epsilon}, 1 - O\left(\frac{\sqrt{\log(C_{\min} \wedge U_{\min})}}{(C_{\min} \wedge U_{\min})^{1/3}}\right)\right\}$-approximate solution to the joint placement, delivery promise and fulfillment problem. Here, both $C_{\min}$ and $U_{\min}$ are exogenous problem primitives. In this way, we get a constant factor performance guarantee, but if the size of the system, measured by the number of units to place and the fulfillment center capacities, grows large, then we get an asymptotically optimal solution. In the latter performance guarantee, the demand and profit parameters play no role. As the number of units and capacities increase, some demand and profit parameters may or may not change. We still get an asymptotically optimal solution.

Bounding Function. Our bounding function is based on a linear programming approximation for the delivery promise and fulfillment problem. This linear programming approximation is a fluid problem that is formulated under the assumption that the demands for the products and the responses to the delivery promises take on their expected values. Our work indicates that we can obtain both constant factor and asymptotically optimal solutions by leveraging a linear programming approximation, but solving a continuous relaxation of the linear programming approximation to make placement decisions may yield a solution that places small fractional units to many fulfillment centers, especially when the expected demand for each product is small. In this case, it is not clear how we can recover integral placement quantities.

Placement Decisions. To make the placement decisions, we maximize our bounding function subject to the capacity constraints at the fulfillment centers. We show that this placement problem is APX-hard. Therefore, we resort to approximation strategies. We give two approximation strategies. First, we solve a continuous relaxation of the placement problem under the assumption that we can place fractional units at each fulfillment center. We give a careful rounding approach that obtains $1 - O\left(\frac{1}{(C_{\min} \wedge U_{\min})^{1/3}}\right)$-fraction of the optimal objective value of the continuous relaxation. Second, we show that our bounding function is monotone and DR-submodular in the placement quantities, so the placement problem can be cast as a problem of maximizing a monotone and DR-submodular function subject to two partition constraints. Considering the problem of maximizing a monotone and submodular set function under k partition constraints, for any $\epsilon > 0$, Lee et al. (2010) give an algorithm to obtain a $\frac{1}{k+\epsilon}$-approximate solution. We formulate our placement problem as an equivalent monotone and submodular set function maximization problem, in which case, we can obtain a $\frac{1}{2+\epsilon}$-approximate solution for the placement problem.

Delivery Promise and Fulfillment Policy. Our bounding function uses a linear programming approximation for the delivery promise and fulfillment problem. Our delivery promise and fulfillment policy is a randomized policy that follows the optimal solution to this linear program. Letting $\widehat{z}_{\min}$ be the smallest nonzero number of units that we place for any product at any fulfillment center, we show that the randomized policy obtains at least a $\max\left\{\frac{1}{2},\, 1-O\left(\sqrt{\frac{\log \widehat{z}_{\min}}{\widehat{z}_{\min}}}\right)\right\}$-fraction of the optimal total expected profit. We also show that we can use the recent work on the magician problem to get a policy with a slightly improved performance guarantee of $\max\left\{\frac{1}{2},\, 1-O\left(\frac{1}{\sqrt{\widehat{z}_{\min}}}\right)\right\}$. The value of $\widehat{z}_{\min}$ is not known before we make the placement decisions, but we show that if we make the placement decisions by rounding the optimal solution to the continuous relaxation as discussed in the previous paragraph, then $\widehat{z}_{\min} \geq (C_{\min} \wedge U_{\min})^{2/3}$, so $1-\sqrt{\frac{\log \widehat{z}_{\min}}{\widehat{z}_{\min}}} = 1-O\left(\frac{\sqrt{\log(C_{\min} \wedge U_{\min})}}{(C_{\min} \wedge U_{\min})^{1/3}}\right)$, yielding a performance guarantee in terms of the exogenous problem primitives.

Related Literature: There is recent work on coordinating offline resource allocation with online rationing. One of the first papers in this area is Chen et al. (2022), in which the authors choose the initial stocking quantities for the products and match each arriving customer to a product. For their most general variant, making stocking decisions requires solving an integer program. Lei et al. (2023) study a joint stocking and pricing problem in an inventory distribution system. If the expected demands for all products are scaled linearly with rate μ, then they give a performance guarantee of $1-O\left(\sqrt{\frac{\log \mu}{\mu}}\right)$. DeValve et al. (2023) consider a joint stocking and fulfillment problem in online retail and use a deterministic approximation similar to our bounding function. If the length of the selling horizon and initial inventories of all products are scaled linearly with rate μ, then their fulfillment policy has a performance guarantee of $1-O\left(\frac{1}{\sqrt{\mu}}\right)$, but they do not establish a similar performance guarantee when the initial stocking quantities are also decision variables.

Our asymptotic optimality result has a nuanced difference from the corresponding results in the papers above. Considering our performance guarantee of $1-O\left(\frac{\sqrt{\log(C_{\min} \wedge U_{\min})}}{(C_{\min} \wedge U_{\min})^{1/3}}\right)$, as long as the total number of units to be placed for each product and capacities of the fulfillment centers get large, we obtain asymptotically optimal solutions. In particular, the expected demands for some products from some demand regions can stay small as the total number of units to be placed for each product and capacities of the fulfillment centers get large. The asymptotic optimality results in the previous paragraph either directly assume that the expected demands for all products get large or the number of time periods in the selling horizon get large, in which case, the expected demands for all products also get large. For our problem, if the expected demands for all products from all demand regions, numbers of units to place for all products and capacities of all fulfillment centers are scaled with rate μ, then we can use a standard fluid approximation to also obtain a performance guarantee of $1-O\left(\sqrt{\frac{\log \mu}{\mu}}\right)$. We will come back to this result later in the paper.

Our work is related to the growing literature on fulfillment in online retail. Acimovic and Graves (2015) give a dynamic programming model for fulfillment by using a linear program to approximate the value functions. Jasin and Sinha (2015) give a fulfillment policy that is asymptotically optimal as the inventories and expected demand increase linearly with the same rate. The authors can handle multi-item orders. We focus on single-item orders. Lei et al. (2022) build a joint product display, pricing and fulfillment model with performance guarantees. Andrews et al. (2019) work with adversarial demands. Wei et al. (2021) give a dynamic program to consolidate shipments with different delivery time slacks. The papers in this paragraph so far assume that the initial product placements are exogenously fixed. Chen and Graves (2021) give an integer program to choose the initial locations of the units under deterministic demand. Lim et al. (2021) give a robust placement model that protects against the worst case demand. DeValve et al. (2022) study a joint network design and distribution model, where all distribution decisions occur in a single stage.

Our bounding functions approximate the delivery promise and fulfillment problem through a linear program. Such approximations are used for revenue management problems with fixed initial resource inventories; see Talluri and van Ryzin (1998), Cooper (2002), Gallego et al. (2004), Liu and van Ryzin (2008), Jasin and Kumar (2012). Using a linear program to guide initial resource allocation decisions, as we do, may have other applications. Subsequent to us, Bai et al. (2022) focus on joint inventory stocking and assortment personalization. Our approximate policy approximates the delivery promise and fulfillment problem, which is an online resource allocation problem. There is other work in approximating similar problems in revenue management; see Golrezaei et al. (2014), Gallego et al. (2015), Wang et al. (2016), Feng et al. (2020), Stein et al. (2020), Rusmevichientong et al. (2020), Ma et al. (2020), Manshadi and Rodilitz (2022), Baek and Ma (2022). There is also recent work on the so-called magician problem that uses prophet inequalities to approximate online resource allocation problems; see Alaei (2014), Amil et al. (2022), Jiang et al. (2023).

Organization: In Section 2, we formulate the joint placement, delivery promise and fulfillment problem. In Section 3, we describe our approximation strategy and give our main performance guarantee. In Section 4, we describe our bounding function. In Section 5, we focus on making the placement decisions by using rounding on a continuous relaxation. In Section 6, we exploit DR-submodularity to make the placement decisions. In Section 7, we give our delivery promise and fulfillment policy. In Section 8, we give computational experiments. In Section 9, we conclude.

2. Problem Formulation

There are n fulfillment centers indexed by $\mathcal{F}=\{1,\ldots,n\}$. The storage capacity of fulfillment center i is U_i units. There are m products indexed by $\mathcal{A}=\{1,\ldots,m\}$. We have C^a units of product a to place over all fulfillment centers. There are T time periods in the selling horizon indexed by $\mathcal{T}=\{1,\ldots,T\}$. For each product, there is at most one demand at each time period. The set of demand regions is $\mathcal{D}$. At time period t, we have a demand for product a in demand region j with probability λ^a_{jt}, where we have $\sum_{j\in\mathcal{D}}\lambda^a_{jt}\le 1$. We offer a delivery promise in response to each demand. The set of possible delivery promises is $\mathcal{K}$. If we offer delivery promise k to a demand for product a in demand region j, then the customer accepts the promise with probability θ^a_{jk}, in which case, if we use fulfillment center i to serve this demand, then we generate a profit of r^a_{ijk}. By convention, there is a dummy promise κ that the customer never accepts, so $\theta^a_{j\kappa}=0$ for all $a\in\mathcal{A}$ and $j\in\mathcal{D}$. Offering promise κ corresponds to displaying the demanded product as unavailable.

Our goal is to decide how many units of each product to place at each fulfillment center at the beginning of the selling horizon and which delivery promise and fulfillment center to use to serve each demand over the selling horizon so that we maximize the total expected profit. We give a dynamic program to characterize the optimal delivery promise and fulfillment policy for each product. We use the vector $\boldsymbol{x}^a=(x^a_1,\ldots,x^a_n)\in\mathbb{Z}^n_+$ as the state variable at the beginning of a generic time period, where x^a_i is the remaining inventory of product a at fulfillment center i. Let $J^a_t(\boldsymbol{x}^a)$ be the optimal total expected profit from the delivery promise and fulfillment decisions for product a over time periods $\{t,\ldots,T\}$ given that the remaining inventories for the product at the beginning of time period t correspond to the vector $\boldsymbol{x}^a$. Letting $\boldsymbol{e}_i\in\mathbb{Z}^n_+$ be the unit vector with a one at fulfillment center i and $\mathbb{1}$ be the indicator function, using the boundary condition $J^a_{T+1}=0$, we can compute the value functions $\{J^a_t : t\in\mathcal{T}\}$ by solving the dynamic program

$$
\begin{aligned}
J^a_t(\boldsymbol{x}^a) \;&=\; \sum_{j\in\mathcal{D}}\lambda^a_{jt}\max_{(i,k)\in\mathcal{F}\times\mathcal{K}}\left\{\mathbb{1}_{(x^a_i\ge 1)}\,\theta^a_{jk}\left[r^a_{ijk}+J^a_{t+1}(\boldsymbol{x}^a-\boldsymbol{e}_i)\right]+\left[1-\theta^a_{jk}\,\mathbb{1}_{(x^a_i\ge 1)}\right]J^a_{t+1}(\boldsymbol{x}^a)\right\}\\
&\qquad+\left\{1-\sum_{j\in\mathcal{D}}\lambda^a_{jt}\right\}J^a_{t+1}(\boldsymbol{x}^a)\\
&=\sum_{j\in\mathcal{D}}\lambda^a_{jt}\max_{(i,k)\in\mathcal{F}\times\mathcal{K}}\left\{\mathbb{1}_{(x^a_i\ge 1)}\,\theta^a_{jk}\left[r^a_{ijk}+J^a_{t+1}(\boldsymbol{x}^a-\boldsymbol{e}_i)-J^a_{t+1}(\boldsymbol{x}^a)\right]\right\}+J^a_{t+1}(\boldsymbol{x}^a). \quad \text{(Fulfillment)}
\end{aligned}
$$

We compute the delivery promise and fulfillment policy separately for each product, but the products will interact due to limited capacity at the fulfillment centers when placing them.

In the dynamic program above, the decision at each time period is the delivery promise that we offer to the arriving customer, along with the fulfillment center that we plan to use if the promise is accepted. In the first equality, at time period t, we have a demand for product a from demand region j with probability λ^a_{jt}. Given that we offer delivery promise k to this demand and plan to use fulfillment center i upon acceptance of the promise, if the customer accepts the delivery promise and we have inventory at the fulfillment center, then we generate a profit of r^a_{ijk} and consume one unit of inventory at fulfillment center i. Note that we immediately deduct the inventory that we plan to use to serve the demand. The second equality follows by rearranging the terms. We use the vector $\boldsymbol{z}^a = (z^a_i : i \in \mathcal{F}) \in \mathbb{Z}^n_+$ to capture the placement decisions for product i, where z^a_i is the number of units of product a that we place at fulfillment center i at the beginning of the selling horizon. Given that we make the placement decisions $\boldsymbol{z}^a$ for product a, the optimal total expected profit from the product over the selling horizon is $J^a_1(\boldsymbol{z}^a)$. To find the optimal placement for the products at the beginning of the selling horizon, using the decision variables $\boldsymbol{z} = (z^a_i : i \in \mathcal{F},\ a \in \mathcal{A}) \in \mathbb{Z}^{n\times m}_+$ to capture the inventory placement decisions for all products, we can solve the problem

$$\zeta_{\mathsf{OPT}} \;=\; \max_{\boldsymbol{z}\in\mathbb{Z}^{n\times m}_+} \left\{ \sum_{a\in\mathcal{A}} J^a_1(\boldsymbol{z}^a) \;:\; \sum_{i\in\mathcal{F}} z^a_i = C^a \quad \forall\, a\in\mathcal{A}, \quad \sum_{a\in\mathcal{A}} z^a_i \le U_i \quad \forall\, i\in\mathcal{F} \right\}. \qquad \text{(Placement)}$$

The first constraint ensures that we place all available units for each product. The second constraint ensures that we do not violate the capacities at the fulfillment centers.

We work with a discrete set of delivery promises. A delivery promise, for example, can take the form of 1-day, 2-day or 1-week delivery. The retailer chooses the delivery promise to offer to a customer, in response to which, the customer decides whether to make a purchase. The delivery promise affects the probability that a customer converts from viewing the product to purchasing, as well as the corresponding shipping cost. Accordingly, the profit from serving a demand can depend on the offered promise, the demand region where the demand occurs and the fulfillment center from which we dispatch the inventory. Noting that the delivery promise affects the probability that the sale is realized along with the profit from the sale, there is no technical difference between promise and pricing decisions. Thus, our model can be useful to coordinate placement, pricing and fulfillment decisions, thereby relating our model to other formulations for coordinating pricing with stocking, fulfillment, matching and resource repositioning; see, for example, Lei et al. (2023), Feng et al. (2023) and Chen et al. (2023). When making a delivery promise decision, geographical locations of the inventories also play role, because we need to have inventory available at a fulfillment center close enough to fulfill the promise. In our formulation, if fulfillment center i is too far from demand region j so that we cannot use fulfillment center i to serve the demands from demand region j for product a with delivery promise k, then we set $r^a_{ijk} = -\infty$.

In our model, for each product a, we focus on placing the total amount of C^a units, which has been dictated by the national target inventory level. Thus, the profit from a sale includes the revenue from the sale and shipping cost, but not the procurement cost. Shipping cost can be a significant portion of the revenue and programs such as Amazon Prime requires the retailer to bear the shipping cost, so it is critical to consider the shipping cost. Even if the shipping cost is a small portion of the revenue, the choice of the delivery promise and fulfillment center to serve a demand involves nontrivial tradeoffs. The decisions for the current demand dictate from which fulfillment center we use inventory to serve the demand, which have implications on choosing a delivery promise and fulfillment center to serve future demands. We assume that $\sum_{i\in\mathcal{F}} U_i \geq \sum_{a\in\mathcal{A}} C^a$, so we have enough capacity in the fulfillment centers to place all units.

One data requirement for our model is the probability λ^a_{jt} that we have a demand for product a in demand region j at time period t. Online retailers have a good understanding of the geographical spread of the demand, as well as the distribution of the national demand. Thus, estimating the total expected demand for product a from demand region j, given by $\sum_{t\in\mathcal{T}} \lambda^a_{jt}$, is certainly within the capabilities of online retailers. Making the placement decisions in our model will require knowing only the total expected demand for each product from each demand region, rather than the detailed evolution of the demand over time. Estimating the total expected demand is more tractable than estimating the detailed evolution of the demand over time.

Computing the objective function of the Placement problem even at fixed placement decisions $\boldsymbol{z} = (z^a_i : i\in\mathcal{F},\ a\in\mathcal{A})$ requires computing the value functions $\{J^a_1 : a\in\mathcal{A}\}$, which, in turn, requires solving a high-dimensional dynamic program. Even if we have access to an oracle that computes each of the value functions $\{J^a_1 : a\in\mathcal{A}\}$ at fixed placement decisions, the objective function of the Placement problem does not necessarily have structural properties that facilitate solving this problem efficiently. One such property is DR-submodularity. The function $f^a : \mathbb{Z}^n_+ \to \mathbb{R}_+$ is DR-submodular if $f^a(\boldsymbol{z}^a + \boldsymbol{e}_i) - f^a(\boldsymbol{z}^a) \leq f^a(\boldsymbol{y}^a + \boldsymbol{e}_i) - f^a(\boldsymbol{y}^a)$ for each $i\in\mathcal{F}$ and $\boldsymbol{z}^a, \boldsymbol{y}^a \in \mathbb{Z}^n_+$ with $\boldsymbol{z}^a \geq \boldsymbol{y}^a$. In the last inequality, we write $\boldsymbol{z}^a \geq \boldsymbol{y}^a$ if and only if $z^a_i \geq y^a_i$ for all $i\in\mathcal{F}$. It is not difficult to come up with counterexamples to demonstrate that the value functions $\{J^a_1 : a\in\mathcal{A}\}$ are not DR-submodular. Consider a problem instance with three fulfillment centers $\mathcal{F} = \{1,2,3\}$, three demand regions $\mathcal{D} = \{1,2,3\}$ and $T = 3$ time periods in the selling horizon. Other than the dummy promise, there is one delivery promise and the probability of accepting this delivery promise is one. Focusing on some product a and indexing the single delivery promise by $k = 1$, we give the profits, demand arrival probabilities and value functions in Table 1. For this problem instance, consider the state vectors $\boldsymbol{z}^a = (0,1,1)$ and $\boldsymbol{y}^a = (0,0,1)$, so $\boldsymbol{z}^a \geq \boldsymbol{y}^a$. We can numerically verify that $J^a_1(\boldsymbol{z}^a + \boldsymbol{e}_1) - J^a_1(\boldsymbol{z}^a) > 20.05 > J^a_1(\boldsymbol{y}^a + \boldsymbol{e}_1) - J^a_1(\boldsymbol{y}^a)$. Thus, the value functions $\{J^a_1 : a\in\mathcal{A}\}$ are not DR-submodular. Motivated by these observations, we focus on obtaining approximate solutions to the Placement problem by using approximations to the objective function of this problem.

$(r^a_{ij1} : i \in \mathcal{F},\ j \in \mathcal{D})$

i	$j=1$	$j=2$	$j=3$
1	16	5	25
2	12	21	3
3	37	50	5

$(\lambda^a_{jt} : j \in \mathcal{D},\ t \in \mathcal{T})$

t	$j=1$	$j=2$	$j=3$
1	0.31	0.28	0.41
2	0.30	0.45	0.25
3	0.28	0.49	0.23

$(J^a_t(\boldsymbol{x}^a) : \boldsymbol{x}^a \in \{0,1\}^{|\mathcal{F}|},\ t \in \mathcal{T})$

t	$\boldsymbol{x}^a$ = (0,0,1)	(0,1,0)	(0,1,1)	(1,0,0)	(1,0,1)	(1,1,0)	(1,1,1)
1	44.674	18.363	57.965	20.136	64.718	38.499	78.019
2	42.603	17.337	54.108	16.756	59.062	34.093	64.243
3	36.010	14.340	36.010	12.680	40.610	20.520	40.610

Table 1 Problem parameters and value functions for the counterexample to the DR-submodularity of value functions.

3. Main Results

The optimal objective value of the Placement problem is ζ_{OPT}, so if we make the optimal placement decisions at the beginning of the selling horizon and subsequently follow the optimal delivery promise and fulfillment policy over the selling horizon, then we obtain a total expected profit of ζ_{OPT}. Our goal is to find approximate placement decisions, along with an approximate delivery promise and fulfillment policy, such that the total expected profit obtained by making the approximate placement decisions and following the approximate delivery promise and fulfillment policy has a bounded relative loss compared to ζ_{OPT}. We use the following three steps to achieve this goal.

Step 1. Bounding Function. For each product a, construct a bounding function $f^a : \mathbb{Z}^n_+ \to \mathbb{R}_+$ such that $f^a(\boldsymbol{z}^a) \geq J^a_1(\boldsymbol{z}^a)$ for each $\boldsymbol{z}^a \in \mathbb{Z}^n_+$. Thus, the bounding function upper bounds the optimal total expected profit when viewed as a function of the initial placement decisions.

Step 2. Approximate Placement. Replacing $\sum_{a\in\mathcal{A}} J^a_1(\boldsymbol{z}^a)$ in the Placement problem with $\sum_{a\in\mathcal{A}} f^a(\boldsymbol{z}^a)$, for some $\alpha \in (0,1]$, compute the placement decisions $\boldsymbol{z}^{\mathsf{APX}} = (z^{a,\mathsf{APX}}_i : i \in \mathcal{F},\ a \in \mathcal{A})$ for all products as an α-approximate solution to the problem

$$\max_{\boldsymbol{z}\in\mathbb{Z}^{n\times m}_+} \left\{ \sum_{a\in\mathcal{A}} f^a(\boldsymbol{z}^a) : \sum_{i\in\mathcal{F}} z^a_i = C^a \quad \forall\, a \in \mathcal{A},\ \sum_{a\in\mathcal{A}} z^a_i \leq U_i \quad \forall\, i \in \mathcal{F} \right\}. \quad (\textsf{Approximate Placement})$$

Step 3. Approximate Fulfillment. For each product a, construct a delivery promise and fulfillment policy $\pi^{a,\mathsf{APX}}$ such that if we start with the placement decisions $\boldsymbol{z}^{a,\mathsf{APX}} = (z^{a,\mathsf{APX}}_i : i \in \mathcal{F})$, then the policy obtains a total expected profit of at least $\beta\, f^a(\boldsymbol{z}^{a,\mathsf{APX}})$ for some $\beta \in (0,1]$.

In Step 2, we make the placement decisions by using a variant of the Placement problem that uses the bounding function in its objective function. In Step 3, we construct a delivery promise and fulfillment policy such that the total expected profit of the policy can be lower bounded by using the bounding function. It is not difficult to see that if we make the placement decisions computed in Step 2 and follow the delivery promise and fulfillment policy constructed in Step 3, then we obtain a total expected profit of at least $\alpha\,\beta\,\zeta_{\mathsf{OPT}}$. In particular, letting $\boldsymbol{z}^{\mathsf{OPT}} = (z^{a,\mathsf{OPT}}_i : i \in \mathcal{F},\ a \in \mathcal{A})$ be an optimal solution to the Placement problem, we have $\zeta_{\mathsf{OPT}} = \sum_{a\in\mathcal{A}} J^a_1(\boldsymbol{z}^{a,\mathsf{OPT}})$, where we use

$\boldsymbol{z}^{a,\mathsf{OPT}} = (z_i^{a,\mathsf{OPT}} : i \in \mathcal{F})$. By Step 1, we have $\sum_{a\in\mathcal{A}} f^a(\boldsymbol{z}^{a,\mathsf{OPT}}) \geq \sum_{a\in\mathcal{A}} J_1^a(\boldsymbol{z}^{a,\mathsf{OPT}})$. Note that $\boldsymbol{z}^{\mathsf{OPT}}$ is a feasible solution to the Approximate Placement problem, but by Step 2, $\boldsymbol{z}^{\mathsf{APX}}$ is an α-approximate solution, so $\sum_{a\in\mathcal{A}} f^a(\boldsymbol{z}^{a,\mathsf{APX}}) \geq \alpha \sum_{a\in\mathcal{A}} f^a(\boldsymbol{z}^{a,\mathsf{OPT}})$. Letting $\mathsf{Prof}^{a,\mathsf{APX}}(\boldsymbol{z}^a)$ be the total expected profit obtained by the policy $\pi^{a,\mathsf{APX}}$ from product a given that placement decisions for the product correspond to the vector $\boldsymbol{z}^a$, by Step 3, $\sum_{a\in\mathcal{A}} \mathsf{Prof}^{a,\mathsf{APX}}(\boldsymbol{z}^{a,\mathsf{APX}}) \geq \beta \sum_{a\in\mathcal{A}} f^a(\boldsymbol{z}^{a,\mathsf{APX}})$. In this case, using the preceding three inequalities in this paragraph, we have

$$\sum_{a\in\mathcal{A}} \mathsf{Prof}^{a,\mathsf{APX}}(\boldsymbol{z}^{a,\mathsf{APX}}) \;\geq\; \beta \sum_{a\in\mathcal{A}} f^a(\boldsymbol{z}^{a,\mathsf{APX}}) \;\geq\; \alpha\,\beta \sum_{a\in\mathcal{A}} f^a(\boldsymbol{z}^{a,\mathsf{OPT}}) \;\geq\; \alpha\,\beta \sum_{a\in\mathcal{A}} J_1^a(\boldsymbol{z}^{a,\mathsf{OPT}}) \;=\; \alpha\,\beta\,\zeta_{\mathsf{OPT}}.$$

Thus, making the placement decisions $\boldsymbol{z}^{a,\mathsf{APX}}$ from Step 2 for each product a and following the delivery promise and fulfillment policy $\pi^{a,\mathsf{APX}}$ from Step 3 for each product a, we obtain a total expected profit of at least $\alpha\,\beta\,\zeta_{\mathsf{OPT}}$. We cap this simple but useful result in the next remark.

Remark 3.1 (Approximation Strategy) *For each product a, if we make the placement decisions $\boldsymbol{z}^{a,\mathsf{APX}}$ from Step 2 and use the delivery promise and fulfillment policy $\pi^{a,\mathsf{APX}}$ from Step 3, then the total expected profit that we obtain from all products is at least $\alpha\,\beta\,\zeta_{\mathsf{OPT}}$.*

Using the remark above, we can get a performance guarantee, but we need to answer three questions to be able to follow the three steps of our approximation strategy. In Step 1, we need to come up with a bounding function that provides an upper bound on the optimal total expected profit from the delivery promise and fulfillment decisions for each product. In Step 2, we need to obtain an α-approximate solution to a variant of the Placement problem that uses our bounding function in the objective function. In Step 3, we need to construct a delivery promise and fulfillment policy for each product such that we can lower bound the total expected profit of the policy by a β-fraction of the bounding function. We show that we can execute all of these steps with a specific bounding function and specific values of α and β, in which case, we obtain placement decisions, along with a delivery promise and fulfillment policy, providing a certain performance guarantee for our joint placement, delivery promise and fulfillment problem. We proceed to giving the specific values for α and β that we can achieve and the corresponding performance guarantee.

Main Performance Guarantee:

We will give a bounding function based on approximating the optimal total expected profit from the delivery promise and fulfillment decisions by using a linear program. This linear program is formulated under the assumption that the demands and responses to the delivery promises take on their expected values. It provides an upper bound on the optimal total expected profit from the delivery promise and fulfillment decisions, as needed in Step 1. We will always use this bounding function. We will show that we can execute Step 2 with $\alpha = \frac{1}{2+\epsilon}$ for any $\epsilon > 0$ and Step 3 with $\beta = \frac{1}{2}$, yielding an $\alpha\beta = \frac{1}{4+\epsilon}$-approximate solution. Using $C_{\min} = \min_{a\in\mathcal{A}} C^a$ to capture the smallest total number of units to be placed for a product and $U_{\min} = \min_{i\in\mathcal{F}} U_i$ to capture the smallest capacity of a fulfillment center, we will also show that we can execute Step 2 with $\alpha = 1 - O\Big(\frac{1}{(C_{\min}\wedge U_{\min})^{1/3}}\Big)$ and Step 3 with $\beta = 1 - O\Big(\sqrt{\frac{\log(C_{\min}\wedge U_{\min})}{(C_{\min}\wedge U_{\min})^{2/3}}}\Big)$, yielding an $\alpha\beta = 1 - O\Big(\frac{\sqrt{\log(C_{\min}\wedge U_{\min})}}{(C_{\min}\wedge U_{\min})^{1/3}}\Big)$-approximate solution. In this case, we obtain the following performance guarantee.

Theorem 3.2 (Performance) *For any $\epsilon > 0$, we can find placement decisions, along with a delivery promise and fulfillment policy, in polynomial time with a total expected profit of at least*

$$\max\left\{\frac{1}{4+\epsilon}\,,\, 1 - O\left(\frac{\sqrt{\log(C_{\min}\wedge U_{\min})}}{(C_{\min}\wedge U_{\min})^{1/3}}\right)\right\}\zeta_{\mathsf{OPT}}.$$

We give a proof for the theorem above at the end of Section 7, after we give the details of how we can execute Steps 2 and 3 of our approximation strategy. The proof is based on developing ways to execute Steps 2 and 3 with specific values for α and β. By Theorem 3.2, for any $\epsilon > 0$, we can obtain a solution with a constant factor guarantee of $\frac{1}{4+\epsilon}$, but if the numbers of units we need to place and capacities of the fulfillment centers get large, then the solution is asymptotically optimal. There is work on asymptotically optimal control policies for revenue management problems when the capacity of each resource and total expected demand are both scaled with the same rate; see Gallego and van Ryzin (1994) and Jasin and Kumar (2012). The performance guarantee in Theorem 3.2 does not make any assumptions on the total expected demand. Also, even if the number of units that we need to place is large, it is not a priori clear that we will place a large number of units at each fulfillment center. Thus, having a large number of units to place does not always translate into having a large number of units at each fulfillment center. Consider two fulfillment centers and two demand regions, where fulfillment center i_1 can only serve demand region j_1 and fulfillment center i_2 can only serve demand region j_2. The demand from demand region j_1 is one, whereas the demand from demand region j_2 is K. The profit from serving the demand from demand regions j_1 and j_2 are, respectively, K and one. We have $1 + K$ units to place. The optimal total expected profit is $2K$, but placing $\Omega(K)$ units at fulfillment center i_1 loses a constant fraction of the optimal total expected profit. Next, we give our bounding function, as needed in Step 1.

4. Constructing the Bounding Function

In our bounding function, we use a linear program to approximate the optimal total expected profit from the delivery promise and fulfillment decisions for each product. We use the decision variables $\boldsymbol{w}^a = (w^a_{ijk} : i \in \mathcal{F},\ j \in \mathcal{D},\ k \in \mathcal{K}) \in \mathbb{R}_+^{n\times|\mathcal{D}|\times|\mathcal{K}|}$, where w^a_{ijk} is the total expected number of times that we offer delivery promise k to a demand for product a from demand region j and plan to use fulfillment center i upon acceptance of the promise. Given the initial inventory placement decisions for product a correspond to the vector $\boldsymbol{z}^a$, we approximate the optimal total expected profit from the delivery promise and fulfillment decisions by using the optimal objective value of the problem

$$f^a_{\mathsf{LP}}(\boldsymbol{z}^a) \;=\; \max_{\boldsymbol{w}^a \in \mathbb{R}_+^{n\times|\mathcal{D}|\times|\mathcal{K}|}} \left\{ \sum_{i\in\mathcal{F}}\sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}} r^a_{ijk}\,\theta^a_{jk}\,w^a_{ijk} \;:\; \sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}} \theta^a_{jk}\,w^a_{ijk} \le z^a_i \quad \forall\, i\in\mathcal{F}, \right.$$
$$\left. \sum_{i\in\mathcal{F}}\sum_{k\in\mathcal{K}} w^a_{ijk} \le \sum_{t\in\mathcal{T}} \lambda^a_{jt} \quad \forall\, j\in\mathcal{D} \right\}. \qquad (\mathsf{Bounding})$$

The Bounding problem is an approximation to the delivery promise and fulfillment problem for product a formulated under the assumption that the demand for the product and promise acceptance decisions take on their expected values. The first constraint ensures that the total expected number of times that the offered delivery promise for product a is accepted and uses the inventory at fulfillment center i does not exceed the initial inventory placement at the fulfillment center. The second constraint ensures that the total expected number of times that we offer some delivery promise to a demand for product a from demand region j and plan to use some fulfillment center does not exceed the total expected demand for the product from demand region j. Throughout the paper, we will always work with the bounding function f^a_{LP} given by the optimal objective value of the Bounding problem. The subscript in f^a_{LP} emphasizes that we compute our bounding function through a linear program. The Bounding problem is a fluid approximation to the control problem formulated in the Fulfillment dynamic program. We can show that the optimal objective value of the Bounding problem is an upper bound on the optimal total expected profit from a product, which implies that our bounding function f^a_{LP} satisfies the requirement in Step 1. We give this result in the next lemma and defer the standard proof to Appendix A.

Lemma 4.1 (Upper Bound) *For each product a, letting the value functions $\{J^a_t : t\in\mathcal{T}\}$ be computed through the* Fulfillment *dynamic program, we have $f^a_{\mathsf{LP}}(\boldsymbol{z}^a) \ge J^a_1(\boldsymbol{z}^a)$ for all $\boldsymbol{z}^a \in \mathbb{Z}^n_+$.*

Our bounding function needs to satisfy three requirements. First, the bounding function should provide an upper bound on the optimal total expected profit from the delivery promise and fulfillment decisions. Second, we should be able to obtain approximate solutions to the Approximate Placement problem efficiently when we use the bounding function in this problem. Third, we should be able to construct a delivery promise and fulfillment policy such that the total expected profit of the policy is lower bounded by a certain fraction of the bounding function. The bounding function given by the Bounding problem will satisfy these requirements. There is other work to approximate value functions, providing other candidates for bounding functions. Rusmevichientong et al. (2020), Ma et al. (2020), Baek and Ma (2022) and Manshadi and Rodilitz (2022) construct value function approximations for revenue management problems under exogenously given initial product inventories. Their value function approximations satisfy the first and third requirements in the sense that one can use their value function approximations to upper bound the optimal total expected reward starting with given initial product inventories, as well as construct an approximate policy such that the total expected reward of the approximate policy can be lower bounded by a fraction of their value function approximations. However, the way their value function approximations are constructed depends on the initial product inventories in a complicated fashion and it is not clear how to use these approximations in the Approximate Placement problem to choose the initial product inventories. Thus, their value function approximations do not satisfy the second requirement. Kunnumkal and Talluri (2016) and Brown and Zhang (2023) build value function approximations by decomposing the underlying dynamic program. Their approximations satisfy the first, but not the second and third requirements. Aouad and Saritac (2022) use the average cost criterion, so their value function approximations do not depend on the initial state.

With the bounding function given by the optimal objective value of the Bounding problem in place, we turn to executing Steps 2 and 3, which are the more challenging tasks.

5. Approximating the Placement Problem: Complexity and Rounding

We consider solving the Approximate Placement problem with the bounding function f^a_{LP} given by the linear program in the Bounding problem. It turns out that this problem is APX-hard when we use the bounding function f^a_{LP}. Motivated by this complexity result, we focus on obtaining approximate solutions. In this section, we use rounding on the optimal solution of a continuous relaxation of the Approximate Placement problem to obtain a $1-O\Big(\frac{1}{(C_{\min} \wedge U_{\min})^{1/3}}\Big)$-approximate solution. This continuous relaxation is efficient to solve as it takes the form of a linear program. In the next section, we show that $f^a_{\mathsf{LP}}(\boldsymbol{z}^a)$ is monotone and DR-submodular in $\boldsymbol{z}^a$, in which case, the Approximate Placement problem takes the form of maximizing a monotone and DR-submodular function subject to two partition constraints. For such a problem, we can obtain a solution with a performance guarantee of $\frac{1}{2+\epsilon}$ for any $\epsilon > 0$. In the next theorem, we show that if we use the bounding function f^a_{LP}, then the Approximate Placement problem is APX-hard even when we have only one product and one delivery promise other than the dummy promise.

Theorem 5.1 (Complexity) *Considering the* Approximate Placement *problem with the bounding function f^a_{LP}, unless $P = NP$, there is no polynomial-time approximation scheme for this problem even when there is one product and one non-dummy delivery promise.*

The proof of the theorem is in Appendix B. It uses a reduction from the maximum vertex cover problem. Theorem 5.1 also has an implication on the possibility of using another bounding function. The right side of the second constraint in the Bounding problem is the total expected demand for product a from demand region j. Alternatively, we can use a bounding function that uses an expected offline bound, where we replace the right side of the second constraint with a realization of the total demand for product a from demand region j. In this case, the optimal objective value of the Bounding problem would be a function of the realizations of the total demand for product a from different demand regions. Computing an expectation over all possible demand realizations, we still obtain an upper bound on the optimal total expected profit from the delivery promise and fulfillment decisions. A standard argument shows that this upper bound is at least as tight as the one provided by the optimal objective value of the Bounding problem, which uses the total expected demands from different demand regions; see Talluri and van Ryzin (1999). In the proof of Theorem 5.1, we use a problem instance with deterministic demands. If the demands are deterministic, then the expected offline bound is precisely given by the Bounding problem. Thus, the Approximate Placement problem is still APX-hard if we use an expected offline bound.

We turn our attention to obtaining an approximate solution to the Approximate Placement problem when we use the bounding function f^a_{LP}. We use rounding for this purpose.

Rounding a Continuous Relaxation:

Using the decision variable w^a_{ijk} with the same interpretation as in the Bounding problem and z^a_i with the same interpretation as in the Approximate Placement problem, consider the problem

$$\max_{(\boldsymbol{z},\boldsymbol{w})\in\mathbb{R}_+^{n\times m\times(1+|\mathcal{D}|\times|\mathcal{K}|)}} \Bigg\{ \sum_{a\in\mathcal{A}}\sum_{i\in\mathcal{F}}\sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}} r^a_{ijk}\,\theta^a_{jk}\,w^a_{ijk} \;:\; \sum_{i\in\mathcal{F}} z^a_i \le C^a \quad \forall\, a\in\mathcal{A},\;\; \sum_{a\in\mathcal{A}} z^a_i \le U_i \quad \forall\, i\in\mathcal{F},$$

$$\sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}} \theta^a_{jk}\,w^a_{ijk} \le z^a_i \quad \forall\, i\in\mathcal{F},\; a\in\mathcal{A},\;\; \sum_{i\in\mathcal{F}}\sum_{k\in\mathcal{K}} w^a_{ijk} \le \sum_{t\in\mathcal{T}} \lambda^a_{jt} \quad \forall\, j\in\mathcal{D},\; a\in\mathcal{A} \Bigg\}, \qquad (\mathsf{Relaxed})$$

where we use the vectors $\boldsymbol{z}=(z^a_i : i\in\mathcal{F},\; a\in\mathcal{A})$ and $\boldsymbol{w}=(w^a_{ijk} : i\in\mathcal{F},\; j\in\mathcal{D},\; k\in\mathcal{K},\; a\in\mathcal{A})$. The linear program above is a continuous relaxation of the Approximate Placement problem when we use the bounding function f^a_{LP} in the Approximate Placement problem. In particular, consider solving the Relaxed problem after fixing the values of the decision variables $\boldsymbol{z}=(z^a_i : i\in\mathcal{F},\; a\in\mathcal{A})$ at $\widehat{\boldsymbol{z}}=(\widehat{z}^a_i : i\in\mathcal{F},\; a\in\mathcal{A})$. Comparing the last two constraints in the problem above with the two constraints in the Bounding problem, if we solve the Relaxed problem after fixing the values of the decision variables $\boldsymbol{z}$ at $\widehat{\boldsymbol{z}}$, then the optimal objective value of the Relaxed problem is $\sum_{a\in\mathcal{A}} f^a_{\mathsf{LP}}(\widehat{\boldsymbol{z}}^a)$. In this case, comparing the first two constraints in the problem above with the two constraints in the Approximate Placement problem, the Relaxed problem is equivalent to the Approximate Placement problem with the bounding function f^a_{LP}, but it allows its decision variables to take on fractional values. Thus, the Relaxed problem is a continuous relaxation of the Approximate Placement problem. In the next theorem, we use rounding on the Relaxed problem to approximate the Approximate Placement problem. In this theorem, we use $a\vee b=\max\{a,b\}$.

Theorem 5.2 (Rounding) *Assume that $C_{\min} \wedge U_{\min} \geq |\mathcal{F}| \vee |\mathcal{A}|$. Letting $(\overline{\boldsymbol{z}}, \overline{\boldsymbol{w}})$ be an optimal solution to the* Relaxed *problem, for any $\delta \in \left[1, \frac{C_{\min} \wedge U_{\min}}{|\mathcal{F}| \vee |\mathcal{A}|}\right] \cap \mathbb{Z}_+$, setting $\eta = 1 - \frac{|\mathcal{F}| \vee |\mathcal{A}|}{C_{\min} \wedge U_{\min}}\, \delta$, define the solution $\widehat{\boldsymbol{z}} = (\widehat{z}_i^a : i \in \mathcal{F},\ a \in \mathcal{A})$ as $\widehat{z}_i^a = \lfloor \eta\, \overline{z}_i^a \rfloor + \delta$ for all $i \in \mathcal{F}$ and $a \in \mathcal{A}$. Then, $\widehat{\boldsymbol{z}}$ is an η-approximate solution to the* Approximate Placement *problem with the bounding function f^a_{LP}.*

Proof: We make two claims. First, we claim that the solution $\widehat{\boldsymbol{z}}$ is feasible to the Approximate Placement problem. In particular, using the definition of $\widehat{z}_i^a$, for all $a \in \mathcal{A}$, we have

$$\sum_{i \in \mathcal{F}} \widehat{z}_i^a \;=\; \sum_{i \in \mathcal{F}} \lfloor \eta\, \overline{z}_i^a \rfloor + \sum_{i \in \mathcal{F}} \delta \;\stackrel{(a)}{\leq}\; \sum_{i \in \mathcal{F}} \left(1 - \frac{|\mathcal{F}|}{C_{\min}}\,\delta\right) \overline{z}_i^a + |\mathcal{F}|\,\delta$$
$$\stackrel{(b)}{\leq}\; \left(1 - \frac{|\mathcal{F}|}{C_{\min}}\,\delta\right) C^a + |\mathcal{F}|\,\delta \;=\; C^a + |\mathcal{F}|\,\delta\left(1 - \frac{C^a}{C_{\min}}\right) \;\stackrel{(c)}{\leq}\; C^a,$$

where (a) holds because the definition of η implies that $\eta \leq 1 - \frac{|\mathcal{F}|}{C_{\min}}\,\delta$, (b) uses the fact that $(\overline{\boldsymbol{z}}, \overline{\boldsymbol{w}})$ satisfies the first constraint in the Relaxed problem so that $\sum_{i \in \mathcal{F}} \overline{z}_i^a \leq C^a$ and (c) is by noting that $C_{\min} \leq C^a$. Through a similar argument, we can show that $\widehat{\boldsymbol{z}}$ satisfies the second constraint in the Approximate Placement problem as well. Furthermore, $\widehat{z}_i^a$ is an integer, so the claim holds. Noting that $(\overline{\boldsymbol{z}}, \overline{\boldsymbol{w}})$ is an optimal solution to the Relaxed problem, define $\widehat{w}_{ijk}^a = \eta\, \overline{w}_{ijk}^a$. Second, using the vectors $\widehat{\boldsymbol{z}}^a = (\widehat{z}_i^a : i \in \mathcal{F})$ and $\widehat{\boldsymbol{w}}^a = (\widehat{w}_{ijk}^a : i \in \mathcal{F},\ j \in \mathcal{D},\ k \in \mathcal{K})$, we claim that $\widehat{\boldsymbol{w}}^a$ is feasible to the Bounding problem when we solve this problem with $\boldsymbol{z}^a = \widehat{\boldsymbol{z}}^a$. Note that we have $\eta \in [0,1]$ by our choice of δ in the theorem. Using the fact that $(\overline{\boldsymbol{z}}, \overline{\boldsymbol{w}})$ satisfies the third constraint in the Relaxed problem, we obtain $\sum_{j \in \mathcal{D}} \sum_{k \in \mathcal{K}} \theta_{jk}^a\, \widehat{w}_{ijk}^a = \sum_{j \in \mathcal{D}} \sum_{k \in \mathcal{K}} \theta_{jk}^a\, \eta\, \overline{w}_{ijk}^a \leq \eta\, \overline{z}_i^a \leq \lfloor \eta\, \overline{z}_i^a \rfloor + \delta = \widehat{z}_i^a$, where the last inequality holds since $a \leq \lfloor a \rfloor + 1$ and $\delta \geq 1$. Furthermore, using the fact that $\eta \in [0,1]$, we have $\sum_{i \in \mathcal{F}} \sum_{k \in \mathcal{K}} \widehat{w}_{ijk}^a = \sum_{i \in \mathcal{F}} \sum_{k \in \mathcal{K}} \eta\, \overline{w}_{ijk}^a \leq \sum_{i \in \mathcal{F}} \sum_{k \in \mathcal{K}} \overline{w}_{ijk}^a \leq \sum_{t \in \mathcal{T}} \lambda_{jt}^a$, where the last inequality holds because $(\overline{\boldsymbol{z}}, \overline{\boldsymbol{w}})$ satisfies the last constraint in the Relaxed problem. Thus, the latter claim holds. By the first claim, $\widehat{\boldsymbol{z}}$ is feasible to the Approximate Placement problem. By the second claim, $f^a_{\mathsf{LP}}(\widehat{\boldsymbol{z}}^a) \geq \sum_{i \in \mathcal{F}} \sum_{j \in \mathcal{D}} \sum_{k \in \mathcal{K}} r_{ijk}^a\, \theta_{jk}^a\, \widehat{w}_{ijk}^a$ for all $a \in \mathcal{A}$. Therefore, letting Z^* be the optimal objective value of the Approximate Placement problem with the bounding function f^a_{LP}, we get

$$\sum_{a \in \mathcal{A}} f^a_{\mathsf{LP}}(\widehat{\boldsymbol{z}}^a) \;\geq\; \sum_{a \in \mathcal{A}} \sum_{i \in \mathcal{F}} \sum_{j \in \mathcal{D}} \sum_{k \in \mathcal{K}} r_{ijk}^a\, \theta_{jk}^a\, \widehat{w}_{ijk}^a \;=\; \eta \sum_{a \in \mathcal{A}} \sum_{i \in \mathcal{F}} \sum_{j \in \mathcal{D}} \sum_{k \in \mathcal{K}} r_{ijk}^a\, \theta_{jk}^a\, \overline{w}_{ijk}^a \;\stackrel{(d)}{\geq}\; \eta\, Z^*,$$

where (d) holds since the Relaxed problem is a continuous relaxation of the Approximate Placement problem, so the optimal objective value of the Relaxed problem is at least Z^*. ■

For large enough values of $C_{\min}$ and $U_{\min}$, the interval $\left[1, \frac{C_{\min} \wedge U_{\min}}{|\mathcal{F}| \vee |\mathcal{A}|}\right]$ is nonempty, in which case, choosing an integer value for δ in this interval and setting $\eta = 1 - \frac{|\mathcal{F}| \vee |\mathcal{A}|}{C_{\min} \wedge U_{\min}}\, \delta$, we can use the theorem above to obtain an η-approximate solution to the Approximate Placement problem. There is a tradeoff in the choice of δ. In Step 2 of our approximation strategy, we want to approximate the Approximate

Placement problem as accurately as possible. Noting that $\eta = 1 - \frac{|\mathcal{F}| \vee |\mathcal{A}|}{C_{\min} \wedge U_{\min}} \delta$ is decreasing in δ, choosing δ smaller ensures that we approximate the Approximate Placement problem more accurately. In Step 3 of our approximation strategy, we want to construct a delivery promise and fulfillment policy such that the total expected profit of the policy is a large fraction of the bounding function evaluated at the approximate placement quantities. In Section 7, we will give a policy such that if the placement quantity for any product at any fulfillment center is at least $\widehat{z}_{\min}$, then the total expected profit of the policy is an $1 - O\Big(\sqrt{\frac{\log \widehat{z}_{\min}}{\widehat{z}_{\min}}}\Big)$-fraction of the bounding function. In Theorem 5.2, the number of units that we place for each product at each fulfillment center is at least δ, yielding placement quantities with $\widehat{z}_{\min} \geq \delta$. Since $1 - \sqrt{\frac{\log x}{x}}$ is increasing in $x \geq 3$, choosing δ larger ensures that the total expected profit of the policy is a larger fraction of the bounding function.

By the preceding discussion, for any δ satisfying the assumption in Theorem 5.2, we can execute Step 2 with $\alpha = 1 - O\Big(\frac{\delta}{C_{\min} \wedge U_{\min}}\Big)$ and Step 3 with $\beta = 1 - O\Big(\sqrt{\frac{\log \delta}{\delta}}\Big)$, so by Theorem 3.2, we obtain an $\alpha\beta = \Big(1 - O\Big(\frac{\delta}{C_{\min} \wedge U_{\min}}\Big)\Big)\Big(1 - O\Big(\sqrt{\frac{\log \delta}{\delta}}\Big)\Big)$-approximate solution to the joint placement, delivery promise and fulfillment problem. To choose the value of δ, we simply use a crude maximizer of the last expression. Noting that $\Big(1 - \frac{\delta}{C_{\min} \wedge U_{\min}}\Big)\Big(1 - \sqrt{\frac{\log \delta}{\delta}}\Big) \geq 1 - \frac{\delta}{C_{\min} \wedge U_{\min}} - \sqrt{\frac{\log \delta}{\delta}}$ and dropping the logarithmic term from the expression on the right side of the inequality, we choose δ as the maximizer of $1 - \frac{\delta}{C_{\min} \wedge U_{\min}} - \frac{1}{\sqrt{\delta}}$. The maximizer of this expression is $(\frac{1}{2}(C_{\min} \wedge U_{\min}))^{2/3}$. Using this value for δ after rounding it down to an integer, we execute Step 2 with $\alpha = 1 - O\Big(\frac{1}{(C_{\min} \wedge U_{\min})^{1/3}}\Big)$ and Step 3 with $\beta = 1 - O\Big(\frac{\sqrt{\log(C_{\min} \wedge U_{\min})}}{(C_{\min} \wedge U_{\min})^{1/3}}\Big)$. For large enough values of $C_{\min}$ and $U_{\min}$, rounding down $(\frac{1}{2}(C_{\min} \wedge U_{\min}))^{2/3}$ yields an integer in $\Big[1, \frac{C_{\min} \wedge U_{\min}}{|\mathcal{F}| \vee |\mathcal{A}|}\Big]$, as required by Theorem 5.2.

In the next section, we focus on obtaining approximate solutions to the Approximate Placement problem by exploiting the DR-submodularity of the bounding function.

6. Approximating the Placement Problem: DR-Submodularity

We will show that our bounding function f^a_{LP} given by the Bounding problem is monotone and DR-submodular. In this case, the Approximate Placement problem takes the form of maximizing a monotone and DR-submodular function subject to two partition constraints, at which point, we can build on results for maximizing such functions. By the definition of f^a_{LP} in the Bounding problem, $f^a_{\mathsf{LP}}(\boldsymbol{z}^a)$ is monotone in $\boldsymbol{z}^a$ in the sense that $f^a_{\mathsf{LP}}(\boldsymbol{z}^a) \geq f^a_{\mathsf{LP}}(\boldsymbol{y}^a)$ for all $\boldsymbol{z}^a, \boldsymbol{y}^a \in \mathbb{Z}^n_+$ with $\boldsymbol{z}^a \geq \boldsymbol{y}^a$. Recall that we write $\boldsymbol{z}^a \geq \boldsymbol{y}^a$ if and only if $z^a_i \geq y^a_i$ for all $i \in \mathcal{F}$. On the other hand, the function $f^a : \mathbb{Z}^n_+ \to \mathbb{R}_+$ is DR-submodular if we have $f^a(\boldsymbol{z}^a + \boldsymbol{e}_i) - f^a(\boldsymbol{z}^a) \leq f^a(\boldsymbol{y}^a + \boldsymbol{e}_i) - f^a(\boldsymbol{y}^a)$ for each $i \in \mathcal{F}$ and $\boldsymbol{z}^a, \boldsymbol{y}^a \in \mathbb{Z}^n_+$ with $\boldsymbol{z}^a \geq \boldsymbol{y}^a$. We proceed to showing that our bounding function $f^a_{\mathsf{LP}}(\boldsymbol{z}^a)$ is DR-submodular in $\boldsymbol{z}^a$. To show DR-submodularity, we use the dual of the Bounding problem. Associating the dual variables $\boldsymbol{\mu} = (\mu_i : i \in \mathcal{F})$ and $\boldsymbol{\sigma} = (\sigma_j : j \in \mathcal{D})$ with the two constraints in the Bounding problem, we can compute $f^a_{\mathsf{LP}}(\boldsymbol{z}^a)$ by solving the dual of the Bounding problem, which is given by the linear program

$$\begin{aligned} f^a_{\mathsf{LP}}(\boldsymbol{z}^a) &= \min_{(\boldsymbol{\mu},\boldsymbol{\sigma})\in\mathbb{R}^{n\times|\mathcal{D}|}_+} \left\{ \sum_{i\in\mathcal{F}} z^a_i\,\mu_i + \sum_{j\in\mathcal{D}}\sum_{t\in\mathcal{T}} \lambda^a_{jt}\,\sigma_j \;:\; \sigma_j \geq \theta^a_{jk}\,[r^a_{ijk}-\mu_i] \;\; \forall\, i\in\mathcal{F},\; j\in\mathcal{D},\; k\in\mathcal{K} \right\} \\ &= \min_{\boldsymbol{\mu}\in\mathbb{R}^n_+} \left\{ \sum_{i\in\mathcal{F}} z^a_i\,\mu_i + \sum_{j\in\mathcal{D}}\sum_{t\in\mathcal{T}} \lambda^a_{jt} \max_{(i,k)\in\mathcal{F}\times\mathcal{K}} \theta^a_{jk}\,[r^a_{ijk}-\mu_i]^+ \right\}, \end{aligned} \qquad \text{(Dual)}$$

where the second equality holds by noting that it is optimal to choose σ_j as small as possible in the first problem above and noting the first constraint in this problem.

We make two observations. First, if we view $f^a_{\mathsf{LP}}(\boldsymbol{z}^a)$ as a function of z^a_i, then it corresponds to the optimal objective value of the linear program in the Dual problem as a function of the objective function coefficient of the decision variable μ_i. Therefore, $f^a_{\mathsf{LP}}(\boldsymbol{z}^a)$ is a continuous and piecewise linear function of z^a_i with a finite number of points of nondifferentiability. Second, using $\widehat{\boldsymbol{\mu}}(\boldsymbol{z}^a) = (\widehat{\mu}_i(\boldsymbol{z}^a) : i\in\mathcal{F})$ to denote an optimal solution to the Dual problem as a function of $\boldsymbol{z}^a$, by linear programming duality, the derivative of $f^a_{\mathsf{LP}}(\boldsymbol{z}^a)$ with respect to z^a_i exists if and only if the optimal solution $\widehat{\boldsymbol{\mu}}(\boldsymbol{z}^a)$ is unique. Furthermore, the derivative of $f^a_{\mathsf{LP}}(\boldsymbol{z}^a)$ with respect to z^a_i is given by $\frac{\partial f^a_{\mathsf{LP}}(\boldsymbol{z}^a)}{\partial z^a_i}\big|_{\boldsymbol{z}^a=\boldsymbol{y}^a} = \widehat{\mu}_i(\boldsymbol{y}^a)$. In this case, by the fundamental theorem of calculus, we obtain $f^a_{\mathsf{LP}}(\boldsymbol{y}^a+\boldsymbol{e}_i) - f^a_{\mathsf{LP}}(\boldsymbol{y}^a) = \int_0^1 \frac{\partial f^a_{\mathsf{LP}}(\boldsymbol{z}^a)}{\partial z^a_i}\big|_{\boldsymbol{z}^a=\boldsymbol{y}^a+h\,\boldsymbol{e}_i}\,dh = \int_0^1 \widehat{\mu}_i(\boldsymbol{y}^a+h\,\boldsymbol{e}_i)\,dh$, where the second equality uses the expression for the derivative $\frac{\partial f^a_{\mathsf{LP}}(\boldsymbol{z}^a)}{\partial z^a_i}\big|_{\boldsymbol{z}^a=\boldsymbol{y}^a+h\,\boldsymbol{e}_i}$ given by the second observation. We need the next auxiliary lemma to show that our bounding function is DR-submodular. In this lemma, to capture the objective function of the Dual problem as a function of $(\boldsymbol{z}^a,\boldsymbol{\mu})$, we set

$$Q(\boldsymbol{z}^a,\boldsymbol{\mu}) = \sum_{i\in\mathcal{F}} z^a_i\,\mu_i + \sum_{j\in\mathcal{D}}\sum_{t\in\mathcal{T}} \lambda^a_{jt} \max_{(i,k)\in\mathcal{F}\times\mathcal{K}} \theta^a_{jk}\,[r^a_{ijk}-\mu_i]^+.$$

Furthermore, for $\boldsymbol{z}^a,\boldsymbol{y}^a\in\mathbb{Z}^n$, we define the component by component maximum and minimum vectors $\boldsymbol{z}^a\vee\boldsymbol{y}^a = (z^a_i\vee y^a_i : i\in\mathcal{F})$ and $\boldsymbol{z}^a\wedge\boldsymbol{y}^a = (z^a_i\wedge y^a_i : i\in\mathcal{F})$.

Lemma 6.1 (Objective of Dual) *For $\boldsymbol{z}^a,\boldsymbol{y}^a\in\mathbb{Z}^n_+$ that satisfy $\boldsymbol{z}^a\geq\boldsymbol{y}^a$, along with $\boldsymbol{\mu},\boldsymbol{\gamma}\in\mathbb{R}^n_+$, we have $Q(\boldsymbol{z}^a,\boldsymbol{\mu}) + Q(\boldsymbol{y}^a,\boldsymbol{\gamma}) \geq Q(\boldsymbol{z}^a,\boldsymbol{\mu}\wedge\boldsymbol{\gamma}) + Q(\boldsymbol{y}^a,\boldsymbol{\mu}\vee\boldsymbol{\gamma})$.*

Proof: Define $L(\boldsymbol{z}^a,\boldsymbol{\mu}) = \sum_{i\in\mathcal{F}} z^a_i\,\mu_i$ and $G_j(\boldsymbol{\mu}) = \max_{(i,k)\in\mathcal{F}\times\mathcal{K}} \theta^a_{jk}\,[r^a_{ijk}-\mu_i]^+$. First, we claim that $L(\boldsymbol{z}^a,\boldsymbol{\mu}) + L(\boldsymbol{y}^a,\boldsymbol{\gamma}) \geq L(\boldsymbol{z}^a,\boldsymbol{\mu}\wedge\boldsymbol{\gamma}) + L(\boldsymbol{y}^a,\boldsymbol{\mu}\vee\boldsymbol{\gamma})$. Since $a+b=(a\vee b)+(a\wedge b)$, we have

$$\begin{aligned} z^a_i\,\mu_i + y^a_i\,\gamma_i \;&=\; z^a_i\left\{(\mu_i\wedge\gamma_i)+(\mu_i\vee\gamma_i)-\gamma_i\right\} + y^a_i\left\{(\mu_i\vee\gamma_i)+(\mu_i\wedge\gamma_i)-\mu_i\right\} \\ &\geq\; z^a_i\,(\mu_i\wedge\gamma_i) + y^a_i\,(\mu_i\vee\gamma_i) + y^a_i\left\{(\mu_i\vee\gamma_i)-\gamma_i+(\mu_i\wedge\gamma_i)-\mu_i\right\} \;=\; z^a_i\,(\mu_i\wedge\gamma_i) + y^a_i\,(\mu_i\vee\gamma_i), \end{aligned}$$

where the inequality holds since $z^a_i\geq y^a_i$ and $(\mu_i\vee\gamma_i)-\gamma_i\geq 0$. Adding the chain of inequalities above over all $i\in\mathcal{F}$, the claim holds. Second, we claim that $G_j(\boldsymbol{\mu}) + G_j(\boldsymbol{\gamma}) \geq G_j(\boldsymbol{\mu}\wedge\boldsymbol{\gamma}) + G_j(\boldsymbol{\mu}\vee\boldsymbol{\gamma})$. Note that if $F:\mathbb{R}\to\mathbb{R}$ is decreasing, then we have the identity $F(a\wedge b)=F(a)\vee F(b)$. Therefore, we have $\theta^a_{jk}\,[r^a_{ijk}-(\mu_i\wedge\gamma_i)]^+=\theta^a_{jk}[r^a_{ijk}-\mu_i]^+\vee\theta^a_{jk}[r^a_{ijk}-\gamma_i]^+$, in which case, taking the maximum of both sides over all $(i,k)\in(\mathcal{F},\mathcal{K})$, we get $G_j(\boldsymbol{\mu}\wedge\boldsymbol{\gamma})=G_j(\boldsymbol{\mu})\vee G_j(\boldsymbol{\gamma})$. Also, using the fact that $G_j(\boldsymbol{\mu})$ is decreasing in $\boldsymbol{\mu}$, we have $G_j(\boldsymbol{\mu})\ge G_j(\boldsymbol{\mu}\vee\boldsymbol{\gamma})$ and $G_j(\boldsymbol{\gamma})\ge G_j(\boldsymbol{\mu}\vee\boldsymbol{\gamma})$, which implies that $G_j(\boldsymbol{\mu})\wedge G_j(\boldsymbol{\gamma})\ge G_j(\boldsymbol{\mu}\vee\boldsymbol{\gamma})$. In this case, using $a+b=(a\vee b)+(a\wedge b)$ once more, we get

$$G_j(\boldsymbol{\mu})+G_j(\boldsymbol{\gamma}) \;=\; \Big\{G_j(\boldsymbol{\mu})\vee G_j(\boldsymbol{\gamma})\Big\}+\Big\{G_j(\boldsymbol{\mu})\wedge G_j(\boldsymbol{\gamma})\Big\} \;\ge\; G_j(\boldsymbol{\mu}\wedge\boldsymbol{\gamma})+G_j(\boldsymbol{\mu}\vee\boldsymbol{\gamma}),$$

so the latter claim also holds as well. The result follows by using the two claims and noting the fact that $Q(\boldsymbol{z}^a,\boldsymbol{\mu})=L(\boldsymbol{z}^a,\boldsymbol{\mu})+\sum_{j\in\mathcal{D}}\sum_{t\in\mathcal{T}}\lambda^a_{jt}\,G_j(\boldsymbol{\mu})$. ■

In the next theorem, we build on the lemma above to show that our bounding function $f^a_{\mathsf{LP}}(\boldsymbol{z}^a)$ given by the Bounding problem is DR-submodular in $\boldsymbol{z}^a$.

Theorem 6.2 (DR-Submodularity) *For each $\boldsymbol{z}^a,\boldsymbol{y}^a\in\mathbb{Z}^n_+$ that satisfy $\boldsymbol{z}^a\ge\boldsymbol{y}^a$, along with each $i\in\mathcal{F}$, we have $f^a_{\mathsf{LP}}(\boldsymbol{z}^a+\boldsymbol{e}_i)-f^a_{\mathsf{LP}}(\boldsymbol{z}^a)\le f^a_{\mathsf{LP}}(\boldsymbol{y}^a+\boldsymbol{e}_i)-f^a_{\mathsf{LP}}(\boldsymbol{y}^a)$.*

Proof: Noting the definition of $Q(\boldsymbol{z}^a,\boldsymbol{\mu})$, the Dual problem is $\min_{\boldsymbol{\mu}\in\mathbb{R}^n_+}Q(\boldsymbol{z}^a,\boldsymbol{\mu})$. Using $\widehat{\boldsymbol{\mu}}(\boldsymbol{z}^a)$ to denote an optimal solution to the Dual problem as a function of $\boldsymbol{z}^a$, we claim that there exists an optimal solution to the Dual problem such that if $\boldsymbol{z}^a\ge\boldsymbol{y}^a$, then $\widehat{\boldsymbol{\mu}}(\boldsymbol{z}^a)\le\widehat{\boldsymbol{\mu}}(\boldsymbol{y}^a)$. Since $\widehat{\boldsymbol{\mu}}(\boldsymbol{y}^a)$ is an optimal solution to the problem $\min_{\boldsymbol{\mu}\in\mathbb{R}^n_+}Q(\boldsymbol{y}^a,\boldsymbol{\mu})$, but $\widehat{\mu}(\boldsymbol{y}^a)\vee\widehat{\mu}(\boldsymbol{z}^a)$ is only a feasible solution, we have $Q(\boldsymbol{y}^a,\widehat{\boldsymbol{\mu}}(\boldsymbol{y}^a))\le Q(\boldsymbol{y}^a,\widehat{\boldsymbol{\mu}}(\boldsymbol{y}^a)\vee\widehat{\boldsymbol{\mu}}(\boldsymbol{z}^a))$. In this case, using Lemma 6.1, we get

$$Q(\boldsymbol{z}^a,\widehat{\boldsymbol{\mu}}(\boldsymbol{z}^a))\ge Q(\boldsymbol{z}^a,\widehat{\boldsymbol{\mu}}(\boldsymbol{z}^a)\wedge\widehat{\boldsymbol{\mu}}(\boldsymbol{y}^a))+Q(\boldsymbol{y}^a,\widehat{\boldsymbol{\mu}}(\boldsymbol{z}^a)\vee\widehat{\boldsymbol{\mu}}(\boldsymbol{y}^a))-Q(\boldsymbol{y}^a,\widehat{\boldsymbol{\mu}}(\boldsymbol{y}^a))\ge Q(\boldsymbol{z}^a,\widehat{\boldsymbol{\mu}}(\boldsymbol{z}^a)\wedge\widehat{\boldsymbol{\mu}}(\boldsymbol{y}^a)),$$

which implies that $\widehat{\boldsymbol{\mu}}(\boldsymbol{z}^a)\wedge\widehat{\boldsymbol{\mu}}(\boldsymbol{y}^a)$ is also an optimal solution to the problem $\min_{\boldsymbol{\mu}\in\mathbb{R}^n_+}Q(\boldsymbol{z}^a,\boldsymbol{\mu})$. Noting that $\widehat{\boldsymbol{\mu}}(\boldsymbol{z}^a)\wedge\widehat{\boldsymbol{\mu}}(\boldsymbol{y}^a)\le\widehat{\boldsymbol{\mu}}(\boldsymbol{y}^a)$, the claim follows.

By the claim, $\widehat{\boldsymbol{\mu}}(\boldsymbol{z}^a+h\,\boldsymbol{e}_i)\le\widehat{\boldsymbol{\mu}}(\boldsymbol{y}^a+h\,\boldsymbol{e}_i)$ for all $h\in\mathbb{R}_+$, so the discussion before Lemma 6.1 yields $f^a_{\mathsf{LP}}(\boldsymbol{z}^a+\boldsymbol{e}_i)-f^a_{\mathsf{LP}}(\boldsymbol{z}^a)=\int_0^1\widehat{\mu}_i(\boldsymbol{z}^a+h\,\boldsymbol{e}_i)\,dh\le\int_0^1\widehat{\mu}_i(\boldsymbol{y}^a+h\,\boldsymbol{e}_i)\,dh=f^a_{\mathsf{LP}}(\boldsymbol{y}^a+\boldsymbol{e}_i)-f^a_{\mathsf{LP}}(\boldsymbol{y}^a)$. ■

By Theorem 6.2, the bounding function $f^a_{\mathsf{LP}}(\boldsymbol{z}^a)$ is DR-submodular in $\boldsymbol{z}^a$. Noting the Bounding problem, if we had $\theta^a_{jk}=1$ for all $j\in\mathcal{D}$ and $k\in\mathcal{K}$, then $f^a_{\mathsf{LP}}(\boldsymbol{z}^a)$ would correspond to the optimal objective value of the transportation problem on a bipartite graph as a function of the supply on one side of the graph. There is work that studies the submodularity properties of the optimal objective value of the transportation problem as a function of various parts of the problem data. Shapley (1962) considers the transportation problem on a bipartite graph, where all nodes on one side of the graph have a supply of zero or one. In his Theorem 1, the author shows that the optimal objective value of the problem is a submodular set function when viewed as a function of the set of nodes that have a supply of one. On the other hand, Nemhauser et al. (1978) consider the transportation problem on a bipartite graph with multiple suppliers, where each supplier provides an exogenously given amount of supply to each node on one side of the graph and there is a fixed cost associated with employing each supplier. In their Proposition 3.4, the authors show that the optimal objective value of the problem is a submodular set function when viewed as a function of the set of suppliers that are employed. Cao and He (2019) consider the transportation problem on a bipartite graph, where the supply and demand on each side of the graph are integer quantities. In their Theorem 31, which is the result closest to ours, the authors show that the optimal objective value of the problem is DR-submodular when viewed as a function of the supply quantities on one side of the graph. Although the Bounding problem has the flavor of a transportation problem, our result is not a direct implication of the existing work. In Theorem 6.2 that we give above, we allow θ^a_{jk} to take a fractional value, so the Bounding problem is a generalization of the transportation problem on a bipartite graph. Also, our DR-submodularity result does not require $\sum_{t \in \mathcal{T}} \lambda^a_{jt}$ in the second constraint in the Bounding problem to take an integer value. Building on the fact that the bounding function $f^a_{\mathsf{LP}}(\boldsymbol{z}^a)$ is monotone and DR-submodular in $\boldsymbol{z}^a$, there are two possible approaches for obtaining an approximate solution to the Approximate Placement problem.

First, there is work on maximizing submodular and monotone set functions subject to partition constraints. Nemhauser et al. (1978) consider the problem of maximizing a monotone and submodular set function under k partition constraints and show that a greedy algorithm provides a $\frac{1}{k+1}$-approximate solution. For the same problem, Lee et al. (2010) give a more involved algorithm that provides a $\frac{1}{k+\epsilon}$-approximate solution for any $\epsilon > 0$. To build on these results, we can represent any placement decision vector as a subset of a certain ground set. Letting $C_{\max} = \max_{a \in \mathcal{A}} C^a$ to capture the maximum number of units to place for any product, we create $C_{\max}$ copies of each product, each copy corresponding to a unit we need to place. We represent any placement decision vector as a subset of the ground set $\mathcal{G} = \{(i, a, k) : i \in \mathcal{F},\ a \in \mathcal{A},\ k = 1, \ldots, C_{\max}\}$. In the placement decision represented by the subset $S \subseteq \mathcal{G}$, if $(i, a, k) \in S$, then we place the k-th copy of product a at fulfillment center i. Using Theorem 6.2, we can show that our bounding function is a monotone and submodular set function when we represent the placement decisions by using subsets of the ground set. In this case, the Approximate Placement problem maximizes a monotone and submodular set function subject to two partition constraints, which allows us to obtain a $\frac{1}{2+\epsilon}$-approximate solution. Because the number of elements in $\mathcal{G}$ is linear in $C_{\max}$, the running time of this approach is polynomial in $C_{\max}$. We give the details of this approach in Appendix C.

Second, there has been a recent surge of work on maximizing DR-submodular functions over an integer lattice by using their multilinear extensions. In this work, one uses a variant of the greedy algorithm to obtain an approximate maximizer of the multilinear extension and uses pipage rounding to obtain an integer solution from the approximate maximizer of the multilinear extension; see Ageev and Sviridenko (2004), Feldman (2013) and Ene and Nguyen (2018). Existing work so far focuses on maximizing a DR-submodular function subject to cardinality and knapsack constraints. We show that we can extend this work to design an algorithm to maximize a DR-submodular function subject to two partition constraints, ultimately allowing us to obtain a $(1-\frac{1}{e}-\epsilon)$-approximate solution to the Approximate Placement problem in running time that is polynomial in $1/\epsilon$ and the problem input. In this approach, we use a binary representation to capture the placement quantity of any product at any fulfillment center. Accordingly, letting $U_{\max}=\max_{i\in\mathcal{F}} U_i$ to capture the maximum fulfillment center capacity, the running time of this approach is polynomial in $\log C_{\max}$, $\log U_{\max}$ and $1/\epsilon$. We give the details of this approach in Appendix D. Therefore, by the discussion in this section, we can exploit the DR-submodularity of our bounding function to execute Step 2 in our approximation strategy with $\alpha=1-\frac{1}{e}-\epsilon$. We have $1-\frac{1}{e}-\epsilon\geq\frac{1}{2}$ for $\epsilon\in(0,0.1)$. In the next section, we focus on executing Step 3 in our approximation strategy, which requires constructing a delivery promise and fulfillment policy.

7. Approximating the Fulfillment Problem

Considering the case where the placement decisions for product a are given by the vector $\widehat{\boldsymbol{z}}^a$, we use $\widehat{z}_{\min}=\min_{i\in\mathcal{F}}\{\widehat{z}_i^a:\widehat{z}_i^a\geq 1\}$ to capture the smallest nonzero placement quantity for the product at any fulfillment center. We construct a delivery promise and fulfillment policy for product a such that if the placement decisions for the product are given by the vector $\widehat{\boldsymbol{z}}^a$, then the policy obtains a total expected profit of at least $\max\left\{\frac{1}{2},\,1-O\left(\sqrt{\frac{\log\widehat{z}_{\min}}{\widehat{z}_{\min}}}\right)\right\} f^a_{\mathsf{LP}}(\widehat{\boldsymbol{z}}^a)$ from this product. As discussed at the end of Section 5, we may obtain the placement decisions $\widehat{\boldsymbol{z}}^a$ by rounding an optimal solution to the Relaxed problem as in Theorem 5.2 and choosing δ as the integer obtained by rounding down $(\frac{1}{2}\,(C_{\min}\wedge U_{\min}))^{2/3}$, in which case, we have $\widehat{z}_{\min}\geq(\frac{1}{2}\,(C_{\min}\wedge U_{\min}))^{2/3}$. Noting that $1-\sqrt{\log x/x}$ is increasing in $x\geq 3$, we get $1-\frac{\log\widehat{z}_{\min}}{\widehat{z}_{\min}}\geq 1-\frac{\frac{2}{3}\log(\frac{1}{2}(C_{\min}\wedge U_{\min}))}{(\frac{1}{2}(C_{\min}\wedge U_{\min}))^{2/3}}$ for large enough values of $C_{\min}$ and $U_{\min}$. In this case, our delivery promise and fulfillment policy has a performance guarantee of $1-O\left(\sqrt{\frac{\log(C_{\min}\wedge U_{\min})}{(C_{\min}\wedge U_{\min})^{2/3}}}\right)$. On the other hand, if we obtain the placement decisions $\widehat{\boldsymbol{z}}^a$ by using the DR-submodularity of the bounding function as in Theorem 6.2, then we will simply use the fact that our delivery promise and fulfillment policy provides a performance guarantee of $\frac{1}{2}$. Throughout this section, we proceed with the understanding that the placement decisions $\widehat{\boldsymbol{z}}^a$ are fixed and focus on constructing the delivery promise and fulfillment policy.

In our policy, we solve the Bounding problem with $\boldsymbol{z}^a=\widehat{\boldsymbol{z}}^a$ once at the beginning of the selling horizon. Letting $\widehat{\boldsymbol{w}}^a$ be an optimal solution, we can assume that $\sum_{i\in\mathcal{F}}\sum_{k\in\mathcal{K}}\widehat{w}^a_{ijk}=\sum_{t\in\mathcal{T}}\lambda^a_{jt}$ for all

$j \in \mathcal{D}$. In particular, since $\theta^a_{j\kappa} = 0$ for the dummy promise κ, if $\sum_{i\in\mathcal{F}} \sum_{k\in\mathcal{K}} \widehat{w}^a_{ijk} < \sum_{t\in\mathcal{T}} \lambda^a_{jt}$, then we can increase the value of the decision variable $\widehat{w}^a_{ij\kappa}$ for some fulfillment center i without changing the objective function value or the left side of the first constraint in the Bounding problem. Thus, we can assume that $\sum_{i\in\mathcal{F}} \frac{\sum_{k\in\mathcal{K}} \widehat{w}^a_{ijk}}{\sum_{t\in\mathcal{T}} \lambda^a_{jt}} = 1$. Letting $\widehat{\eta}^a_{ij} = \frac{\sum_{k\in\mathcal{K}} \widehat{w}^a_{ijk}}{\sum_{t\in\mathcal{T}} \lambda^a_{jt}}$ for notational brevity, by the last equality, we have $\sum_{i\in\mathcal{F}} \widehat{\eta}^a_{ij} = 1$ for all $j \in \mathcal{D}$, in which case, for each demand region j, the vector $(\widehat{\eta}^a_{ij} : i \in \mathcal{F})$ characterizes a probability distribution over the fulfillment centers. In our policy, we assign each demand for product a from demand region j to fulfillment center i with probability $\gamma\,\widehat{\eta}^a_{ij}$, where $\gamma \in (0,1)$ is a parameter we tune. In this way, each fulfillment center faces an exogenous demand stream. We solve a dynamic program to choose a delivery promise for the demands assigned to each fulfillment center so that we maximize the total expected profit from the demands assigned to the fulfillment center. In particular, let $V^a_{it}(x^a_i)$ be the optimal total expected profit from the demands for product a assigned to fulfillment center i over time periods $\{t, \ldots, T\}$, given that the remaining inventory of product a at fulfillment center i at the beginning of time period t is x^a_i. Using the boundary condition $V^a_{i,T+1} = 0$, the value functions $\{V^a_{it} : t \in \mathcal{T}\}$ are given by

$$V^a_{it}(x^a_i) \;=\; \sum_{j\in\mathcal{D}} \lambda^a_{jt}\,\gamma\,\widehat{\eta}^a_{ij} \max_{k\in\mathcal{K}} \left\{ \mathbb{1}_{(x^a_i \geq 1)}\, \theta^a_{jk} \Big[r^a_{ijk} + V^a_{i,t+1}(x^a_i - 1) - V^a_{i,t+1}(x^a_i) \Big] \right\} + V^a_{i,t+1}(x^a_i).$$

The dynamic program above follows from a reasoning similar to the one in the Fulfillment dynamic program, but we emphasize several differences. Noting that each fulfillment center faces an exogenous demand stream, we can compute the total expected profit from the demands assigned to each fulfillment center by solving a separate dynamic program for different fulfillment centers. Thus, the state variable in the dynamic program above only keeps the remaining number of units for product a at fulfillment center i. Furthermore, if we have a demand for product a from demand region j at time period t, then we assign this demand to fulfillment center i with probability $\gamma\,\widehat{\eta}^a_{ij}$. Thus, we only choose a delivery promise in the maximization problem above. Our policy makes its decisions as follows. We compute the value functions $\{V^a_{it} : t \in \mathcal{T}\}$ by solving the dynamic program above. At time period t, if the remaining inventories for product a at the fulfillment centers correspond to the vector $\boldsymbol{x}^a$ and we have a demand for this product from demand region j, then we assign this demand to fulfillment center i with probability $\gamma\,\widehat{\eta}^a_{ij}$. For this demand, we offer the delivery promise $\arg\max_{k\in\mathcal{K}} \big\{ \mathbb{1}_{(x^a_i \geq 1)}\, \theta^a_{jk} \big[r^a_{ijk} + V^a_{i,t+1}(x^a_i - 1) - V^a_{i,t+1}(x^a_i) \big] \big\}$ and use the sampled fulfillment center i if the customer accepts the offered promise.

We refer to our policy as the randomized policy, as it randomly chooses a fulfillment center for each demand. If the placement decisions of product a are given $\widehat{\boldsymbol{z}}^a$, then the total expected profit of the randomized policy from product a is $\sum_{i\in\mathcal{F}} V^a_{i1}(\widehat{z}^a_i)$. The randomized policy has the flavor of the primal routing algorithm in Gallego et al. (2015), but our asymptotic optimality result for it will be independent of how the demand is scaled. We have the next performance guarantee.

Theorem 7.1 (Policy Performance) *There exists a choice of the tuning parameter γ such that if the placement decisions for product a are given by the vector $\widehat{\boldsymbol{z}}^a$ with $\widehat{z}_{\min} = \min_{i \in \mathcal{F}} \{\widehat{z}_i^a : \widehat{z}_i^a \geq 1\}$, then the total expected profit that the randomized policy obtains from product a is at least*

$$\max\left\{\frac{1}{2},\, 1 - O\left(\sqrt{\frac{\log \widehat{z}_{\min}}{\widehat{z}_{\min}}}\right)\right\} f_{\mathsf{LP}}^a(\widehat{\boldsymbol{z}}^a).$$

The result above holds for any $\widehat{\boldsymbol{z}}^a$. The proof is in Appendix E and it has two parts. In the first part, we construct lower bounds on the value functions $\{V_{it}^a : t \in \mathcal{T}\}$. Using $\{\widehat{V}_{it}^a : t \in \mathcal{T}\}$ to denote the lower bounds and noting that the total expected profit of the randomized policy is $\sum_{i \in \mathcal{F}} V_{i1}^a(\widehat{z}_i^a)$, we show that $\sum_{i \in \mathcal{F}} V_{i1}^a(\widehat{z}_i^a) \geq \sum_{i \in \mathcal{F}} \widehat{V}_{i1}^a(\widehat{z}_i^a) \geq \frac{1}{2} f_{\mathsf{LP}}^a(\widehat{\boldsymbol{z}}^a)$. In the second part, we consider a policy that assigns a demand for product a from demand region j to fulfillment center and promise pair (i,k) with probability $\gamma \frac{\widehat{w}_{ijk}^a}{\sum_{t \in \mathcal{T}} \lambda_{jt}^a}$. We call this policy the fully randomized policy, as it chooses both the fulfillment center and delivery promise randomly. Under the fully randomized policy, each fulfillment center and delivery promise faces an exogenous demand. We show that the total expected profit of the randomized policy is at least as large as that of the fully randomized policy and the total expected profit of the fully randomized policy is at least $\left(1 - O\left(\sqrt{\frac{\log \widehat{z}_{\min}}{\widehat{z}_{\min}}}\right)\right) f_{\mathsf{LP}}^a(\widehat{\boldsymbol{z}}^a)$. Lastly, it is possible to construct a policy with a slightly improved performance guarantee of $\max\left\{\frac{1}{2}, 1 - O\left(\frac{1}{\sqrt{\widehat{z}_{\min}}}\right)\right\}$ by using the recent results on the magician problem; see Alaei (2014) and Jiang et al. (2023). We give this result in Appendix F. Next, we discuss performing rollout on the randomized policy to further improve its performance.

Performing Rollout on the Randomized Policy:

To perform rollout on an initial policy on hand, we follow the greedy action with respect to the value functions of the initial policy. If the remaining inventories for product a at the beginning of time period t are given by $\boldsymbol{x}^a$, then the total expected profit that the randomized policy obtains from product a over time periods $\{t, \ldots, T\}$ is $\sum_{i \in \mathcal{F}} V_{it}^a(x_i^a)$. To perform rollout on the randomized policy, we replace $J_{t+1}^a(\boldsymbol{x}^a)$ in the Fulfillment dynamic program with $\sum_{i \in \mathcal{F}} V_{i,t+1}^a(x_i^a)$ and solve the corresponding maximization problem. Thus, if the remaining inventories for product a at the beginning of time period t correspond to the vector $\boldsymbol{x}^a$ and we have a demand for this product from demand region j, then the rollout policy picks the fulfillment center and promise pair $\arg\max_{(i,k) \in \mathcal{F} \times \mathcal{K}} \left\{\mathbb{1}_{(x_i^a \geq 1)} \theta_{jk}^a \left[r_{ijk}^a + V_{i,t+1}^a(x_i^a - 1) - V_{t+1}^a(x_i^a)\right]\right\}$. Noting the expression $\mathbb{1}_{(x_i^a \geq 1)}$ in the last objective function, if we do not have remaining inventory for product a at fulfillment center i so that $x_i^a = 0$, then using fulfillment center i would yield an objective value of zero, whereas offering the dummy promise would already yield the same optimal objective value. Thus, the rollout policy does not consider using a fulfillment center without any remaining inventory. In contrast, the randomized policy may assign a demand to a fulfillment center without any remaining inventory, even when there are other fulfillment centers with remaining inventories. Due to this difference, the rollout policy can significantly improve upon the randomized policy in practice, yielding improvements as large as 21% in our test problems. It is known that the rollout policy performs at least as well as the initial policy on hand; see Section 6.1.3 in Bertsekas and Tsitsiklis (1996). Thus, by Theorem 7.1, the total expected profit of the rollout policy from product a is at least $\max\left\{\frac{1}{2}, 1-O\left(\sqrt{\frac{\log \widehat{z}_{\min}}{\widehat{z}_{\min}}}\right)\right\} f^a_{\mathsf{LP}}(\widehat{\boldsymbol{z}}^a)$, so we can use the rollout policy in Step 3 of our approximation strategy. Rusmevichientong et al. (2023) also use rollout for revenue management with single capacity resources. In Theorem 3.2, we gave the performance guarantee that we get for the joint placement, delivery promise and fulfillment problem by using our approximation strategy. We put together the development in the paper to give a proof for this performance guarantee.

Proof of Theorem 3.2:

We consider two cases. First, by the discussion at the end of Section 6, for any $\epsilon > 0$, we can execute Step 2 in our approximation strategy with $\alpha = \frac{1}{2+\epsilon/2}$, whereas by Theorem 7.1, we can execute Step 3 with $\beta = \frac{1}{2}$, so we get $\alpha\beta = \frac{1}{4+\epsilon}$. Second, we can make the placement decisions by rounding an optimal solution to the Relaxed problem as in Theorem 5.2 and choosing δ as the integer obtained by rounding down $(\frac{1}{2}\,(C_{\min} \wedge U_{\min}))^{2/3}$. Using this value for δ, Theorem 5.2 implies that we obtain a $1-O\left(\frac{1}{(C_{\min}\wedge U_{\min})^{1/3}}\right)$-approximate solution to the Approximate Placement problem, so we can execute Step 2 with $\alpha = 1-O\left(\frac{1}{(C_{\min}\wedge U_{\min})^{1/3}}\right)$. By the discussion at the beginning of this section, if we choose δ as integer obtained by rounding down $(\frac{1}{2}\,(C_{\min} \wedge U_{\min}))^{2/3}$, then we can execute Step 3 in our approximation strategy with $\beta = 1-O\left(\sqrt{\frac{\log(C_{\min}\wedge U_{\min})}{(C_{\min}\wedge U_{\min})^{2/3}}}\right)$, so $\alpha\beta = 1-O\left(\frac{\sqrt{\log(C_{\min}\wedge U_{\min})}}{(C_{\min}\wedge U_{\min})^{1/3}}\right)$. In this case, the result follows by Remark 3.1 with the two specific values for $\alpha\beta$ above. ■

Considering the performance guarantee of $1-O\left(\frac{\sqrt{\log(C_{\min}\wedge U_{\min})}}{(C_{\min}\wedge U_{\min})^{1/3}}\right)$ for our approximation strategy, we obtain asymptotically optimal solutions as long as the total number of units to be placed for each product and capacities of the fulfillment centers get large. In this result, the expected demands for some products from some demand regions can stay small as the total number of units to be placed and capacities of the fulfillment centers get large. We can also provide an asymptotic optimality result under an alternative asymptotic regime that makes stronger assumptions on how the problem is scaled. If the total number of units to be placed for all products, capacities of all fulfillment centers and total expected demands for all products from all demand regions are scaled with rate μ, then we can use a standard argument on a fluid approximation to obtain a performance guarantee of $1-O\left(\sqrt{\frac{\log \mu}{\mu}}\right)$. We give the details of this result in Appendix G.

When compared with our asymptotic optimality result, the alternative asymptotic regime makes significantly stronger assumptions on the way we scale the problem. In our result, the expected demand for some products from some demand regions can stay small. We obtain an asymptotically optimal solution as long as the numbers of units to place for the products and capacities of the fulfillment centers get large. The alternative asymptotic regime requires that the total expected demands for all products from all demand regions to also get large. Also, we get a constant factor performance guarantee from our approximation strategy, but as far as we are aware, the argument in Appendix G does not yield an analogue of this constant factor performance guarantee.

8. Computational Experiments

We discuss our benchmark strategies, followed by two sets of computational experiments. The first set uses synthetically generated data. The second set uses data from an online retailer in Brazil.

8.1 Benchmark Strategies

We compare our approach with a Lagrangian relaxation strategy, as well as heuristics that do not explicitly consider the interaction between placement, delivery promise and fulfillment.

Approximation Strategy (APS): In APS, we use our approximation strategy. To make the placement decisions, we approximately solve the Approximate Placement problem with the bounding function f^a_{LP}. By the discussion in Section 6, we can get a constant factor approximation to this problem by using the greedy algorithm. To make the delivery promise and fulfillment decisions, we perform rollout on the randomized delivery promise and fulfillment policy given in Section 7.

Uncoordinated Placement and Greedy Fulfillment (UPG): In UPG, we make the placement decisions by using a heuristic that does not explicitly consider the interaction between placement, delivery promise and fulfillment. The heuristic proceeds as follows. Corresponding to demands for product a from demand region j, we choose an ideal fulfillment center and promise pair given by $(i^*_{aj}, k^*_{aj}) = \arg\max_{(i,k)\in\mathcal{F}\times\mathcal{K}} \theta^a_{jk}\, r^a_{ijk}$, which is the fulfillment center and promise pair that maximizes the immediate expected profit from a demand for product a from demand region j. In this case, the load that the demand for product a from demand region j puts on fulfillment center i at time period t is a Bernoulli random variable with parameter $\gamma^a_{ijt} = \sum_{k\in\mathcal{K}} \mathbb{1}_{(i^*_{aj}=i,k^*_{ij}=k)}\theta^a_{jk}\,\lambda^a_{jt}$. Thus, letting B^a_{ijt} be a Bernoulli random variable with parameter γ^a_{ijt}, we use the random variable $N_{ija} = \sum_{t\in\mathcal{T}} B^a_{ijt}$ to characterize the load that the demands for product a from demand region j put on fulfillment center i over the whole selling horizon. Noting that the demands at different time periods are independent, N_{ija} is a sum of independent Bernoulli random variables. We approximate the cumulative distribution function of N_{ija} by using Monte Carlo simulation.

Given that we place z^a_i units of product a at fulfillment center i, we use $\Pi^a_i(z^a_i)$ as an approximation to the total expected profit generated from these units. To construct this approximation, using the decision variable y^a_{ij} to capture the number of units of product a that we ship from fulfillment center i to demand region j over the whole selling horizon, we set

$$\Pi^a_i(z^a_i) = \mathbb{E}\left\{ \max_{\boldsymbol{y}^a_i \in \mathbb{Z}^{|\mathcal{D}|}_+} \left\{ \sum_{j\in\mathcal{D}} \sum_{k\in\mathcal{K}} \mathbb{1}_{(i^*_{aj}=i, k^*_{aj}=k)}\, r^a_{ijk}\, \theta^a_{jk}\, y^a_{ij} \;:\; \sum_{j\in\mathcal{D}} y^a_{ij} \le z^a_i, \quad y^a_{ij} \le N^a_{ij} \quad \forall\, j\in\mathcal{D} \right\} \right\},$$

where we use the vector $\boldsymbol{y}^a_i = (y^a_{ij} : j \in \mathcal{D})$. The expectation above involves the random variables $\{N^a_{ij} : j \in \mathcal{D}\}$. Cheung and Powell (1996) show that we can compute this expectation efficiently. We explain their approach in Appendix H. Given the placement decisions $\boldsymbol{z}^a = (z^a_i : i \in \mathcal{F})$ for product a, we approximate the total expected profit from this product by using $\sum_{i\in\mathcal{F}} \Pi^a_i(z^a_i)$. To make the placement decisions, we replace $f^a(\boldsymbol{z}^a)$ in the Approximate Placement problem with $\sum_{i\in\mathcal{F}} \Pi^a_i(z^a_i)$ and solve this problem. We can show that $\Pi^a_i(z^a_i)$ defined above is concave in z^a_i. In Appendix I, using this concavity result, we show that if we replace $f^a(\boldsymbol{z}^a)$ in the Approximate Placement problem with $\sum_{i\in\mathcal{F}} \Pi^a_i(z^a_i)$, then we can formulate this problem as a min-cost network flow problem. To make the delivery promise and fulfillment decisions, UPG acts greedily. If the remaining inventories for product a at the beginning of the current time period are given by the vector $\boldsymbol{x}^a = (x^a_i : i \in \mathcal{F})$ and we have a demand for the product from demand region j, then we find a fulfillment center and promise pair $(i^*, k^*) = \arg\max \mathbb{1}_{(x^a_i \ge 1)} \theta^a_{jk}\, r^a_{ijk}$. If the optimal objective value of the last problem is strictly positive, then we offer promise k^* and use fulfillment center i^* upon acceptance of the promise. Otherwise, we offer the dummy promise.

Uncoordinated Placement and Randomized Fulfillment (UPR): In UPR, we make the placement decisions as in UPG and the delivery promise and fulfillment decisions as in APS.

Lagrangian Relaxation (LAG): In LAG, we decompose the dynamic programming formulation for the delivery promise and fulfillment problem for each product by the fulfillment centers. In this way, we obtain approximations to the value functions $\{J^a_t : t \in \mathcal{T}\}$ in the Fulfillment dynamic program, which, in turn, guide all of the placement, delivery promise and fulfillment decisions. We overview LAG but defer its details to Appendix J. In the Fulfillment dynamic program, if we have a demand for product a from demand region j, then we can use at most one fulfillment center and promise pair for this demand. We relax this constraint by associating the nonnegative Lagrange multiplier β^a_j to obtain a relaxed dynamic program. Using $\{\widetilde{J}^a_t : t \in \mathcal{T}\}$ to denote the value functions of the relaxed dynamic program, these value functions are separable functions of the form $\widetilde{J}^a_t(\boldsymbol{x}^a; \boldsymbol{\beta}^a) = \sum_{i\in\mathcal{F}} \widetilde{J}^a_{it}(x^a_i; \boldsymbol{\beta}^a)$, where we use the vector of $\boldsymbol{\beta}^a = (\beta^a_j : j \in \mathcal{D})$ and note that the value functions of the relaxed dynamic program depend on the choice of the Lagrange multipliers. We can compute the value functions $\{\widetilde{J}^a_{it} : t \in \mathcal{T}\}$ efficiently by focusing only on fulfillment center i. For any nonnegative Lagrange multipliers, the value functions from the relaxed dynamic program are upper bounds on the value functions from the Fulfillment dynamic program, so $\widetilde{J}^a_t(\boldsymbol{x}^a; \boldsymbol{\beta}^a) \ge J^a_t(\boldsymbol{x}^a)$

for all $\boldsymbol{x}^a \in \mathbb{Z}_+^n$ and $\boldsymbol{\beta}^a \in \mathbb{R}_+^{|\mathcal{D}|}$. In this case, we can obtain an upper bound on the optimal objective value of the Placement problem by replacing $J_1^a(\boldsymbol{z}^a)$ with $\widetilde{J}_1^a(\boldsymbol{z}^a;\boldsymbol{\beta}^a)$ and solving this problem with any nonnegative Lagrange multipliers. The bounding function $f_{\mathsf{LP}}^a(\boldsymbol{z}^a)$ provides an upper bound on $J_1^a(\boldsymbol{z}^a)$, so we can also get an upper bound on the optimal objective value of the Placement problem by replacing $J_1^a(\boldsymbol{z}^a)$ in the Placement problem with $f_{\mathsf{LP}}^a(\boldsymbol{z}^a)$ and solving the continuous relaxation of this problem. In Appendix J, we give an approach to choose the Lagrange multipliers so that the upper bound that we obtain by using the value functions from the relaxed dynamic program is at least as tight as the one that we obtain by using the bounding function. This discussion settles the choice of the Lagrange multipliers. Let $\widehat{\boldsymbol{\beta}}^a$ be the vector of Lagrange multipliers we choose.

To make the placement decisions, we replace $f^a(\boldsymbol{z}^a)$ in the Approximate Placement problem with $\widetilde{J}_1^a(\boldsymbol{z}^a;\widehat{\boldsymbol{\beta}}^a)$ and solve this problem. On the other hand, to make the delivery promise and fulfillment decisions, we replace $J_{t+1}^a(\boldsymbol{x}^a)$ on the right side of the Fulfillment dynamic program with $\widetilde{J}_{t+1}^a(\boldsymbol{x}^a;\widehat{\boldsymbol{\beta}}^a)$ and solve the maximization problem. In particular, if the remaining inventories for the product a at the beginning of time period t are given by the vector $\boldsymbol{x}^a = (x_i^a : i \in \mathcal{F})$ and we have a demand for the product from demand region j, then we find the fulfillment center and promise pair $(i^*, k^*) = \arg\max \mathbb{1}_{(x_i^a \geq 1)} \theta_{jk}^a \left[r_{ijk}^a + \widetilde{J}_{t+1}^a(\boldsymbol{x}^a - \boldsymbol{e}_i;\widehat{\boldsymbol{\beta}}^a) - \widetilde{J}_{t+1}^a(\boldsymbol{x}^a;\widehat{\boldsymbol{\beta}}^a) \right]$. If the optimal objective value of the last problem is strictly positive, then we offer promise k^* and use fulfillment center i^* upon acceptance of the promise. Otherwise, we offer the dummy promise. The delivery promise and fulfillment decisions of LAG and UPG for a particular demand depend jointly on the inventories at all fulfillment centers. Thus, it is difficult to perform rollout on them, unless one uses Monte Carlo simulation to estimate their value functions. The benchmark UPR uses the same policy as APS.

8.2 Computational Experiments Based on Synthetic Data

We report on computational experiments carried out on synthetically generated datasets. We describe our experimental setup, followed by our computational results.

Experimental Setup: We generate our test problems as follows. In all of our test problems, we have $|\mathcal{F}| = 50$ fulfillment centers and $|\mathcal{D}| = 150$ demand regions. We generate the locations of all fulfillment centers and demand regions uniformly over a 100×100 square. We have $|\mathcal{A}| = 500$ products. Other than the dummy one, there are $|\mathcal{K}| = 2$ possible delivery promises. The first promise represents faster service at more expensive shipping cost, whereas the second promise represents slower service at cheaper shipping cost. We choose five fulfillment centers randomly as hubs. We offer only the fast promise out of hubs and the shipping cost out of hubs is cheaper when compared with non-hubs. To come up with the promise acceptance probabilities, for each product a and demand region j, we sample α_{j1}^a and α_{j2}^a from the uniform distribution over $[0.3, 0.8]$. We set the promise acceptance probabilities for the two promises as $\theta^a_{j1} = \max\{\alpha^a_{j1}, \alpha^a_{j2}\}$ and $\theta^a_{j2} = \min\{\alpha^a_{j1}, \alpha^a_{j2}\}$. To come up with the profits from serving a demand, letting p^a be the margin for product a, we sample p^a from the uniform distribution over $[1,3]$. Using $\mathsf{dist}(i,j)$ to denote the Euclidean distance from fulfillment center i to demand region j and letting $\overline{\mathsf{dist}}$ be the average Euclidean distance between all pairs of fulfillment centers and demand regions, we set the profits from serving a demand as $r^a_{ij1} = p^a - 1.25\,\frac{\mathsf{dist}(i,j)}{\overline{\mathsf{dist}}}$ and $r^a_{ij2} = p^a - \frac{\mathsf{dist}(i,j)}{\overline{\mathsf{dist}}}$, as long as fulfillment center i is non-hub. If p^a takes, for example, its expected value of 2 and $\mathsf{dist}(i,j) = \overline{\mathsf{dist}}$, then we have $r^a_{ij1} = 0.75$ and $r^a_{ij2} = 1$. The first promise is more likely to be accepted, but fulfilling through the first promise incurs a larger shipping cost. If fulfillment center i is a hub, then we offer only the first promise out of this fulfillment center with the corresponding profit $r^a_{ij1} = p^a - 1.25\,\frac{\mathsf{dist}(i,j)}{3\times\overline{\mathsf{dist}}}$, so the shipping cost out of hubs is cheaper. We have $r^a_{ij2} = -\infty$, as we cannot offer the second promise out of a hub.

To come up with the number of units that we need to place for each product, we sample the number of units C^a for product a from the uniform distribution over $\{10-\delta, \ldots, 10+\delta\}$, where δ is a parameter that we vary to control the spread of the number of units for the products. Thus, we have a total of about $10 \times 500 = 5000$ units to be placed over all fulfillment centers, which roughly matches the stocking quantity for a typical bulky product category requiring a specific type of storage bin size. Total capacity at the fulfillment centers exceeds the total number of units for the products by a factor η, where η is another parameter that we vary to control the tightness of the capacities. In particular, to come up with the capacity of each fulfillment center, we set the total capacity of the fulfillment centers as $\overline{U} = \eta \sum_{a\in\mathcal{A}} C^a$. To allocate the total capacity over the different fulfillment centers, for each fulfillment center i, we sample γ_i from the uniform distribution over $[0.1, 1]$ and set the capacity of fulfillment center i as $U_i = \left\lceil \frac{\gamma_i}{\sum_{\ell\in\mathcal{F}} \gamma_\ell} \overline{U} \right\rceil$.

There are $T = 1000$ time periods in the selling horizon. We divide the selling horizon into four seasons such that the probability of getting a demand for a product from a demand region does not change within a season. Furthermore, even if we offer the promise with the smaller acceptance probability, the total expected demand for a product exceeds the number of units available for the product by a factor μ, where μ is the last parameter that we vary to control the scarcity of supply. Thus, the total expected number of arrivals for the demand for product a is μC^a. Using $\{1, \ldots, 4\}$ to denote the set of seasons, to allocate the total expected number of demand arrivals over different seasons, for each season q, we sample ν_q from the uniform distribution over $[0,1]$. Using Λ^a_q to denote the total expected number of demand arrivals for product a in season q, we set $\Lambda^a_q = \frac{\nu_q}{\sum_{\ell=1}^4 \nu_\ell}\,\mu C^a$. The quantity ν_q can interpreted as the demand potential for season q and modulates the total expected demand arrivals for all products in season q.

To come up with the demand arrival probabilities, we proceed as follows. For each product, only a subset of the demand regions produces demand. Using $\mathcal{D}^a$ to denote the subset of demand regions that produce demand for product a, letting K^a be the cardinality of $\mathcal{D}^a$, we sample the K^a from the uniform distribution over $\{1,\ldots,|\mathcal{D}|\}$ and set $\mathcal{D}^a$ be a random subset of $\mathcal{D}$ with cardinality K^a. For each demand region j and product a, we sample ρ_j^a from the uniform distribution over $[0,1]$. If time period t falls into season q, then we set the probability that we have a demand for product a from demand region j at time period t as $\lambda_{jt}^a = \mathbb{1}_{(j\in\mathcal{D}^a)} \frac{\rho_j^a}{\sum_{\ell\in\mathcal{D}^a}\rho_\ell^a} \Lambda_q^a \frac{1}{\theta_{j2}^a \times 250}$, where the first fraction allocates the total expected demand arrivals for product a over different demand regions, whereas the second fraction allocates the total expected demand arrivals for product a over different time periods in the season. Promise acceptance probability in the denominator ensures that the total expected demand for product a exceeds the number of units available by a factor μ, even if we offer the promise with the smaller acceptance probability.

We vary $\delta\in\{0,5,10\}$, $\eta\in\{1.25,1.5,1.75,2\}$ and $\mu\in\{0.8,1\}$ to get 24 test problems. To estimate the total expected profits of the benchmarks, we simulate the policies for 500 sample paths.

Computational Results: We obtain an upper bound on the optimal total expected profit in each test problem. In particular, noting the discussion of LAG in the previous section, using $\widehat{\boldsymbol{\beta}}^a$ to denote the vector of Lagrange multipliers that we use for product a, $\widetilde{J}_1^a(\boldsymbol{z}^a;\widehat{\boldsymbol{\beta}}^a)$ provides an upper bound on $J_1^a(\boldsymbol{z}^a)$. In this case, we can replace $J_1^a(\boldsymbol{z}^a)$ in the Placement problem with $\widetilde{J}_1^a(\boldsymbol{z}^a;\widehat{\boldsymbol{\beta}}^a)$ and solve this problem to obtain an upper bound on the optimal total expected profit. We compute the value functions $\{\widetilde{J}_t^a : t\in\mathcal{T}\}$ by focusing on one fulfillment center at a time so that the value functions have the form $\widetilde{J}_1^a(\boldsymbol{z}^a,\widehat{\boldsymbol{\beta}}^a) = \sum_{i\in\mathcal{F}} \widetilde{J}_{i1}^a(z_i^a,\widehat{\boldsymbol{\beta}}^a)$. Furthermore, we can show that $\widetilde{J}_{i1}^a(z_i^a;\boldsymbol{\beta}^a)$ is concave in z_i^a, in which case, by the same argument in Appendix I, we can compute the upper bound on the optimal total expected profit by solving a min-cost network flow problem. We normalize the total expected profit obtained by each benchmark by using the upper bound.

We give our computational results in Table 2. The first column in the table shows the parameter configuration for our test problems by using the tuple (δ,η,μ). The next four columns show the total expected profits obtained by the four benchmark strategies APS, UPG, UPR and LAG, normalized by the upper bound. The last three columns show the percent gap between the total expected profit obtained by APS and the total expected profits obtained by the remaining three benchmark strategies. Overall, APS performs consistently better than all other benchmark strategies. The average gap between the total expected profits obtained by APS and UPG is 3.83%. By switching from a greedy approach to the randomized delivery promise and fulfillment policy, the average performance gap between APS and UPR reduces to 2.13%. In terms of average performance gaps, LAG sits between UPG and UPR. The average performance gap between APS and LAG is 3.52%.

In our computational results, two factors affect the benefit from coordinating the offline and online decisions. First, as the value of μ gets larger, APS, which tries to coordinate the offline

Params.	Total Exp. Profit				% Gap with APS		
(δ, η, μ)	APS	UPG	UPR	LAG	UPG	UPR	LAG
$(0, 1.25, 0.8)$	87.61	83.92	85.18	82.32	4.22	2.78	6.04
$(0, 1.25, 1.0)$	90.35	85.80	88.01	87.03	5.04	2.59	3.68
$(0, 1.50, 0.8)$	87.67	84.45	85.63	82.59	3.67	2.32	5.79
$(0, 1.50, 1.0)$	90.52	86.49	88.64	87.72	4.45	2.07	3.09
$(0, 1.75, 0.8)$	87.75	84.78	85.92	83.14	3.38	2.09	5.25
$(0, 1.75, 1.0)$	90.74	87.00	89.05	88.18	4.12	1.86	2.81
$(0, 2.00, 0.8)$	87.82	85.12	86.24	83.41	3.08	1.80	5.02
$(0, 2.00, 1.0)$	90.93	87.42	89.38	88.47	3.86	1.70	2.70
$(5, 1.25, 0.8)$	87.66	84.05	85.21	83.02	4.13	2.79	5.30
$(5, 1.25, 1.0)$	90.31	86.07	88.24	87.76	4.70	2.29	2.82
$(5, 1.50, 0.8)$	87.71	84.52	85.66	83.07	3.64	2.34	5.30
$(5, 1.50, 1.0)$	90.48	86.57	88.68	88.33	4.32	1.99	2.38
$(5, 1.75, 0.8)$	87.78	84.91	85.91	83.44	3.27	2.13	4.95
$(5, 1.75, 1.0)$	90.76	87.07	89.05	88.70	4.06	1.88	2.27
$(5, 2.00, 0.8)$	87.93	85.21	86.26	83.80	3.10	1.91	4.70
$(5, 2.00, 1.0)$	90.92	87.41	89.31	89.05	3.85	1.76	2.05
$(10, 1.25, 0.8)$	88.24	84.80	85.81	84.93	3.90	2.75	3.75
$(10, 1.25, 1.0)$	90.56	86.63	88.54	89.58	4.34	2.23	1.09
$(10, 1.50, 0.8)$	88.32	85.27	86.27	84.70	3.46	2.33	4.10
$(10, 1.50, 1.0)$	90.77	87.10	88.96	89.78	4.05	2.00	1.09
$(10, 1.75, 0.8)$	88.39	85.65	86.55	84.90	3.09	2.07	3.94
$(10, 1.75, 1.0)$	90.96	87.54	89.29	89.95	3.76	1.83	1.10
$(10, 2.00, 0.8)$	88.54	85.98	86.83	84.80	2.89	1.93	4.23
$(10, 2.00, 1.0)$	91.20	87.94	89.61	90.24	3.58	1.74	1.05
Average	89.33	85.90	87.43	86.20	3.83	2.13	3.52

Table 2 Total expected profits obtained by the benchmark strategies for the synthetically generated datasets.

and online decisions, provides more pronounced improvements over UPG, which does not explicitly consider the interactions between the offline and online decisions. As μ increases, the total expected demand for each product is a larger fraction of the total number of units to be placed, so the availability of each product becomes scarcer. Under scarcer product availabilities, coordinating the offline and online decisions becomes more important. For the test problems with $\mu = 0.8$ and $\mu = 1$, the average percent gap between the total expected profits obtained by APS and UPG are, respectively, 3.49% and 4.18%. Second, as the value of η gets smaller, APS provides more pronounced improvements over UPG. As η decreases, the capacities of the fulfillment centers get tighter when compared with the total number of units to be placed. For the test problems with $\eta = 2$, $\eta = 1.75$, $\eta = 1.5$ and $\eta = 1.25$, the average percent gap between the total expected profits obtained by APS and UPG are, respectively, 3.39%, 3.61%, 3.93% and 4.49%. To check the statistical significance of these trends, we regressed the performance gaps between APS and UPG on (δ, η, μ) by using the model $y = \beta_0 + \beta_1 \delta + \beta_2 \eta + \beta_3 \mu$. The coefficients came out to be $\beta_0 = 3.04$, $\beta_1 = -0.03$, $\beta_2 = -1.32$ and $\beta_3 = 3.46$. The R-square statistic is 96.62. For each coefficient, we get a p-value of virtually zero for the t-test to check that the coefficient is nonzero.

Beside the rollout policy used by APS, we can plug different delivery promise and fulfillment policies into our approximation framework to try to obtain even better solutions for the joint placement, delivery promise and fulfillment problem. In Appendix K, we report on our experience with a policy based on periodically solving a linear programming approximation. While this policy did not provide improvements over APS, there is about 10% gap between the total expected profits obtained by our solution methods and the upper bounds on the optimal total expected profits, so there may be room for either constructing better policies or obtaining tighter upper bounds.

We performed our computational experiments in Python 3.10 with 3.1 GHz CPU and 16 GB RAM, using Gurobi 9.5 as our linear programming solver. Across all of our test problems, the average running time for APS is 7494 seconds, which includes the time for making the placement decisions and rolling out the randomized policy. The average running times for UPG, UPR and LAG are, respectively, 860, 1164 and 1181 seconds.

8.3 Computational Experiments Based on Online Retail Data

We have access to the order transactions from the online retailer Olist Store in Brazil; see Kaggle (2021). We carry out computational experiments using datasets based on these transactions.

Experimental Setup: The order transactions take place over two years. Olist Store acts as an intermediary marketplace between customers and sellers. Thus, each transaction record includes the price of the purchased product, location of the customer, purchase time stamp, shipping cost, location of the seller and delivery completion time stamp. We explain the main points of how we preprocess the data and give the remaining details in Appendix L. We use two criteria to choose the products to focus on. First, we consider the products for which the ratio between the average selling price and average shipping cost is in the interval $[0.8, 3.5]$. In this way, we ensure that the shipping cost is a considerable portion of the selling price, so making placement decisions carefully becomes important. Second, we consider the products for which the expected demand over each three-month period is at most 50. When the demand quantities are large, the number of units to place for the products also become large, in which case, due to its asymptotic optimality guarantee, our approach already performs quite well. Thus, having smaller numbers of units to place results in more challenging problems. Based on these criteria, we end up with 75 products. Among these 75 products, two products have average selling price below average shipping cost.

We aggregate the seller locations provided in the data to obtain 30 locations to serve as fulfillment centers. We proceed with the understanding that we can store any of the products at each of these locations. Similarly, we aggregate the customer locations to get 80 locations to serve as demand regions. The selling horizon in our model is one year. Each 12-hour period corresponds to a time period in our model, so assuming that there are 360 days in a year, the number of time periods in the selling horizon is $T = 720$. The data does not provide enough information to deduce the number of possible delivery promises, if any, that are offered by the sellers and the probability that a customer accepts a delivery promise. We follow the same setup in our synthetic datasets to come up with the delivery promises. Therefore, we have two possible delivery promises other than the dummy one. To generate the promise acceptance probabilities, proceeding as in Section 8.2, for each product a, we sample α_1^a and α_2^a from the uniform distribution over $[0.3, 0.8]$. For all $a \in \mathcal{A}$ and $j \in \mathcal{D}$, we set the promise acceptance probabilities as $\theta_{j1}^a = \max\{\alpha_1^a, \alpha_2^a\}$ and $\theta_{j2}^a = \min\{\alpha_1^a, \alpha_2^a\}$. The first delivery promise has a larger probability of being accepted, so we interpret the first delivery promise as a faster promise and it will have a larger shipping cost. In Appendix L, we explain how we come up with the profits from serving a demand through the two delivery promises.

To generate the demand arrival probabilities, we split the year into winter, spring, summer and fall seasons with stationary demand arrival probabilities in each season. Using $\{1, \ldots, 4\}$ to index the seasons, let D_s^a be the average demand for product a during season s and γ_j^a be the fraction of time that the demand for product a occurs in demand region j, both computed from the data through simple counting. Using $s(t)$ to denote the season for time period t, noting that there are 180 time periods in each season, for all $a \in \mathcal{A}$, $j \in \mathcal{D}$ and $t \in \mathcal{T}$, we set the demand arrival probabilities as $\lambda_{jt}^a = \frac{1}{180}\, \gamma_j^a \, \frac{D_{s(t)}^a}{\min\{\alpha_1^a, \alpha_2^a\}}$. Thus, even if we offer the delivery promise with low acceptance probability throughout season s, the total expected demand for product a from demand region j is $\gamma_j^a \, D_s^a$. Total expected demand for product a over the whole selling horizon is $\sum_{s=1}^4 D_s^a$, so we set the total available units for product a as $C^a = \lceil \mu \sum_{s=1}^4 D_s^a \rceil$, where we vary μ.

The data does not fully conform to the setup of our model as each seller has its own fulfillment center and there is no central planner that makes the placement decisions. Nevertheless, we use the following approach to come up with the capacities of the fulfillment centers. Assigning the total number of orders received by each seller to a nearby fulfillment center, we compute the fraction of orders that are assigned to each fulfillment center. Letting α_i be the fraction of orders assigned to fulfillment center i in this way and noting that we need at least a total of $\sum_{a \in \mathcal{A}} C^a$ units of capacity to accommodate the available units for all products, we set the capacities of the fulfillment centers as $U_i = \lceil \alpha_i \, \eta \sum_{a \in \mathcal{A}} C^a \rceil$ for all $i \in \mathcal{F}$, where η is another parameter that we vary. We obtain eight parameter configurations by varying $\mu \in \{0.8, 1\}$ and $\eta \in \{1.25, 1.5, 1.75, 2\}$. We use the preceding approach to generate a test problem for each parameter combination.

Computational Results: In Table 3, we give the total expected profits obtained by our benchmarks. The format of this table is identical to that of Table 2. The average gap between the total expected profits obtained by APS and UPG is 4.07%. By using a more sophisticated approach for making the delivery promise and fulfillment decisions, UPR reduces its average performance

Params.	Total Exp. Profit				% Gap with APS		
(η, μ)	APS	UPG	UPR	LAG	UPG	UPR	LAG
$(1.25, 0.8)$	87.47	84.21	85.33	84.18	3.72	2.44	3.76
$(1.25, 1.0)$	90.26	84.42	86.75	91.50	6.48	3.90	-1.38
$(1.50, 0.8)$	87.43	84.48	85.37	83.28	3.38	2.36	4.76
$(1.50, 1.0)$	89.91	85.33	87.37	89.25	5.10	2.83	0.73
$(1.75, 0.8)$	87.75	85.19	85.80	84.19	2.92	2.22	4.06
$(1.75, 1.0)$	89.89	85.98	87.86	88.33	4.35	2.26	1.73
$(2.00, 0.8)$	87.89	85.66	86.09	83.15	2.54	2.05	5.39
$(2.00, 1.0)$	90.27	86.59	88.09	89.45	4.08	2.41	0.91
Average	88.86	85.23	86.58	86.67	4.07	2.56	2.49

Table 3 Total expected profits obtained by the benchmark strategies for the datasets based on Olist Store.

gap with APS to 2.56%. Overall, APS consistently performs better than UPG and UPR. On average, the total expected profits obtained by APS exceed those obtained by LAG by 2.49%, but there is one problem instance for which the performance of APS lags behind that of LAG. The performance gap between APS and LAG is more pronounced when μ gets smaller so that the total expected demand for each product exceeds the number of available units by a larger margin. When the total expected demand for the products exceeds the numbers of available units by larger margins, it is especially important to carefully ration inventory to pick the demands that should be served. To our knowledge, LAG is one of the strongest benchmarks for the class of joint placement, delivery promise and fulfillment problems that we focus on, but this benchmark strategy does not come with a theoretical performance guarantee. It is encouraging that APS, at minimum, performs comparably to LAG while having a theoretical performance guarantee.

9. Conclusions

Considering the problem of coordinating the placement, delivery promise and fulfillment decisions of an online retailer, we give an approach to obtain a $\max\left\{\frac{1}{4+\epsilon}, 1 - O\left(\frac{\sqrt{\log(C_{\min} \wedge U_{\min})}}{(C_{\min} \wedge U_{\min})^{1/3}}\right)\right\}$-approximate solution for any $\epsilon > 0$. There are several directions for future work. First, there are many problem settings other than online retail where offline resource allocation decisions are followed by online rationing decisions so that these two classes of decisions need to be coordinated. It may be possible to follow the blueprint provided by our approximation framework in other problem settings. Furthermore, we can plug different delivery promise and fulfillment policies into our approximation framework to try to obtain better solutions for our problem. In Appendix K, inspired by the work in the revenue management literature, we experiment with a policy that is based on periodically solving a linear programming approximation. We can also try other policies that are specifically designed for online retail problems.

Second, in our model, we offer a delivery promise to each demand, in response to which the customer decides whether to accept the offered delivery promise. We can consider a version of our model, where we offer an assortment of delivery promises to each demand and the customer picks a delivery promise according to a discrete choice model. In this case, we can use a decision variable of the form $h^a_{ij}(S)$ in the Bounding problem, which corresponds to the total expected number of times that we offer the assortment S of delivery promises to a demand for product a from demand region j and plan to use fulfillment center i to serve the demand upon acceptance of the promise. The proofs of Theorems 5.2 and 7.1 go through with minor modifications when we offer an assortment of delivery promises, in which case, we obtain a $1 - O\Big(\frac{\sqrt{\log(C_{\min} \wedge U_{\min})}}{(C_{\min} \wedge U_{\min})^{1/3}}\Big)$-approximate solution to the joint placement, delivery promise and fulfillment problem. However, if we offer an assortment of delivery promises, then it is straightforward to construct counterexamples to demonstrate that the analogue of the bounding function is no longer DR-submodular in the placement decisions even when the customers choose under simple choice models, such as the multinomial logit model. One can construct such counterexamples directly by building on Appendix C in El Housni and Topaloglu (2023), where the authors consider assortment optimization problems under the multinomial logit model and demonstrate the lack of submodularity of the optimal expected revenue when viewed as a function of the underlying product universe. Thus, Theorem 6.2 does not hold when we offer assortments of delivery promises and it is not clear how we can obtain a constant factor performance guarantee. It would be useful to investigate constant factor performance guarantees.

Third, the performance guarantee for our approach is $1 - O\Big(\frac{\sqrt{\log(C_{\min} \wedge U_{\min})}}{(C_{\min} \wedge U_{\min})^{1/3}}\Big)$, which converges to one as the number of units to be placed for the products and capacities of the fulfillment centers get large. In Appendix G, we also show that if the number of units to be placed for all products, capacities of all fulfillment centers and total expected demands for all products from all demand regions are simultaneously scaled with rate μ, then a standard argument yields a performance guarantee of $1 - O\Big(\sqrt{\frac{\log \mu}{\mu}}\Big)$. In a typical resource allocation setting, one expects that scaling the number of units for all products and capacities of all fulfillment centers without scaling the demand makes the problem easier. Thus, scaling the number of units for all products, capacities of all fulfillment centers and total expected demands for all products from all demand regions should intuitively be the most relevant regime. Nevertheless, our performance guarantee allows us to make statements that the performance guarantee of $1 - O\Big(\sqrt{\frac{\log \mu}{\mu}}\Big)$ does not allow. For the performance guarantee of $1 - O\Big(\sqrt{\frac{\log \mu}{\mu}}\Big)$ to hold, the number of units for all products, capacities of all fulfillment centers and total expected demands for all products from all demand regions must be scaled simultaneously with the same rate μ. For the performance guarantee of $1 - O\Big(\frac{\sqrt{\log(C_{\min} \wedge U_{\min})}}{(C_{\min} \wedge U_{\min})^{1/3}}\Big)$ to hold, however, the number of units for different products and capacities of different fulfillment centers can be scaled at different rates. All we need is $C_{\min}$ and U_{min} to get large. Also, the latter performance guarantee is independent of the demand and profit parameters. As we scale the number of units for the products and capacities of the fulfillment centers, some of the demand and profit parameters may be scaled, but some others may not. The total demand may get large, but not the demands from all demand regions. We still get an asymptotically optimal solution. That said, it would be useful to see if we can improve our performance guarantee to $1 - O\left(\frac{\sqrt{\log(C_{\min} \wedge U_{\min})}}{(C_{\min} \wedge U_{\min})^{1/2}}\right)$.

Acknowledgements: We thank the area editor, associate editor and three anonymous referees, whose comments strengthened both the technical results and exposition in the paper.

References

Acimovic, J., S. C. Graves. 2015. Making better fulfillment decisions on the fly in an online retail environment. *Manufacturing & Service Operations Management* **17**(1) 34–51.

Ageev, A. A., M. Sviridenko. 2004. Pipage rounding: A new method of constructing algorithms with proven performance guarantee. *Journal of Combinatorial Optimization* **8** 307–328.

Alaei, S. 2014. Bayesian combinatorial auctions: Expanding single buyer mechanisms to many buyers. *SIAM Journal on Computing* **43**(2) 930–972.

Amil, A., A. Makhdoumi, Y. Wei. 2022. Multi-item order fulfillment revisited: LP formulation and prophet inequality. Available at SSRN 4176274.

Andrews, J. M., V. Farias, , A. I. Khojandi, C. M. Yan. 2019. Primal–dual algorithms for order fulfillment at Urban Outfitters, Inc. *INFORMS Journal on Applied Analytics* **49**(5) 355–370.

Aouad, A., O. Saritac. 2022. Dynamic stochastic matching under limited time. *Operations Research* **70**(4) 2349–2383.

Baek, J., W. Ma. 2022. Technical note – Bifurcating constraints to improve approximation ratios for network revenue management with reusable resources. *Operations Research* **70**(4) 2226–2236.

Bai, Y., O. El Housni, P. Rusmevichientong, H. Topaloglu. 2022. Coordinated inventory stocking and assortment personalization. Tech. rep., Cornell University, New York, NY.

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

Brown, D. B., J. Zhang. 2023. Technical note – On the strength of relaxations of weakly coupled stochastic dynamic programs. *Operations Research* **71**(6) 2374–2389.

Buchbinder, N., M. Feldman. 2019. Constrained submodular maximization via a nonsymmetric technique. *Mathematics of Operations Research* **44**(3) 988–1005.

Calinescu, G., C. Chekuri, M. Pal, J. Vondrak. 2011. Maximizing a monotone submodular function subject to a matroid constraint. *SIAM Journal on Computing* **40**(6) 1740–1766.

Cao, S., S. He. 2019. Discrete convex optimization and applications in supply chain management. D.-Z. Du, P. M. Pardalos, Z. Zhang, eds., *Nonlinear Combinatorial Optimization*. Springer, Cham, Switzerland, 81–121.

Chen, A. I., S. C. Graves. 2021. Item aggregation and column generation for online-retail inventory placement. *Manufacturing & Service Operations Management* **23**(5) 1062–1076.

Chen, Q. G., Y. M. Lei, S. Jasin. 2023. Real-time spatial-intertemporal pricing and relocation in a ride-hailing network: Near-optimal policies and the value of dynamic pricing. *Operations Research* (forthcoming).

Chen, X., J. Feldman, S. H. Jung, P. Kouvelis. 2022. Approximation schemes for the joint inventory selection and online resource allocation problem. *Production and Operations Management* **31**(8) 3143–3159.

Cheung, R. K., W. B. Powell. 1996. An algorithm for multistage dynamic networks with random arc capacities, with an application to dynamic fleet management. *Operations Research* **44**(6) 951–963.

Cooper, W. L. 2002. Asymptotic behavior of an allocation policy for revenue management. *Operations Research* **50**(4) 720–727.

DeValve, L., S. Pekec, Y. Wei. 2022. Approximate submodularity in network design problems. *Operations Research* (forthcoming).

DeValve, L., Y. Wei, D. Wu, R. Yuan. 2023. Understanding the value of fulfillment flexibility in an online retailing environment. *Manufacturing & Service Operations Management* **25**(2) 391–408.

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

Ene, A., H. Nguyen. 2020. Parallel algorithm for non-monotone DR-submodular maximization. H. D. III, A. Singh, eds., *Proceedings of the 37th International Conference on Machine Learning*, vol. 119. PMLR, 2902–2911.

Ene, A., H. L. Nguyen. 2016. Constrained submodular maximization: Beyond $1/e$. *IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*. 248–257.

Ene, A., H. L. Nguyen. 2018. A reduction for optimizing lattice submodular functions with diminishing returns. Tech. rep., University of Warwick, Coventry, UK.

Feldman, M. 2013. Maximization problems with submodular objective functions. Ph.D. thesis, Technion Israel Institute of Technology, Haifa, Israel.

Feldman, M., J. Naor, R. Schwartz. 2011. A unified continuous greedy algorithm for submodular maximization. *Proceedings of IEEE 52nd Annual Symposium on Foundations of Computer Science*. San Diego, CA, 570–579.

Feng, Y., R. Niazadeh, A. Saberi. 2020. Near-optimal Bayesian online assortment of reusable resources. Tech. rep., University of Chicago, Chicago, IL.

Feng, Y., R. Niazadeh, A. Saberi. 2023. Two-stage stochastic matching and pricing with applications to ride hailing. *Operations Research* (forthcoming).

Gallego, G., G. Iyengar, R. Phillips, A. Dubey. 2004. Managing flexible products on a network. Computational Optimization Research Center Technical Report TR-2004-01, Columbia University.

Gallego, G., A. Li, V.-A. Truong, X. Wang. 2015. Online resource allocation with customer choice. Tech. rep., Columbia University, New York, NY.

Gallego, Guillermo, Garret van Ryzin. 1994. Optimal dynamic pricing of inventories with stochastic demand over finite horizons. *Management Science* **40**(8) 999–1020.

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

Jasin, S., S. Kumar. 2012. A re-solving heuristic with bounded revenue loss for network revenue management with customer choice. *Mathematics of Operations Research* **37**(2) 313–345.

Jasin, S., A. Sinha. 2015. An LP-based correlated rounding scheme for multi-item ecommerce order fulfillment. *Operations Research* **63**(6) 1336–1351.

Jiang, J., W. Ma, J. Zhang. 2023. Tight guarantees for multi-unit prophet inequalities and online stochastic knapsack. Available at arXiv 2107.02058.

Kaggle. 2021. Kaggle: Brazilian e-commerce public dataset by Olist. URL `https://www.kaggle.com/datasets/olistbr/brazilian-ecommerce`.

Kashiwabara, K., Y. Okamoto, T. Uno. 2007. Matroid representation of clique complexes. *Discrete Applied Mathematics* **155**(15) 1910–1929.

Kunnumkal, S., K. Talluri. 2016. On a piecewise-linear approximation for network revenue management. *Mathematics of Operations Research* **41**(1) 72–91.

Lee, J., V. S. Mirrokni, V. Nagarajan, M. Sviridenko. 2010. Maximizing nonmonotone submodular functions under matroid or knapsack constraints. *SIAM Journal on Discrete Mathematics* **23**(4) 2053–2078.

Lei, M., S. Liu, S. Jasin, A. Vakhutinsky. 2023. Joint inventory and pricing for a one-warehouse multistore problem: Spiraling phenomena, near optimal policies, and the value of dynamic pricing. *Operations Research* (forthcoming).

Lei, Y. M., S. Jasin, J. Uichanco, A. Vakhutinsky. 2022. Joint product framing (display, ranking, pricing) and order fulfillment under the multinomial logit model for e-commerce retailers. *Manufacturing & Service Operations Management* **24**(3) 1529–1546.

Lim, Y. F., S. Jiu, M. Ang. 2021. Integrating anticipative replenishment allocation with reactive fulfillment for online retailing using robust optimization. *Manufacturing & Service Operations Management* **23**(6) 1616–1633.

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

Ma, Y., P. Rusmevichientong, M. Sumida, H. Topaloglu. 2020. An approximation algorithm for network revenue management under nonstationary arrivals. *Operations Research* **68**(3) 834–855.

Manshadi, V., S. Rodilitz. 2022. Online policies for efficient volunteer crowdsourcing. *Management Science* **68**(9) 6572–6590.

Nemhauser, G., L. Wolsey, M. Fisher. 1978. An analysis of approximations for maximizing submodular set functions—i. *Mathematical Programming* **14** 265–294.

Petrank, E. 1994. The hardness of approximation: Gap location. *Computational Complexity* **4**(2) 133–157.

Rusmevichientong, P., M. Sumida, H. Topaloglu. 2020. Dynamic assortment optimization for reusable products with random usage durations. *Management Science* **66**(7) 2820–2844.

Rusmevichientong, P., M. Sumida, H. Topaloglu, Y. Bai. 2023. Revenue management with heterogeneous resources: Unit resource capacities, advance bookings, and itineraries over time intervals. *Operations Research* **71**(6) 2196–2216.

Shapley, L. S. 1962. Complements and substitutes in the optimal assignment problem. *Naval Research Logistics Quarterly* **9**(1) 45–48.

Stein, C., V.-A. Truong, X. Wang. 2020. Advance service reservations with heterogeneous customers. *Management Science* **66**(7) 2929–2950.

Talluri, K., G. van Ryzin. 1998. An analysis of bid-price controls for network revenue management. *Management Science* **44**(11) 1577–1593.

Talluri, Kalyan, Garrett van Ryzin. 1999. A randomized linear programming method for computing network bid prices. *Transportation Science* **33**(2) 207–216.

Topaloglu, H. 2009. Using Lagrangian relaxation to compute capacity-dependent bid-prices in network revenue management. *Operations Research* **57**(3) 637–649.

Vondrak, J. 2017. Lecture note on polyhedral techniques in combinatorial optimization (Lecture 15). URL `https://theory.stanford.edu/~jvondrak/MATH233B-2017/lec15.pdf`.

Wang, X., V.-A. Truong, D. Bank. 2016. Online advance admission scheduling for services, with customer preferences. Tech. rep., Columbia University, New York, NY.

Wei, L., R. Kapuscinski, S. Jasin. 2021. Shipping consolidation across two warehouses with delivery deadline and expedited options for e-commerce and omni-channel retailers. *Manufacturing & Service Operations Management* **23**(6) 1634–1650.

Electronic Companion:
Joint Placement, Delivery Promise and Fulfillment in Online Retail

Yicheng Bai, Paat Rusmevichientong, Huseyin Topaloglu

Appendix A: Upper Bound on the Optimal Total Expected Profit

We give a proof for Lemma 4.1. Fix the inventory placement vector for product a as $\boldsymbol{z}^a$. We define two Bernoulli random variables. The random variable W^a_{ijkt} takes value one if we have a demand for product a from demand region j at time period t, the optimal policy offers promise k to this demand and plans to use fulfillment center i upon acceptance of the promise. The random variable X^a_{ijkt} takes value one if we have a demand for product a from demand region j at time period t, the optimal policy offers promise k to this demand and plans to use fulfillment center i upon acceptance of the promise and the customer accepts the promise. The total profit of the optimal policy from product a is $\sum_{t\in\mathcal{T}}\sum_{i\in\mathcal{F}}\sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}} r^a_{ijk}\, X^a_{ijkt}$. By the definition of W^a_{ijkt} and X^a_{ijkt}, we have $\mathbb{E}\{X^a_{ijkt}\} = \theta^a_{jk}\,\mathbb{E}\{W^a_{ijkt}\}$, so letting $\overline{w}^a_{ijk} = \sum_{t\in\mathcal{T}}\mathbb{E}\{W^a_{ijkt}\}$, the optimal total expected profit from product a is $\sum_{t\in\mathcal{T}}\sum_{i\in\mathcal{F}}\sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}} r^a_{ijk}\,\mathbb{E}\{X^a_{ijkt}\} = \sum_{i\in\mathcal{F}}\sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}} r^a_{ijk}\,\theta^a_{jk}\,\overline{w}^a_{ijk}$. We claim that the solution $\overline{\boldsymbol{w}}^a = (\overline{w}^a_{ijk} : i\in\mathcal{F},\ j\in\mathcal{D},\ k\in\mathcal{K})$ is feasible to the Bounding problem. The total number of times that the optimal policy uses fulfillment center i to serve any promise for product a over the whole selling horizon cannot exceed the number of units of product a placed at fulfillment center i, so $\sum_{t\in\mathcal{T}}\sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}} X^a_{ijkt} \le z^a_i$. Taking the expectations in the last inequality, noting that $\mathbb{E}\{X^a_{ijkt}\} = \theta^a_{jk}\,\mathbb{E}\{W^a_{ijkt}\}$, we get $\sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}} \theta^a_{jk}\,\overline{w}^a_{ijk} \le z^a_i$, which implies that $\overline{\boldsymbol{w}}^a$ satisfies the first constraint in the Bounding problem. Let D^a_j be the total amount of demand for product a from demand region j over the whole selling horizon. Note that $\mathbb{E}\{D^a_j\} = \sum_{t\in\mathcal{T}}\lambda^a_{jt}$. Over the whole selling horizon, the total number of times that the optimal policy offers any promise and plans to use any fulfillment center to serve a demand for product a from demand region j cannot exceed the total demand. Thus, we have $\sum_{t\in\mathcal{T}}\sum_{i\in\mathcal{F}}\sum_{k\in\mathcal{K}} W^a_{ijkt} \le D^a_j$, so taking expectations in the last inequality yields $\sum_{i\in\mathcal{F}}\sum_{k\in\mathcal{K}} \overline{w}^a_{ijk} \le \sum_{t\in\mathcal{T}}\lambda^a_{jt}$, which implies that $\overline{\boldsymbol{w}}^a$ satisfies the second constraint in the Bounding problem, establishing the claim. In this case, we obtain

$$f^a_{\mathsf{LP}}(\boldsymbol{z}^a) \;\ge\; \sum_{i\in\mathcal{F}}\sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}} r^a_{ijk}\,\theta^a_{jk}\,\overline{w}^a_{ijk} \;=\; J^a_1(\boldsymbol{z}^a),$$

where the inequality holds since $\overline{\boldsymbol{w}}^a$ is a feasible solution to the Bounding problem and the equality holds since $\sum_{i\in\mathcal{F}}\sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}} r^a_{ijk}\,\theta^a_{jk}\,\overline{w}^a_{ijk}$ is the optimal total expected profit from product a. ■

Appendix B: Computational Complexity of Approximate Placement

We give a proof for Theorem 5.1. Given an undirected graph and a subset of vertices, we say that a particular edge is covered by the subset if the subset includes a vertex adjacent to the edge. We show that the Approximate Placement problem is a special case of the maximum vertex cover problem over a graph with a bounded degree of B. The latter problem is defined as follows.

Maximum Vertex Cover:

Given an undirected graph $(\mathcal{V},\mathcal{E})$ with degree B, as well as positive integer K, we want to find a subset of vertices with cardinality of at most K that covers the largest number of edges.

Unless P = NP, there is no polynomial-time approximation scheme for the maximum vertex cover problem; see Petrank (1994). We use the vector $\boldsymbol{z}=(z_i : i\in\mathcal{V})\in\{0,1\}^{|\mathcal{V}|}$ to denote the chosen subset of vertices in maximum vertex cover, where $z_i = 1$ if and only if vertex i is in the chosen subset. Total number of edges covered by the vector $\boldsymbol{z}$ is $\theta(\boldsymbol{z})=\sum_{(u,v)\in\mathcal{E}}\mathbb{1}_{(z_u=1 \text{ or } z_v=1)}$, so we can formulate the maximum vertex cover problem as $\max_{\boldsymbol{z}\in\{0,1\}^{|\mathcal{V}|}}\{\theta(\boldsymbol{z}) : \sum_{i\in\mathcal{V}} z_i\le K\}$. We give a linear program to compute the number of edges covered by the vector $\boldsymbol{z}$. We use the decision variables $\boldsymbol{y}=((y_{u,(u,v)},y_{v,(u,v)}) : (u,v)\in\mathcal{E})\in\mathbb{R}_+^{2|\mathcal{E}|}$ with the interpretation that $y_{u,(u,v)}=1$ if and only if we cover edge (u,v) by including vertex u in the subset. We will not need to restrict these decision variables to be binary. For fixed $\boldsymbol{z}\in\{0,1\}^{|\mathcal{V}|}$, consider the problem

$$\eta(\boldsymbol{z})=\max_{\boldsymbol{y}\in\mathbb{R}_+^{2|\mathcal{E}|}}\left\{\sum_{(u,v)\in\mathcal{E}}\Big(y_{u,(u,v)}+y_{v,(u,v)}\Big) \;:\; y_{u,(u,v)}+y_{v,(u,v)}\le 1 \quad \forall (u,v)\in\mathcal{E},\right.$$
$$\left.\sum_{(u,v)\in\mathcal{E}}\mathbb{1}_{(i=u \text{ or } i=v)}\, y_{i,(u,v)}\le B\, z_i \quad \forall i\in\mathcal{V}\right\}. \tag{1}$$

In the next lemma, we show that we can use the linear program above to compute the number of edges covered by a subset of vertices in a graph with degree B.

Lemma B.1 (Counting Covered Edges) *Noting that the optimal objective value of problem* (1) *as a function of* $\boldsymbol{z}$ *is* $\eta(\boldsymbol{z})$*, we have* $\eta(\boldsymbol{z})=\theta(\boldsymbol{z})$ *for all* $\boldsymbol{z}\in\{0,1\}^{|\mathcal{V}|}$.

Proof: We claim that $\theta(\boldsymbol{z})\ge\eta(\boldsymbol{z})$ for any $\boldsymbol{z}\in\{0,1\}^{|\mathcal{V}|}$. Fix $\boldsymbol{z}\in\{0,1\}^{|\mathcal{V}|}$ and let $\boldsymbol{y}^*$ be an optimal solution to problem (1) with this particular value of $\boldsymbol{z}$. If $z_u=0$ or $z_v=0$, then the second constraint in (1) implies that $y^*_{u,(u,v)}=0$ and $y^*_{v,(u,v)}=0$. In this case, by the first constraint in (1), we have $y^*_{u,(u,v)}+y^*_{v,(u,v)}\le\mathbb{1}_{(z_u=1 \text{ or } z_v=1)}$ for all $(u,v)\in\mathcal{E}$. Adding the inequality over all $(u,v)\in\mathcal{E}$, we obtain $\eta(\boldsymbol{z})=\sum_{(u,v)\in\mathcal{E}}(y^*_{u,(u,v)}+y^*_{v,(u,v)})\le\sum_{(u,v)\in\mathcal{V}}\mathbb{1}_{(z_u=1 \text{ or } z_v=1)}=\theta(\boldsymbol{z})$. Next, we claim that $\eta(\boldsymbol{z})\ge\theta(\boldsymbol{z})$. Fixing $\boldsymbol{z}\in\{0,1\}^{|\mathcal{V}|}$, we define the solution $\widehat{\boldsymbol{y}}$ to problem (1) as follows. If $(z_u,z_v)=(0,0)$, then set $\widehat{y}_{u,(u,v)}=0$ and $\widehat{y}_{v,(u,v)}=0$. If $(z_u,z_v)=(1,0)$, then set $\widehat{y}_{u,(u,v)}=1$ and $\widehat{y}_{v,(u,v)}=0$. Similarly, if $(z_u,z_v)=(0,1)$, then set $\widehat{y}_{u,(u,v)}=0$ and $\widehat{y}_{v,(u,v)}=1$. Lastly, if $(z_u,z_v)=(1,1)$, then we arbitrarily set $\widehat{y}_{u,(u,v)}=1$ and $\widehat{y}_{v,(u,v)}=0$. By the definition of $\widehat{\boldsymbol{y}}$, we have $\widehat{y}_{u,(u,v)}+\widehat{y}_{v,(u,v)}=\mathbb{1}_{(z_u=1 \text{ or } z_v=1)}$, which implies that $\widehat{\boldsymbol{y}}$ satisfies the first constraint in (1). On the other hand, by the definition of $\widehat{\boldsymbol{y}}$, if $z_i = 0$, then $\widehat{y}_{i,(u,v)} = 0$ and $\widehat{y}_{i,(v,u)} = 0$ when $i = u$ or $i = v$, so $\widehat{\boldsymbol{y}}$ satisfies the second constraint in (1) when $z_i = 0$. Noting that we have a graph with degree B, if $z_i = 1$, then the left side of the second constraint adds up to at most B, whereas the right side of the constraint is B, so $\widehat{\boldsymbol{y}}$ satisfies the second constraint in (1) when $z_i = 1$. Thus, $\widehat{\boldsymbol{y}}$ is a feasible solution to problem (1). In this case, using the fact that $\widehat{y}_{u,(u,v)} + \widehat{y}_{v,(u,v)} = \mathbb{1}_{(z_u=1 \text{ or } z_v=1)}$, we obtain $\eta(\boldsymbol{z}) \geq \sum_{(u,v)\in\mathcal{E}} (\widehat{y}_{u,(u,v)} + \widehat{y}_{v,(u,v)}) = \sum_{(u,v)\in\mathcal{E}} \mathbb{1}_{(z_u=1 \text{ or } z_v=1)} = \theta(\boldsymbol{z})$. The lemma follows from the two claims. ■

We use the lemma above to give a proof for Theorem 5.1.

Proof of Theorem 5.1:

Using the decision variable w^a_{ijk} with the same interpretation as in the Bounding problem, the Approximate Placement with the bounding function f^a_{LP} is equivalently given by

$$\max_{(\boldsymbol{z},\boldsymbol{w})\in\mathbb{Z}_+^{n\times m}\times\mathbb{R}_+^{n\times|\mathcal{D}|\times|\mathcal{K}|\times m}} \left\{ \sum_{a\in\mathcal{A}}\sum_{i\in\mathcal{F}}\sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}} r^a_{ijk}\,\theta^a_{jk}\,w^a_{ijk} \;:\; \sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}} \theta^a_{jk}\,w^a_{ijk} \leq z^a_i \quad \forall\, i\in\mathcal{F},\ a\in\mathcal{A}, \right.$$
$$\sum_{i\in\mathcal{F}}\sum_{k\in\mathcal{K}} w^a_{ijk} \leq \sum_{t\in\mathcal{T}} \lambda^a_{jt} \quad \forall\, j\in\mathcal{D},\ a\in\mathcal{A},$$
$$\left. \sum_{i\in\mathcal{F}} z^a_i = C^a \quad \forall\, a\in\mathcal{A},\ \sum_{a\in\mathcal{A}} z^a_i \leq U_i \quad \forall\, i\in\mathcal{F} \right\}, \tag{2}$$

where we use the vectors $\boldsymbol{z} = (z^a_i : i\in\mathcal{F},\ a\in\mathcal{A})$ and $\boldsymbol{w} = (w^a_{ijk} : i\in\mathcal{F},\ j\in\mathcal{D},\ k\in\mathcal{K},\ a\in\mathcal{A})$. We show that the maximum vertex cover is a special case of the problem above.

Given a maximum vertex cover problem over the undirected graph $(\mathcal{V},\mathcal{E})$ with degree B and the limit K on the number of vertices selected, we consider the following instance of our joint placement, delivery promise and fulfillment problem. The set of fulfillment centers corresponds to the set of vertices $\mathcal{V}$. We can store infinite number of units at each fulfillment center. The set of demand regions corresponds to the set of edges $\mathcal{E}$. Thus, we index the demand regions by (u,v), where u and v are two vertices. There is one product. We have K units of the product to place over all fulfillment centers. From each demand region, the total demand for the product takes the deterministic value of one. There is one promise other than the dummy promise and all customers accept this promise with probability $\frac{1}{B}$. If we use fulfillment center i to serve a demand at demand region (u,v), then we obtain a profit of B whenever $i = u$ or $i = v$. If $i \neq u$ and $i \neq v$, then we obtain a profit of zero. We argue that the Approximate Placement problem in (2) for this problem instance is equivalent to the maximum vertex cover problem. Next, we write problem (2) for this problem instance, which corresponds to the Approximate Placement problem with $f^a = f^a_{\mathsf{LP}}$.

We obtain nonzero profit only if we serve a demand from demand region (u,v) from fulfillment center u or v. In this case, dropping the index for the single product and single promise, we capture the decision variables in (2) as $\boldsymbol{w} = ((w_{u,(u,v)}, w_{v,(u,v)}) : (u,v) \in \mathcal{E})$ and $\boldsymbol{z} = (z_i : i \in \mathcal{V})$. Therefore, we write problem (2) for the problem instance in the previous paragraph as

$$\max_{(\boldsymbol{z},\boldsymbol{w}) \in \mathbb{Z}_+^{|\mathcal{V}|} \times \mathbb{R}_+^{2|\mathcal{E}|}} \left\{ \sum_{(u,v)\in\mathcal{E}} \Big(w_{u,(u,v)} + w_{v,(u,v)} \Big) \; : \; \sum_{(u,v)\in\mathcal{E}} \mathbb{1}_{(i=u \text{ or } i=v)} \frac{1}{B} w_{i,(u,v)} \le z_i \quad \forall\, i \in \mathcal{V}, \right.$$
$$\left. w_{u,(u,v)} + w_{v,(u,v)} \le 1 \quad \forall\, (u,v) \in \mathcal{E}, \quad \sum_{i\in\mathcal{V}} z_i = K \right\},$$

where we use the fact that we can store infinite number of units at each fulfillment center, so we drop the last constraint in problem (2).

If we replace the last constraint with less than or equal to, then the optimal objective value does not change. Arranging the terms in the first constraint, the problem above is equivalent to

$$\max_{(\boldsymbol{z},\boldsymbol{w}) \in \mathbb{Z}_+^{|\mathcal{V}|} \times \mathbb{R}_+^{2|\mathcal{E}|}} \left\{ \sum_{(u,v)\in\mathcal{E}} \Big(w_{u,(u,v)} + w_{v,(u,v)} \Big) \; : \; \sum_{(u,v)\in\mathcal{E}} \mathbb{1}_{(i=u \text{ or } i=v)} w_{i,(u,v)} \le B\, z_i \quad \forall\, i \in \mathcal{V}, \right.$$
$$\left. w_{u,(u,v)} + w_{v,(u,v)} \le 1 \quad \forall\, (u,v) \in \mathcal{E}, \quad \sum_{i\in\mathcal{V}} z_i \le K \right\}. \quad (3)$$

The graph has degree of B, so there are at most B decision variables on the left side of the first constraint. Since $w_{u,(u,v)} \le 1$ by the second constraint, we have $z_i \in \{0,1\}$ in an optimal solution.

By Lemma B.1, if we solve problem (3) for fixed $\boldsymbol{z}$, then the optimal objective value is $\theta(\boldsymbol{z})$, so (3) is equivalent to $\max_{\boldsymbol{z}\in\{0,1\}^{|\mathcal{V}|}} \{\theta(\boldsymbol{z}) : \sum_{i\in\mathcal{V}} z_i \le K\}$, which is the maximum vertex cover. ■

Appendix C: Approximating the Placement Problem Through Submodular Set Functions

We show that we can formulate the Approximate Placement problem as the problem of maximizing a monotone and submodular set function subject to two partition constraints. Creating $C_{\max}$ copies of each product, define the ground set $\mathcal{G} = \{(i,a,k) : i \in \mathcal{F},\ a \in \mathcal{A},\ k = 1,\ldots,C_{\max}\}$, so the ground set includes one element for each copy of product a that we can place at fulfillment center i. We will represent the placement decisions as a subset of the ground set $\mathcal{G}$. In the placement decision represented by the subset $S \subseteq \mathcal{G}$, if we have $(i,a,k) \in S$, then we place the k-th copy of product a at fulfillment center i. Let $\mathcal{N}_i^a = \{(i,a,k) : k = 1,\ldots,C_{\max}\}$ to capture the elements of the ground set that correspond to the copies of product a that we place at fulfillment center i. To express our bounding function f_{LP}^a as a set function, for any subset S of the ground set, we define the vector of placement decisions $\overline{\boldsymbol{z}}(S) = (\overline{z}_i^a(S) : i \in \mathcal{F},\ a \in \mathcal{A})$ as $\overline{z}_i^a(S) = |S \cap \mathcal{N}_i^a|$ for all $i \in \mathcal{F}$ and $a \in \mathcal{A}$. Therefore, $\overline{z}_i^a(S)$ captures the number of elements in S corresponding to the copies of product a placed at fulfillment center i. As a function of the subsets of the ground set, using the vector $\overline{\boldsymbol{z}}^a(S) = (\overline{z}_i^a(S) : i \in \mathcal{F})$, we express our bounding function as $g_{\mathsf{LP}}^a(S) = f_{\mathsf{LP}}^a(\overline{\boldsymbol{z}}^a(S))$. Considering the set function $g^a : 2^{\mathcal{G}} \to \mathbb{R}_+$ that maps subsets of the ground set $\mathcal{G}$ to positive reals, the function g^a is monotone if we have $g^a(T) \ge g^a(S)$ for all $T, S \subseteq \mathcal{G}$ with $T \supseteq S$. On the other hand, the function g^a is submodular if we have $g^a(T \cup \{(i,b,k)\}) - g^a(T) \le g^a(S \cup \{(i,b,k)\}) - g^a(S)$ for all $T, S \subseteq \mathcal{G}$ with $T \supseteq S$ and $(i,b,k) \in \mathcal{G} \setminus T$. In the next lemma, using the fact that f^a_{LP} is monotone and DR-submodular, we show that g^a_{LP} is monotone and submodular.

Lemma C.1 (Monotone and Submodular) *Defining $g^a_{\mathsf{LP}} : 2^{\mathcal{G}} \to \mathbb{R}_+$ as $g^a_{\mathsf{LP}}(S) = f^a_{\mathsf{LP}}(\overline{\boldsymbol{z}}^a(S))$ for any $S \subseteq \mathcal{G}$, the function g^a_{LP} is a monotone and submodular set function.*

Proof: Fix $T, S \subseteq \mathcal{G}$ with $T \supseteq S$ and $(i,b,k) \in \mathcal{G} \setminus T$. Because $T \supseteq S$, we have $|T \cap \mathcal{N}^a_i| \ge |S \cap \mathcal{N}^a_i|$, so $\overline{\boldsymbol{z}}^a(T) \ge \overline{\boldsymbol{z}}^a(S)$. Thus, the monotonicity of f^a_{LP} yields $g^a_{\mathsf{LP}}(T) = f^a_{\mathsf{LP}}(\boldsymbol{z}^a(T)) \ge f^a_{\mathsf{LP}}(\boldsymbol{z}^a(S)) = g^a_{\mathsf{LP}}(S)$, which implies that g^a_{LP} is also monotone. To check the submodularity of g^a_{LP}, consider the case $a = b$. Because $(i,b,k) \notin T$ and $a = b$, we have $|(T \cup \{(i,b,k)\}) \cap \mathcal{N}^a_i| = |T \cap \mathcal{N}^a_i| + 1$, so we get $\overline{\boldsymbol{z}}^a(T \cup \{(i,a,k)\}) = \overline{\boldsymbol{z}}^a(T) + \boldsymbol{e}_i$. Similarly, $\overline{\boldsymbol{z}}^a(S \cup \{(i,b,k)\}) = \overline{\boldsymbol{z}}^a(S) + \boldsymbol{e}_i$. Thus, we have

$$\begin{aligned} g^a_{\mathsf{LP}}(T \cup \{(i,b,k)\}) - g^a_{\mathsf{LP}}(T) &= f^a_{\mathsf{LP}}(\overline{\boldsymbol{z}}^a(T) + \boldsymbol{e}_i) - f^a_{\mathsf{LP}}(\overline{\boldsymbol{z}}^a(T)) \\ &\le f^a_{\mathsf{LP}}(\overline{\boldsymbol{z}}^a(S) + \boldsymbol{e}_i) - f^a_{\mathsf{LP}}(\overline{\boldsymbol{z}}^a(S)) = g^a_{\mathsf{LP}}(S \cup \{(i,b,k)\}) - g^a_{\mathsf{LP}}(S), \end{aligned}$$

where the inequality holds because f^a_{LP} is DR-submodular and $\overline{\boldsymbol{z}}^a(T) \ge \overline{\boldsymbol{z}}^a(S)$. If $a \ne b$, then $|(T \cup \{(i,b,k)\}) \cap \mathcal{N}^a_i| = |T \cap \mathcal{N}^a_i|$ and $|(S \cup \{(i,b,k)\}) \cap \mathcal{N}^a_i| = |S \cap \mathcal{N}^a_i|$, so all differences are zero. ■

To capture the constraints of the Approximate Placement problem when we use subsets of the ground set to represent the placement decisions, define $\mathcal{P}^a = \{(i,a,k) : i \in \mathcal{F},\ k = 1, \ldots, C_{\max}\}$, so that $\mathcal{P}^a$ includes the elements of the ground set that correspond to the copies of product a that we place at all fulfillment centers. Note that the collection $\{\mathcal{P}^a : a \in \mathcal{A}\}$ partitions the ground set $\mathcal{G}$ in the sense that we have $\mathcal{P}^a \cap \mathcal{P}^b = \varnothing$ for $a \ne b$ and $\cup_{a \in \mathcal{A}} \mathcal{P}^a = \mathcal{G}$. Similarly, define $\mathcal{R}_i = \{(i,a,k) : a \in \mathcal{A},\ k = 1, \ldots, C_{\max}\}$, so that $\mathcal{R}_i$ includes the elements of the ground set that correspond to the copies of all products that we place at fulfillment center i. The collection $\{\mathcal{R}_i : i \in \mathcal{F}\}$ partitions the ground set $\mathcal{G}$ as well. Considering the placement decisions represented by the subset $S \subseteq \mathcal{G}$, $|S \cap \mathcal{P}^a|$ captures the number of copies of product a that we place at all fulfillment centers, whereas $|S \cap \mathcal{R}_i|$ captures the number of copies of all products that we place at fulfillment center i. Thus, the Approximate Placement problem with $f^a = f^a_{\mathsf{LP}}$ is equivalent to

$$\max_{S \subseteq \mathcal{G}} \left\{ \sum_{a \in \mathcal{A}} g^a_{\mathsf{LP}}(S) \ : \ |S \cap \mathcal{P}^a| = C^a \quad \forall\, a \in \mathcal{A},\ |S \cap \mathcal{R}_i| \le U_i \quad \forall\, i \in \mathcal{F} \right\}.$$

The constraints in the problem above are two partition constraints; see Section 3 in Kashiwabara et al. (2007). Nemhauser et al. (1978) consider the problem of maximizing a monotone and submodular set function under k partition constraints and show that a greedy algorithm provides a $\frac{1}{k+1}$-approximate solution. For the same problem, Lee et al. (2010) give a more involved algorithm that provides a $\frac{1}{k+\epsilon}$-approximate solution for any $\epsilon > 0$. The running time of both of these approaches is polynomial in the number of elements of the ground set. Noting that $|\mathcal{G}|$ is linear in $C_{\max}$, using these results, for any $\epsilon > 0$, in running time polynomial in $C_{\max}$, we can obtain a $\frac{1}{2+\epsilon}$-approximate solution to the Approximate Placement problem.

Appendix D: Approximating the Placement Problem Through Multilinear Extensions

We use a continuous relaxation based on the multilinear extension of the bounding function to obtain an approximate solution to the Approximate Placement problem.

D.1 Main Result

To capture the objective function of the Approximate Placement problem with the bounding function f^a_{LP}, we define $f : \mathbb{Z}_+^{n\times m} \to \mathbb{R}_+$ as $f(\boldsymbol{z}) = \sum_{a\in\mathcal{A}} f^a_{\mathsf{LP}}(\boldsymbol{z}^a)$. Also, we define the polytope

$$\mathcal{P} = \left\{ \boldsymbol{z} \in \mathbb{R}_+^{n\times m} : \sum_{i\in\mathcal{F}} z_i^a \le C^a \quad \forall\, a\in\mathcal{A}, \ \sum_{a\in\mathcal{A}} z_i^a \le U_i \quad \forall\, i \in \mathcal{F} \right\}.$$

Because f^a_{LP} is monotone and DR-submodular by the discussion in Section 6, f is monotone and DR-submodular. Using $\mathbf{0} \in \mathbb{R}^n_+$ and $\overline{\mathbf{0}} \in \mathbb{R}_+^{n\times m}$ to denote vectors of all zeros, by the definition of f^a_{LP}, we have $f^a_{\mathsf{LP}}(\mathbf{0}) = 0$, so $f(\overline{\mathbf{0}}) = 0$ as well. Furthermore, $\mathcal{P}$ is a polytope in $\mathbb{R}_+^{n\times m}$ that is downward closed, which is to say that if $\boldsymbol{z} \in \mathcal{P}$, then $\boldsymbol{z}' \in \mathcal{P}$ for all $\boldsymbol{z}' \le \boldsymbol{z}$. Because f^a_{LP} is monotone, we can replace the first constraint in the Approximate Placement problem with $\sum_{i\in\mathcal{F}} z_i^a \le C^a$. In this case, the Approximate Placement problem with the bounding function f^a_{LP} is equivalent to

$$\mathsf{OPT} = \max_{\boldsymbol{z}\in\mathcal{P}\cap\mathbb{Z}_+^{n\times m}} f(\boldsymbol{z}). \tag{4}$$

In the next theorem, we give the main result of this section. We devote the rest of this section to showing this result.

Theorem D.1 (Constant Factor Approximation) *There exists an approximation algorithm that provides a vector of placement decisions $\widehat{\boldsymbol{z}} \in \mathcal{P} \cap \mathbb{Z}_+^{n\times m}$ that satisfies $f(\widehat{\boldsymbol{z}}) \ge \left(1 - \frac{1}{e} - \epsilon\right) \mathsf{OPT}$ in running time that is polynomial in n, m, $\log C_{\max}$, $\log U_{\max}$ and $1/\epsilon$.*

If, for example, $\epsilon \le 0.1$, then we get a performance guarantee of $1 - \frac{1}{e} - \epsilon \ge 0.532$. In the proof of Theorem D.1, we follow the framework in Ene and Nguyen (2018) to transform the problem of maximizing a DR-submodular function over an integer lattice to a problem of maximizing a submodular set function over an appropriate ground set. We approximate the latter problem through a continuous relaxation that uses the multilinear extension of the submodular set function. We use pipage rounding to round the solution to the continuous relaxation. To execute this outline, for each $\boldsymbol{z} \in \mathcal{P} \cap \mathbb{Z}_+^{n\times m}$, note that we have $0 \le z_i^a \le (C^a \wedge U_i)$ for all $i \in \mathcal{F}$ and $a \in \mathcal{A}$. Using $\lceil\cdot\rceil$ to denote the round up function, if $z_i^a \in \mathbb{Z}_+$ satisfies $0 \le z_i^a \le (C^a \wedge U_i)$, then we can find $y_i^a(k) \in \{0,1\}$ for $k = 0,1,\ldots,\lceil\log(C^a \wedge U_i)\rceil$ to express z_i^a as

$$z_i^a \;=\; \sum_{k=0}^{\lceil\log(C^a\wedge U_i)\rceil} 2^k \times y_i^a(k). \tag{5}$$

We define the ground set $\mathcal{E} = \big\{(i,a,k) : i \in \mathcal{F},\ \ a \in \mathcal{A},\ \ k = 0,1,\ldots,\lceil\log(C^a \wedge U_i)\rceil\big\}$. Letting $\mathsf{dim} = \sum_{a\in\mathcal{A}}\sum_{i\in\mathcal{F}}(1 + \lceil\log(C^a \wedge U_i)\rceil)$, we have $|\mathcal{E}| = \sum_{a\in\mathcal{A}}\sum_{i\in\mathcal{F}}(1 + \lceil\log(C^a \wedge U_i)\rceil) = \mathsf{dim}$. Let $\mathbf{M} : [0,1]^{|\mathcal{E}|} \to \mathbb{R}_+^{n\times m}$ be a linear mapping that maps a vector in $[0,1]^{|\mathcal{E}|}$ to a vector in $\mathbb{R}_+^{n\times m}$ according to (5). In particular, using the vector $\boldsymbol{y} = (y_i^a(k) : i \in \mathcal{F},\ a \in \mathcal{A},\ k = 0,1,\ldots,\lceil\log(C^a \wedge U_i)\rceil) \in [0,1]^{|\mathcal{E}|}$, the (i,a)-th element of $\mathbf{M}(\boldsymbol{y})$ is given by the expression on the right side of (5). Viewing the linear mapping $\mathbf{M}$ as a matrix, we write $\mathbf{M}(\boldsymbol{y})$ equivalently as $\mathbf{M}\boldsymbol{y}$.

Let the function $g : \{0,1\}^{|\mathcal{E}|} \to \mathbb{R}_+$ be defined as $g(\boldsymbol{y}) = f(\mathbf{M}\boldsymbol{y})$. Because there is a one-to-one correspondence between a vector in $\{0,1\}^{|\mathcal{E}|}$ and a subset of $\mathcal{E}$, we can view g as a set function over the ground set $\mathcal{E}$. In other words, for $A \subseteq \mathcal{E}$, using $\mathbb{1}_A \in \{0,1\}^{|\mathcal{E}|}$ to denote a binary vector with ones in the coordinates corresponding to the elements of A, we use $g(\mathbb{1}_A)$ and $g(A)$ interchangeably. We have $g(\varnothing) = 0$, because $f(\bar{\mathbf{0}}) = 0$. Because f is monotone and DR-submodular, g is monotone and submodular. Thus, using the definition of $\mathbf{M}$, problem (4) is equivalent to

$$\mathsf{OPT} \;=\; \max_{\boldsymbol{y}\in\{0,1\}^{|\mathcal{E}|}} \Big\{ g(\boldsymbol{y}) \;:\; \mathbf{M}\boldsymbol{y} \in \mathcal{P} \Big\}. \tag{6}$$

Note that the set $\{\boldsymbol{y} \in \{0,1\}^{|\mathcal{E}|} : \mathbf{M}\boldsymbol{y} \in \mathcal{P}\}$ is downward closed in the sense that if $\mathbf{M}\boldsymbol{y} \in \mathcal{P}$, then $\mathbf{M}\boldsymbol{y}' \in \mathcal{P}$ for all $\boldsymbol{y}' \in \{0,1\}^{|\mathcal{E}|}$ such that $\boldsymbol{y}' \le \boldsymbol{y}$. Thus, we can assume that each element $(i,a,k) \in \mathcal{E}$ of the ground set is feasible, satisfying $\mathbf{M}\mathbb{1}_{\{(i,a,k)\}} \in \mathcal{P}$. Otherwise, we can remove the element from the ground set $\mathcal{E}$ without changing the optimal objective value of problem (6).

We use $\mathsf{R}(\boldsymbol{y})$ to denote a random binary vector in $\{0,1\}^{|\mathcal{E}|}$ obtained by rounding each component of $\boldsymbol{y}$ independently, where for each element $e \in \mathcal{E}$, we round up the e-th component of $\boldsymbol{y}$ to one with probability y_e and round down to zero with probability $1 - y_e$. In this case, consider the <u>*multilinear extension*</u> $G : [0,1]^{|\mathcal{E}|} \to \mathbb{R}_+$ of g, which is defined by

$$G(\boldsymbol{y}) \;=\; \sum_{S\subseteq\mathcal{E}} g(\mathbb{1}_S) \prod_{e\in S} y_e \prod_{e\notin S} (1 - y_e) \;=\; \mathbb{E}\{g(\mathsf{R}(\boldsymbol{y}))\} \;=\; \mathbb{E}\{f(\mathbf{M}\mathsf{R}(\boldsymbol{y}))\}, \tag{7}$$

where the second equality uses the definition of $\mathsf{R}(\boldsymbol{y})$ and the third equality uses the definition of g. Since g is normalized to satisfy $g(\varnothing) = 0$ and is a monotone and submodular set function on the ground set $\mathcal{E}$, we can apply the continuous greedy algorithm below to obtain an approximate solution to the continuous optimization problem $\max_{\boldsymbol{y}\in[0,1]^{|\mathcal{E}|}}\left\{G(\boldsymbol{y}) : \mathbf{M}\boldsymbol{y} \in \mathcal{P}\right\}$. Below, we fix $\epsilon \in (0,1)$ such that $1/\epsilon$ is an integer and recall that $\mathsf{dim} = |\mathcal{E}|$.

Continuous Greedy:

Initialization. Set $\delta = \frac{\epsilon}{2\,\mathsf{dim}^3}$, $t=0$ and $\boldsymbol{y}_0 = \underline{\mathbf{0}}$, where $\underline{\mathbf{0}} \in \mathbb{R}_+^{|\mathcal{E}|}$ is the vector of all zeros.

Algorithm. While $t<1$, do the following four steps.

Step 1. For each $(i,a,k)\in\mathcal{E}$, let $w_{(i,a,k),t} = G(\boldsymbol{y}_t \vee \mathbb{1}_{\{(i,a,k)\}}) - G(\boldsymbol{y}_t)$, where $\mathbb{1}_{\{(i,a,k)\}} \in \{0,1\}^{|\mathcal{E}|}$ is a binary vector with one corresponding to the element (i,a,k) and zero everywhere else. Define the vector $\boldsymbol{w}_t = (w_{(a,i,k),t} : (i,a,k)\in\mathcal{E})$.

Step 2. Solve the linear program $\max_{\boldsymbol{d}\in[0,1]^{|\mathcal{E}|}} \{\boldsymbol{w}_t^\top \boldsymbol{d} : \mathbf{M}\boldsymbol{d} \in \mathcal{P}\}$ and let $\boldsymbol{d}_t$ denote an optimal solution of this linear program.

Step 3. Set $\boldsymbol{y}_{t+\delta} = \boldsymbol{y}_t + \delta\,\boldsymbol{d}_t$.

Step 4. Increment t by δ.

Output. Return the value of $\boldsymbol{y}_1$.

The continuous greedy algorithm is a variation of the algorithm proposed by Calinescu et al. (2011). The specific variant above is given and analyzed in Feldman (2013). We proceed with the understanding that we can compute the function $G(\boldsymbol{y})$ exactly for each $\boldsymbol{y}\in[0,1]^{|\mathcal{E}|}$. This assumption is common in the literature; see, for example, Feldman et al. (2011), Ene and Nguyen (2016), and Ene and Nguyen (2020). We can also obtain an accurate approximation of $G(\boldsymbol{y})$ by sampling the random vector $\mathsf{R}(\boldsymbol{y})$ in (7). In this case, we obtain a randomized algorithm with essentially the same performance guarantee that holds with high probability. The random sampling is a standard technique in the literature; see, for example, Vondrak (2017) and Buchbinder and Feldman (2019). Thus, we can solve the linear program $\max_{\boldsymbol{d}\in[0,1]^{|\mathcal{E}|}}\left\{\boldsymbol{w}_t^\top \boldsymbol{d} : \mathbf{M}\boldsymbol{d}\in\mathcal{P}\right\}$ efficiently. Because $\delta = \epsilon/\mathsf{dim}^3$, the algorithm will terminate after $1/\delta = \mathsf{dim}^3/\epsilon$ iterations. The value of dim is polynomial in n, m, $\log C_{\max}$ and $\log U_{\max}$. Each iteration requires solving a linear program over the polytope $\{\boldsymbol{y}\in[0,1]^{|\mathcal{E}|} : \mathbf{M}\boldsymbol{y}\in\mathcal{P}\}$, which can be done in polynomial time. Theorem 3.1.1 in Feldman (2013) gives a performance guarantee for the output $\boldsymbol{y}_1$ of the continuous greedy algorithm. We give this result in the next lemma. For completeness, we provide a proof in Appendix D.3.

Lemma D.2 (Continuous Greedy) *Noting that* OPT *is the optimal objective value of* (6)*, the continuous greedy algorithm terminates with the output* $\boldsymbol{y}_1$ *that satisfies* $\boldsymbol{y}_1\in[0,1]^{|\mathcal{E}|}$, $\mathbf{M}\boldsymbol{y}_1\in\mathcal{P}$ *and* $G(\boldsymbol{y}_1) \geq \left(1-\frac{1}{e}-\epsilon\right)\mathsf{OPT}$ *in running time that is polynomial in* n*,* m*,* $\log C_{\max}$*,* $\log U_{\max}$ *and* $1/\epsilon$*.*

Using $\overline{\boldsymbol{y}}\in[0,1]^{|\mathcal{E}|}$ to denote the output of the continuous greedy algorithm, let $\overline{\boldsymbol{z}} = \mathbf{M}\overline{\boldsymbol{y}}$. By the lemma above, we have $\overline{\boldsymbol{z}}\in\mathcal{P}$. Using the fact that g is monotone, a simple lemma, given as

Lemma D.5(c) in Appendix D.2, shows that G is monotone as well. In this case, we can assume without loss of generality that $\sum_{i\in\mathcal{F}} \overline{z}_i^a = C^a$, so we place all units of all products.

By exploiting the submodularity of g, we will show that we can round the fractional solution $\overline{\boldsymbol{z}}$ without any loss in performance. Using $\lfloor\cdot\rfloor$ to denote the round down function, we define $\lfloor\boldsymbol{z}\rfloor = (\lfloor z_i^a\rfloor : i\in\mathcal{F},\ a\in\mathcal{A})$. Define the set function $h : 2^{\mathcal{F}\times\mathcal{A}} \to \mathbb{R}_+$, so for any $S\subseteq\mathcal{F}\times\mathcal{A}$,

$$h(S) \;=\; f(\lfloor\overline{\boldsymbol{z}}\rfloor + \mathbb{1}_S).$$

Using the fact that f is a monotone and DR-submodular function on the integer lattice, we can check that h is a monotone and submodular set function on the ground set $\mathcal{F}\times\mathcal{A}$. Let $H : [0,1]^{n\times m}\to\mathbb{R}_+$ be the multilinear extension of h. That is, we use $\mathsf{R}(\boldsymbol{v})$ to denote a binary random vector in $\{0,1\}^{n\times m}$ obtained by rounding each coordinate of $\boldsymbol{v}$ independently, where for each element $(i,a)\in\mathcal{F}\times\mathcal{A}$, we round up the (i,a)-th component to one with probability v_i^a and round down to zero with probability $1-v_i^a$. So, for each $\boldsymbol{v} = (v_i^a : i\in\mathcal{F},\ a\in\mathcal{A})\in[0,1]^{n\times m}$, we have

$$H(\boldsymbol{v}) \;=\; \mathbb{E}\Big\{f(\lfloor\overline{\boldsymbol{z}}\rfloor + \mathsf{R}(\boldsymbol{v}))\Big\}.$$

Observe that if $\boldsymbol{v}$ is integral with $v_i^a\in\{0,1\}$ for all $i\in\mathcal{F}$ and $a\in\mathcal{A}$, then we have $H(\boldsymbol{v}) = f(\lfloor\overline{\boldsymbol{z}}\rfloor+\boldsymbol{v})$.

Noting that $\overline{\boldsymbol{z}} - \lfloor\overline{\boldsymbol{z}}\rfloor\in[0,1]^{n\times m}$, the following lemma shows that $H(\overline{\boldsymbol{z}}-\lfloor\overline{\boldsymbol{z}}\rfloor)$ is an upper bound on $G(\overline{\boldsymbol{y}})$. This result was shown on Page 5 in Ene and Nguyen (2018), but for completeness, we provide a proof in Appendix D.4.

Lemma D.3 (Upper Bound) *We have* $H(\overline{\boldsymbol{z}}-\lfloor\overline{\boldsymbol{z}}\rfloor) \;\geq\; G(\overline{\boldsymbol{y}})$.

Using the results so far in this section, we can give a proof for Theorem D.1.

Proof of Theorem D.1:

Let $\overline{\boldsymbol{v}} = \overline{\boldsymbol{z}} - \lfloor\overline{\boldsymbol{z}}\rfloor\in[0,1]^{n\times m}$. We will show that using the pipage rounding technique pioneered by Ageev and Sviridenko (2004), we can construct a binary vector $\widehat{\boldsymbol{v}}\in\{0,1\}^{n\times m}$ such that $\lfloor\overline{\boldsymbol{z}}\rfloor + \widehat{\boldsymbol{v}}\in\mathcal{P}$ and $H(\widehat{\boldsymbol{v}})\geq H(\overline{\boldsymbol{v}})$. In this case, because $\widehat{\boldsymbol{v}}$ is integral, we have

$$f(\lfloor\overline{\boldsymbol{z}}\rfloor + \widehat{\boldsymbol{v}}) \;=\; H(\widehat{\boldsymbol{v}}) \;\geq\; H(\overline{\boldsymbol{v}}) \;\geq\; G(\overline{\boldsymbol{y}}) \;\geq\; \left(1-\frac{1}{e}-\epsilon\right)\mathsf{OPT}, \tag{8}$$

where the last two inequalities follow from Lemmas D.3 and D.2. Thus, the desired performance guarantee follows. We turn to obtaining the vector $\widehat{\boldsymbol{v}}$ such that $H(\widehat{\boldsymbol{v}})\geq H(\overline{\boldsymbol{v}})$.

Using the vector $\boldsymbol{v}^a = (v_i^a : i\in\mathcal{F})\in[0,1]^n$, noting that $f(\boldsymbol{z}) = \sum_{a\in\mathcal{A}} f_{\mathsf{LP}}^a(\boldsymbol{z}^a)$ by the definition of f, we have $H(\boldsymbol{v}) = \sum_{a\in\mathcal{A}}\mathbb{E}\{f_{\mathsf{LP}}^a(\lfloor\overline{\boldsymbol{z}}^a\rfloor + \mathsf{R}(\boldsymbol{v}^a))\}$. For each $a\in\mathcal{A}$, define the function $H^a : [0,1]^n\to\mathbb{R}_+$ as $H^a(\boldsymbol{v}^a) = \mathbb{E}\{f_{\mathsf{LP}}^a(\lfloor\overline{\boldsymbol{z}}^a\rfloor + \mathsf{R}(\boldsymbol{v}^a))\}$, so we have $H(\boldsymbol{v}) = \sum_{a\in\mathcal{A}} H^a(\boldsymbol{v}^a)$. Because f_{LP}^a is monotone and

DR-submodular, H^a is the multilinear extension of a monotone and submodular set function on the ground set $\{0,1\}^n$. We exploit this property of H^a in pipage rounding.

By the discussion just after Lemma D.2, we have $\overline{\boldsymbol{z}} \in \mathcal{P}$ and $\sum_{i\in\mathcal{F}} \overline{z}_i^a = C^a$ for all $a \in \mathcal{A}$. If $\overline{z}_i^a$ is an integer for all $a \in \mathcal{A}$ and $i \in \mathcal{F}$, then we have $\overline{v}_i^a = \overline{z}_i^a - \lfloor \overline{z}_i^a \rfloor = 0$ for all $i \in \mathcal{F}$ and $a \in \mathcal{A}$, in which case, we can choose $\widehat{\boldsymbol{v}} = \overline{\boldsymbol{v}} = \overline{\boldsymbol{0}}$, so by (8), $\overline{\boldsymbol{z}}$ provides the desired performance guarantee. Thus, we assume that $\overline{\boldsymbol{z}}$ is not integral. Let $C(\overline{\boldsymbol{z}}) = \{\boldsymbol{x} \in \mathbb{R}_+^{n\times m} : \lfloor \overline{z}_i^a \rfloor \le x_i^a \le \lfloor \overline{z}_i^a \rfloor + 1 \;\; \forall a \in \mathcal{A},\ i \in \mathcal{F}\}$. Define $\mathcal{P}' = \{\boldsymbol{x} - \lfloor \boldsymbol{x} \rfloor : \boldsymbol{x} \in \mathcal{P} \cap C(\overline{\boldsymbol{z}})\}$. It is easy to verify that $\mathcal{P}'$ is obtained by translating $\mathcal{P} \cap C(\overline{\boldsymbol{z}})$ by $-\lfloor \overline{\boldsymbol{z}} \rfloor$ and restricting to $[0,1]^{n\times m}$. In other words, we have

$$\mathcal{P}' = \left\{ \boldsymbol{x} \in [0,1]^{n\times m} \;:\; \sum_{i\in\mathcal{F}} x_i^a \;\le\; C^a - \sum_{i\in\mathcal{F}} \lfloor \overline{z}_i^a \rfloor \quad \forall a \in \mathcal{A}, \quad \sum_{a\in\mathcal{A}} x_i^a \;\le\; U_i - \sum_{a\in\mathcal{A}} \lfloor \overline{z}_i^a \rfloor \quad \forall i \in \mathcal{F} \right\}.$$

Because $\overline{\boldsymbol{z}} \in \mathcal{P}$ and $\overline{\boldsymbol{v}} = \overline{\boldsymbol{z}} - \lfloor \overline{\boldsymbol{z}} \rfloor$, we have $\overline{\boldsymbol{v}} \in \mathcal{P}'$, so $\sum_{i\in\mathcal{F}} \overline{v}_i^a = C^a - \sum_{i\in\mathcal{F}} \lfloor \overline{z}_i^a \rfloor$ for all $a \in \mathcal{A}$.

Let $\mathcal{G}_{\overline{\boldsymbol{v}}}$ be a bipartite graph with the sets of nodes on the two sides of the graph corresponding to the fulfillment centers and products. We have an edge (i,a) if and only $\overline{v}_i^a$ is fractional; that is, there is an edge (i,a) if and only if $0 < \overline{v}_i^a < 1$. Because $\overline{\boldsymbol{z}}$ is not integral, $\overline{v}_i^a$ is fractional for some $i \in \mathcal{F}$ and $a \in \mathcal{A}$. Thus, there is a path or a cycle in $\mathcal{G}_{\overline{\boldsymbol{v}}}$. We denote the set of nodes in this path or cycle by Υ. If Υ corresponds to a cycle, then every node in Υ has degree two. If Υ corresponds to a path, then we assume that Υ is a maximal path in the sense that it is not a subset of any other path. Furthermore, noting that $\sum_{i\in\mathcal{F}} \overline{v}_i^a = C^a - \sum_{i\in\mathcal{F}} \lfloor \overline{z}_i^a \rfloor$ and $C^a - \sum_{i\in\mathcal{F}} \lfloor \overline{z}_i^a \rfloor$ is an integer, it must be the case that every node in $\Upsilon \cap \mathcal{A}$ must have degree two. In this case, if Υ corresponds to a path, then the end nodes of the path must be in $\mathcal{F}$.

Because $\mathcal{G}_{\overline{\boldsymbol{v}}}$ is bipartite, we can represent Υ uniquely by a union of two matchings, which we denote by M_1 and M_2. Following the pipage rounding idea introduced by Ageev and Sviridenko (2004), for small $\varepsilon > 0$, we construct the vector $\boldsymbol{v}(\varepsilon, \Upsilon) \in [0,1]^{n\times m}$ as

$$v_i^a(\varepsilon, \Upsilon) = \begin{cases} \overline{v}_i^a & \text{if } (i,a) \notin M_1 \cup M_2 \\ \overline{v}_i^a + \varepsilon & \text{if } (i,a) \in M_1 \\ \overline{v}_i^a - \varepsilon & \text{if } (i,a) \in M_2. \end{cases}$$

Setting $\varepsilon_1 = \min_{(i,a)\in M_1} \overline{v}_i^a \wedge \min_{(i,a)\in M_2} (1 - \overline{v}_i^a)$ and $\varepsilon_2 = \min_{(i,a)\in M_1} (1 - \overline{v}_i^a) \wedge \min_{(i,a)\in M_2} \overline{v}_i^a$, we set $\boldsymbol{v}_1 = \boldsymbol{v}(-\varepsilon_1, \Upsilon)$ and $\boldsymbol{v}_2 = \boldsymbol{v}(\varepsilon_2, \Upsilon)$. Using the fact that Υ either corresponds to a cycle or a maximal path, we can check that both $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ are in $\mathcal{P}'$. By our construction of the vector $\boldsymbol{v}(\varepsilon, \Upsilon)$, as well as the choice of ε_1 and ε_2, the number of fractional coordinates in each of $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ is strictly less than that in $\overline{\boldsymbol{v}}$. If $H(\boldsymbol{v}_1) > H(\boldsymbol{v}_2)$, then we output $\boldsymbol{v}_1$. Otherwise, we output $\boldsymbol{v}_2$.

We show that $\max\{H(\boldsymbol{v}_1), H(\boldsymbol{v}_2)\} \ge H(\overline{\boldsymbol{v}})$. For each product $a \in \Upsilon \cap \mathcal{A}$, let $(q_1(a), a)$ and $(q_2(a), a)$ denote the edges in the matchings M_1 and M_2, respectively. Comparing the vectors $\overline{\boldsymbol{v}}$ and $\boldsymbol{v}(\varepsilon, \Upsilon)$, pipage rounding changes the values of $\overline{v}^a_{q_1(a)}$ and $\overline{v}^a_{q_2(a)}$ in the vector $\overline{\boldsymbol{v}}$ to $\overline{v}^a_{q_1(a)}+\varepsilon$ and $\overline{v}^a_{q_2(a)}-\varepsilon$ in the vector $\boldsymbol{v}(\varepsilon,\Upsilon)$. Under the vector $\boldsymbol{v}(\varepsilon,\Upsilon)$, we have

$$H(\boldsymbol{v}(\varepsilon,\Upsilon))=\sum_{a\in\mathcal{A}} H^a(\overline{\boldsymbol{v}}+\varepsilon(\mathbb{1}_{\{q_1(a)\}}-\mathbb{1}_{\{q_2(a)\}})).$$

Since H^a is the multilinear extension of a monotone and submodular set function, by Lemma D.5(e), the mapping $\varepsilon\to H^a(\overline{\boldsymbol{v}}+\varepsilon(\mathbb{1}_{\{q_1(a)\}}-\mathbb{1}_{\{q_2(a)\}}))$ is convex in ε. Thus, the mapping $\varepsilon\to H(\boldsymbol{v}(\varepsilon,\Upsilon))$ is convex in ε as well. In this case, the value of $H(\boldsymbol{v}(\varepsilon,\Upsilon))$ at $\varepsilon=-\varepsilon_1$ or $\varepsilon=\varepsilon_2$ is at least as large as the value at $\varepsilon=0$. Therefore, we have $\max\{H(\boldsymbol{v}_1),H(\boldsymbol{v}_2)\}\geq H(\overline{\boldsymbol{v}})$.

Starting from the vector $\overline{\boldsymbol{v}}$, each iteration of pipage rounding yields a new vector $\boldsymbol{v}_0\in\mathcal{P}'$, while ensuring that $H(\boldsymbol{v}_0)\geq H(\overline{\boldsymbol{v}})$. Furthermore, the number of fractional coordinates of the vector $\boldsymbol{v}_0$ is strictly less than the number of fractional coordinates of the vector $\overline{\boldsymbol{v}}$. Thus, after at most nm iterations, we obtain a binary vector $\widehat{\boldsymbol{v}}$ such that $\lfloor\overline{\boldsymbol{z}}\rfloor+\widehat{\boldsymbol{v}}\in\mathcal{P}$ and $H(\widehat{\boldsymbol{v}})\geq H(\overline{\boldsymbol{v}})$. ■

D.2 Properties of Monotone and Submodular Set Functions

We give useful properties of monotone and submodular set functions. We have a ground set X and let $q:2^X\to\mathbb{R}_+$ be a monotone and submodular set function so that it satisfies $q(B)\geq q(A)$ for all $A\subseteq B\subseteq X$ and $q(B\cup\{i\})-q(B)\leq q(A\cup\{i\})-q(A)$ for all $A\subseteq B\subseteq X$ and $i\in X\setminus B$. The following property follows immediately from the definition of submodularity.

Lemma D.4 (Subadditivity) *For each $A,B\subseteq X$, $q(A\cup B)-q(A)\leq\sum_{i\in B}q(A\cup\{i\})-q(A)$.*

Proof: Without loss of generality, we assume that $A\cap B=\varnothing$. Assuming $B=\{i_1,\ldots,i_k\}$, we have

$$q(A\cup B)-q(A)\;=\;\sum_{h=1}^{k} q(A\cup\{i_1,\ldots,i_h\})-q(A\cup\{i_1,\ldots,i_{h-1}\})\;\leq\;\sum_{h=1}^{k} q(A\cup\{i_h\})-q(A)\;,$$

where the inequality follows from the submodularity of q. ■

Because there is a one-to-one correspondence between each subset of X and a binary vector in $\{0,1\}^{|X|}$, we can view $q:\{0,1\}^{|X|}\to\mathbb{R}_+$ as a function whose domain is $\{0,1\}^{|X|}$. Let $Q:[0,1]^X\to\mathbb{R}_+$ be the multilinear extension of q, so for each $\boldsymbol{y}\in[0,1]^{|X|}$, we have

$$Q(\boldsymbol{y})\;=\;\mathbb{E}\{q(\mathsf{R}(\boldsymbol{y}))\}\;=\;\sum_{S\subseteq X} q(S)\prod_{i\in S} y_i\prod_{i\notin S}(1-y_i),$$

where $\mathsf{R}(\boldsymbol{y})$ denotes a random subset of X such that, for each $i\in X$, we have $i\in\mathsf{R}(\boldsymbol{y})$ with probability y_i and $i\notin\mathsf{R}(\boldsymbol{y})$ with probability $1-y_i$. Note that $Q(\boldsymbol{y})=q(\boldsymbol{y})$ for all $\boldsymbol{y}\in\{0,1\}^{|X|}$.

In the next lemma, we give important properties of the multilinear extension. These results are standard in the literature and follow from the definition of the multilinear extension; see, for example, Vondrak (2017). In this lemma, we use $\partial_i Q(\boldsymbol{y})$ to denote the partial derivative of Q with respect to y_i and $\partial_{i,j} Q(\boldsymbol{y})$ to denote the second-order partial derivative of Q with respect to y_i and y_j. For each $A \subseteq X$, $\mathbb{1}_A \in \{0,1\}^{|X|}$ is a binary vector with one in the components corresponding to the elements in A and zero everywhere else. Lastly, for the vectors $\boldsymbol{x} = (x_i : i \in X) \in \mathbb{R}^{|X|}$ and $\boldsymbol{y} = (y_i : i \in X) \in \mathbb{R}^{|X|}$, we write $\boldsymbol{x} \vee \boldsymbol{y} = (x_i \vee y_i : i \in X)$ and $\boldsymbol{x} \wedge \boldsymbol{y} = (x_i \wedge y_i : i \in X)$.

Lemma D.5 (Multilinear Extension) *The multilinear extension Q of a monotone and submodular set function q satisfies the following properties.*

(a) *For all $i \in X$ and $\boldsymbol{y} \in [0,1]^X$, we have $\partial_i Q(\boldsymbol{y}) \;=\; Q(\boldsymbol{y} \vee \mathbb{1}_{\{i\}}) - Q(\boldsymbol{y} \wedge \mathbb{1}_{X\setminus\{i\}})$.*

(b) *For all $\boldsymbol{y}, \boldsymbol{y}' \in [0,1]^{|X|}$, if $\boldsymbol{y}$ and $\boldsymbol{y}'$ differ only in the i-th coordinate, then we have*

$$Q(\boldsymbol{y}') - Q(\boldsymbol{y}) \;=\; (y_i' - y_i)\, \partial_i Q(\boldsymbol{y}).$$

(c) *For all $i, j \in X$ and $\boldsymbol{y} \in [0,1]^{|X|}$, we have $\partial_i Q(\boldsymbol{y}) \geq 0$ and $\partial_{i,j} Q(\boldsymbol{y}) \leq 0$.*

(d) *For all $\boldsymbol{d} \in \mathbb{R}_+^{|X|}$ and $\boldsymbol{y}_0 \in [0,1]^{|X|}$, the mapping $t \to Q(\boldsymbol{y}_0 + t\,\boldsymbol{d})$ is increasing and concave in t.*

(e) *For all $i, j \in X$ and $\boldsymbol{y}_0 \in [0,1]^{|X|}$, the mapping $t \to Q(\boldsymbol{y}_0 + t\,(\mathbb{1}_{\{i\}} - \mathbb{1}_{\{j\}}))$ is convex in t.*

Proof: Differentiating the expression for the multilinear extension, Part (a) follows as

$$\partial_i Q(\boldsymbol{y}) = \sum_{S \subseteq X : i \in S} q(S) \prod_{\ell \in S \setminus \{i\}} y_\ell \prod_{\ell \notin S} (1 - y_\ell) - \sum_{S \subseteq X : i \notin S} q(S) \prod_{\ell \in S} y_i \prod_{\ell \notin S \cup \{i\}} (1 - y_i) \;=\; Q(\boldsymbol{y} \vee \mathbb{1}_{\{i\}}) - Q(\boldsymbol{y} \wedge \mathbb{1}_{X\setminus\{i\}}).$$

To see that Part (b) holds, by the definition of the multilinear extension, we have

$$\begin{aligned} Q(\boldsymbol{y}') - Q(\boldsymbol{y}) \;&=\; (y_i' - y_i) \sum_{S \subseteq X : i \in S} q(S) \prod_{\ell \in S \setminus \{i\}} y_\ell \prod_{\ell \notin S} (1 - y_\ell) \;-\; (y_i' - y_i) \sum_{S \subseteq X : i \notin S} q(S) \prod_{\ell \in S} y_\ell \prod_{\ell \notin S \cup \{i\}} (1 - y_\ell) \\ &=\; (y_i' - y_i) \left\{ \sum_{S \subseteq X : i \in S} q(S) \prod_{\ell \in S \setminus \{i\}} y_\ell \prod_{\ell \notin S} (1 - y_\ell) \;-\; \sum_{S \subseteq X : i \notin S} q(S) \prod_{\ell \in S} y_\ell \prod_{\ell \notin S \cup \{i\}} (1 - y_\ell) \right\} \\ &=\; (y_i' - y_i)\, \partial_i Q(\boldsymbol{y}), \end{aligned}$$

where the last equality uses Part (a). The first inequality in Part (c) follows from Part (a) because we have $\partial_i Q(\boldsymbol{y}) = Q(\boldsymbol{y} \vee \mathbb{1}_{\{i\}}) - Q(\boldsymbol{y} \wedge \mathbb{1}_{X\setminus\{i\}}) = \mathbb{E}\{q(\mathsf{R}(\boldsymbol{y}) \cup \{i\}) - q(\mathsf{R}(\boldsymbol{y}) \setminus \{i\})\} \geq 0$, where the last inequality uses the monotonicity of q. For the second inequality in Part (c), for $i \neq j$, we have

$$\begin{aligned} &\partial_{i,j} Q(\boldsymbol{y}) = \partial_j Q(\boldsymbol{y} \vee \mathbb{1}_{\{i\}}) - \partial_j Q(\boldsymbol{y} \wedge \mathbb{1}_{X\setminus\{i\}}) \\ &= \mathbb{E}\Big\{ q(\mathsf{R}(\boldsymbol{y}) \cup \{i\} \cup \{j\}\,) - q(\,\mathsf{R}(\boldsymbol{y}) \cup \{i\} \setminus \{j\}) \Big\} - \mathbb{E}\Big\{ q(\mathsf{R}(\boldsymbol{y}) \setminus \{i\} \cup \{j\}\,) - q(\mathsf{R}(\boldsymbol{y}) \setminus \{i\} \setminus \{j\}) \Big\} \leq 0, \end{aligned}$$

where the last inequality follows by using the fact that the submodularity of q implies the inequality $q(\mathsf{R}(\boldsymbol{y}) \cup \{i\} \cup \{j\}\,) - q(\,\mathsf{R}(\boldsymbol{y}) \cup \{i\} \setminus \{j\}) \leq q(\mathsf{R}(\boldsymbol{y}) \setminus \{i\} \cup \{j\}\,) - q(\mathsf{R}(\boldsymbol{y}) \setminus \{i\} \setminus \{j\}) \leq 0$. For $i = j$, because $Q(\boldsymbol{y})$ is linear in y_i, we have $\partial_{i,i}Q(\boldsymbol{y})=0$. To see that Part (d) holds, let $\phi(t)=Q(\boldsymbol{y}_0+t\boldsymbol{d})$ for $\boldsymbol{d}\in\mathbb{R}_+^{|X|}$. Using the chain rule and Part (c), we get $\phi'(t)=\sum_{i\in X} d_i\,\partial_i Q(\boldsymbol{y}_0+t\,\boldsymbol{d}) \geq 0$. Furthermore, we have $\phi''(t)=\sum_{i,j\in X} d_i\,d_j\,\partial_{i,j}Q(\boldsymbol{y}_0+t\,\boldsymbol{d})\leq 0$, where the inequality also follows from Part (c). Lastly, to show Part (e), consider $\psi(t)=Q(\boldsymbol{y}_0+t(\mathbb{1}_{\{i\}}-\mathbb{1}_{\{j\}}))$. Because $\partial_{i,i}Q(\boldsymbol{y})=0$, we have

$$\begin{aligned}\psi''(t) &= \partial_{i,i}Q(\boldsymbol{y}_0+t\,(\mathbb{1}_{\{i\}}-\mathbb{1}_{\{j\}}))-2\,\partial_{i,j}Q(\boldsymbol{y}_0+t\,(\mathbb{1}_{\{i\}}-\mathbb{1}_{\{j\}}))+\partial_{j,j}Q(\boldsymbol{y}_0+t\,(\mathbb{1}_{\{i\}}-\mathbb{1}_{\{j\}}))\\ &= -2\,\partial_{i,j}Q(\boldsymbol{y}_0+t\,(\mathbb{1}_{\{i\}}-\mathbb{1}_{\{j\}})) \geq 0,\end{aligned}$$

where the last inequality uses Part (c). ■

D.3 Proof of Lemma D.2: Maximizing the Multilinear Extension

We give a proof for Lemma D.2, which involves using the continuous greedy algorithm to maximize the multilinear extension of a monotone and submodular set function over a downward closed polytope. We give the proof within the context of a general monotone and submodular set function. We have a ground set X. Let $q:2^X\to\mathbb{R}_+$ be a monotone and submodular set function that is normalized so that $q(\varnothing)=0$. We use $Q:[0,1]^{|X|}\to\mathbb{R}_+$ to denote the multilinear extension of q, so for all $\boldsymbol{y}\in[0,1]^{|X|}$, we have $Q(\boldsymbol{y})=\mathbb{E}\{q(\mathsf{R}(\boldsymbol{y}))\}=\sum_{S\subseteq X} q(S)\prod_{i\in S} y_i\prod_{i\notin S}(1-y_i)$, where $\mathsf{R}(\boldsymbol{y})$ is a random subset of X such that each $i\in X$ is independently included in $\mathsf{R}(\boldsymbol{y})$ with probability y_i. Let $\mathcal{Q}\subseteq[0,1]^{|X|}$ be a downward closed polytope; that is, if $\boldsymbol{y}\in\mathcal{Q}$, then $\boldsymbol{y}'\in\mathcal{Q}$ for all $\boldsymbol{y}'\in[0,1]^{|X|}$ such that $\boldsymbol{y}'\leq\boldsymbol{y}$. We are interested in solving the problem

$$Z^* = \max_{\boldsymbol{y}\in\mathcal{Q}\cap\{0,1\}^{|X|}} q(\boldsymbol{y}). \tag{9}$$

To get an approximate solution to problem (9) through the multilinear extension of q, we consider the following continuous greedy algorithm. Below, we fix $\epsilon\in(0,1)$ such that $1/\epsilon$ is an integer.

<u>Continuous Greedy:</u>

Initialization. Set $\delta=\frac{\epsilon}{2\,|X|^3}$, $t=0$ and $\boldsymbol{y}_0=\underline{\boldsymbol{0}}$, where $\underline{\boldsymbol{0}}\in\mathbb{R}_+^{|X|}$ is the vector of all zeros.

Algorithm. While $t<1$, do the following steps.

Step 1. For each $i\in X$, let $w_{it}=Q(\boldsymbol{y}_t\vee\mathbb{1}_{\{i\}})-Q(\boldsymbol{y}_t)$. Define the vector $\boldsymbol{w}_t=(w_{it}:i\in X)$.

Step 2. Solve the linear program $\max_{\boldsymbol{d}\in[0,1]^{|X|}}\{\boldsymbol{w}_t^\top\boldsymbol{d}:\boldsymbol{d}\in\mathcal{Q}\}$ and let $\boldsymbol{d}_t$ denote an optimal solution of this linear program.

Step 3. Set $\boldsymbol{y}_{t+\delta}=\boldsymbol{y}_t+\delta\,\boldsymbol{d}_t$.

Step 4. Increment t by δ.

Output. Return the value of $\boldsymbol{y}_1$.

Note that because of our choice of ϵ, $1/\delta$ is an integer. The algorithm above and our subsequent analysis are virtually the same as those in Feldman (2013). Without loss of generality, we assume that $\mathbb{1}_{\{i\}} \in \mathcal{Q} \cap \{0,1\}^{|X|}$ for all $i \in X$. Noting that $\mathcal{Q}$ is downward closed, if $\mathbb{1}_{\{i\}} \notin \mathcal{Q} \cap \{0,1\}^{|X|}$ for some $i \in X$, then the optimal solution to problem (9) will never contain element i, so we can remove element i from the ground set. The main result of this section is stated in the following theorem. Lemma D.2 follows from this result.

Theorem D.6 (Continuous Greedy in General Form) *The continuous greedy algorithm terminates with the output $\boldsymbol{y}_1 \in \mathcal{Q}$ that satisfies $Q(\boldsymbol{y}_1) \geq \left(1 - \frac{1}{e} - \epsilon\right) Z^*$ in running time that is polynomial in $|X|$, $1/\epsilon$ and the description length of the polytope $\mathcal{Q}$ in bits.*

The theorem above is given as Theorem 3.1.1 in Feldman (2013). For completeness, we provide a proof. The proof uses three auxiliary lemmas. In the next lemma, we give an upper bound on the difference of the partial derivatives of Q at two points that are close to each other.

Lemma D.7 (Difference in Partial Derivatives) *For each $\boldsymbol{y}, \boldsymbol{y}' \in [0,1]^{|X|}$, if $0 \leq y'_i - y_i \leq \gamma$ for all $i \in X$, then we have $|\partial_i Q(\boldsymbol{y}') - \partial_i Q(\boldsymbol{y})| \leq 2\,\gamma |X|^2\, Z^*$ for all $i \in X$.*

Proof: Recall that $\mathsf{R}(\boldsymbol{y})$ is a random set, where for each $i \in X$, $i \in \mathsf{R}(\boldsymbol{y})$ with probability y_i and $i \notin \mathsf{R}(\boldsymbol{y})$ with probability $1 - y_i$, so $Q(\boldsymbol{y}) = \mathbb{E}\{q(\mathsf{R}(\boldsymbol{y}))\}$. Define the random set D as follows. Each $i \in X \setminus \mathsf{R}(\boldsymbol{y})$ is included in D with probability $1 - \frac{1-y'_i}{1-y_i}$. Note that $0 \leq \frac{1-y'_i}{1-y_i} \leq 1$. Noting that $\mathsf{R}(\boldsymbol{y}) \cap D = \varnothing$, each element $i \in X$ is included in the random set $\mathsf{R}(\boldsymbol{y}) \cup D$ with probability $y_i + (1-y_i)\left(1 - \frac{1-y'_i}{1-y_i}\right) = y_i + (y'_i - y_i) = y'_i$. Therefore, $\mathsf{R}(\boldsymbol{y}) \cup D$ has the same distribution as $\mathsf{R}(\boldsymbol{y}')$, which implies that $Q(\boldsymbol{y}') = \mathbb{E}\big\{q(\,\mathsf{R}(\boldsymbol{y}) \cup D\,)\big\}$. In this case, by Lemma D.5(a), we get

$$\begin{aligned}
\partial_i Q(\boldsymbol{y}') \;&=\; Q(\boldsymbol{y}' \vee \mathbb{1}_{\{i\}}) - Q(\boldsymbol{y}' \wedge \mathbb{1}_{X\setminus\{i\}}) \;\overset{(a)}{=}\; \mathbb{E}\Big\{q(\mathsf{R}(\boldsymbol{y}') \cup \{i\}) \;-\; q(\mathsf{R}(\boldsymbol{y}') \setminus \{i\})\Big\} \\
&\overset{(b)}{=}\; \mathbb{E}\Big\{q(\mathsf{R}(\boldsymbol{y}) \cup D \cup \{i\}) \;-\; q(\mathsf{R}(\boldsymbol{y}) \cup D \setminus \{i\})\Big\} \\
&=\; \mathbb{P}\{D = \varnothing\}\, \mathbb{E}\Big\{q(\mathsf{R}(\boldsymbol{y}) \cup \{i\}) \;-\; q(\mathsf{R}(\boldsymbol{y}) \setminus \{i\}) \;\Big|\; D = \varnothing\Big\} \\
&\qquad + \mathbb{P}\{D \neq \varnothing\}\, \mathbb{E}\Big\{q(\mathsf{R}(\boldsymbol{y}) \cup D \cup \{i\}) \;-\; q(\mathsf{R}(\boldsymbol{y}) \cup D \setminus \{i\}) \;\Big|\; D \neq \varnothing\Big\}, \quad (10)
\end{aligned}$$

where (a) follows because the vector $\boldsymbol{y}' \vee \mathbb{1}_{\{i\}}$ has a one corresponding to element $i \in X$, so element i is always included in $\boldsymbol{y}' \vee \mathbb{1}_{\{i\}}$, whereas $\boldsymbol{y}' \wedge \mathbb{1}_{X\setminus\{i\}}$ has a zero corresponding to element $i \in X$, so element i is always excluded in $\boldsymbol{y}' \wedge \mathbb{1}_{X\setminus\{i\}}$ and (b) follows because $\mathsf{R}(\boldsymbol{y}) \cup D$ has the same distribution as $\mathsf{R}(\boldsymbol{y}')$. Using Lemma D.5(a) similarly once more, we also have

$$\partial_i Q(\boldsymbol{y}) \;=\; Q(\boldsymbol{y} \vee \mathbb{1}_{\{i\}}) - Q(\boldsymbol{y} \wedge \mathbb{1}_{X\setminus\{i\}}) \;=\; \mathbb{E}\Big\{q(\mathsf{R}(\boldsymbol{y}) \cup \{i\}) \;-\; q(\mathsf{R}(\boldsymbol{y}) \setminus \{i\})\Big\}. \qquad (11)$$

Because we have $q(\varnothing) = 0$, by Lemma D.4, we get $0 \leq q(S) \leq |S| \max_{i \in X} q(\{i\}) \leq |X| Z^*$. By our construction, for each $i \in X$, we have $\mathbb{P}\{i \in D\} = (1 - y_i)\left(1 - \frac{1-y'_i}{1-y_i}\right) = y'_i - y_i$. Furthermore, we include each $i \in X$ in D independently. Thus, using the fact that $0 \le y'_i - y_i \le \gamma$, we obtain the chain of inequalities

$$\mathbb{P}\{D \ne \varnothing\} = 1 - \mathbb{P}\{D = \varnothing\} = 1 - \prod_{i \in X}(1 - (y'_i - y_i)) \le 1 - \prod_{i \in X}(1 - \gamma) \le 1 - (1 - \gamma|X|) = \gamma|X|,$$

where the last inequality follows because $(1-\gamma)^{|X|} \ge 1 - \gamma|X|$. Writing the expectation in (11) by conditioning on the events $D = \varnothing$ and $D \ne \varnothing$, subtracting (11) from (10), we obtain

$$\begin{aligned} |\partial_i Q(\boldsymbol{y}') - \partial_i Q(\boldsymbol{y})| \; &\le \; \mathbb{P}\{D \ne \varnothing\} \left| \mathbb{E}\Big\{ q(\mathsf{R}(\boldsymbol{y}) \cup D \cup \{i\}) \; - \; q(\mathsf{R}(\boldsymbol{y}) \cup \{i\}) \;\Big|\; D \ne \varnothing \Big\} \right| \\ &\quad + \mathbb{P}\{D \ne \varnothing\} \left| \mathbb{E}\Big\{ q(\mathsf{R}(\boldsymbol{y}) \cup D \setminus \{i\}) - q(\mathsf{R}(\boldsymbol{y}) \setminus \{i\}) \;\Big|\; D \ne \varnothing \Big\} \right| \; \le \; 2\gamma |X|^2 Z^*, \end{aligned}$$

where the last inequality uses the fact that $\mathbb{P}\{D \ne \varnothing\} \le \gamma|X|$ and $q(S) \le |X|\, Z^*$ for any $S \subseteq X$. ■

In the next lemma, we give a lower bound on the difference of the values of Q at two points that are close to each other.

Lemma D.8 (Difference in Values) *For each $\boldsymbol{y}, \boldsymbol{y}' \in [0,1]^{|X|}$, if $0 \le y'_i - y_i \le \gamma$ for all $i \in X$, then we have $Q(\boldsymbol{y}') - Q(\boldsymbol{y}) \ge \sum_{i \in X}(y'_i - y_i)\, \partial_i Q(\boldsymbol{y}) \; - \; 2\gamma^2 |X|^3 Z^*$.*

Proof: Letting $n = |X|$, we denote the elements of X by $\{1, \ldots, n\}$ in some arbitrary order. For each $k \in \{0, 1, \ldots, n\}$, let $\boldsymbol{x}^{(k)} \in [0,1]^{|X|}$ be such that $\boldsymbol{x}^{(k)}$ agrees with $\boldsymbol{y}'$ on the first k coordinates and agrees with $\boldsymbol{y}$ on the remaining coordinates, so we have

$$\boldsymbol{x}_i^{(k)} = \begin{cases} y'_i & \text{if } i \in \{1, 2, \ldots, k\} \\ y_i & \text{if } i \in \{k+1, k+2, \ldots, n\}. \end{cases}$$

Note that $\boldsymbol{x}^{(0)} = \boldsymbol{y}$ and $\boldsymbol{x}^{(n)} = \boldsymbol{y}'$. By a telescoping sum, we have

$$\begin{aligned} Q(\boldsymbol{y}') - Q(\boldsymbol{y}) \; &= \; \sum_{k=1}^{n} Q(\boldsymbol{x}^{(k)}) - Q(\boldsymbol{x}^{(k-1)}) \; \overset{(a)}{=} \; \sum_{k=1}^{n} (y'_k - y_k)\, \partial_k Q(\boldsymbol{x}^{(k-1)}) \\ &= \; \sum_{k=1}^{n} (y'_k - y_k)\, \partial_k Q(\boldsymbol{y}) + \sum_{k=1}^{n} (y'_k - y_k)\, (\partial_k Q(\boldsymbol{x}^{(k-1)}) - \partial_k Q(\boldsymbol{y})) \\ &\ge \; \sum_{k=1}^{n} (y'_k - y_k)\, \partial_k Q(\boldsymbol{y}) - \sum_{k=1}^{n} |y'_k - y_k|\, |\partial_k Q(\boldsymbol{x}^{(k-1)}) - \partial_k Q(\boldsymbol{y})| \\ &\overset{(b)}{\ge} \; \sum_{k=1}^{n} (y'_k - y_k)\, \partial_k Q(\boldsymbol{y}) \; - \; 2\gamma^2 |X|^3 Z^*, \end{aligned}$$

where (a) follows from Lemma D.5(b) along with the fact that $\boldsymbol{x}^{(k)}$ and $\boldsymbol{x}^{(k-1)}$ differ from each other only in the k-th coordinate and (b) follows from Lemma D.7. ■

In the next lemma, we give an upper bound on a sequence defined by a recursive relationship.

Lemma D.9 (Sequence Bound) *For $0 < \alpha < 1$ and $\beta \geq 0$, if the sequence of real numbers $\{x_k : k \geq 0\}$ satisfies $x_{k+1} \leq \alpha x_k + \beta$ for all $k \geq 0$, then $x_k \leq \alpha^k x_0 + \beta \sum_{h=0}^{k-1} \alpha^h$ for all $k \geq 0$.*

Proof: We show the result by using induction. The result is clearly true for $k = 0$ and $k = 1$. Assume that the result holds for x_k. By the hypothesis of the lemma and induction assumption, we have

$$x_{k+1} \leq \alpha x_k + \beta \leq \alpha\Big(\alpha^k x_0 + \beta \sum_{h=0}^{k-1} \alpha^h\Big) + \beta \leq \alpha^{k+1} x_0 \; + \; \beta \sum_{h=0}^{k} \alpha^h,$$

which completes the induction. ■

Here is the proof of Theorem D.6.

<u>**Proof of Theorem D.6**</u>:

By Step 3 of the continuous greedy algorithm, we have $\boldsymbol{y}_1 = \sum_{k=0}^{\frac{1}{\delta}-1} \delta\, \boldsymbol{d}_{k\delta}$, which is a convex combination of $\{\boldsymbol{d}_{k\delta} : k = 0, \ldots, \frac{1}{\delta} - 1\}$. Thus, because $\mathcal{Q}$ is a polytope and $\boldsymbol{d}_{k\delta} \in \mathcal{Q}$ for all $k = 0, \ldots, \frac{1}{\delta} - 1$, we have $\boldsymbol{y}_1 \in \mathcal{Q}$. The algorithm terminates in $1/\delta$ iterations, each iteration solving a linear program, so the running time is polynomial. We lower bound the improvement of the objective value in each iteration. For all $t \in \big\{k\delta : k = 0, 1, \ldots, \frac{1}{\delta} - 1\big\}$, we claim that

$$Q(\boldsymbol{y}_{t+\delta}) - Q(\boldsymbol{y}_t) \geq \delta\,(Z^* - Q(\boldsymbol{y}_t)) - 2\,\delta^2 |X|^3 Z^*. \tag{12}$$

To establish the claim, consider an arbitrary $t \in \big\{k\delta : k = 0, 1, \ldots, \frac{1}{\delta} - 1\big\}$ and let R_t be a random subset of X where each element $i \in X$ is included with probability y_{it}, so $Q(\boldsymbol{y}_t) = \mathbb{E}\{q(R_t)\}$. Letting $\boldsymbol{y}^*$ be an optimal solution to the problem $Z^* \;=\; \max_{\boldsymbol{y} \in \mathcal{Q} \cap \{0,1\}^{|X|}} \; q(\boldsymbol{y})$, we use $S^* = \{i \in X : y_i^* = 1\}$ to denote the elements in the optimal solution. By Step 1 of the continuous greedy algorithm, we have $w_{it} = Q(\boldsymbol{y}_t \vee \mathbb{1}_{\{i\}}) - Q(\boldsymbol{y}_t)$ for all $i \in X$, so we obtain

$$\begin{aligned}
\boldsymbol{w}_t^\top \boldsymbol{y}^* \;&=\; \sum_{i \in S^*} w_{it} = \sum_{i \in S^*} \Big(Q(\boldsymbol{y}_t \vee \mathbb{1}_{\{i\}}) - Q(\boldsymbol{y}_t)\Big) = \sum_{i \in S^*} \mathbb{E}\Big\{q(R_t \cup \{i\}) - q(R_t)\Big\} \\
&\overset{(a)}{\geq} \; \mathbb{E}\Big\{q(R_t \cup S^*) - q(R_t)\Big\} \;=\; Q(\boldsymbol{y}_t \vee \mathbb{1}_{S^*}) - Q(\boldsymbol{y}_t) \;=\; Q(\boldsymbol{y}_t \vee \boldsymbol{y}^*) - Q(\boldsymbol{y}_t),
\end{aligned}$$

where (a) follows from Lemma D.4. Because $\boldsymbol{y}^* \in \mathcal{Q}$, by the definition of $\boldsymbol{d}_t$ in Step 2 of the continuous greedy algorithm, we get $\boldsymbol{w}_t^\top \boldsymbol{d}_t \geq \boldsymbol{w}_t^\top \boldsymbol{y}^* \geq Q(\boldsymbol{y}_t \vee \boldsymbol{y}^*) - Q(\boldsymbol{y}_t)$. Thus, we have

$$\begin{aligned}
\sum_{i \in X} d_{it}\, \partial_i Q(\boldsymbol{y}_t) \;&\overset{(b)}{=}\; \sum_{i \in X} d_{it} \Big[Q(\boldsymbol{y}_t \vee \mathbb{1}_{\{i\}}) - Q(\boldsymbol{y}_t \wedge \mathbb{1}_{X \setminus \{i\}})\Big] \;\overset{(c)}{\geq}\; \sum_{i \in X} d_{it} \Big[Q(\boldsymbol{y}_t \vee \mathbb{1}_{\{i\}}) - Q(\boldsymbol{y}_t)\Big] \\
&=\; \boldsymbol{w}_t^\top \boldsymbol{d}_t \;\geq\; Q(\boldsymbol{y}_t \vee \boldsymbol{y}^*) - Q(\boldsymbol{y}_t) \;\overset{(d)}{\geq}\; Q(\boldsymbol{y}^*) - Q(\boldsymbol{y}_t) \;\overset{(e)}{=}\; Z^* - Q(\boldsymbol{y}_t),
\end{aligned} \tag{13}$$

where (b) uses the expression for the partial derivative of Q in Lemma D.5(a), (c) follows by noting that Q is increasing in each coordinate by Lemma D.5(c), along with the fact that $\boldsymbol{y}_t \wedge \mathbb{1}_{X \setminus \{i\}} \leq \boldsymbol{y}_t$

and $0 \le d_{it} \le 1$, (d), once again, uses the monotonicity of Q by Lemma D.5(c) and (e) holds because Q coincides with q on $\{0,1\}^{|X|}$ and we have $\boldsymbol{y}^* \in \{0,1\}^{|X|}$. By Step 3 of the continuous greedy algorithm, we have $0 \le y_{i,t+\delta} - y_{it} = \delta\, d_{it} \le \delta$ for all $i \in X$, in which case, Lemma D.8 yields the chain of inequalities

$$\begin{aligned} Q(\boldsymbol{y}_{t+\delta}) - Q(\boldsymbol{y}_t) \;&\ge\; \sum_{i\in X} (y_{i,t+\delta} - y_{it})\, \partial_i Q(\boldsymbol{y}_t) - 2\,\delta^2 |X|^3 Z^* \;=\; \delta \sum_{i\in X} d_{it}\, \partial_i Q(\boldsymbol{y}_t) - 2\,\delta^2 |X|^3 Z^* \\ &\overset{(f)}{\ge}\; \delta\,(Z^* - Q(\boldsymbol{y}_t)) - 2\,\delta^2|X|^3 Z^*, \end{aligned}$$

where (f) follows from (13). The chain of inequalities above establishes the claim.

By (12), we have $Z^* - Q(\boldsymbol{y}_{t+\delta}) \le (1-\delta)(Z^* - Q(\boldsymbol{y}_t)) + 2\,\delta^2|X|^3 Z^*$. For each $k = 0, \ldots, \frac{1}{\delta} - 1$, let $x_k = Z^* - Q(\boldsymbol{y}_{k\delta})$. Note that $x_0 = Z^*$ because $Q(\underline{\mathbf{0}}) = 0$. Therefore, we have the inequality $x_{k+1} \le (1-\delta)x_k + 2\,\delta^2 |X|^3 Z^*$ for all $k = 0, 1, \ldots, \frac{1}{\delta} - 1$. In this case, applying Lemma D.9 with $\alpha = 1 - \delta$ and $\beta = 2\,\delta^2|X|^3 Z^*$, we obtain the chain of inequalities

$$\begin{aligned} Z^* - Q(\boldsymbol{y}_1) \;=\; x_{\frac{1}{\delta}} \;\le\; (1-\delta)^{1/\delta} x_0 + 2\,\delta^2 |X|^3 Z^* \sum_{h=0}^{\frac{1}{\delta}-1} (1-\delta)^h \\ \le\; (1-\delta)^{1/\delta} x_0 + \frac{2\,\delta^2|X|^3 Z^*}{\delta} \;\le\; \frac{Z^*}{e} + \epsilon\, Z^*, \end{aligned}$$

where the last inequality follows because $(1-a)^{1/a} \le \frac{1}{e}$ for all $0 < a \le 1$, $x_0 = Z^*$ and $\delta = \frac{\epsilon}{2\,|X|^3}$. Thus, we have $Q(\boldsymbol{y}_1) \ge \left(1 - \frac{1}{e} - \epsilon\right) Z^*$, which is the desired result. ■

D.4 Proof of Lemma D.3: Upper Bounding the Multilinear Extension

In this section, we give a proof for Lemma D.3. Our proof follows the same technique as in Ene and Nguyen (2018). Let $\overline{\boldsymbol{v}} = \overline{\boldsymbol{z}} - \lfloor \overline{\boldsymbol{z}} \rfloor \in [0,1]^{n\times m}$. By the definitions of H, G, $\overline{\boldsymbol{z}}$ and $\overline{\boldsymbol{y}}$, we have the identities $H(\overline{\boldsymbol{v}}) = \mathbb{E}\{f(\lfloor \overline{\boldsymbol{z}} \rfloor + \mathsf{R}(\overline{\boldsymbol{v}}))\} = \mathbb{E}\{f(\lfloor \mathbf{M}\overline{\boldsymbol{y}} \rfloor + \mathsf{R}(\overline{\boldsymbol{v}}))\}$ and $G(\overline{\boldsymbol{y}}) = \mathbb{E}\{f(\mathbf{M}\mathsf{R}(\overline{\boldsymbol{y}}))\}$. We want to show that $H(\overline{\boldsymbol{v}}) \ge G(\overline{\boldsymbol{y}})$. Letting $K = nm$, we enumerate the elements of the set $\mathcal{F} \times \mathcal{A}$ in an arbitrary order, say $\{u_1, u_2, \ldots, u_K\}$. For each $k = 0, 1, \ldots, K$, let $\mathbf{Z}^{(k)}$ be an integer random vector taking values in $\mathbb{Z}_+^K$ with the first k components having the same distribution as $[\mathbf{M}\mathsf{R}(\overline{\boldsymbol{y}})]_{u_i}$ for $i = 1, \ldots, k$, whereas the last $K - k$ components picked among $\{\lfloor \overline{z}_{u_i} \rfloor, \lceil \overline{z}_{u_i} \rceil\}$ so that the expectation is $\overline{z}_{u_i}$ for $i = k+1, \ldots, K$. Here, we use $[\mathbf{M}\mathsf{R}(\overline{\boldsymbol{y}})]_{u_i}$ to denote the component of $\mathbf{M}\mathsf{R}(\overline{\boldsymbol{y}})$ corresponding to u_i. In particular, we construct the vector $\mathbf{Z}^{(k)} = (Z^{(k)}_{u_i} : i = 1, \ldots, K)$ as follows. For each $i = 1, \ldots, k$, we have $Z^{(k)}_{u_i} = [\mathbf{M}\mathsf{R}(\overline{\boldsymbol{y}})]_{u_i}$, whereas for each $i = k+1, \ldots, K$, we have $Z^{(k)}_{u_i} = \lceil \overline{z}_{u_i} \rceil$ with probability $\overline{z}_{u_i} - \lfloor \overline{z}_{u_i} \rfloor$ and $Z^{(k)}_{u_i} = \lfloor \overline{z}_{u_i} \rfloor$ with probability $\lceil \overline{z}_{u_i} \rceil - \overline{z}_{u_i}$. In this case, for all $i = k+1, \ldots, K$, we have $\mathbb{E}\{Z^{(k)}_{u_i}\} = \lceil \overline{z}_{u_i} \rceil\,(\overline{z}_{u_i} - \lfloor \overline{z}_{u_i} \rfloor) + \lfloor \overline{z}_{u_i} \rfloor\,(\lceil \overline{z}_{u_i} \rceil - \overline{z}_{u_i}) = \overline{z}_{u_i}$, as desired. If $\overline{z}_{u_i}$ is an integer for some $i = k+1, \ldots, K$, then we simply set $Z_{u_i}^{(k)} = \overline{z}_{u_i}$. By our construction, we have $\mathbf{Z}^{(0)} = \lfloor \overline{\boldsymbol{z}} \rfloor + \mathsf{R}(\overline{\boldsymbol{v}})$ and $\mathbf{Z}^{(K)} = \mathbf{M}\mathsf{R}(\overline{\boldsymbol{y}})$, which implies that $\mathbb{E}\{f(\mathbf{Z}^{(0)})\} = H(\overline{\boldsymbol{v}})$ and $\mathbb{E}\{f(\mathbf{Z}^{(K)})\} = G(\overline{\boldsymbol{y}})$. To complete the proof, it suffices to show that

$$\mathbb{E}\{f(\mathbf{Z}^{(k-1)})\} \;\geq\; \mathbb{E}\{f(\mathbf{Z}^{(k)})\} \tag{14}$$

for all $k = 1, \ldots, K$. To show the inequality above, fix an arbitrary k. For all $j \neq k$, $Z_{u_j}^{(k)}$ and $Z_{u_j}^{(k-1)}$ are identically distributed, so we can couple the randomness so that $Z_{u_j}^{(k)} = Z_{u_j}^{(k-1)}$ for all $j \neq k$. Let $\mathbf{W}$ be a random vector obtained from $\mathbf{Z}^{(k)}$ by zeroing out the component corresponding to u_k.

Consider an arbitrary realization $\boldsymbol{w}$ of $\mathbf{W}$. Define a function on the nonnegative integers $L_{\boldsymbol{w}} : \mathbb{Z}_+ \to \mathbb{R}_+$ as $L_{\boldsymbol{w}}(a) = f(\boldsymbol{w} + a\mathbb{1}_{\{u_k\}})$ for each $a \in \mathbb{Z}_+$. By our construction, we have the identity $L_{\mathbf{W}}(Z_{u_k}^{(k)}) = f(\mathbf{Z}^{(k)})$ and $L_{\mathbf{W}}(Z_{u_k}^{(k-1)}) = f(\mathbf{Z}^{(k-1)})$. Furthermore, note that $Z_{u_k}^{(k)}$ is independent of $\mathbf{W}$ because each u_k corresponds to a particular pair $(i,a) \in \mathcal{F} \times \mathcal{A}$ and the value for this pair only depends on the random vector $\mathsf{R}(\overline{\boldsymbol{y}})$ through the components (i,a,k) for $k = 0, 1, 2, \ldots, \lceil \log(C^a \wedge U_i) \rceil$, independent of other random variables. By an analogous argument, it follows that $Z_{u_k}^{(k-1)}$ is independent of $\mathbf{W}$ as well. Because $\mathbf{W}$ is independent of $Z_{u_k}^{(k)}$ and $Z_{u_k}^{(k-1)}$, we can compute each of the expectations $\mathbb{E}\{f(\mathbf{Z}^{(k-1)})\}$ and $\mathbb{E}\{f(\mathbf{Z}^{(k)})\}$ as

$$\mathbb{E}\{f(\mathbf{Z}^{(k-1)})\} \;=\; \sum_{\boldsymbol{w} \in \mathbb{Z}_+^K} \mathbb{P}\{\mathbf{W} = \boldsymbol{w}\}\, \mathbb{E}\{f(\boldsymbol{w} + Z_{u_k}^{(k-1)} \mathbb{1}_{\{u_k\}})\} \;=\; \sum_{\boldsymbol{w} \in \mathbb{Z}_+^K} \mathbb{P}\{\mathbf{W} = \boldsymbol{w}\}\, \mathbb{E}\{L_{\boldsymbol{w}}(Z_{u_k}^{(k-1)})\},$$

$$\mathbb{E}\{f(\mathbf{Z}^{(k)})\} \;=\; \sum_{\boldsymbol{w} \in \mathbb{Z}_+^K} \mathbb{P}\{\mathbf{W} = \boldsymbol{w}\}\, \mathbb{E}\{f(\boldsymbol{w} + Z_{u_k}^{(k)} \mathbb{1}_{\{u_k\}})\} \;=\; \sum_{\boldsymbol{w} \in \mathbb{Z}_+^K} \mathbb{P}\{\mathbf{W} = \boldsymbol{w}\}\, \mathbb{E}\{L_{\boldsymbol{w}}(Z_{u_k}^{(k)})\}.$$

To establish (14), it suffices to show that we have $\mathbb{E}\{L_{\boldsymbol{w}}(Z_{u_k}^{(k-1)})\} \;\geq\; \mathbb{E}\{L_{\boldsymbol{w}}(Z_{u_k}^{(k)})\}$ for any realization $\boldsymbol{w}$ of $\mathbf{W}$. Thus, we fix an arbitrary realization $\boldsymbol{w}$. Let $\overline{L}_{\boldsymbol{w}} : \mathbb{R}_+ \to \mathbb{R}_+$ denote a linear interpolation of $L_{\boldsymbol{w}}$, consisting of straight lines that connect $(a, L_{\boldsymbol{w}}(a))$ with $(a+1, L_{\boldsymbol{w}}(a+1))$ for each $a \in \mathbb{Z}_+$. Note that $\overline{L}_{\boldsymbol{w}}$ is a piecewise linear function on the real line that agrees with $L_{\boldsymbol{w}}$ on nonnegative integers and $\overline{L}_{\boldsymbol{w}}$ does not have any breakpoint other than integers. Because f is DR-submodular, $\overline{L}_{\boldsymbol{w}}$ is concave, so by Jensen's inequality, we have $\overline{L}_{\boldsymbol{w}}(\mathbb{E}\{Z_{u_k}^{(k)}\}) \geq \mathbb{E}\{\overline{L}_{\boldsymbol{w}}(Z_{u_k}^{(k)})\}$. On the other hand, $Z_{u_k}^{(k-1)}$ is a random variable that takes one of the values $\lfloor \overline{z}_{u_k} \rfloor$ or $\lceil \overline{z}_{u_k} \rceil$, in which case, since $\overline{L}_{\boldsymbol{w}}$ is linear on the interval $[\lfloor \overline{z}_{u_k} \rfloor, \lceil \overline{z}_{u_k} \rceil]$, we have

$$\mathbb{E}\Big\{\overline{L}_{\boldsymbol{w}}\Big(Z_{u_k}^{(k-1)}\Big)\Big\} \;=\; \overline{L}_{\boldsymbol{w}}\Big(\mathbb{E}\Big\{Z_{u_k}^{(k-1)}\Big\}\Big) \;\overset{(a)}{=}\; \overline{L}_{\boldsymbol{w}}(\overline{z}_{u_k}) \;\overset{(b)}{=}\; \overline{L}_{\boldsymbol{w}}\Big(\mathbb{E}\Big\{Z_{u_k}^{(k)}\Big\}\Big),$$

where (a) holds because we have $\mathbb{E}\Big\{Z_{u_k}^{(k-1)}\Big\} = \overline{z}_{u_k}$ due to our construction of $Z^{(k-1)}$. The last equality (b) follows because we have $\mathbb{E}\{\mathsf{R}(\overline{\boldsymbol{y}})\} = \overline{\boldsymbol{y}}$ by the definition of the random vector R and the mapping $\mathbf{M}$ is linear, so $\mathbb{E}\{Z_{u_k}^{(k)}\} = [\mathbb{E}\{\mathbf{M}\mathsf{R}(\overline{\boldsymbol{y}})\}]_{u_k} = [\mathbf{M}\overline{\boldsymbol{y}}]_{u_k} = \overline{z}_{u_k}$, where the last equality holds because $\overline{\boldsymbol{z}} = \mathbf{M}\overline{\boldsymbol{y}}$. We have $\overline{L}_{\boldsymbol{w}}(\mathbb{E}\{Z^{(k)}_{u_k}\}) \geq \mathbb{E}\{\overline{L}_{\boldsymbol{w}}(Z^{(k)}_{u_k})\}$ by the earlier discussion in this paragraph, in which case, the chain of inequalities above yields $\mathbb{E}\{\overline{L}_{\boldsymbol{w}}(Z^{(k-1)}_{u_k})\} \geq \mathbb{E}\{\overline{L}_{\boldsymbol{w}}(Z^{(k)}_{u_k})\}$. Because $L_{\boldsymbol{w}}$ and $\overline{L}_{\boldsymbol{w}}$ agree on nonnegative integers and both $Z^{(k-1)}_{u_k}$ and $Z^{(k)}_{u_k}$ always take integer values, the last inequality is equivalent to $\mathbb{E}\{L_{\boldsymbol{w}}(Z^{(k-1)}_{u_k})\} \geq \mathbb{E}\{L_{\boldsymbol{w}}(Z^{(k)}_{u_k})\}$. Since this inequality holds for any realization $\boldsymbol{w}$ of $\mathbf{W}$, it follows that $\mathbb{E}\{f(\mathbf{Z}^{(k-1)})\} \geq \mathbb{E}\{f(\mathbf{Z}^{(k)})\}$, which is the desired result. ■

Appendix E: Performance Guarantee for the Randomized Policy

We give a proof for Theorem 7.1. The proof uses two lemmas. In the next lemma, we focus on the performance guarantee of $\frac{1}{2} f^a_{\mathsf{LP}}(\widehat{\boldsymbol{z}}^a)$ for the randomized policy.

Lemma E.1 (Constant Factor) *There exists a choice of the tuning parameter γ such that if the placement decisions for product a are given by the vector $\widehat{\boldsymbol{z}}^a$, then the total expected profit that the randomized policy obtains from product a is at least $\frac{1}{2} f^a_{\mathsf{LP}}(\widehat{\boldsymbol{z}}^a)$.*

Proof: Considering the dynamic program that we use to compute the value functions $\{V^a_{it} : t \in \mathcal{T}\}$, the dummy promise provides an objective value of zero for the maximization problem in this dynamic program. Thus, we have $V^a_{it}(x^a_{it}) \geq V^a_{i,t+1}(x^a_i)$. Using a standard argument, we can also show that the value functions $\{V^a_{it} : t \in \mathcal{T}\}$ are concave, so $V^a_{it}(\ell) - V^a_{it}(\ell - 1) \geq V^a_{it}(\widehat{z}^a_i) - V^a_{it}(\widehat{z}^a_i - 1)$ for all $\ell = 1, 2, \ldots, \widehat{z}^a_i$. Thus, we get $\widehat{V}^a_{it}(\widehat{z}^a_i) = \sum_{\ell=1}^{\widehat{z}^a_i} \big[V^a_{it}(\ell) - V^a_{it}(\ell-1)\big] \geq \widehat{z}^a_i \, \big[\widehat{V}^a_{it}(\widehat{z}^a_i) - \widehat{V}^a_{it}(\widehat{z}^a_i - 1)\big]$. Lastly, for $(\alpha_k : k \in \mathcal{K}) \in \mathbb{R}^{|\mathcal{K}|}_+$ and $(\beta_k : k \in \mathcal{K}) \in \mathbb{R}^{|\mathcal{K}|}$, we have $\big(\sum_{k\in\mathcal{K}} \alpha_k\big)\big(\max_{k\in\mathcal{K}} \beta_k\big) \geq \sum_{k\in\mathcal{K}} \alpha_k \, \beta_k$. In this case, letting $v^a_{it} = V^a_{it}(\widehat{z}^a_i)/\widehat{z}^a_i$ for notational brevity, if $\widehat{z}^a_i \geq 1$, then using the dynamic program that we use to compute the value functions $\{V^a_{it} : t \in \mathcal{T}\}$, we have

$$
\begin{aligned}
v^a_{it} &= \frac{\gamma}{\widehat{z}^a_i} \sum_{j\in\mathcal{D}} \lambda^a_{jt}\, \widehat{\eta}^a_{ij} \max_{k\in\mathcal{K}} \left\{ \theta^a_{jk} \Big[r^a_{ijk} + V^a_{i,t+1}(\widehat{z}^a_i - 1) - V^a_{i,t+1}(\widehat{z}^a_i) \Big] \right\} + v^a_{i,t+1} \\
&\overset{(a)}{\geq} \frac{\gamma}{\widehat{z}^a_i} \sum_{j\in\mathcal{D}} \lambda^a_{jt}\, \widehat{\eta}^a_{ij} \max_{k\in\mathcal{K}} \left\{ \theta^a_{jk} \Big[r^a_{ijk} - v^a_{i,t+1} \Big] \right\} + v^a_{i,t+1} \\
&\overset{(b)}{=} \frac{\gamma}{\widehat{z}^a_i} \sum_{j\in\mathcal{D}} \frac{\lambda^a_{jt}}{\sum_{\tau\in\mathcal{T}} \lambda^a_{j\tau}} \left(\sum_{k\in\mathcal{K}} \widehat{w}^a_{ijk} \right) \max_{k\in\mathcal{K}} \left\{ \theta^a_{jk} \Big[r^a_{ijk} - v^a_{i,t+1} \Big] \right\} + v^a_{i,t+1} \\
&\overset{(c)}{\geq} \frac{\gamma}{\widehat{z}^a_i} \sum_{j\in\mathcal{D}} \frac{\lambda^a_{jt}}{\sum_{\tau\in\mathcal{T}} \lambda^a_{j\tau}} \sum_{k\in\mathcal{K}} \widehat{w}^a_{ijk}\, \theta^a_{jk} \Big[r^a_{ijk} - v^a_{i,t+1} \Big] + v^a_{i,t+1} \\
&\overset{(d)}{\geq} \frac{\gamma}{\widehat{z}^a_i} \sum_{j\in\mathcal{D}} \frac{\lambda^a_{jt}}{\sum_{\tau\in\mathcal{T}} \lambda^a_{j\tau}} \sum_{k\in\mathcal{K}} \widehat{w}^a_{ijk}\, \theta^a_{jk} \Big[r^a_{ijk} - v^a_{i1} \Big] + v^a_{i,t+1}
\end{aligned}
$$

where (a) holds because $\widehat{V}^a_{it}(\widehat{z}^a_i)/\widehat{z}^a_i \geq \widehat{V}^a_{it}(\widehat{z}^a_i) - \widehat{V}^a_{it}(\widehat{z}^a_i - 1)$, (b) uses the definition of $\widehat{\eta}^a_{ij}$, (c) uses the fact that $\big(\sum_{k\in\mathcal{K}} \alpha_k\big)\big(\max_{k\in\mathcal{K}} \beta_k\big) \geq \sum_{k\in\mathcal{K}} \alpha_k \, \beta_k$ for all $(\alpha_k : k \in \mathcal{K}) \in \mathbb{R}^{|\mathcal{K}|}_+$ and $(\beta_k : k \in \mathcal{K}) \in \mathbb{R}^{|\mathcal{K}|}$, and (d) follows by noting that $V^a_{it}(x^a_{it}) \ge V^a_{i,t+1}(x^a_i)$. Adding the chain of inequalities above over all $t \in \mathcal{T}$ and using the boundary condition $V^a_{i,T+1}(x^a_i) = 0$, we obtain

$$\begin{aligned} v^a_{i1} \;&\ge\; \frac{\gamma}{\widehat{z}^a_i} \sum_{t\in\mathcal{T}} \sum_{j\in\mathcal{D}} \frac{\lambda^a_{jt}}{\sum_{\tau\in\mathcal{T}} \lambda^a_{j\tau}} \sum_{k\in\mathcal{K}} \widehat{w}^a_{ijk}\, \theta^a_{jk} \Big[r^a_{ijk} - v^a_{i1} \Big] \\ &\stackrel{(e)}{=}\; \frac{\gamma}{\widehat{z}^a_i} \sum_{j\in\mathcal{D}} \sum_{k\in\mathcal{K}} r^a_{ijk}\, \theta^a_{jk}\, \widehat{w}^a_{ijk} - \frac{\gamma}{\widehat{z}^a_i}\, v^a_{i1} \sum_{j\in\mathcal{D}} \sum_{j\in\mathcal{K}} \theta^a_{jk}\, \widehat{w}^a_{ijk} \;\stackrel{(f)}{\ge}\; \frac{\gamma}{\widehat{z}^a_i} \sum_{j\in\mathcal{D}} \sum_{k\in\mathcal{K}} r^a_{ijk}\, \theta^a_{jk}\, \widehat{w}^a_{ijk} - \gamma\, v^a_{i1}, \end{aligned}$$

where (e) is by arranging the terms and (f) holds because $\widehat{\boldsymbol{w}}^a$ is an optimal solution to the Bounding problem with $\boldsymbol{z}^a = \widehat{\boldsymbol{z}}^a$ so that $\sum_{j\in\mathcal{D}} \sum_{k\in\mathcal{K}} \theta^a_{jk}\, w^a_{ijk} \le \widehat{z}^a_i$. By the chain of inequalities above, we obtain $\frac{1+\gamma}{\gamma}\, v^a_{i1}\, \widehat{z}^a_i \ge \sum_{j\in\mathcal{D}} \sum_{k\in\mathcal{K}} r^a_{ijk}\, \theta_{jk} \widehat{w}^a_{ijk}$. Choosing the tuning parameter as $\gamma = 1$, noting that $v^a_{i1}\, \widehat{z}^a_i = V^a_{i1}(\widehat{z}^a_i)$ and adding the last inequality over all $i \in \mathcal{F}$, we have $2 \sum_{i\in\mathcal{F}} V^a_{i1}(\widehat{z}^a_i) \ge \sum_{i\in\mathcal{F}} \sum_{j\in\mathcal{D}} \sum_{k\in\mathcal{K}} r^a_{ijk}\, \theta_{jk} \widehat{w}^a_{ijk} = f^a_{\mathsf{LP}}(\widehat{\boldsymbol{z}}^a)$, where the last equality holds because $\widehat{\boldsymbol{w}}^a$ is an optimal solution to the Bounding problem with $\boldsymbol{z}^a = \widehat{\boldsymbol{z}}^a$. The result follows because if the placement decisions for product a are $\widehat{\boldsymbol{z}}^a$, then the total expected profit of the randomized policy is $\sum_{i\in\mathcal{F}} V^a_{i1}(\widehat{z}^a_i)$. ■

We consider a policy that assigns a demand for product a from demand region j to fulfillment center and promise pair (i,k) with probability $\gamma \frac{\widehat{w}^a_{ijk}}{\sum_{t\in\mathcal{T}} \lambda^a_{jt}}$. In particular, at any time period, if we have a demand for product a from demand region j, then we sample a fulfillment center and promise pair (i,k) with probability $\gamma \frac{\widehat{w}^a_{ijk}}{\sum_{t\in\mathcal{T}} \lambda^a_{jt}}$. If there is remaining inventory for product a at the sampled fulfillment center i, then we offer the sampled promise k and use the sampled fulfillment center i if the customer accepts the given promise. If there is no remaining inventory at the sampled fulfillment center, then we offer the dummy promise. We refer to this policy as the fully randomized policy. The total expected profit of our original randomized policy is at least as large as the total expected profit of the fully randomized policy. In particular, both policies assign a demand for product a from demand region j to fulfillment center i with probability $\gamma \frac{\sum_{k\in\mathcal{K}} \widehat{w}^a_{ijk}}{\sum_{t\in\mathcal{T}} \lambda^a_{jt}}$. However, the original randomized policy chooses a delivery promise for the demand to maximize the total expected profit, whereas the fully randomized policy randomly chooses a delivery promise for the demand. Thus, to lower bound the performance of the original randomized policy, it is enough to lower bound the performance of the fully randomized policy. In the next lemma, we focus on the performance guarantee of $1 - O\Big(\sqrt{\frac{\log \widehat{z}_{\min}}{\widehat{z}_{\min}}}\Big)$ for the randomized policy.

Lemma E.2 (Asymptotic) *There exists a choice of the tuning parameter γ such that if the placement decisions for product a are given by the vector $\widehat{\boldsymbol{z}}^a$ with $\widehat{z}_{\min} = \min_{i\in\mathcal{F}}\{\widehat{z}^a_i : \widehat{z}^a_i \ge 1\}$, then the total expected profit of the randomized policy from product a is at least $\Big(1 - O\Big(\sqrt{\frac{\log \widehat{z}_{\min}}{\widehat{z}_{\min}}}\Big)\Big) f^a_{\mathsf{LP}}(\widehat{\boldsymbol{z}}^a)$.*

Proof: By the discussion just before the lemma, it is enough to lower bound the total expected profit of the fully randomized policy with $1 - O\Big(\sqrt{\frac{\log \widehat{z}_{\min}}{\widehat{z}_{\min}}}\Big) f^a_{\mathsf{LP}}(\widehat{\boldsymbol{z}}^a)$. We define three random variables.

The Bernoulli random variable N^a_{it} takes value one if the fully randomized policy assigns a demand for product a at time period t to fulfillment center i along with some delivery promise and the customer accepts the promise. We have $\mathbb{E}\{N^a_{it}\} = \sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}} \lambda^a_{jt}\,\gamma\,\frac{\widehat{w}^a_{ijk}}{\sum_{\ell\in\mathcal{T}}\lambda^a_{j\ell}}\,\theta^a_{jk}$. The Bernoulli random variable G^a_{it} takes value one if there is remaining inventory for product a at fulfillment center i at time period t under the fully randomized policy. We have $G^a_{it}=1$ if and only of $\sum_{\ell=1}^{t-1} N^a_{i\ell} < \widehat{z}^a_i$. Lastly, the random variable R^a_{it} corresponds to the profit that the fully randomized policy obtains at time period t from the inventory for product a available at fulfillment center i. We have $\mathbb{E}\{R^a_{it}\,|\,G^a_{it}=1\} = \sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}} r^a_{ijk}\,\lambda^a_{jt}\,\gamma\,\frac{\widehat{w}^a_{ijk}}{\sum_{\ell\in\mathcal{T}}\lambda^a_{j\ell}}\,\theta^a_{jk}$ and $\mathbb{E}\{R^a_{it}\,|\,G^a_{it}=0\}=0$. The total expected profit of the fully randomized policy from product a is given by

$$\begin{aligned}\mathsf{APX} \;=\; \sum_{t\in\mathcal{T}}\sum_{i\in\mathcal{F}}\mathbb{E}\{R^a_{it}\} \;&=\; \sum_{t\in\mathcal{T}}\sum_{i\in\mathcal{F}}\mathbb{E}\{R^a_{it}\,|\,G^a_{it}=1\}\,\mathbb{P}\{G^a_{it}=1\}\\ &=\; \sum_{t\in\mathcal{T}}\sum_{i\in\mathcal{F}}\sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}} r^a_{ijk}\,\lambda^a_{jt}\,\gamma\,\frac{\widehat{w}^a_{ijk}}{\sum_{\ell\in\mathcal{T}}\lambda^a_{j\ell}}\,\theta^a_{jk}\,\mathbb{P}\{G^a_{it}=1\}.\end{aligned}$$

To lower bound the total expected profit of the randomized policy from product a, we proceed to lower bounding the probability $\mathbb{P}\{G^a_{it}=1\}$ on the right side above.

We have remaining inventory for product a at fulfillment center i at time period t if and only if the number of times a demand for product a is assigned to fulfillment center i and the promise is accepted over the first $t-1$ time periods does not exceed the number of units at the fulfillment center. Thus, $G^a_{it}=1$ if and only of $\sum_{\ell=1}^{t-1} N^a_{i\ell} < \widehat{z}^a_i$, so we get $\mathbb{P}\{G^a_{it}=1\} = \mathbb{P}\{\sum_{\ell=1}^{t-1} N^a_{i\ell} < \widehat{z}^a_i\} \geq \mathbb{P}\{\sum_{\ell\in\mathcal{T}} N^a_{i\ell} < \widehat{z}^a_i\}$, where the inequality holds because $\sum_{\ell=1}^{t-1} N^a_{i\ell} \leq \sum_{\ell\in\mathcal{T}} N^a_{i\ell}$ with probability one. Also, we have $\sum_{t\in\mathcal{T}}\mathbb{E}\{N^a_{it}\} = \sum_{t\in\mathcal{T}}\sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}}\lambda^a_{jt}\,\gamma\,\frac{\widehat{w}^a_{ijk}}{\sum_{\ell\in\mathcal{T}}\lambda^a_{j\ell}}\,\theta^a_{jk} = \gamma\sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}}\theta^a_{jk}\,\widehat{w}^a_{ijk} \leq \gamma\,\widehat{z}^a_i$, where the inequality uses the fact that $\widehat{\boldsymbol{w}}^a$ is a feasible solution to the $\mathsf{Bounding}$ problem with $\boldsymbol{z}^a = \widehat{\boldsymbol{z}}^a$. Since $\{N^a_{it} : t\in\mathcal{T}\}$ are independent Bernoulli random variables, we get $\mathrm{Var}\big(\sum_{t\in\mathcal{T}} N^a_{it}\big) = \sum_{t\in\mathcal{T}}\mathrm{Var}(N^a_{it}) \leq \sum_{t\in\mathcal{T}}\mathbb{E}\{N^a_{it}\} \leq \gamma\,\widehat{z}^a_i$. In this case, the one-sided Bernstein inequality yields

$$\begin{aligned}\mathbb{P}\left\{\sum_{t\in\mathcal{T}} N^a_{it} \geq \widehat{z}^a_i\right\} &\overset{(a)}{\leq} \mathbb{P}\left\{\sum_{t\in\mathcal{T}}\Big(N^a_{it} - \mathbb{E}\{N^a_{it}\}\Big) \geq (1-\gamma)\,\widehat{z}^a_i\right\}\\ &\overset{(b)}{\leq} \exp\left(-\frac{\frac{1}{2}(1-\gamma)^2(\widehat{z}^a_i)^2}{\sum_{t\in\mathcal{T}}\mathrm{Var}(N^a_{it}) + \frac{1}{3}(1-\gamma)\,\widehat{z}^a_i}\right) \overset{(c)}{\leq} \exp\left(-\frac{\frac{1}{2}(1-\gamma)^2(\widehat{z}^a_i)^2}{\gamma\,\widehat{z}^a_i + \frac{1}{3}(1-\gamma)\,\widehat{z}^a_i}\right)\\ &\overset{(d)}{\leq} \exp\left(-\frac{\frac{1}{2}(1-\gamma)^2\,\widehat{z}_{\min}}{\gamma + \frac{1}{3}(1-\gamma)}\right) \overset{(e)}{\leq} \exp\left(-\frac{1}{2}(1-\gamma)^2\,\widehat{z}_{\min}\right),\end{aligned}$$

where (a) holds because $\sum_{t\in\mathcal{T}}\mathbb{E}\{N^a_{it}\} \leq \gamma\,\widehat{z}^a_i$, (b) is the one-sided Bernstein inequality, (c) holds because $\sum_{i\in\mathcal{T}}\mathrm{Var}(N^a_{it}) \leq \gamma\,\widehat{z}^a_i$, (d) is by $\widehat{z}^a_i \geq \widehat{z}_{\min}$ and (e) uses the fact that $\frac{2}{3}\gamma + \frac{1}{3} \leq 1$.

Choosing the tuning parameter as $\gamma = 1 - \sqrt{\frac{2\log\widehat{z}_{\min}}{\widehat{z}_{\min}}}$, we have $\exp(-\frac{1}{2}(1-\gamma)^2\,\widehat{z}_{\min}) = 1/\widehat{z}_{\min}$ and the chain of inequalities above yields $\mathbb{P}\{G^a_{it}=1\} \geq \mathbb{P}\{\sum_{\ell\in\mathcal{T}} N^a_{i\ell} < \widehat{z}^a_i\} \geq 1 - 1/\widehat{z}_{\min}$. Noting our choice of the tuning parameter, we can lower bound the total expected profit of the fully randomized policy from product a as

$$\begin{aligned}
\mathsf{APX} \;&=\; \left(1-\sqrt{\frac{2\log\widehat{z}_{\min}}{\widehat{z}_{\min}}}\right)\sum_{t\in\mathcal{T}}\sum_{i\in\mathcal{F}}\sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}} r^a_{ijk}\,\lambda^a_{jt}\,\frac{\widehat{w}^a_{ijk}}{\sum_{\ell\in\mathcal{T}}\lambda^a_{j\ell}}\,\theta^a_{jk}\,\mathbb{P}\{G^a_{it}=1\}\\
&\overset{(f)}{\geq}\; \left(1-\sqrt{\frac{2\log\widehat{z}_{\min}}{\widehat{z}_{\min}}}\right)\left(1-\frac{1}{\widehat{z}_{\min}}\right)\sum_{i\in\mathcal{F}}\sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}} r^a_{ijk}\,\theta^a_{jk}\,\widehat{w}^a_{ijk}\\
&\overset{(g)}{=}\; \left(1-\sqrt{\frac{2\log\widehat{z}_{\min}}{\widehat{z}_{\min}}}\right)\left(1-\frac{1}{\widehat{z}_{\min}}\right) f^a_{\mathsf{LP}}(\widehat{\boldsymbol{z}}^a)\\
&\overset{(h)}{=}\; \left(1-O\left(\sqrt{\frac{\log\widehat{z}_{\min}}{\widehat{z}_{\min}}}\right)\right) f^a_{\mathsf{LP}}(\widehat{\boldsymbol{z}}^a),
\end{aligned}$$

where (f) holds by $\mathbb{P}\{G^a_{it}=1\}\geq 1-1/\widehat{z}_{\min}$, (g) holds because $\widehat{\boldsymbol{w}}^a$ is an optimal solution to the Bounding problem with $\boldsymbol{z}^a=\widehat{\boldsymbol{z}}^a$ and (h) uses $(1-a)(1-b)\geq 1-a-b$ for $a,b\in[0,1]$. ∎

We use the two lemmas in this section to give a proof for Theorem 7.1.

Proof of Theorem 7.1:

Considering the performance guarantee of $\max\left\{\frac{1}{2},\,1-O\left(\sqrt{\frac{\log\widehat{z}_{\min}}{\widehat{z}_{\min}}}\right)\right\} f^a_{\mathsf{LP}}(\widehat{\boldsymbol{z}}^a)$, the two terms in the maximum operator, respectively, follow from Lemmas E.1 and E.2. ∎

In our discussion of the fully randomized policy earlier in this section, we argued that the total expected profit of the original randomized policy is at least as large as the total expected profit of the fully randomized policy. We can also show this fact by constructing a dynamic program to compute the total expected profit of the fully randomized policy. Let $\nu^a_{it}(x^a_i)$ be the total expected profit from the demands for product a assigned to fulfillment center i by the fully randomized policy over time periods $\{t,\ldots,T\}$, given that the remaining inventory of product a at fulfillment center i at the beginning of time period t is x^a_i. Using the boundary condition $\nu^a_{i,T+1}=0$, we can compute the value functions $\{\nu^a_{it}:t\in\mathcal{T}\}$ by using the dynamic program

$$\nu^a_{it}(x^a_i) \;=\; \sum_{j\in\mathcal{D}}\lambda^a_{jt}\,\gamma\sum_{k\in\mathcal{K}}\frac{\widehat{w}^a_{ijk}}{\sum_{\tau\in\mathcal{T}}\lambda^a_{j\tau}}\left\{\mathbb{1}_{(x^a_i\geq 1)}\,\theta^a_{jk}\Big[r^a_{ijk}+\nu^a_{i,t+1}(x^a_i-1)-\nu^a_{i,t+1}(x^a_i)\Big]\right\}+\nu^a_{i,t+1}(x^a_i). \quad (15)$$

The dynamic program above follows from the same reasoning as the one in the dynamic program that we use to compute the value functions $\{V^a_{it}:t\in\mathcal{T}\}$, but the dynamic program above chooses a promise randomly. Comparing the dynamic program in (15) with the one that we use to compute the value functions $\{V^a_{it}:t\in\mathcal{T}\}$, noting the fact that $\widehat{\eta}^a_{ij}=\frac{\sum_{k\in\mathcal{K}}\widehat{w}^a_{ijk}}{\sum_{\tau\in\mathcal{T}}\lambda^a_{j\tau}}$, we can use induction over the time periods to show that $V^a_{it}(x^a_i)\geq\nu^a_{it}(x^a_i)$ for all $x^a_i\in\mathbb{Z}_+$. The only critical step is to use the fact that $\left(\sum_{k\in\mathcal{K}}\alpha_k\right)\left(\max_{k\in\mathcal{K}}\beta_k\right)\geq\sum_{k\in\mathcal{K}}\alpha_k\,\beta_k$ for two vectors $(\alpha_k:k\in\mathcal{K})\in\mathbb{R}^{|\mathcal{K}|}_+$ and $(\beta_k:k\in\mathcal{K})\in\mathbb{R}^{|\mathcal{K}|}$. This approach provides an alternative argument that the total expected profit of the original randomized policy is at least as large as that of the fully randomized policy.

Appendix F: Using the Magician Problem to Improve the Performance Guarantee

Using $\widehat{z}_i^a$ to denote the number of units for product a that we place at fulfillment center i at the beginning of the selling horizon and letting $\widehat{z}_{\min} = \min_{i \in \mathcal{F}} \{\widehat{z}_i^a : \widehat{z}_i^a \geq 1\}$ to capture the smallest nonzero placement quantity for product a at any fulfillment center, we construct a delivery promise and fulfillment policy for product a with a performance guarantee of $\max\left\{\frac{1}{2}, 1 - O\left(\frac{1}{\sqrt{\widehat{z}_{\min}}}\right)\right\}$. To obtain this performance guarantee, we use the recent results on the magician problem. Magician problem was coined by Alaei (2014). In this section, we follow Jiang et al. (2023), where the authors generalize and tighten the performance guarantees in Alaei (2014). The policy that we construct is similar to the fully randomized policy discussed right after Theorem 7.1, as it assigns a demand for product a from demand region j to fulfillment center and promise pair (i,k) with a prefixed probability. However, for the demand assigned to fulfillment center i, we flip a coin with a success probability that depends on the inventory availability at the fulfillment center. If the coin flip is a success, then we offer the assigned delivery promise k. If the coin flip is a failure, then we offer the dummy promise. Below is a description of the approximate policy that we study.

Approximate Delivery Promise and Fulfillment Policy:

Given the placement decisions $\widehat{\boldsymbol{z}}^a = (\widehat{z}_i^a : i \in \mathcal{F})$ for product a, we solve the linear program in the Bounding problem with $\boldsymbol{z}^a = \widehat{\boldsymbol{z}}^a$. Letting $\widehat{\boldsymbol{w}}^a = (\widehat{w}_{ijk}^a : i \in \mathcal{F},\ j \in \mathcal{D},\ k \in \mathcal{K})$ be an optimal solution, if we have a demand for product a from demand region j at time period t, then we assign this demand to fulfillment center and promise pair (i,k) with probability $\frac{\widehat{w}_{ijk}^a}{\sum_{\ell \in \mathcal{T}} \lambda_{j\ell}^a}$. In this case, a demand for product a at time period t is assigned to fulfillment center i and this demand is willing to accept the assigned promise with probability $p_{it}^a = \sum_{j \in \mathcal{D}} \sum_{k \in \mathcal{K}} \lambda_{jt}^a \frac{\widehat{w}_{ijk}^a}{\sum_{\ell \in \mathcal{T}} \lambda_{j\ell}^a} \theta_{jk}^a$. Given that we served $q-1$ units of demand so far from fulfillment center i, so that the demand to be served will use the q-th unit, if fulfillment center i receives a demand for product a at time period t, then we are willing to serve the demand with probability $\delta_{it}^a(q)/p_{it}^a$. If we are willing to serve the demand, then we offer the assigned promise k to the demand. Otherwise, we offer the dummy promise. We set $\delta_{it}^a(\widehat{z}_i^a + 1) = 0$, so that if fulfillment center i is out of inventory, then we are not willing to serve the demand. We shortly specify the choice of the tuning parameters $(\delta_{it}^a(q) : i \in \mathcal{F},\ t \in \mathcal{T},\ q = 1, \ldots, \widehat{z}_i^a)$. We have the following performance guarantee for the approximate policy.

Theorem F.1 (Policy Performance) *There exists a choice of the tuning parameters* $(\delta_{it}^a(q) : i \in \mathcal{F},\ t \in \mathcal{T},\ q = 1, \ldots, \widehat{z}_i^a)$ *such that if the placement decisions for product a are given by the vector $\widehat{\boldsymbol{z}}^a$ with $\widehat{z}_{\min} = \min_{i \in \mathcal{F}} \{\widehat{z}_i^a : \widehat{z}_i^a \geq 1\}$, then the total expected profit that the approximate policy obtains from product a is at least* $\left(1 - \frac{1}{\sqrt{\widehat{z}_{\min} + 3}}\right) f_{\mathsf{LP}}^a(\widehat{\boldsymbol{z}}^a)$.

Note that because $\widehat{z}_{\min} \geq 1$, we have $1 - \frac{1}{\sqrt{\widehat{z}_{\min} + 3}} \geq \frac{1}{2}$. Before we give the proof of the theorem above, we will connect our approximate policy to the magician problem. Our vocabulary in this section will primarily follow Jiang et al. (2023), where the authors generalize the approach in Alaei (2014) to additional problem classes and give a tightened analysis.

Relationship to the Magician Problem:

In the magician problem, we have a sequence of queries indexed by $\mathcal{T} = \{1, \ldots, T\}$. The queries arrive sequentially. The status of each query can be active or inactive. Query t is active with probability ρ_t. The status of the queries are independent. When a query arrives, we decide whether we are willing to serve the query. Subsequently, we observe the status of the query. If we are willing to serve the query and the status of the query is active, then we serve the query. Otherwise, we do not serve the query. We can serve at most a total of κ queries. Once we serve κ queries, we cannot be willing to serve other queries. For given $\gamma \in (0,1)$, our goal is to find a policy that ensures that we are willing to serve each query with a probability of at least γ. Naturally, to maximize the expected number of served queries, we need to choose the value of γ as large as possible. If $\sum_{t=1}^{T} \rho_t \leq \kappa$, so that the total expected number of active queries does not exceed the service capacity, then Alaei (2014) gives a policy that ensures that we can choose $\gamma = 1 - \frac{1}{\sqrt{\kappa+3}}$. Jiang et al. (2023) show that one can formulate a linear program that finds the maximum value of γ, while ensuring that we are willing to serve each query with a probability of at least γ. This linear program does not have a closed form solution, but it is guaranteed to find a value for γ that is at least $1 - \frac{1}{\sqrt{\kappa+3}}$.

We map this description of the magician problem to our approximate policy. We focus on each fulfillment center separately. Each time period corresponds to a query. Consider the queries assigned to fulfillment center i. Given that we served $q-1$ queries so far from fulfillment center i, we are willing to serve the query at time period t with probability $\delta^a_{it}(q)/p^a_{it}$. The query at fulfillment center i at time period t is active with probability p^a_{it}. If we are willing to serve the query and the status of the query is active, then we serve the query. Therefore, noting that p^a_{it} is the probability that a demand for product a at time period t is assigned to fulfillment center i and this demand is willing to accept the assigned promise, we serve a demand for product a using fulfillment center i whenever this demand is assigned to fulfillment center i, we are willing to serve the demand and the demand is willing to accept the offered promise. We can serve at most $\widehat{z}^a_i$ queries from fulfillment center i. Using the definition of p^a_{it}, because $\widehat{\boldsymbol{w}}^a$ is a feasible solution to the Bounding problem with $\boldsymbol{z}^a = \widehat{\boldsymbol{z}}^a$, we get $\sum_{t\in\mathcal{T}} p^a_{it} = \sum_{t\in\mathcal{T}} \sum_{j\in\mathcal{D}} \sum_{k\in\mathcal{K}} \lambda^a_{jt} \frac{\widehat{w}^a_{ijk}}{\sum_{\ell\in\mathcal{T}} \lambda^a_{j\ell}} \theta^a_{jk} = \sum_{j\in\mathcal{D}} \sum_{k\in\mathcal{K}} \theta^a_{jk} \widehat{w}^a_{ijk} \leq \widehat{z}^a_i$, so the total expected number of active queries at fulfillment center i does not exceed the service capacity, which is needed in the analyses in Alaei (2014) and Jiang et al. (2023).

Performance Guarantee:

The performance guarantee in Theorem F.1 follows from the ideas in Jiang et al. (2023). Under our approximate policy, letting ζ^a_{it} be the probability that we are willing to serve a demand for product a assigned to fulfillment center i at time period t, the expected profit from product a at fulfillment center i at time period t is $\sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}}\lambda^a_{jt}\frac{\widehat{w}^a_{ijk}}{\sum_{\ell\in\mathcal{T}}\lambda^a_{j\ell}}\zeta^a_{it}\,\theta^a_{jk}\,r^a_{ijk}$, where we use the fact that we collect profit if we assign the demand to fulfillment center i, we are willing to serve the demand and the offered promise is accepted. So, letting APX be the total expected profit of our approximate policy, noting that $\widehat{\boldsymbol{w}}^a$ is optimal to the $\mathsf{Bounding}$ problem with $\boldsymbol{z}^a=\widehat{\boldsymbol{z}}^a$, if we can lower bound the probability ζ^a_{it} with γ for all $i\in\mathcal{F}$ and $t\in\mathcal{T}$, then the total expected profit of our approximate policy is $\mathsf{APX}=\sum_{i\in\mathcal{F}}\sum_{t\in\mathcal{T}}\sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}}\lambda^a_{jt}\frac{\widehat{w}^a_{ijk}}{\sum_{\ell\in\mathcal{T}}\lambda^a_{j\ell}}\zeta^a_{it}\,\theta^a_{jk}\,r^a_{ijk}\geq\gamma\sum_{i\in\mathcal{F}}\sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}}r^a_{ijk}\,\theta^a_{jk}\,\widehat{w}^a_{ijk}=\gamma\,f^a_{\mathsf{LP}}(\widehat{\boldsymbol{z}}^a)$. In this case, Theorem F.1 follows as long as we can choose the tuning parameters $(\delta^a_{it}(q):i\in\mathcal{F},\ t\in\mathcal{T},\ q=1,\ldots,\widehat{z}^a_i)$ so that we are willing to serve each demand for product a assigned to fulfillment center i at least with probability $1-\frac{1}{\sqrt{\widehat{z}_{\min}+3}}$. We use a linear program to choose the tuning parameters. We focus on each fulfillment center separately. We use the decision variable ζ^a_i to capture the lower bound on the probability that we are willing to serve a demand for product a assigned to fulfillment center i, whereas we use the decision variable $u^a_{it}(q)$ to capture the probability of serving a demand assigned to fulfillment center i at time period t using the q-th unit of inventory at the fulfillment center. Jiang et al. (2023) propose the linear program

$$\max_{(\zeta^a_i,\boldsymbol{u}^a_i)\in\mathbb{R}_+^{1+T\times\widehat{z}^a_i}}\left\{\zeta^a_i\ :\ p^a_{it}\,\zeta^a_i\leq\sum_{q=1}^{\widehat{z}^a_i}u^a_{it}(q)\quad\forall\,t\in\mathcal{T},\ \ u^a_{it}(1)\leq p^a_{it}\left(1-\sum_{\ell=1}^{t-1}u^a_{i\ell}(1)\right)\quad\forall\,t\in\mathcal{T},\right.$$
$$\left.u^a_{it}(q)\leq p^a_{it}\left(\sum_{\ell=1}^{t-1}u^a_{i\ell}(q-1)-\sum_{\ell=1}^{t-1}u^a_{i\ell}(q)\right)\quad\forall\,t\in\mathcal{T},\ q=2,\ldots,\widehat{z}^a_i\right\}.\qquad(16)$$

In the linear program above, we use the vector $\boldsymbol{u}^a_i=(u^a_{it}(q):t\in\mathcal{T},\ q=1,\ldots,\widehat{z}^a_i)$. To give an interpretation for the constraints, by the definition of the decision variable $u^a_{it}(q)$, the right side of the first constraint corresponds to the probability of serving a demand for product a from fulfillment center i at time period t. Noting that p^a_{it} is the probability that a demand for product a at time period t is assigned to fulfillment center i and this demand is willing to accept the assigned promise, the left side of the first constraint is a lower bound on the probability that a demand for product a at time period t is assigned to fulfillment center i, we are willing to serve the demand and the demand is willing to accept the assigned promise, which is equivalent to being a lower bound on the probability of serving a demand for product a from fulfillment center i at time period t. Therefore, the first constraint ensures that the value of the decision variable ζ^a_i is a lower bound on the probability that we are willing to serve a demand for product a assigned to fulfillment center i under the approximate policy. On the right side of the third constraint, $\sum_{\ell=1}^{t-1}u^a_{i\ell}(q-1)$ is the probability that we use the $(q-1)$-th unit of inventory at fulfillment center i before time period t, which corresponds to the probability that we serve $q-1$ or more units of demand by the beginning of time period t. Similarly, $\sum_{\ell=1}^{t-1}u^a_{i\ell}(q)$ is the probability that that we serve q or more units of demand by the beginning of time period t. In this case, the difference on the right side of the third constraint corresponds to the probability that we serve $q-1$ units of demand by the beginning of time period t. Therefore, the third constraint ensures that the probability of serving the demand assigned to fulfillment center i at time period t using the q-th unit of inventory is at most the probability that we serve $q-1$ units of demand from fulfillment center i by the beginning of time period t, the demand for product a at time period t is assigned to fulfillment center i and the demand is willing to accept the assigned promise. The interpretation of the second constraint is similar to that of the third constraint, once we view the difference on the right side of the second constraint as the probability that we do not serve any demand from fulfillment center i by the beginning of time period t. The optimal objective value of the linear program in (16) gives the largest lower bound on the probability that we are willing to serve a request for product a assigned to fulfillment center i at any time period under our approximate policy. There is no closed form solution for this linear program, but Jiang et al. (2023) use a differential equation to compute a lower bound on the optimal objective value of (16) and show that this lower bound is at least $1-\frac{1}{\sqrt{\widehat{z}_{\min}+3}}$. To complete our discussion, we explain how we can use problem (16) to choose the tuning parameters $(\delta^a_{it}(q) : i \in \mathcal{F},\ t \in \mathcal{T},\ q = 1, \ldots, \widehat{z}^a_i)$ in our approximate policy.

Letting $(\widehat{\zeta}^a_i, \widehat{\boldsymbol{u}}^a_i)$ be an optimal solution to problem (16), by the discussion in the previous paragraph, $\sum_{\ell=1}^{t-1} \widehat{u}^a_{i\ell}(q-1) - \sum_{\ell=1}^{t-1} \widehat{u}^a_{i\ell}(q)$ is the probability that we serve $q-1$ units of demand for product a from fulfillment center i by the beginning of time period t, whereas $\widehat{u}^a_{it}(q)$ is the probability that we serve the demand for product a assigned to fulfillment center i at time period t using the q-th unit of inventory. To serve the demand at time period t using the q-th unit of inventory, we must have served $q-1$ units of demand by the beginning of time period t, the demand at time period t must be assigned to fulfillment center i, we must be willing to serve the demand and the demand must be willing to accept the assigned promise. Thus, the probability of being willing to serve a demand, which is given by $\delta^a_{it}(q)/p^a_{it}$, needs to satisfy $(\sum_{\ell=1}^{t-1} \widehat{u}^a_{i\ell}(q-1) - \sum_{\ell=1}^{t-1} \widehat{u}^a_{i\ell}(q)) \times p^a_{it}\,\frac{\delta^a_{it}(q)}{p^a_{it}} = \widehat{u}^a_{it}(q)$ for all $q = 2, \ldots, \widehat{z}^a_i$. By a similar argument, it follows that we also need to have $(1 - \sum_{\ell=1}^{t-1} u^a_{i\ell}(1))\, p^a_{it}\,\frac{\delta^a_{it}(1)}{p^a_{it}} = \widehat{u}^a_{it}(1)$. Thus, if we set the tuning parameters as $\delta^a_{it}(1) = \frac{\widehat{u}^a_{it}(1)}{1-\sum_{\ell=1}^{t-1} \widehat{u}^a_{i\ell}(1)}$ and $\delta^a_{it}(q) = \frac{\widehat{u}^a_{it}(q)}{\sum_{\ell=1}^{t-1} \widehat{u}^a_{i\ell}(q-1) - \sum_{\ell=1}^{t-1} \widehat{u}^a_{i\ell}(k)}$ for all $q = 2, \ldots, \widehat{z}^a_i$, then the value of $\widehat{u}^a_{it}(q)$ precisely corresponds to the probability that we serve the demand for product a assigned to fulfillment center i at time period t using the q-th unit of inventory under our approximate policy. In this case, by the first constraint in (16), the value of $\widehat{\zeta}^a_i$, which is at least $1 - \frac{1}{\sqrt{\widehat{z}_{\min}+3}}$, is a lower bound on the probability that we are willing to serve a demand for product a assigned to fulfillment center i.

Our approach for choosing the tuning parameters $(\delta^a_{it}(q) : i \in \mathcal{F},\ t \in \mathcal{T},\ q = 1, \ldots, \widehat{z}^a_i)$ as in the previous paragraph corresponds to Algorithm 4 in Jiang et al. (2023).

Appendix G: Performance Guarantee in a Fluid Regime

We consider an asymptotic regime where the numbers of units to be placed for all products, capacities of all fulfillment centers and total expected demands for all products from all demand regions are scaled with rate μ. In this asymptotic regime, we show that we can obtain a solution for the joint placement, delivery promise and fulfillment problem with a performance guarantee of $1-O\Big(\sqrt{\frac{\log\mu}{\mu}}\Big)$. We consider a sequence of joint placement, delivery promise and fulfillment problem instances $\{\mathcal{P}^\mu : \mu\in\mathbb{Z}_+\}$ indexed by the integer μ. In problem instance $\mathcal{P}^\mu$, the storage capacity of fulfillment center i is μU_i. We have μC^a units of product a to place over all fulfillment centers. There are μT time periods in the selling horizon indexed $\mathcal{T}^\mu=\{1,\ldots,\mu T\}$. At time period t, we have a demand for product a from demand region j with probability $\lambda^a_{j,\lceil t/\mu\rceil}$. Therefore, each block of consecutive μ time periods in problem instance $\mathcal{P}^\mu$ involve identical demand arrival probabilities, so the total expected demand for product a from demand region j is $\sum_{t=1}^{\mu T}\lambda^a_{j,\lceil t/\mu\rceil}=\mu\sum_{t=1}^{T}\lambda^a_{jt}$. All other problem parameters in problem instance $\mathcal{P}^\mu$, including the profits associated with serving demands for different products from different demand regions through different fulfillment centers and delivery promise acceptance probabilities, are the same as those given in Section 2. For problem instance $\mathcal{P}^\mu$, we use OPT^μ to denote the optimal total expected profit for the joint placement, delivery promise and fulfillment problem. Let $(\widehat{\boldsymbol{z}},\widehat{\boldsymbol{w}})$ be an optimal solution to linear program

$$Z^* = \max_{(\boldsymbol{z},\boldsymbol{w})\in\mathbb{R}_+^{n\times m\times(1+|\mathcal{D}|\times|\mathcal{K}|)}} \Bigg\{\sum_{a\in\mathcal{A}}\sum_{i\in\mathcal{F}}\sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}} r^a_{ijk}\,\theta^a_{jk}\,w^a_{ijk} \;:\; \sum_{i\in\mathcal{F}} z^a_i\le C^a \quad \forall a\in\mathcal{A},\;\; \sum_{a\in\mathcal{A}} z^a_i\le U_i \quad \forall i\in\mathcal{F},$$
$$\sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}}\theta^a_{jk}\,w^a_{ijk}\le z^a_i \quad \forall i\in\mathcal{F},\; a\in\mathcal{A},\;\; \sum_{i\in\mathcal{F}}\sum_{k\in\mathcal{K}} w^a_{ijk}\le\sum_{t\in\mathcal{T}}\lambda^a_{jt} \quad \forall j\in\mathcal{D},\; a\in\mathcal{A}\Bigg\}. \tag{17}$$

By the discussion that follows the Relaxed problem, the problem above is the continuous relaxation for the Approximate Placement problem with the bounding function f^a_{LP} for problem instance $\mathcal{P}^1$. Noting that OPT^1 is the optimal total expected profit for problem instance $\mathcal{P}^1$, we have $Z^*\ge\mathsf{OPT}^1$. On the other hand, consider the continuous relaxation for the Approximate Placement problem with the bounding function f^a_{LP} for problem instance $\mathcal{P}^\mu$. We can obtain this continuous relaxation from problem (17) by replacing the right side of the first, second and fourth constraints, respectively, with μC^a, μU_i and $\sum_{t=1}^{\mu T}\lambda^a_{j,\lceil t/\mu\rceil}$. Thus, noting that $\sum_{t=1}^{\mu T}\lambda^a_{j,\lceil t/\mu\rceil}=\mu\sum_{t\in\mathcal{T}}\lambda^a_{jt}$, it follows that μZ^* is the optimal objective value of the continuous relaxation for the Approximate Placement problem for problem instance P^μ, in which case, we obtain $\mu Z^*\ge\mathsf{OPT}^\mu$.

Approximation Strategy and Performance Guarantee:

We use the following approach to obtain an approximate solution to the joint placement, delivery promise and fulfillment problem for problem instance $\mathcal{P}^\mu$. Recalling that $(\widehat{\boldsymbol{z}},\widehat{\boldsymbol{w}})$ is an optimal solution to problem (17), at the beginning of the selling horizon, we place $\lfloor \mu\, \widehat{z}_i^a \rfloor$ units of product a at fulfillment center i. We refer to these placement quantities as the rounded placement quantities. Letting $\gamma \in (0,1)$ be a tuning parameter, over the selling horizon, if we have a demand for product a from demand region j, then we sample the fulfillment center and promise pair (i,k) with probability $\gamma\, \widehat{w}_{ijk}^a / \sum_{t\in\mathcal{T}} \lambda_{jt}^a$. If the sampled fulfillment center i does not have any remaining inventory for product a, then we offer the dummy promise. Otherwise, we offer the sampled promise k and use the sampled fulfillment center i upon acceptance of the offered promise. We refer to this delivery promise and fulfillment policy as the randomized policy. We shortly specify the value of the tuning parameter. In the next theorem, we give a performance guarantee for problem instance $\mathcal{P}^\mu$ by using the rounded placement quantities at the beginning of the selling horizon, along with following the randomized delivery promise and fulfillment policy later on.

Theorem G.1 (Performance in Fluid Regime) *For problem instance $\mathcal{P}^\mu$, there exists a choice of the tuning parameter $\gamma \in (0,1)$ such that if we make the rounded placement decisions and use the randomized delivery promise and fulfillment policy with this tuning parameter, then we obtain a total expected profit of at least $\left(1 - O\left(\sqrt{\frac{\log \mu}{\mu}}\right)\right) \mathsf{OPT}^\mu$.*

We introduce some notation to give a proof for Theorem G.1. We define three random variables. The Bernoulli random variable N_{it}^a takes value one if the randomized policy assigns a demand for product a at time period t to fulfillment center i along with some delivery promise and the customer accepts the delivery promise. We have $\mathbb{E}\{N_{it}^a\} = \sum_{j\in\mathcal{D}} \sum_{k\in\mathcal{K}} \lambda_{j,\lceil t/\mu \rceil}^a\, \gamma\, \frac{\widehat{w}_{ijk}^a}{\sum_{\ell\in\mathcal{T}} \lambda_{j\ell}^a}\, \theta_{jk}^a$. The Bernoulli random variable G_{it}^a takes value one if there is remaining inventory for product a at fulfillment center i at time period t under the randomized policy. Note that $G_{it}^a = 1$ if and only if $\sum_{\ell=1}^{t-1} N_{i\ell}^a < \lfloor \mu\, \widehat{z}_i^a \rfloor$. Lastly, the random variable R_{it}^a corresponds to the profit that the randomized policy obtains at time period t from demand for product a it assigns to fulfillment center i. We have $\mathbb{E}\{R_{it}^a \,|\, G_{it}^a = 1\} = \sum_{j\in\mathcal{D}} \sum_{k\in\mathcal{K}} r_{ijk}^a\, \lambda_{j,\lceil t/\mu \rceil}^a\, \gamma\, \frac{\widehat{w}_{ijk}^a}{\sum_{\ell\in\mathcal{T}} \lambda_{j\ell}^a}\, \theta_{jk}^a$, where we use the fact that to realize the profit R_{it}^a, a demand must occur, it must be assigned to fulfillment center i and the promise must be accepted. We have $\mathbb{E}\{R_{it}^a \,|\, G_{it}^a = 0\} = 0$. The total expected profit of the randomized policy is

$$\begin{aligned}
\mathsf{APX} \;=\; \sum_{a\in\mathcal{A}} \sum_{t=1}^{\mu T} \sum_{i\in\mathcal{F}} \mathbb{E}\{R_{it}^a\} \;&=\; \sum_{a\in\mathcal{A}} \sum_{t=1}^{\mu T} \sum_{i\in\mathcal{F}} \mathbb{E}\{R_{it}^a \,|\, G_{it}^a = 1\}\, \mathbb{P}\{G_{it}^a = 1\} \\
&=\; \sum_{a\in\mathcal{A}} \sum_{t=1}^{\mu T} \sum_{i\in\mathcal{F}} \sum_{j\in\mathcal{D}} \sum_{k\in\mathcal{K}} r_{ijk}^a\, \lambda_{j,\lceil t/\mu \rceil}^a\, \gamma\, \frac{\widehat{w}_{ijk}^a}{\sum_{\ell\in\mathcal{T}} \lambda_{j\ell}^a}\, \theta_{jk}^a\, \mathbb{P}\{G_{it}^a = 1\}. \qquad (18)
\end{aligned}$$

Setting $\epsilon = \min_{i\in\mathcal{F}, a\in\mathcal{A}} \{\widehat{z}_i^a : \widehat{z}_i^a > 0\}$ to capture the smallest nonzero and possibly fractional placement quantity in $\widehat{\boldsymbol{z}}$, we lower bound the availability probability $\mathbb{P}\{G_{it}^a = 1\}$.

Lemma G.2 (Availability Probability) *For any choice of the tuning parameter* $\gamma \in (0, 1 - \frac{1}{\mu\epsilon})$, *if we have* $\widehat{z}_i^a > 0$, *then the availability probability for problem instance* $\mathcal{P}^\mu$ *satisfies*

$$\mathbb{P}\{G_{it}^a = 1\} \geq 1 - \exp\left(-\frac{1}{2}\Big(1 - \frac{1}{\mu\epsilon} - \gamma\Big)^2 \mu\epsilon \right).$$

Proof: Noting the definition of N_{it}^a, we have $G_{it}^a = 1$ if and only if $\sum_{\ell=1}^{t-1} N_{i\ell}^a < \lfloor \mu\, \widehat{z}_i^a \rfloor$. Therefore, we have $\mathbb{P}\{G_{it}^a = 1\} \geq \mathbb{P}\{\sum_{t=1}^{\mu T} N_{it}^a < \lfloor \mu\, \widehat{z}_i^a \rfloor\}$. By the discussion just after Theorem G.1, we have $\sum_{t=1}^{\mu T} \mathbb{E}\{N_{it}^a\} = \sum_{t=1}^{\mu T} \sum_{j \in \mathcal{D}} \sum_{k \in \mathcal{K}} \lambda_{j,\lceil t/\mu \rceil}^a\, \gamma\, \frac{\widehat{w}_{ijk}^a}{\sum_{\ell \in \mathcal{T}} \lambda_{j\ell}^a}\, \theta_{jk}^a = \gamma\, \mu \sum_{j \in \mathcal{D}} \sum_{k \in \mathcal{K}} \theta_{jk}^a\, \widehat{w}_{ijk}^a \leq \gamma\, \mu\, \widehat{z}_i^a$, where the second equality uses the fact that $\sum_{t=1}^{\mu T} \lambda_{j,\lceil t/\mu \rceil}^a = \mu \sum_{t \in \mathcal{T}} \lambda_{jt}^a$ and the inequality holds because $(\widehat{\boldsymbol{z}}, \widehat{\boldsymbol{w}})$ satisfies the third constraint in problem (17). Note that N_{it}^a is a Bernoulli random variable. Because the variance of a Bernoulli random variable is upper bounded by its expectation, by the last chain of inequalities, we have $\sum_{t=1}^{\mu T} \mathrm{Var}(N_{it}^a) \leq \gamma\, \mu\, \widehat{z}_i^a$ as well. Therefore, letting $\overline{N}_i^a = \sum_{t=1}^{\mu T} N_{it}^a$ for notational brevity, we have $\mathbb{P}\{G_{it}^a = 1\} \geq \mathbb{P}\{\overline{N}_i^a < \lfloor \mu \widehat{z}_i^a \rfloor\}$, $\mathbb{E}\{\overline{N}_i^a\} \leq \gamma\, \mu\, \widehat{z}_i^a$ and $\mathrm{Var}(\overline{N}_i^a) \leq \gamma\, \mu\, \widehat{z}_i^a$. In this case, we obtain the chain of inequalities

$$\begin{aligned}
\mathbb{P}\Big\{G_{it}^a = 0\Big\} &\leq \mathbb{P}\Big\{\overline{N}_i^a \geq \lfloor \mu\, \widehat{z}_i^a \rfloor\Big\} \leq \mathbb{P}\Big\{\overline{N}_i^a \geq \mu\, \widehat{z}_i^a - 1\Big\} \overset{(a)}{\leq} \mathbb{P}\Big\{\overline{N}_i^a - \mathbb{E}\{\overline{N}_i^a\} \geq \mu\, \widehat{z}_i^a - 1 - \gamma\, \mu\, \widehat{z}_i^a\Big\} \\
&\overset{(b)}{\leq} \mathbb{P}\Big\{\overline{N}_i^a - \mathbb{E}\{\overline{N}_i^a\} \geq \mu\, \widehat{z}_i^a - \frac{\widehat{z}_i^a}{\epsilon} - \gamma\, \mu\, \widehat{z}_i^a\Big\} = \mathbb{P}\Big\{\overline{N}_i^a - \mathbb{E}\{\overline{N}_i^a\} \geq \mu\, \widehat{z}_i^a \Big(1 - \frac{1}{\mu\epsilon} - \gamma\Big)\Big\} \\
&\overset{(c)}{\leq} \exp\left(- \frac{\frac{1}{2}\Big(1 - \frac{1}{\mu\epsilon} - \gamma\Big)^2 (\mu\, \widehat{z}_i^a)^2}{\mathrm{Var}(\overline{N}_i^a) + \frac{1}{3}\Big(1 - \frac{1}{\mu\epsilon} - \gamma\Big)\mu\, \widehat{z}_i^a} \right) \overset{(d)}{\leq} \exp\left(- \frac{\frac{1}{2}\Big(1 - \frac{1}{\mu\epsilon} - \gamma\Big)^2 (\mu\, \widehat{z}_i^a)^2}{\gamma\, \mu\, \widehat{z}_i^a + \frac{1}{3}\Big(1 - \frac{1}{\mu\epsilon} - \gamma\Big)\mu\, \widehat{z}_i^a} \right) \\
&= \exp\left(- \frac{\frac{1}{2}\Big(1 - \frac{1}{\mu\epsilon} - \gamma\Big)^2 \mu\, \widehat{z}_i^a}{\gamma + \frac{1}{3}\Big(1 - \frac{1}{\mu\epsilon} - \gamma\Big)} \right) = \exp\left(- \frac{\frac{3}{2}\Big(1 - \frac{1}{\mu\epsilon} - \gamma\Big)^2 \mu\, \widehat{z}_i^a}{1 + 2\,\gamma - \frac{1}{\mu\epsilon}} \right) \\
&\overset{(e)}{\leq} \exp\left(-\frac{1}{2}\Big(1 - \frac{1}{\mu\epsilon} - \gamma\Big)^2 \mu\epsilon \right),
\end{aligned}$$

where (a) uses $\mathbb{E}\{\overline{N}_i^a\} \leq \gamma\, \mu\, \widehat{z}_i^a$, (b) is by $\widehat{z}_i^a / \epsilon \geq 1$ by the definition of ϵ, (c) is the one-sided Bernstein inequality, (d) uses $\mathrm{Var}(\overline{N}_i^a) \leq \gamma\, \mu\, \widehat{z}_i^a$ and (e) holds because $1 + 2\gamma - \frac{1}{\mu\epsilon} \leq 3$ and $\widehat{z}_i^a \geq \epsilon$. ■

By the lemma above, if the scaling factor μ is large enough so that $\mu > 1/\epsilon$, then we can lower bound the availability probability $\mathbb{P}\{G_{it}^a = 1\}$. Using this lower bound on the availability probability in (18) yields a proof for Theorem G.1, as given below.

Proof of Theorem G.1:

We proceed with the understanding that the scaling factor μ is large enough that $\mu \geq 6/\epsilon$, because it is enough to show the result for a large value of the scaling factor. We choose the tuning parameter as $\gamma = 1 - \sqrt{\frac{2\log(\mu\epsilon)}{\mu\epsilon}} - \frac{1}{\mu\epsilon}$. It is simple to check that if $\mu \geq 6/\epsilon$, then $\gamma \geq 0$. With this choice of γ, the right side of the inequality in Lemma G.2 evaluates to $1 - \frac{1}{\mu\epsilon}$, so (18) yields

$$\begin{aligned}
\mathsf{APX} &\geq \Big(1 - \frac{1}{\mu\epsilon}\Big) \sum_{a\in\mathcal{A}} \sum_{t=1}^{\mu T} \sum_{i\in\mathcal{F}} \sum_{j\in\mathcal{D}} \sum_{k\in\mathcal{K}} r^a_{ijk}\, \lambda^a_{j,\lceil t/\mu\rceil}\, \gamma\, \frac{\widehat{w}^a_{ijk}}{\sum_{\ell\in\mathcal{T}} \lambda^a_{j\ell}}\, \theta^a_{jk} \\
&\overset{(a)}{=} \Big(1 - \frac{1}{\mu\epsilon}\Big)\gamma\,\mu \sum_{a\in\mathcal{A}} \sum_{i\in\mathcal{F}} \sum_{j\in\mathcal{D}} \sum_{k\in\mathcal{K}} r^a_{ijk}\, \theta^a_{jk}\, \widehat{w}^a_{ijk} \overset{(b)}{=} \Big(1 - \frac{1}{\mu\epsilon}\Big) \left(1 - \sqrt{\frac{2\log(\mu\epsilon)}{\mu\epsilon}} - \frac{1}{\mu\epsilon}\right) \mu Z^* \\
&\overset{(c)}{\geq} \Big(1 - \frac{1}{\mu\epsilon}\Big) \left(1 - \sqrt{\frac{2\log(\mu\epsilon)}{\mu\epsilon}} - \frac{1}{\mu\epsilon}\right) \mathsf{OPT}^\mu \geq \left(1 - \sqrt{\frac{2\log(\mu\epsilon)}{\mu\epsilon}} - \frac{2}{\mu\epsilon}\right) \mathsf{OPT}^\mu \\
&= \left(1 - O\left(\sqrt{\frac{\log\mu}{\mu}}\right)\right) \mathsf{OPT}^\mu,
\end{aligned}$$

where (a) uses $\sum_{t=1}^{\mu T} \lambda^a_{j,\lceil t/\mu\rceil} = \mu \sum_{t\in\mathcal{T}} \lambda^a_{jt}$, (b) holds because $(\widehat{\boldsymbol{z}}, \widehat{\boldsymbol{w}})$ is an optimal solution to problem (17) and (c) holds because $\mu Z^* \geq \mathsf{OPT}^\mu$ by the discussion at the beginning of this section. ■

Appendix H: Expected Recourse Function for Stochastic Programs Over Trees

Using the decision variables $\boldsymbol{y} = (y_j : j = 1, \ldots, n)$, nonnegative cost coefficients $(\theta_j : j = 1, \ldots, n)$ and random variables $(N_j : j = 1, \ldots, n)$ taking nonnegative integer values, letting

$$\pi(K) = \mathbb{E}\left\{ \max_{\boldsymbol{y}\in\mathbb{Z}^n_+} \left\{ \sum_{j=1}^n \theta_j\, y_j \;:\; \sum_{j=1}^n y_j \leq K, \quad y_j \leq N_j \quad \forall\, j = 1, \ldots, n \right\}\right\},$$

we consider computing $\pi(K)$ as a function of $K \in \mathbb{Z}_+$. The function $\Pi^a_i(z^a_i)$ used in the benchmarks UPG and UPR is computed through an expectation of the same form. The maximization problem above is a knapsack problem with n items, where we have N_j copies of item j, we obtain a utility of θ_j for each copy of item j that we fit into the knapsack, each copy of an item takes up one unit of capacity and we have K units of knapsack capacity. Thus, we can solve the maximization problem above by filling the knapsack starting from the item with the largest utility. We index the items such that $\theta_1 \geq \theta_2 \geq \ldots \geq \theta_n$. For fixed realizations of the random variables $(N_j : j = 1, \ldots, n)$, if $k \in \mathbb{Z}_+$ satisfies $\sum_{\ell=1}^{j-1} N_\ell < k \leq \sum_{\ell=1}^{j} N_\ell$, then the utility that we obtain from the k-th unit of capacity in the knapsack is θ_j. In particular, the copies of items $1, \ldots, j-1$ occupy strictly less than k units of capacity, whereas the copies of items $1, \ldots, j$ occupy at least k units of capacity, so the k-th unit of capacity must be occupied by a copy of item j. We express the utility that we obtain from the k-th unit of capacity in the knapsack as $\sum_{j=1}^n \theta_j\, \mathbb{1}_{\{\sum_{\ell=1}^{j-1} N_\ell < k \leq \sum_{\ell=1}^{j} N_\ell\}}$. Thus, for fixed realizations of the random variables $(N_j : j = 1, \ldots, n)$, the optimal objective value of the maximization problem is $\sum_{k=1}^K \sum_{j=1}^n \theta_j\, \mathbb{1}_{\{\sum_{\ell=1}^{j-1} N_\ell < k \leq \sum_{\ell=1}^{j} N_\ell\}}$. Taking the expectation of the last expression and noting that this expectation yields $\pi(K)$, we get $\pi(K) = \sum_{k=1}^K \sum_{j=1}^n \theta_j\, \mathbb{P}\{\sum_{\ell=1}^{j-1} N_\ell < k \leq \sum_{\ell=1}^{j} N_\ell\}$. So, we have a closed form expression for $\pi(K)$ as long as we have access to the convolutions of $(N_j : j = 1, \ldots, n)$. In our computational experiments, we use simulation to approximate these convolutions.

Appendix I: Placement with a Separable and Concave Approximation

Consider the case where $f^a(\boldsymbol{z}^a)$ is given by a separable function of the form $\sum_{i\in\mathcal{F}} \Pi_i^a(z_i^a)$ and $\Pi_i^a(z_i^a)$ is a piecewise linear and concave function of z_i^a with points of nondifferentiability at integers. In this case, we demonstrate that we can solve the Approximate Placement problem as a min-cost network flow problem. Letting $C_{\max} = \max_{a\in\mathcal{A}} C^a$, noting that $\Pi_i^a(z_i^a)$ is concave in z_i^a, we have $\Pi_i^a(\ell) - \Pi_i^a(\ell-1) \geq \Pi_i^a(\ell+1) - \Pi_i^a(\ell)$ for all $\ell = 2,\ldots,C_{\max}-1$. Let $\eta_{i\ell}^a = \Pi_i^a(\ell) - \Pi_i^a(\ell-1)$ for each $\ell = 1,\ldots,C_{\max}$, so we have $\Pi_i^a(z_i^a) = \sum_{\ell=1}^{z_i^a} \eta_{i\ell}^a$. In this case, using the vector of decision variables $\boldsymbol{y} = (y_{i\ell}^a : i\in\mathcal{F},\ \ell = 1,\ldots,C_{\max},\ a\in\mathcal{A}) \in \{0,1\}^{n\times C_{\max}\times m}$, if we set $f^a = \sum_{i\in\mathcal{F}} \Pi_i^a$ in the Approximate Placement problem, then this problem is equivalent to

$$\max_{\boldsymbol{z}\in\mathbb{Z}_+^{n\times m}} \left\{ \sum_{a\in\mathcal{A}} \sum_{i\in\mathcal{F}} \Pi_i^a(z_i^a) : \sum_{i\in\mathcal{F}} z_i^a = C^a \quad \forall\, a\in\mathcal{A}, \quad \sum_{a\in\mathcal{A}} z_i^a \leq U_i \quad \forall\, i\in\mathcal{F} \right\}$$
$$= \max_{\boldsymbol{y}\in\{0,1\}^{n\times C_{\max}\times m}} \left\{ \sum_{a\in\mathcal{A}} \sum_{i\in\mathcal{F}} \sum_{\ell=1}^{C_{\max}} \eta_{i\ell}^a\, y_{i\ell}^a : \sum_{i\in\mathcal{F}} \sum_{\ell=1}^{C_{\max}} y_{i\ell}^a = C^a \quad \forall\, a\in\mathcal{A}, \ \sum_{a\in\mathcal{A}} \sum_{\ell=1}^{C_{\max}} y_{i\ell}^a \leq U_i \quad \forall\, i\in\mathcal{F} \right\},$$

where the equality follows by using the change of variables $z_i^a = \sum_{\ell=1}^{C_{\max}} y_{i\ell}^a$ and noting that we can compute $\Pi_i^a(z_i^a)$ as $\sum_{\ell=1}^{C_{\max}} \eta_{i\ell}^a\, y_{i\ell}^a$. The problem above is a transportation problem on a bipartite graph. The supply vertices are indexed by $\mathcal{F}$ and the demand vertices are indexed by $\mathcal{A}$. The supply at vertex $i\in\mathcal{F}$ is U_i. The demand at vertex $a\in\mathcal{A}$ is C^a. The edge corresponding to the decision variable $y_{i\ell}^a$ connects vertex $i\in\mathcal{F}$ to vertex $a\in\mathcal{A}$. The edges corresponding to the decision variables $(y_{i\ell}^a : \ell = 1,\ldots,C_{\max})$ all connect vertex $i\in\mathcal{F}$ to vertex $a\in\mathcal{A}$. In the problem above, we find the flows over the edges to maximize the total contribution, while serving the demand at the demand vertices and not violating the supply at the supply vertices.

The preceding discussion holds when we replace $f^a(\boldsymbol{z}^a)$ with any $\sum_{i\in\mathcal{F}} g_i^a(z_i^a)$, where $g_i^a(z_i^a)$ is piecewise linear and concave in z_i^a with points of nondifferentiability at integers.

Appendix J: Benchmark Strategy Based on Lagrangian Relaxation

We give the details of the benchmark strategy that is based on using Lagrangian relaxation on the Fulfillment dynamic program. Our starting point is an alternative representation of the Fulfillment dynamic program. To equivalently express the maximization problem on the right side of this dynamic program, we use the decision variables $\boldsymbol{y}^a = (y_{ijk}^a : i\in\mathcal{F},\ j\in\mathcal{D}\ , k\in\mathcal{K}) \in \{0,1\}^{n\times|\mathcal{D}|\times|\mathcal{K}}$, where $y_{ijk}^a = 1$ if and only if we offer delivery promise k to a demand for product a from demand region j, we plan to use fulfillment center i upon acceptance of the promise and there is inventory for product a at fulfillment center i to serve the demand. In other words, we have $y_{ijk}^a = 1$ if and only if $(i,k)\in\mathcal{F}\times\mathcal{K}$ is an optimal solution to the maximization problem on the right side of the Fulfillment dynamic program for demand region j and $\mathbb{1}_{(x_i^a \geq 1)} = 1$. In this case, we express this dynamic program equivalently as

$$J_t^a(\boldsymbol{x}^a) = \max_{\boldsymbol{y}^a \in \{0,1\}^{n \times |\mathcal{D}| \times |\mathcal{K}}} \Bigg\{ \sum_{j\in\mathcal{D}} \sum_{i\in\mathcal{F}} \sum_{k\in\mathcal{K}} \lambda_{jt}^a \, y_{ijk}^a \, \theta_{jk}^a \Big[r_{ijk}^a + J_{t+1}^a(\boldsymbol{x}^a - \boldsymbol{e}_i) - J_{t+1}^a(\boldsymbol{x}^a) \Big] + J_{t+1}(\boldsymbol{x}^a) \ : \sum_{i\in\mathcal{F}} \sum_{k\in\mathcal{K}} y_{ijk}^a \leq 1 \ \ \forall\, j \in \mathcal{D}, \quad y_{ijk}^a \leq x_i^a \ \ \forall\, i \in \mathcal{F},\ j \in \mathcal{D},\ k \in \mathcal{K} \Bigg\}. \tag{19}$$

To see the equivalence of (19) to the Fulfillment dynamic program, for each $j \in \mathcal{D}$, if $(\ell, q) \in \mathcal{F} \times \mathcal{K}$ is an optimal solution to the maximization problem in the Fulfillment dynamic program and $x_\ell^a \geq 1$, then setting $y_{\ell j q}^a = 1$ and setting $y_{ijk}^a = 0$ for all $(i,k) \neq (\ell, q)$ gives a feasible solution to the maximization problem in (19) so that the objective values of the two problems match. Similarly, if $(\ell, q) \in \mathcal{F} \times \mathcal{K}$ is an optimal solution to the maximization problem in the Fulfillment dynamic program and $x_\ell^a = 0$, then we can set $y_{ijk}^a = 0$ for all $i \in \mathcal{F}$ and $k \in \mathcal{K}$. On the other hand, for each $j \in \mathcal{D}$, if $y_{\ell j q}^a = 1$ for some $(\ell, q) \in \mathcal{F} \times \mathcal{K}$ in an optimal solution to the maximization problem in (19), then using the solution (ℓ, q) gives a feasible solution to the maximization problem in the Fulfillment dynamic program so that the objective values of the two problems match. Lastly, if $y_{ijk}^a = 0$ for all $i \in \mathcal{F}$ and $k \in \mathcal{K}$, then we can use (i, κ) for any $i \in \mathcal{F}$ in the Fulfillment dynamic program, where we recall that κ is the dummy promise. To obtain an upper bound on the optimal total expected profit, we associate the Lagrange multipliers $\boldsymbol{\beta}^a = (\beta_j^a : j \in \mathcal{D})$ with the first set of constraints in (19) and relax these constraints, in which case, we obtain the relaxed dynamic program

$$\widetilde{J}_t^a(\boldsymbol{x}^a; \boldsymbol{\beta}^a) = \max_{\boldsymbol{y}^a \in \{0,1\}^{n \times |\mathcal{D}| \times |\mathcal{K}}} \Bigg\{ \sum_{j\in\mathcal{D}} \sum_{i\in\mathcal{F}} \sum_{k\in\mathcal{K}} \lambda_{jt}^a \, y_{ijk}^a \Big\{ \theta_{jk}^a \Big[r_{ijk}^a + \widetilde{J}_{t+1}^a(\boldsymbol{x}^a - \boldsymbol{e}_i; \boldsymbol{\beta}^a) - \widetilde{J}_{t+1}^a(\boldsymbol{x}^a; \boldsymbol{\beta}^a) \Big] - \beta_j^a \Big\} + \sum_{j\in\mathcal{D}} \lambda_{jt}^a \, \beta_j^a + \widetilde{J}_{t+1}^a(\boldsymbol{x}^a; \boldsymbol{\beta}^a) \ : \sum_{k\in\mathcal{K}} y_{ijk}^a \leq 1 \ \ \forall\, i \in \mathcal{F},\ j \in \mathcal{D}, \ \ y_{ijk}^a \leq x_i^a \ \ \forall\, i \in \mathcal{F},\ j \in \mathcal{D},\ k \in \mathcal{K} \Bigg\}, \tag{20}$$

where we scaled β_j^a with λ_{jt}^a in (20). The first constraint in (20) does not appear in (19), but this constraint would be redundant in (19) because of the constraint $\sum_{i\in\mathcal{F}} \sum_{j\in\mathcal{D}} y_{ijk}^a \leq 1$.

We can show two results for the value functions in the dynamic program in (20). Analogues of these two results appear in the literature, so we give the results without proofs but point out references. First, for any set of nonnegative Lagrange multipliers, the value functions in (20) are upper bounds on those in (19). Thus, if $\beta_j^a \geq 0$ for all $j \in \mathcal{D}$, then we have $\widetilde{J}_t^a(\boldsymbol{x}^a; \boldsymbol{\beta}^a) \geq J_t^a(\boldsymbol{x}^a)$ for each $\boldsymbol{x}^a \in \mathbb{Z}_+^n$. We can show this result by using induction over the time periods. An analogue of this result appears, for example, in Proposition 2 in Topaloglu (2009). Second, the dynamic program in (20) decomposes by the fulfillment centers, where the dynamic program that we solve for each fulfillment center has a scalar state variable. In particular, using the vector of decision variables $\boldsymbol{y}_i^a = (y_{ijk}^a : j \in \mathcal{D}\ , k \in \mathcal{K}) \in \{0,1\}^{|\mathcal{D}|\times|\mathcal{K}}$, consider the dynamic program

$$\begin{aligned}\widetilde{J}_{it}^a(x_i^a;\boldsymbol{\beta}^a) \quad &= \max_{\boldsymbol{y}_i^a \in \{0,1\}^{|\mathcal{D}|\times|\mathcal{K}}} \Bigg\{ \sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}} \lambda_{jt}^a\, y_{ijk}^a \Big\{ \theta_{jk}^a \Big[r_{ijk}^a + \widetilde{J}_{i,t+1}^a(x_i^a - 1;\boldsymbol{\beta}^a) - \widetilde{J}_{i,t+1}^a(x_i^a;\boldsymbol{\beta}^a)\Big] - \beta_j^a \Big\} \\ &\qquad + \widetilde{J}_{i,t+1}^a(x_i^a;\boldsymbol{\beta}^a) \ : \\ &\qquad \sum_{k\in\mathcal{K}} y_{ijk}^a \le 1 \ \ \forall\, j\in\mathcal{D}, \quad y_{ijk}^a \le x_i^a \ \ \forall\, j \in\mathcal{D},\ k\in\mathcal{K} \Bigg\}. \end{aligned} \tag{21}$$

The value functions in (20) and (21) satisfy $\widetilde{J}_t^a(\boldsymbol{x}^a;\boldsymbol{\beta}^a) = \sum_{i\in\mathcal{F}} \widetilde{J}_{it}^a(x_i^a;\boldsymbol{\beta}^a) + \sum_{j\in\mathcal{D}}\sum_{s=t}^{T} \lambda_{js}^a\,\beta_j^a$. We can show this result by using induction over the time periods as well. An analogue of this result appears in Proposition 1 in Topaloglu (2009). Thus, we can solve the dynamic program in (20) efficiently through the dynamic program in (21). By the two results in this paragraph, for any nonnegative Lagrange multipliers $\boldsymbol{\beta}^a$, if the placement decisions for product a at the beginning of the selling horizon is $\boldsymbol{z}^a$, then the optimal total expected profit from this product is upper bounded by $\sum_{i\in\mathcal{F}} \widetilde{J}_{i1}^a(z_i^a;\boldsymbol{\beta}^a) + \sum_{t\in\mathcal{T}}\sum_{j\in\mathcal{D}} \lambda_{jt}^a\,\beta_j^a$. Thus, as long as $\beta_j^a \ge 0$ for all $a\in\mathcal{A}$ and $j\in\mathcal{D}$, we can replace $f^a(\boldsymbol{z}^a)$ in the objective function of the Approximate Placement problem with $\sum_{i\in\mathcal{F}} \widetilde{J}_{i1}^a(z_i^a;\boldsymbol{\beta}^a) + \sum_{t\in\mathcal{T}}\sum_{j\in\mathcal{D}} \lambda_{jt}^a\,\beta_j^a$ and solve this problem to obtain an upper bound on the optimal objective value of the Placement problem. In the rest of this section, we focus on choosing the Lagrange multipliers. Our choice of Lagrange multipliers will ensure that $\sum_{i\in\mathcal{F}} \widetilde{J}_{i1}^a(z_i^a;\boldsymbol{\beta}^a) + \sum_{t\in\mathcal{T}}\sum_{j\in\mathcal{D}} \lambda_{jt}^a\,\beta_j^a \le f_{\mathsf{LP}}^a(\boldsymbol{z}^a)$ for all $\boldsymbol{z}^a \in \mathbb{Z}_+^n$. Thus, by using our choice of the Lagrange multipliers, we can obtain an upper bound on the optimal objective value of the Placement problem that is at least as tight as the one provided by using our bounding functions.

Choosing the Lagrange Multipliers: Considering the Bounding problem, we relax the second constraint in this problem through the Lagrange multipliers $\boldsymbol{\beta}^a$ to obtain the problem

$$f_{\mathsf{LP}}^a(\boldsymbol{z}^a;\boldsymbol{\beta}^a) = \max_{\boldsymbol{w}^a \in \mathbb{R}_+^{n\times|\mathcal{D}|\times|\mathcal{K}|}} \Bigg\{ \sum_{i\in\mathcal{F}}\sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}} (r_{ijk}^a\,\theta_{jk}^a - \beta_j^a)\, w_{ijk}^a \ : \ \sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}} \theta_{jk}^a\, w_{ijk}^a \le z_i^a \quad \forall\, i\in\mathcal{F} \Bigg\}. \tag{22}$$

By strong duality, we have $f_{\mathsf{LP}}^a(\boldsymbol{z}^a) = \min_{\boldsymbol{\beta}^a \ge \boldsymbol{0}} \big\{ f_{\mathsf{LP}}^a(\boldsymbol{z}^a;\boldsymbol{\beta}^a) + \sum_{t\in\mathcal{T}}\sum_{j\in\mathcal{D}} \lambda_{jt}^a\,\beta_j^a \big\}$. In the next proposition, we compare the objective function of the last problem with $\widetilde{J}_1^a(\boldsymbol{z}^a;\boldsymbol{\beta}^a)$.

Proposition J.1 *For any $\boldsymbol{\beta}^a \ge \boldsymbol{0}$, the value functions computed through the dynamic program in (20) satisfies $f_{\mathsf{LP}}^a(\boldsymbol{z}^a;\boldsymbol{\beta}^a) + \sum_{t\in\mathcal{T}}\sum_{j\in\mathcal{D}} \lambda_{jt}^a\,\beta_j^a \ge \widetilde{J}_1^a(\boldsymbol{z}^a;\boldsymbol{\beta}^a)$ for all $\boldsymbol{z}^a \in \mathbb{Z}_+^n$ and $t\in\mathcal{T}$.*

Proof: Fix the Lagrange multipliers $\boldsymbol{\beta}^a$. For the fixed Lagrange multipliers, let $\boldsymbol{\alpha}^a = (\alpha_i^a : i \in\mathcal{F})$ be the optimal values of the dual variables associated with the constraint in problem (22). By the constraint in the dual of problem (22), we have $\theta^a_{jk}\,\alpha^a_i \geq r^a_{ijk}\,\theta^a_{jk} - \beta^a_j$ for all $i\in\mathcal{F}$, $j\in\mathcal{D}$ and $k\in\mathcal{K}$. We use induction over the time periods to show that $\alpha^a_i\,x^a_i \geq \widetilde{J}^a_{it}(x^a_i;\boldsymbol{\beta}^a)$ for all $x^a_i\in\mathbb{Z}_+$ and $t\in\mathcal{T}$. Since α^a_i is the optimal value of the dual variable associated with an inequality constraint in problem (22), we have $\alpha^a_i\geq 0$. Thus, $\alpha^a_i\,x^a_i \geq 0 = \widetilde{J}^a_{i,T+1}(x^a_i;\boldsymbol{\beta}^a)$, so the result holds at time period $T+1$. Assuming that the result holds at time period $t+1$, we show that the result holds at time period t as well. Using the induction assumption, we can upper bound the objective function of the maximization problem in (21) at any feasible solution to this problem as

$$
\begin{aligned}
&\sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}}\lambda^a_{jt}\,y^a_{ijk}\Big\{\theta^a_{jk}\Big[r^a_{ijk}+\widetilde{J}^a_{i,t+1}(x^a_i-1;\boldsymbol{\beta}^a)-\widetilde{J}^a_{i,t+1}(x^a_i;\boldsymbol{\beta}^a)\Big]-\beta^a_j\Big\}+\widetilde{J}^a_{i,t+1}(x^a_i;\boldsymbol{\beta}^a)\\
&\overset{(a)}{=}\sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}}\lambda^a_{jt}\,y^a_{ijk}\Big\{\theta^a_{jk}\Big[r^a_{ijk}+\widetilde{J}^a_{i,t+1}(x^a_i-1;\boldsymbol{\beta}^a)\Big]-\beta^a_j\Big\}+\Bigg(1-\sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}}\lambda^a_{jt}\,y^a_{ijk}\,\theta^a_{jk}\Bigg)\widetilde{J}^a_{i,t+1}(x^a_i;\boldsymbol{\beta}^a)\\
&\overset{(b)}{\leq}\sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}}\lambda^a_{jt}\,y^a_{ijk}\Big\{\theta^a_{jk}\Big[r^a_{ijk}+\alpha^a_i\,(x^a_i-1)\Big]-\beta^a_j\Big\}+\Bigg(1-\sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}}\lambda^a_{jt}\,y^a_{ijk}\,\theta^a_{jk}\Bigg)\alpha^a_i\,x^a_i\\
&\overset{(c)}{=}\sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}}\lambda^a_{jt}\,y^a_{ijk}\Big\{\theta^a_{jk}\Big[r^a_{ijk}-\alpha^a_i\Big]-\beta^a_j\Big\}+\alpha^a_i\,x^a_i\overset{(d)}{\leq}\alpha^a_i\,x^a_i
\end{aligned}
$$

where (a) is by arranging the terms, (b) follows by noting that $\sum_{k\in\mathcal{K}}y^a_{ijk}\leq 1$ in a feasible solution to (21), in which case, we get $\sum_{j\in\mathcal{D}}\sum_{k\in\mathcal{K}}\lambda^a_{jt}\,y^a_{ijk}\,\theta^a_{jk}\leq\sum_{j\in\mathcal{D}}\lambda^a_{jt}\sum_{k\in\mathcal{K}}y^a_{ijk}\leq\sum_{j\in\mathcal{D}}\lambda^a_{jt}\leq 1$, as well as using the induction assumption, (c) is by arranging the terms and (d) follows by the fact that $\theta^a_{jk}\,\alpha^a_i\geq r^a_{ijk}\,\theta^a_{jk}-\beta^a_j$ for all $j\in\mathcal{D}$ and $k\in\mathcal{K}$. Thus, the objective function of problem (21) at any feasible solution is at most $\alpha^a_i\,x^a_i$, so $\widetilde{J}^a_{it}(x^a_i;\boldsymbol{\beta}^a)\leq\alpha^a_i\,x^a_i$, which completes the induction argument. By the earlier discussion in this section, we have $\widetilde{J}^a_t(\boldsymbol{x}^a;\boldsymbol{\beta}^a)=\sum_{i\in\mathcal{F}}\widetilde{J}^a_{it}(x^a_i;\boldsymbol{\beta}^a)+\sum_{j\in\mathcal{D}}\sum_{s=t}^T\lambda^a_{js}\,\beta^a_j$, so $\widetilde{J}^a_1(\boldsymbol{z}^a;\boldsymbol{\beta}^a)=\sum_{i\in\mathcal{F}}\widetilde{J}^a_{i1}(x^a_i;\boldsymbol{\beta}^a)+\sum_{t\in\mathcal{T}}\sum_{j\in\mathcal{D}}\lambda^a_{jt}\,\beta^a_j\leq\sum_{i\in\mathcal{F}}\alpha^a_i\,z^a_i+\sum_{t\in\mathcal{T}}\sum_{j\in\mathcal{D}}\lambda^a_{jt}\,\beta^a_j$.

For the fixed Lagrange multipliers $\boldsymbol{\beta}^a$, an optimal dual solution to (22) is $\boldsymbol{\alpha}^a$, so $f^a_{\mathsf{LP}}(\boldsymbol{z}^a;\boldsymbol{\beta}^a)=\sum_{i\in\mathcal{F}}\alpha^a_i\,z^a_i$. Thus, by the last inequality, we get $\widetilde{J}^a_1(\boldsymbol{z}^a;\boldsymbol{\beta}^a)\leq f^a_{\mathsf{LP}}(\boldsymbol{z}^a;\boldsymbol{\beta}^a)+\sum_{t\in\mathcal{T}}\sum_{j\in\mathcal{D}}\lambda^a_{jt}\,\beta^a_j$. ■

Let $\mathcal{Z}$ be the set of feasible solutions for the Placement problem and $\overline{\mathcal{Z}}$ be the set of feasible solutions for the continuous relaxation of the same problem. Using $\overline{U}_{\mathsf{LP}}$ to denote the upper bound provided by using our bounding functions, this upper bound is given by $\overline{U}_{\mathsf{LP}}=\max_{\boldsymbol{z}\in\overline{\mathcal{Z}}}\sum_{a\in\mathcal{A}}f^a_{\mathsf{LP}}(\boldsymbol{z}^a)$, where $\boldsymbol{z}=(z^a_i : i\in\mathcal{F},\ a\in\mathcal{A})$ captures the inventory placement decisions for all products. We have $f^a_{\mathsf{LP}}(\boldsymbol{z}^a)=\min_{\boldsymbol{\beta}^a\geq\mathbf{0}}\big\{f^a_{\mathsf{LP}}(\boldsymbol{z}^a;\boldsymbol{\beta}^a)+\sum_{t\in\mathcal{T}}\sum_{j\in\mathcal{D}}\lambda^a_{jt}\,\beta^a_j\big\}$ by the discussion just after (22). In this case, combining the last two equalities, the upper bound from our bounding functions is $\overline{U}_{\mathsf{LP}}=\max_{\boldsymbol{z}\in\overline{\mathcal{Z}}}\sum_{a\in\mathcal{A}}\min_{\boldsymbol{\beta}^a\geq\mathbf{0}}\big\{f^a_{\mathsf{LP}}(\boldsymbol{z}^a;\boldsymbol{\beta}^a)+\sum_{t\in\mathcal{T}}\sum_{j\in\mathcal{D}}\lambda^a_{jt}\,\beta^a_j\big\}$. Noting (22), observe that $f^a_{\mathsf{LP}}(\boldsymbol{z}^a,\boldsymbol{\beta}^a)$ corresponds to the optimal objective value of a linear program as a function of the right side of a constraint when viewed as a function of $\boldsymbol{z}^a$ and the optimal objective value of a linear program as a function of the objective function coefficients when viewed as a function of $\boldsymbol{\beta}^a$. Thus, $f^a_{\mathsf{LP}}(\boldsymbol{z}^a,\boldsymbol{\beta}^a)$ is concave in $\boldsymbol{z}^a$ and convex in $\boldsymbol{\beta}^a$. Also, $\overline{\mathcal{Z}}$ is convex. Thus, we can change the order of maximization and minimization in the definition of $\overline{U}_{\mathsf{LP}}$ to get $\overline{U}_{\mathsf{LP}} = \min_{\boldsymbol{\beta}\geq \mathbf{0}} \max_{\boldsymbol{z}\in\overline{\mathcal{Z}}}\{\sum_{a\in\mathcal{A}} f^a_{\mathsf{LP}}(\boldsymbol{z}^a;\boldsymbol{\beta}^a) + \sum_{a\in\mathcal{A}}\sum_{t\in\mathcal{T}}\sum_{j\in\mathcal{D}} \lambda^a_{jt}\,\beta^a_j\}$, where $\boldsymbol{\beta} = (\beta^a_j : a\in\mathcal{A},\ j\in\mathcal{D})$ is the vector of Lagrange multipliers for all products. Furthermore, since $f^a_{\mathsf{LP}}(\boldsymbol{z}^a,\boldsymbol{\beta}^a)$ is convex in $\boldsymbol{\beta}^a$ and pointwise maximum of convex functions is convex, we can solve the last outer minimization problem using standard convex optimization tools to obtain an optimal solution $\widehat{\boldsymbol{\beta}}_{\mathsf{LP}} = (\widehat{\beta}^a_{j,\mathsf{LP}} : a\in\mathcal{A}\ ,j\in\mathcal{D})$. Thus, recalling that $\overline{\mathcal{Z}}$ is the continuous relaxation of $\mathcal{Z}$ so that $\overline{\mathcal{Z}}\supseteq\mathcal{Z}$ and using Proposition J.1, we obtain the chain of inequalities

$$
\begin{aligned}
&\overline{U}_{\mathsf{LP}} = \min_{\boldsymbol{\beta}\geq\mathbf{0}}\max_{\boldsymbol{z}\in\overline{\mathcal{Z}}}\left\{\sum_{a\in\mathcal{A}} f^a_{\mathsf{LP}}(\boldsymbol{z}^a;\boldsymbol{\beta}^a) + \sum_{a\in\mathcal{A}}\sum_{t\in\mathcal{T}}\sum_{j\in\mathcal{D}}\lambda^a_{jt}\,\beta^a_j\right\} \\
&= \max_{\boldsymbol{z}\in\overline{\mathcal{Z}}}\left\{\sum_{a\in\mathcal{A}} f^a_{\mathsf{LP}}(\boldsymbol{z}^a;\widehat{\boldsymbol{\beta}}^a_{\mathsf{LP}}) + \sum_{a\in\mathcal{A}}\sum_{t\in\mathcal{T}}\sum_{j\in\mathcal{D}}\lambda^a_{jt}\,\widehat{\beta}^a_{j,\mathsf{LP}}\right\} \;\geq\; \max_{\boldsymbol{z}\in\mathcal{Z}}\left\{\sum_{a\in\mathcal{A}} f^a_{\mathsf{LP}}(\boldsymbol{z}^a;\widehat{\boldsymbol{\beta}}^a_{\mathsf{LP}}) + \sum_{a\in\mathcal{A}}\sum_{t\in\mathcal{T}}\sum_{j\in\mathcal{D}}\lambda^a_{jt}\,\widehat{\beta}^a_{j,\mathsf{LP}}\right\} \\
&\geq \max_{z\in\mathcal{Z}}\sum_{a\in\mathcal{A}}\widetilde{J}^a_1(\boldsymbol{z}^a;\widehat{\boldsymbol{\beta}}^a_{\mathsf{LP}}) \;\geq\; \max_{z\in\mathcal{Z}}\sum_{a\in\mathcal{A}} J^a_1(\boldsymbol{z}^a) \;=\; \zeta_{\mathsf{OPT}}
\end{aligned}
$$

where the last inequality holds since $\widetilde{J}^a_t(\boldsymbol{x}^a;\widehat{\boldsymbol{\beta}}^a_{\mathsf{LP}}) \geq J^a_t(\boldsymbol{x}^a)$ for any $\boldsymbol{x}^a\in\mathbb{Z}^n_+$ by the first result regarding the dynamic program in (20) and the last equality uses the Placement problem.

By the chain of inequalities above, we can solve the problem $\max_{z\in\mathcal{Z}}\sum_{a\in\mathcal{A}}\widetilde{J}^a_1(\boldsymbol{z}^a;\widehat{\boldsymbol{\beta}}^a_{\mathsf{LP}})$ to obtain an upper bound on the optimal objective value of the Placement problem and this upper bound is at least as tight as the one provided by our bounding functions. We choose our Lagrange multipliers as $\widehat{\boldsymbol{\beta}}_{\mathsf{LP}}$. By the discussion in the previous paragraph, we can compute $\widehat{\boldsymbol{\beta}}_{\mathsf{LP}}$ through a convex program. By the second result regarding the dynamic program in (20), recall that $\widetilde{J}^a_1(\boldsymbol{z}^a;\widehat{\boldsymbol{\beta}}^a_{\mathsf{LP}}) = \sum_{i\in\mathcal{F}}\widetilde{J}^a_{i1}(z^a_i;\widehat{\boldsymbol{\beta}}^a_{\mathsf{LP}})$. We can show that $\widetilde{J}^a_{i1}(z^a_i;\widehat{\boldsymbol{\beta}}^a_{\mathsf{LP}})$ is concave in z^a_i. In this case, we can formulate the problem $\max_{z\in\mathcal{Z}}\sum_{a\in\mathcal{A}}\widetilde{J}^a_1(\boldsymbol{z}^a;\widehat{\boldsymbol{\beta}}^a_{\mathsf{LP}})$ as a min-cost network flow problem as in Appendix I, so we can obtain the upper bound from the Lagrangian relaxation strategy efficiently.

Appendix K: A Policy Based on Periodically Solving a Linear Programming Approximation

We report on our experience with a delivery promise and fulfillment policy that is based on periodically solving a linear programming approximation. The linear programming approximation that we use is a variant of the Bounding problem, where we update the inventory availabilities and expected demands for the products based on the current inventories and the time period. We divide the selling horizon into Q equal segments, so that segment q includes the time periods $\{1+(q-1)\,T/Q,\ldots,q\,T/Q\}$. We solve our linear programming approximation at the beginning of each segment. In particular, considering the Bounding problem, if the remaining inventory for product a at fulfillment center i at the beginning of segment q is $x^a_{i,1+(q-1)T/Q}$, then we replace the right side of the first constraint with $x^a_{i,1+(q-1)T/Q}$ and the right side of the second constraint with $\sum_{\ell=1+(q-1)T/Q}^{T} \lambda^a_{j\ell}$. In this case, the right sides of the first and second constraints, respectively, reflect the inventory availabilities at different fulfillment centers and the total expected demands to come from different demand regions. Letting $\widehat{\boldsymbol{w}}^a$ be an optimal solution, if we have a demand for product a from demand region j, then we sample the fulfillment center and delivery promise pair (i,k) with probability $\widehat{\eta}^a_{ijk} = \frac{\widehat{w}^a_{ijk}}{\sum_{\ell=1+(q-1)T/Q}^{T} \lambda^a_{jt}}$. We offer the sampled delivery promise and use the sampled fulfillment center upon acceptance of the offered promise. We continue sampling the fulfillment center and delivery promise pairs according to these probabilities until we reach the beginning of the next segment and solve the Bounding problem once more. This policy can sample a fulfillment center with no remaining inventory. If the sampled fulfillment center does not have remaining inventory, then we revert to offering the dummy promise. We set $Q = 4$, so we have four segments. Solving the Bounding problem four times over the selling horizon resulted in the longest running times that we could afford. We shortly comment on the running times.

We implemented two variants of the policy. In the first variant, we implemented the policy precisely as described in the previous paragraph. If we have a demand for product a from demand region j, then we sample the fulfillment center and delivery promise pair (i,k) with probability $\widehat{\eta}^a_{ijk}$, where $\widehat{\eta}^a_{ijk}$ is as given in the previous paragraph. We offer the sampled delivery promise and use the sampled fulfillment center upon acceptance of the offered promise. If the sampled fulfillment center does not have remaining inventory, then we revert to offering the dummy promise. The first variant can assign a demand to a fulfillment center with no inventory. In the second variant, if the remaining inventories of product a at the fulfillment centers at time period t are given by the vector $\boldsymbol{x}^a_t = (x^a_{it} : i \in \mathcal{F})$ and we have a demand for product a from demand region j at time period t, then we sample the fulfillment center and delivery promise pair (i,k) with probability $\frac{\mathbb{1}_{(x^a_{it} \geq 1)}\, \widehat{\eta}^a_{ijk}}{\sum_{f \in \mathcal{F}} \mathbb{1}_{(x^a_{ft} \geq 1)}\, \widehat{\eta}^a_{fjk}}$. In the second variant, we rescale the sampling probabilities by focusing only on the fulfillment centers with remaining inventories. The second variant performed significantly better than the first one. We present the results for the second variant.

We use LPR to refer to the second variant of the policy that periodically solves the linear programming approximation. We tested the performance of LPR on the test problems with $\delta = 0$ in Section 8.2. We give our computational results in Table EC.1. The first column in the table shows the parameter configuration for our test problems by using the tuple (δ, η, μ). The next two columns show the total expected profits obtained by APS and LPR, normalized by the tightest upper bound we have on the optimal total expected profit, which is provided by the Lagrangian relaxation strategy. The last column shows the percent gap between the total expected profits obtained by APS and LPR. To make the placement decisions, LPR uses the same strategy as APS. Therefore, the

Params. (δ, η, μ)	Total Exp. Profit APS	LPR	% Gap
$(0, 1.25, 0.8)$	87.61	81.65	6.81
$(0, 1.25, 1.0)$	90.35	84.31	6.68
$(0, 1.50, 0.8)$	87.67	81.80	6.69
$(0, 1.50, 1.0)$	90.52	84.53	6.62
$(0, 1.75, 0.8)$	87.75	81.93	6.63
$(0, 1.75, 1.0)$	90.74	84.73	6.62
$(0, 2.00, 0.8)$	87.82	82.08	6.54
$(0, 2.00, 1.0)$	90.93	84.91	6.61
Average	89.17	83.24	6.65

Table EC.1 Total expected profits obtained by the benchmark strategies for the synthetically generated datasets.

difference in the performance of APS and LPR is only due to the fact that these two benchmarks use different delivery promise and fulfillment policies. Our results indicate that APS provides noticeable improvements over LPR. The average percent gap between the total expected profits obtained by APS and LPR is 6.65%. The average running time for LPR is 30430 seconds. This running time includes the time to simulate the performance of the delivery promise and fulfillment policy for 500 sample paths, which requires solving the linear programming approximation four times for each of the 500 products in each sample path, resulting in solving 10^6 linear programs. On the other hand, the average running time for APS is 7494 seconds. Because we compute the value functions used by the delivery promise and fulfillment policy in APS once at the beginning of the selling horizon, once we compute the value functions, simulating the performance of the delivery promise and fulfillment policy in APS only involves sampling the demands and implementing the decision of the policy. Therefore, the running times for APS are shorter.

Even though one can give a performance guarantee for a delivery promise and fulfillment policy that may assign a demand to a fulfillment center with no remaining inventory, the practical performance of such a policy can be poor. The first variant of LPR discussed earlier in this section is indeed a policy that may assign a demand to a fulfillment center with no remaining inventory. The second variant of LPR tried to make up for this shortcoming by rescaling the sampling probabilities to focus only on the fulfillment centers with remaining inventories. In our test problems, the second variant improved over the first variant by 17% on average, so focusing only on fulfillment centers with remaining inventories can provide substantial improvements. Because the delivery promise and fulfillment decision of the second variant of LPR for a particular demand depends jointly on the inventories at all fulfillment centers, it is computationally difficult to compute the value functions of the policy, making it difficult to perform rollout on the second variant of LPR. However, we can perform rollout on the first variant of LPR because this variant, just like the randomized policy in Section 7, assigns each demand to a fulfillment center with a prefixed probability. When we

Params. (δ, η, μ)	Total Exp. Profit Rand. Pol.	APS	% Gap
$(0, 1.25, 0.8)$	71.55	87.61	18.33
$(0, 1.25, 1.0)$	70.60	90.35	21.86
$(0, 1.50, 0.8)$	71.70	87.67	18.21
$(0, 1.50, 1.0)$	70.81	90.52	21.78
$(0, 1.75, 0.8)$	71.92	87.75	18.03
$(0, 1.75, 1.0)$	71.07	90.74	21.67
$(0, 2.00, 0.8)$	72.13	87.82	17.86
$(0, 2.00, 1.0)$	71.27	90.93	21.62
Average	71.38	89.17	19.92

Table EC.2 Total expected profits obtained by APS with and without rollout.

performed rollout on the first variant of LPR, its performance is essentially identical to that of APS, but such a policy, by embedding rollout, deviates from the traditional policies in the revenue management literature that are based on periodically solving a linear programming approximation. Lastly, the delivery promise and fulfillment policy used by APS is based on performing rollout on the randomized policy in Section 7. The randomized policy may assign a demand to a fulfillment center with no remaining inventory, but performing rollout makes up for this shortcoming in the sense that the rollout policy does not consider using a fulfillment center with no remaining inventory. As explained in Section 7, because of this difference, the practical performance of the rollout policy can be significantly better than that of the randomized policy. Considering the test problems in Table EC.1, in Table EC.2, we give the total expected profits of APS with and without rollout. The benchmark APS without rollout corresponds to the randomized policy. Thus, the second and third columns in the table, respectively, give the total expected profits obtained by the randomized policy and APS. The last column gives the percent gap between the two total expected profits. Our results indicate that using rollout to avoid assigning a demand to a fulfillment center with no remaining inventory is important and can improve the total expected profits by more than 20%.

Appendix L: Processing the Order Transaction Data

We give the additional details of our approach for coming up with the fulfillment centers, demand regions and profit parameters in our computational experiments. There are a number of transaction records with missing or corrupt seller or customer locations, purchase or delivery time stamps, selling prices and shipping costs for the products. We eliminate such transaction records. The available data provides the latitude and longitude of the seller involved in each transaction record. To come up with the fulfillment centers, we treat each seller location as a potential fulfillment center. Among all existing seller locations, we choose 30 locations to serve as fulfillment centers in such a way that we minimize the largest distance between any chosen location and an existing seller location. Minimizing the largest distance between any chosen location and an existing seller location ensures that the chosen locations are geographically well-spread. We solve a simple integer program to minimize such largest distance. The available data also provides the latitude and longitude of the customer involved in each transaction record. To come up with the demand regions, we treat each customer location as a potential demand region. We use k-means to build 80 clusters of customer locations and use the centroid of each cluster as a demand region.

To come up with the profits from serving a demand, we begin by estimating the shipping costs. We split the transaction records into seven distance categories according to the miles that the order travels from seller to customer. Distance category ℓ corresponds to orders with travel distance in $[(\ell-1)\,500, \ell\,500]$. For each distance category, we refer to the orders in the top 20-th to 40-th percentile of the shipping costs as the faster delivery promise and the orders in the bottom 20-th to 40-th percentile of the shipping costs as the slower delivery promise. Combining all orders with faster delivery promises, we use linear regression to estimate the per mile shipping cost. We repeat the same process for the orders with slower delivery promises. Letting σ_1 and σ_2 be the estimated per mile shipping cost for faster and slower delivery promises, we set $r^a_{ij1} = p^a - \sigma_1\,\mathsf{dist}(i,j)$ and $r^a_{ij2} = p^a - \sigma_2\,\mathsf{dist}(i,j)$, where p^a is the selling price for product a and $\mathsf{dist}(i,j)$ is the distance between fulfillment center i and demand region j. We tried to use the weights and volumes of the products as well to estimate the shipping costs, but did not find a reliable relationship.

Lastly, recall that we use α_i to denote the fraction of orders that we assign to fulfillment center i, which we compute by assigning the total number of orders from each seller to a nearby fulfillment center. The challenge is that many of the sellers are tightly clustered in urban areas. If we assign the total number of orders from a seller to the closest fulfillment center, then a fulfillment center in an urban area gets an exorbitant share, making the other fulfillment centers irrelevant. We use the following approach to come up with the parameters $(\alpha_i : i \in \mathcal{F})$. Let n_ℓ be the number of units shipped out by seller ℓ. We can compute n_ℓ from the data through simple counting. We assign the total units shipped out of each seller ℓ to one of its κ closest fulfillment centers so that we maximize the smallest total number of units assigned to a fulfillment center. In this way, we ensure that a fulfillment center does not get an exorbitant share of the units. We solve an integer program to maximize the smallest total number of units assigned to a fulfillment center. Letting $\mathcal{S}_i$ be the set of sellers assigned to fulfillment center i, we set $\alpha_i = \frac{\sum_{\ell \in \mathcal{S}_i} n_\ell}{\sum_{k \in \mathcal{F}} \sum_{\ell \in \mathcal{S}_k} n_\ell}$.