This article was downloaded by: [76.17.100.218] On: 07 October 2023, At: 13:01
Publisher: Institute for Operations Research and the Management Sciences (INFORMS)
INFORMS is located in Maryland, USA

Manufacturing & Service Operations Management

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Order-Optimal Correlated Rounding for Fulfilling Multi-Item E-Commerce Orders

Will Ma

To cite this article:
Will Ma (2023) Order-Optimal Correlated Rounding for Fulfilling Multi-Item E-Commerce Orders. Manufacturing & Service Operations Management 25(4):1324-1337. https://doi.org/10.1287/msom.2023.1219

Full terms and conditions of use: https://pubsonline.informs.org/Publications/Librarians-Portal/PubsOnLine-Terms-and-Conditions

This article may be used only for the purposes of research, teaching, and/or private study. Commercial use or systematic downloading (by robots or other automatic processes) is prohibited without explicit Publisher approval, unless otherwise noted. For more information, contact permissions@informs.org.

The Publisher does not warrant or guarantee the article's accuracy, completeness, merchantability, fitness for a particular purpose, or non-infringement. Descriptions of, or references to, products or publications, or inclusion of an advertisement in this article, neither constitutes nor implies a guarantee, endorsement, or support of claims made of that product, publication, or service.

Copyright © 2023, INFORMS

Please scroll down for article—it is on subsequent pages

With 12,500 members from nearly 90 countries, INFORMS is the largest international association of operations research (O.R.) and analytics professionals and students. INFORMS provides unique networking and learning opportunities for individual professionals, and organizations of all types and sizes, to better understand and use O.R. and analytics tools and methods to transform strategic visions and achieve better outcomes.
For more information on INFORMS, its publications, membership, or meetings visit http://www.informs.org

https://pubsonline.informs.org/journal/msom

MANUFACTURING & SERVICE OPERATIONS MANAGEMENT
Vol. 25, No. 4, July–August 2023, pp. 1324–1337
ISSN 1523-4614 (print), ISSN 1526-5498 (online)

Order-Optimal Correlated Rounding for Fulfilling Multi-Item E-Commerce Orders

Will Ma[a]

[a] Graduate School of Business, Columbia University, New York, New York 10027
Contact: wm2428@gsb.columbia.edu, https://orcid.org/0000-0002-2420-4468 (WM)

Received: August 10, 2022
Revised: January 18, 2023
Accepted: March 26, 2023
Published Online in Articles in Advance: May 2, 2023

https://doi.org/10.1287/msom.2023.1219

Copyright: © 2023 INFORMS

Abstract. ***Problem definition:*** We study the dynamic fulfillment problem in e-commerce, in which incoming (multi-item) customer orders must be immediately dispatched to (a combination of) fulfillment centers that have the required inventory. ***Methodology/results:*** A prevailing approach to this problem, pioneered by Jasin and Sinha in 2015, has been to write a "deterministic" linear program that dictates, for each item in an incoming multi-item order from a particular region, how frequently it should be dispatched to each fulfillment center (FC). However, dispatching items in a way that satisfies these frequency constraints, without splitting the order across too many FCs, is challenging. Jasin and Sinha in 2015 identified this as a *correlated rounding* problem and proposed an intricate rounding scheme that they proved was suboptimal by a factor of at most $\approx q/4$ on a q-item order. This paper provides, to our knowledge, the first substantially improved scheme for this correlated rounding problem, which is suboptimal by a factor of at most $1 + \ln(q)$. We provide another scheme for sparse networks, which is suboptimal by a factor of at most d if each item is stored in at most d FCs. We show both of these guarantees to be tight in terms of the dependence on q or d. Our schemes are simple and fast, based on an intuitive idea; items wait for FCs to "open" at random times but observe them on "dilated" time scales. This also implies a new randomized rounding method for the classical Set Cover problem, which could be of general interest. ***Managerial implications:*** We numerically test our new rounding schemes under the same realistic setups as Jasin and Sinha and find that they improve runtimes, shorten code, and robustly improve performance. Our code is made publicly available online.

History: This paper was selected for Fast Track in the *M&SOM* Journal from the 2022 MSOM Supply Chain Management SIG Conference.
Funding: This research was partially funded by a grant from Amazon.com Inc., which was awarded through collaboration with the Columbia Center of AI Technology (CAIT).

Keywords: inventory theory and control • math programming • retailing

1. Introduction

E-commerce has exploded in recent times, achieving unbelievable global scale, delivery speed, and system complexity. The short-term operations of a typical e-commerce giant involve pulling inventory from suppliers into its fulfillment centers (FCs), including retail stores that can also be used to fulfill online orders; awaiting purchases from online customers, which can be influenced by a powerful search/recommendation engine; and finally, delivering the goods to the customer's doorstep through a flexible transportation system that allows different FCs in the network to be used for fulfilling demand from any particular region.

This paper focuses on the final part of these operations, which is the problem of dynamically dispatching incoming customer orders to FCs while treating this customer demand (as influenced by search/recommendation) and inventory replenishment as exogenous. The decision is on how to dynamically allocate finite inventories of multiple items, each of which has been placed in multiple FCs, over a finite time horizon, representing the duration until the next inventory replenishment. The objective is to minimize the total costs from fulfillment and inventory stockouts.

This dynamic fulfillment problem is challenging for several reasons. First, decisions must be made with consideration of the future orders to come, because depleting inventories at the wrong places can set off a chain reaction of long-distance and split shipments, as originally demonstrated by Xu et al. (2009). Moreover, because of the uncertainty in future orders, forward-lookingness requires a high-dimensional stochastic dynamic program that is intractable to solve, as noted by Acimovic and Farias (2019). Finally, the mere scale and speed of the problem restricts us to fast and simple heuristics, with more elaborate optimizations exacerbating the issue of system complexity.

In light of these challenges, a prevailing approach to the dynamic fulfillment problem is deterministic-relaxation-based, as pioneered by Jasin and Sinha (2015). Namely, a linear program (LP) that views the system as deterministic

Downloaded from informs.org by [76.17.100.218] on 07 October 2023, at 13:01 . For personal use only, all rights reserved.

Downloaded from informs.org by [76.17.100.218] on 07 October 2023, at 13:01 . For personal use only, all rights reserved.

is written, describing inventory levels of every item at every FC, and expected demands at different regions, which includes information about items frequently purchased together in the same order. The objective captures fixed shipping costs (mostly dependent on the number of distinct FCs used to fulfill an order), variable shipping costs (dependent on items and distances), and shortage costs (dependent on penalties paid for orders not fulfilled). The LP is then solved, providing a "master plan" of matching supply to demand, which prescribes for different orders from different regions how frequently each FC should be used to fulfill each item in that order. As orders came in real-time, Jasin and Sinha (2015) randomly dispatched the items to FCs, making sure to follow the fulfillment frequencies outlined in the LP's plan.

Although seemingly uninformed, this *randomized fulfillment* approach is simple, fast, and highly parallelizable because it does not require real-time inventory information across the network once the LP solution is given. Under large system scales, it also pays variable shipping and shortage costs similar to what is outlined in the LP. However, fixed costs remain a challenge; the problem of covering all the items in an order using a small number of distinct FC's was already difficult, and the LP's fulfillment frequencies now impose additional constraints. Moreover, it has been shown that fixed costs, capturing the number of boxes from different origins shipped, constitute the majority of e-commerce fulfillment costs (Xu et al. 2009, Jasin and Sinha 2015), so this presents a major issue. The seminal insight, according to Jasin and Sinha (2015), is that these frequencies are actually helpful; when they are used to randomly assign an FC to each item, if *positive correlation* is induced in the assignments across items, then many items end up assigned to the same FC, and not many distinct FCs are used. These authors derived an intricate method for inducing this correlation.

Despite its significance and impact on subsequent work (see, e.g., Lei et al. 2018, 2021; and Zhao et al. 2020), to the best of our knowledge, the correlation method of Jasin and Sinha (2015) has never been substantially improved, until now. This paper derives a new correlation method that is intuitively simpler and computationally faster and achieves tight performance in two different regimes.

1.1. Correlated Rounding Problem of Jasin and Sinha (2015)

Consider a single order (from a particular region at a particular time) consisting of q items. For each item in the order, denoted using $i \in \{1, \dots, q\}$, we are told by the fraction of time u_{ki} that it must be fulfilled from each FC $k \in \{1, \dots, K\}$. Every item must be fulfilled, so $\sum_k u_{ki} = 1$ for all i. We must randomly choose an FC for each item i according to these probabilities u_{ki}, and an FC is used if any item is assigned to it (meaning we would ship a box out of that FC).

Intuitively, the goal is to not use many distinct FCs. This is formalized as no FC being used with a probability greater than necessary. Specifically, for each FC k, it has to used with probability of at least u_{ki} to fulfill any item i, and hence, $y_k := \max_i u_{ki}$ is a lower bound on its probability of being used. A method that randomly assigns every item i to an FC following its marginal probability vector $(u_{ki})_{k=1}^K$ is called a *rounding* scheme, and the rounding scheme is said to be *α-competitive* if it uses every FC k with probability at most $\alpha \cdot y_k$ for some $\alpha \geq 1$. Here, α is referred to as the *guarantee* of the rounding scheme, which is ideally as small as possible.

A naive rounding scheme is to independently draw an FC for each item. However, such a random outcome likely uses many distinct FCs, causing FCs to be used more frequently than necessary, and Jasin and Sinha (2015) showed that the guarantee of the independent rounding scheme can be as bad as $\alpha = q$ on a q-item order. Jasin and Sinha (2015) derived an improved rounding scheme that correlates positively the FCs drawn across items so that the random outcome uses fewer distinct FCs. They establish that given any q marginal distributions over FCs, this correlated rounding scheme is $\approx q/4$-competitive, improving the guarantee of the naive rounding by a factor of 4.

In this paper, we derive two new correlated rounding schemes. The first is $(1 + \ln(q))$-competitive, completely improving the earlier guarantees in terms of order dependence on q — from linear to logarithmic. The second is d-competitive, where d is a *sparsity* parameter that describes the maximum number of options that any item has in terms of where to be fulfilled, that is, $d = \max_i |\{k : u_{ki} > 0\}|$. Both of these guarantees are tight for the correlated rounding problem, as we will show.

1.1.1. Implications for Dynamic Fulfillment.

Our rounding schemes are directly applicable to the original dynamic fulfillment problem via the approach of Jasin and Sinha (2015). Indeed, each incoming order can be separately and randomly dispatched, using our choice of rounding scheme. The results of Jasin and Sinha (2015) then imply that in the dynamic fulfillment problem, the total cost paid is asymptotically at most β times the optimum, where β is a constant that depends on the average value of $\min\{1 + \ln(q), d\}$ across orders (different orders have different sizes q and sparsity parameters d, and we can choose the rounding scheme with the better guarantee between $1 + \ln(q)$ and d for each incoming order). As a special case, if the largest order has size $\overline{q}$, then the guarantee is $1 + \ln(\overline{q})$, matching computational hardness results for the dynamic fulfillment problem even when there is a single order. Further details can be found in Section 5.

We note, however, that reducing everything down to the correlated rounding subroutine is not the only approach to dynamic fulfillment. Indeed, the correlated rounding problem imposes frequency constraints on

Downloaded from informs.org by [76.17.100.218] on 07 October 2023, at 13:01 . For personal use only, all rights reserved.

every order (that every item i in every order is assigned to each FC k with marginal probability exactly u_{ki}), with the rationale being that the variable shipping and inventory shortage costs become relatively inconsequential under large system scales; unfortunately, this can be restrictive compared with some alternative approaches, as summarized in Acimovic and Farias (2019). Nonetheless, this simple and fast approach performs well numerically in realistic setups, as shown in Jasin and Sinha (2015). In Section 6, we show using the same setups that our new rounding schemes robustly bolster the performance of the randomized fulfillment approach while shortening code and runtimes. Our code is made publicly available.

1.2. Main Idea Behind New Rounding Schemes and Analysis

Another benefit of our rounding schemes is that they have a simple intuition; each FC draws a random "opening time," and each item is assigned to the first FC that it sees open under its own item-specific "time dilation." We now describe in detail our two rounding schemes and analysis.

Recall that we are trying to induce positive correlation in the FCs assigned across items. To do this, we imagine a process where each FC is initially closed and opens at a random time. Items are assigned to the first FC that they see open. Importantly, each item i views the openings of FCs on its own dilated time scale, calibrated so that the probability of it seeing any FC k open first is exactly u_{ki}. Because an FC opening early means that it will be seen first by more (but not necessarily all) items, this induces positive correlation in the FCs assigned across different items.

To make this precise, for each FC k, we draw its opening time E_k independently from an exponential distribution with mean $1/y_k$, where $y_k := \max_i u_{ki}$. We then define the dilated time scale for an item i as it sees each FC k open at time $\frac{y_k}{u_{ki}}E_k$, which we note is no earlier than E_k, because $\frac{y_k}{u_{ki}} \geq 1$. (If $u_{ki}=0$, then $\frac{y_k}{u_{ki}}E_k = \infty$, and item i never sees FC k open.) The dilated opening times $\frac{y_k}{u_{ki}}E_k$ are exponentially distributed with means $\frac{y_k}{u_{ki}} \cdot \frac{1}{y_k} = \frac{1}{u_{ki}}$ and independent across k. Through the lens of Poisson processes, it is easy to see that the probability of each FC k arriving first into the view of item i is exactly $\frac{u_{ki}}{u_{1i}+\cdots+u_{Ki}} = u_{ki}$, as desired.

The Poisson lens also helps us upper-bound the probability of an FC k getting used at all. Indeed, because an FC k can be seen only at times later than E_k, it can get used only if it arrives when at least one item is still waiting, an event whose probability is exponentially decaying over time. Unfortunately, random variable E_k is correlated with the latter event, making the analysis complicated. To fix this, we instead consider a related process where FC k is "repeatedly opening" following a Poisson process of rate y_k, which allows us to exploit the memoryless property and take an elementary integral to show that the probability of FC k opening is at most $(1+\ln(q))y_k$, completing our sketch of why our first rounding scheme is $(1+\ln(q))$-competitive.

To motivate our second rounding scheme, we note that the preceding analysis is poor when q is enormous, because for a long time at least one item will still be waiting, during which FC openings will result in usage. Therefore, we consider a modified scheme where each FC k is "forced open" at time $1/y_k$, even if $E_k > 1/y_k$. For each item i, it will see each FC k forced open at time $\frac{y_k}{u_{ki}} \cdot \frac{1}{y_k} = \frac{1}{u_{ki}}$. Therefore, item i will get "force-assigned" by time $\frac{1}{\max_k u_{ki}}$, and all items will be force-assigned by time $\alpha := \frac{1}{\min_i \max_k u_{ki}}$, regardless of how many items there are. Moreover, if d is an upper bound on $|\{k : u_{ki} > 0\}|$, then $\max_k u_{ki} \geq 1/d$ for all i, and hence, $\alpha \leq d$. The fact that all items are assigned by time d w.p. 1 allows us to show that no FC gets used with probability more than dy_k.

However, these forced openings cause each item i to be overfulfilled from the FC $m(i)$ that it would first see forced open. Therefore, we make a second modification where for each item i, if the overfulfilled FC $m(i)$ were to "naturally" open (i.e., $E_{m(i)} < 1/y_{m(i)}$), then it is hidden from the view of item i (until it is forced open) with some likelihood. This likelihood can be calibrated so that i ends up seeing every FC k open first with probability exactly u_{ki}, as desired.

1.3. Further Technical Details and Relationship with Set Cover

We now outline all our new results for the correlated rounding problem and the related technical results.

- Our main results are a $(1+\ln(q))$-competitive rounding scheme and a d-competitive rounding scheme (where d denotes the sparsity parameter $\max_i |\{k : u_{ki} > 0\}|$). These rounding schemes and their analyses are presented in Section 2.
- The exact guarantee for the rounding scheme of Jasin and Sinha (2015) is given by a function B of the order size, where $B(q) = \frac{(q+1)^2}{4q}$ if q is odd and $B(q) = \frac{q+2}{4}$ if q is even. For small values of q, this is better than our guarantee of $1+\ln(q)$; for example, if $q=2$, then $B(1)=1$.
- Both of our rounding schemes have a runtime of $O(qK)$. By contrast, the rounding scheme of Jasin and Sinha (2015) has a runtime of $O(q^2K)$, containing a loop that is quadratic in the number of items q.
- If there are only two FC's, that is, $K=2$, then a 1-competitive rounding scheme was recently discovered by Zhao et al. (2020). In this scenario, our second rounding scheme would only be 2-competitive, because $d = K = 2$. However, we emphasize that parameter d represents the maximum number of distinct FCs that hold an item and can generally be much smaller than K, whereas their rounding scheme works only when $K=2$.

Downloaded from informs.org by [76.17.100.218] on 07 October 2023, at 13:01 . For personal use only, all rights reserved.

• In Section 4, we establish an additional result that computes the optimal guarantee α and rounding scheme for a given instance, using an LP of size $O(2^K)$. Jasin and Sinha (2015) also showed how to compute instance-optimal schemes using an LP of size $O(K^q)$. Although both are exponentially sized, our LP can be applied when K is small; theirs can be applied when q is small.

1.3.1. Relating the Correlated Rounding Problem to Set Cover.

• In Section 3, we show that an α-competitive rounding scheme implies a procedure for rounding a fractional Set Cover solution into a randomized cover, that is, feasible w.p. 1, and has no set chosen with probability more than α times its fractional weight.

• Therefore, we can leverage hardness results from Set Cover to show that an α-competitive rounding scheme must have $\alpha = \Omega(\log(q))$ and $\alpha \geq d$. The former lower bound establishes our $(1+\ln(q))$-competitive rounding scheme to be order-optimal in q, whereas the latter lower bound establishes our d-competitive rounding scheme to be exactly tight in d.

• Our $(1+\ln(q))$-competitive rounding scheme also improves guarantees in the aforementioned randomized rounding problem for Set Cover. To the best of our knowledge, existing rounding methods for Set Cover take each set with probability at least $\ln(q)+\omega(1)$ times its fractional weight (Raghavan and Tompson 1987); see also Motwani and Raghavan (1995) and Vazirani (2001, section 14.2). Although our improvement to $\ln(q)+1$ is only in lower-order terms, our approach via the correlated rounding problem is both new and simpler than many of the commonly-taught methods.

We note that for the Set Cover problem itself, which has nothing to do with randomization, the Greedy algorithm has a guarantee of $1+1/2+\cdots+1/q$, which is slightly smaller (better) than our $1+\ln(q)$. Nonetheless, we believe that these connections highlight how the correlated rounding problem is a harder version of Set Cover, in which a randomized solution that must satisfy constraints on how often each set is used to cover each element is required. Furthermore, it is interesting to us that a modern problem from e-commerce practice, identified by Jasin and Sinha (2015), can lead us to improve randomized rounding schemes for the age-old Set Cover problem from CS theory.

1.4. Further Related Work

The dynamic fulfillment problem, and in particular the correlated rounding approach, is more challenging and relevant in large fulfillment networks. Fulfillment networks have been getting larger with the advent of omnichannel retailing, which allows for online orders to be fulfilled from small retail stores (Acimovic and Farias 2019). Although order sizes have been decreasing with the advent of fast shipping, online retailers have been making greater efforts to delay fulfillment and consolidate multiple orders into one before fulfilling (Wei et al. 2021, Wang et al. 2022). Consequently, the dynamic fulfillment problem with multi-item orders and flexibility in how to fulfill them is as relevant as ever (DeValve et al. 2023).

In terms of the overall LP-based approach that justifies the correlated rounding problem, we should note that LP-based approaches are also heavily employed in the revenue management literature (see, e.g., Talluri and Van Ryzin 2004). They enjoy many benefits, such as scalability and ability to incorporate side constraints, and the given probabilities u_{ki} can always be updated over time through resolving (see, e.g., Jasin and Kumar 2012) to adjust for updated inventories and demand predictions over time. An early work advocating for the LP-based approach in e-commerce fulfillment is Acimovic and Graves (2015). Very recently, Amil et al. (2022) proposed a novel LP that can be used in place of the standard one, which we discuss at the end of Section 5.

2. Formal Specification and Analysis of Rounding Schemes

We recap the correlated rounding problem from the Introduction, our main object of study.

Definition 1 (Recap of Problem, Notation, and Terminology).

• An *instance* of the α-competitive rounding scheme problem consists of q marginal distributions over K FC's, given by probabilities u_{ki} satisfying $\sum_{k=1}^{K} u_{ki} = 1$ for all $i = 1,\ldots,q$.

• A *rounding scheme* must randomly assign each item i to an FC $Z_i \in \{1,\ldots,K\}$, satisfying the marginal conditions $\Pr[Z_i = k] = u_{ki}$ for all i and k.

• An FC k is *used* if any item is assigned to it, denoted by the event $\bigcup_{i=1,\ldots,q}(Z_i = k)$, which must occur with probability at least $y_k := \max_i u_{ki}$. Assume without loss that $y_k > 0$ for all k.

• A rounding scheme is *α-competitive* if, given any instance, it uses each FC k with probability at most $\alpha \cdot y_k$. The guarantee α can depend on parameters of the instance.

• The *sparsity* parameter of an instance is defined as $d = \max_i |\{k : u_{ki} > 0\}|$, the maximum number of distinct FCs that one item i could get assigned to.

We now provide efficient algorithmic specifications of our rounding schemes and analyze them. We believe both our algorithms and proofs to be quite intuitive and will frequently provide proof sketches that refer back to the intuition from Section 1.2, where items are waiting for FCs to open on their own dilated time scales.

2.1. $(1+\ln(q))$-Competitive Rounding Scheme

Our rounding scheme is specified in Algorithm 1. Relating back to the intuitive description, E_k is the time at

Downloaded from informs.org by [76.17.100.218] on 07 October 2023, at 13:01 . For personal use only, all rights reserved.

which FC k opens, and $\frac{y_k}{u_{ki}}E_k$ is the *delayed* time (because $\frac{y_k}{u_{ki}} \geq 1$) at which item i sees it open, with $\frac{y_k}{u_{ki}}E_k = \infty$ if $u_{ki} = 0$. Every item is assigned to the first FC that it sees open.

Algorithm 1 ($(1+\ln(q))$-Competitive Rounding Scheme)
For $k = 1, \ldots, K$, **do**
  $E_k \leftarrow$ independent draw from exponential distribution with mean $1/y_k$
end for
for $i = 1, \ldots, q$ **do**
  $Z_i \leftarrow \arg\min_{k=1,\ldots,K} \frac{y_k}{u_{ki}}E_k$
  ▷ Break ties arbitrarily.
end for

We now prove that Algorithm 1 is a $(1+\ln(q))$-competitive rounding scheme, where q is the number of items. To establish the marginals condition, we use the interpretation that from the perspective of any individual item, the FCs open according to independent Poisson processes.

Lemma 1. *Under Algorithm 1,* $\Pr[Z_i = k] = u_{ki}$ *for all* $i = 1, \ldots, q$ *and* $k = 1, \ldots, K$.

Proof of Lemma 1. Consider the perspective of any item i. Index Z_i is determined by the smallest realization among $\left\{\frac{y_k}{u_{ki}}E_k : k = 1, \ldots, K\right\}$, which are independent exponential random variables with means $\left\{\frac{1}{u_{ki}} : k = 1, \ldots, K\right\}$. Equivalently, Z_i is determined by the first arrival among independent Poisson processes with rates $\{u_{ki} : k = 1, \ldots, K\}$. By the Poisson merging theorem, each Poisson process k will be the first to arrive with probability $\frac{u_{ki}}{u_{1i}+\cdots+u_{Ki}}$, which equals u_{ki} because $u_{1i} + \cdots + u_{Ki} = 1$. Therefore, $\Pr[Z_i = k] = u_{ki}$ for all $k = 1, \ldots, K$, completing the proof. □

We now prove an intermediate lemma that, intuitively, bounds the probability of any item i still "waiting" (to be assigned to an FC) up to time t, which can be expressed as the event $(\min_k \frac{y_k}{u_{ki}}E_k \geq t)$. The final statement then takes a union bound of having any item still waiting, which intuitively is not too loose because these events are positively correlated; one item waiting implies that FCs were late to open, which makes other items more likely to also be waiting.

Lemma 2. *Under Algorithm 1,* $\Pr[\bigcup_{i=1}^{q}(\min_k \frac{y_k}{u_{ki}}E_k \geq t)] \leq qe^{-t}$ *for all* $t \geq 0$.

Proof of Lemma 2. First, consider any item i. Random variables $\left\{\frac{y_k}{u_{ki}}E_k : k = 1, \ldots, K\right\}$ are independent and exponentially distributed with means $\left\{\frac{1}{u_{ki}} : k = 1, \ldots, K\right\}$. Therefore, $\min_k \frac{y_k}{u_{ki}}E_k$ is exponentially distributed with mean $\frac{1}{u_{1i}+\cdots+u_{Ki}} = 1$. Consequently, $\Pr\left[\min_k \frac{y_k}{u_{ki}}E_k \geq t\right] = e^{-t}$, and by the union bound, $\Pr\left[\bigcup_{i=1}^{q}\left(\min_k \frac{y_k}{u_{ki}}E_k \geq t\right)\right] \leq qe^{-t}$, completing the proof. □

We are now ready to prove our main result for Algorithm 1. Although technical, the argument uses a simple intuitive trick. Lemma 2 has upper-bounded the probability of any item still waiting at a time t. If an FC k opens at a time when no item is still waiting, then it is guaranteed to not get used (because items can only see it open at a delayed time). Unfortunately, the opening time of an FC k is correlated with the event of having an item still waiting. To fix this, we imagine FC k as "repeatedly opening" following a Poisson process of rate y_k, with it being "used" every time it opens, as long as there is an item still waiting. Because Poisson processes are memoryless, this now decorrelates the events of FC k opening from the event of still having an item waiting. Lemma 2 can then apply, and the analysis finishes by taking an integral. The formal proof is presented below.

Theorem 1. *Algorithm 1 is a* $(1+\ln(q))$*-competitive rounding scheme with runtime* $O(qK)$.

Proof of Theorem 1. The runtime is $O(qK)$ because taking the arg min over $k = 1, \ldots, K$ for all $i = 1, \ldots, q$ is the bottleneck operation in Algorithm 1. Meanwhile, Lemma 1 has already shown that the marginals condition is satisfied. It remains to show that $\Pr[\bigcup_{i=1,\ldots,q}(Z_i = k)] \leq \alpha y_k$ for all k, with $\alpha = 1 + \ln(q)$.

Fix any FC k. For all items i with $u_{ki} > 0$, event $Z_i = k$ can occur only if k lies in the arg min in Algorithm 1, that is, if $\min_{k'} \frac{y_{k'}}{u_{k'i}}E_{k'} \geq \frac{y_k}{u_{ki}}E_k$. We now rewrite this event as follows. Define $S_k^1, S_k^2, \ldots$ to be the arrival times of a Poisson process of rate y_k. More specifically, we will let $S_k^1 = E_k$ and S_k^{j+1} be the sum of S_k^j with an independent exponential random variable of mean $1/y_k$ for all $j \geq 1$. We can derive

$$\begin{aligned}(Z_i = k) &\subseteq \left(\min_{k'} \frac{y_{k'}}{u_{k'i}}E_{k'} \geq \frac{y_k}{u_{ki}}E_k\right) \\ &= \left(\min_{k' \neq k} \frac{y_{k'}}{u_{k'i}}E_{k'} \geq \frac{y_k}{u_{ki}}S_k^1\right) \\ &= \bigcup_{j=1}^{\infty}\left(\min\left\{\min_{k' \neq k} \frac{y_{k'}}{u_{k'i}}E_{k'}, \min_{j' < j} \frac{y_k}{u_{ki}}S_k^{j'}\right\} \geq \frac{y_k}{u_{ki}}S_k^j\right),\end{aligned} \tag{1}$$

where the final equality (1) holds because the events with $j > 1$ never occur (in particular, $\min_{j' < j} \frac{y_k}{u_{ki}}S_k^{j'} \geq \frac{y_k}{u_{ki}}S_k^j$ is impossible because $S_k^{j'} < S_k^j$). The purpose of this vacuous decomposition is to later relax the event (by decreasing the RHS) and then apply the memorylessness property of Poisson processes.

We now take a union bound of events (1) over i and analyze the probability of this union by conditioning on the event that $S_k^j = t$ for any $j \geq 1$ over all times

Downloaded from informs.org by [76.17.100.218] on 07 October 2023, at 13:01 . For personal use only, all rights reserved.

$t \geq 0$. Formally,

$$\begin{aligned}
&\Pr\left[\bigcup_{i:u_{ki}>0}\bigcup_{j=1}^{\infty}\left(\min\left\{\min_{k'\neq k}\frac{y_{k'}}{u_{k'i}}E_{k'}, \min_{j'<j}\frac{y_k}{u_{ki}}S_k^{j'}\right\} \geq \frac{y_k}{u_{ki}}S_k^j\right)\right] \\
&= \int_0^\infty \Pr\left[\bigcup_{i:u_{ki}>0}\left(\min\left\{\min_{k'\neq k}\frac{y_{k'}}{u_{k'i}}E_{k'}, \min_{j'<j}\frac{y_k}{u_{ki}}S_k^{j'}\right\} \geq \frac{y_k}{u_{ki}}t\right)\middle| \exists j : S_k^j = t\right] y_k dt \\
&\leq \int_0^\infty \Pr\left[\bigcup_{i:u_{ki}>0}\left(\min\left\{\min_{k'\neq k}\frac{y_{k'}}{u_{k'i}}E_{k'}, \min_{j'<j}\frac{y_k}{u_{ki}}S_k^{j'}\right\} \geq t\right)\middle| \exists j : S_k^j = t\right] y_k dt \\
&= \int_0^\infty \Pr\left[\bigcup_{i:u_{ki}>0}\min_{k'=1,\ldots,K}\frac{y_{k'}}{u_{k'i}}E_{k'} \geq t\right] y_k dt \\
&\leq y_k \int_0^\infty \min\{qe^{-t}, 1\} dt,
\end{aligned}$$

where the first equality holds because the PDF of the event $(\exists j : S_k^j = t)$ takes value y_k for all t, the first inequality holds because $\frac{y_k}{u_{ki}} \geq 1$, the second equality applies the memorylessness property of Poisson processes, and the final inequality applies Lemma 2 (along with the trivial upper bound of 1). Note that this analysis holds for any FC $k = 1, \ldots, K$. Therefore, the proof is now completed by taking an elementary integral:

$$\begin{aligned}
\int_0^\infty \min\{qe^{-t}, 1\} dt &= \ln(q) + \int_{\ln(q)}^\infty qe^{-t} dt \\
&= \ln(q) + qe^{-\ln(q)} \\
&= 1 + \ln(q). \quad \square
\end{aligned}$$

Remark 1. Our Algorithm 1 and Theorem 1 close the gap that was left open by the correlated rounding scheme of Jasin and Sinha (2015), whose guarantee grew linearly (instead of logarithmically) in the number of items q. Their scheme partitions the $[0,1]$ interval and makes the positive correlation in the FCs assigned very explicit. By contrast, our rounding schemes are based on a "trick" of dilating memoryless random variables, and the positive correlation is implicit. Our trick is designed to facilitate a short analysis, which closes the gap in the correlated rounding problem.

2.2. *d*-Competitive Rounding Scheme

Our modified rounding scheme is specified in Algorithm 2. Relating back to the intuitive description from Section 1.2, $m(i)$ is the first FC that item i would see "forced" open, which it would get assigned to if it was still unassigned at that point. X_{ki} is a random variable denoting the time at which item i sees FC k open, which equals $\frac{y_k}{u_{ki}}E_k$ like before if $k \neq m(i)$. On the other hand, $X_{m(i),i}$ is upper-bounded by $1/u_{m(i),i}$, because that is when item i would see FC $m(i)$ forced open. The final wrinkle is that if FC $m(i)$ were to "naturally" open before it is forced open, then it needs to be hidden from i's view (until it is forced open) with some probability, which is indicated by the random variable H_i. Finally, every item is assigned to the first FC that it sees open, after taking into consideration hiding and forced opening.

Algorithm 2 (*d*-Competitive Rounding Scheme)

For $k = 1, \ldots, K$ **do**
  $E_k \leftarrow$ independent draw from exponential distribution with mean $1/y_k$
end for
for $i = 1, \ldots, q$ **do**
  $m(i) \leftarrow \arg\max_k u_{ki}$
  for $k = 1, \ldots, K, k \neq m(i)$ **do**
    $X_{ki} \leftarrow \frac{y_k}{u_{ki}}E_k$
  end for
  $H_i \leftarrow$ independent draw from Bernoulli distribution with mean $\frac{1-u_{m(i),i}}{1-u_{m(i),i}+u_{m(i),i}e^{1/u_{m(i),i}}-e}$
    ▷ $H_i = 1$ means FC $m(i)$ is hidden from item i until the FC is forced open at time $1/y_{m(i)}$
  $X_{m(i),i} \leftarrow \frac{y_{m(i)}}{u_{m(i),i}}\min\left\{\frac{E_{m(i)}}{1-H_i}, \frac{1}{y_{m(i)}}\right\}$
    ▷ $H_i = 1$ means $\frac{E_{m(i)}}{1-H_i} = \infty$, and hence, $X_{m(i),i} = \frac{1}{u_{m(i),i}}$
  $Z_i \leftarrow \arg\min_{k=1,\ldots,K} X_{ki}$
end for

It can be checked that the probability with which $H_i = 1$ defined in Algorithm 2 does indeed lie in $[0,1]$ for all possible values of $u_{m(i),i} \in (0,1]$. The hiding probability is in fact increasing in $u_{m(i),i}$, which is intuitive because a larger value of $u_{m(i),i}$ implies an earlier forced opening, suggesting that FC $m(i)$ should be hidden more often to prevent it from overfulfilling item i. We now prove that this hiding probability has been calibrated so that the marginals condition is satisfied exactly.

Lemma 3. *Under Algorithm 2,* $\Pr[Z_i = k] = u_{ki}$ *for all* $i = 1, \ldots, q$ *and* $k = 1, \ldots, K$.

Proof of Lemma 3. Fix any item i. We show that $\Pr[Z_i = k] = u_{ki}$ for all $k \neq m(i)$, which would automatically imply that $\Pr[Z_i = m(i)] = 1 - \sum_{k\neq m(i)}\Pr[Z_i = k] = 1 - \sum_{k\neq m(i)}u_{ki} = u_{m(i),i}$. We need to consider two cases: $H_i = 1$ and $H_i = 0$. Hereafter, omit index i.

First, if $H = 1$, then the item does not observe FC m before time $1/u_m$. Therefore, $Z = k$ if and only if X_k is the smallest among random variables $\{X_{k'} : k' \neq m\}$ and also $X_k < 1/u_m$. Recall that $X_{k'}$ is exponentially distributed with mean $1/u_{k'}$ for all $k' \neq m$, and the $X_{k'}$s are independent across k'. Therefore, the probability that $\min_{k'\neq m}X_{k'} < 1/u_m$ is equal to the probability that a Poisson process with rate $\sum_{k'\neq m}u_{k'} = 1 - u_m$ generates an arrival before time $1/u_m$, which occurs w.p. $1 - e^{-(1-u_m)/u_m}$. Conditional on this, the probability that

Downloaded from informs.org by [76.17.100.218] on 07 October 2023, at 13:01 . For personal use only, all rights reserved.

$\min_{k' \neq m} X_{k'} = X_k$ is exactly $\frac{u_k}{1-u_m}$ by the Poisson merging theorem. Therefore,

$$\Pr[Z = k | H = 1] = (1 - e^{-(1-u_m)/u_m}) \frac{u_k}{1-u_m}. \tag{2}$$

Otherwise, if $H = 0$, then the item observes all FCs before time $1/u_m$. In this case, $Z = k$ if and only if X_k is the smallest among all random variables $\{X_{k'} : k' = 1, \ldots, K\}$ and also $X_k < 1/u_m$. By a similar argument as above, the probability that $\min_{k'=1,\ldots,K} X_{k'} < 1/u_m$ is $1 - e^{1/u_m}$, and conditional on this, the probability that $\min_{k'=1,\ldots,K} X_{k'} = X_k$ is u_k. Therefore,

$$\Pr[Z = k | H = 0] = (1 - e^{1/u_m}) u_k. \tag{3}$$

Let η denote $\frac{1-u_m}{1-u_m+u_m e^{1/u_m}-e}$, the probability that $H = 1$. Combining (2) and (3), we derive

$$\begin{aligned}
&\Pr[Z_i = k] \\
&= \eta(1 - e^{-(1-u_m)/u_m}) \frac{u_k}{1-u_m} + (1-\eta)(1 - e^{-1/u_m}) u_k \\
&= u_k \left(1 - e^{-1/u_m} + \eta \left(\frac{1 - e^{-(1-u_m)/u_m}}{1-u_m} - (1 - e^{-1/u_m}) \right) \right) \\
&= u_k \left(1 - e^{-1/u_m} + \eta \cdot \frac{-e^{-(1-u_m)/u_m} + u_m + e^{-1/u_m} - u_m e^{-1/u_m}}{1-u_m} \right) \\
&= u_k \left(1 - e^{-1/u_m} + e^{-1/u_m} \eta \cdot \frac{1 - u_m + u_m e^{1/u_m} - e}{1-u_m} \right) \\
&= u_k
\end{aligned}$$

which completes the proof. □

We now prove our main result for Algorithm 2. We establish the stronger guarantee of $\alpha = \frac{1}{\min_i \max_k u_{ki}}$, which is easily seen to be at most d because $\max_k u_{ki} \geq 1/d$ for all i. The proof sketch is that because of the forced openings, all items are guaranteed to be assigned by time α. Therefore, an FC k can get used only if it opens before time α (because items can only see it open with a delay), which occurs with probability no greater than αy_k.

Theorem 2. *Algorithm 2 is an* $\frac{1}{\min_i \max_k u_{ki}}$*-competitive rounding scheme with runtime* $O(qK)$.

Proof of Theorem 2. The runtime is $O(qK)$, because inside the loop for $i = 1, \ldots, q$ in Algorithm 2, there are three bottleneck operations that each take time $O(K)$: the defining of $m(i)$, the inner loop for k, and the defining of Z_i. Meanwhile, Lemma 3 has already shown that the marginals condition is satisfied. It remains to be shown that $\Pr[\bigcup_{i=1,\ldots,q} (Z_i = k)] \leq \alpha y_k$ for all k, with $\alpha = \frac{1}{\min_i \max_{k'} u_{k'i}}$.

Fix an FC k. If $y_k \geq \min_i \max_{k'} u_{k'i}$, then $\alpha y_k \geq 1$, and there is nothing to prove. Therefore, assume that $y_k < \min_i \max_{k'} u_{k'i}$, and we must show that $\Pr[\bigcup_{i=1,\ldots,q} (Z_i = k)] \leq \alpha y_k$. Because $y_k < \max_{k'} u_{k'i}$ for all i, we know that $k \neq m(i)$ for all i. Thus, we have $X_{ki} = \frac{y_k}{u_{ki}} E_k$ for all i and can write

$$\begin{aligned}
(Z_i = k) &\subseteq \left(\frac{y_k}{u_{ki}} E_k \leq \min_{k'=1,\ldots,K} X_{k'i} \right) \\
&\subseteq \left(\frac{y_k}{u_{ki}} E_k \leq X_{m(i),i} \right) \\
&\subseteq \left(\frac{y_k}{u_{ki}} E_k \leq 1/u_{m(i),i} \right) \\
&= \left(\frac{y_k}{u_{ki}} E_k \leq \frac{1}{\max_{k'} u_{k'i}} \right) \\
&\subseteq \left(\frac{y_k}{u_{ki}} E_k \leq \alpha \right) \\
&\subseteq (E_k \leq \alpha),
\end{aligned}$$

with the final relationship between events holding because $\frac{y_k}{u_{ki}} \geq 1$. Note that the final event is independent of i. Therefore,

$$\Pr\left[\bigcup_{i=1,\ldots,q} (Z_i = k) \right] \leq \Pr[E_k \leq \alpha] = 1 - e^{-\alpha y_k},$$

which is at most αy_k, completing the proof. □

3. Connections with Set Cover

In this section we establish our rounding schemes to be order-optimal in terms of the dependence on q or d by reducing our problem to that of rounding a fractional solution for Set Cover. We first define the Set Cover problem and some basic concepts using our language of items and FC's. We refer to Vazirani (2001) for further background.

Problem 1 (Weighted Set Cover). There are items $i = 1, \ldots, q$ to be covered by FCs $k = 1, \ldots, K$. Each FC k requires a fixed cost of c_k to open and, if opened, can cover all items in a set $U_k \subseteq \{1, \ldots, q\}$. The objective is to find a collection of FCs to open that covers all the items and minimizes the sum of fixed costs paid for opening FCs. The *sparsity* of the instance is defined as $d := \max_i |\{k : i \in U_k\}|$, the maximum number of different FCs that an item i can be covered by.

Definition 2 (Set Cover Linear/Integer Programs). The following Integer Program is called the *Set Cover IP*. In it, binary variable y_k represents FC k being opened. It is an equivalent formulation of the Weighted Set Cover problem:

$$\min \sum_{k=1}^{K} c_k y_k$$

$$\text{s.t.} \sum_{k: i \in U_k} y_k \geq 1 \qquad \forall i = 1, \ldots, q \tag{4}$$

$$y_k \in \{0, 1\} \qquad \forall k = 1, \ldots, K \tag{5}$$

Meanwhile, the *Set Cover LP* is defined as the relaxation of the Set Cover IP, with constraint (5) changed to $y_k \in [0, 1]$, for all $K = 1, \ldots, K$.

Downloaded from informs.org by [76.17.100.218] on 07 October 2023, at 13:01 . For personal use only, all rights reserved.

We now define the problem of rounding a fractional solution for Set Cover in a way that is analogous to an α-competitive rounding scheme, except that we will call it an α-competitive "covering" scheme instead.

Definition 3 (α-Competitive Covering Scheme). For $\alpha \geq 1$, an *α-competitive covering scheme* is a method for constructing random variables $Y_1, \ldots, Y_K \in \{0,1\}$ satisfying

$$\sum_{k:i\in U_k} Y_k \geq 1 \qquad \forall i = 1, \ldots, q, \ w.p.\ 1 \tag{6}$$

$$\mathbb{E}[Y_k] \leq \alpha \cdot y_k \qquad \forall k = 1, \ldots, K, \tag{7}$$

given any feasible solution $(y_k)_{k=1}^K$ to the Set Cover LP.

We now show that coming up with α-competitive rounding schemes is a harder problem than coming up with α-competitive covering schemes.

Lemma 4. *An α-competitive rounding scheme can be efficiently applied as an α-competitive covering scheme. Moreover, any dependence of α on the parameters q or d translates over directly.*

Proof of Lemma 4. Take any instance of Set Cover and a feasible solution $(y_k)_{k=1}^K$ to its LP. For each item i, arbitrarily set $u_{ki} \in [0, y_k]$ for each FC k that can cover it so that $\sum_{k:i\in U_k} u_{ki} = 1$. We note that this is always possible because $y_k \geq 0$ and $\sum_{k:i\in U_k} y_k \geq 1$ by (4). Meanwhile, set $u_{ki} = 0$ if $i \notin U_k$.

The marginal distributions $(u_{k1})_{k=1}^K, \ldots, (u_{kn})_{k=1}^K$ now define an instance for an α-competitive rounding scheme, with the same number of items q and a sparsity d that is no greater than before. We apply the α-competitive rounding scheme that is assumed to exist on this instance and define random variables $Y_k = \mathbb{1}(\bigcup_i (Z_i = k))$ for all $k = 1, \ldots, K$. By the definition of a rounding scheme, for each item i, we know that $Z_i = k$ is true for some index $k \in \{1, \ldots, K\}$, with $k \in U_k$ because otherwise $u_{ki} = 0$. Therefore, $Y_k = 1$ for this index k and condition (6) for the covering scheme is satisfied. Meanwhile, applying the definition of an α-competitive rounding scheme, we have

$$\mathbb{E}[Y_k] = \Pr\left[\bigcup_i (Z_i = k)\right] \leq \alpha \cdot \max_i u_{ki} \leq y_k.$$

We conclude that condition (7) for the covering is satisfied. We also note that if α depends on the sparsity parameter d, then the same guarantee continues to hold under the old sparsity parameter for Set Cover, which is no less than d, completing the proof. □

3.1. Negative Results for α-Competitive Rounding Schemes

Equipped with Lemma 4, we can now translate hardness results for the α-competitive covering scheme problem into hardness results for the α-competitive rounding scheme problem.

Corollary 1 (of Lemma 4). *An α-competitive covering scheme must have $\alpha = \Omega(\log(q))$ (Vazirani 2001, Ex. 13.4). Therefore, an α-competitive rounding scheme must also have $\alpha = \Omega(\log(q))$. Consequently, the $(1 + \ln(q))$-competitive rounding scheme established in Theorem 1 achieves the order-optimal dependence on q.*

Proposition 1. *An α-competitive covering scheme must have $\alpha \geq d$, where d denotes the sparsity of the instance.*

Proof of Lemma 1. Consider a Set Cover instance with d fixed, K large, and one item for each subset of $\{1, \ldots, K\}$ of size d. Each such item can only be covered by the d FCs in its corresponding subset, with the total number of items being $q = \binom{K}{d}$. The sparsity of this instance is d by definition.

Setting $y_k = 1/d$ for all $k = 1, \ldots, K$ forms a feasible solution to the Set Cover LP, because $|\{k : i \in U_k\}| = d$ for all items i, and hence, LP constraints (4) are satisfied. On the other hand, any α-competitive covering scheme must set $\sum_{k=1}^K Y_k > K - d$ w.p. 1, because otherwise there would be an uncovered item, violating (6). Using the linearity of expectation, we derive

$$K - d \leq \sum_{k=1}^K \mathbb{E}[Y_k] \leq \sum_{k=1}^K \alpha \cdot y_k = K\alpha \frac{1}{d},$$

with the second inequality coming from (7). Therefore, $\alpha \geq d\left(1 - \frac{d}{K}\right)$, with $\frac{d}{K}$ approaching for arbitrarily large K, completing the proof. □

Corollary 2 (of Lemma 4 and Proposition 1). *An α-competitive rounding scheme must have $\alpha \geq d$. Consequently, the d-competitive rounding scheme established in Theorem 2 achieves the optimal (not just order-optimal) dependence on d.*

4. Instance-Optimal Rounding Schemes

The $(1 + \ln(q))$-competitive and d-competitive rounding schemes discussed in Sections 2 and 3 were only order-optimal in the worst case. For a particular instance given by q marginals over $\{1, \ldots, K\}$, one could also consider the problem of computing the maximum guarantee α and rounding scheme that satisfies the marginal frequency constraints.

We formulate this problem using an LP with the following variables. For all subsets S of the FCs $\{1, \ldots, K\}$, let $z(S)$ denote the probability that exactly the set of FCs in S get used. For all $S \subseteq \{1, \ldots, K\}$, FC's $k \in S$, and items i, let $u_{ki}(S)$ denote the probability that the set of FCs in S get used and that item i is fulfilled from FC $k \in S$. The problem of minimizing α in an α-competitive rounding

Downloaded from informs.org by [76.17.100.218] on 07 October 2023, at 13:01 . For personal use only, all rights reserved.

scheme for this particular instance can then be formulated as

$$\min \alpha \tag{8}$$

$$\text{s.t.} \sum_{k\in S} u_{ki}(S) = z(S) \qquad \forall S, i = 1,\dots,q \tag{9}$$

$$\sum_{S} u_{ki}(S) = u_{ki} \qquad k = 1,\dots,K, i = 1,\dots,q \tag{10}$$

$$\sum_{S\ni k} z(S) \le \alpha \cdot y_k \qquad \forall k = 1,\dots,K \tag{11}$$

$$\sum_{S} z(S) = 1 \tag{12}$$

$$z(S) \ge 0 \qquad \forall S \tag{13}$$

$$u_{ki}(S) \ge 0 \qquad \forall S, k \in S, i = 1,\dots,q, \tag{14}$$

where constraints (9) enforce that every item i must be fulfilled from exactly one FC on each subset S, constraints (10) and (11) enforce the marginal and α-competitive properties of a rounding scheme, constraints (12) and (13) enforce that exactly one subset S is selected, and, last but not least, (14) ensures that there is only a variable $u_{ki}(S)$ if $k \in S$.

Our LP has size $O(nK2^K)$, which is exponential in K but tractable if K is a fixed constant. Jasin and Sinha (2015) derived an exponential-sized LP for the same purpose, except instead there is a variable for every possible *mapping* from $\{1,\dots,q\}$ to $\{1,\dots,K\}$, for which there are K^q possibilities. Our LPs are more practical in situations where K is small but q is large, which is the case in the application of, for example, Zhao et al. (2020).

5. α-Competitive Rounding Scheme Applied to Dynamic Fulfillment

In this section, we recap the general dynamic fulfillment problem from Jasin and Sinha (2015) and formalize the implication of our α-competitive rounding schemes for the overall problem.

5.1. Problem Definition

There is a horizon consisting of time steps $t = 1,\dots,T$, during which items $i = 1,\dots,n$ are fulfilled from FCs $k = 1,\dots,K$. Each item i starts with b_{ki} units of inventory at each FC k, with the end of the horizon representing the time at which inventories are replenished again. Orders come from one of regions $j = 1,\dots,J$ and are described by a subset[1] of items $a \subseteq \{1,\dots,n\}$ that was just purchased. During each time step, up to one order arrives, which is from region j and is for subset a with probability λ_j^a, with $\sum_{a,j}\lambda_j^a \le 1$. As is standard in revenue management, we assume a granular division of time such that at most one order can arrive during each time step. Also, as justified in Jasin and Sinha (2015), we assume that orders cannot contain more than one of any item, and we assume a small universe of possible subsets a. We let c_{kij}^{unit} denote the variable cost of fulfilling one unit of item i from FC k to location j and let c_{kj}^{fixed} denote the fixed cost of sending a package (containing one or more items) from FC k to location j.

The goal is to dynamically decide the FCs to use to fulfill the items in each order that arrives over the time horizon, to minimize total expected cost. Note that if an FC k is used to fulfill a subset $a' \subseteq a$ of an order from a location j, then the cost required to send that package is $c_{kj}^{\text{fixed}} + \sum_{i\in a'} c_{kij}^{\text{unit}}$. All items in each arriving order must be fulfilled from some FC, where we assume the existence of a null FC 0 with infinite inventory so that this is always feasible, with c_{0ij}^{unit} denoting the "shortage" cost of failing to fulfill one unit of item i to region j.

5.2. LP Benchmark

Solving for the optimal dynamic fulfillment policy using dynamic programming is intractable, because the state space is exponential in the number of items. Thus, the following "deterministic" LP benchmark[2] is often used to derive heuristic policies and bound their suboptimality relative to the optimal dynamic programming policy.

$$\mathsf{DLP} := \min \sum_{a,k,j} T\lambda_j^a \left(\sum_{i\in a} c_{kij}^{\text{unit}} u_{kij}^a + c_{kj}^{\text{fixed}} y_{kj}^a \right)$$

$$\text{s.t.} \sum_{j}\sum_{a\ni i} T\lambda_j^a u_{kij}^a \le b_{ki} \qquad \forall k, i$$

$$\sum_{k} u_{kij}^a = 1 \qquad \forall a, j, i \in a$$

$$y_{kj}^a \ge u_{kij}^a \ge 0 \qquad \forall a, k, j, i \in a$$

In the linear program defining DLP, for any subset a of items ordered from any region j, variable u_{kij}^a represents the proportion of times that item $i \in a$ should be fulfilled from FC k, with constraint $\sum_k u_{kij}^a = 1$ for each such item i in the order. Meanwhile, variable y_{kj}^a represents the probability that a FC k would have to be used at all, which is constrained to be at least u_{kij}^a for any single item $i \in a$. Note that in an optimal solution we can always assume that $y_{kj}^a = \max_{i\in a} u_{kij}^a$ for all a, k, j. These variables u_{kij}^a and y_{kj}^a correspond to our variables u_{ki} and y_k from earlier, where we had dropped scripts a, j to focus on a single multi-item order from a single region.

Moreover, the first constraint enforces that the expected number of times any FC k fulfills any item i (to any region j, as part of any subset a containing i) does not exceed its starting inventory b_{ki}. Finally, the objective value defining DLP represents the total expected cost of the LP benchmark over the time horizon, accounting for unit costs and fixed costs as well as shortage costs (recalling that there is a null FC $k = 0$). This interpretation of DLP intuitively leads to the following lemma.

Lemma 5 (Jasin and Sinha 2015). *For any instance of the problem, the expected cost paid by any dynamic fulfillment policy must be at least the value of* DLP *for that instance.*

Downloaded from informs.org by [76.17.100.218] on 07 October 2023, at 13:01 . For personal use only, all rights reserved.

5.3. Randomized Fulfillment Algorithm and Reduction Result

In light of the interpretation of the linear program defining DLP above, Jasin and Sinha (2015) also used it to derive the following *randomized fulfillment* heuristic. First, we solve the LP, hereafter using u^a_{kij}, y^a_{kj} to refer to a fixed optimal solution. At each time step $t = 1, \ldots, T$, if an order for subset a comes from region j, the heuristic policy randomly chooses an FC k to fulfill each item $i \in a$ according to probabilities u^a_{kij}, independently across time steps, without adapting at all to the remaining inventory. If the chosen FC for an item has stocked out, then that item is simply not fulfilled (i.e., the null FC is used).

This randomized fulfillment heuristic that does not rely on real-time inventory information has been shown to perform well asymptotically, although its theoretical guarantee depends on how exactly FCs are chosen to fulfill items during each time step, namely, the α-competitive rounding scheme that is used. Jasin and Sinha (2015) showed that the unit and shortage costs paid by the randomized fulfillment heuristic are asymptotically optimal relative to the DLP, but the bottleneck is the fixed cost, where every time an order for subset a comes from region j (regardless of asymptotics), the cost paid could be α times as much as the DLP. Here, α depends on a and j, and using the correlated rounding schemes from Theorems 1 and 2 in this paper in conjunction with the results from Jasin and Sinha (2015), we can always guarantee an α-competitive rounding scheme where

$$\alpha = \min\left\{1 + \ln(|a|), \left(\min_{i \in a} \max_k u^a_{kij}\right)^{-1}, B(|a|)\right\} \tag{15}$$

and $B(\cdot)$ is the function from Jasin and Sinha (2015).

Jasin and Sinha (2015) showed that the asymptotic cost paid by the randomized fulfillment heuristic relative to DLP, assuming it chooses the correlated rounding scheme corresponding to the smallest argument in (15) whenever any subset a is ordered from any region j, is a weighted average of expression (15) over a and j. To formally state this result, we need to finally define what "asymptotic" means. Here, one considers a scaling regime where for any fixed instance and any $\theta \geq 0$, the "scaled instance" is defined to the one where the horizon length T has been replaced by θT, whereas each starting inventory b_{ki} has also been replaced by θb_{ki}. Let $\mathsf{DLP}(\theta)$ denote the optimal objective value DLP on the instance scaled by θ, and let $\mathsf{ALG}(\theta)$ denote the expected cost paid by the randomized fulfillment heuristic on the same scaled instance. The following is then implied by the proof of Theorems 1 and 2 from Jasin and Sinha (2015) (see Jasin and Sinha 2015, p. ec5), combined with our discussion above.

Theorem 3 (Jasin and Sinha 2015). *In the multi-item e-commerce fulfillment problem,*

$$\lim_{\theta \to \infty} \frac{\mathsf{ALG}(\theta)}{\mathsf{DLP}(\theta)} \leq \frac{\sum_{a,k,j} \lambda^a_j c^{\text{fixed}}_{kj} y^a_{kj} \min\{1 + \ln(|a|), (\min_{i \in a} \max_{k'} u^a_{k'ij})^{-1}, B(|a|)\}}{\sum_{a,k,j} \lambda^a_j c^{\text{fixed}}_{kj} y^a_{kj}}. \tag{16}$$

Because any fulfillment policy must pay cost at least $\mathsf{DLP}(\theta)$ by Lemma 5, this shows that the randomized fulfillment heuristic cannot be worse than the optimal dynamic program by a factor greater than the RHS of (16). The RHS of (16) is a weighted average of the minimum of the guarantees from three different rounding schemes and was referred to as β in the Introduction. In order to achieve this, the randomized fulfillment heuristic must choose for every incoming order a the rounding scheme with the best guarantee among that of Jasin and Sinha (2015) and our two new ones. Simpler bounds can also be derived by relaxing the RHS of (16); for example,

$$\begin{aligned} \lim_{\theta \to \infty} \frac{\mathsf{ALG}(\theta)}{\mathsf{DLP}(\theta)} &\leq \frac{\sum_{a,k,j} \lambda^a_j c^{\text{fixed}}_{kj} y^a_{kj} (1 + \ln(|a|))}{\sum_{a,k,j} \lambda^a_j c^{\text{fixed}}_{kj} y^a_{kj}} \\ &\leq \frac{\sum_{a,k,j} \lambda^a_j c^{\text{fixed}}_{kj} y^a_{kj} \max_{a'} \{1 + \ln(|a'|)\}}{\sum_{a,k,j} \lambda^a_j c^{\text{fixed}}_{kj} y^a_{kj}} \\ &= 1 + \ln\left(\max_{a'} |a'|\right). \end{aligned} \tag{17}$$

Jasin and Sinha (2015, Thm. 2) proved the same guarantee as Theorem 3, except with the $\min\{\cdot\}$ replaced by just $B(|a|)$, whereas Zhao et al. (2020) proved the same result where the upper bound on the RHS is 1 (i.e., it proves asymptotic optimality) if there are only two FCs in the network. We emphasize that all of these asymptotic guarantees which have eliminated the unit and shortage costs only hold if the LP inventory constraints are satisfied in expectation at every time step, justifying why all of these papers study correlated rounding schemes.

Finally, we argue that the corollary of Theorem 3 depicted in (17), arising from the $(1 + \ln(|a|))$-competitive rounding scheme in our paper, is in fact tight. In Section 3, we had already shown that $1 + \ln(|a|)$ is the best possible guarantee for the correlated rounding problem, but here we show computational hardness for dynamic fulfillment, again through a reduction to Set Cover.

Proposition 2. *For any positive integer q, it is NP-hard to solve dynamic fulfillment using total cost less than $(1 - o(1))\ln(q)$ times the optimal cost (given by a computationally unconstrained dynamic program), even if all orders have size q and even on the scaled instance as $\theta \to \infty$.*

Downloaded from informs.org by [76.17.100.218] on 07 October 2023, at 13:01 . For personal use only, all rights reserved.

Proof of Proposition 2. Given any instance of (unweighted) Set Cover, as defined in Problem 1, we show how it can be represented by an instance of the dynamic fulfillment problem, as defined in this section. Recall that q, K were the number of items, sets, respectively, in the Set Cover problem. Consider a dynamic fulfillment problem with base time horizon $T = 1$, one region, a deterministic order type a of size q, and K FCs. All unit shipping costs are 0, and fixed shipping costs are 1. Starting inventory b_{ki} equals 1 if $i \in U_k$ and 0 otherwise. Let θ, the positive integer by which both the time horizon and starting inventories are scaled, be arbitrary.

Because of the inventory configuration, an item i can only be feasibly assigned to an FC k if $i \in U_k$, that is, if item i was covered by set k. The assignment of an item i to any feasible FC k that is already being used (i.e., whose fixed shipping cost is being paid) is irrelevant, because all such FCs would start with θ units of item i, and the inventory constraint is not binding. Therefore, the decision at every period in the dynamic fulfillment problem is identical and equivalent to the minimization problem of the given Set Cover instance, where the goal is to choose a minimum set of FCs to use such that each of the q items in the order can be assigned. The objective functions also coincide.

Therefore, if it were possible to solve dynamic fulfillment using total cost less than $(1 - o(1))\ln(q)$ times the optimum, then it would be possible to solve Set Cover using total cost less than $(1 - o(1))\ln(q)$ times the optimum. By Dinur and Steurer (2014), the latter statement would imply that P = NP. Because the scaling parameter θ was arbitrary, the proof is now complete. □

Remark 2. Very recently, Amil et al. (2022) proposed an eye-opening approach to the dynamic fulfillment problem that still uses the randomized fulfillment heuristic but solves a bigger LP that is tighter than DLP. This bigger LP explicitly models the different "methods" by which the items in an order can be split across FCs and fulfilled, obfuscating the need for a correlated rounding scheme. These authors showed that the value of ALG(θ) relative to their LP approaches 1 as $\theta \to \infty$, achieving asymptotic optimality and seemingly contradicting Proposition 2. However, generally there could be exponentially many ways to split a q-item order across K FCs, so without restrictions on the methods, their LP cannot be solved in polynomial time (unless P = NP). Therefore, in unrestricted settings with large orders, solving the smaller LP and using our correlated rounding procedure is still highly relevant.

6. Numerical Study

We test our α-competitive rounding schemes on the general multi-item dynamic fulfillment problem formalized in Section 5. We construct instances aimed to model the operations of a large e-tailer in the continental United States, following the setup of Jasin and Sinha (2015) as closely as possible. Our code is in Julia, uses the JuMP (Dunning et al. 2017) package, and has been made publicly available at https://github.com/Willmasaur/multi_item_e_commerce_fulfillment.

6.1. Regions, Fulfillment Centers, Costs

We allow orders to arrive from regions corresponding to the 99 largest metropolitan areas in the United States, excluding Honolulu, Hawaii. The arrival rate from each region is scaled by its 2022 population according to the U.S. Cities Database on https://simplemaps.com/data. Meanwhile, we take the 10 largest Amazon.com, Inc. fulfillment centers that were operational[3] as of 2015 and assume that all items are shipped from one of these centralized FCs. Following Jasin and Sinha (2015), the fixed cost of packaging a box at any FC k for any region j is $c_{kj}^{\text{fixed}} = 8.759$, whereas the cost of shipping a single item i from any FC k to any region j is $c_{kij}^{\text{unit}} = 0.423 + 0.000541\text{dist}_{kj}$, where dist_{kj} is the air distance between FC k and region j in miles. Not fulfilling an item i costs double the maximum distance; see Jasin and Sinha (2015) for details.

We note that our city populations and FC locations may differ from Jasin and Sinha (2015), because the exact sources they cited are no longer publicly available. We also procedurally diverge from Jasin and Sinha (2015) by always selecting the largest cities and fulfillment centers, whereas they select randomly when fewer than 99 cities or fewer than 10 FCs are needed. We believe this to generate a more interesting smaller network, because the 10 largest cities are spread out across the corners, whereas the five FCs are located in the middle, resulting in difficult fulfillment decisions where a city can be "nearby" to multiple FCs (see the data files provided with our code for details).

6.2. Order Types, Demand Rates, Starting Inventories

Order types a each denote a subset of size up to $n_{\max}$ from a universe of n items. For each size in $1, \ldots, n_{\max}$, there are n_{per} fixed order types, each of which is a subset of the n items drawn uniformly at random with the correct size. There is also an order type with size 0, which represents the lack of a customer arrival at a time step. Note that the total number of order types is $1 + n_{\max}n_{\text{per}}$, which we denote using Q.

The demand probabilities are first split randomly between the order sizes $0, 1, \ldots, n_{\max}$ and then for each size, split randomly between the types with that size. This yields a Q-dimensional probability vector, that is, a vector whose entries are nonnegative and sum to 1. Then, a QJ-dimensional probability vector is constructed by further splitting each order type among the metropolitan areas according to their populations. This vector

Downloaded from informs.org by [76.17.100.218] on 07 October 2023, at 13:01 . For personal use only, all rights reserved.

$(\lambda_j^a)_{a,j}$ is then used as input for the dynamic fulfillment problem.

Finally, to determine starting inventories, each FC k first randomly decides whether to carry each item i independently with probability p_{carry}. Then, for each region j, the closest FC k that carries each item i is identified as $\mathsf{closest}_{i,j}$. For an item i, its "demand" at an FC k is

$$\mathsf{dem}_{k,i} = \sum_{a \ni i} \sum_j \mathbb{1}(\mathsf{closest}_{i,j} = k) \cdot \lambda_j^a,$$

where we sum over all queries a containing a copy of item i and consider only the regions j for which FC k is identified as the closest when summing over arrival probabilities λ_j^a. Given these values, starting inventories are then placed so that $b_{ki} = T\mathsf{dem}_{k,i} + z_{\text{safety}}\sqrt{T\mathsf{dem}_{k,i}(1-\mathsf{dem}_{k,i})}$ for all k and i, where we note that the total demand for item i closest to FC k over T time steps is binomially distributed, with mean $T\mathsf{dem}_{k,i}$ and variance $T\mathsf{dem}_{k,i}(1-\mathsf{dem}_{k,i})$. The formula for b_{ki} is the ideal inventory level to start with according to a Newsvendor model, with safety stock multiplier z_{safety} set to 0.5 for all items.

We note that our procedures for randomly generating order types, demand rates, and carrying decisions follow (Jasin and Sinha 2015, EC.3) exactly, in which these methods are justified. The details of these methods can also be found in our code.

6.3. Algorithms

Like Jasin and Sinha (2015), we test the Myopic fulfillment policy as a baseline, which fulfills each item from the closest FC that carries it, not accounting for split orders and minimizing the number of boxes shipped. We then consider four different algorithms following the randomized fulfillment heuristic described in Section 5:

- Indep: independent rounding, as described in Jasin and Sinha (2015);
- JS: correlated rounding scheme of Jasin and Sinha (2015) based online partitions;
- Dilate: $(1+\ln(|a|))$-competitive scheme based on dilated opening times (Section 2.1);
- ForceOpen: d-competitive scheme based on dilated times and forced openings (Section 2.2).

Jasin and Sinha (2015) compared Indep and JS to the Myopic fulfillment policy; we additionally compare our new correlated rounding schemes Dilate and Force Open.

6.4. Experimental Setups

We consider two experimental setups. First, in **Section 6.5,** we let the number of regions, FCs, items, and time steps be $J = 10, K = 5, n = 20, T = 10^5$, respectively. The queries and starting inventories are generated with $n_{\max}$ varying in {2, 5, 10}, n_{per} fixed to 5, and p_{carry} fixed to 0.75. We note that when $n_{\max} = 5$, this is exactly the "base case" simulated in Jasin and Sinha (2015). We vary $n_{\max}$ to see how different rounding schemes handle different order sizes.

We consider a bigger network in **Section 6.6**, where the number of regions, FCs, and items are $J = 99, K = 10, n = 100$, respectively. These represent the largest values considered in Jasin and Sinha (2015), and we also increase T to 10^6 to better capture asymptotic performance. The queries are generated with $n_{\max}$ and n_{per} increased to 10. Meanwhile, we vary p_{carry} in $\{0.25, 0.5, 0.75\}$ to investigate how in a big sparse network with a small value p_{carry}, the problem can still be easy, and in particular ForceOpen can perform well because d is small.

For each experimental setup, we randomly generate 30 instances and then randomly generate 30 arrival sequences for each instance. We use the same arrival sequences for every algorithm to minimize the discrepancy caused by variance in arrival sequences. We fix z_{safety} to be 0.5 throughout our experiments. All of these aspects match what is done in Jasin and Sinha (2015).

6.5. Performance on Smaller Network with Varying Order Size

We consider the first experimental setup with the smaller network, generating 30 random instances for each value of $n_{\max}$ in {2, 5, 10}. For each instance, we consider the benchmark DLP described in Section 5, which is a lower bound on the cost of any fulfillment algorithm. We draw 30 arrival sequences to test the performance of the five specific algorithms discussed earlier and compute the average cost of each algorithm over these 30 arrival sequences. We consider how much greater this average cost is than the value of DLP for that instance, expressed as a percentage. The average of these "loss" percentages over the 30 instances is then reported in Table 1 for each algorithm. We also report the average runtime[4] of each algorithm, which we note is the total runtime used to evaluate the 30 arrival sequences for an instance, averaged over instances. Finally, we report for each algorithm the average number of FCs used per order (not counting the "null" FC 0).

6.5.1. Observations from Results in Table 1.

Our algorithms perform favorably in comparison with Myopic, Indep, and JS. Indeed, they pay marginally more cost than JS when $n_{\max} = 2$ and overtake JS as soon as $n_{\max} = 5$, that is, orders have sizes between 1 and 5. This is surprising in that the theoretical guarantee of JS is better for the values of n in this range. Similar improvements are observed in terms of the average number of FCs used per order. Also, we note that the losses of 8.3% and 8.6% for Dilate are relative to an (unreasonable) LP benchmark that does not face any stochastic fluctuation; the loss relative to an actual fulfillment policy that can be implemented (e.g., the optimal dynamic programming policy, given the exponential time required to

Downloaded from informs.org by [76.17.100.218] on 07 October 2023, at 13:01 . For personal use only, all rights reserved.

Table 1. Performance and Runtime Metrics for the 5 Different Algorithms Under the 3 Different Values of n_{max}

	Myopic	Indep	JS	Dilate	ForceOpen
$n_{max} = 2$ Avg. Loss	4.3%	3.1%	**2.3%**	2.4%	2.4%
$n_{max} = 5$ Avg. Loss	12.9%	14.9%	9.3%	**8.3%**	8.9%
$n_{max} = 10$ Avg. Loss	17.7%	16.5%	11.7%	**8.6%**	9.4%
$n_{max} = 2$ Runtime per Instance	0.33s	0.38s	1.17s	0.39s	0.43s
$n_{max} = 5$ Runtime per Instance	0.52s	0.59s	3.37s	0.62s	0.72s
$n_{max} = 10$ Runtime per Instance	0.84s	1.03s	9.12s	1.09s	1.32s
$n_{max} = 2$ Avg. FC's per Order	0.68	0.67	**0.66**	**0.66**	**0.66**
$n_{max} = 5$ Avg. FC's per Order	1.29	1.22	1.16	**1.15**	1.16
$n_{max} = 10$ Avg. FC's per Order	1.73	1.6	1.51	**1.44**	1.46

Note. The best (smallest) performances are bolded for each row.

compute it) would be much smaller. For this reason, we consider the numbers in Table 1 to be more useful for comparing algorithms than for evaluating absolute performance.

A further, perhaps more salient feature of our algorithms is their simplicity and interpretability. As evidenced in our code, the rounding scheme in Dilate (Algorithm 1) takes 10 lines to write, whereas the rounding scheme in JS took us 100 lines. Also, the average runtime per instance for Dilate is better than JS by a factor of 5–10. This seemingly innocuous difference on the smaller network becomes more pronounced on the bigger network, as we see next.

6.6. Performance on Bigger Network with Varying Fulfillment Flexibility

We consider the second experimental setup described in Section 6.4. We report average losses and runtimes for each of the five algorithms, in the same way as defined in Section 6.5. We generate 30 random instances for each value of p_{carry} in $\{0.25, 0.5, 0.75\}$ and report the averages in Table 2.

We note that p_{carry} is a measure of fulfillment flexibility in that a higher value of p_{carry} leads to more FCs being able to fulfill each item and hence, more flexibility in the network. Generally, this results in a harder fulfillment problem, with a larger value of d, which we recall denotes the maximum number FCs carrying any item. A lower value of p_{carry}, on the other hand, results in a smaller d and a better guarantee for ForceOpen.

6.6.1. Observations from Results in Table 2.

In this bigger network that also has larger order sizes, all algorithms perform worse. Myopic performs particularly poorly with large order sizes because it will likely always split the order (because not all FCs stock all items). We can see a greater separation between the performance of our algorithms, Dilate and ForceOpen, versus the performance of the other algorithms. And although we had always observed ForceOpen to both be more complex and perform slightly worse than Dilate, we now see that when $p_{carry} = 0.25$, it in fact performs better. This is related to its theoretical guarantee; the value of d tends to be smaller when $p_{carry} = 0.25$, because each item in expectation is carried in only 2.5 FCs.

There is also now a factor-10 speedup in the runtime of our algorithms compared with JS, which means that the time to finish per instance is on the order of tens of seconds instead of minutes.

6.7. Takeaways from Numerical Study

Under the randomized fulfillment heuristic of Jasin and Sinha (2015), one should generally default to Dilate to perform correlated rounding because it is simple to implement, is fast to run, and performs either the best or

Table 2. Performance and Runtime Metrics for the 5 Different Algorithms Under the 3 Different Values of p_{carry}

	Myopic	Indep	JS	Dilate	ForceOpen
$p_{carry} = 0.25$ Avg. Loss	34.8%	10.2%	7.6%	5.6%	**5.3%**
$p_{carry} = 0.50$ Avg. Loss	26.7%	23.4%	17.7%	**12.6%**	13.1%
$p_{carry} = 0.75$ Avg. Loss	22.3%	34.2%	23.2%	**16.1%**	17.7%
$p_{carry} = 0.25$ Runtime per Instance	11.23s	15.43s	162.31s	14.24s	16.88s
$p_{carry} = 0.50$ Runtime per Instance	11.89s	18.25s	162s	17.01s	19.25s
$p_{carry} = 0.75$ Runtime per Instance	13.01s	19.33s	169.27s	18.59s	22.09s
$p_{carry} = 0.25$ Avg. FC's per Order	3.22	1.31	1.27	**1.22**	1.24
$p_{carry} = 0.50$ Avg. FC's per Order	2.4	1.98	1.87	**1.76**	1.78
$p_{carry} = 0.75$ Avg. FC's per Order	1.71	1.79	1.62	**1.50**	1.53

Note. The best (smallest) performances are bolded for each row.

Downloaded from informs.org by [76.17.100.218] on 07 October 2023, at 13:01 . For personal use only, all rights reserved.

close to the best across the different setups. For orders with two items, JS may perform slightly better. In large, sparse networks where each item is carried at very few FCs, ForceOpen may perform slightly better.

7. Conclusion

We provide the first improvements to the celebrated correlated rounding procedure of Jasin and Sinha (2015) for the problem of multi-item e-commerce order fulfillment. We derive rounding schemes with guarantees of $1 + \ln(q)$ and d respectively, where q is the number of items in the order and d is the maximum number of fulfillment centers containing any item. The first of these guarantees improves the guarantee of $\approx q/4$ from Jasin and Sinha (2015) by an order of magnitude in terms of the dependence on q. We also show both of our guarantees to be tight by deriving new relationships with the Set Cover problem. Testing under a realistic setup originated by Jasin and Sinha (2015), we find the improvement provided by our new rounding schemes to in fact be greater than what their theoretical guarantees suggest.

Acknowledgments

An earlier version of this paper was accepted to the Manufacturing & Service Operations Management (MSOM) conference Special Interest Group (SIG) for Supply Chain Management, 2022, whose anonymous reviewers provided excellent comments. The author also thanks Levi DeValve, Stefanus Jasin, Aravind Srinivasan, Yehua Wei, and Linwei Xin for background information about this problem.

Endnotes

[1] This section introduces the broader problem with n items in the universe and orders a, which are subsets of $\{1,\ldots,n\}$. The earlier Sections 2 – 4 are applied by focusing on a single-order a, letting $q := |a|$ and renumbering the items in a to be $1,\ldots,q$, ignoring all other items. Generally, in e-commerce fulfillment, n can be much larger than q.

[2] This is identical to the linear program defining $\tilde{J}_{DLP}$ (Jasin and Sinha 2015, p. 1340), except that we have let u^a_{kij} and y^a_{kj} represent their variables U^a_{kij} and Y^a_{kj} divided by $T\lambda^a_j$, respectively.

[3] Compiled from the information at https://www.mwpvl.com/html/amazon_com.html; available with our code.

[4] See our code for the exact timing functions used. The time (in seconds) was measured on a Dell Latitude 5510 laptop with an Intel(R) Core(TM) i7-10810U CPU @ 1.10GHz processor and 32GB of RAM.

References

Acimovic J, Farias VF (2019) The fulfillment-optimization problem. Operations Research & Management Science in the age of analytics. *Informs* 218–237.

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

Amil A, Makhdoumi A, Wei Y (2022) Multi-item order fulfillment revisited: LP formulation and prophet inequality. Preprint, submitted August 4, https://dx.doi.org/10.2139/ssrn.4176274.

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

Dinur I, Steurer D (2014) Analytical approach to parallel repetition. *Proc. 46th Annu. ACM Sympos. Theory Comput.* (ACM, New York) 624–633.

Dunning I, Huchette J, Lubin M (2017) Jump: A modeling language for mathematical optimization. *SIAM Rev.* 59(2):295–320.

Jasin S, Kumar S (2012) A re-solving heuristic with bounded revenue loss for network revenue management with customer choice. *Math. Oper. Res.* 37(2):313–345.

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

Lei Y, Jasin S, Sinha A (2018) Joint dynamic pricing and order fulfillment for e-commerce retailers. *Manufacturing Service Oper. Management* 20(2):269–284.

Lei Y, Jasin S, Uichanco J, Vakhutinsky A (2021) Joint product framing (display, ranking, pricing) and order fulfillment under the multinomial logit model for e-commerce retailers. *Manufacturing Service Oper. Management.* 24(3):1529–1546.

Motwani R, Raghavan P (1995) *Randomized Algorithms* (Cambridge University Press, Cambridge, UK).

Raghavan P, Tompson CD (1987) Randomized rounding: A technique for provably good algorithms and algorithmic proofs. *Combinatorica* 7(4):365–374.

Talluri KT, Van Ryzin G (2004) *The Theory and Practice of Revenue Management*, volume 1 (Springer, New York).

Vazirani VV (2001) *Approximation Algorithms*, vol. 1 (Springer, New York).

Wang Y, Wang X, Deng Y, Cao L, Wang T (2022) Data-driven order fulfillment consolidation for online grocery retailing. Working paper.

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

Xu PJ, Allgor R, Graves SC (2009) Benefits of reevaluating real-time order fulfillment decisions. *Manufacturing Service Oper. Management* 11(2):340–355.

Zhao Y, Wang X, Xin L (2020) Multi-item online order fulfillment: A competitive analysis. *Chicago Booth Research Paper* (20–41).