Service Science

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

Analysis of Real-Time Order Fulfillment Policies: When to Dispatch a Batch?

Natarajan Gautam, Joseph Geunes

To cite this article:
Natarajan Gautam, Joseph Geunes (2024) Analysis of Real-Time Order Fulfillment Policies: When to Dispatch a Batch?. Service Science 16(2):85-106. https://doi.org/10.1287/serv.2022.0042

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

SERVICE SCIENCE
Vol. 16, No. 2, June 2024, pp. 85–106
ISSN 2164-3962 (print), ISSN 2164-3970 (online)

Analysis of Real-Time Order Fulfillment Policies: When to Dispatch a Batch?

Natarajan Gautam,[a,*] **Joseph Geunes**[b]

[a] Department of Electrical Engineering and Computer Science, Syracuse University, Syracuse, New York 13244; [b] Wm Michael Barnes '64 Department of Industrial and Systems Engineering, Texas A&M University, College Station, Texas 77843

*Corresponding author

Contact: ngautam@syr.edu, https://orcid.org/0000-0002-6408-216X (NG); geunes@tamu.edu, https://orcid.org/0000-0001-5613-8146 (JG)

Received: May 20, 2022
Revised: April 3, 2023; September 19, 2023
Accepted: October 10, 2023
Published Online in Articles in Advance: October 30, 2023

https://doi.org/10.1287/serv.2022.0042

Abstract. This paper considers orders that arrive one-by-one over time to a fulfillment center. Each order requests a product with some degree of customization that needs to be delivered expeditiously to a nearby location using a delivery vehicle. However, each vehicle can batch multiple orders together for delivery within a single trip. The benefits of batching include more efficient capacity utilization, lower total vehicle ownership requirements, and reduced environmental impact. The main drawback of batching is the consequent reduced average quality of service due to associated delivery delays when waiting for additional orders to arrive and executing a delivery route. To address this trade-off, we consider a set of threshold policies for batching and dispatching groups of orders, and characterize the associated long-run average cost per unit time for each policy that explicitly accounts for the customer's total order lead time, including the time between order dispatch and delivery to the customer which, in turn, depends on route sequencing policies. For the threshold policies, our state variable may not only include the number of outstanding orders, but may also incorporate information on order arrival times and delivery locations. We model the stochastic dynamics of the system and obtain the long-run average cost per unit time, which we compute using a renewal-reward approach. We also consider different delivery sequencing approaches, including first-come, first-served and shortest traveling salesperson. In addition, we evaluate the effectiveness of accounting for all order information in the decision-making process, as opposed to just the number of outstanding orders or the time in the system for each order. Our analysis shows that a generalized class of cost- and quantity-based threshold policies often outperforms existing policies in the literature with the additional benefits of being robust to overestimates of the optimal cost threshold value and achieving strong delay cost performance.

History: This paper has been accepted for the *Service Science*/*Stochastic Systems* Joint Special Issue.

Supplemental Material: The online appendix is available at https://doi.org/10.1287/serv.2022.0042.

Keywords: renewal-reward process • threshold-policies • partial-differential equations

1. Introduction

A massive shift has occurred in the past few years within the area of time-based logistics, due in no small part to the so-called "Amazon effect" (Lynch 2018) associated with the retail giant's ability to deliver a vast multitude of items to the home within two days at a reasonable price. Amazon Prime Now, for example, delivers items (including select restaurant food and groceries) within two hours in several densely populated cities around the world (and within one hour for an added premium). As a result, a large number of today's consumers have come to expect exceptionally fast order fulfillment for a wide range of consumer goods (Beckwith 2017). Firms in many industries have responded and are scrambling to develop fast-response fulfillment systems in order to gain a speed-based competitive advantage. In the fast-food industry, for example, UberEATS and Jimmy John's (the latter of which uses the trademark Freaky Fast®) have based their reputations and business models largely on fulfillment speed. Firms seeking to use order fulfillment speed as a competitive advantage rely on flexible and responsive delivery systems that leverage relatively recently developed technologies to efficiently match orders to fulfillment capacity. Maximizing the efficiency with which these systems operate (and, therefore, minimizing the system's cost burden) requires understanding and solving a complex problem class that is not currently well defined or understood in the academic literature. Furthermore, in the aftermath of the COVID-19 pandemic, there is a greater need to solve the problem of efficient real-time order fulfillment.

This problem class addresses a fundamental trade-off between speed of service and economies of scale in real-time order fulfillment. In the relevant application settings briefly described above, orders arrive randomly and continuously throughout time, leading to a class of so-called "online" operations problems. Fulfilling an order requires allocating the requested product(s) to an available fulfillment resource and executing delivery. In some contexts, the relevant products are retrieved from inventory at a stocking point, whereas in other settings the products may be assembled-to-order from a stock of components (e.g., in fast-food delivery). Because order fulfillment operations often involve economies of scale, this introduces an important trade-off between fulfillment speed and efficient resource utilization. At one extreme, for example, allocating each order to a unique fulfillment resource may maximize dispatch speed, but may lead to poor resource utilization and cost performance, while increasing the probability of a lack of available fulfillment resources (as a result of all available resources having been allocated to previous orders when a new order arrives). At the other extreme, orders might be dispatched and delivery executed only when a required fulfillment resource's capacity is fully utilized. For example, if the fulfillment resource is a delivery van, such a policy only permits dispatching the van when it is fully loaded with orders. This approach of batching orders reduces the delivery cost incurred per order while increasing the average order fulfillment time. In general, given a random order arrival process, we seek to solve the problem of batching received orders together and determining when to dispatch the resource to execute fulfillment while explicitly considering the delivery process.

This broad discussion immediately raises fundamental questions that must be resolved within this problem class: (1) What approach(es) can best characterize consumer trade-offs between delivery speed and cost? (2) What is the structure of the best policy for grouping orders and dispatching fulfillment resources? (3) How do the properties of the order arrival process and delivery sequence affect the best policy structure? To address these questions, we begin by exploring the relevant literature in Section 2. Then, in Section 3, we present an illustrative deterministic model that lays the foundation to study such policies and points to some insights about the inherent trade-offs. Subsequently, in Section 4, we consider a stochastic model, discuss static policies, and introduce two delivery mechanisms. Then, in Section 5, we describe dynamic policies that we evaluate by characterizing the system using the state evolution dynamics. In particular, we propose a class of cost- and quantity-based threshold trigger policies that use information on accumulated orders and cost to determine when to dispatch. To test the effectiveness of the proposed policies, we present the results of several numerical experiments in Section 6. We start with experiments that use real location data from three cities in the United States, followed by simulated experiments on a variety of problem instances. As the results show, the proposed cost- and quantity-threshold policies turn out to be not only effective in minimizing long-run average costs, but insensitive to overestimates of the optimal cost threshold value, while also providing relatively low delay costs, a measure of service-level performance. Section 7 presents concluding remarks and ideas for future work.

2. Relevant Literature

Although the problem we consider is related to several well-studied problems in operations research, supply chain logistics, and transportation, to the best of our knowledge, it has not been explicitly considered in the open literature. Recent informal discussions with researchers at Amazon also reveal that these problems are not thoroughly understood in practice, and there is a need for new methodologies to address them effectively. Nonetheless, some relevant literature does exist. The related literature generally falls within the areas of online routing decisions, last-mile delivery, delivery dispatching service–cost trade-offs, and same-day delivery (SDD) problems.

Whereas we primarily emphasize the time between order arrivals and when these orders are delivered to customers, most of the existing and vast literature focuses on the delivery process after a dispatch occurs. A voluminous body of work exists dealing with vehicle routing problems (VRPs) with or without uncertainty. A segment of this literature considers the probabilistic traveling salesperson problem (TSP). Daganzo (1984, 2005) provides excellent resources that discuss a wide range of routing and VRP issues. In particular, Daganzo (2005) shows that for spatial Poisson locations, the total TSP distance for n stops is approximately $a + \sqrt{n}b$, where a and b are positive constants, which we have verified computationally for random sites selected in various cities, as we later note. Indeed, a large body of work exists on refining and generalizing TSP and vehicle routing tour length approximations. A comprehensive survey of the tour length approximation literature can be found in Choi and Schonfeld (2022), whereas Ansari et al. (2018) provide an in-depth look at the use of tour length approximations in location and transportation and logistics planning problems. Recent developments in this vein include data-based approaches for last-mile delivery route duration (Liu et al. 2021), distribution-free regression-based methods (Cavdar and Sokol 2015), and estimates based on the mean and standard deviation of randomly selected tours (see Kou et al. 2022, 2023).

Numerous articles have appeared dealing with probabilistic versions of the TSP since Jaillet's (1985) thesis. Henchiri et al. (2014) present an excellent survey of those articles. In addition, real-time solutions for VRPs, wherein routes may be modified during execution based on new order arrivals, have been addressed by Gendreau et al. (1999) and Ghiani et al. (2003). Ichoua et al. (2000) incorporate diversion-based real-time dispatching. However, real-time dispatching has also been considered by Krumke et al. (2002) using soft windows, whereas Du et al. (2007) focused on milk runs. Several authors have applied customer-based analytics to characterize how various factors influence last-mile delivery costs. For example, Boyer et al. (2009) consider how customer density and delivery window patterns affect efficiency, whereas Chen and Pan (2016) demonstrate how crowdsourcing can be used for last-mile delivery.

Cattaruzza et al. (2017) discuss real-time vehicle routing in cities and the challenges presented in urban settings, whereas real-time dispatching for truckload motor carrier fleets is considered by Powell (2007). The notion of electronic freight matching using auctions and other mechanisms has been well studied (viz., Duin and Kneyber 2004, Nandiraju and Regan 2008, De Maio et al. 2018). Furthermore, the dissertation by Minkoff (1985) and subsequent article (Minkoff 1993) fall within the category of dynamic inventory replenishment dispatching, and an excellent survey of more recent work on this topic is provided by Coelho et al. (2013). Van Heeswijk et al. (2019) generalized Minkoff's dynamic dispatching model to account for delivery time windows, using a Markov decision model and Daganzo's (1984) tour length approximation approach for estimating vehicle route duration.

Whereas all of the above articles are focused on the delivery side, there are very few articles that consider combined batching and dispatching policies and their impacts on customer waiting costs. The seminal article by Çetinkaya and Bookbinder (2003) discusses stochastic models for consolidating shipments and then dispatching deliveries, focusing on dispatching trucks to a fixed set of retailers. Our research generalizes this work along several contemporary and practically relevant dimensions; for example, the number and locations of delivery customers (i.e., retailers) may be random; a dispatched vehicle must deliver to multiple different destinations; and real-time system state information is available that can be used for making efficient batching decisions. Interestingly, Mutlu et al. (2010) consider dispatch policy structures similar to those applied in the model we later present. In particular, these policies trigger a delivery dispatch following a fixed time interval T, a fixed dispatch quantity q, or some threshold that depends on a combination of T and q. The approaches taken by both Çetinkaya and Bookbinder (2003) and Mutlu et al. (2010) account for holding, or waiting, costs that accumulate between customer order arrival time and the time at which the subsequent batch is dispatched, and do not account for the additional time between dispatch and customer delivery. A distinguishing feature of the modeling approach we propose lies in its ability to account for the total time between order arrival and delivery, that is, the entire customer order lead time. This additional time is affected by delivery sequencing policies; that is, as we later discuss, average customer order lead time depends on whether deliveries are made in a first-come, first-served (FCFS) order, which customers may perceive as fair, versus a shortest-delivery-route order. Our preliminary results indicate trends similar to theirs, while accounting for delivery routing economies of scale. However, as we later discuss, we find that supplementing the state space with an accumulated cost dimension facilitates additional policies that can outperform fixed time interval or dispatch quantity based threshold policies, while providing additional service performance benefits and being robust to overestimation of the optimal cost threshold parameter value.

A number of works have appeared that address generalizations of the shipment consolidation problem proposed by Çetinkaya and Bookbinder (2003), as well as more general so-called stochastic clearing systems. Bookbinder et al. (2011) consider a shipment consolidation problem in which order arrivals follow a Markovian process in discrete time and order weights follow a discrete distribution. Cai et al. (2014) generalized this work further by accounting for individual order-based waiting costs, as well as waiting costs that increase at a nondecreasing rate with time. As we later discuss, the modeling approach we use permits accounting for waiting costs that are nondecreasing in the batch delivery size, and therefore in the average customer waiting time. Satir et al. (2018) generalize the shipment consolidation problem to permit two demand classes, regular and expedited, which requires rationing delivery capacity between the two classes. A key contribution of this work lies in demonstrating the ability to address the infinite-horizon version of the problem and to characterize the optimal policy structure for this case. Wei et al. (2022) consider the more general class of stochastic clearing systems and propose measuring system performance based on average order delay, providing an analytical characterization of this measure for clearing systems with Poisson arrival processes and under a variety of clearing policies.

Works dealing with batching, consolidating, and bundling are abundant in the operations literature. The notion of batching and serving in bulk is well studied in manufacturing (e.g., baking of multiple products). A wide variety of such models is presented in books such as those by Curry and Feldman (2010), Hopp and Spearman (2011), and Buzacott and Shanthikumar (1993). Furthermore, dispatching problems are also common in transportation systems, such as those using shuttle buses (see Lee and Srinivasan 1990, Ceder and Yim 2003, Chen and Wan 2003).

Online versions of production batching decisions arise in make-to-order systems when customer orders arrive according to a random process. Zhang et al. (2001) consider a single machine problem in which n jobs will arrive with certainty, although their processing and arrival times are not known in advance. The time required to produce a batch equals the maximum processing time among all jobs in the batch, and the objective minimizes the maximum completion time for all jobs. They propose online batching algorithms with worst-case performance bounds. Tian et al. (2007) extended this problem to account for delivery times after production completion and also provide online algorithms with worst-case performance guarantees. The problem difficulty increases considerably when multiple products facing stochastic demand require production using a single machine that requires changeover times between product batches. This class of *stochastic economic lot scheduling problems* requires dynamically determining which product to produce during each time instant, and when to switch production between products, based on the stock and demand state of each product in the system. Researchers have approached this problem using a number of strategies that consider repeatable cyclic production rotation cycles (see, e.g., Markowitz et al. 2000, Markowitz and Wein 2001) as well as standard inventory policy structures such as (s, Q) or (s, S) policies (see, e.g., Anupindi and Tayur 1998, Wagner and Smits 2004, Eisenstein 2005). An in-depth discussion of this literature stream can be found in Winands et al. (2011). The most closely related work in this stream to our work considers problems where items are made to order (see, e.g., Dellaert 2012), although we permit batching any set of items together and consider the cost and service implications in the downstream delivery process.

A sizeable related literature exists dealing with batching decisions for warehouse picking with online order arrivals, a recent comprehensive survey of which can be found in Pardo et al. (2023). Chew and Tang (1999) provided one of the earliest contributions in this stream, considering a rectangular warehouse layout, and characterizing an estimate of the average throughput time of an order using queueing analysis. Le-Duc and De Koster (2007) later extended this analysis to a more general class of warehouse layouts, and provided a more accurate characterization of average throughput time. Çeven and Gue (2017) consider warehouse operations in which orders are collected and released to pickers in successive waves, where each wave has a corresponding dispatch deadline. Their goal is to maximize the fraction of orders that are processed by a given deadline.

Recent literature considers the closely related SDD problem of dynamically assigning orders to delivery vehicles that execute so-called last-mile delivery routes, while accounting for the possibility of future uncertain orders. Azi et al. (2012) use large-scale neighborhood search to solve an SDD problem with online order arrivals, where orders may be rejected. Ulmer et al. (2019) solve a single-vehicle SDD problem using approximate dynamic programming and demonstrate the value of preemptive returns to the depot. Klapp et al. (2018b) consider a single-vehicle version of the problem on a line, where decisions occur in waves, at particular time points. Their model minimizes expected delivery plus lost demand costs by solving a stochastic dynamic program over a finite horizon. Klapp et al. (2018a) later generalized this to permit general network structures and delayed order acceptance/rejection decisions. Voccia et al. (2019) provide what appears to be the most comprehensive operational model and solution approach for the SDD problem to date, accounting for multiple vehicles and delivery time windows, and modeling the problem as a Markov decision process. Stroh et al. (2022) consider tactical dispatch timing decisions for SDD problems with deterministic demand rates in which all deliveries must be completed by the end of each day, with a goal of minimizing the total time required for deliveries. They use continuous time dispatch duration approximations to develop heuristic policies with worst-case performance guarantees for practical problem settings. In contrast to this stream of literature, the work we propose emphasizes the structure and value of simple batching and dispatch policies, as well as the interdependence of speed of service and demand rates, and the impacts these factors have on profitability and operations efficiency.

3. Illustrative Model: Deterministic, Speed-Dependent Order Rate

This section describes a set of stylized models that frame the class of time-based fulfillment problems we study. Consider a setting in which orders arrive, one at a time, to a fulfillment center over time. Fulfilling an order requires dispatching some type of fulfillment resource. Let us assume that the fulfillment center has an unlimited number of such resources. Thus, it is possible to fulfill each order as it arrives by allocating an available fulfillment resource. Economies of scale exist, however, in fulfillment operations; that is, the cost (or time) required to use a single resource to fulfill n orders simultaneously is less than the sum of the costs (or time) required when a single resource fulfills each of the n orders individually (or in multiple subsets of orders). This provides an incentive for batching orders and executing their fulfillment using a common resource. Doing so requires waiting for multiple orders to accumulate before batching them together and dispatching a resource for their fulfillment.

At the same time, customers are becoming increasingly time sensitive, such that the rate of order arrivals, as well as the total revenue or cost associated with each order, may to some extent depend on a measure of order fulfillment speed. As a result, although exploiting economies of scale in fulfillment operations may reduce average fulfillment cost, this may, in turn, reduce the order rate and/or revenue per order. This creates a tension between the desire to exploit economies of scale in fulfillment operations and the goal of maximizing total revenue. We illustrate this class of problems beginning with a stylized model with deterministic and stationary order interarrival times that depend on service speed. We will later illustrate a modeling basis for the study of online decision problems with uncertain arrivals.

Consider an order fulfillment service in which the service provider accepts time-sensitive orders for a product that must be delivered to the customer, where customers are distributed uniformly over a service region of area A. Assume that orders arrive at a deterministic rate of λ per unit time over some horizon of length T_H, and that this order arrival rate is decreasing in average service time, that is, shorter fulfillment times increase the demand rate. As noted earlier, we assume the fulfillment service provider has an adequate number of delivery resources (e.g., drivers and vehicles, or perhaps drones) available such that delivery routes are not delayed by the absence of a driver or vehicle (we can easily specify conditions under which this holds). Suppose that a delivery resource is dispatched after every q orders arrive, where q is a decision variable; that is, each delivery resource serves a batch of q customers whose calls arrived consecutively, and deliveries within a batch may be sequenced in order to minimize route-sequencing-based costs. These assumptions are initially somewhat restrictive, and we will later consider the implications of relaxing some of them. Note that delivery in batches of size q corresponds to a policy choice, and it is not immediately clear under what conditions (if any) such a policy provides an optimal policy structure.

We compute approximate distance-related delivery cost/time using routing distance approximation approaches from Burns et al. (1985). Letting θ denote the average customer distance from the depot, they approximate total routing distance when serving q customers using the equation

$$D(q) \equiv 2\theta + \sqrt{\kappa q A}, \tag{1}$$

where κ is a scaling constant. The first term in (1) corresponds to an estimate of the average *line-haul distance*, that is, the out-and-back distance between the depot (dispatch) location and the customer region, whereas the second term corresponds to the *local delivery distance*, which accounts for local routing within the region. If we serve N_H customers over the time horizon of length T_H, then the total number of delivery routes will equal $\lceil \frac{N_H}{q} \rceil$, for an approximate total travel distance during the time horizon of

$$D_T(q) \approx \left\lceil \frac{N_H}{q} \right\rceil \left(2\theta + \sqrt{\kappa q A}\right). \tag{2}$$

The total number of customers requiring delivery, N_H, depends on the order arrival rate λ and the planning horizon length, denoted by T_H, that is, $N_H = \lambda T_H$. We assume that faster deliveries result in an increased demand rate, where average delivery speed depends on the number of customers served per route, q. Clearly, an increasing value of q increases the average delivery time, which reduces demand; thus, N_H is decreasing in q, and we henceforth explicitly account for the dependence of N_H on q using the notation $N_H(q)$. (We assume a fixed area A; thus, as $N_H(q)$ increases, the customer density, that is, $N_H(q)/A$, increases as well.)

Let us assume that the average net revenue per customer before accounting for delivery routing costs equals π, which implies a net revenue over the time horizon of $\pi N_H(q)$. Letting ℓ denote a cost per unit distance (which may account for both time and distance related costs), the approximate total profit during the time horizon, $\Pi(q)$, can be written as

$$\Pi(q) = \pi N_H(q) - \ell \left\lceil \frac{N_H(q)}{q} \right\rceil \left(2\theta + \sqrt{\kappa q A}\right). \tag{3}$$

(Note that any fixed route dispatch cost can be incorporated within the $2\ell\theta$ line-haul cost term, and we therefore do not explicitly include such a term in the above profit equation.) Because we expect this model will apply in contexts with a large number of customers relative to the number of customers per delivery route, and because the relative difference between $N_H(q)/q$ and $\lceil N_H(q)/q \rceil$ approaches zero as the number of routes increases, we approximate the profit equation as

$$\Pi(q) = N_H(q)\left(\pi - \ell\left(\frac{2\theta}{q} + \sqrt{\frac{\kappa A}{q}}\right)\right). \tag{4}$$

Observe that this profit equation may potentially be quite complex in general, depending on the form of $N_H(q)$. For illustrative purposes, suppose we assume that demand is an isoelastic function of q, that is, the demand rate takes the form $N_H(q) = aq^{-b}$, with $a, b > 0$;

in this case, we can show that the only positive real stationary point solution exists at

$$q^* = \left(\frac{\ell\sqrt{\kappa A}(2b+1) + \sqrt{\ell^2\kappa A(2b+1)^2 + 16b\pi\ell\theta(b+1)}}{4b\pi}\right)^2. \quad (5)$$

We can show that $\Pi''(0) < 0$, and that $\Pi''(q)$ is strictly increasing in q, and thus equals zero at a single value of $q > 0$. This implies that a value of $q = \hat{q} > 0$ exists such that $\Pi(q)$ is concave for $0 \le q \le \hat{q}$ and is convex for $q \ge \hat{q}$. Moreover, the stationary point (5) occurs in the concave portion of $\Pi(q)$, implying that q^* in Equation (5) is a global maximum among all $q \ge 0$.

If the radial distance θ is negligible (e.g., the depot lies within the region of area A), then Equation (5) becomes

$$q^* = \frac{\ell^2\kappa A(2b+1)^2}{4(b\pi)^2}. \quad (6)$$

The optimal batch size is directly proportional to the area of the region, A, and the square of the cost per unit distance ℓ, whereas it is inversely proportional to the square of the unit net revenue π and the demand elasticity parameter b, indicating that higher-priced items justify higher costs and faster service, as we might expect, although this effect is diminished over larger areas or under higher distribution costs per unit distance.

At the other extreme, consider the case in which routing distance is negligible relative to straight-line distance, as would be the case if the dispatch location is relatively far from customers, who are densely populated in a local area (e.g., pizza or sub deliveries to a college campus, when the store is off campus). In this case, Equation (5) becomes

$$q^* = \frac{\ell\theta(b+1)}{\pi b}, \quad (7)$$

and the optimal batch quantity is proportional to the straight-line distance θ and the cost per unit distance ℓ, while again decreasing in the net revenue term π and the elasticity parameter b.

The deterministic demand assumption is quite restrictive in practice, although it permits obtaining closed-form expressions that encode the key trade-offs in time-sensitive batch deliveries. Accounting for uncertainty in demand arrivals across a range of potential dispatch policy types requires more advanced analytical techniques in order to capture expected delivery costs, while permitting online decision making in real time. Addressing demand uncertainty requires confronting some modeling choices in order to arrive at a realistic model that is also tractable for each policy type considered. In particular, we found that the inclusion of a batch-size-dependent demand rate leads to an overly complex mathematical model that does not lead to tractable analysis for some of the policies we wish to consider. For this reason, instead of directly using a batch-size-dependent demand rate, we assume that a cost penalty is applied per unit of delay between order placement and delivery. We later discuss deterministic and static versions of these models and their relation to those discussed in this section.

4. Stochastic Order Arrivals and Static Policies

Next, we outline a class of models that addresses situations in which order arrival times are uncertain. Our policy approach prescribed in the previous section under deterministic order interarrival times ensures that each dispatch occurs immediately following some order arrival. In the deterministic case, it is easy to show that any policy in which a dispatch occurs strictly between arrivals cannot be optimal. In the stochastic order arrival case, however, it is not immediately clear that this property holds, even if the arrival rate and all costs are stationary. Because of this we will initially consider two fundamental policy structures. The first structure involves a quantity-based dispatch trigger, and is the same as in the deterministic case, that is, a dispatch occurs immediately after q orders have accumulated, for some value of q. The second structure uses a time-based trigger in which an order is dispatched T time units after the first order in the batch has arrived. Later, in Section 5, we will introduce a "cost-based" trigger, the analysis of which will require a slightly more sophisticated set of mathematical tools, but will lead to policies that, on average, outperform both the quantity- and time-based policies.

For illustrative purposes, the models we describe here assume a stationary order arrival rate and an infinite horizon (i.e., $T_H = \infty$), and do not permit intentionally delaying an order for processing with some future batch. For example, if among the previous q orders, $q-1$ of these orders must be delivered within a small radius of one another, and one of these orders is far away from the others, then it may make sense to delay this geographic "outlier" until additional orders arrive with delivery locations close to that of the delayed order. As we later discuss, properly handling and identifying useful batching policies serves as an important goal of this work. We first describe results that apply to general dispatching and sequencing decisions (assuming no order delays for future batches), then discuss the implications of different delivery sequencing decisions within each batch, and then analyze particular static time-based and quantity-based batch and dispatch rules under a stationary arrival rate.

4.1. Analysis of an Arbitrary Dispatch Trigger via Renewal Model

Consider a fulfillment center responsible for delivering customer orders that arrive in real-time at an average rate of λ per unit time according to a Poisson process. We continue to assume that sufficient delivery resources are available such that a policy that dispatches a delivery resource whenever a policy trigger condition is met is always feasible. We assume that each delivery resource (i.e., vehicle) waits for the first order to arrive and then is dispatched after a random time T, and during this random time, a random number of orders $(N-1)$ arrive. (For the quantity-based trigger with policy parameter q, this condition is met upon accumulating $N-1=q-1$ order arrivals after the first order; for the time-based trigger with constant time T, the condition is met when a constant T_d time units have elapsed since the first order arrival in the batch.) Figure 1 illustrates a timeline of vehicle arrivals, one at a time, followed by vehicle dispatches. (In the figure, n_1 orders are dispatched in the first batch, n_2 in the second batch, and so on. Note that a vehicle "arrival" may correspond to a return from a prior trip, or simply to the vehicle's availability as next in line for dispatching; in either case, we assume a vehicle is available for dispatching whenever needed.)

We assume that order processing time is negligible (e.g., the time required to pick the item, customize it, and allocate it to the delivery resource). In contrast to the preliminary deterministic model discussed earlier, we will assume for this analysis that the order arrival rate is independent of delivery speed; instead, the models in this subsection permit decreasing net order revenue per order as a function of the customer's delivery lead time via an order delay cost. Thus, the models in this subsection use a cost minimization approach instead of profit maximization, and we define h_n as a per unit delay cost rate assessed between order arrival and delivery when there are n orders waiting for delivery. That is, instead of directly considering a demand rate function that depends on the batch delivery size (and, therefore, the average waiting time until delivery), we will instead incur a service-related delay cost that effectively leads to reduced net revenue as batch quantities and corresponding delivery times increase. As noted earlier, this approach is necessary for permitting performance comparison with the dynamic policies we later introduce in Section 5 (we will later discuss the relationship between the static models that employ a delay cost and the deterministic models in the previous section). We assume that h_n is a nondecreasing function of n, and at any instant of time when there are n orders (either awaiting dispatch or en route for deliveries), the delay cost per unit time incurred is nh_n. Because the time required to accumulate and deliver a batch is nondecreasing in n, this provides a general framework for defining a nonlinear and nondecreasing penalty cost for waiting. We will also consider the special cases of $h_n=h$ and $h_n=(n-1)h$ for all n when presenting the results. The former case corresponds to a time invariant delay cost rate, and the latter permits accounting for a delay cost rate that increases with time, while also permitting tractable analysis.

We define a fixed dispatch cost of ϕ for each delivery route and incorporate our previously defined delivery cost of ℓ per unit time spent in delivery. Another difference between the deterministic model and the stochastic model is that the system performance is analyzed using *distance* in the former case, and *time* in the latter. In this vein, we define some time-related parameters. We defined the order arrival process as a Poisson process with rate λ orders per unit time. Hence, the interarrival time of orders follows an $\exp(\lambda)$ distribution. Recall that the time from when the first order is placed to when a set of orders is dispatched is a random variable T (for the moment we do not elaborate on the distribution of T). Let Y_i denote a random variable for the time required to travel from the ith to $(i+1)$st delivery location. Suppose $Y_1, Y_2, \ldots$ are independent, identically distributed (IID) random variables. We let X_1 and X_0 be the time to travel from the dispatching station to the first customer location and from the last customer location back to the dispatching station, respectively. The timeline for a single delivery resource is depicted in Figure 2.

Let us take the perspective of one route taken by a delivery resource. We reset time to zero when the first order arrives for this resource. Suppose $N-1$ additional orders arrive between zero and T, the time at which orders are batched for the delivery resource (the

Figure 1. Timeline Representation of Order Arrivals, Vehicle Arrivals, and Vehicle Dispatch

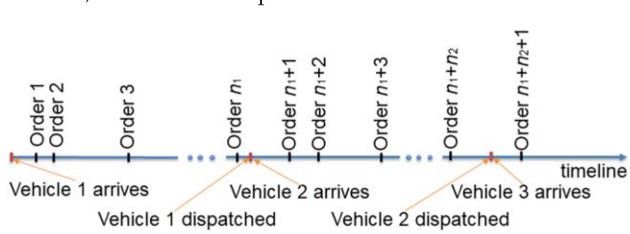


Figure 2. Timeline of Activities for a Single Dispatch (Resetting Order 1 Arrival Time as 0)

arrival times are $S_1 = 0, S_2, \ldots, S_N$, as shown in Figure 2). So, from time 0 to S_2 there is one order, from S_2 to S_3 there are two orders, and so on, and from S_n to T we have N orders. At time T, the delivery resource is dispatched with the N orders. The times spent by the N orders until the vehicle is dispatched are $T, T - S_2, \ldots, T - S_N$. The total waiting time starting from dispatch (aggregated over all N orders) is $NX_1 + (N-1)Y_1 + (N-2)Y_2 + \cdots + 1Y_{N-1}$, as N orders spend X_1 time in delivery, then the first delivery occurs, $N-1$ orders spend an additional Y_1 time units until the second delivery occurs, and so on. The delay cost incurred for the dispatched vehicle is then

$$Z_h(N,T) = h_1 S_2 + 2h_2(S_3 - S_2) + \cdots + Nh_N(T - S_N) + (Nh_N X_1 + (N-1)h_{N-1}Y_1 + (N-2)h_{N-2}Y_2 + \cdots + 1h_1 Y_{N-1}),$$

which is a random variable. The total travel time is $X_1 + Y_1 + Y_2 + \cdots + Y_{N-1} + X_0$; hence, the delivery cost incurred for the vehicle is

$$Z_\ell(N,T) = \ell(X_1 + Y_1 + Y_2 + \cdots + Y_{N-1} + X_0),$$

which is also a random variable. Notice that the X_0 term is not included in the delay cost, as during this time, the delivery resource returns empty.

Theorem 1. *The long-run average cost per unit time $E[\mathcal{C}(N,T)]$ incurred by the system is*

$$E[\mathcal{C}(N,T)] = \frac{\phi + E[Z_h(N,T)] + E[Z_\ell(N,T)]}{E[T] + \frac{1}{\lambda}}. \tag{8}$$

Proof. Please see Online Appendix A.

Our eventual goal is to determine policy parameters that minimize $E[\mathcal{C}(N,T)]$. However, before discussing cost minimization, we next explore the delivery sequence, as this impacts total cost.

4.2. Routing Time and Delivery Sequence—FCFS vs. TSP

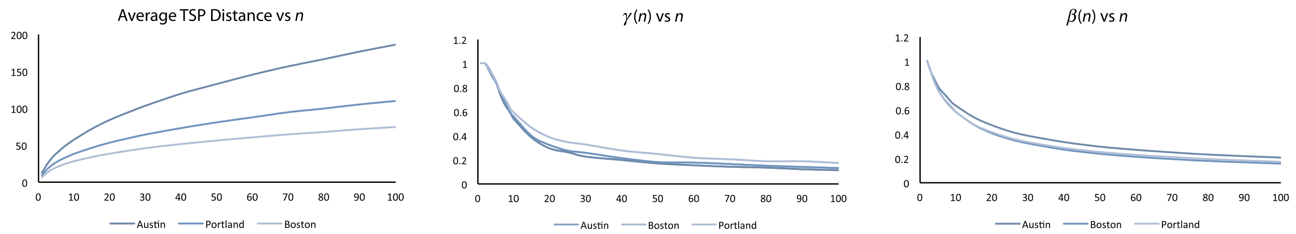
Time-based economies of scale exist in delivery operations, and we initially assume that orders are delivered in an FCFS order of arrival (we later consider the implications of route sequencing to minimize batch routing costs). Situations in which FCFS might be appropriate arise when delivery operations are transparent to customers. Customers may consider it fair that orders are delivered in the sequence in which they were received, hence, FCFS. Let θ denote the expected value of the time to travel from the distribution center to an arbitrary customer (which is analogous to the line-haul distance we defined earlier). Likewise, let τ be the average time to travel between any two arbitrarily selected customers. Because customer orders are delivered according to FCFS, $E[X_1] = E[X_0] = \theta$. Also, $E[Y_1] = E[Y_2] = \cdots = \tau$. It is worthwhile reiterating that this holds only for FCFS, and other delivery sequences can produce different results. Although we will describe the costs explicitly in the next subsection, note that under this FCFS approach, the expected time to deliver n items and return to the fulfillment center equals $2\theta + (n-1)\tau$.

Under the FCFS fulfillment sequence, dispatching decisions do not consider the cost implications associated with route sequencing. As a first step toward capturing these implications, we introduce a more general route approximation approach that uses scaling functions $\gamma(n)$ and $\beta(n)$, with the expected travel distance to the first customer on the route $E[X_1] = \gamma(n)\theta = E[X_0]$, and the expected distance between successive customers on the route $E[Y_1] = E[Y_2] = \cdots = E[Y_{n-1}] = \beta(n)\tau$. Suppose the route sequence is determined according to a TSP solution. Using this approach, the expected time to deliver n items and return to the fulfillment center equals $2\gamma(n)\theta + \beta(n)(n-1)\tau$ (with an additional $\gamma(n)\theta$ on average to return) for some scaling functions $\gamma(n)$ and $\beta(n)$.

Now, contrast the results for $E[X_1]$, $E[X_0]$, $E[Y_i]$ for $i = 1, \ldots, n-1$, between FCFS and TSP. We can think of FCFS as a special case where $\gamma(n) = \beta(n) = 1$ for all n. This raises the question: does this conform to real data, and if so, what is a good approach to estimating the scaling functions? In previous work (Bassett and Gautam 2019), one of the coauthors considered Airbnb data with given latitude and longitude values of rental properties in several cities. Using these data, they solved TSP instances for various values of n by randomly sampling from rental locations in each city. They used Euclidean distances between points, as actual addresses were unavailable, and then averaged over multiple samples to obtain the distance traveled as a function of n, the number of stops. The left panel in Figure 3 shows the average distance of the TSP route to travel to n delivery points and return to the fulfillment center (which was chosen randomly using the latitude and longitude values), and resembles the form $a + \sqrt{n}b$ stated in Daganzo (2005) for spatial Poisson locations.

Notice from the left panel in Figure 3 that the three cities chosen have relatively different distances covered, due to the topology, spread of homes, and the location of the fulfillment center. However, when the total distances are decomposed into the distance to the first customer (or from the last) and the distances between delivery stops, we can estimate the scaling functions $\gamma(n)$ and $\beta(n)$. This can be done by first estimating θ and τ assuming a constant delivery speed. We can see that both $\gamma(n)$ and $\beta(n)$ are nonincreasing functions of n for various cities, and are less than or equal to one (see the middle and right panels in Figure 3). Now we are in a

Figure 3. Mean TSP Distance $2\gamma(n)\theta + \beta(n)(n-1)\tau$, $\gamma(n)$ and $\beta(n)$ for Three Cities Using Airbnb Data



position to discuss dispatching policies (for which we will use both FCFS and TSP mechanisms).

4.3. Time- and Quantity-Based Static Dispatch Policies

Here we consider two static dispatch policies: quantity- and time-based triggers. We present results assuming TSP-based delivery, with the understanding that by simply letting $\gamma(n) = \beta(n) = 1$, we obtain the FCFS results. We first illustrate the analysis using a quantity-based dispatch trigger policy with batch size q (and later consider the time-based approach). Thus, after q orders arrive, the vehicle is dispatched immediately. Hence $n = q$, a deterministic value, and $E[T] = (q-1)/\lambda$, as this is the expected time for $q-1$ additional orders to arrive after the first. Under this quantity-based trigger approach, the associated renewal-reward process has an expected cost per unit time (from Equation (8)) of

$$E[\mathcal{C}(q,T)] = \frac{\left\{\phi + \frac{1}{\lambda}\left[\sum_{n=1}^{q-1} nh_n\right] + \left[qh_q\gamma(q)\theta + \beta(q)\tau\sum_{n=1}^{q-1} nh_n\right] + \ell(2\gamma(q)\theta + \beta(q)(q-1)\tau)\right\}}{q/\lambda}. \quad (9)$$

Although the above equation can directly be used for the TSP delivery sequence, we can easily obtain the corresponding expression for the FCFS case by letting $\gamma(n) = \beta(n) = 1$. Closed-form expressions can be obtained for two special cases. When $h_n = h$ for all n, we have

$$E[\mathcal{C}(q,T)] = \frac{\phi + \frac{h}{\lambda}\left[\frac{q(q-1)}{2}\right] + h\left[q\gamma(q)\theta + \frac{\beta(q)\tau q(q-1)}{2}\right] + \ell(2\gamma(q)\theta + \beta(q)(q-1)\tau)}{q/\lambda}. \quad (10)$$

Furthermore, when $h_n = (n-1)h$ for all n, we have

$$E[\mathcal{C}(q,T)] = \frac{\left\{\phi + \frac{h}{\lambda}\left[\frac{q(q-1)(q-2)}{3}\right] + h\left[q(q-1)\gamma(q)\theta + \frac{\beta(q)\tau q(q-1)(q-2)}{3}\right] + \ell(2\gamma(q)\theta + \beta(q)(q-1)\tau)\right\}}{q/\lambda}. \quad (11)$$

The above results can easily be derived using Equation (9).

It is worthwhile noting that for the TSP case, based on the Airbnb data (as shown in Figure 3), a reasonable approximation is $\gamma(n) = b_\gamma/\sqrt{n}$ and $\beta(n) = b_\beta/\sqrt{n}$ for some constants b_γ and b_β.[1] Using these expressions, we can minimize $E[\mathcal{C}(q,T)]$ by taking the derivative with respect to q to obtain the optimal q^*. However, this does not provide a closed-form expression even for the two special cases in Equations (10) and (11), as the equation results in a high order polynomial in terms of q. For the FCFS case, there is a corresponding closed-form expression of the form

$$q^*_{FCFS} = \sqrt{\frac{2\lambda(\phi + \ell(2\theta - \tau))}{h(1+\lambda\tau)}}, \quad (12)$$

for the special case in Equation (10), with the requirement that $2\theta > \tau$, which will be satisfied due to the triangle inequality (note that $X_0 + X_1 > Y_1$ and taking expectations yields the result).

Furthermore, suppose we let $\lambda = a - bq$ or $\lambda = aq^{-b}$, where the order arrival rate is a function of the number of orders in a batch, consistent with the idea of demand elasticity based on service quality. Then, for the TSP and the FCFS cases, we can obtain the optimal batch size q^* by maximizing $\pi\lambda - E[\mathcal{C}(q,T)]$ (for which in the TSP case we assume $\gamma(n) = b_\gamma/\sqrt{n}$ and $\beta(n) = b_\beta/\sqrt{n}$ for some constants b_γ and b_β). Again, the optimal value is not expressible in closed form. However, numerical values can be obtained (when $\gamma(n) = 1$ and $\beta(n) = b_\beta/\sqrt{n}$, as discussed in the endnote, the expected cost of the resulting model under isoelastic demand, if we ignore the delay cost terms, is consistent with our prior deterministic model, and we can use Equation (5)). Thus, under a static, quantity-based policy, we can handle the batch-size-based demand rate assumption used in the deterministic model analysis in the previous section.

Next we describe a time-based dispatch trigger (where T_d is the fixed time we wait for any additional deliveries). Note that for a given constant T_d, the number of arrivals in time T_d follows a Poisson distribution with mean λT_d. Thus, the probability that there are n items in a batch (i.e., there are $n-1$ arrivals in time T_d) is $e^{-\lambda T_d}(\lambda T_d)^{n-1}/(n-1)!$. By conditioning on the number of arrivals in time T_d, we can obtain the expected cost per unit time as

$$E[\mathcal{C}(N,T_d)] = \frac{\left\{\phi + \sum_{n=1}^{\infty}\left[\frac{T_d}{n-1}\sum_{i=1}^{n-1}(h_i i) + (h_n n + 2\ell)\gamma(n)\theta + \tau\beta(n)\left(\sum_{i=1}^{n-1}(h_i i) + \ell(n-1)\right)\right]e^{-\lambda T_d}\frac{(\lambda T_d)^{n-1}}{(n-1)!}\right\}}{T_d + 1/\lambda}, \quad (13)$$

where T_d is a constant. As before, the above equation can be used not only for the TSP delivery sequence but also under the FCFS case by letting $\gamma(n) = \beta(n) = 1$. For two special cases discussed earlier, we can derive expressions. When $h_n = h$ for all n, we have

$$E[\mathcal{C}(N,T_d)] = \frac{\left\{\phi + h\left[T_d + \frac{\lambda T_d^2}{2}\right] + \sum_{n=1}^{\infty}\left[(hn + 2\ell)\gamma(n)\theta + \tau\beta(n)\frac{hn(n-1)+2\ell(n-1)}{2}\right]e^{-\lambda T_d}\frac{(\lambda T_d)^{n-1}}{(n-1)!}\right\}}{T_d + 1/\lambda}. \quad (14)$$

Furthermore, when $h_n = (n-1)h$ for all n, we have

$$E[\mathcal{C}(N,T_d)](T_d + 1/\lambda) = \phi + h\lambda T_d^2[3 + \lambda T_d]/3 + \sum_{n=1}^{\infty}\left[(hn(n-1) + 2\ell)\gamma(n)\theta + \tau\beta(n)\frac{hn(n-1)(n-2) + 3\ell(n-1)}{3}\right]e^{-\lambda T_d}\frac{(\lambda T_d)^{n-1}}{(n-1)!}. \quad (15)$$

The above results can be derived by taking the summations in (13) and computing expectations.

We can minimize $E[\mathcal{C}(N,T_d)]$ in T_d by taking the derivative with respect to T_d. However, it is intractable to obtain a closed-form expression even for the approximations $\gamma(n) = b_\gamma/\sqrt{n}$ and $\beta(n) = b_\beta/\sqrt{n}$. For the FCFS case, the optimal time to dispatch is

$$T^*_{FCFS} = \frac{1}{\lambda}\left\{\sqrt{\frac{2\lambda\left(\phi + \ell(2\theta - \tau)\right)}{h(1+\lambda\tau)} - 1} - 1\right\}, \quad (16)$$

for the special case of $h_n = h$ using Equation (14), where the requirement is $2\theta > \tau$ as in Equation (12). Also, because $q^*_{FCFS} \geq 1$, the term under the square root in Equation (16) is nonnegative. Because the expected number of orders in time T^*_{FCFS} is λT^*_{FCFS}, which is the time until dispatch after the first order is received, the value is extremely close to $q^*_{FCFS} - 1$, which leads us to wonder whether the TSP cases might also result in the costs being similar for the special case of $h_n = h$ for all n.

If we again allow the demand rate to depend on the batch delivery quantity, for example, $\lambda = a - bq$ or $\lambda = aq^{-b}$, then for the TSP and the FCFS cases, we can obtain an optimal dispatch time T^* by maximizing $\pi\lambda - E[\mathcal{C}(N,T_d)]$ (for which, in the TSP case, we can either use data to estimate $\gamma(n)$ and $\beta(n)$, or assume $\gamma(n) = b_\gamma/\sqrt{n}$ and $\beta(n) = b_\beta/\sqrt{n}$ for some constants b_γ and b_β). As before, the structure of this optimal value is not expressible in closed form. However, numerical values can be obtained. Thus, for this class of time-based static policies, we are again able to incorporate the batch-size-based demand rate assumption. Unfortunately this will not be the case for the dynamic policies discussed in the following section. In order to provide a valid comparison between the performance of the static and dynamic policies proposed, our computational tests in Section 6 therefore employ delay costs in lieu of the batch-size-based demand rate assumption. Clearly, we expect that the demand rate will generally decrease in the delay cost h_n for models that use a delay cost instead of a batch-size-based demand rate. Moreover, like explicit shortage costs for stockouts in inventory models (see Nahmias and Lennon Olsen 2020, section 5.5), calibrating the delay cost exactly can be quite difficult or impossible in practice. Equally difficult in practice is precisely mapping a delay cost to a demand rate. The deterministic model then gives a way to calibrate this relationship; that is, we can, for example, use the deterministic version of (9) to obtain an "imputed delay cost" by first using the deterministic model (5) to obtain q and the corresponding demand rate. Then, using this demand rate in (9), and because q is nonincreasing in h_n, we can use (9) to determine the "imputed" or implied value of h_n for use in comparing the static and dynamic policies in the stochastic case.

5. Dynamic Threshold Policies

The policy structures proposed thus far are static in nature, using some fixed value of batch quantity q or elapsed time T_d to trigger a dispatch of order deliveries. Consider, however, a quantity-based dispatch rule for consecutive orders that uses $q = 2$. If two orders arrive consecutively with delivery locations that are extremely far apart, then batching them together may result in a fulfillment time for the second order that is unacceptable, i.e., for which the fixed order dispatch cost is far outweighed by the waiting cost associated with the second delivery on the route. Alternatively, suppose two consecutive orders arrive with a small interarrival time and which are in close proximity to one another. Then, waiting for a third arrival to batch together with these two on a single route may actually result in lower cost and more efficient utilization of resources. It is easy then to see that a dynamic policy that continuously assesses the state of the system, in terms of the number and locations of orders that have arrived since the first order in the batch, and which bases decisions on this system state, is likely to produce superior performance when compared with a static policy. We next describe one such potential policy.

5.1. Cost-Based Dispatch Trigger: *Z*-Threshold Policy

Let us assume for this discussion, for ease of exposition, that orders are batched consecutively, and consider a policy that at each point in time keeps track of the accumulated waiting and delivery cost associated with the delivery of all orders that have arrived since the previous dispatch (in other words, this cost includes all aspects except the fixed cost ϕ). When the accumulated waiting cost associated with orders that have arrived reaches or exceeds some value z, the batch of accumulated orders is dispatched for delivery (we say reaches or exceeds because the accumulated cost may not be continuous; i.e., as soon as a new order arrives, the implied additional waiting and delivery cost required is immediately added to the previously accumulated cost). Thus, we refer to such a policy as a cost-based dispatch trigger, and we refer to the time at which the accumulated waiting cost reaches a value of at least z as the *hitting time*, denoted by $T(z)$. Observe that this policy is stationary with respect to accumulated waiting cost z, but implies dynamic values of batch quantity and dispatch time (denoted by N and T, respectively, in Section 4).

We define $Z(t)$ as the accumulated waiting and delivery cost in the time interval $(0, t]$, where time 0 corresponds to the time of the first order arrival in the batch, and observe that the events $\{Z(t) \le z\}$ and $\{T(z) \ge t\}$ are identical, implying $P\{Z(t) \le z\} = P\{T(z) \ge t\}$. Computing the long-run average cost of the renewal-reward process implied by this policy requires characterizing the distribution of $Z(t)$ (and thereby $T(z)$ by the above relation). Then, using Equation (8) in Theorem 1, the long-run average cost per unit time can be expressed as

$$E[\mathcal{C}(z)] = \frac{\phi + E[Z(T(z)+)]}{E[T(z)] + 1/\lambda}, \tag{17}$$

where $T(z)+$ denotes the instant immediately after $T(z)$, and is necessary as there might have been an order arrival right at $T(z)$ which resulted in a jump. (Figure 4 illustrates the cases when the threshold is reached under two scenarios.)

Our goal then is to characterize $Z(t)$, which can then be used to characterize $T(z)$. Because the accumulated waiting cost $Z(t)$ depends on the number of order arrivals in a time interval of length t, denoted by $N(t)$ (with $N(0) = 1$), we are interested in first characterizing the joint distribution $H_n(t, z)$, defined as

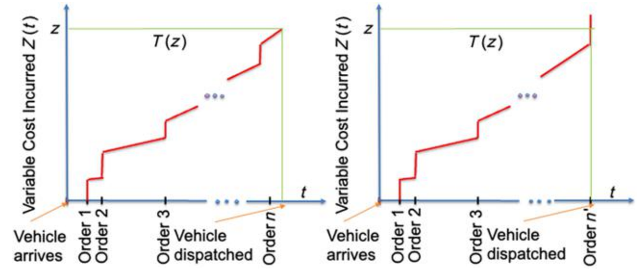


Figure 4. Variable-Cost-Based Z-Threshold (Reached Between Arrivals on Left and at Arrival on Right)

$$H_n(t, z) = P\{Z(t) \le z, N(t) = n\}. \tag{18}$$

The waiting cost after dispatch plus delivery cost in our general TSP-based routing model (with $\gamma(n) = \beta(n) = 1$ $\forall n$ for FCFS) for a batch of n orders can be characterized as equal to $h_n[nX_1 + (n-1)Y_1 + (n-2)Y_2 + \cdots + Y_{n-1}] + \ell[X_1 + Y_1 + Y_2 + \cdots + Y_{n-1} + X_0] = V_1 + V_2 + \cdots + V_n$, where X_1 and Y_j denote the time until the first and $(j+1)$st delivery, respectively, and V_n is a random variable for the additional delivery cost incurred when the nth order is added to a batch, with cumulative distribution function $G_n(\cdot)$. Note that V_n does not include the waiting cost before dispatch. This is with the understanding that $X_i = \gamma(n)X$ and $Y_i = \beta(n)Y$ are IID, with X denoting the random time between departing the fulfillment center and reaching any arbitrary customer, and Y denoting the time between any two delivery locations. Next we state a theorem (with proof in Online Appendix A) showing that the joint distribution $H_n(t, z)$ satisfies a partial differential equation (PDE) in $G_n(\cdot)$ for any $n \ge 2$.

Theorem 2. *The joint distribution $H_n(t, z)$ satisfies the following partial differential equation for all $n \ge 2$:*

$$\frac{\partial H_n(t,z)}{\partial t} + nh_n \frac{\partial H_n(t,z)}{\partial z} = -\lambda H_n(t,z) + \lambda \int_0^z H_{n-1}(t, z-u)\, dG_n(u).$$

Solving this PDE requires establishing some initial conditions and a set of assumptions to ensure a solution exists. First, note that $H_1(0, z) = P\{Z(0) \le z, N(0) = 1\} = G_1(z)$, and

$$H_1(t, z) = e^{-\lambda t} G_1(z - h_1 t). \tag{19}$$

In addition, the initial conditions consist of $H_n(t, 0) = 0$ and $H_n(0, z) = 0$ for $n \ge 2$. We assume that $G_n(\cdot)$ can be characterized using data for all n, and that $V_n \ge 0$, for which a sufficient condition is that the triangle inequality holds for travel times between locations. We further assume that the Laplace–Stieltjes transform (LST) $\tilde{G}_n(s) = E[e^{-sV_n}]$ exists for each n, and that $V_1, V_2, \ldots, V_n$ are independent. The LST of $H_n(t,z)$ with respect to z is defined as

$$\tilde{H}_n(t,w) = \int_0^\infty e^{-wz} \frac{\partial H_n(t,z)}{\partial z} dz.$$

Then, if we take the LST of the above PDE with respect to z, we obtain the following ordinary differential equation (ODE) for each $n \geq 2$:

$$\frac{d\tilde{H}_n(t,w)}{dt} = -nh_n w\tilde{H}_n(t,w) - \lambda\tilde{H}_n(t,w) + \lambda\tilde{H}_{n-1}(t,w)\tilde{G}_n(w). \tag{20}$$

Using the fact that $H_1(t,z) = e^{-\lambda t} G_1(z - h_1 t)$ from Equation (19), we can show that

$$\tilde{H}_1(t,w) = e^{-(\lambda + h_1 w)t}\tilde{G}_1(w), \tag{21}$$

which is a result that will be used in the proof of the next theorem, which essentially solves the ODE (20). However, the integral that is part of solving the ODE can be performed only on a case-by-case basis depending on the h_n structure. As in the previous section, we again present the two special cases with $h_n = h$ and $h_n = (n-1)h$.

For any generic $n \geq 1$, we have the following (with the proof presented in Online Appendix A).

Theorem 3. *The solution to ODE (20) in the form of the LST for the special case $h_n = h$ is*

$$\tilde{H}_n(t,w) = \frac{(\lambda/(hw))^{n-1}}{(n-1)!} \tilde{G}_1(w)\tilde{G}_2(w)\ldots\tilde{G}_n(w)e^{-(\lambda+nhw)t}[e^{hwt}-1]^{n-1},$$

and for the special case $h_n = (n-1)h$ is

$$\tilde{H}_n(t,w) = \frac{(\lambda/(hw))^{n-1}}{(2n-2)!}\tilde{G}_1(w)\tilde{G}_2(w)\ldots\tilde{G}_n(w) \sum_{i=1}^{n} \alpha_{n,i} e^{-(\lambda+i(i-1)hw)t},$$

where both special cases are for all $n \geq 1$, and for the special case $h_n = (n-1)h$,

$$\alpha_{n+1,i} = \binom{2n}{n-i+1} \frac{2i-1}{n+i}(-1)^{i+1} \tag{22}$$

for all $i \in \{1, 2, \ldots, n+1\}$.

Inverting the LST in Theorem 3 to obtain the distribution of $Z(t)$, that is, $P\{Z(t) \leq z\}$, is intractable. However, it is possible to obtain this numerically, as several algorithms exist for this. But, to obtain the long-run average cost per unit time $E[\mathcal{C}(z)]$ in Equation (17), all we need are $E[T(z)]$ and $E[Z(T(z)+)]$, as shown in the next theorem, with proof in Online Appendix A.

Theorem 4. *A lower bound for $E[\mathcal{C}(z)]$ is*

$$E[\mathcal{C}_{LB}(z)] = \frac{\phi + z}{E[T(z)] + 1/\lambda},$$

where $E[T(z)]$ can be computed by inverting the LST in Equation (30) for the special case of $h_n = h$, and in Equation (31) for the special case of $h_n = (n-1)h$. Furthermore, as an approximation for $E[\mathcal{C}(z)]$, we have, for the special case of $h_n = h$,

$$E[\mathcal{C}_{approx}(z)] = \frac{\phi + z + \beta(q^*)(q^*h/2 + \ell)\frac{Var[Y]+\tau^2}{2\tau}}{E[T(z)] + 1/\lambda},$$

and for the special case of $h_n = (n-1)h$,

$$E[\mathcal{C}_{approx}(z)] = \frac{\phi + z + \beta(q^*)(q^*(q^*-1)h/2 + \ell)\frac{Var[Y]+\tau^2}{2\tau}}{E[T(z)] + 1/\lambda}.$$

Using the results in Theorem 4, we find the value of z that minimizes $E[\mathcal{C}_{approx}(z)]$ with the conjecture that the minimum for $E[\mathcal{C}(z)]$ also occurs at the same value of z. We will perform simulations to empirically evaluate the effectiveness of the approximation and illustrate the lower bound derived in Theorem 4.

5.2. Cost- and Quantity-Based Dispatch Trigger: The *Zq**-Threshold Policy

In Section 5.1, although the state of the random process was two-dimensional, that is, $\{(Z(t), N(t)), \forall t \geq 0\}$, for our threshold, we only considered a hitting time to reach cost z. In the process, we ignored explicit consideration of an important element, namely, the number of orders received (which is implicitly considered only through its impact on cost). Space and scope considerations preclude analyzing a general two-dimensional dynamic policy in detail within this article. However, we present a restricted version of this class of policies and use this approach in our numerical study discussed later. Suppose we precompute q^*, the optimal batch size derived in Section 4, and consider the following policy. When either the accumulated waiting and delivery cost reaches z or the number of orders equals q^*, the batch is dispatched for delivery. We refer to this as the *cost- and quantity-based dispatch trigger*, and call this the Zq^*-threshold policy.

All the analysis in Section 5.1 for this special dispatch rule remains unchanged until the end of Theorem 3. Then, we obtain the long-run average cost per unit time under this policy, which is denoted by $E[\mathcal{C}(z|q^*)]$. As before, to use Equation (17), all we need are $E[T(z|q^*)]$ and $E[Z(T(z|q^*)+)]$. We compute $E[T(z|q^*)]$, using

$K(z|q^*)$ as

$$K(z|q^*) = E[T(z)|q^*] = \int_0^\infty P\{T(z) \geq t, N(t) \leq q^*\}dt$$
$$= \int_0^\infty P\{Z(t) \leq z, N(t) \leq q^*\}dt$$
$$= \sum_{n=1}^{q^*} \int_0^\infty P\{Z(t) \leq z, N(t) = n\}dt$$
$$= \sum_{n=1}^{q^*} \int_0^\infty H_n(t,z)dt.$$

Taking the LST of $K(z|q^*)$, denoted by $\tilde{K}(w|q^*)$, when $h_n = h$, we obtain

$$\tilde{K}(w|q^*) = \sum_{n=1}^{q^*} \left(\frac{\lambda}{hw}\right)^{n-1} \tilde{G}_1(w)\tilde{G}_2(w)\ldots\tilde{G}_n(w) \frac{1}{hw}\frac{\Gamma(\lambda/(hw)+1)}{\Gamma(\lambda/(hw)+1+n)}, \quad (23)$$

and, when $h_n = (n-1)h$, we have

$$\tilde{K}(w|q^*) = \sum_{n=1}^{q^*} \left(\frac{\lambda}{hw}\right)^{n-1} \frac{\tilde{G}_1(w)\tilde{G}_2(w)\ldots\tilde{G}_n(w)}{(2n-2)!} \sum_{i=1}^{n} \frac{\alpha_{n,i}}{\lambda + hw(i-1)i}, \quad (24)$$

both of which can be numerically inverted.

Next, for $E[Z(T(z|q^*)+)]$, as an approximation when $h_n = h$, we use

$$E[Z(T(z|q^*)+)]_{approx} = \min\left\{z + \beta(q^*)(q^*h/2 + \ell)\frac{Var[Y] + \tau^2}{2\tau}, E[\mathcal{C}(q^*, T)]q^*/\lambda - \phi\right\},$$

where $E[\mathcal{C}(q^*, T)]$ is from Equation (10) computed at $q = q^*$. Likewise, for an approximation when $h_n = (n-1)h$, we use

$$E[Z(T(z|q^*)+)]_{approx} = \min\left\{z + \beta(q^*)(q^*(q^*-1)h/2 + \ell)\frac{Var[Y] + \tau^2}{2\tau}, E[\mathcal{C}(q^*, T)]q^*/\lambda - \phi\right\},$$

where $E[\mathcal{C}(q^*, T)]$ is from Equation (11) computed at $q = q^*$.

Thus, as an approximation for $E[\mathcal{C}(z|q^*)]$, using the appropriate expressions for the cases of $h_n = h$ and $h_n = (n-1)h$, we compute

$$E[\mathcal{C}_{approx}(z|q^*)] = \frac{\phi + E[Z(T(z|q^*)+)]_{approx}}{K(z|q^*) + 1/\lambda}, \quad (25)$$

and find the value of z that minimizes $E[\mathcal{C}_{approx}(z|q^*)]$.

6. Numerical Results

Although we performed detailed analysis, several research questions remain: (a) Does the model capture real-life data, in terms of travel times and distances, as well as the TSP approximation? (b) How do the approximation and the bound derived in Section 5 perform? (c) Does the dynamic policy always dominate the static policies? (d) How do the costs compare between TSP- and FCFS-based delivery? (e) Does taking into account more state information yield better performance? (f) Are there conditions when one policy does better than the others?

Motivated by these questions, this section discusses computational experiments designed to assess the performance of the various policies presented. We begin our numerical experiments by first considering a mini case study consisting of delivery locations in three cities in the United States in Section 6.1, both for the time-invariant and time-varying delay cost cases. Then, in Section 6.2, we evaluate performance across a broad set of randomly generated problem instances with a goal of characterizing conditions under which a given policy might be preferred. Section 6.3 provides an analysis of policy performance based on an additional set of tests intended to characterize the effects of individual cost and distance parameters on policy performance. A detailed discussion of these tests is provided in Online Appendix B.

6.1. A Tale of Three Cities: A Mini Case Study

We arbitrarily chose Austin, Texas; Boston, Massachusetts; and Portland, Oregon, Airbnb data sets, and selected consecutive delivery locations by randomly sampling from the list of locations within a city's corresponding data set. For each city, we estimated $\gamma(n)$ and $\beta(n)$ for various n as shown in Figure 3. Table 1 summarizes each region's parameters used in the numerical analysis for the special case of delay cost $h_n = h$. As

Table 1. Numerical Values Used for the Simulation and Analytical Results (for the $h_n = h$ Case)

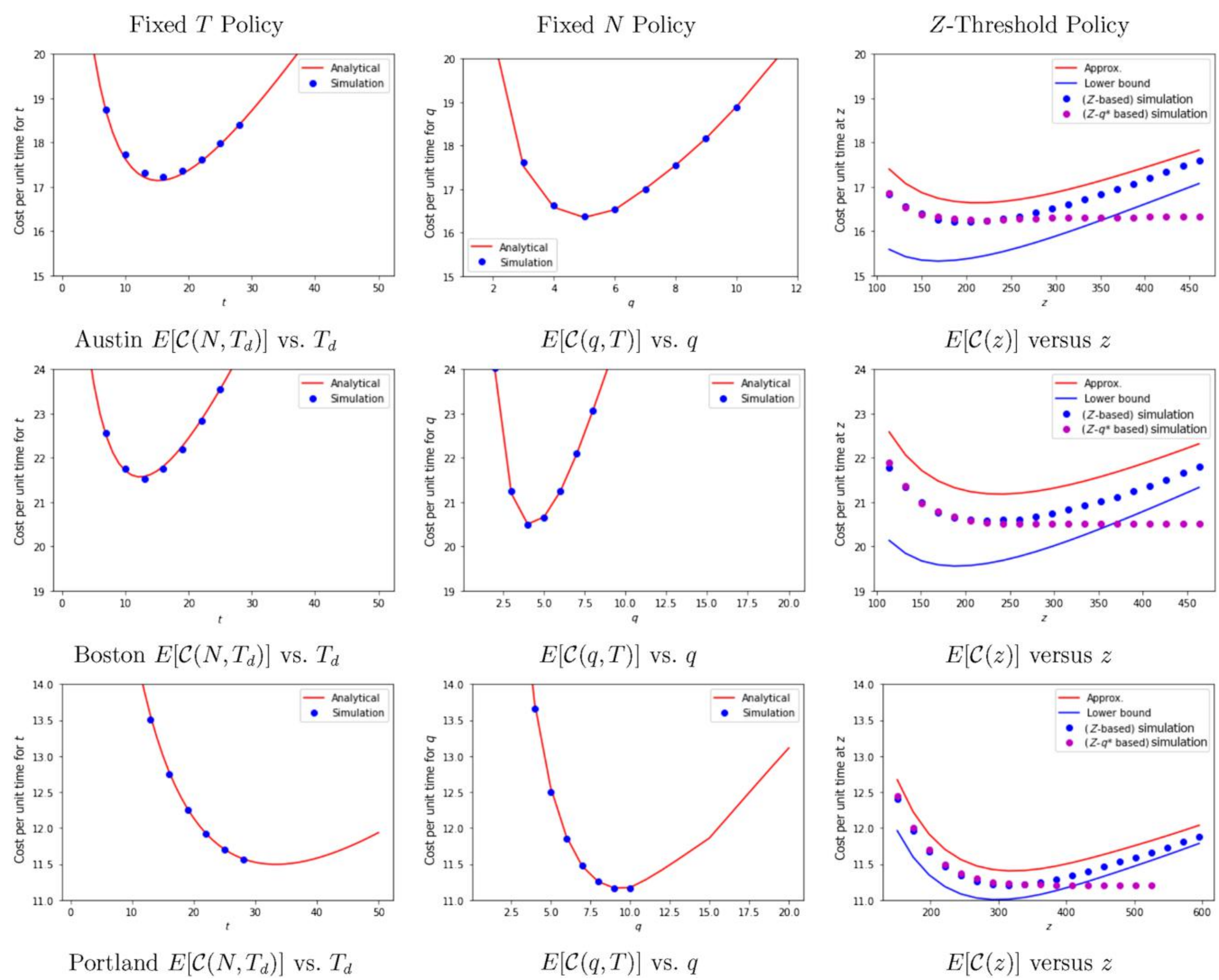
	ϕ	h	λ	ℓ	θ	τ	$Stdev[X]$	$Stdev[Y]$
Austin	100	1.2	0.25	1.6	6.143	9.0029	4.9151	6.4606
Boston	100	2.4	0.25	4.8	3.3258	4.7341	2.0358	2.9010
Portland	100	0.6	0.25	3.2	4.5371	6.3507	2.4422	3.5176

described in Section 5, X denotes the time between the fulfillment center and a randomly selected delivery location, whereas Y denotes the time between two randomly selected delivery locations. Recall that $E[X] = \theta$ and $E[Y] = \tau$. Then we fit a gamma distribution to the distance data for X and Y, with parameters computed from the mean and standard deviation of the data set in each case. (Polus (1979) finds that a gamma distribution is appropriate for travel times on arterial routes, whereas Lin et al. (2012) validate the application of a gamma distribution for random local travel times in a metropolitan area.) Recall that the LST of a gamma random variable Z with parameters α and β is $E[e^{-sZ}] = (\beta^{-1}/(\beta^{-1} + s))^{\alpha}$.

We considered the performance of four threshold policies: (i) time (T_d)–based dispatch with cost $E[\mathcal{C}(N, T_d)]$ per unit time, (ii) quantity (q)–based dispatch with cost $E[\mathcal{C}(q, T)]$ per unit time, (iii) cost (z)–based dispatch with cost $E[\mathcal{C}(z)]$ per unit time (Z-threshold policy), and (iv) cost (z) and quantity (q^*)–based dispatch with cost $E[\mathcal{C}(z|q^*)]$ per unit time (Zq^*-threshold policy). Our goal is to compare cost and service performance to understand how these vary with various parameter choices and how they compare against each other. For the time- and quantity-based policies (i) and (ii), we used both analytical results as well as simulations. For the Z-threshold policy (iii), we used analytical results to get a lower bound and an approximation, as well as simulations. For the Zq^*-threshold policy (iv), we used only simulations, as the analytical models were unstable. We first illustrate comparative policy costs by plotting $E[\mathcal{C}(N, T_d)]$ versus T_d, $E[\mathcal{C}(q, T)]$ versus q, and $E[\mathcal{C}(z)]$ versus z (which also includes $E[\mathcal{C}(z|q^*)]$ versus z). In all cases, there is a significant drop in cost between leaving immediately after the first arrival and waiting for additional orders.

Before considering the results, it is worthwhile discussing a few computational aspects, especially with respect to the right panels in Figures 5 and 7. We built

Figure 5. Average Cost per Unit Time for TSP dispatch Using Various Policies



a code in Python to numerically invert the LST in Equation (30) and then compute $K(z)$. For this, we first consider a small n (such as 15) and then keep adding terms until there is no gain in increasing n (in our case, at about $n = 50$). We obtain the lower bound in Equation (32) as well as the approximation in Equation (33) for the case of $h_n = h$, and that in Equation (34) for the case of $h_n = (n-1)h$. Corresponding graphs of these functions are shown in the right panels of Figures 5 and 7. The right panels also include simulations of the Zq^*-threshold policy (iv), which appears to flatten out. The reason for this is that when z is large, the threshold q^* is typically reached, and hence the results mimic those of the fixed q^* policy (for this reason, the inversion algorithms for the analytical results of policy (iv) are unstable). Note that we performed numerical simulations to obtain the dotted points in all figures. The perfect match between the simulation and analytical results for the left and center panels is illustrated to serve as validation of the results. The following two subsections discuss the results for the two special cases of $h_n = n$ and $h_n = (n-1)h$, respectively, in greater detail.

6.1.1. The Special Case of $h_n = h$. We used the parameters shown in Table 1 for the computations that resulted in Figure 5. The optimal objective function value is somewhat similar under all four policies, although the graphs differ, with the dynamic policies having a bathtub-like shape. In all cases, the time-based threshold is not optimal. However, the other policies are nearly tied, with the quantity-based threshold performing slightly better in Portland. We noted earlier that $E[\mathcal{C}(q^*,T)] < E[\mathcal{C}(N,T^*)]$ for the FCFS case. However, this inequality is guaranteed to hold only for exponential order interarrival times. Because the exponential distribution has the memoryless property, there is no point in waiting for an additional order. Hence, it is not surprising that the fixed batch size policy performs better than the fixed time policy. In our simulations with nonexponential interarrival distributions (not shown here), the fixed T policy does perform better in some cases. Notice from the right panels of Figure 5 how the lower bound and the approximation behave. In all the examples, the argument of the minima is nearly the same for the approximation and the simulation, and the approximation is closer to the simulations than the lower bound.

For the dynamic Z- and Zq^*-threshold policies (iii) and (iv), we observe a large range of z values where the objective function value is flat and, hence, robust to some uncertainty in parameter estimation. But, is there a relationship between the optimal values across the policies? The optimal T^* when T_d is fixed and the optimal q^* when q is fixed are related as $q^* - 1 \approx \lambda T^*$; that is, the number of additional orders we wait for roughly equals the expected number of orders that arrive in time T^*. However, the optimal cost threshold z^* does not correspond to the expected variable cost of a single delivery with q^* orders, which equals $E[\mathcal{C}(q^*,T)]q^*/\lambda - \phi$. This is because the resulting variable delivery cost can be greater than z for this policy (as we serve the order that resulted in overshooting the threshold). Also, from the graphs, the Zq^*-threshold (cost- and quantity-based) policy (iv) diverges a little beyond where z^* occurs in the cost-based Z-threshold policy (iii).

Recall that the model contains a fixed cost, ϕ; a delay cost rate, h; and the variable delivery cost, ℓ. For each city (Austin, Boston, and Portland) and policy (i.e., time based, quantity based, cost based, and cost and quantity based), we characterize how the optimal cost per unit time is divided among these cost categories, in order to gain some insight into the way in which different policies address the trade-off between delivery cost and service, measured by delay cost. To refine the analysis, we split the delay cost into two components: before and after dispatch. Note that we used simulations to obtain the cost fractions for the cost-based threshold because the analytical model only uses z and that cannot be used to obtain the various components.

Figure 6 shows how the percentage cost at the optimal policy is divided across the four factors: fixed cost, delay cost before dispatch, delay cost after dispatch, and delivery cost. Notice from the figure that the split does not change significantly from policy to policy; thus, it is not surprising that the optimal values are nearly the same. Across all cases, observe that the total delay costs are between 44% and 52% of the total cost, with the waiting costs incurred during delivery comprising at least 25% of total cost. The relatively high delay cost incurred during delivery for Austin coincides with the relatively high TSP tour length illustrated in Figure 3. Isolating these delay costs permits gauging the service performance of the different policies, as these costs correspond to scaled values of the average order delay. In each of the regions, the Zq^* policy achieves a minimum delay cost percentage, whereas the time-based threshold policy matches this in two of the three regions. Thus, the Zq^* policy achieves the strong cost performance of the quantity-based policy while also matching the strong service delay performance of the time-based policy.

6.1.2. The Special Case of $h_n = (n-1)h$. To consider the impacts of a nondecreasing delay cost rate with time, we used the parameters shown in Table 1, except for the values of h. Because this special case assumes $h_n = (n-1)h$, using the values of h in Table 1 would make comparison with the time-invariant delay cost case impractical. Instead, we considered two scenarios for setting the value of h for this special case, as shown

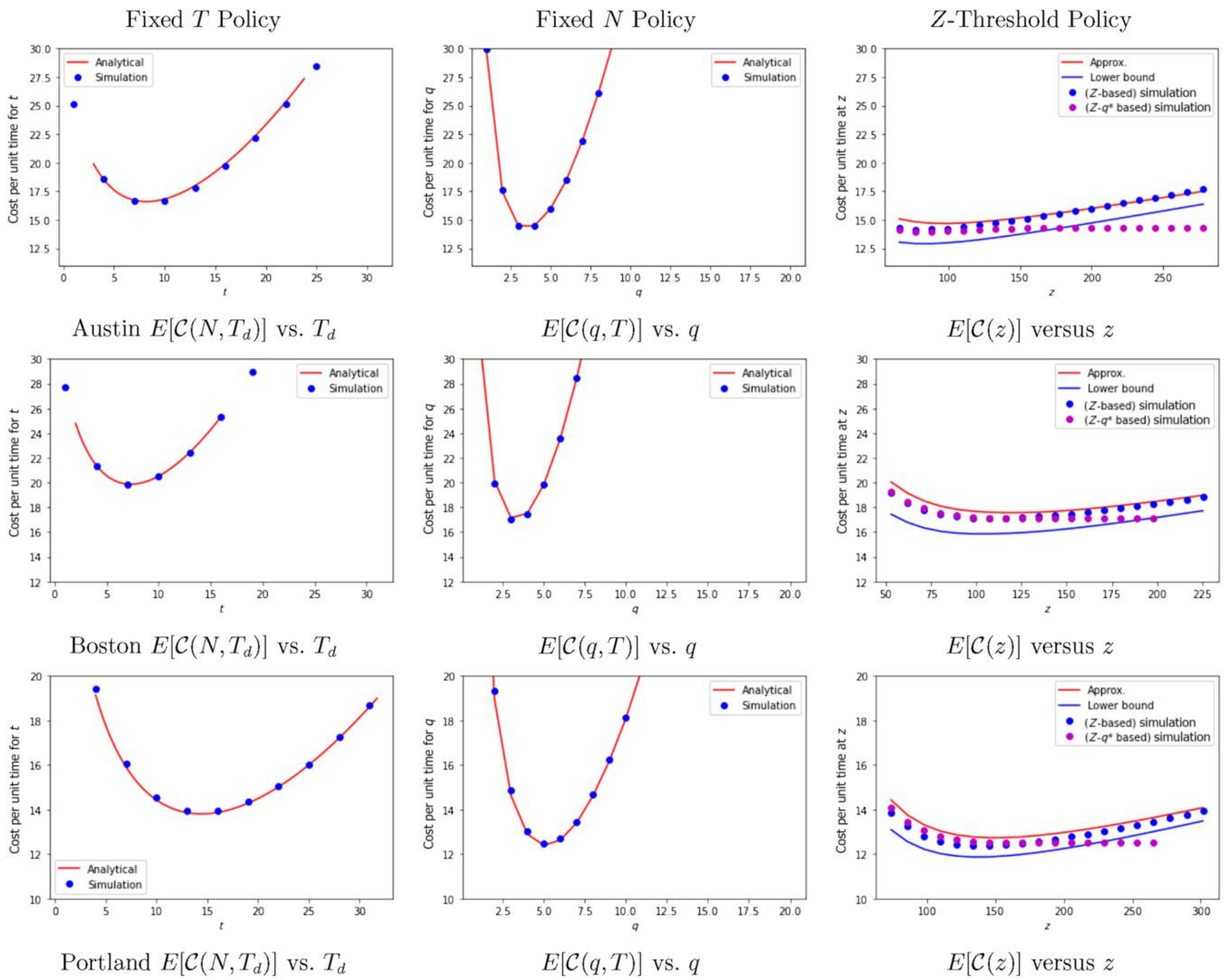
Figure 6. At Optimal Points of Policies, Percentage of Average Cost Per Unit Time Across Factors

in Table 2. Note that the values of h for Scenario 1 in Table 2 are 20% of those from Table 1; thus, for dispatch quantities less than six, the delay cost rate will be less than the time-invariant delay cost rate from Table 1, whereas higher batch quantities will imply a higher delay cost rate. The values of h for Scenario 2 are 40% of those in Table 1, implying that dispatch quantities less than or equal to three will see a lower delay cost rate than in the time-invariant case, whereas higher values imply a higher delay cost rate.

Using the values of h for both scenarios from Table 2 and the remainder from Table 1, we performed numerical experiments for the same three cities under the case of a delay cost rate of $h_n = (n-1)h$. For Scenario 1, because the batch quantities for Austin and Boston were already below six, there was very little change in the optimal quantity and time thresholds when compared with the time-invariant h case, although the expected costs decreased in both cases because of the lower delay cost rate at those batch sizes. For Portland, because the optimal quantity threshold in the time-invariant delay cost case was higher than six (at around nine), under the $h_n = (n-1)h$ model in Scenario 1, both the quantity and time thresholds were driven down as a result of the increasing delay cost rate. It is interesting to note that for Portland, the optimal quantity threshold went down from around nine to about seven, implying a higher delay cost rate than in the time-invariant delay cost case. Despite this, the expected cost per unit time at the optimal quantity threshold actually *decreased*: although the delay cost rate was higher, the average delay time for orders was sufficiently small as to reduce expected total cost.

The results corresponding to the Scenario 2 h values in Table 2 more dramatically illustrate the implications of an nondecreasing delay cost rate for all three regions, and are shown in Figure 7. For all three regions, the optimal quantity and time thresholds are substantially smaller than in the time-invariant delay cost case. For Austin and Boston, the expected cost per unit time is also smaller; however, for Portland, a batch size small enough to achieve similar delay costs to the time invariant case

Figure 7. Average Cost per Unit Time for TSP Dispatch Using Various Policies for $h_n = (n-1)h$



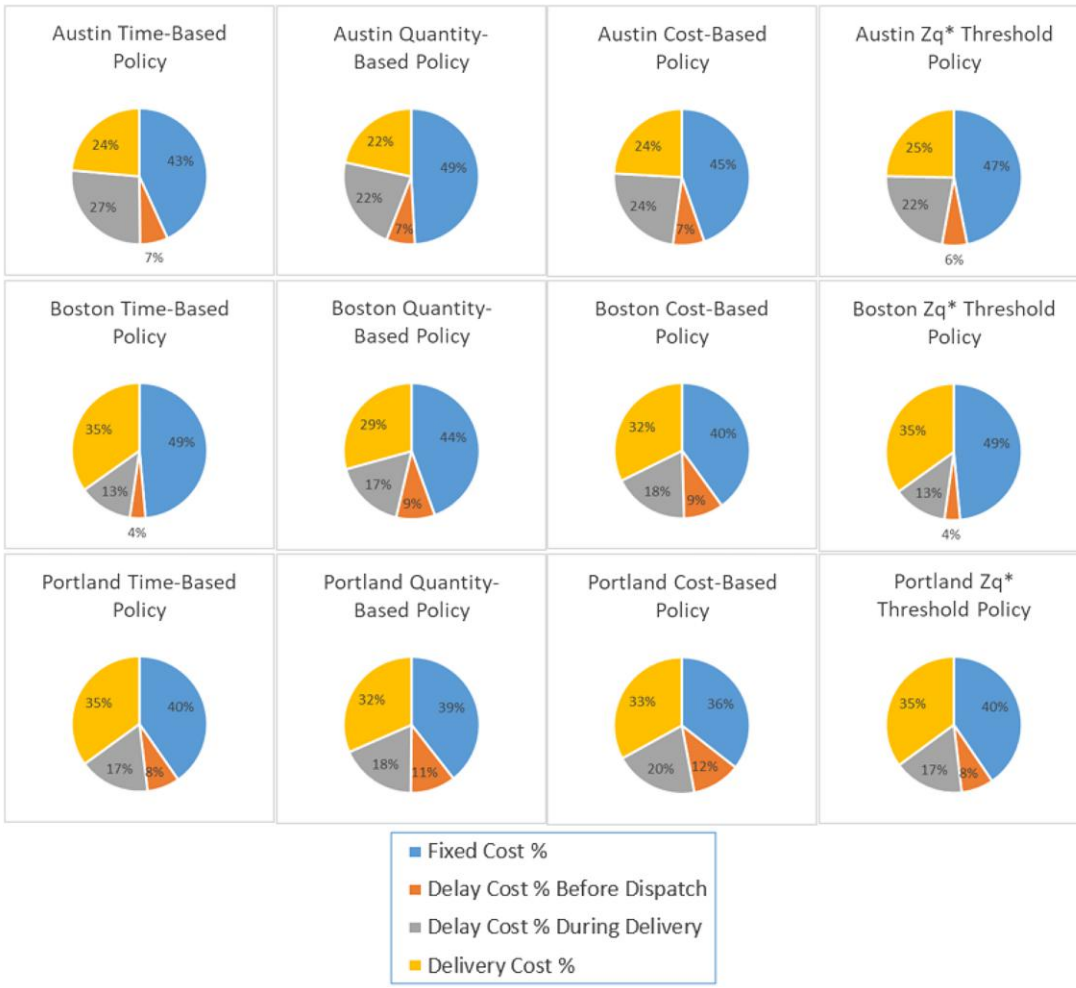
requires incurring significantly higher fixed plus delivery costs, and the reduced delay times are not sufficient to offset the increased delay cost rate in this scenario. For this set of tests, again the optimal objective function value was somewhat similar under all four policies. As before, in all cases, the time-based threshold is not optimal. The other policies are nearly tied, with the quantity-based threshold performing slightly better in Boston and Portland when compared with the cost-based threshold (policy (iii)). However, the Zq^* policy, which uses cost and quantity, outperformed the quantity-based threshold. The right panels of Figure 7 illustrate how the lower bound and approximation behave relative to one another. In this case, as before, the argument of the minima is nearly the same for the approximation and the simulation, and the approximation is closer to the simulations than the lower bound.

Figure 8 illustrates how the percentage cost at the optimal policy is divided across the four cost factors under the Scenario 2 delay costs. Notice from the figure that the distribution of costs is much more uneven when compared with the $h_n = h$ case, as the increasing delay cost rate favors lower batch sizes and therefore lower average order delay costs. This effect, however, does not significantly differ from policy to policy; thus, it is not surprising that the optimal values are nearly the same. For this case, again, the Zq^* policy achieved a minimum delay cost percentage, and the time-based threshold policy matched this in one of the three regions. However, the worst-case delay cost as a percentage of overall cost was 33% in this case (compared

Table 2. Numerical Values of h Used for the Simulation and Analytical Results (for the $h_n = (n-1)h$ Case)

Scenario	Austin	Boston	Portland
1	0.24	0.48	0.12
2	0.48	0.96	0.24

Figure 8. At Optimal Points of Policies, Percentage of Average Cost per Unit Time Across Factors for the $h_n = (n-1)h$ Case



with 52% in the time-invariant delay cost rate case in the previous section).

6.2. Great Expectations: Empirical Policy Comparison

Duly acknowledging Charles Dickens' works for the titles of this and the previous subsection, with great expectations we ask whether we can come up with rules regarding when the fixed dispatch quantity policy (ii) works best versus the other dynamic policies. To gain some insight, we designed an experiment with settings that put the results to the test, such as ignoring the triangle inequality, using extreme parameter values, and widely varying times. The alternate values used in the experimental design are shown in Table 3. Note that $\{\beta(n), \gamma(n)\} = \{1, 1\}$ is the FCFS case, whereas the other two sets are TSP cases. We only present the special case of $h_n = h$, as the insights do not significantly vary with this case.

Because there are three choices of $\{\beta(n), \gamma(n)\}$ and two choices each of ϕ, h, λ, ℓ, θ, τ, $Stdev[X]$, and $Stdev[Y]$, there was a total of 768 experiments. However, when $\ell = 5$ and $h = 0.3$, in the TSP cases, because $\beta(n)$ and $\gamma(n)$ decrease with n, for some sufficiently large n, the random variable V_n (the additional delivery cost when the nth order arrives) can be negative. This is because the delay cost (being small) is dominated by ℓ. In other words, the graph in Figure 4 is no longer

Table 3. Numerical Values Used for Experimental Design

Type: $\{\beta(n), \gamma(n)\}$	ϕ	h	λ	ℓ	θ	τ	$Stdev[X]$	$Stdev[Y]$
TSP 1: $\{1/\sqrt{n}, \min(0.8/\sqrt{n}, 1)\}$	200	3	0.1	5	2.8	3	0.25θ	0.25τ
TSP 2: $\{\min(1.7/\sqrt{n}, 1), \min(1.5/\sqrt{n}, 1)\}$	50	0.3	2.5	0.5	40	5.5	2θ	2τ
FCFS: $\{1, 1\}$								

nondecreasing, as some of the vertical jumps may go downward. For this reason, those instances were removed from our analysis, and we analyzed a total of 640 experiments (instead of 768). A further 11 instances produced too few replications where the cost was not increasing in the number of orders. Hence, in total we had 629 valid experiments.

Before presenting the results, it is important to discuss a few caveats. One is that, for each of the experiments, we ran 1,000 replications of simulations to get $E[\mathcal{C}(z)]$ for various z. When the standard deviation of X or Y is at least twice its mean, this implies that 1,000 replications is not sufficiently high to estimate the expected cost with precision. When the variability is so high, there are often many extremely small values and a few very large values of cost (in our results, we found two orders of magnitude differences in the spread). This would also result in violating the triangle inequality. We built 95% confidence intervals for the costs and used them in the comparisons. Another caveat is that in order to determine $E[\mathcal{C}(z^*)]$ and $E[\mathcal{C}(z^*|q^*)]$ based on simulated values of $E[\mathcal{C}(z)]$ and $E[\mathcal{C}(z|q^*)]$, respectively, we enumerate 20 values of z that are equally spaced (and this spacing depends on the magnitude of the expected cost of the policy with a fixed order quantity, i.e., $E[\mathcal{C}(q^*, T)]$). This caveat also applies to the results in Section 6.1.

In all 629 experiments, we observed $E[\mathcal{C}(q^*, T)] < E[\mathcal{C}(N, T^*)]$, which leads us to believe that in the TSP case (in the FCFS case this can be proved) with Poisson arrivals, a policy based on a fixed batch quantity outperforms one based on time. However, in 39% of the cases, the three quantity-based policies (static quantity-based policy (ii), Z-threshold policy (iii), and Zq^*-threshold policy (iv)) were not statistically significantly different (based on the values falling within the associated confidence intervals). Only in 2.7% of the cases, however, was the static quantity-based threshold policy (ii) a clear winner. In 17% of the cases, the Z-threshold dynamic policy based on cost alone, policy (iii), was better than the Zq^*-threshold policy with dynamic cost and quantity, policy (iv), and this is based on the fact that their confidence intervals do not intersect. Likewise, in 15% of cases, the results were reversed, that is, the Zq^*-threshold dynamic policy was statistically better. In the remaining nearly 26% of the runs, the Z-threshold and the Zq^*-threshold policies were statistically equal, and both better than the static quantity-based policy.

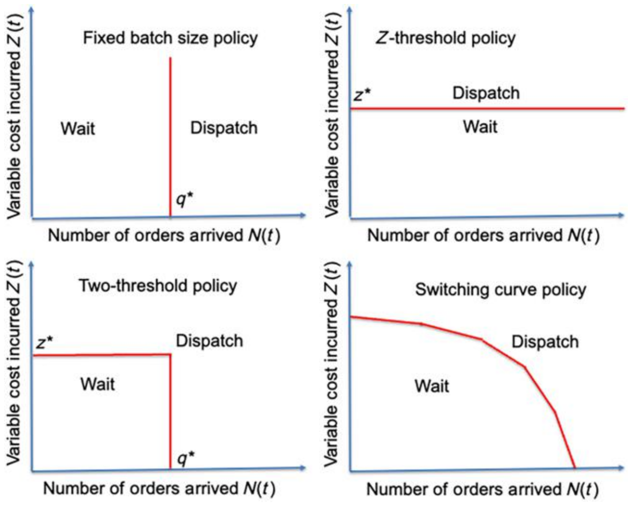
In summary, in our experiments, the time-based dispatch policy was dominated by all other policies. However, neither the Z-threshold (delivery-cost-based threshold) policy, nor the Zq^*-threshold (cost- and quantity-based) dynamic policy was universally better than the static quantity-based policy, although in most cases they were fairly close. Unfortunately, there was no clear pattern in terms of parameters to indicate when a quantity-based dispatch policy works best (however, later we do present some aggregated results that provide a clearer picture). We note that the approximation $E[\mathcal{C}_{approx}(z)]$ in Equation (33) was within the confidence interval of $E[\mathcal{C}(z^*)]$ in 31% of the experiments, and an upper bound (i.e., higher than the upper confidence interval) in over 54% of the experiments. However, the approximation can be higher or lower than $E[\mathcal{C}(z^*)]$ (on average, about 3% higher). On one hand, we observed a trend in which the approximation is much smaller than $E[\mathcal{C}(z^*)]$ in a majority of cases when h is high ($h=3$), θ is high ($\theta=40$), ϕ is low ($\phi = 50$), ℓ is low (i.e., $\ell = 0.5$), $Stdev[X]$ is high (i.e., 2θ), and $Stdev[Y]$ is low (i.e., 0.25τ). On the other hand, the approximation is much larger than $E[\mathcal{C}(z^*)]$ when h is high ($h=3$), θ is low ($\theta = 2.8$), ϕ is low ($\phi = 50$), and $Stdev[Y]$ is high (i.e., 2τ).

6.3. Policy Performance Analysis

Online Appendix B discusses the results of additional computational tests intended to assess the impact of individually varying each of the parameter values h (delay cost rate), ℓ (delivery time cost rate), ϕ (fixed dispatch cost), τ (expected travel time between customers), θ (expected travel time between customer and depot), and the coefficient of variation of travel times. These results generally conform to expectations, with higher delivery-related costs leading to higher batch quantities, and higher delay costs reducing the average batch delivery quantity. A somewhat less expected result we observed was that the cost performance advantage of the dynamic policies increased significantly over the performance of the quantity threshold policy as the coefficient of variation values increased. This is because the dynamic policies are not required to wait for the required quantity threshold when a high cost has accumulated but the required quantity has not.

In further comparing the dynamic policies to the quantity threshold policy, let us revisit the issue of the fixed batch size q^* being close to the average batch size $E[N(T(z^*)+)]$ under the dynamic policy. Only in a little over 20% of the cases was the absolute difference between q^* and $E[N(T(z^*)+)]$ more than one, indicating that the policies are doing similar things. Also, on average, across all the runs, $E[N(T(z^*)+)]$ is about 97% of q^* in the corresponding problem settings. Now, let us compare z^* in the dynamic policy against the average delivery cost of the fixed batch size policy given by $E[\mathcal{C}(q^*, T)]q^*/\lambda - \phi$, and let us call this $E[Z(q^*)]$. Then, in about 80% of the cases $E[Z(q^*)]$ was higher than z^* (understanding that $E[Z(T(z^*)+)] > z^*$ would serve as a more appropriate point of comparison). This is the reason the dynamic policy based on both cost and quantity (the Zq^*-threshold policy) works well, because

Figure 9. Analyzed Policies and Conjectured Policy



it is beneficial to dispatch when either the cost or the quantity threshold is reached, whichever occurs first.

In particular, in almost 84% of the cases, only one of the two happened: either $q^* < E[N(T(z^*)+)]$ or $z^* < E[Z(q^*)]$. This means that in a vast majority of cases, we use whichever happens first, that is, either a threshold q^* batch size is reached or a delivery cost z^* is reached, at which time we dispatch. This is the two-dimensional dynamic policy based on cost and quantity. In fact, we can summarize our three policies by observing the state as the tuple $\{N(t), Z(t)\}$ at time t. In the fixed batch size policy, we say that at time t, if $N(t) < q^*$ we wait, and otherwise dispatch. In the Z-threshold policy, we say that if $Z(t) < z^*$, we wait, and otherwise dispatch. In the time-and-cost policy, we wait as long as both $N(t) < q^*$ and $Z(t) < z^*$, and otherwise dispatch. Is a dynamic policy based on fixed cost and fixed quantity (whichever is reached first) the optimal one? Although we leave this as an open and high-value direction for further research, we conjecture that the optimal policy is likely to consist of a more generic switching curve (the development and analysis of which is beyond the scope of this paper). These four types of policies are shown in Figure 9.

7. Concluding Remarks and Future Work

This work first analyzed a delivery system using a deterministic model and then extended this to a stochastic version. For the stochastic setting, we considered three types of costs: a fixed dispatch cost, a delay cost per unit time for each order, and a cost per unit time for delivery. Our model considered two types of delay costs: a cost that is linear in the delay time and a delay cost that increases in the batch size, and thus the batch accumulation and delivery time. We sought a policy that minimizes the long-run average cost per unit time. We considered two policies that are fairly traditional and agnostic to the delivery locations: a fixed batch size (when a batch size q is reached the vehicle is dispatched) and a fixed time (the vehicle waits a fixed time T_d after first order arrives, and is then dispatched). We considered two delivery strategies: a TSP and an FCFS tour, where the order of delivery matches the order arrival sequence.

We also considered a dynamic policy that accounts for the delivery locations of orders that have arrived. We wait until total delivery cost (i.e., all costs except fixed costs) reaches a threshold z, and then dispatch. We extended this to the dynamic policy where we wait either until a threshold cost z is reached or the batch size is q^* (the optimal one corresponding to the fixed batch size), whichever happens first. We analyzed each of these policies using renewal processes. Then, using the renewal-reward theorem, we obtained the long-run average cost per unit time. For the dynamic policy with threshold z, we used partial differential equations to characterize the joint dynamics of the total delivery cost at time t, $Z(t)$, and the number of orders at time t, $N(t)$. We used LSTs to solve the differential equations and inverted the transforms numerically. The key methodological contributions of the work are in obtaining expressions for long-run average costs, and in the analysis of the z-threshold policy.

We implemented the policies for delivery data sampled from Airbnb locations and showed in those examples that all the policies produced similar cost per unit time. However, when we simulated examples through an experimental design setup, we found some cases where the fixed batch size policy was the best and other cases where the dynamic policy was the best. As a compromise, the dynamic policy with cost *and* quantity thresholds generally performs well in all cases. It should be noted that all the policy analyses required a Poisson order arrival process. However, when the arrivals are generic renewal processes, the fixed time model can be better than the fixed batch size. Furthermore, we conjecture that the optimal policy is a switching curve, although this is generally intractable analytically. It may be possible to perform a simulation optimization and obtain thresholds $z_1, z_2, z_3, \ldots$, so that when there are $1, 2, 3, \ldots$ orders and the delivery costs are lower than the thresholds, we wait (or else dispatch).

There are many paths that one could take in the future. One is to incorporate revenue of the ordered items (especially with multiple orders) and obtain a policy that maximizes the expected revenue per unit time. Also, in certain settings, it may be advantageous to consider multiple parallel batching operations (perhaps to different parts of the delivery region). So instead of solving just the batching and dispatching problem, we also perform matching of vehicles to deliveries. This itself can be done either statically by creating regions or dynamically as orders arrive. This paper can be used to design an optimal number of vehicles based on a desired level of service. One could think of many other extensions, like adding constraints, using multiple pickup locations, and adding order processing times, which we have not considered.

Acknowledgments

The authors gratefully acknowledge the related research undertaken by their students Breanna Bassett, Sudarshan Rajan, and Jin Xu. The work was performed when the authors were both at Texas A&M. The authors thank the reviewers, associate editor, and special issue editors, whose comments and suggestions significantly improved the content and presentation of this work.

Endnote

[1] Observe that if $\gamma(n) = 1$ and $\beta(n) = b_\beta/\sqrt{n}$, then the route time $2\gamma(n)\theta + \beta(n)(n-1)\tau$ becomes equal to $2\theta + b_\beta\tau\sqrt{n}$ in the limit as $n \to \infty$, which is consistent with the route distance used in the previous deterministic model.

References

Ansari S, Başdere M, Li X, Ouyang Y, Smilowitz K (2018) Advancements in continuous approximation models for logistics and transportation systems: 1996–2016. *Transportation Res. Part B: Methodological* 107:229–252.

Anupindi R, Tayur S (1998) Managing stochastic multiproduct systems: Model, measures, and analysis. *Oper. Res.* 46(3-Supplement-3):S98–S111.

Azi N, Gendreau M, Potvin JY (2012) A dynamic vehicle routing problem with multiple delivery routes. *Ann. Oper. Res.* 199:103–112.

Bassett B, Gautam N (2019) The effect of adding stops to a delivery route. *2019 Inst. Indust. Systems Engineers Annual Conf. Expo* (Institute of Industrial and Systems Engineers, Peachtree Corners, GA), 529–534.

Beckwith S (2017) The Amazon effect: No longer a phantom menace. *Inbound Logist.* (November), https://www.inboundlogistics.com/wp-content/uploads/IL_Digital_November2017.pdf.

Bookbinder JH, Cai Q, He QM (2011) Shipment consolidation by private carrier: The discrete time and discrete quantity case. *Stochastic Models* 27(4):664–686.

Boyer KK, Prud'homme AM, Chung W (2009) The last mile challenge: Evaluating the effects of customer density and delivery window patterns. *J. Bus. Logist.* 30(1):185–201.

Burns L, Hall R, Blumenfeld D, Daganzo C (1985) Distribution strategies that minimize transportation and inventory costs. *Oper. Res.* 33(3):469–490.

Buzacott JA, Shanthikumar JG (1993) *Stochastic Models of Manufacturing Systems*, vol. 4 (Prentice Hall, Englewood Cliffs, NJ).

Cai Q, He QM, Bookbinder JH (2014) A tree-structured Markovian model of the shipment consolidation process. *Stochastic Models* 30(4):521–553.

Cattaruzza D, Absi N, Feillet D, González-Feliu J (2017) Vehicle routing problems for city logistics. *EURO J. Transportation Logist.* 6(1):51–79.

Cavdar B, Sokol J (2015) A distribution-free TSP tour length estimation model for random graphs. *Eur. J. Oper. Res.* 243(2):588–598.

Ceder A, Yim Y (2003) Integrated smart feeder/shuttle bus service. Working paper, California Partners for Advanced Transportation Technology, Institute of Transportation Studies, University of California at Berkeley, Berkeley, CA.

Çetinkaya S, Bookbinder JH (2003) Stochastic models for the dispatch of consolidated shipments. *Transportation Res. Part B: Methodological* 37(8):747–768.

Çeven E, Gue KR (2017) Optimal wave release times for order fulfillment systems with deadlines. *Transportation Sci.* 51(1):52–66.

Chen C, Pan S (2016) Using the crowd of taxis to last mile delivery in e-commerce: A methodological research. Borangiu T, Trentesaux D, Thomas A, McFarlane D, eds. *Service Orientation in Holonic and Multi-Agent Manufacturing* (Springer, Cham, Switzerland), 61–70.

Chen H, Wan YW (2003) The optimal admission and dispatching control policy of a two-terminal transportation system. *IIE Trans.* 35(9):895–906.

Chew EP, Tang LC (1999) Travel time analysis for general item location assignment in a rectangular warehouse. *Eur. J. Oper. Res.* 112(3):582–597.

Choi Y, Schonfeld PM (2022) Review of length approximations for tours with few stops. *Transportation Res. Rec.* 2676(3):201–213.

Coelho LC, Cordeau JF, Laporte G (2013) Thirty years of inventory routing. *Transportation Sci.* 48(1):1–19.

Curry GL, Feldman RM (2010) *Manufacturing Systems Modeling and Analysis*, 2nd ed. (Springer, Heidelberg, Germany).

Daganzo CF (1984) The distance traveled to visit *N* points with a maximum of *C* stops per vehicle: An analytic model and an application. *Transportation Sci.* 18(4):331–350.

Daganzo CF (2005) *Logistics Systems Analysis* (Springer, Berlin).

De Maio A, Violi A, Laganà D, Beraldi P (2018) A freight adviser for a delivery logistics service e-marketplace. Daniele P, Scrimali L, eds. *New Trends in Emerging Complex Real Life Problems* (Springer, Cham, Switzerland), 219–226.

Dellaert N (2012) *Production to Order: Models and Rules for Production Planning*, vol. 333 (Springer-Verlag, Berlin).

Du T, Wang F, Lu PY (2007) A real-time vehicle-dispatching system for consolidating milk runs. *Transportation Res. Part E: Logist. Transportation Rev.* 43(5):565–577.

Duin Jv, Kneyber J (2004) Toward a matching system for the auction of transport orders. Taniguchi E, Thompson RG, eds. *Logistics Systems for Sustainable Cities* (Emerald Group Publishing Limited, Bingley, UK), 163–177.

Eisenstein DD (2005) Recovering cyclic schedules using dynamic produce-up-to policies. *Oper. Res.* 53(4):675–688.

Gendreau M, Guertin F, Potvin JY, Taillard E (1999) Parallel tabu search for real-time vehicle routing and dispatching. *Transportation Sci.* 33(4):381–390.

Ghiani G, Guerriero F, Laporte G, Musmanno R (2003) Real-time vehicle routing: Solution concepts, algorithms and parallel computing strategies. *Eur. J. Oper. Res.* 151(1):1–11.

Henchiri A, Bellalouna M, Khaznaji W (2014) A probabilistic traveling salesman problem: A survey. *Ann. Comput. Sci. Inform. Systems* 3:55–60.

Hopp WJ, Spearman ML (2011) *Factory Physics* (Waveland Press, Long Grove, IL).

Ichoua S, Gendreau M, Potvin JY (2000) Diversion issues in real-time vehicle dispatching. *Transportation Sci.* 34(4):426–438.

Jaillet P (1985) Probabilistic traveling salesman problems. Unpublished PhD thesis, Massachusetts Institute of Technology, Cambridge.

Klapp M, Erera A, Toriello A (2018a) The dynamic dispatch waves problem for same-day delivery. *Eur. J. Oper. Res.* 271:519–534.

Klapp M, Erera A, Toriello A (2018b) The one-dimensional dynamic dispatch waves problem. *Transportation Sci.* 52(2):402–415.

Kou S, Golden B, Poikonen S (2022) Optimal TSP tour length estimation using standard deviation as a predictor. *Comput. Oper. Res.* 148:105993.

Kou S, Golden B, Poikonen S (2023) Estimating optimal objective values for the TSP, VRP, and other combinatorial problems using randomization. *Internat. Trans. Oper. Res.*, ePub ahead of print January 17.

Krumke SO, Rambau J, Torres LM (2002) Real-time dispatching of guided and unguided automobile service units with soft time windows. Möhring R, Raman R, eds. *Algorithms—ESA2022*. Lecture Notes in Computer Science, vol. 2461 (Springer, Berlin), 637–648.

Le-Duc T, De Koster RM (2007) Travel time estimation and order batching in a 2-block warehouse. *Eur. J. Oper. Res.* 176(1):374–388.

Lee HS, Srinivasan MM (1990) The shuttle dispatch problem with compound Poisson arrivals: Controls at two terminals. *Queueing Systems* 6(1):207–221.

Lin Z, Dong J, Liu C, Greene D (2012) Estimation of energy use by plug-in hybrid electric vehicles: Validating gamma distribution for representing random daily driving distance. *Transportation Res. Rec.* 2287(1):37–43.

Liu S, He L, Max Shen ZJ (2021) On-time last-mile delivery: Order assignment with travel-time predictors. *Management Sci.* 67(7):4095–4119.

Lynch C (2018) The time has come. *DC Velocity* 16(1):39.

Markowitz DM, Wein LM (2001) Heavy traffic analysis of dynamic cyclic policies: A unified treatment of the single machine scheduling problem. *Oper. Res.* 49(2):246–270.

Markowitz DM, Reiman MI, Wein LM (2000) The stochastic economic lot scheduling problem: Heavy traffic analysis of dynamic cyclic policies. *Oper. Res.* 48(1):136–154.

Minkoff AS (1985) Real-time dispatching of delivery vehicles. Unpublished PhD thesis, Massachusetts Institute of Technology, Cambridge.

Minkoff AS (1993) A Markov decision model and decomposition heuristic for dynamic vehicle dispatching. *Oper. Res.* 41(1):77–90.

Mutlu F, Çetinkaya S, Bookbinder JH (2010) An analytical model for computing the optimal time-and-quantity-based policy for consolidated shipments. *IIE Trans.* 42(5):367–377.

Nahmias S, Lennon Olsen T (2020) *Production and Operations Analysis*, 8th ed. (Waveland Press, Long Grove, IL).

Nandiraju S, Regan A (2008) Freight transportation electronic marketplaces: A survey of the industry and exploration of important research issues. Technical report, University of California Transportation Center, Berkeley, CA.

Pardo EG, Gil-Borrás S, Alonso-Ayuso A, Duarte A (2023) Order batching problems: Taxonomy and literature review. *Eur. J. Oper. Res.* 313(1):1–24.

Polus A (1979) A study of travel time and reliability on arterial routes. *Transportation* 8(2):141–151.

Powell WB (2007) Real-time dispatching for truckload motor carriers. *Logistics Engineering Handbook* (CRC Press, Boca Raton, FL), 323–342.

Satir B, Erenay FS, Bookbinder JH (2018) Shipment consolidation with two demand classes: Rationing the dispatch capacity. *Eur. J. Oper. Res.* 270(1):171–184.

Stroh AM, Erera AL, Toriello A (2022) Tactical design of same-day delivery systems. *Management Sci.* 68(5):3444–3463.

Tian J, Fu R, Yuan J (2007) On-line scheduling with delivery time on a single batch machine. *Theoret. Comput. Sci.* 374(1–3):49–57.

Ulmer MW, Thomas BW, Mattfeld DC (2019) Preemptive depot returns for dynamic same-day delivery. *EURO J. Transportation Logist.* 8(4):327–361.

Van Heeswijk WJ, Mes MR, Schutten JM (2019) The delivery dispatching problem with time windows for urban consolidation centers. *Transportation Sci.* 53(1):203–221.

Voccia S, Campbell A, Thomas B (2019) The same-day delivery problem for online purchases. *Transportation Sci.* 53(1):167–184.

Wagner M, Smits SR (2004) A local search algorithm for the optimization of the stochastic economic lot scheduling problem. *Internat. J. Production Econom.* 90(3):391–402.

Wei B, Çetinkaya S, Cline DB (2022) Analytical results on the service performance of stochastic clearing systems. *Probab. Engrg. Inform. Sci.* 36(2):217–236.

Winands EM, Adan IJ, van Houtum GJ (2011) The stochastic economic lot scheduling problem: A survey. *Eur. J. Oper. Res.* 210(1):1–9.

Zhang G, Cai X, Wong C (2001) On-line algorithms for minimizing makespan on batch processing machines. *Naval Res. Logist.* 48(3):241–258.