Tactical Design of Same-Day Delivery Systems

Alexander M. Stroh Alan L. Erera Alejandro Toriello

H. Milton Stewart School of Industrial and Systems Engineering
Georgia Institute of Technology
Atlanta, Georgia 30332
alexmstroh *at* gatech *dot* edu, {aerera, atoriello} *at* isye *dot* gatech *dot* edu

December 15, 2020

Abstract

We study tactical models for the design of same-day delivery (SDD) systems. Same-day fulfillment in e-commerce has seen substantial growth in recent years, and the underlying management of such services is complex. While the literature includes operational models to study SDD, they tend to be detailed, complex, and computationally difficult to solve, and thus may not provide any insight into tactical SDD design variables and their impact on the average performance of the system. We propose a simplified vehicle dispatching model that captures the "average" behavior of an SDD system from a single stocking location by utilizing continuous approximation techniques. We analyze the structure of optimal vehicle dispatching policies given our model for two important instance families, the single-vehicle case and the case in which the delivery fleet is large, and develop techniques to find these policies that require only simple computations. We also leverage these results to analyze the case of a finite fleet, proposing a heuristic policy with a worst-case approximation guarantee. We then demonstrate with several example problem settings how this model and these policies can help answer various tactical design questions, including how to select a fleet size, determine an order cutoff time, and combine SDD and overnight order delivery operations. We validate model predictions empirically against a detailed operational model in a computational case study using geographic and census data for the northeastern metro Atlanta region, and we demonstrate that our model predicts the average number of orders served and dispatch time to within 1%.

1 Introduction

Total annual retail sales in the United States grew by an estimated 2.76% from 2015 to 2016, in part via electronic shopping, which increased by 12.40% [6]. Within the growing space of e-commerce, same-day delivery (SDD) services are commonly offered by large retailers and logistics providers. A survey of over 500 North American retailers found that 51% claimed to provide SDD fulfillment options in 2017, up from the 16% reported in 2016 [11]. Amazon, one of the current SDD industry leaders, began offering an SDD fulfillment option in October 2009 across seven major U.S. cities [16]. By April 2016, Amazon offered the service to over 1,000 cities and towns, a figure which rose to over 8,000 by November 2018 [17, 18]. These statistics highlight recent demand increases in the e-commerce space, as well as the adoption of SDD systems by many retailers.

Managing an SDD system is potentially problematic for retailers with already thin profit margins. SDD systems inherit and exacerbate many of the issues faced by more traditional two-day or next-day last mile logistics systems, including tight deadlines, low order volumes, and a high level of order variability and dynamism; in general, the uncertainty increases and the time to react decreases [22, 23]. Therefore, dispatching and routing orders may be costly and inefficient if not planned carefully.

Like more traditional e-retail delivery services, SDD requires two core logistics processes: order management at the stocking location, including receiving, picking, and packing; and order distribution from the stocking location to customer delivery addresses. We focus here on the second of these processes. Order distribution requires operational decisions, such as when to dispatch a delivery vehicle (timing), and which subset of awaiting customers it will serve (composition). There are clear trade-offs between the timing and composition of SDD dispatches. In some cases, it may be best for a delivery vehicle to wait as long as possible at the stocking location, allowing the accumulation of orders and greater routing efficiency upon dispatch. Alternatively, shorter, more time-inefficient trips could be made in order to reduce the workload within the system, leaving more flexibility to serve orders later in the service day. Such decisions are made numerous times, across a fleet of vehicles, during each service day. Even when the orders to be loaded onto each vehicle for a dispatch are known, deciding the sequence in which to visit delivery locations is a traveling salesman problem (TSP) with possible side constraints; the problem of order assignment and routing in e-commerce has only recently attracted attention from the research community, e.g. [25]. Additionally for SDD, retailers often constrain themselves to serve all SDD demand in a given service day, and thus restrict the latest possible time SDD orders can be placed [12, 19]. These order cutoff times can be static or determined dynamically.

Over the past few years, the logistics research community has proposed and studied operational policies for SDD distribution systems using a variety of models and assumptions, e.g. [21, 22, 23, 33, 36]. The models considered in the literature to date typically assume a fixed SDD system design, including service area, delivery vehicle fleet size, service time window, etc., and then perform a detailed analysis, optimization and/or simulation of operating policies.

In contrast, the logistics research community has not focused its analysis on the tactical design decisions important for SDD distribution: How large should the SDD delivery vehicle fleet be? How late in the day should SDD service be offered to customers? How large should the service area be? To our knowledge, no papers in the literature address these and other important questions, and our goal is to offer a first attempt. While detailed operational models can in principle offer some insights about such tactical decisions, their granularity implies significant complexity, which in turn renders tactical analysis difficult and less transparent – the models have “too many moving parts”. One goal of this paper is to develop simplified models of operational decisions while maintaining fidelity at the aggregate level; we propose models that do not attempt to capture each order realization and operational decision, but rather to capture the system behavior “on average” so that we may approximate the impact of various design choices on day-to-day operations.

We develop a distribution modeling approach for a single SDD stocking location or dispatch facility, where orders are packed for last-mile delivery and dispatched on delivery vehicles. As in most e-retail settings, we assume orders are customer-specific and cannot be packed or dispatched preemptively, but rather only after they are placed. We also assume a common delivery deadline, e.g. the end of the business day, rather than order-specific deadlines more common in food delivery services [29]. Since SDD systems face tight delivery deadlines and comparatively low order volume, time (not vehicle capacity) tends to be the limiting resource and one of the primary constraints in our models. We initially assume uncapacitated vehicles, and later show that our results extend to the capacitated case with only slight modifications.

To build a simplified SDD dispatch model that still accurately captures system performance, we use a continuous approximation approach in which the expected durations of vehicle routing tours are approximated using a concave, increasing function of the number of orders served. The use of such approximations is well established in logistics [15], with some canonical results dating back several decades [3, 13, 27]. When order locations are randomly distributed in the service region according to a continuous distribution, continuous approximations are known to be quite accurate. Such approximations have recently been successfully applied in a last-mile operational context [34], and can also be calibrated with empirical observations (see, e.g., [20]). We provide our own computational validation of the approximation model and dispatching policies we develop, and we show that they are remarkably accurate when compared to much more detailed operational models.

Furthermore, although our model is motivated by SDD, our main results rely only on the concavity and monotonicity of the expected dispatch time required to serve a number of orders. Our model could therefore also have applications in other areas where expected processing time is concave and increasing, including batched queueing and warehousing systems, see e.g. [10].

1.1 Contributions

We formulate an SDD dispatching model based on continuous approximations with a single depot and its fleet serving SDD orders in a specified service region. We study the structure of optimal dispatch policies for such models, and make the following specific contributions.

(1) We propose a simple model for SDD dispatching that captures aggregate SDD system behavior by leveraging the elegant structure of continuous approximations. To our knowledge, this is the first such use of this methodology in SDD applications.

(2) We use the dispatching model to analyze two important operational cases for SDD fleets. The first case is when a single vehicle is assigned to a service area and is dispatched multiple times during the operating day, and the second case is when the fleet is large enough that each vehicle is dispatched once per day. We characterize the structure of optimal dispatching policies for these two cases using our model, and show that the optimal policies can be determined using very simple computational techniques, such as finding the roots of equations with single unknowns. Although the case in which multiple vehicles must make multiple dispatches is significantly more difficult to optimize, we propose a heuristic policy for this case with a worst-case performance guarantee.

(3) We use the simple dispatching approximation model and the optimal policies that result to answer various tactical system design questions, including fleet sizing, length of service window, and whether SDD orders should be combined with overnight orders. In all cases, our conclusions rely on simple and transparent analytical properties. The model predictions are validated in a computational study against a detailed operational model using realistic data.

The remainder of the paper is organized as follows. Section 1 concludes with a literature review. In Section 2 we formulate a continuous approximation model of vehicle dispatch operations from a single stocking location and justify our model assumptions. We then describe optimal dispatch policies for specific instances of the proposed model in Section 3. Section 4 provides a managerial analysis of tactical SDD system design using our model and its solutions. We detail a realistic computational experiment using the model and policies in Section 5, and conclude in Section 6. An appendix contains proofs omitted from the main body.

1.2 Literature Review

SDD models can be classified within the rich family of vehicle routing problems (VRPs). The defining features of an SDD model include stochastic order arrivals, order cutoff times and/or a delivery deadline, and perhaps most salient, the overlap in time of dispatching and order arrivals. Examples of model objectives are maximizing expected orders served in a service day, minimizing penalties from undelivered orders, and/or minimizing total routing distance or time given that most or all orders are served. For these reasons, we reference the VRP with probabilistic customer arrivals as studied in [1, 7, 35]. Additionally, dynamic vehicle routing problems such as [4, 26] broadly encompass SDD modeling.

We now survey some operational SDD models from the literature. One such problem is the dynamic dispatch waves problem (DDWP) [22, 23]. The DDWP discretizes the dispatch decision epochs, or "waves", for an operator managing an SDD vehicle fleet. Customer orders arrive according to a known stochastic process and must be served by the end of the service day, or the operator will receive an order-based penalty. Additionally, all delivery vehicles are constrained to return back to the depot before the end of the service day. The objective of the DDWP is to minimize the sum of expected routing cost and penalty cost. In [23] a deterministic variant of the DDWP is solved, where order arrival locations and times are known exactly, over a 1-dimensional service region using the optimal policy structure found in a dynamic programming formulation. In [22] the same authors model the DDWP in 2-dimensions using an integer-programming approach to solve a deterministic variant. Using these deterministic solutions, the authors compute *a priori* dispatch policies for stochastic variants. These policies are further expanded to dynamic policies in their respective papers. In [23], optimal dispatch policies for an SDD variant (one-dimensional service region, one vehicle) were found to have the property that once the first dispatch occurred, the vehicle never waited at the depot again. Additionally, the durations of successive dispatches are decreasing.

Another SDD model found in literature is the same-day delivery problem for online purchases (SDDP) [36]. The authors provide a general framework for SDD modeling, using a fleet of delivery vehicles of known size, a fixed cutoff time for SDD orders, a known arrival rate and distribution of orders, and a service time and a delivery time window on each order. Like the DDWP, all vehicles operate from a single depot. Unlike the DDWP, the objective of the SDDP is to maximize the expected number of SDD requests that are fulfilled in a service day. A model of the SDDP as a Markov decision process (MDP) is proposed, and dispatch policies are found via a sample-scenario approach with orienteering subproblems. The authors discuss delaying delivery vehicles at the depot for as long as possible without violating delivery time-window constraints or altering vehicle return times. Such properties were shown to exist in optimal dispatching solutions, allowing restriction of the search space.

The DDWP and the SDDP, as well as other works in the literature [30, 31, 32, 33], tend to study the daily operations of SDD logistics, while assuming implicit or explicit knowledge of tactical system design features. It is possible to use these more complicated models to gain tactical-level insight. In [30] the author performs an analysis of the relationship between fleet size and delivery capacity of the system. The authors in [36] observe how the number of fulfilled orders can increase with fleet size by re-solving their operational SDD model multiple times. Such procedures can be useful for determining managerial decisions. However, they require repeatedly solving complex models, often with heuristic methods that do not guarantee optimal solutions; in contrast, our approach will be to exactly optimize a simplified approximation model.

We use continuous approximations to model an SDD system. Specifically, we assume orders arrive at a constant rate, and allow a non-integer number of orders to be served in a dispatch. We approximate the expected time of a dispatch's duration using a non-decreasing, concave function of the number of orders served; a typical example would be the square root of the number of orders, scaled by an appropriate constant [3]. For a recent survey on continuous approximation models in freight logistics, see [15]. The use of such approximations in the field goes back to the BHH theorem [3], a formula for the expected length of a TSP tour as a function of the number of stops visited when locations are drawn from a continuous distribution over the service region. [13, 27] then expanded upon this approximation in an analysis of vehicle routing problems with specified dispatch depots for logistics distribution and collection problems, and studied how different zone shapes affect tour lengths and how to select best zone shapes. The work in [20] considers similar approximation ideas for the Held-Karp TSP bound, and [2] calculates empirical constants for TSP length as a function of the number of stops in a tour; see also [37]. Although our study is the first to apply continuous approximations in SDD, there are other applications of these techniques in urban logistics. For example, the work in [14] considers the efficiency of urban commercial vehicles using continuous approximations, and the techniques are used to study drones in last-mile delivery in [9]. Furthermore, a continuous approximation approach is used in [8] to partition a service region for vehicle routing. Operational models for urban last-mile delivery are considered in [34] and continuous approximations are deployed within an approximate dynamic programming framework for optimization. Continuous approximations are frequently applied in facility location; see [24] for a recent example with applications in SDD and the last mile.

2 Model Formulation

We consider an SDD system in which a single depot serves as a stocking location, and its vehicle fleet serves all orders placed from a defined service region. Orders accumulate throughout the service region over the course of the day, and the dispatcher must ensure that all orders are served by the end of the service day at minimum cost. We next formally define our problem by describing the relevant notation and model elements.

Service Day: The first time any vehicle can leave the depot is time $t = 0$, and the end of the service day is $t = T$. For convenience, we refer to the service day as having T units of time.

Customer Orders and Geography: Demand for SDD in the service region continuously accumulates over time at a constant rate of λ orders per time unit. This demand is served by a fleet of m vehicles, each departing from the single depot. For convenience, we assume without loss of generality that $\lambda = 1$ and all other parameters are appropriately scaled.

Order Cutoff Time: Customer orders become ready for dispatch starting at time $t = 0$ and continue until time $t = N < T$ in the service day. The service day begins with no orders requiring delivery, and thus the total number of orders that accumulate over the day is $\lambda N = N$. Depending on the context, we may refer to N as the order cutoff time or the number of orders to serve. We also discuss the problem extension when an initial set of orders is ready at the start of the service day.

Vehicle Restrictions: All vehicles must return to the depot by time $t = T$. We do not initially constrain the capacity of any vehicle, nor do we restrict any vehicle to carry an integer number of orders. Vehicles may be dispatched more than once during the service day. We later discuss the model extension with capacitated vehicles.

Dispatch Time Function: The time it takes for a vehicle to serve $n \in \mathbb{R}_{\geq 0}$ orders and return to the depot is given by a function $f : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ that is concave and increasing. This property indicates that serving more orders should take more time, but that there will be a gain in marginal time efficiencies when consolidating orders. An example dispatch time function is of the form $f(n) = a + bn + c\sqrt{n}$. This function includes a constant setup time at the depot, a service time per order, and a BHH routing time [3] between orders; [20] showed computationally that these approximations work well in practice even for small order numbers, assuming order locations are independently drawn from a continuous distribution over the service region.

The objective of our model is to choose a set of feasible dispatches that serves all of the orders while minimizing total dispatch time incurred by all vehicles. We define the d-th dispatch as a tuple (t_d, q_d, i_d), where t_d indicates the time when vehicle i_d leaves the depot with an order quantity, q_d. We assume without loss of generality that dispatches are ordered by time of dispatch, then if necessary by vehicle index. A set of dispatches $\{(t_d, q_d, i_d)\}_{d=1}^{D}$ is feasible for our model if the following conditions are satisfied:

$$\sum_{d=1}^{D} q_d = N, \tag{1a}$$

$$q_d \geq 0 \quad \forall d, \tag{1b}$$

$$t_d + f(q_d) \leq T \quad \forall d, \tag{1c}$$

$$t_d + f(q_d) \leq t_\delta \quad \forall d, \delta \text{ s.t. } i_d = i_\delta,\ d < \delta \tag{1d}$$

$$t_d \geq 0 \quad \forall d, \tag{1e}$$

$$\sum_{\delta=1}^{d} q_\delta \leq t_d \quad \forall d \tag{1f}$$

Our problem is to choose dispatches that minimize $\sum_{d=1}^{D} f(q_d)$, subject to (1a)-(1f), over all $D \geq 1$. Constraints (1a)-(1b) guarantee that all dispatches serve a non-negative order number, and that these sum to the total number of orders. Constraint (1c) requires all vehicles to return to the depot by the end of the service day. Constraint (1d) guarantees that any vehicle performs one dispatch at a time. Constraint (1e) ensures the vehicles dispatch only after the service day begins. Finally, constraint (1f) guarantees that orders are served only after they realize.

3 Optimal Policies

We initially focus on two important families of instances of our SDD model, the many-vehicle case and the single-vehicle case. In the former, we assume any number of vehicles can be added to the delivery fleet at negligible cost; this situation applies, for example, when we can allocate vehicles for SDD from other resources. In the latter, we focus on the simplest case in which the fleet is constrained, as this case is already of significant interest operationally [22, 23]. For the general case with a finite fleet greater than one, we leverage these results to construct a heuristic dispatching policy that combines the policies used at the two extremes. Table 1 summarizes results in this section, including the type of result we obtain and additional conditions required for the result to hold.

Section	Fleet	Result	Additional Conditions?
§3.1	unlimited	optimal policy	no
§3.2	single vehicle	optimal policy	sufficient processing speed sufficient gap time minimum dispatch size
§3.3	finite	heuristic policy with approximation guarantee	single-vehicle conditions $f(n) = bn + c\sqrt{n}$

Table 1: Summary of Section 3 results.

3.1 Many Vehicles

We first assume the fleet consists of as many vehicles as we like. Consider the following dispatch policy, and the accompanying theorem. The proof is deferred to the appendix.

Many-Vehicle Policy (MVP) Starting at time $t = 0$, dispatch a delivery vehicle at the moment when it can take all of the realized orders waiting at the depot and return at exactly time $t = T$. Repeat this process until only a single vehicle is needed to deliver all remaining orders and return to the depot before the end of the service day. Dispatch this last vehicle at time $t = N$.

Theorem 1. *The MVP is an optimal dispatch policy for the many-vehicle case. Furthermore, if the number of vehicles used by MVP is m^*, the total dispatch time used by this policy provides a lower bound for the model objective with fleet size $m < m^*$.*

The optimal times given by the policy are easy to solve for in practice, since all that is required is to solve equations of the form, $t + f(t) = C$, for different values of C. If more than one vehicle is required, then $t_1 + f(t_1) = T$, and if more than two vehicles are required then $t_2 + f(t_2 - t_1) = T$, and so on. For problems with realistic parameters, computational results show that the MVP will often require a reasonably sized fleet. Figure 1 depicts an example MVP dispatch plan with four vehicles. Each curved arc corresponds to a vehicle dispatch, while the preceding horizontal straight line represents the time in which that dispatch's orders accumulate while the vehicle waits. Line styles are alternated for visual clarity.

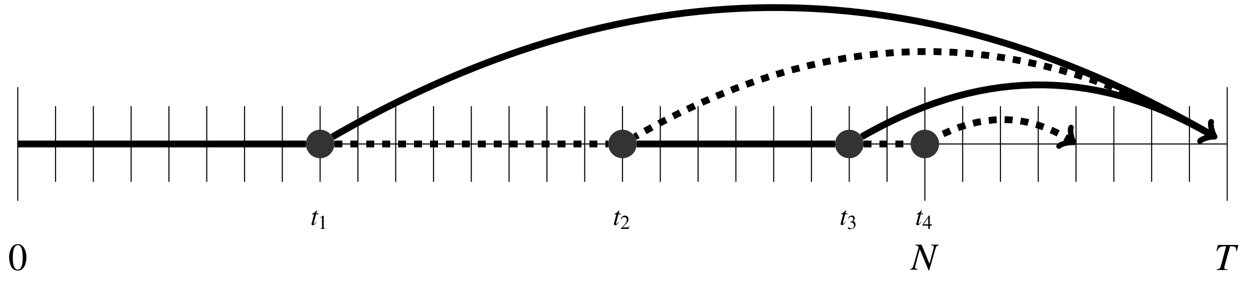

Figure 1: Visual representation of an MVP requiring four vehicles.

Example 2. A retailer provides SDD service for an 8 mile by 8 mile service region, with an average of 75 orders placed over a 10-hour cutoff time. The retailer operates over a 12-hour service day and has an unrestricted fleet size. We scale time to 8 minutes per time unit, and the model parameters are set to $N = 75, T = 90$. Additionally, suppose that the dispatch time function is $f(n) = 2.15\sqrt{n} + .13n$, roughly equivalent to a routing time approximation (Manhattan distances [20], with vehicles traveling at 25 miles per hour), plus a service time of 1 minute per order. The MVP returns the optimal solution,

$$\begin{aligned} t_1 &= 64.38, & q_1 &= 64.38, & i_1 &= 1, \\ t_2 &= 75, & q_2 &= 10.62, & i_2 &= 2, \end{aligned}$$

with 272.06 total minutes of dispatch time.

3.2 One Vehicle

Now assume the fleet consists of a single delivery vehicle. If $N + f(N) \leq T$, this vehicle can (optimally) wait until time $t = N$ to dispatch once with all N orders. More generally, we use the following lemma in our analysis. Qualitatively, this lemma suggest that given a fixed amount of orders to be served, it would be better to split the orders between two dispatches as unevenly as possible to maximize routing efficiencies. This result follows directly from the concavity of f.

Lemma 3. *The optimal solution of*

$$\min_{a \leq y \leq b} \{f(y) + f(Y - y)\}$$

is $y^* = a$ if $a \leq Y - b$, and $y^* = b$ otherwise, for any $0 \leq a \leq b \leq Y$.

Before stating our main results for a single vehicle, we must address a pathology arising from Lemma 3. Consider an instance with $N = 9, T = 11.99, f(n) = \sqrt{n}$; clearly, a single dispatch is infeasible, since the vehicle would return to the depot too late by 0.01 time units. It can be shown that any feasible two-dispatch policy satisfies $0.06 \leq q_1 \leq 8.25$ and $q_2 = 9 - q_1$. By the lemma, the best solution using two dispatches is given by $q_1^* = 0.06$. Furthermore, this policy is in fact optimal for a single vehicle. There are two characteristics of this optimal policy worth noting. First, the first time of dispatch can be adjusted to any time in the range of $q_1^* \le t_1^* \le N - f(q_1^*)$, while $t_2^* = N$. Second, the first dispatch size is very small, which is likely to be unreasonable since the square root approximation of routing time tends to be inaccurate for small numbers.

This example shows that the model requires additional assumptions to produce meaningful answers. We now introduce three additional conditions that address these concerns.

Sufficient processing speed: There exists $q_{\min} < \infty$ such that $f(x) \le x/\lambda = x$, for all $x \ge q_{\min}$.

Sufficient gap time: The parameters $T, N, q_{\min}$ satisfy $T - N \ge f(2q_{\min})$.

Minimum dispatch size: Any feasible solution $\{(t_d, q_d, i_d)\}_{d=1}^{D}$ satisfies $q_d \ge q_{\min}$ for all $d < D$.

The first condition ensures that the system can "keep up" with orders; that is, whenever the delivery vehicle is dispatched with a large enough quantity, it is guaranteed to arrive back at the depot to find a lesser number of unserved orders. The second condition ensures that there is enough time between the last order arrival and the end of service day. Finally, from a modeling perspective, a constraint on the minimum size of a dispatch is justifiable economically. The last dispatch is not subject to this constraint, since it must serve all remaining orders, regardless of their number. The sufficient processing speed and sufficient gap time conditions imply feasibility, including satisfying the minimum dispatch size condition; the next lemma formalizes this argument and is proved in the appendix.

Lemma 4. *An SDD problem instance with a single vehicle and parameters satisfying the sufficient processing speed and sufficient gap time conditions has a feasible solution satisfying the minimum dispatch size constraints.*

We next state our main results on optimal dispatch policies for a single vehicle, which are proved in the appendix.

Theorem 5. *A single-vehicle SDD instance satisfying the sufficient processing speed and sufficient gap time conditions, and with the additional constraints imposed by the minimum dispatch size condition, has an optimal dispatch policy such that*

(C1) each dispatch takes all available unserved orders at the depot at the time of dispatch,

(C2) after the first dispatch, the vehicle never waits at the depot again, and

(C3) if the vehicle is dispatched more than once, the last dispatch arrives back at the depot exactly at time $t = T$.

Theorem 5 implies that there exists an optimal policy $\{(t_d^*, q_d^*)\}_{d=1}^{D^*}$ that can be described completely by t_1^*. By (C1), we have $q_1^* = \lambda t_1^* = t_1^*$, and then (C2) yields $t_2^* = t_1^* + f(q_1^*)$. Applying this reasoning recursively, $q_2^* = \lambda(t_2^* - t_1^*) = t_2^* - t_1^*$ and so on, continuing until the last dispatch covers all the remaining orders, leaving at or after time $t = N$. In contrast, the example after Lemma 3 shows that when an instance does not satisfy the three conditions, the optimal solution need not have the structure outlined in the theorem. Specifically, the example's optimal solution does not satisfy at least one of (C1) or (C2).

Algorithmically, we apply Theorem 5 to restrict our search for an optimal policy to feasible policies that satisfy (C1) and (C2). Among such policies, we search for one that satisfies (C3) and has a first dispatch time $t_1 = \alpha$ as large as possible; (C2) and (C3) guarantee that maximizing α is equivalent to minimizing the vehicle's total dispatch time.

Formally, for positive integers δ, d, let

$$f^1 := f, \qquad f^{\delta}(\cdot) := f(f^{\delta-1}(\cdot)), \quad \delta \geq 2, \qquad h_d(\alpha) := \alpha + \sum_{\delta=1}^{d-1} f^{\delta}(\alpha).$$

Intuitively, f^{δ} is the δ-th composition of f, and we use f^{δ} to define $h_d(\alpha)$, the number of orders served by the first d dispatches for a policy satisfying (C1) and (C2) with first dispatch at time α. If a feasible policy satisfying (C1) and (C2) has a first dispatch at time α and a total of $D \geq 2$ dispatches, it serves $h_{D-1}(\alpha)$ orders with its first $D-1$ dispatches, while the last dispatch serves the remaining $N - h_{D-1}(\alpha)$ orders, where $h_{D-1}(\alpha) \leq N \leq h_{D-1}(\alpha) + f^{D-1}(\alpha) = h_D(\alpha)$. This implies natural bounds on the first dispatch time when using exactly D dispatches: If we define $\alpha_1 = N$ and α_2 such that $\alpha_2 + f(\alpha_2) = N$, then all feasible dispatch policies satisfying (C1), (C2) and using exactly two dispatches have a time of first dispatch in $[\alpha_2, \alpha_1)$. Generalizing, if we define α_D as the unique α satisfying $h_D(\alpha) = N$, all feasible policies that satisfy (C1), (C2) and use exactly D dispatches have a time of first dispatch in $[\alpha_D, \alpha_{D-1})$.

Because the α_D values are decreasing in D and we want to maximize α, we can iteratively search for this time of first dispatch by fixing the policy's total number of dispatches D, starting with $D = 1$. For each D, we attempt to find the largest $\alpha \in [\alpha_D, \alpha_{D-1})$ satisfying $h_D(\alpha) + f(N - h_{D-1}(\alpha)) = T$. If such an α exists, it must correspond to the first dispatch time of an optimal policy satisfying (C1), (C2) and (C3); if no such α exists, we increase D by one and repeat. Algorithm 1 states this procedure in pseudo-code, and Corollary 6 is proved in the appendix.

Corollary 6. *Suppose a single-vehicle SDD instance satisfies the sufficient processing speed and sufficient gap time conditions, and we impose the additional constraints of the minimum dispatch size condition. Algorithm 1 determines the time of first dispatch for an optimal policy satisfying (C1), (C2), and (C3) from Theorem 5.*

Algorithm 1 Calculating the optimal time of first dispatch in the single-vehicle case.

```
$D \leftarrow 1$, $\alpha^* \leftarrow 0$
if $N + f(N) \leq T$ then
    $\alpha^* \leftarrow N$
else
    while $\alpha^* = 0$ do
        $D \leftarrow D + 1$
        $A \leftarrow \{\alpha \in [\alpha_D, \alpha_{D-1}) : h_D(\alpha) + f(N - h_{D-1}(\alpha)) = T\}$
        if $A \neq \emptyset$ then
            $\alpha^* \leftarrow \max_{\alpha \in A} \alpha$
        end if
    end while
end if
return $\alpha^*$
```

Example 7. Consider the same instance as in Example 2, that is, $N = 75, T = 90, f(n) = 2.15\sqrt{n} + .13n$. Now suppose the fleet has a single delivery vehicle, and $q_{\min} = 12$. This set of model parameters satisfy the minimal dispatch size, sufficient gap time, and sufficient processing speed conditions, so we can use Theorem 5 and our root finding algorithm to compute an optimal dispatch policy:

We first set $D \leftarrow 1$, and $\alpha_1 \leftarrow 75$. As $N + f(N) = 103.37 > 90 = T$, a single dispatch is insufficient and we must consider using two or more dispatches. For two dispatches, we calculate that $\alpha_2 = 52.58$, and thus consider all policies with two dispatches where $\alpha \in [52.58, 75)$. From here we wish to determine if there is an α in this range satisfying $\alpha + f(\alpha) + f(N-\alpha) = T$. Indeed, $\alpha = 54.65$ solves this expression, and the algorithm terminates with $\alpha^* = 54.65$ and $D^* = 2$. The calculated optimal policy is

$$\begin{aligned} t_1 &= 54.65, & q_1 &= 54.65, & i_1 &= 1, \\ t_2 &= 77.65, & q_2 &= 20.35, & i_2 &= 1, \end{aligned}$$

with a total dispatch time of 282.74 minutes; see Figure 2. By comparing this example to the previous one, we can see that by decreasing the fleet from two to one vehicles we would increase the total dispatch time by less than 4% (272.06 minutes to 282.74 minutes).

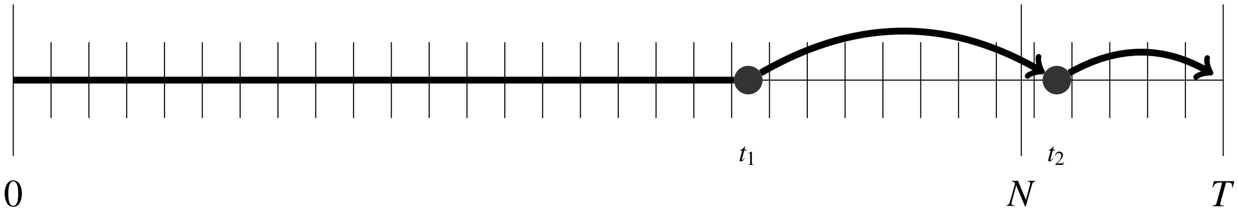

Figure 2: Visual representation of optimal dispatch policy for Example 7.

3.3 General Fleet Size

We now consider the more complex case in which the fleet is finite but greater than one, and thus many vehicles may need to be dispatched more than once. Unfortunately, it is no longer possible to show that optimal policies satisfy simple structural properties in this case. For example, consider a family of instances where the fleet has two vehicles, but three dispatches are required to serve the orders feasibly. Depending on the parameters T, N and f, it is possible to construct both instances where it is optimal for the vehicle dispatched first to return before T and to make a second dispatch while the second vehicle is used only once, but also instances where it is instead optimal for the vehicle dispatched second to make the additional dispatch.

Although the optimization of this case is significantly more complex, we can leverage our analysis of the previous cases to construct a heuristic policy, described next.

Hybrid Policy For a fleet with m vehicles, the first $m-1$ are dispatched according to the MVP. The final vehicle serves all remaining orders according to the single-vehicle policy computed with Algorithm 1.

The next result shows that this heuristic policy produces solutions within a worst-case factor of optimality for an important class of dispatch time functions. The proof is in the Appendix.

Theorem 8. *Assuming the sufficient processing speed and sufficient gap time conditions, the hybrid policy is feasible, including satisfying the minimum dispatch size condition constraints. Furthermore, suppose $f(n) = bn + c\sqrt{n}$, where $b \geq 0$ and $c \geq 0$. If the hybrid policy dispatches the last vehicle D_m times, its total dispatch time is guaranteed to be within a factor $\frac{m-1+D_m\sqrt{D_m}}{m-1+D_m}$ of the Many-Vehicle Policy's dispatch time, which is itself a lower bound for any m-vehicle solution.*

See Figure 3 below for an example of the hybrid policy compared to the MVP.

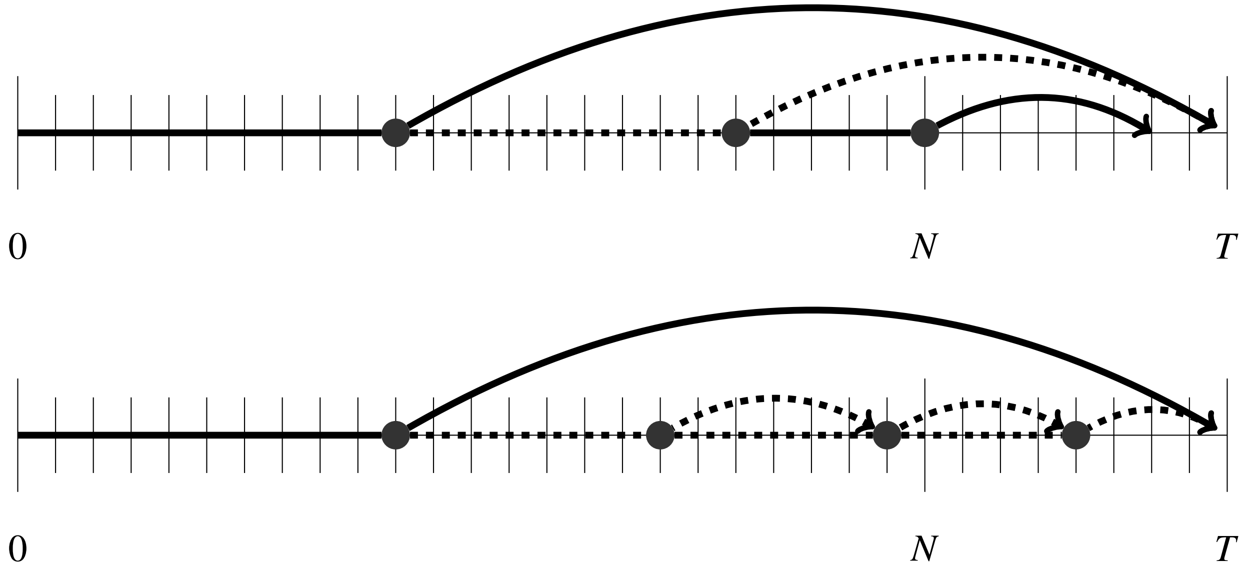

Figure 3: In this example, the MVP uses three vehicles. If only two are available, the hybrid policy stipulates that the first vehicle behaves as in the MVP, while the second performs an optimal single-vehicle dispatch policy on the remaining orders, which in this case requires three dispatches.

Example 9. Consider again the problem instance from Examples 2 and 7. The single-vehicle policy is a special case of the hybrid policy when $m = 1$ (and thus a single vehicle must make all dispatches even if the MVP uses multiple vehicles). Because the single-vehicle policy uses two dispatches in this instance, we conclude from Theorem 8 that its cost is at most a factor $\sqrt{2} \approx 1.41$ larger than the MVP cost. In this case, we know from direct calculation that this cost difference is much smaller, only around 4%.

The guarantee provided by Theorem 8 improves as m grows, since more of the hybrid policy's dispatches exactly mimic what the MVP does; for example, with $m = 2$ and $D_2 = 2$, the guarantee improves to $(1 + 2\sqrt{2})/3 \approx 1.28$.

4 Model Applications

The discussion in Examples 2 and 7 demonstrate how our model can be applied for tactical design, specifically in fleet sizing. We next discuss other potential uses of the model.

4.1 Serving the Entire Region versus Partitioning

Location analysis and customer assignment are important strategic and tactical questions in logistics, and continuous approximation models have been successfully applied for service region design, e.g. [8]. We can similarly ask in an SDD context whether partitioning the service region offers advantages over simply having every vehicle serve the entire region.

Consider a dispatch time function $f(n) = a + bn + c\sqrt{n}$ consisting of a setup time at the depot, a service time per order, and a BHH [3] routing time approximation, which depends on the size of the service region. Suppose we partition this region into m sub-regions of equal size, so that the demand arrival rate in each is $1/m$; each sub-region would then have a dispatch time function of the form $\hat{f}(n) = a + bn + c\sqrt{n/m}$, since the area the vehicle serves is scaled down by a factor of $1/m$. At time $t = N$, if a single vehicle can serve each sub-region with a single dispatch, the total dispatch time for all vehicles would be

$$m \times \hat{f}(N/m) = m\Big(a + b(N/m) + c\sqrt{N/m^2}\Big) = am + bN + c\sqrt{N};$$

the last two terms correspond exactly to the service and routing time a single vehicle would need to serve all N orders in a single dispatch. Therefore, if the MVP policy uses m^* vehicles and it is feasible to partition the region into m^* sub-regions and serve each with a single dispatch, partitioning is preferable. However, the number of required vehicles for a partitioning strategy with a single dispatch per vehicle may differ from m^* and be either larger or smaller.

Example 10. A retailer provides SDD service for an 8 mile by 8 mile service region, with an average of 75 orders placed over a 10-hour cutoff time. The retailer operates over an 11 hour and 20 minute service day. We scale time to 8 minutes per time unit, and the model parameters are set to $N = 75, T = 85$. Additionally, take the dispatch time function as $f(n) = 1.88 + .25n + 2.15\sqrt{n}$, roughly equivalent to a routing time approximation (Manhattan distances [20], with vehicles traveling at 25 miles per hour), plus a service time of 2 minutes per order and a setup time of 15 minutes. The MVP returns the optimal solution

$$\begin{aligned} t_1 &= 53.87, & q_1 &= 53.87, & i_1 &= 1,\\ t_2 &= 70.30, & q_2 &= 16.43, & i_2 &= 2,\\ t_3 &= 75, & q_3 &= 4.70, & i_3 &= 3, \end{aligned}$$

with 428.37 total minutes of dispatch time. In contrast, the minimum number of vehicles needed for a partition strategy as described above is five, with each delivering 15 orders in a total of 374.16 minutes. The manager must then decide whether saving 54.21 minutes per service day is worth an additional two vehicles in the SDD fleet. We can similarly use our single-vehicle policy to develop partitioning strategies with a single vehicle serving each sub-region but performing multiple dispatches.

4.2 Orders at the Start of the Service Day

Thus far, our model assumes no orders are ready for dispatch at the start of the service day. It may be that the SDD system is also required to serve some next-day or overnight orders. In the model, this translates to a number $N' \geq 0$ of orders that are ready at the start of the service day.

In the many-vehicle case, the optimal policy is similar to the MVP, with one modification. Let $Q = f^{-1}(T)$, i.e. Q is the unique number satisfying $f(Q) = T$ (the inverse exists and Q is unique because f is increasing); this number is implicitly a capacity on the number of orders a vehicle can carry during the service day to remain time-feasible. We can now define a generalized MVP, which returns an optimal policy when $N' \geq 0$ orders are ready at the depot at the start of the service day. The proof of this claim is found in the appendix.

Generalized MVP At time $t = 0$, dispatch as many vehicles as possible each carrying exactly Q orders. The subsequent dispatches are calculated via the MVP.

If the number of orders available at the start of the day is large, the generalized MVP may not capture additional opportunities for routing efficiency stemming from directly optimizing a vehicle routing problem for these orders; however, such opportunities do not relate to the SDD system and would rely on more established routing models.

In the single vehicle case, it is possible that a problem instance previously defined by N, T, and f is still feasible for $N' > 0$. Define an augmented problem with $\bar{N} = N' + N$, $\bar{T} = N' + T$, and $\bar{N}' = 0$. If the solution to this problem has an initial dispatch quantity of $\bar{q}_1^* \geq N'$, then the optimal quantities for the original instance are identical to those of the augmented one, with dispatch times moved up by N'.

By relaxing the gap time condition from section 3.2, we can solve the single-vehicle problem for any instance of $N' > 0$. Practically, this involves increasing the gap between the order cutoff time and the end of the service day.

Generalized gap time condition: If the parameters $T, N, N', q_{\min}$ satisfy $N + N' \leq T - f(2q_{\min})$, the instance is feasible with a single dispatch vehicle.

Assume the generalized gap time condition holds for parameters $T, N, q_{\min}, N'$. As before, define and solve an augmented problem with $\bar{N} = N' + N$, $\bar{T} = N' + T$, and $\bar{N}' = 0$. If one dispatch is optimal, it follows that $\bar{q}_1^* = \bar{N} \geq N'$. In the case of a multiple-dispatch optimal solution, the vehicle will only ever be idle at the depot at the start of the service day and will be done serving orders exactly at time $\bar{T}$. Thus, the total dispatching time is equal to $\bar{T} - \bar{q}_1^* = T + N' - \bar{q}_1^*$. Because the original problem over N, N', and T is feasible, we know that the total dispatch time for any optimal policy is less than or equal to T units of time. Additionally, any feasible solution to the original problem can be implemented in the augmented problem. Therefore the optimal solution to the augmented problem must use less than or equal to T units of dispatch time. It follows that $\bar{q}_1^* \geq N'$. Therefore, in all cases it is true that $\bar{q}_1^* \geq N'$, which implies that the optimal quantities for the original instance are identical to those of the augmented one, with dispatch times moved up by N'.

4.3 Capacitated Vehicles

Compared to traditional delivery settings, SDD systems operate in an environment with reduced order volume and much tighter time constraints. Therefore, in many cases the number of orders that can be delivered while satisfying time constraints is relatively small, and thus vehicle capacity is not a binding constraint. Nevertheless, there may be situations in which capacities must also be considered; we next discuss how our results extend to this case. Extending the notation we use in Section 4.2, suppose that each vehicle has capacity to serve at most Q orders. Consider first the many-vehicle case, and the following natural extension to the MVP.

Capacitated MVP Compute the MVP; if the first (and largest) dispatch serves Q or fewer orders, implement the policy. Otherwise, dispatch the first vehicle with Q orders, update T and N by subtracting Q, and recompute the MVP on the updated instance. Repeat until the computed MVP is feasible.

Intuitively, the MVP tries to serve as many orders as possible with each successive dispatch while still having the corresponding vehicle return by the end of the service day. The capacitated version of the policy does the same, but must also respect the additional vehicle capacity. As with the Generalized MVP in the previous section, this policy is also an optimal dispatching policy. The proof of this is identical to that of the Generalized MVP found in the appendix.

We can implement a similar policy modification in the single-vehicle case, where we additionally assume $Q \geq q_{\min}$. Note that the proof of Lemma 4 is not affected by adding a capacity, so the instance remains feasible. As in the many-vehicle case, Algorithm 1 constructs a solution in which the first dispatch is the largest; when this dispatch is too large for the capacity, the modification replaces it with the maximum possible quantity and iterates on the remaining, smaller instance.

Capacitated Single-Vehicle Policy Run Algorithm 1; if $\alpha^* \leq Q$, implement this solution. Otherwise, update T and N by subtracting Q, and run Algorithm 1 on the smaller instance. Repeat until $\alpha^* \leq Q$ in

Algorithm 1. If this process occurs k times before α^* is feasible, the solution to the instance has k dispatches of size Q before the dispatches given by Algorithm 1.

As a simple example, suppose $N + f(N) \leq T$ but $Q < N \leq 2Q$, i.e. a single dispatch is time-feasible but capacity-infeasible, and two dispatches suffice (although this is not what we expect in SDD, the setting serves for illustration purposes). In this case, it is optimal for the first dispatch to take Q orders, and the second dispatch takes the rest; the two dispatches can take place consecutively, returning at time T. More generally, we are able to give the following guarantee.

Corollary 11. *Suppose $Q \geq 2q_{\min}$. The capacitated single-vehicle policy produces an optimal solution to the single-vehicle instance with vehicle capacity Q.*

The proof of Theorem 5 can be modified slightly to prove Corollary 11. The only additional consideration needed is that each vehicle takes all unserved orders at the depot at the time of dispatch, *up to quantity Q*. The requirement $Q \geq 2q_{\min}$ is a technical condition necessary to use that same proof; however, in practice it is reasonable to expect that the capacity of a delivery vehicle is at least twice its minimum quantity.

4.4 Choosing Order Cutoff Time

Consider again the instance in Example 2. In an optimal solution, the second vehicle has almost 53 minutes of slack between its earliest possible arrival back to the depot and the end of the service day T. A system designer could consider either reducing N so the system requires only one vehicle, or increasing N to serve more orders with this second vehicle and increase its utilization.

For this discussion, we fix $T, f, q_{\min}$ and allow N to vary in $[0, U]$, where U is an upper bound chosen by the system manager. We assume that earned revenue from orders served is proportional to N with constant β, and operating costs are proportional to the dispatch time of the optimal policy, denoted by $g(N)$. Without loss of generality, we scale β so we can compare revenue against cost. Suppose the SDD manager wishes to choose a cutoff time that maximizes system profit as measured by earned revenue minus operating costs. Equivalently, this cutoff time would also minimize the cost of serving orders that occur before the cutoff plus the opportunity cost of not serving orders after the cutoff. The profit maximizing cutoff time is then given by

$$\max_{0 \leq N \leq U} \pi(N) = \beta N - g(N). \tag{2}$$

First, we analyze the many-vehicle case. Recall that in the MVP, if the solution uses m vehicles, the first $m-1$ return exactly at time T. If the cutoff time is chosen carefully, the last dispatch also returns precisely at this time. Specifically, let N_i be the cutoff time at which the MVP uses exactly i vehicles, with all vehicles returning exactly at time T. Letting, $N_0 = 0$, we have the recursion $N_i = N_{i-1} + \Delta_i$, where Δ_i uniquely solves $\Delta_i + f(\Delta_i) = T - N_{i-1}$. Define N_U as the largest such value such that $N_U \leq U$. The following theorem leverages these values to search for an optimal cutoff time for (2) in the many-vehicle case; it is proved in the appendix.

Theorem 12. *In the many-vehicle case, an optimal solution for* (2) *can be found in the set* $\{N_0, N_1, \ldots, N_U, U\}$.

Now we analyze the single-vehicle case, where we impose the upper bound $U \leq T - f(2q_{\min})$ on N. Define i_{max} as the number of dispatches in the optimal dispatch policy when $N = U$, and suppose $i_{max} \leq 2$. Define $\bar{N}_1$ such that for $N \in (0, \bar{N}_1]$, the optimal dispatch policy determined via Theorem 5 uses exactly one dispatch.

Proposition 13. *In the single vehicle case, for any U such that $0 \leq U \leq T - f(2q_{\min})$ and $i_{max} \leq 2$, the optimal solution of* (2) *satisfies $N^* \in \{0, \bar{N}_1, U\}$.*

The proof of the proposition can be found in the appendix. The proof for the general case of i_{max} would follow from showing that $g(N)$ is piecewise concave with breakpoints at each $\bar{N}_i$. We have empirical evidence that this is indeed the case but have not been able to prove it.

We now return to the instance in Examples 2 and 7, and calculate an optimal value for the cutoff time N.

Example 14. Consider the same instance as in the previous examples with $T = 90, f(n) = 2.15\sqrt{n} + .13n$, and $q_{\min} = 12$. Suppose $\beta = 0.80$ and profit is given in scaled monetary units.

For the many-vehicle case, we calculated the first four values of N_i to be $N_0 = 0, N_1 = 64.38, N_2 = 79.62$, and $N_3 = 84.57$. The associated dispatch times are 0, 25.62, 36.00, and 41.43, which result in profits of $\pi(N_0) = 0$, $\pi(N_1) = 25.88$, $\pi(N_2) = 27.70$, and $\pi(N_3) = 26.22$. By Theorem 12, the optimal order cutoff time is $N^* = N_2$, with an optimal dispatch policy of $\{(t_1 = 64.38, q_1 = 64.38, i_1 = 1), (t_2 = 79.62, q_2 = 15.24, i_2 = 2)\}$.

For the single-vehicle case, we let $U = T - f(2q_{\min}) = 76.35$, and it follows that $i_{max} = 2$. Note that, $\bar{N}_1 = 64.38$. The associated costs of $N^* \in \{0, \bar{N}_1, U\}$ are 0, 25.62, and 36.04, which result in profits of $\pi(0) = 0$, $\pi(\bar{N}_1) = 25.88$, and $\pi(U) = 25.38$. Thus, the optimal cutoff time is $N^* = \bar{N}_1$ with the corresponding policy $\{(t_1 = 64.38, q_1 = 64.38, i_1 = 1)\}$.

Figures 4 and 5 plot $\pi(N)$ for this instance in the many and single vehicle cases, respectively.

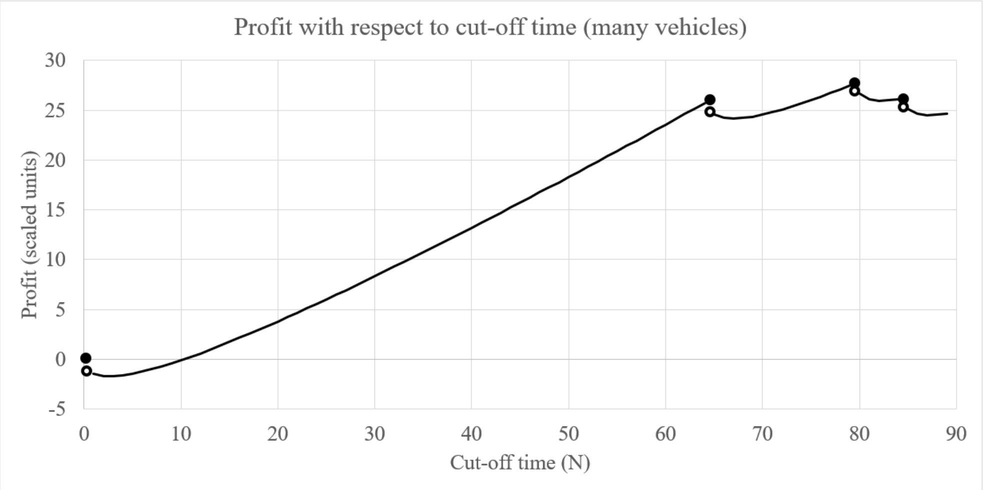

Figure 4: Profit function with respect to cutoff time, many-vehicle case

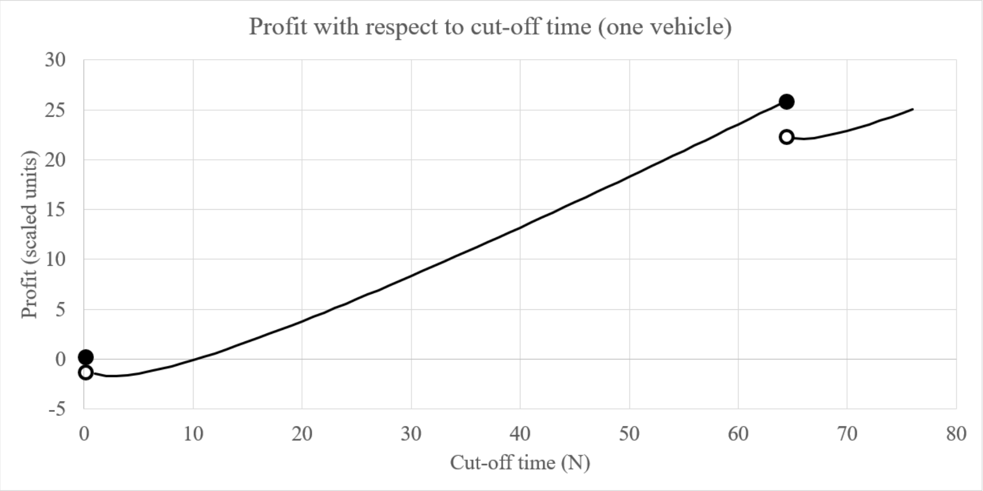

Figure 5: Profit function with respect to cutoff time, single-vehicle case

5 Computational Study

Our SDD planning model uses continuous approximations to preserve simplicity and transparency of analysis, and the previous sections discuss various ways in which the model can be used to inform managerial decisions about an SDD system, including fleet sizing, the choice of service cutoff time, and so forth. When making such decisions, we naturally depend on the accuracy and fidelity of the model when compared to more granular operational models. We now present a computational case study that empirically demonstrates the accuracy of the model and its potential practical use.

Our study considers a hypothetical SDD system where one service area comprises roughly 26 square miles in northeastern metro Atlanta. Specifically, this service region consists of the 22 census tracts north of Interstate 85, south of Interstate 285 and east of Georgia Highway 400, with a population of 92,198 as measured by the U.S. Census Bureau [5]. For the study, we chose five representative addresses within each tract, for a total of 110 potential customer locations, plus a depot location on the northeast border, on Interstate 285. Figure 6 depicts the service region and the 110 representative customer locations.

We assume SDD orders begin at 9 am, and the service day ends at 6 pm; we evaluate two different order cutoff times below in different experiments. Assuming 5% of the working population in the region would like to use the SDD service once every other month within the service day, an order would arrive on average approximately once every six minutes over this time, and we use this rate in the model. However, we do not assume orders are equally likely to appear anywhere in the region: We assign each tract a weight proportional to the product of its median household income and its population (both available from [5]); assuming an order originates in a tract, one of its five representative locations is chosen uniformly at random.

To construct the routing component of our dispatch time function, we sampled customer locations with replacement according to the weighted distribution described above, with sample sizes ranging from 10 to 75 locations and a total of 1,980 samples. We used the Google Maps API [28] to query driving time between every pair of locations, and for each sample calculated the optimal TSP route time for a vehicle route visiting these locations from the depot. From this, we obtained via linear regression a routing time approximation of $24\sqrt{n}$ minutes for n orders, with an an R-squared value of 0.94. Furthermore, we include 1.5 minutes of service time per order and a fixed setup time of 10 minutes per dispatch. After re-scaling the instance parameters to be measured in increments of six minutes, this results in an order arrival rate $\lambda = 1$, service

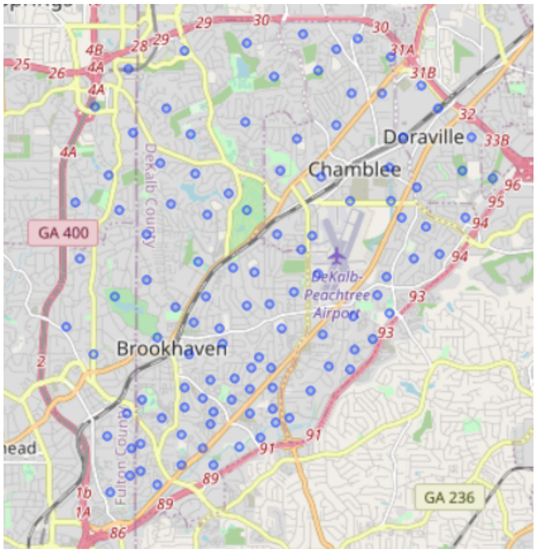

Figure 6: Service region in northeast Atlanta, Georgia, USA. Image generated in RStudio with the library “leaflet”, containing all 110 possible customer locations taken from 22 census tracts [5].

day length $T = 90$ and dispatch time function

$$f(n) = 1.67 + 0.25n + 4\sqrt{n}.$$

For our operational simulation, we replace the constant arrival rate with a Poisson arrival process that uses the same rate; as in our calibration experiment, an order’s tract is chosen randomly with weights proportional to the product of median household income and population, and its location within the tract is chosen uniformly at random. We use actual driving times between pairs of locations given by the Google API and determine the total routing time for a dispatch by solving a TSP using a standard integer programming formulation implemented in Gurobi 7.5.2. Our experiments are coded in Python 3.6.3 and run on a Linux computing cluster, which employs HTCondor 8.8.4.

5.1 Many-Vehicle Policy

First, we assume that this service region is served by two delivery vehicles each performing a single daily dispatch. Following the reasoning from Section 4.4, we choose a cutoff time to fully utilize both vehicles. With these parameters, the MVP prescribes that the first dispatch takes 48.40 orders and its route requires a duration of just under 250 minutes, while the second dispatch takes 18.26 orders with a duration of approximately 140 minutes; the total dispatch time sums to about 389 minutes. The predicted dispatch time for the second vehicle translates to an order cutoff time of 3:40 PM ($N = 66.66$).

We now describe our simulated operational benchmark for the many-vehicle policy. The dispatch time required to serve a set of orders is the sum of the setup, service and routing time, where the routing time is given by a TSP from the depot to each of the orders’ locations using actual driving times. The dispatcher allows unserved orders to accumulate as long as their dispatch time is less than the remaining time in the service day; when the two times are equal, a vehicle is dispatched with the unserved orders. When a new order arrives, the dispatcher recalculates the total dispatch time including the new order; if the new dispatch time exceeds the remaining time in the service day, the vehicle is immediately dispatched *without this order*, to ensure the vehicle returns before the end of the service day. If this is the first dispatch, the new order is assigned to the second vehicle and the process repeats; otherwise, this order is not accepted for SDD. More generally, the dispatcher stops accepting orders once the second vehicle is dispatched, which may occur before or after the nominal cutoff time of 3:40 PM; this represents a dynamic modification of the cutoff time at the operational level and is in line with how some SDD services operate in practice.

We simulated the operational benchmark 300 times. For each realization, we record the the number of orders served and the dispatch time for each vehicle, respectively. Table 2 reports results; for each quantity we include the prediction of our tactical model, and the sample mean and 95% confidence intervals of the operational benchmark. As the table shows, our tactical model predicts the expected number of orders served and the expected total dispatch time to within less than 1%.

	Tactical	**Operational**	**A Posteriori**
First Dispatch Quantity	48.40 units	48.20 units (± 0.51)	43.90 units (± 0.63)
First Dispatch Time	249.58 min.	249.69 min. (± 1.81)	228.07 min. (± 2.62)
Second Dispatch Quantity	18.26 units	18.45 units (± 0.35)	22.75 units (± 0.45)
Second Dispatch Time	139.95 min.	139.16 min. (± 1.47)	144.88 min. (± 1.48)
Total Quantity	66.66 units	66.65 units (± 0.71)	66.65 units (± 0.71)
Total Time	389.53 min.	388.85 min. (± 2.85)	372.95 min. (± 3.29)

Table 2: Computational study results, many-vehicle policy.

Because our main goal is tactical design and describing average behavior rather than operational management, we do not consider many potential dispatching improvements or modifications that could be used at the operational stage; the logistics literature has several works dedicated to this question, e.g. [22, 33, 36]. Nevertheless, it is also important to assess the quality of our prescribed solution when compared to what an operational decision support tool could accomplish.

Motivated by this question, we compute an *a posteriori* or "hindsight-optimal" solution for each simulated realization, which provides a lower bound on the dispatch time any operational policy can achieve. For each realization, we assume that the dispatcher knows in advance the exact time and location of each order served by our operational benchmark and then optimizes the two vehicles' routes with this knowledge. For example, in one of the realizations the operational benchmark could have both vehicles visiting the same neighborhood to deliver two different orders; the *a posteriori* solution could use its advance knowledge to shift the first order to be served by the second vehicle (with virtually no increase in its dispatch time), while deleting this order from the first vehicle's dispatch would reduce its dispatch time. This example also illustrates that despite having advance knowledge of order times and locations, the *a posteriori* solution must still satisfy operational constraints; in particular, an order can only be served by a dispatch that departs the depot after the order ready time, and the vehicles must return to the depot by the end of the service day. In the appendix, we include the formulation we use to compute these solutions.

In our experiments, for each of the 300 simulated realizations we optimize the *a posteriori* solution in Gurobi with a two-hour time limit. As Table 2 details, the sample mean of the total dispatch time in the *a posteriori* solution is within approximately 4% of the operational benchmark. For comparison, operational SDD models are notoriously difficult to benchmark; many works in the SDD literature do not include lower bounds at all, and those that do often report larger gaps against *a posteriori solutions* even for complex heuristic policies, e.g. [22]. We therefore conclude that our model prescribes reasonable operational behavior, in line with what a dispatcher could accomplish with sophisticated decision-support tools.

5.2 Single-Vehicle Policy

We now suppose the service region is served with a single delivery vehicle. Since only a single vehicle is available, we also move the cutoff time back to 2:00 PM ($N = 50$), which results in a more reasonable workload. Using Algorithm 1 and the results in Section 3.2, we obtain a two-dispatch solution: The first dispatch serves 35.01 orders with a time of about 205 minutes, and the second takes the remaining 14.99 orders with a duration of approximately 125 minutes.

The operational benchmark for the single-vehicle policy here is similar in spirit to the many-vehicle one. As our discussion in Section 3.2 suggests, each dispatch should take all currently unserved orders. To determine the time of first dispatch α, we operationally mimic the equation $\alpha + f(\alpha) + f(N - \alpha) = T$, which determines the dispatch time in the tactical model with two dispatches. Before the first dispatch, orders accumulate and the dispatcher iteratively recalculates their total dispatch time. Define τ to be the elapsed time in the service day, O_τ to be the set of orders that have arrived by τ, and $\text{dispatch}(O_\tau)$ to be their dispatch time. While $\tau + \text{dispatch}(O_\tau) + f(N - \tau) < T$, the dispatcher allows orders to accumulate; the first dispatch occurs when equality holds. As in the previous case, if a new order causes the left-hand value to exceed T, the vehicle is immediately dispatched without this order. While the vehicle is en route, we know its return time and thus the maximum possible duration of the next dispatch such that it returns by T. As new orders arrive, the dispatcher again calculates their dispatch time, and accept orders only until the next dispatch's total duration matches the remaining time in the service day. As in the many-vehicle case, this corresponds to an operational dynamic adjustment of the order cutoff time that, in expectation, should match N if our model is accurate. Note also that we can extend the benchmark in an analogous fashion to single-vehicle problems with more than two dispatches.

We again simulated 300 realizations of the operational benchmark, and report results in Table 3, in a similar fashion to Table 2. As in the previous experiment, our tactical model predicts the expected number of orders served and the expected total dispatch time of the operational benchmark to within 1% or less. In the second dispatch, we see a slightly higher quantity served on average in the operational benchmark compared to the tactical prediction. This is due to the routing time approximation's slight conservatism for smaller values, which allows the operational policy to serve a slightly higher number of orders than expected on average. Furthermore, we again implemented an *a posteriori* benchmark, which allows the dispatcher to optimize the two delivery routes with full advance knowledge of the time and location of each order, using the same experimental setup as in the many-vehicle case. In this case, the operational policy is within approximately 13% of the *a posteriori* benchmark, suggesting again that our system is modeling reasonable behavior when compared to what a complex operational decision support tool can hope for. Interestingly, in the single-vehicle case, we observe that the *a posteriori* solution moves more orders to the second dispatch; with advance knowledge of future order locations, the dispatcher is able to anticipate areas where more orders will take place late in the day, and wait until the second dispatch to serve these locations more efficiently. Nevertheless, the gaps we observe between the operational benchmark and a posteriori solution in this experiment are still in line with other results in the operational SDD literature [22].

	Tactical	**Operational**	**A Posteriori**
First Dispatch Quantity	35.01 units	34.96 units (± 0.30)	12.99 units (± 1.32)
First Dispatch Time	204.52 min.	203.45 min. (± 0.74)	87.28 min. (± 6.48)
Second Dispatch Quantity	14.99 units	15.57 units (± 0.63)	37.54 units (± 1.45)
Second Dispatch Time	125.41 min.	123.73 min. (± 3.27)	196.91 min. (± 6.61)
Total Quantity	50.00 units	50.53 units (± 0.73)	50.53 units (± 0.73)
Total Time	329.93 min.	327.18 min. (± 3.79)	284.19 min. (± 3.26)

Table 3: Computational study results, single-vehicle policy.

5.3 Many Capacitated Vehicles

Having established the model's prediction accuracy for our two base cases in the previous experiments, we next examine vehicle capacities and their impact on model accuracy. Specifically, we consider the same experimental setup from the many-vehicle policy in Section 5.1, with order cutoff at 3:40 PM ($N = 66.66$), but we additionally suppose that delivery vehicles have a capacity of $Q = 20$ orders. For this instance, the capacitated MVP (Section 4.3) has four dispatches; the first three are at capacity, each serving 20 orders with dispatch duration of about 147 minutes; the fourth dispatch serves the remaining 6.66 orders with a predicted dispatch duration of about 82 minutes.

The operational benchmark here is almost identical to the one used in Section 5.1, with the additional constraint that when 20 orders accumulate, the current vehicle is dispatched immediately. We examine the performance of the operational benchmark on 300 simulated service days (the same simulations used in the uncapacitated experiment), with results in Table 4. As before, the tactical model predicts orders served within 1% of its operational benchmark. In this case, the total predicted dispatch time is within 3.5% of the benchmark. We conclude that the tactical model still makes accurate predictions in the presence of a larger number of vehicles with a maximum capacity. Moreover, as the table indicates, the slightly larger discrepancy between predicted and observed dispatch time is mostly due to two factors: First, the third dispatch is not always at capacity, serving about 19 orders on average; second, and most importantly, the fourth dispatch's observed duration is lower than its prediction. Both factors can be explained by the routing approximation's slight conservatism and decrease in accuracy for a small number of locations, particularly the very small number of orders served by the last dispatch. (Since continuous approximations assume a large number of locations, they can be slightly inaccurate when the number of locations is very small.) In practice, our discussion in Section 4.4 suggests that the manager of this SDD system would probably prefer to either decrease the order cutoff to 60 and fully utilize only three vehicles, or perhaps to increase the cutoff so the fourth vehicle can be better utilized.

	Tactical	**Operational**
First Dispatch Quantity	20.00 units	20.00 units (± 0.00)
First Dispatch Time	147.33 min.	145.80 min. (± 0.83)
Second Dispatch Quantity	20.00 units	20.00 units (± 0.00)
Second Dispatch Time	147.33 min.	145.89 min. (± 0.91)
Third Dispatch Quantity	20.00 units	18.87 units (± 0.27)
Third Dispatch Time	147.33 min.	141.18 min. (± 1.52)
Fourth Dispatch Quantity	6.66 units	7.28 units (± 0.64)
Fourth Dispatch Time	81.93 min.	72.54 min. (± 3.62)
Total Quantity	66.66 units	66.15 units (± 0.89)
Total Time	523.92 min.	505.42 min. (± 6.51)

Table 4: Computational study results, four capacitated vehicles at $Q = 20$.

6 Conclusions

We have proposed a tactical analysis model for same-day delivery that captures operations at the level of a single depot and its service region. By approximating the order arrival process and the dispatch time, we are able to derive simple and transparent optimal solutions for the model that describe the average performance of a reasonable SDD system; our empirical validation shows that the model can indeed predict system behavior very accurately at an operational level.

Using our model, a system manager can easily perform what-if analysis on various potential system configurations, and compare the cost and operating conditions of these configurations to decide various tactical questions, such as the size of the delivery fleet, the order cutoff time, or whether to have vehicles deliver to the entire service region versus partitioning the region by vehicle. We similarly hope the community derives other applications of the model in SDD tactical design.

Our results motivate several interesting avenues for research. One possibility is to further investigate the interplay of service region partitioning with our model. For example, it would be useful for SDD managers to know precisely when partitioning is preferable to serving the whole region, or to determine if the system can operate more efficiently by serving different parts of the service region differently. A manager may wish to offer SDD with different cutoff times in different areas, based on how efficiently customers in the different areas can be served; perhaps more densely populated urban centers can be profitably served until later in the day while outlying suburban areas need an earlier cutoff. More generally, it may be useful to address partitioning and fleet sizing in tandem, where some sub-regions are served by more vehicles because of higher order density, while others get a smaller delivery fleet because of relative order paucity.

Acknowledgements

The authors thank the department editor, associate editor and referees for their prompt feedback and constructive suggestions, which helped greatly improve the manuscript.

References

[1] M. Albareda-Sambola, E. Fernández, and G. Laporte, *The dynamic multiperiod vehicle routing problem with probabilistic information*, Computers & Operations Research **48** (2014), 31–39.

[2] D. Applegate, R. Bixby, V. Chvátal, and W. Cook, *The traveling salesman problem: A computational study*, Princeton University Press, 2011.

[3] J. Beardwood, J. Halton, and J. Hammersley, *The shortest path through many points*, Mathematical Proceedings of the Cambridge Philosophical Society **55(4)** (1959), 299–327.

[4] R. Bent and P. Van Hentenryck, *Scenario-based planning for partially dynamic vehicle routing with stochastic customers*, published online in Operations Research **52(6)** (2004), 977–987.

[5] United States Census Bureau, *Census explorer*, `https://www.census.gov/censusexplorer/censusexplorer.html`, [Online; accessed 06-November-2019].

[6] ———, *Annual Retail Trade Survey: Sales Estimates (1992-2016)*, `https://www.census.gov/data/tables/2016/econ/arts/annual-report.html`, 2018, [Online; accessed 18-November-2018].

[7] A. Campbell and B. Thomas, *Probabilistic traveling salesman problem with deadlines*, published online in Transportation Science **42(1)** (2008), 1–21.

[8] J. Carlsson, *Dividing a territory among several vehicles*, INFORMS Journal on Computing **24(4)** (2012), 565–577.

[9] J. Carlsson and S. Song, *Coordinated logistics with a truck and a drone*, published online in Management Science **64(9)** (2017), 1–31.

[10] E. Çeven and K. Gue, *Optimal Wave Release Times for Order Fulfillment Systems with Deadlines*, Transportation Science **51** (2017), 52–66.

[11] BRP Consulting, *Digital commerce survey: Annual survey 2017*, Tech. report, 2017, [Online; accessed 18-November-2018].

[12] FedEx Corp., *Same-day delivery*, `https://www.fedex.com/en-us/shipping/same-day.html#sameday-city-standard`, 2019, [Online; accessed 19-February-2019].

[13] C. Daganzo, *The length of tours in zones of different shapes*, Transportation Research Part B: Methodological **18(2)** (1984), 135–145.

[14] M. Figliozzi, *Analysis of the efficiency of urban commercial vehicle tours: Data collection, methodology, and policy implications*, Transportation Research Part B: Methodological **41(9)** (2007), 1014–1032.

[15] A. Franceschetti, O. Jabali, and G. Laporte, *Continuous approximation models in freight distribution management*, TOP **25(3)** (2017), 413–433.

[16] Amazon.com Inc., *Amazon launches same day delivery in seven major cities and expands saturday delivery options [press release]*, `https://press.aboutamazon.com/news-releases/news-release-details/amazon-launches-same-day-delivery-seven-major-cities-and-expands`, 2009, [Online; accessed 18-November-2018].

[17] ———, *Prime free same-day delivery expands to 11 new metro areas [press release]*, `https://press.aboutamazon.com/news-releases/news-release-details/prime-free-same-day-delivery-expands-11-new-metro-areas`, 2016, [Online; accessed 18-November-2018].

[18] ———, *Amazon expands free shipping to everyone for the holiday season [press release]*, https://press.aboutamazon.com/news-releases/news-release-details/amazon-expands-free-shipping-everyone-holiday-season, 2018, [Online; accessed 18-November-2018].

[19] ———, *Order with prime free same-day delivery*, https://www.amazon.com/gp/help/customer/display.html?nodeId=201631620, 2019, [Online; accessed 19-February-2019].

[20] D. Johnson, L. McGeoch, and E. Rothberg, *Asymptotic experimental analysis for the Held-Karp traveling salesman bound*, Proceedings of the Seventh Annual ACM-SIAM Symposium on Discrete Algorithms (1996), 341–350.

[21] M. Klapp, *Dynamic Optimization for Same-Day Delivery Operations*, Ph.D. thesis, Georgia Institute of Technology, 2016.

[22] M. Klapp, A. Erera, and A. Toriello, *The dynamic dispatch waves problem for same-day delivery*, European Journal of Operations Research **271(2)** (2018), 519–534.

[23] ———, *The one-dimensional dynamic dispatch waves problem*, Transportation Science **52(2)** (2018), 402–415.

[24] M. Lim, H. Mak, and Z. Shen, *Agility and proximity considerations in supply chain design*, Management Science **63** (2016), no. 4, 1026–1041.

[25] S. Liu, L. He, and Z. Shen, *On-time last mile delivery: order assignment with travel time predictors*, Available at SSRN: https://ssrn.com/abstract=3179994 (2018).

[26] A. Nabila, M. Gendreau, and J. Potvin, *A dynamic vehicle routing problem with multiple delivery routes*, Annals of Operations Research **199(1)** (2012), 103–112.

[27] G. Newell and C. Daganzo, *Design of multiple-vehicle delivery tours I: A ring-radial network*, Transportation Research Part B: Methodological **20(5)** (1986), 345–363.

[28] Google Maps Platform, *Distance matrix api*, https://developers.google.com/maps/documentation/distance-matrix/intro#DistanceMatrixRequests, [Online; accessed 06-November-2019].

[29] D. Reyes, A.L. Erera, and M.W.P. Savelsbergh, *Complexity of routing problems with release dates and deadlines*, European Journal of Operational Research **266** (2018), no. 1, 29–34.

[30] M. Ulmer, *Delivery deadlines in same-day delivery*, Logistics Research **10(3)** (2017), 1–15.

[31] ———, *Dynamic pricing for same-day delivery routing*, (2017), Working Paper.

[32] M. Ulmer and B. Thomas, *Same-day delivery with a heterogeneous fleet of drones and vehicles*, Networks **72(4)** (2018), 475–505.

[33] M. Ulmer, B. Thomas, and Mattfeld D., *Preemptive depot returns for a dynamic same-day delivery problem*, EURO journal on Transportation and Logistics (2018).

[34] W. Van Heeswijk, M. Mes, and J. Schutten, *The delivery dispatching problem with time windows for urban consolidation centers*, published online in Transportation Science **53(1)** (2017), 1–19.

[35] S. Voccia, A. Campbell, and B. Thomas, *The probabilistic traveling salesman problem with time windows*, EURO Journal on Transportation and Logistics **2(1-2)** (2013), 89–107.

[36] ______, *The same-day delivery problem for online purchases*, published online in Transportation Science **53(1)** (2017), 1–39.

[37] H. Wang and A. Odoni, *Approximating the performance of a "last mile" transportation system*, Transportation Science **50** (2014), no. 2, 659–675.

7 Appendix

7.1 Proof of Theorem 1

We start our proof by stating the optimization problem presented in Section 2 for the many-vehicle case. Without loss of generality, we can assume that no vehicle is used more than once. Here, a given dispatch policy will have the form $\{(t_d, q_d)\}_{d=1}^{D}$, where the d-th vehicle serves order quantity q_d at time t_d. For any such dispatch policy to be feasible, the following constraints must be satisfied:

$$\sum_{d=1}^{D} q_d = N, \tag{3a}$$

$$q_d \geq 0 \quad \forall d, \tag{3b}$$

$$t_d + f(q_d) \leq T \quad \forall d, \tag{3c}$$

$$t_d \geq 0 \quad \forall d, \tag{3d}$$

$$\sum_{\delta=1}^{d} q_\delta \leq t_d \quad \forall d \tag{3e}$$

An optimal dispatch policy minimizes $\sum_{d=1}^{D} f(q_d)$, subject to (3a)-(3e), over all $D \geq 1$. The Many-Vehicle Policy is feasible, satisfies (3c) at equality for $d = 1, 2, \ldots, D-1$, and (3e) at equality for all d. By construction, the policy is unique for a given problem instance. Theorem 1 claims that this policy is optimal.

First, we prove that there exists some optimal dispatch policy, $\{(t_d^*, q_d^*)\}_{d=1}^{D^*}$, which uses a finite number of vehicles, D^*. For the problem instance to be feasible, there must be a quantity q' with $0 < q' \leq N$, and $f(q') \leq T - N$. Now consider any feasible dispatch policy that uses more than $2\lceil N/q' \rceil$ dispatches. There are at least two dispatches with $q'/2$ or fewer orders. By Lemma 3, we can consolidate these two dispatches into a single dispatch that leaves at time $t = N$ while preserving feasibility, and without increasing the total dispatch time of the policy. Therefore, there exists some optimal dispatch policy $\{(t_d^*, q_d^*)\}_{d=1}^{D^*}$ with $D^* \leq 2\lceil N/q' \rceil$. Fix such a policy.

We now show that this policy can be transformed into one where all constraints (3e) hold at equality. We first reduce t_1^* until (3e) is satisfied at equality for $d = 1$. This process can be repeated for $d = 2, 3, \ldots$ until we set $t_{D^*} = N$. As we only possibly decreased each time of dispatch, constraints (3a)-(3d) remain satisfied, and the objective value remains the same. This adjusted policy that satisfies all constraints (3e) at equality is also an optimal dispatch policy.

We next show that it is possible to re-order, if necessary, the dispatch quantities of the optimal policy so that they are non-increasing. Suppose two consecutive dispatches (t_d^*, q_d^*) and (t_{d+1}^*, q_{d+1}^*) have $q_d^* < q_{d+1}^*$; we can replace these two dispatches with $(t_d^* + (q_{d+1}^* - q_d^*), q_{d+1}^*)$ and (t_{d+1}^*, q_d^*). The d-th dispatch now leaves with q_{d+1}^* units at an earlier time than t_{d+1}^* and thus is feasible. The new $(d+1)$-th dispatch leaves at the same time as before, but now with fewer units, so it is also feasible. As the total dispatch time remains the same, this feasible dispatch remains optimal. Furthermore, all of the constraints in (3e) are still satisfied at equality by construction. By repeating this operation as necessary, we may assume our optimal dispatch policy has non-increasing dispatch quantities.

If constraint (3c) holds at equality for $d = 1, 2, \ldots, D^* - 1$, we are done. Assume this is not the case. Take the first dispatch d' such that $t^*_{d'} + f(q^*_{d'}) < T$. This dispatch and its successor are given by $(t^*_{d'}, q^*_{d'})$ and $(t^*_{d'+1}, q^*_{d'+1})$. Replace these dispatches with $(t^*_{d'} + \sigma, q^*_{d'} + \sigma)$ and $(t^*_{d'+1}, q^*_{d'+1} - \sigma)$, where $\sigma = \min\{\varepsilon, q^*_{d'+1}\}$, and $\varepsilon > 0$ is the unique value with $t^*_{d'} + \varepsilon + f(q^*_{d'} + \varepsilon) = T$. By the definition of σ, this adjusted dispatch policy is feasible. Furthermore, since $q^*_{d'} \geq q^*_{d'+1}$, Lemma 3 implies that moving σ units from dispatch $d' + 1$ to d' cannot increase the total dispatch time; thus this adjusted policy remains optimal. See Figure 7 for an illustration.

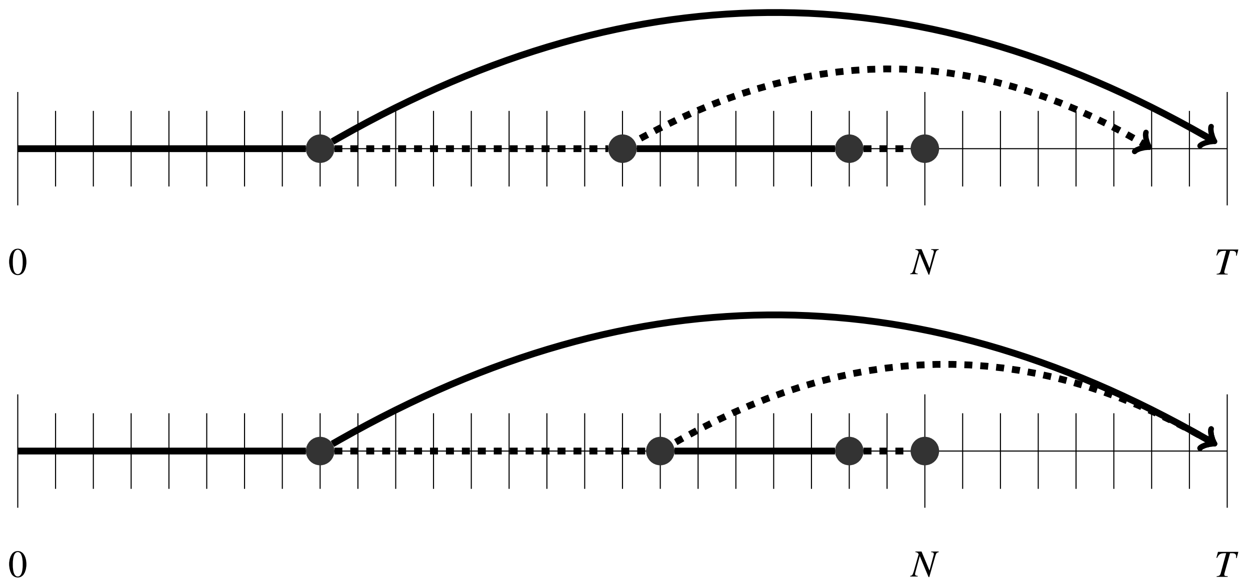

Figure 7: If some dispatch, not the last one, returns to the depot before T, it would be better for this dispatch to stay at the depot longer before dispatching, "eating away" at the next dispatch. Note that the time of the next dispatch does not change.

If $\sigma = \varepsilon$, the resulting policy now has (3c) holding at equality for d'. If $\sigma = q^*_{d'+1}$, the resulting policy has one fewer dispatch. After either adjustment we can again again re-order the remaining dispatches to be non-increasing if necessary. Thus, after this operation, we have either reduced the number of dispatches or the optimal policy satisfies the first d' constraints (3c) at equality. Furthermore, constraints (3e) still hold at equality by construction. Therefore, after a finite number of adjustments the resulting optimal policy satisfies (3c) at equality for all $d' \leq D^* - 1$.

We have constructed an optimal dispatch policy satisfying (3c) at equality for $d = 1, 2, \ldots, D^* - 1$ and (3e) at equality for all d. This optimal dispatch policy is identical to the one prescribed by the Many-Vehicle Policy, so the first part of Theorem 1 is proved. The final claim of the theorem states that if the optimization problem presented in Section 2 is constrained to use $m < m^* = D^*$ vehicles, its objective value is bounded below by the Many-Vehicle Policy's objective value. This follows because the Many-Vehicle Policy attains the minimum total dispatch time for any number of vehicles.

7.2 Proof of Lemma 4

Assume we are given an SDD problem instance with a single vehicle satisfying the sufficient processing speed and sufficient gap time conditions. We show that the instance has a feasible policy also satisfying the minimum dispatch size condition. First, let $D' = \lfloor N/q_{\min} \rfloor$.

Suppose $D' \leq 1$; then $N < 2q_{\min}$. Consider the dispatch policy $(t_1 = N, q_1 = N, 1)$. Because f is increasing we have $f(N) < f(2q_{\min})$. By the sufficient gap time condition, $N + f(N) < N + f(2q_{\min}) \leq T$, we see that the dispatch returns to the depot by the end of the service day and is therefore feasible. Additionally, note that the minimum dispatch size condition holds trivially.

Now assume $D' \geq 2$. Consider the dispatch policy

$$\begin{aligned}
&(t_1 = q_{\min}, q_1 = q_{\min}, 1), (t_2 = 2q_{\min}, q_2 = q_{\min}, 1), \ldots, \\
&(t_{D'-1} = (D'-1)q_{\min}, q_{D'-1} = q_{\min}, 1), (t_{D'} = N, q_{D'} = N - (D'-1)q_{\min}, 1).
\end{aligned}$$

By the sufficient processing speed condition, $f(q_{\min}) \leq q_{\min}$, which implies that the first $D'-1$ dispatches return to the depot before $q_{\min}$ more orders accumulate. Additionally, $q_{D'} \geq q_{\min}$, so the first $D'-1$ dispatches return to the depot before the next dispatch must leave. The last dispatch takes all of the remaining orders. By the choice of D' we have that $2q_{\min} > q_{D'}$, implying $f(q_{D'}) < f(2q_{\min})$ and subsequently $N + f(q_{D'}) < N + f(2q_{\min}) \leq T$ by the sufficient gap time condition. As the minimum dispatch size condition holds and the final dispatch returns to the depot by the end of the service day, the proposed policy is feasible and we are done.

7.3 Proof of Theorem 5

We start our proof by formulating the optimization problem presented in Section 2 for the single-vehicle case. A given dispatch policy in this case has the form $\{(t_d, q_d)\}_{d=1}^{D}$. For any such dispatch policy to be feasible, the following constraints must be satisfied:

$$\sum_{d=1}^{D} q_d = N, \tag{4a}$$

$$q_d \geq 0 \quad \forall d, \tag{4b}$$

$$t_D + f(q_D) \leq T, \tag{4c}$$

$$t_d + f(q_d) \leq t_{d+1} \quad \forall d \leq D-1, \tag{4d}$$

$$t_d \geq 0 \quad \forall d, \tag{4e}$$

$$\sum_{\delta=1}^{d} q_\delta \leq t_d \quad \forall d. \tag{4f}$$

An optimal policy minimizes $\sum_{d=1}^{D} f(q_d)$, subject to (4a)-(4f), over all $D \geq 1$. Additionally, Theorem 5 assumes the following conditions are met:

$$f(x) \leq x/\lambda = x \quad \forall x \geq q_{\min}, \tag{5a}$$

$$q_d \geq q_{\min} \quad \forall d < D, \tag{5b}$$

$$T - N \geq f(2q_{\min}) \tag{5c}$$

Note that (5a), (5b) and (5c) correspond to the sufficient processing speed, minimum dispatch size and sufficient gap time conditions, respectively. We prove that there exists an optimal dispatch policy $\{(t_d^*, q_d^*)\}_{d=1}^{D^*}$ satisfying the following conditions:

(C1) each dispatch takes all available unserved orders at the depot at the time of dispatch,

(C2) after the first dispatch, the vehicle never waits at the depot again, and

(C3) if the vehicle is dispatched more than once, the last dispatch arrives back at the depot at exactly time $t = T$.

Mathematically, these conditions can be expressed as

(C1) $q_1^* = t_1^*$ and $\sum_{\delta=1}^{d} q_\delta^* = t_d^* \;\; \forall d \leq D^* - 1$,

(C2) $t_d^* + f(q_d^*) = t_{d+1}^* \;\; \forall d \leq D^* - 1$, and

(C3) if $D^* \geq 2$, then $t_{D^*}^* + f(q_{D^*}^*) = T$.

We also prove that $q_1^* \geq q_2^* \geq \cdots \geq q_{D^*}^*$ (C4), i.e. the policy's order quantities are non-increasing. We now assume that (5a) and (5c) hold for a given problem instance and that any feasible policy must satisfy (4a)-(4f) as well as (5b). By Lemma 4, such a feasible policy must exist.

If $N + f(N) \leq T$, a single dispatch is optimal and trivially satisfies (C1) through (C4). Now assume $N + f(N) > T$. Because of (4a) and (5b), a feasible dispatch policy cannot use more than $\lceil N/q_{\min} \rceil$ dispatches. From this we conclude that an optimal policy with finitely many dispatches exists and we can subsequently fix such a policy as $\{(t_d^*, q_d^*)\}_{d=1}^{D^*}$, where $D^* \geq 2$.

Suppose (4c) holds strictly; then we can increase $t_{D^*}^*$ until the constraint is binding without loss of optimality. Assuming (4c) is binding, if any of (4d) hold strictly, we can similarly increase dispatch times, beginning with $t_{D^*-1}^*$ and working backwards to t_1^*, until all constraints (4d) are also binding, without loss of optimality. This transformed policy satisfies (C2) and (C3).

Next, suppose the policy does not satisfy (C4). Fix the smallest index d such that $q_d^* < q_{d+1}^*$. Similarly to the proof of Theorem 1, we can replace dispatches (t_d^*, q_d^*), (t_{d+1}^*, q_{d+1}^*) with (t_d^*, q_{d+1}^*), $(t_d^* + f(q_{d+1}^*), q_d^*)$. This swap does not alter the total dispatch time, and (C2) and (C3) still hold after the change. The feasibility of the new solution follows by showing that the d-th dispatch can take q_{d+1}^* units:

$$t_d^* = t_{d+1}^* - f(q_d^*) \geq \sum_{\delta=1}^{d+1} q_\delta^* - f(q_d^*) \geq \sum_{\delta=1}^{d-1} q_\delta^* + q_{d+1}^*.$$

Note that $t_d^* = t_{d+1}^* - f(q_d^*)$ follows from (C2), $t_{d+1}^* \geq \sum_{\delta=1}^{d+1} q_\delta^*$ follows from (4f), and $q_d^* - f(q_d^*) \geq 0$ follows from (5a) as (5b) is satisfied. We can iteratively perform this operation as necessary, preserving feasibility and the total dispatch time, until the solution satisfies (C4).

We may now assume the policy satisfies (C2) through (C4), and suppose by contradiction that it does not satisfy (C1). Let $d < D^*$ be the first dispatch that violates (C1), so we have $\varepsilon = t_d^* - t_{d-1}^* - q_d^* > 0$ (with $t_0^* = 0$ if $d = 1$). As in the proof of Theorem 1, we consider shifting units from the $(d+1)$-th dispatch to the d-th. Let $\sigma = \min\{\varepsilon, q_{d+1}^* - q_{\min}\}$ if $d \leq D^* - 2$, or $\sigma = \varepsilon$ if $d = D^* - 1$. If $\sigma > 0$, we can redefine the two dispatches as $(t_d^*, q_d^* + \sigma)$, $(t_d^* + f(q_d^* + \sigma), q_{d+1}^* - \sigma)$. Alternatively, suppose $d \leq D^* - 2$, $\sigma = 0$. Then (C4) implies $q_{d+1}^* = \cdots = q_{D^*-1}^* = q_{\min} \geq q_{D^*}^*$. In this case, we can combine the quantities in the last two dispatches into a single dispatch: $q_{D^*-1}^* + q_{D^*}^* \leq 2q_{\min}$, while (5c), (C2) and (C3) imply $t_{D^*-1}^* > N$. So we can replace $(t_{D^*-1}^*, q_{D^*-1}^*)$, $(t_{D^*}^*, q_{D^*}^*)$ with a single dispatch $(t_{D^*-1}^*, q_{D^*-1}^* + q_{D^*}^*)$. In either case, the new policy will remain feasible and optimal while still satisfying (C2) and (C3); if it no longer satisfies (C4), we can proceed as before so it again satisfies this conditions. After this operation, the policy satisfies (C2) through (C4) and either also satisfies (C1), or we have increased the index of the first dispatch that does not satisfy (C1), or we have decreased the total number of dispatches used in the optimal policy. As with (C4), we can iteratively perform these policy transformations as necessary until the solution satisfies (C1), which completes the proof.

7.4 Proof of Corollary 6

Theorem 5 implies that the search for an optimal dispatch policy for a single-vehicle SDD problem instance satisfying the stated conditions can be restricted to policies satisfying (C1), (C2), and (C3). By (C2) and (C3), such an optimal policy has an objective value $T-\alpha^*$, where α^* is its time of first dispatch. Thus, over all feasible dispatch policies satisfying (C1), (C2), and (C3), the one maximizing its time of first dispatch must be optimal.

Feasible dispatch policies with D dispatches satisfying (C1) and (C2) have a time of first dispatch in $[\alpha_D, \alpha_{D-1})$, and the α_D values are decreasing in D, so we can further restrict our search by minimizing the number of dispatches. Finally, if two feasible dispatch policies satisfying (C1), (C2), and (C3) use the same number of dispatches, the one with the later time of first dispatch has a lower objective value. Thus, Algorithm 1 returns the optimal time of first dispatch.

7.5 Proof of Theorem 8

Assume the instance satisfies the minimum dispatch size, sufficient gap and sufficient processing speed conditions. Then Lemma 4 implies the instance is feasible for a single vehicle. In the hybrid policy, if the $(m-1)$-th vehicle departs at time t_{m-1}, then the first $m-1$ vehicles serve $\lambda t_{m-1} = t_{m-1}$ orders, leaving the remaining $N_1 := N - t_{m-1}$ orders to be served by the m-th vehicle; since all three conditions still hold in this reduced problem, the last vehicle can feasible serve these remaining orders.

The only potential concern is whether any of the first $m-1$ dispatches serve a quantity smaller than $q_{\min}$, which would violate the minimum dispatch size condition. However, this is impossible, as all of these dispatches depart before time N and return at time T; this implies a dispatch time for each of these dispatches greater than $T-N \geq f(2q_{\min})$, by the sufficient gap time condition, which means each dispatch serves at least $2q_{\min}$ orders.

For the second part of the Theorem, assume $f(n) = bn + c\sqrt{n}$. Given that the heuristic uses $m-1+D_m$ dispatches in total, the Many-Vehicle Policy must also use no more than $m-1+D_m$ vehicles. Let z^{many} represent the objective value of the Many-Vehicle Policy, z^{m-1+D_m} represent the objective value of the policy using $m-1+D_m$ vehicles and z^m represent the objective value of the optimal policy where m vehicles are used. Finally, let z^{HmD_m} represent the objective value of the heuristic policy, using m vehicles where the last vehicle is dispatched D_m times. Thus, we know that $z^{many} = z^{m-1+D_m} \leq z^m \leq z^{HmD_m}$ and our objective is to show that $z^{HmD_m} \leq \frac{m-1+D_m\sqrt{D_m}}{m-1+D_m} z^{many} \leq \frac{m-1+D_m\sqrt{D_m}}{m-1+D_m} z^m$.

Let z_1^{many} represent the total dispatch time in the Many-Vehicle Policy of using the first $m-1$ vehicles, while z_2^{many} represents the remaining dispatch time from the m-th dispatch onward. Similarly, let $z_1^{HmD_m}$ represent the total dispatch time in the heuristic of using the first $m-1$ vehicles, while $z_2^{HmD_m}$ represents the remaining dispatch time from the last vehicle, which dispatches D_m times from the depot. We have that $z_1^{many} = z_1^{HmD_m}$ by construction of the heuristic. If we assume that some N_1 orders remain to be dispatched after each policy identically dispatches the first $m-1$ vehicles, then by concavity we maximize the total dispatch time of the last vehicle's D_m dispatches if they are all equal, each serving $\frac{N_1}{D_m}$ orders; this has a total dispatch time of $D_m f(\frac{N_1}{D_m})$. The most effective (though possibly infeasible) policy for the remaining N_1 orders would be to serve them together, incurring a dispatch time of $f(N_1)$. Therefore, $f(N_1) \leq z_2^{many} \leq z_2^{HmD_m} \leq D_m f(\frac{N_1}{D_m})$. As we have $f(n) = bn + c\sqrt{n}$, it follows that $D_m f(\frac{N_1}{D_m}) \leq \sqrt{D_m} f(N_1)$. This implies that $z_2^{HmD_m} \leq \sqrt{D_m} z_2^{many}$.

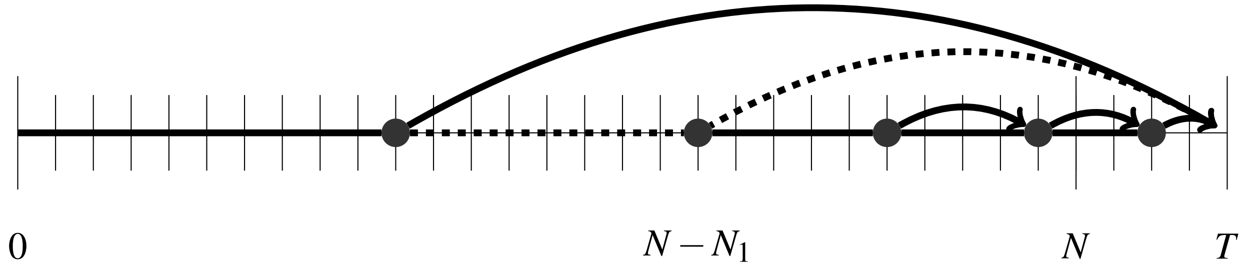

Figure 8: Visual representation of the heuristic with $m=3$ and $D_m=3$. The first two dispatches account for the cost in $z_1^{HmD_m}$ while the last three dispatches account for the cost in $z_2^{HmD_m}$

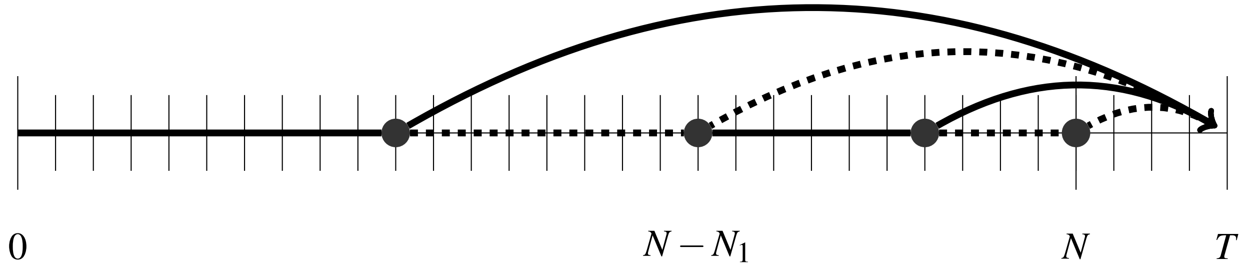

Figure 9: Visual representation of many-vehicle policy over the same problem instance as in Figure 8. Note that only 4 vehicles are used. The first two dispatches account for the cost in z_1^{many} while the last two dispatches account for the cost in z_2^{many}

Because the dispatch sizes are non-increasing in the MVP, the time of every one of the first $m-1$ dispatches, which sum to z_1^{many}, is greater than or equal to that of any of the remaining dispatches, which sum to z_2^{many}. Thus, $z_1^{many} \geq \frac{m-1}{m-1+D_m} z^{many}$, which implies that $z_2^{many} \leq \frac{D_m}{m-1+D_m} z^{many}$. Combining the two, $z^{HmD_m} = z_1^{HmD_m} + z_2^{HmD_m} \leq z_1^{many} + \sqrt{D_m} z_2^{many} = z^{many} + (\sqrt{D_m}-1) z_2^{many} \leq z^{many}(1+\frac{D_m(\sqrt{D_m}-1)}{m-1+D_m}) = z^{many}(\frac{m-1+D_m\sqrt{D_m}}{m-1+D_m}) \leq z^{m}(\frac{m-1+D_m\sqrt{D_m}}{m-1+D_m})$, as desired.

7.6 Proof of optimality for the Generalized MVP

In the case that $N' \leq Q$, we can transform the problem instance of N', N, f, T to $\bar{N}' = 0, \bar{N} = N + N', f, \bar{T} = T + N'$. This transformed instance serves the same amount of total orders and is subject to the same gap time between the end of service day and the order cutoff time as in the original problem instance. We know that we can use the MVP to solve for the optimal policy in such an instance. Fix this optimal policy as $\{(\bar{t}_d^*, \bar{q}_d^*)\}_{d=1}^{D^*}$. It must be the case that $\bar{t}_1^* = \bar{q}_1^* \geq N'$ as $N' + f(N') \leq N' + f(Q) = N' + T = \bar{T}$. It follows that we can subtract N' time units from each $\bar{t}_d^*$ and to obtain the policy $\{(\bar{t}_d^* - N', \bar{q}_d^*)\}_{d=1}^{D^*}$ which is feasible in the original problem instance. This policy must also be optimal in the original instance as it is optimal in the transformed instance, which is a relaxation of the original instance. Thus in the case that $N' \leq Q$ the generalized MVP is an optimal policy as it exactly prescribes the same dispatch sizes of the transformed policy.

In the case that $N' > Q$, it follows from Lemma 3 that any feasible dispatching policy that does not leave with as many as possible orders of size Q at time $t=0$, as prescribed in the generalized MVP, can be transformed into one which does without increasing the total dispatch time of the policy. After these dispatches occur at time $t = 0$ some $0 \leq \bar{N}' < Q$ orders will remain at the depot at the start of the service day. We have already shown that the generalized MVP is optimal in the case of $N' \leq Q$ so the remaining orders are also served optimally, and thus we are done.

7.7 Proof of Theorem 12

For cutoff times in the interval of $(N_i, N_{i+1}]$ we know that exactly $i+1$ vehicles will be used in the MVP, and that $g(N) = f(N_1) + f(N_2 - N_1) + \cdots + f(N_i - N_{i-1}) + f(N - N_i)$ is concave in N (within the interval). It follows that $\pi(N)$ on $(N_i, N_{i+1}]$ is convex and is therefore maximized at one of its endpoints. Thus to maximize $\pi(N)$ over $N \in [0, U]$ we only need to consider the breakpoints of $N_1, N_2, \ldots, N_U$ as well as the endpoints of 0 and U. This completes the proof.

7.8 Proof of Proposition 13

We split this proof into cases. Assume we are given a U such that $0 \leq U \leq T - f(2q_{\min})$ and $i_{max} \leq 2$. Suppose $U = 0$, then trivially we have $N^* = 0$. So, assume that $U > 0$, and therefore $2 \geq i_{max} \geq 1$.

Assume that $i_{max} = 1$. Then the range $0 \leq N \leq U$ can be partitioned as $N \in \{0\} \cup (0, U]$. We know from Theorem 5 that given $N \in (0, U]$, the optimal dispatch policy is given by a single dispatch of size N at cost $f(N)$. Therefore, $\pi(N)$, is a convex function over the interval $(0, U]$. Thus, either $N = 0$, or $N = U$ will maximize the profit function.

Now assume that $i_{max} = 2$. Then the range $0 \leq N \leq U$ can be partitioned as $N \in \{0\} \cup (0, \bar{N}_1] \cup (\bar{N}_1, U]$. We know from Theorem 5 that given $N \in (\bar{N}_1, U]$, the optimal dispatch policy can be fully described by the time of first departure α_N. Additionally we know $g(N) = T - \alpha_N$, and $\alpha_N + f(\alpha_N) + f(N - \alpha_N) = T$. Which means we can write $N = \alpha_N + f^{-1}(T - \alpha_N - f(\alpha_N))$. Thus N can be written as a convex function of α_N, and thus $g(N)$ is a concave function with respect to N. Thus $\pi(N)$ is a convex function over the interval $(\bar{N}_1, U]$. From before we also have that $\pi(N)$ is a convex function over the interval $(0, \bar{N}_1]$. Thus the solution to the profit maximization function can be found at $N = 0$, $N = \bar{N}_1$, or $N = U$. Thus, the claim is proven.

7.9 *A Posteriori* Formulations

We use the following integer programming formulations to compute *a posteriori* solutions in our computational study.

Parameters

- Node set: order locations $L = \{1, \ldots, n\}$, depot 0
- Arc set: ordered pairs of nodes, $a = (i, j)$.
- Travel time: τ_{ij}, $i, j \in L \cup 0$, includes depot setup time and order service time as necessary
- Release time: $r_i \geq 0$, $i \in L$, the time when order i is ready
- Deadline: T
- Fleet size or number of routes: K ($K = 2$ in our experiments)

Decision Variables

- x_{ij}^k: indicates if vehicle/route k goes from i to j
- d_k: departure time of vehicle/route k

The many-vehicle formulation is then given by

$$
\begin{aligned}
\min_{d,x} \quad & \sum_{k=1}^{K} \sum_{i,j} \tau_{ij} x_{ij}^k && && \text{(6a)} \\
\text{s.t.} \quad & \sum_{a \in \delta^+(i)} x_a^k = \sum_{a \in \delta^-(i)} x_a^k, && i \in L \cup 0, \quad k = 1, \ldots, K && \text{(6b)} \\
& \sum_{k=1}^{K} \sum_{a \in \delta^+(i)} x_a^k = 1, && i \in L && \text{(6c)} \\
& \sum_{a \subseteq S} x_a^k \leq |S| - 1, && S \subseteq L, \quad k = 1, \ldots, K && \text{(6d)} \\
& d_k + \sum_{i,j} \tau_{ij} x_{ij}^k \leq T, && k = 1, \ldots, K && \text{(6e)} \\
& d_k - r_i \sum_{a \in \delta^+(i)} x_a^k \geq 0, && i \in L, \quad k = 1, \ldots, K && \text{(6f)} \\
& d_k \geq 0, \quad x_{ij}^k \in \{0, 1\}. && && \text{(6g)}
\end{aligned}
$$

In the formulation, (6b) ensures flow balance; (6c) requires each order to be served by a vehicle; (6d) eliminates subtours; (6e) establishes a route duration limit, so each vehicle returns by T; (6f) prevents a vehicle serving i from departing the depot before the order is ready.

In the single-vehicle case, assume routes are indexed in order of departure. For all routes except K, we replace (6e) with

$$
d_k - d_{k+1} + \sum_{i,j} \tau_{ij} x_{ij}^k \leq 0, \qquad k = 1, \ldots, K - 1. \tag{7}
$$

This ensures route k finishes before $k+1$ begins, so one vehicle can perform all routes.