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

June 7, 2019

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 a service 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 for finding best configurations of these policies that require only simple computations. 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 also validate the predictions of the model and policies for example problem instances against a detailed operational model and demonstrate that our simple approximation model can predict the average number of orders served and minutes driven to within about 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% [5]. 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 [9]. Amazon, one of the current SDD industry leaders, began offering an SDD fulfillment option in October 2009 across 7 major U.S. cities [14]. 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 [15, 16]. 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 [20, 21]. 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. In addition, 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 [10, 17]. These order cut-off 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. [19, 21, 20, 28, 31]. 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 pre-emptively, 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 [24]. Since SDD systems face tight delivery deadlines and comparatively low order volume, time (not vehicle capacity) tends to be the constraining resource and thus is an important focus of our models.

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 [13], with some canonical results dating back several decades [3, 11, 23]. 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 [29], and can also be calibrated with empirical observations (see, e.g., [18]). 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.
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 multiple vehicles are assigned but are 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. We also show via a computational study that the performance metrics of these optimal policies as predicted by our continuous approximation model closely match more detailed simulations.

(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 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 our computational validation of 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 cut-off 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, 6, 30]. Additionally, dynamic vehicle routing problems such as [4, 22] broadly encompass SDD modeling.

We now survey some operational SDD models from the literature. One such problem is the dynamic dispatch waves problem (DDWP) [20, 21]. 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 [21] 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 [20] the same authors model the DDWP in 2-dimensions using an integer-programming approach to solve the problem’s 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 [21], optimal dispatch policies for an SDD variant (1-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) [31]. The authors provide a general framework for SDD modeling, using a fleet of delivery vehicles of known size, a fixed cut-off 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 [25, 26, 27, 28], 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 [25] the author performs an analysis of the relationship between fleet size and delivery capacity of the system. The authors in [31] 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 for modeling an SDD system. For a recent survey on continuous approximation models in freight logistics, see [13]. 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. [11] 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 [18] 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. Although our study is the first that applies continuous approximations in SDD, there are other applications of these techniques in urban logistics and the last mile. For example, [12] studies the efficiency of urban commercial vehicles using continuous approximations, and [8] apply the techniques to study the use of drones in last-mile delivery. [29] study an operational model in urban last-mile delivery and use continuous approximations withing an approximate dynamic programming heuristic policy.

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 depot’s 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 $\lambda$ 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 depot begins the service day with no orders requiring delivery, and thus the total number of orders that accumulate is $N$. Depending on the context, we may refer to $N$ as the order cutoff time or the number of orders to serve. We later discuss extending the model to include orders at the start of the service day.

**Vehicle Restrictions:** All vehicles must return to the depot by time $t = T$. We do not explicitly constrain the space 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.

**Routing 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 $f : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ with the following properties:

1. $f(0) = 0$,
2. $f(n)$ is concave and increasing,
3. there exists $q^* < \infty$ such that $f(x) \leq x$, for all $x \geq q^*$.

These properties indicate that serving more orders should take more time, but that there will be a gain in marginal time efficiencies when consolidating orders. The last constraint ensures that the system can “keep up” with the orders; that is, the time required to serve a number of orders is smaller than the time it takes for them to accumulate when the number is large enough.

An example routing time function is of the form $f(n) = a + bn + c\sqrt{n}$ for $n > 0$, where $a \ll T - N$, $b < 1$. This function includes a constant setup time at the depot, a service time per order, and a BHH routing time [3] between orders; [13] 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 $N$ orders while minimizing total routing time incurred by all vehicles. We formally 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, $q_d$. A set of dispatches $\{(t_d, q_d, i_d)\}_{d=1}^D$ is feasible for our model if the following conditions are satisfied:

\[
\begin{align*}
\sum_{d=1}^D q_d &= N, \\
q_d &\geq 0 \quad \forall d, \\
t_d + f(q_d) &\leq T \quad \forall d, \\
t_d + f(q_d) &\leq t_\delta \quad \forall d, \delta \text{ s.t. } i_d = i_\delta, \; d < \delta \\
t_d &\geq 0 \quad \forall d, \\
\sum_{\delta=1}^d q_\delta &\leq t_d \quad \forall d
\end{align*}
\]

Our problem is to choose $D \geq 1$ dispatch tuples that minimize $\sum_{d=1}^D f(q_d)$, subject to (1a)-(1f). 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 is performing 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 focus on two important families of instances of our SDD model, namely 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 [20, 21]. For the general case with a finite fleet size, we discuss some of the challenges and possible strategies for analysis in our conclusions, and leave them for future work.

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 theorem’s 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 vehicles case. Furthermore, if the number of vehicles used by MVP is $m^*$, the total routing time used by this policy provides a lower bound for the model’s 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 $t_1 + f(t_1) = N$, and if more than two vehicles are required then $(t_2 - t_1) + f(t_2 - t_1) = N - t_1$, 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.

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 routing time function is $f(n) = 2.15\sqrt{n} + .13n$, roughly equivalent to a routing time approximation (Manhattan distances [13], with vehicles traveling at 25 miles per hour), plus a service time of 1 minute per order. The MVP returns the optimal solution,

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

with 272.06 total minutes of routing 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 below in our analysis, which tells us that unbalanced dispatch sizes are preferable; this follows directly from the concavity of $f(\cdot)$.

**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 $q_1^* \leq t_1^* \leq N - f(q_1^*)$, 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.

The previous example shows that the model requires additional constraints to produce meaningful answers. We now introduce two additional conditions that address these concerns.

**Minimal Dispatch Size:** Any feasible solution $\{(i_d, q_d, i_d)\}_{d=1}^D$ must satisfy $q_d \geq q_{\min}$ for all $d < D$. The parameter $q_{\min}$ is chosen by the operator such that $q_{\min} \geq q^*$. 

**Guaranteed Feasibility:** The parameters $T, N, q_{\min}$ must satisfy $N \leq T - f(2q_{\min})$.

A minimal dispatch size is justifiable in practice; a retailer may not want to pay to send out a driver and a delivery vehicle if there are only a small handful of orders to serve. We bound this minimal dispatch size by $q^*$, so whenever the delivery vehicle is dispatched, it is guaranteed to arrive back at the depot to find a number of unserved orders that is less than or equal to the number at the depot prior to this dispatch; this is desirable since all orders must be served by the end of the day. The last dispatch is not subject to this constraint, since it must serve all remaining orders, regardless of their number. Our second additional constraint places an upper bound on the maximum cutoff time. Unlike the many vehicle case, which is feasible under most reasonable circumstances, the single vehicle case may not be feasible for a given $q_{\min}$. The additional condition is sufficient for feasibility, as the vehicle can simply perform as many dispatches with $q_{\min}$ orders as necessary to serve all orders; the next lemma formalizes this argument and is proven in the appendix.
Lemma 4. An SDD problem instance with a single vehicle and parameters satisfying the Minimal Dispatch Size and Guaranteed Feasibility conditions has a feasible solution.

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

Theorem 5. Assuming the Minimal Dispatch Size and Guaranteed Feasibility conditions hold, there exists an optimal dispatch policy for the SDD problem with one delivery vehicle such that

1. each dispatch takes all available, unserved orders at the depot at the time of dispatch,
2. after the first dispatch, the vehicle never waits at the depot again, and
3. if the vehicle is dispatched more than once, the last dispatch arrives back at the depot at exactly time $t = T$.

Theorem 5 implies that an optimal policy can be described by a single parameter, $t_1$. By (1), we have $q_1 = t_1$, and then (2) yields $t_2 = t_1 + f(q_1)$. Applying this reasoning recursively, $q_2 = t_2 - t_1$ and so on, continuing until the last dispatch covers all the remaining orders, leaving at or after time $t = N$. This structure implies that an optimal dispatch policy can be computed via an optimization model over the single variable, $\alpha = t_1$, the time of first dispatch. For the non-trivial case in which the vehicle must be dispatched more than once, the theorem implies we can solve for the optimal $\alpha$ via an iterative root-finding algorithm. Informally, the algorithm works by fixing a number of dispatches and attempting to find a feasible $\alpha$, which is a root of a routing time function. If a root is found the algorithm terminates, otherwise the number of dispatches increases by one and the process repeats.

Algorithm 1 Calculating the optimal time of first dispatch, $\alpha^*$

1. Set $d \leftarrow 1$, $\alpha_1 \leftarrow N$, $found \leftarrow FALSE$
2. Define $f^{\delta}(\alpha) := f(f(\cdots f(\alpha))))$, $(\delta$ times$)$
3. if $N + f(N) \leq T$ then
   4. Set $\alpha^* \leftarrow N$
   5. Set $d^* \leftarrow 1$
   6. Set $found = TRUE$
   7. else
   8. while $found = FALSE$ do
   9. Set $d \leftarrow d + 1$
10. Define $h_d(\alpha) \leftarrow \alpha + \sum_{i=1}^{d-1} f^{i}(\alpha)$
11. Set $\alpha_d \leftarrow \arg\min_{\alpha \geq 0} \{ \alpha \} \text{ subject to } h_d(\alpha) = N$
12. Set $\alpha^* \leftarrow \arg\max_{\alpha \geq 0} \{ \alpha \} \text{ subject to } h_d(\alpha) \leq T,\ \alpha \in [\alpha_d, \alpha_{d-1})$
13. if $\alpha^*$ exists then
   14. Set $found = TRUE$
   15. Set $d^* \leftarrow d$
16. end if
17. end while
18. end if

Example 6. Consider the same instance as in Example 2 that is, $N = 75, T = 90, f(n) = 2.15\sqrt{n} + 0.13n$. Now suppose the fleet has a single delivery vehicle, and set $q_{\min} = 12$. This set of model parameters satisfy
the Minimal Dispatch and Guaranteed Feasibility conditions, so we can use Theorem 5 and our root finding algorithm to compute an optimal dispatch policy,

\[ t_1 = 54.65, \quad q_1 = 54.65, \quad i_1 = 1, \]
\[ t_2 = 77.65, \quad q_2 = 20.35, \quad i_2 = 1, \]

with a cost of 282.74 minutes; see Figure 2. In this example, we can see that by decreasing the fleet from two to one vehicles, we increase the total routing time by less than 4%. Recall also that there is no reduction in total routing time possible by using more than two vehicles.

![Figure 2: Visual representation of optimal dispatch policy for Example 6.](image)

### 4 Model Applications

The discussion in Examples 6 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. [7]. 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 routing time function \( f(n) = a + bn + c\sqrt{n} \). Suppose we partition the service 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 routing time function of the form \( h(n) = a + bn + c\sqrt{n/m} \), 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 routing time for all vehicles would be

\[ m \ast h(N/m) = am + bN + c\sqrt{N}; \]

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 7.** 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 routing time function as \( f(n) = 1.88 + .25n + 2.15\sqrt{n} \), roughly equivalent to a routing time
approximation (Manhattan distances [18], 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

\[ t_1 = 53.87, \quad q_1 = 53.87, \quad i_1 = 1, \]
\[ t_2 = 70.30, \quad q_2 = 16.43, \quad i_2 = 2, \]
\[ t_3 = 75, \quad q_3 = 4.70, \quad i_3 = 3, \]

with 428.37 total minutes of routing 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 system’s 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 one-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 service day’s start.

In the many vehicle case, the approach 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. The generalized MVP’s optimality proof follows directly from Theorem 1 and Lemma 3.

**Generalized MVP** At time \( t = 0 \), dispatch as many vehicles as possible each carrying \( Q \) 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 \( f(\cdot) \) 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 guaranteed feasibility condition from section 3.2, we can solve the one vehicle problem for any instance of \( N' > 0 \). Practically, this involves increasing the gap between the order cut-off time and the end of the service day.

**Generalized Guaranteed Feasibility**: If the parameters \( T, N, N', q_{\text{min}} \) satisfy \( N + N' \leq T - f(2q_{\text{min}}) \), the instance is feasible with a single dispatch vehicle.

Assume the generalized guaranteed feasibility condition holds for parameters \( T, N, q_{\text{min}}, N' \). As before, define and solve an augmented problem with \( \bar{N} = N' + N \), \( \bar{T} = N' + T \), and \( \bar{N}' = 0 \). In the case that one dispatch is optimal it is necessary 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 at 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 routing 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 routing time. It follows that $\bar{q}_i^* \geq N'.$ Therefore, in all cases it is true that $\bar{q}_i^* \geq N'$, 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 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 $T$. The manager could consider either reducing $N$ so the system requires only one vehicle, or increasing $N$ to serve more orders with this vehicle and increase its utilization.

For this discussion, we fix $T, f(\cdot), q_{\min}$ and allow $N$ to vary. We assume that earned revenue from orders served is proportional to $N$ with constant $\beta$, and routing costs are proportional to the routing time of the optimal policy, denoted by $g(N)$. Without loss of generality we scale $\beta$ so we can compare revenue against cost. Therefore, the profit maximizing cutoff time is given by

$$\max_{0 \leq N < T} \pi(N) = \beta N - g(N). \quad (2)$$

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. The proof of Theorem 8 follows directly from Theorem 1 and the concavity of $f(\cdot)$, which implies that $g(N)$ is piecewise concave.

**Theorem 8.** For the many vehicle case, in the optimal solution $N^*$ of (2), the MVP policy’s solution has all dispatches returning exactly at time $T$.

Let $N_i$ be the order number 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 $\Delta_i$ uniquely solves $\Delta_i + f(\Delta_i) = T - N_{i-1}$ for all $i$. With a slight abuse of notation, we can denote $\pi(i) = \pi(N_i)$ as the profit obtained from completely utilizing $i$ vehicles; this profit is convex in $i$, and therefore we can calculate the optimal $N^*$ by iteratively calculating each $\pi(N_i)$ and stopping when this quantity decreases.

Now we analyze the one 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 9.** 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) subject to $N \leq U$ satisfies $N^* \in \{0, \bar{N}_1, U\}$.

The proposition’s proof 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 6, and calculate an optimal value for the cutoff time $N$.

**Example 10.** Consider the same instance as in the previous examples with $T = 90, f(n) = 2.15\sqrt{n} + .13n$, and $q_{\min} = 12$. Suppose $\beta = 0.80$.

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 costs 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 $\pi(N_3) = 26.22$. Thus, the optimal order cutoff time is $N^* = N_2$, with an optimal dispatch policy of $\{t_1 = 64.38, q_1 = 64.38, i_1 = 1\}, \{t_2 = 79.62, q_2 = 15.24, i_2 = 2\}$. 

11
For the one 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 3 and 4 plot $\pi(N)$ for this instance in the many and single vehicle cases, respectively.

**Figure 3**: Profit function with respect to cut-off time, many vehicle case

**Figure 4**: Profit function with respect to cut-off time, one vehicle case
5 Computational Validation

Our dispatch model uses continuous approximations in order to preserve simplicity and transparency of analysis. We next demonstrate empirically that the predictions of this model under the studied dispatching policies closely match predicted implementations of these policies using much more detailed operational models.

Given an SDD model instance, consider a detailed operational counterpart that has order arrivals given by a homogeneous Poisson process with unit rate and order locations that are sampled uniformly from the service region. To determine the actual routing time for a dispatch, we compute an optimal TSP tour for the given set of delivery orders using a standard integer programming formulation. We programmed our experiments in MATLAB R2016b and ran them in our departmental computing cluster, which uses HTCondor.

5.1 Many Vehicle Policy

For this experiment, we used an 8-by-8 mile square service region with a centrally located depot. SDD orders arrive according to a Poisson process, with rate of one order per six minutes, and the service day lasts 12 hours. We use Manhattan distances and assume a vehicle speed of 25 miles per hour. Additionally, there is a two minute service time per order as well as a five minute dispatch setup time.

Based on this problem setup, we first define a routing time approximation function. The BHH constant used for TSP tours with a Manhattan metric was empirically estimated as 0.8943 in [18]. However, this constant is asymptotic in the number of stops. Following [2, 18], we randomly generated instances of 30, 60, and 90 customers, then calculated an optimal TSP tour (including the depot location) and its duration for each instance. Figure 1 reports the average ratio of tour length to the square root of the number of stops, which provide us with our own empirically-estimated BHH constants for Manhattan distances.

<table>
<thead>
<tr>
<th>Tour size</th>
<th>Average ratio</th>
<th>Std. dev. of ratio</th>
<th>Sample size</th>
</tr>
</thead>
<tbody>
<tr>
<td>30</td>
<td>1.0363</td>
<td>.0625</td>
<td>1000</td>
</tr>
<tr>
<td>60</td>
<td>0.9917</td>
<td>.0402</td>
<td>1000</td>
</tr>
<tr>
<td>90</td>
<td>0.9718</td>
<td>.0302</td>
<td>1000</td>
</tr>
</tbody>
</table>

Table 1: Empirical BHH constant for varying tour sizes.

Based on these results, and to be conservative in our routing time estimates, we use the constant 1.0363 in our routing function approximation. After normalizing the system to have a demand rate of one order per time unit, we arrive at a problem instance with $T = 120, f(n) = .83 + .33n + 3.32\sqrt{n}$.

Following the discussion in Section 4.3, we choose the cutoff time so that three vehicles are dispatched fully utilized, which corresponds to $N = 104.7$. The optimal solution’s dispatch quantities are 68.70, 25.30, and 10.70, and it requires 554 minutes in total routing time. (As an example, this $N$ would be optimal for the profit maximizing problem (2) with $\beta = 10$).

We now describe our operational benchmark for the many vehicle policy. As customer orders arrive according to the Poisson process, we solve a TSP on all currently unserved orders. A vehicle is dispatched to deliver the orders when the dispatch setup, service time for accumulated orders, and the duration of the optimal TSP tour added together exactly match the remaining time in the service day, i.e. each vehicle returns at time $T$. Once the third vehicle is dispatched, we stop accepting orders, which may occur before or after the nominal cutoff time of $N$; this represents a dynamic modification of the cutoff time at the operational level. We simulate this process 1,000 times.

Our tactical model predicts 104.70 orders served on average, with routing time of 554 minutes. Our simulation results for the operational benchmark of the many vehicle policy serve an average of 103.37
orders, with standard deviation of 9.02 orders; these are served with an average routing time of 549.63 minutes, with a standard deviation of 32.96 minutes. Thus, for this experimental instance, our tactical model predicts the expected number of orders served and the expected total routing time within roughly 1%. The histogram in Figure 5 depicts the number of orders served across the 1,000 simulated instances.

![Simulated total orders served (many vehicles)](image)

Figure 5: Order delivery results for many vehicle experiment.

### 5.2 Single Vehicle Policy

For the single vehicle case, we use a 6-by-6 mile square service region with a centrally located depot. SDD orders arrive according to a Poisson process, with rate of one order per six minutes, and the service day lasts 12 hours. The cut-off time here occurs at 9 hours. As before, we use Manhattan distances and assume a vehicle speed of 25 miles per hour. Additionally, there is a two minute service time per order as well as a five minute dispatch setup time.

For determining the routing time approximation, we use the same BHH routing constant from the previous experiment, 1.0363. After normalizing the system to have a demand rate of one order per time unit, we arrive at a problem instance with \( N = 90, T = 120, f(n) = .83 + .33n + 2.49\sqrt{n} \). Using our single vehicle algorithm, we obtain an optimal policy consisting of two dispatches of sizes 55.10 and 34.90, with total routing time of 389 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 \( \alpha \), we want to operationally mimic the equation \( \alpha + f(\alpha) + f(N - \alpha) = T \), which determines the dispatch time in the tactical model. As orders arrive, we iteratively solve a TSP for the unserved orders and track \( \tau \), the sum of setup time, service time for accumulated orders, and optimal tour duration. We dispatch the vehicle at the time \( \alpha \) satisfying \( \alpha + \tau + f(N - \alpha) = T \). 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, we again solve the TSP for these locations, 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 situations with more dispatches.

We again simulated 1,000 realizations and implemented the operational benchmark policy. We observed an average of 89.54 orders served with a standard deviation of 7.43 orders, and the total routing time took an average of 383.93 minutes with a standard deviation of 22.86 minutes; the tactical model again predicts both within about 1%. Figure 6 displays a histogram of the number of orders served in the simulations.
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 delivery vehicle routing time, we are able to derive simple and transparent optimal solutions for the model that describe the SDD system’s behavior on average; our empirical validation shows that the model can indeed predict the system’s 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 further research. An immediate issue is the study of the model for the case of a finite fleet of size greater than one. Our initial attempts show that even when the fleet has two vehicles but two dispatches aren’t enough, the model becomes quite complex. The model does preserve some of the structure from the single-vehicle case – for example, once a vehicle starts deliveries, it continues until the end of the service day, and every departing dispatch takes all unserved orders – but this structure does not readily result in a simple algorithm.

Another interesting direction 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.

References


7 Appendix

7.1 Proof of Theorem 1

Fix any non-trivial (strictly positive dispatch sizes) optimal set of dispatches as \( \{(t^*_d, q^*_d, i^*_d)\}_{d=1}^{D^*} \). Without loss of generality we can assume that each dispatch in this solution is made with a unique vehicle. Additionally, we can assume that \( t^*_d = q^*_d \) and \( t^*_{d-1} = t^*_{d-1} + q^*_d \), for \( d = 2, 3, ..., D^* \), which is to say the vehicles do not idly wait at the depot after they have enough orders to dispatch. Because of the property that \( f(\cdot) \) is increasing in the number of orders served we can w.l.o.g. assume that consecutive dispatch sizes are non-increasing. Say this was untrue for a pair of dispatches, namely \( (t^*_d, q^*_d, i^*_d) \) and \( (t^*_{d+1}, q^*_{d+1}, i^*_{d+1}) \). Then from the above assumptions it must be true that \( t^*_{d+1} - t^*_d = q^*_{d+1} \). Thus we can "swap" these dispatches and arrive at a new feasible solution of equal cost: \( (t^*_d + q^*_{d+1} - q^*_{d+1}, t^*_{d+1}, i^*_{d+1}) \), and \( (t^*_{d+1}, q^*_{d+1}, i^*_{d+1}) \). Repeating this process will induce the desired ordering on the consecutive dispatch sizes.

With our above assumptions on the fixed optimal dispatch policy, we now claim that the first \( D^* - 1 \) dispatches arrive back at the depot at exactly time \( t = T \). If this claim is true, then each of the first \( D^* - 1 \) dispatch vehicles will take all of the realized orders waiting at the depot at the time of dispatch and return at exactly time \( t = T \), thus proving that the MVP is optimal.

For the sake of contradiction, assume there exists at least one of the first \( D^* - 1 \) dispatches in the optimal solution such that the corresponding vehicle arrives back to the depot before time \( T \). Observe the first such dispatch, indexed by \( d' \), and the subsequent dispatch \( d' + 1 \). These dispatches serve \( q^*_{d'} \) and \( q^*_{d'+1} \) orders respectively. There must exist an \( \varepsilon > 0 \) such that the dispatches \((t^*_d + \varepsilon, q^*_d + \varepsilon, i^*_d)\), with the exception of the case where it is feasible to choose \( \varepsilon = q^*_{d'+1} \), that is, \( t^*_d + q^*_d + f(q^*_d) \leq t^*_{d+1} \), remove the first \( d' \) dispatch entirely, and then repeat the argument. Thus, either the \( d' \) dispatch will now arrive back at \( t = T \), and we can re-order the remaining dispatches and repeat the argument with one less dispatch in question, or we can remove the \( d' + 1 \) dispatch entirely, and then repeat the argument. Therefore, by finite iteration, we can transform our previous optimal solution into one where all of the non-last dispatches arrive back at the depot at exactly time \( t = T \) or we can contradict that this policy was optimal to begin with.

Therefore the MVP is an optimal dispatch policy. Clearly, any problem that needs to be solved with fleet size less than \( D^* \) cannot possibly return better objective value than the MVP over \( D^* \) vehicles.

7.2 Proof of Lemma 4

Assume we are given an SDD problem with a singleton fleet, which satisfies the Minimal Dispatch Size and Guaranteed Feasibility conditions. Let \( D' = \text{floor}(N/q_{\text{min}}) \).

Assume \( D' \leq 1 \). Observe the dispatch policy \( (t_1 = N, q_1 = N, 1) \). Note that \( D' \leq 1 \) implies that \( N < 2q_{\text{min}} \). Because \( f(\cdot) \) is increasing we have \( f(N) < f(2q_{\text{min}}) \). Because \( N + f(N) < N + f(2q_{\text{min}}) \leq T \), we see that the final dispatch returns to the depot by the end of the service day, and the lemma is proven for \( D' \leq 1 \).

Assume \( D' \geq 2 \). Observe the dispatch policy \( (t_1 = q_{\text{min}}, q_1 = q_{\text{min}}, 1), (t_2 = 2q_{\text{min}}, q_2 = q_{\text{min}}, 1), ..., (t_{D'-1} = (D' - 1)q_{\text{min}}, q_{D'-1} = q_{\text{min}}, 1), (t_{D'} = N, q_{D'} = N - (D' - 1)q_{\text{min}}, 1) \). By construction, it must be the case that \( f(q_{\text{min}}) \leq q_{\text{min}} \). Thus, all of the first \( D' - 1 \) dispatches return to the depot before or just as \( q_{\text{min}} \) orders accumulate. Additionally by choice of \( D' \) it must be the case the \( q_{D'} \geq q_{\text{min}} \). Thus, the first \( D' - 1 \) dispatches all return to the depot before the next dispatch must leave. The last dispatch takes all of the remaining orders...
and thus it remains to be seen that the last dispatch will return by time \( t = T \). By the choice of \( D' \) we have the \( 2q_{\text{min}} > q_{D'} \). Because \( f(\cdot) \) is increasing we have \( f(q_{D'}) < f(2q_{\text{min}}) \). Because \( N + f(q_{D'}) < N + f(2q_{\text{min}}) \leq T \), we see that the final dispatch returns to the depot by the end of the service day, and the lemma is proven for \( D' \geq 2 \).

### 7.3 Proof of Theorem 5

Assume we are given an SDD problem with a singleton fleet, which satisfies the Minimal Dispatch Size and Guaranteed Feasibility conditions. By Lemma 4, the problem is feasible. If the policy \( (i_1 = N, q_1 = N, i_i = 1) \) is feasible, that is, \( N + f(N) \leq T \), then we are done. So, assume any feasible dispatch policy requires two or more dispatches from the vehicle. Therefore, we can fix an optimal policy, \( P_1 \), as \( \{(t^*_d, q^*_d, i^*_d)\}_{d=1}^{D^*_p} \) where \( D^*_p \geq 2 \).

In the case that \( P_1 \) violates (2) or (3), simply push all of the idle waiting time before the first dispatch. Additionally, push any remaining time at the end of the service day to be before the first dispatch. This new policy will still be feasible, and therefore be cost optimal. Furthermore, this transformed policy will now satisfy (2) and (3). Fix this policy, \( P_2 \), as \( \{(t^{**}_d, q^{**}_d, i^{**}_d)\}_{d=1}^{D^{**}_p} \).

Now we show that \( P_2 \) can be transformed into an optimal dispatch policy such that consecutive dispatch sizes are non-increasing. Assume that this is not already the case. Compute the smallest index, \( d' \), such that \( q^{**}_{d'} < q^{**}_{d'+1} \), and observe the dispatches \( (t^{**}_{d'}, q^{**}_{d'}, 1) \) and \( (t^{**}_{d'+1}, q^{**}_{d'+1}, 1) \). Because \( d' < D^{**} \), we know that \( q^{**}{d'} \geq q_{\text{min}} \) by the Minimum Dispatch Size condition. It follows that \( f(q^{**}_{d'}) \leq f(q^{**}_{d'+1}) \). Because \( t^{**}_{d'+1} = t^{**}_{d'} + f(q^{**}_{d'}) \) (no idle waiting), it must be the case that \( t^{**}_{d'+1} - t^{**}_{d'} \leq q^{**}_{d'} \). This means that at the time of the \( d' \) dispatch the number of realized orders at the depot decreases by \( q^{**}_{d'} \), and does not increase by more than \( q^{**}_{d'+1} \) orders by the time of the \( d' + 1 \) dispatch, when \( q^{**}_{d'+1} \) orders are delivered. Thus, it must be the case that there were enough realized orders at the depot to send \( q^{**}_{d'+1} \) orders for delivery at time \( t^{**}_{d'} \). This implies that the dispatches \( (t^{**}_{d'}, q^{**}_{d'+1}, 1) \) and \( (t^{**}_{d'} + f(q^{**}_{d'+1}), q^{**}_{d'+1}, 1) \) are feasible. This process can be iterated until all consecutive dispatch sizes are non-increasing; denote this policy as \( P_3 \), or \( \{(t^{***_d}, q^{***_d}, i^{***_d})\}_{d=1}^{D^{***}_p} \). Note that the re-ordering from \( P_2 \) to \( P_3 \) implies that \( P_3 \) also defines an optimal dispatch policy where (2) and (3) remain satisfied.

If (1) is satisfied for \( P_3 \), then we are done. Now assume (1) remains unsatisfied. Observe the first dispatch in which not all of realized orders at the depot are taken for delivery, indexed by \( d' \), and its consecutive dispatch \( d'+1 \). Let \( \varepsilon > 0 \) represent the amount of orders left at the depot when the \( d' \) dispatch occurs. Also define \( \sigma \) to be equal to \( \varepsilon \) in the case that \( d'+1 = D^{***} \), and equal to \( \min\{\varepsilon, q^{**}_{d'+1} - q_{\text{min}}\} \), otherwise. From here, it is feasible to use dispatches of \( (t^{***}_{d'}, q^{***}_{d'+1} + \sigma, 1) \) and \( (t^{***}_{d'} + f(q^{***}_{d'+1} + \sigma), q^{***}_{d'+1} + \sigma, 1) \) instead of \( (t^{***}_{d'}, q^{***}_{d'+1}, 1) \) and \( (t^{***}_{d'+1}, q^{***}_{d'+1}, 1) \). Name this re-balanced dispatch policy \( P_4 \). By Lemma 3, \( P_4 \) defines an optimal dispatch policy where (2) and (3) remain satisfied or we have contradicted the optimality of \( P_3 \). If \( \sigma = \varepsilon \), then \( P_4 \) is an optimal dispatch policy where all of the first \( d' \) dispatches leave the depot with all of the realized orders, and we can repeat the re-balancing argument with less dispatches in question for a re-ordered \( P_4 \) (P5). In the case that \( \varepsilon > \sigma = q^{**}_{d'+1} - q_{\text{min}} \), and \( d'+1 < D^{***} \) then we can use the arguments above to re-order the dispatch sizes in \( P_4 \) to be non-increasing. We can continually re-balance the \( d' \) dispatch with a newly ordered \( d'+1 \) dispatch until either a) dispatch \( d' \) will still be feasible, and therefore be cost optimal. Furthermore, this transformed policy will now satisfy (2) and (3). Fix this policy, \( P_2 \), as \( \{(t^{**}_d, q^{**}_d, i^{**}_d)\}_{d=1}^{D^{**}_p} \).

To summarize, the only way for (1) to remain unsatisfied for an optimal policy, \( P_6 \) or \( \{(t^{**}_d, q^{**}_d, i^{**}_d)\}_{d=1}^{D^{**}_p} \), which satisfies (2) and (3), is in the case that there is some first dispatch, indexed by \( d' \), in which not all of realized orders at the depot are taken for delivery and all of the remaining dispatches are of size \( q_{\text{min}} \) except for the last dispatch which may be smaller.

If it is the case for \( P_6 \), that \( t^{***}_{D^{**}_p - 1} < N \leq t^{***}_{D^{**}_p} \), then we can consolidate the orders to a single dispatch...
of \((N, q_{D^*_6-1}^{6*}, q_{D^*_6}^{6*})\) producing a transformed policy, P7. By the Guaranteed Feasibility condition, P7 is feasible as \(q_{D^*_6-1}^{6*} + q_{D^*_6}^{6*} \leq 2q_{\text{min}}\). By Lemma 3 it must be true that \(f(q_{D^*_6-1}^{6*}) \leq f(q_{D^*_6}^{6*} + f(q_{D^*_6}^{6*})\). If this inequality is strict then we violate P6 being optimal (contradiction). If this inequality is an equality then (3) must have been untrue of P6 (contradiction). Thus, it must be for P6 that \(N \geq t_{D^*_6-1}^{6*} \leq t_{D^*_6}^{6*}\).

Thus we have for P6, that \(N \geq t_{D^*_6-1}^{6*} \leq t_{D^*_6}^{6*}\). We can consolidate the orders to a single dispatch of \((t_{D^*_6-1}^{6*}, q_{D^*_6-1}^{6*} + q_{D^*_6}^{6*}, 1)\) producing a transformed policy, P8. By Lemma 3, P8 defines an optimal dispatch policy where (2) and (3) remain satisfied or we have contradicted the optimality of P6. We can derive another optimal policy, P9, by re-ordering the dispatch sizes of P8 to be non-increasing. Note that P9 also satisfies (2) and (3), and has one less dispatch than P6.

It must be the case for P9 that either the \(d'\) dispatch serves strictly more orders than the \(d'\) dispatch of P6, or the \(d'\) dispatch serves the same number of orders as it did in P6, but the \(d'+1\) dispatch serves strictly more than \(q_{\text{min}}\) orders. Thus we can continuously re-balance, consolidate, and re-order until we arrive at a dispatch policy which satisfies (2) and (3) and the first \(d'\) orders all serve all of the realized orders at the depot at the time of dispatch, or until the \(d'\) dispatch has become the last dispatch as all of the later dispatches have been consolidated and re-balanced into the \(d'\) dispatch. Thus we continue the process until (1) is also satisfied and we are done.

### 7.4 Proof of Proposition 9

Assume we are given a \(U\) such that \(0 \leq U \leq T - f(2q_{\text{min}})\) and \(i_{\text{max}} \leq 2\).

Suppose \(U = 0\), then trivially we have \(N^* = 0\). So, assume that \(U > 0\), and therefore \(2 \geq i_{\text{max}} \geq 1\).

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