

Planning the City Operations of a Parcel Express Company

Haotian Wu*, Martin Savelsbergh

H. Milton Stewart School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, Georgia 30332
haotian.wu@gatech.edu (HW), martin.savelsbergh@isye.gatech.edu (MS)

Yixiao Huang

SF Express (Group) Co., Ltd., Shenzhen, China 518048
yixiaohuang@sffmail.sf-express.com (YH)

We introduce an interesting and challenging routing and scheduling problem arising in the city operations of SF Express, a large package express carrier in China. Vehicles execute multiple trips during a planning horizon spanning multiple shifts, where a trip can involve deliveries only, pickups only, or deliveries followed by pickups. Complicating factors include split deliveries and pickups, cross-trip consistency requirements, and limited unloading capacity at the main hubs. We develop an optimization-based multi-phase heuristic solution approach seeking to minimize the number of vehicles used. An extensive computational study using real-world instances demonstrates the effectiveness of the approach.

Key words: city operations, package express carrier, multi-trip routes, backhauls, cross-trip consistency, unloading capacity

1. Introduction

Package delivery represents a significant part of the transportation industry. Measured by revenue, package delivery has been the fastest growing segment of the freight transport business in the United States in the 21st century, with one representative player, UPS, reporting a revenue increase from \$59.1 billion in 2015 to \$84.6 billion in 2020 (Statistica). A critical aspect of package delivery is timely service, which is driven, in part, by the growth of e-commerce, which relies heavily on fast delivery.

To provide an economically viable delivery service, package express carriers need to carefully allocate and utilize their resources. The primary challenge is to identify consolidation opportunities (so as to keep the costs down) while satisfying the service guarantees offered to customers (so as to maintain or increase market share). Package express carriers typically employ a complex hub

* Corresponding author.

network, in which vehicles transport packages between the hubs, and packages are unloaded, sorted, and loaded at the hubs. Routing packages through intermediate hubs is key to consolidation, but requires additional time and additional loading and unloading.

We consider the city operations of SF Express, a large Chinese package express carrier. That is, we consider the pickup and transport of packages to and the transport and delivery of packages from a gateway hub. A gateway hub provides an entry and exit point into and out of the carrier’s linehaul network connecting different cities. All cities served by SF Express have a gateway hub with some large cities having more than one gateway hub. The city operations can cover a large geographic area, and, as will become clear shortly, can be quite complex. The most challenging instance used in our computational study, for example, covers a service area of over 4,000 square miles and has about 200 local hubs feeding a single gateway hub.

In the past, when the fastest delivery option offered to customers was next-day delivery, packages picked up and delivered in the same city would also be transported to and from the gateway hub (to the gateway hub on the day they were picked up and from the gateway hub on the day they were delivered). Now that same-day or even 2-hour and 1-hour delivery options within a city are offered to customers, a separate same-day service network that does not involve the gateway hub has evolved (see, for example, Wu et al. 2020). Even though same-day intra-city delivery volume is growing rapidly, it is still small compared to the inter-city delivery volume.

We focus on the city operations for inter-city packages. A major challenge in these city operations is that packages arrive at the hubs (both at local hubs and at the gateway hub) throughout the day and that the gateway hub has limited capacity, in terms of docks for loading and unloading as well as for sorting packages. Consequently, city operations (have to) take place throughout the day across multiple shifts.

We develop optimization-based decision support technology for the planning of the transportation component of the city operations with a focus on minimizing the number of vehicles needed to serve all demand (pickup and delivery). The main features of the underlying routing and scheduling problem are that vehicles can make multiple trips during the day, that backhauls are allowed, i.e., pickups can follow deliveries in a vehicle trip, and that limited unloading capacity at the gateway has to be taken into account. There are other complicating factors, such as split delivery and pickup, and consistency requirements for deliveries to local hubs, which will be discussed in more detail in Section 3. Our multi-phase heuristic solution approach is able to produce high-quality vehicle schedules within a limited amount of computing time.

The contributions of this research are summarized as follows:

- We introduce an interesting and challenging routing and scheduling problem arising in the city operations of a package express carrier.

- We develop an optimization-based multi-phase solution approach capable of producing high-quality solutions efficiently.
- We conduct an extensive computational study providing insight into current operations, its bottlenecks, and potential adjustments to improve efficiency.

The remainder of the paper is organized as follows. In Section 2, we review relevant prior research. In Section 3, we give a detailed description of the city operations. In Section 4, we introduce an optimization-based multi-phase solution approach. In Section 5, we present and interpret the results of an extensive computational study. Finally, in Section 6, we briefly discuss insight gained and potential future research.

2. Literature Review

The city operations of a package express carrier can be viewed as a highly complex variant of a vehicle routing problem (VRP). In this section, we briefly review relevant VRP literature, i.e., VRP variants that have features in common with the routing and scheduling environment considered in this paper.

Motivated by the many complicating features encountered in real-life environments, the research community has recently started to focus on, so-called, *rich* VRP models; see Caceres-Cruz et al. (2014) for a survey. The size and complexity of rich VRP instances tend to make exact approaches impractical, and therefore researchers usually rely on heuristics to obtain high-quality solutions; see Vidal et al. (2013a) and Vidal et al. (2013b) for discussions of heuristic solution of VRPs.

The multi-trip VRP and the VRP with backhauls have many industrial applications; Cattaruzza et al. (2016b) and Koç and Laporte (2018) provide surveys of these two variants. Wassan et al. (2017) is the only paper we are aware of that considers the combination of these two VRP variants and proposes a two-level variable neighborhood search method for its solution.

Multi-period routing problems with release dates and due dates have been considered in Archetti et al. (2015b). Cattaruzza et al. (2016a) introduce a multi-trip VRP with time windows and release dates, in which goods to be delivered continuously arrive at a city distribution center. Research on the complexity of routing problems with release dates, e.g., Archetti et al. (2015a) and Reyes et al. (2018) have provided further insight into this VRP variant.

Most of the VRP literature assumes either instantaneous delivery (or pickup) or, possibly, a fixed service time at a delivery (or pickup) location. However, when delivery (or pickup) locations have limited capacity, then delivery routes have to be planned carefully, because otherwise delays may occur. Lam and Van Hentenryck (2016) consider a vehicle routing problem with pickup and delivery, time windows, and location congestion. Grangier et al. (2019) consider an environment in which the number of docks that can be used simultaneously for cross-docking is limited. Both studies

highlight the fact that accounting for limited loading or unloading capacity is challenging as it requires simultaneously solving a routing problem and a resource-constrained scheduling problem.

Allowing deliveries or pickups to be split has been shown to be beneficial in many VRP environments, but, at the same time, introduces modeling and computational challenges; Archetti and Speranza (2008) and Irnich et al. (2014) provide surveys, and a summary and analysis of different formulations and modeling approaches can be found in Munari and Savelsbergh (2020).

Companies often prefer to have the same drivers visit the same customers to increase customer satisfaction and to create driver familiarity with their tasks, which can be viewed as a consistency requirement. Consistency requirements have become more prevalent in recent years. Groër et al. (2009) study a situation when the same driver always visits the same customers at roughly the same time, and develop a record-to-record travel algorithm for generating high-quality consistent delivery routes. Kovacs et al. (2015) consider a more general and relaxed version of the previous problem and demonstrate significant cost savings from allowing more than one driver per customer. A survey of early solution approaches in which service consistency is acknowledged as side benefit is provided by Kovacs et al. (2014).

Compared to the many studies focusing on minimizing routing costs, there are relatively few that focus on fleet sizing, especially in a multi-trip context. Alinaghian and Shokouhi (2018) apply a hybrid approach integrating adaptive large neighborhood search and variable neighborhood search to first minimize the number of vehicles and then to minimize the total traversed distance in a multi-depot multi-compartment vehicle routing problem. Battarra et al. (2009) propose an iterative solution approach to minimize the fleet size based on decomposition and the use of a self-adaptive guidance mechanism to increase the likelihood of being able to combine individual trips into multi-trips. Cattaruzza et al. (2014) develop an iterated local search approach where a labelling procedure is designed to select paths and assign trips to vehicles, which they show outperforms the algorithm proposed by Battarra et al. (2009).

There are a number of papers that consider city operations of package service carriers, but none of them consider as many features as we do in this research. For example, some research considers either only pickup or only delivery operations (Yan et al. 2013, Pei et al. 2021). Other research considers both pickup and delivery operations (Sungur et al. 2010, Chang and Yen 2012), but do not allow multi-trip routes. Other features, such as limited hub capacity, split pickup or delivery, and strategic fleet sizing, have not been studied in this context.

It is also worth pointing to the growing interest in strategic customer clustering, in daily demand fluctuation in pickup or delivery services, in sustainability and environmental issues associated with city operations of package delivery companies, and in the impact of the presence of competing package express carriers in urban areas, e.g., Bard and Jarrah (2013), Bender et al. (2020), Baldi et al. (2019), and Perboli and Rosano (2019).

3. Problem Description

Let $D = (\{0\} \cup N, A)$ be a directed graph with $\{0\} \cup N$ a set of nodes representing the gateway hub (GH) and the local hubs (LHs), respectively, and A a set of directed arcs representing the possibility to travel between two hubs. A travel time $\tau_{ij} \in \mathbb{R}_+$ and a travel cost $c_{ij} \in \mathbb{R}_+$ are associated with each arc $a = (i, j) \in A$. Demand is partitioned into delivery demand, which becomes available at the GH and is destined for the LHs, and pickup demand, which becomes available at the LHs and is destined for the GH. Demand materializes over time during the planning horizon. This aspect is captured by partitioning the planning horizon into intervals and having a set of delivery demands D (one for each interval) and a set of pickup demands \hat{D} (one for each interval). Specifically, let T be the (ordered) set of end points of the intervals, then delivery demand $d_{it} \in D$ represents demand for LH i that becomes available at the GH during the time interval that ends at time $t \in T$ and consists of $q_{it} \in \mathbb{Z}_+$ packages with total weight $w_{it} \in \mathbb{R}_+$ that need to be dispatched from the GH before latest departure time v_{it} and have to be delivered and unloaded at the LH before due time ζ_{it} . Pickup demand $\hat{d}_{it} \in \hat{D}$ represents demand for the GH that becomes available at LH i during the time interval that ends at time $t \in T$ and consists of $\hat{q}_{it} \in \mathbb{Z}_+$ packages with total weight $\hat{w}_{it} \in \mathbb{R}_+$ that need to be dispatched from the LH before latest departure time \hat{v}_{it} and have to be delivered and unloaded at the GH before due time $\hat{\zeta}_{it}$. These timed demands are derived from historical data and represent estimates of future timed demands. There is a set of non-overlapping delivery shifts S and a set of non-overlapping pickup shifts \hat{S} , where the pickup shifts do not necessarily coincide with the delivery shifts. Delivery demand for the same LH that becomes available during the same delivery shift has the same latest departure and due time. Similarly, pickup demand from the same LH that becomes available during the same pickup shift has the same latest departure and due time. A heterogeneous fleet of vehicles is available at the GH at the start of the planning period to serve both delivery demand and pickup demand. Each vehicle type $m \in M$ has a maximum number of packages $Q_m \in \mathbb{Z}_+$ and a maximum weight $W_m \in \mathbb{R}_+$ that it can accommodate, and a loading time $\tau_m^l \in \mathbb{Z}_+$ and an unloading time $\tau_m^u \in \mathbb{Z}_+$. We assume the availability of an unlimited number of vehicles of each type. Every LH can accommodate the smallest vehicle type, but not all LHs can accommodate the larger vehicle types. The set of LHs that can be visited by a vehicle of type $m \in M$ is denoted by $N_m \subseteq N$.

A solution consists of a set of vehicle routes, each of which consists of a sequence of vehicle trips. In a vehicle trip, delivery demand has to be served before pickup demand. Thus, a vehicle trip involves loading delivery demand at the GH, traveling to each LH in the delivery path and unloading the corresponding delivery demand, traveling to each LH in the pickup path and loading the corresponding pickup demand, and, finally, traveling back to the GH and unloading the pickup demand. A vehicle trip can have a delivery path only or a pickup path only, but, for convenience,

in the discussion below, we assume that a vehicle trip has both a delivery path and a pickup path; the cases with a delivery path only or a pickup path only can be handled analogously.

For a vehicle trip to be feasible, the vehicle type has to be allowed at all LHs visited and the total delivery demand and the total pickup demand have to be less than or equal to the vehicle capacity (in terms of number of packages and weight). We assume that a LH can be visited at most twice in a vehicle trip, once in the delivery path and once in the pickup path.

We assume that it is possible to load only a fraction of a demand when the remaining vehicle capacity is insufficient to load the entire demand. Furthermore, since, in practice, it is challenging and inefficient, operationally, to separate packages with the same destination in a LH, we require that when a vehicle visits a LH to pick up demand and the vehicle's remaining capacity is sufficient, the vehicle picks up all the demand that is available at the time of the visit.

After completing a delivery path, a vehicle has to wait at least $\phi \in \mathbb{R}_+$ units of time before commencing a pickup path. The delay ϕ is introduced to capture the fact that often the last LH of the delivery path is equal to the first LH of the pickup path and the loading of a vehicle cannot start immediately after the unloading of the vehicle as the dock needs to be cleared of unloaded packages and the packages to be loaded need to be fetched. After completing a pickup path, a vehicle has to wait at least $\hat{\phi} \in \mathbb{R}_+$ units of time before commencing a next trip. The delay $\hat{\phi}$ is introduced to capture the time it takes to move a vehicle from an unloading dock to a loading dock (at a gateway hub there are dedicated loading and unloading docks).

The number of LHs visited in a delivery path cannot be more than $\kappa \in \mathbb{Z}_+$. Furthermore, when a vehicle delivers to κ LHs in one of its trips, it can only deliver to either the same or a subset of these LHs on any of its other trips. This requirement will be referred to as the *consistency* requirement, as it enforces consistency across the delivery paths performed by a vehicle. The main reason for imposing the consistency requirement is that the loading docks at the gateway hub are dedicated to a small number, κ , of LHs to simplify the sorting and staging operations at the gateway hub, and such dedicated dock - destination assignments are common in cross-docking terminals as well (see, for example, Van Belle et al. 2012). Other considerations relate to the spacing in time of deliveries at LHs and the driver familiarity with LHs. There is no limit on the number of LHs that can be visited in a pickup path and the pickup paths of the trips of a vehicle do not have to satisfy any consistency requirement, as there are no restrictions on the inbound side of the sorting operations.

The GH has $U \in \mathbb{Z}_+$ docks and thus at most U vehicles can be unloaded simultaneously. If a vehicle arrives at the GH at a time when U vehicles are being unloaded, it must wait until a dock becomes available; vehicles are processed in first-in first-out order.

The city operations planning problem seeks to find a set of vehicle routes, where a vehicle route consists of one or more vehicle trips, that feasibly serves all delivery demand and all pickup demand.

The objective is to minimize the number of vehicles required, and, given the minimum number of vehicles required, to minimize the total travel cost of the vehicle routes (where the travel cost of a vehicle route is the sum of the travel costs of the arcs in the route).

4. Solution Approach

A multi-phase decision framework has been developed which solves a series of optimization problems to handle the multi-period nature of the problem as well as the different complexities associated with the delivery and pickup components of the problem.

Algorithm 1 Multi-phase Solution Approach

- 1: **input:** directed graph $D = (\{0\} \cup N, A)$, delivery and pickup demand $D \cup \hat{D}$, delivery and pickup shifts $S \cup \hat{S}$
 - 2: **output:** a set of multi-trip routes R^* feasibly serving all delivery and pickup demand
 - 3: Determine delivery groups and delivery paths in all delivery shifts $s \in S$ based on a set partitioning model (1)
 - 4: Sort delivery and pickup shifts $S \cup \hat{S}$ with shift start times in non-decreasing order
 - 5: $R^* \leftarrow \emptyset$
 - 6: **for** $s \in S \cup \hat{S}$ **do**
 - 7: **if** $s \in S$ **then**
 - 8: Extend multiple-trip routes through a set partitioning model (5) to execute delivery paths in delivery shift s , update R^*
 - 9: **else**
 - 10: Determine back-haul pickup paths in pickup shift s based on an integer programming model (2)
 - 11: **for** $\delta \in \Delta^{\hat{s}}$ **do**
 - 12: Enumerate a set of stand-alone pickup paths that unload during δ
 - 13: Choose stand-alone pickup paths based on an integer programming model (3)
 - 14: Improve stand-alone pickup paths based on a hierarchical local search model (4)
 - 15: **end for**
 - 16: Extend multiple-trip routes through a set partitioning model (5) to execute pickup paths in pickup shift s , update R^*
 - 17: **end if**
 - 18: **end for**
 - 19: **return** R^* ;
-

4.1. Decision framework

The general idea of the proposed decision framework is to start with an empty set of multi-trip routes and repeatedly extend these routes with trips to serve demands during different parts of the planning horizon until all demands are served. Algorithm 1 shows a high-level description of our multi-phase heuristic solution approach.

Because city operations take place across multiple shifts and because demands associated with a LH that become available during the same shift have the same latest departure time and due time, we propose a natural time-decomposition approach: solve the planning problem in $|S| + |\hat{S}|$ phases (or steps) each associated with either a delivery or a pickup shift and processed in nondecreasing order of their start times. For a given shift, we create delivery and pickup paths that not only feasibly serve all demands in the shift, but that also anticipate the integration with the evolving partial multi-trip routes (so as to reduce the total number of required routes); these ideas are described in more detail in Section 4.2 and 4.3. After processing the individual shifts, the paths are merged into the partial multiple-trip routes exploiting any flexibility in the start and end times of the paths; these ideas are described in more detail in Section 4.4.

Because most LHs can only be visited by the smallest vehicle type and because all trips in a route have to be performed by the same vehicle, we make the simplifying assumption that only the smallest vehicle type is used.

For convenience, we define for each delivery shift and for each pickup shift the skeleton demand for a LH (in that shift) as the as-yet unserved demand for that LH with the tightest service time requirements in the shift, i.e., the demand with the latest available time among all the as-yet unserved demands in the shift. Skeleton demands have a major impact on the number of routes required to serve all demands as these demands are the most restricted in terms of delivery or pickup time.

4.2. The delivery component

Recall that a vehicle performing multiple trips can only deliver to the same group of LHs on each of its trips. Therefore, to increase the chance that effective multi-trip routes can be constructed, we assume that the delivery groups and the sequence in which the LHs in a delivery group are visited are the same for each of the delivery shifts.

This implies that we have to carefully choose delivery groups. Because it is possible that a LH may have no demand in one of the delivery shifts, and, therefore, that LH does not have to be visited in that shift, when creating delivery groups, we only group LHs with the same “delivery pattern”, i.e., we only group LHs that require deliveries in the same set of shifts. Using depth-first search, we enumerate every possible delivery group, i.e., every unique sequence of LHs satisfying

the upper bound on the number of LHs visited, and each is considered as a group candidate. Thus, delivery groups visiting the same set of LHs, but in a different order, are considered to be different group candidates.

If we assume that vehicles only need to serve delivery demand (so that their trips do not include pickup paths), then for a given delivery group $g \in G$, we can easily compute the minimum number of vehicles (and associated trips) required to serve the delivery demands of the LHs in the group (across all delivery shifts). Details of this computation can be found in Appendix A.

Let $n_g^1 \in \mathbb{Z}_+$ denote the minimum number of vehicles, let $n_g^2 \in \mathbb{Z}_+$ denote the associated minimum number of trips, and let $c_g \in \mathbb{R}_+$ denote the travel cost required to serve the delivery demands for the LHs in candidate group $g \in G$. We use hierarchical optimization to choose delivery groups from among the possible candidate groups. Let $G_i \subseteq G$ denote the set of delivery group candidates that include LH i and let y_g be a binary decision variable representing whether candidate group $g \in G$ is chosen ($y_g = 1$) or not ($y_g = 0$). We solve the following hierarchical optimization model:

$$\begin{aligned}
 & \min \sum_{g \in G} n_g^1 y_g && \text{(Phase 1)} \\
 & \min \sum_{g \in G} n_g^2 y_g && \text{(Phase 2)} \\
 & \min \sum_{g \in G} c_g y_g && \text{(Phase 3)} \\
 & \text{s.t.} \\
 & \sum_{g \in G_i} y_g = 1 && \forall i \in N; && (1a) \\
 & y_g \in \{0, 1\} && \forall g \in G. && (1b)
 \end{aligned}$$

That is, the set partitioning model seeks to minimize the number of vehicles needed to feasibly serve all delivery demands in Phase 1, to minimize the number of trips while ensuring the minimum number of vehicles is used in Phase 2, and to minimize the travel cost while ensuring the minimum number of vehicles and trips are used in Phase 3. Constraints (1a) ensure that each LH is assigned to exactly one delivery group. Constraints (1b) define the variables and their domains. Observe that because all possible candidate groups are enumerated (which implies that every possible delivery subgroup of a delivery group is present as well), the selected delivery groups will form a partition of the set of LHs.

The chosen set of delivery groups implies a set of delivery paths. Each of these delivery paths has some flexibility in terms of the departure time from the GH and the time of the delivery at the last LH in the path. In the integration phase of the approach, this flexibility is exploited to merge delivery paths (as well as pickup paths) into the evolving (partial) multi-trip routes.

4.3. The pickup component

Whereas the primary challenge in the delivery component is the consistency requirement across trips performed by the same vehicle, the primary challenge in the pickup component is the limited unloading capacity at the GH.

For each pickup shift $\hat{s} \in \hat{S}$, we partition the time interval between the earliest possible unload start time and the latest possible unload end time into a sequence of non-overlapping time intervals $\Delta^{\hat{s}} = \{\delta_1, \delta_2, \dots, \delta_{|\Delta^{\hat{s}}|}\}$, where the length of each time interval equals the unloading time of the smallest vehicle. Thus, the end time of interval δ_i equals the start time of interval δ_{i+1} for $1 \leq i \leq |\Delta^{\hat{s}}| - 1$ and the end time of interval $\delta_{|\Delta^{\hat{s}}|}$ equals the latest time the unloading of a vehicle can be completed in pickup shift \hat{s} . Let $\Delta = \bigcup_{\hat{s} \in \hat{S}} \Delta^{\hat{s}}$, and let $\tau_{\delta} \in \mathbb{Z}_+$ denote the start time of interval $\delta \in \Delta$. We assume that the unloading operation of any pickup path occurs during one of the time intervals $\delta \in \Delta$.

When processing a shift, we create two sets of pickup paths for vehicle trips, back-haul pickup paths and stand-alone pickup paths, to feasibly serve the pickup demands in the shift. We also introduce a, so-called, *anticipation* mechanism to increase the chance of being able to merge the created pickup paths with existing partial multi-trip routes.

4.3.1. Anticipation mechanism We try to discourage the creation of pickup paths that cannot be connected to existing partial routes, and, thereby hope to decrease the number of vehicles required to serve the demands.

More specifically, each time we create a tentative back-haul or stand-alone pickup path originating at a LH i , we determine a, so-called, critical time τ_i^* , the minimum of the latest possible departure time from LH i (in the shift) and the earliest possible time an existing partial route can depart from LH i , and force the pickup path to depart at or after this critical time, but at or before the latest possible departure time. After a tentative back-haul or stand-alone pickup path originating at LH i has been created, we connect it to the partial multi-trip route that arrives at LH i as close as possible to the critical time. This partial route will be excluded from the list of partial multi-trip routes when determining the critical times for subsequently created tentative pickup paths.

This anticipation mechanism has some similarities to the adaptive guidance mechanism adopted in Battarra et al. (2009) and Cattaruzza et al. (2014). However, our anticipation mechanism is simpler to incorporate in a multi-trip route construction environment as no critical feature detection procedure is required. Also, our anticipation mechanism does not require iterative execution and, thus, reduces the computational effort required.

4.3.2. The back-haul pickup problem Rather than forcing vehicles to return to the GH after performing deliveries, allowing vehicles to perform pickups on their way back to the GH, i.e., allowing back-hauls, can potentially reduce the number of vehicles required and the transportation cost by taking advantage of otherwise unused capacity.

The back-haul pickup problem for a shift seeks to find a set of pickup paths that can be connected to the delivery paths for the shift. For each delivery path, we try to construct a set of potential

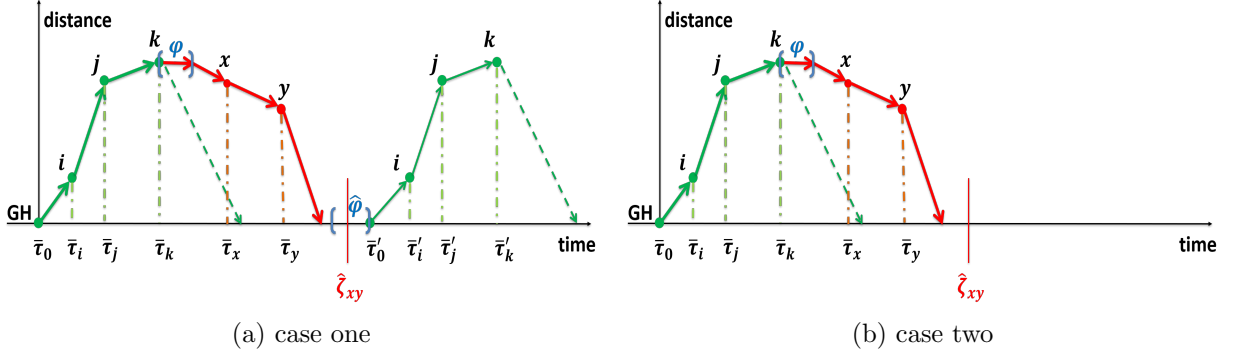


Figure 1 Back-haul Pickup Path Creation Illustration

back-haul pickup paths. Two situations need to be considered, illustrated in Figure 1. In the situation depicted on the left, the vehicle performing the delivery path under consideration is scheduled to perform another delivery path after returning to the GH. Therefore, the potential back-haul pickup paths have to be such that they can feasibly be performed between the two delivery paths. In Figure 1a, the green lines represent two consecutive delivery trips of a partial route $((0, \bar{\tau}_0), (i, \bar{\tau}_i), (j, \bar{\tau}_j), (k, \bar{\tau}_k), (0, \bar{\tau}_k + \tau_{k0}), (0, \bar{\tau}'_0), (i, \bar{\tau}'_i), (j, \bar{\tau}'_j), (k, \bar{\tau}'_k), (0, \bar{\tau}'_k + \tau_{k0}))$. For simplicity, assume that loading and unloading are instantaneous. A feasible back-haul pickup path (represented in red) visiting LH x and LH y at times $\bar{\tau}_x$ and $\bar{\tau}_y$, respectively, has to satisfy $\bar{\tau}_x \geq \bar{\tau}_k + \phi + \tau_{kx}$, $\bar{\tau}_y \geq \bar{\tau}_x + \tau_{xy}$, and $\min\{\zeta_{xy}, \bar{\tau}'_0 - \hat{\phi}\} \geq \bar{\tau}_y + \tau_{y0}$, where $\zeta_{xy} = \zeta_x = \zeta_y$ represents the due time of pickup demand served by the pickup path. In the situation depicted on the right, the vehicle performing the delivery path under consideration does not have to perform another delivery path after returning to the GH. In Figure 1b, the green lines represent the last delivery trip performed in the partial route. A feasible back-haul pickup path (represented in red) visiting LH x and LH y at times $\bar{\tau}_x$ and $\bar{\tau}_y$, respectively, has to satisfy $\bar{\tau}_x \geq \bar{\tau}_k + \phi + \tau_{kx}$, $\bar{\tau}_y \geq \bar{\tau}_x + \tau_{xy}$, and $\zeta_{xy} \geq \bar{\tau}_y + \tau_{y0}$, where $\zeta_{xy} = \zeta_x = \zeta_y$ represents the due time of pickup demand served by the pickup path.

In both situations, we force the back-haul pickup paths to depart at or after the critical time (at LH x). In order to use fewer trips to serve pickup demands, we restrict ourselves to back-haul pickup paths that visit a LH only if the visit time equals either the time when (1) the skeleton

demand becomes available, or when (2) all demands in the pickup shift becoming available after the visit time sum up to the vehicle capacity (in which case a single visit to the LH to pickup these demands suffices). Also, to increase our chances of being able to connect the resulting pickup paths to delivery paths, we limit the number of LHs visited in the shift by back-haul pickup paths to $\hat{\kappa}_b^{\hat{s}} \in \mathbb{Z}_+$ and we enforce an LH is visited at most once by a back-haul pickup path in the shift.

After enumerating all back-haul pickup paths satisfying the above requirements, we use hierarchical optimization to choose back-haul pickup paths. Let P denote the set of delivery paths for the shift, let \hat{P} denote the set of candidate back-haul pickup paths, and let $n_b \in \mathbb{Z}_+$ denote the number of LHs visited in back-haul pickup path $b \in \hat{P}$. For convenience, let \hat{P}_d denote the set of candidate pickup paths that could serve as back-hauls for delivery path $d \in P$, let $\Delta_b^{\hat{s}}$ denote the set of possible time intervals in the shift during which back-haul pickup path $b \in \hat{P}$ can unload, and let \hat{P}_i denote the set of back-haul pickup paths that visit LH i . Let binary variable λ_b indicate whether back-haul trip $b \in \hat{P}$ is selected ($\lambda_b = 1$) or not ($\lambda_b = 0$) and let binary variable η_{bd}^{δ} indicate whether back-haul pickup path $b \in \hat{P}$ serves as back-haul for delivery path $d \in P$ and unloads during time period $\delta \in \Delta_b^{\hat{s}}$ ($\eta_{bd}^{\delta} = 1$) or not ($\eta_{bd}^{\delta} = 0$). We solve the following hierarchical optimization problem:

$$\begin{aligned}
& \max \quad \sum_{b \in \hat{P}} n_b \lambda_b \quad (\text{Phase 1}) \\
& \min \quad \sum_{d \in P} \sum_{b \in \hat{P}_d} \sum_{\delta \in \Delta_b^{\hat{s}}} \tau_{\delta} \eta_{bd}^{\delta} \quad (\text{Phase 2}) \\
& \text{s.t.} \\
& \sum_{b \in \hat{P}_d} \sum_{\delta \in \Delta_b^{\hat{s}}} \eta_{bd}^{\delta} \leq 1 \quad \forall d \in P; \tag{2a} \\
& \sum_{d \in P: b \in \hat{P}_d} \sum_{\delta \in \Delta_b^{\hat{s}}} \eta_{bd}^{\delta} = \lambda_b \quad \forall b \in \hat{P}; \tag{2b} \\
& \sum_{d \in P} \sum_{b \in \hat{P}_d: \delta \in \Delta_b^{\hat{s}}} \eta_{bd}^{\delta} \leq U \quad \forall \delta \in \Delta^{\hat{s}}; \tag{2c} \\
& \sum_{b \in \hat{P}_i} \lambda_b \leq 1 \quad \forall i \in N; \tag{2d} \\
& \lambda_b \in \{0, 1\} \quad \forall b \in \hat{P}; \tag{2e} \\
& \eta_{bd}^{\delta} \in \{0, 1\} \quad \forall d \in P \quad \forall b \in \hat{P}_d \quad \forall \delta \in \Delta_b^{\hat{s}}. \tag{2f}
\end{aligned}$$

That is, the integer program seeks to maximize the number of LHs visited by the chosen back-haul pickup paths in Phase 1, and to minimize the sum of the unload start times of the chosen back-haul pickup paths while ensuring the maximum number of LHs visited in Phase 2. Constraints (2a) ensure that at most one back-haul pickup path is chosen for each delivery path. Constraints

(2b) ensure that each chosen back-haul pickup path unloads during one of the available time intervals at the GH. Constraints (2c) ensure that the number of chosen back-haul pickup paths that unload during the same time interval at the GH does not exceed the number of vehicles that can simultaneous unload at the GH. Constraints (2d) ensure that a LH is visited at most once in the shift. Constraints (2e) and (2f) define the variables and their domains.

The chosen set of back-haul pickup paths forms a subset of the pickup paths to be merged with the partial multi-trip routes in the integration problem. Each back-haul pickup path has some flexibility in its start and end time, depending on the set of demands served and the LHs visited by the path.

4.3.3. The stand-alone pickup problem The stand-alone pickup problem for a shift seeks to find a set of pickup paths serving the pickup demands that are not served by the back-haul pickup paths for the shift.

More specifically, stand-alone pickup paths are created using any remaining capacity in the unloading time intervals at the GH in reverse chronological order (because the later we visit a LH the more demand we may be able to pickup). For a given time interval $\delta \in \Delta^s$, we use breadth-first search to generate a set of candidate stand-alone pickup paths with the following characteristics:

- Unloading at the GH takes place during time interval δ ;
- No more than $\hat{\kappa}_a^s \in \mathbb{Z}_+$ LHs are visited, each LH is visited once, and travel times between LHs visited consecutively are relatively small;
- Departure from the first LH is no earlier than its critical time;
- Time constraints of the pickup demands loaded are respected.
- At each LH visited demands that have become available later are loaded first until either all demands available at the time of the visit have been loaded or until the vehicle capacity is reached (the last demand loaded can be split and partially loaded).

For a given unloading time interval δ , a path-based integer programming model is used to choose a subset of the candidate stand-alone pickup paths. Let \hat{P}^δ denote the set of candidate stand-alone pickup paths, let U^δ denote the number of additional paths that can unload in time interval δ (in addition to the back-haul pickup paths that unload during the time interval), and let d_p denote the size of the demands served by stand-alone pickup path $p \in \hat{P}^\delta$, where size is expressed as a fraction of the vehicle capacity and based on the number of packages as well as the weight of the packages (choosing the maximum of the two relative to the vehicle capacity). In order to minimize the number of pickup paths needed to serve remaining pickup demand, we compute and use two evaluation metrics for each of the enumerated pickup paths. We prefer pickup paths that visit a LH after its skeleton demand becomes available, or at a time when the demand that becomes available

after that time sums up to the vehicle capacity. Let $N^p \subseteq N$ denote the set of LHs visited by pickup path $p \in \hat{P}^\delta$. Let $\psi_{ip}^1 = 1$ if pickup path p visits LH i at a desirable time and $\psi_{ip}^1 = 0$ otherwise, and let d_{ip} denote the size of the demands available at LH i at the time that pickup path p visits LH i , where size is expressed as a multiple of the vehicle capacity and based on the number of packages as well as the weight of the packages (choosing the maximum of the two relative to the vehicle capacity). Note that the size of the demands available at LH i may exceed the vehicle capacity. The first evaluation metric ψ_p^1 for path $p \in \hat{P}^\delta$ is defined as $\psi_p^1 = \sum_{i \in N^p} \psi_{ip}^1 \times \lceil d_{ip} \rceil$. For each LH visited in pickup path p , we can determine the difference between the earliest time its skeleton demand can be delivered at the GH (assuming a direct dispatch from the LH to the GH) and the start time of time interval δ . The second evaluation metric ψ_p^2 for path $p \in \hat{P}^\delta$ is defined to be the average of these differences for the LHs visited in pickup path p .

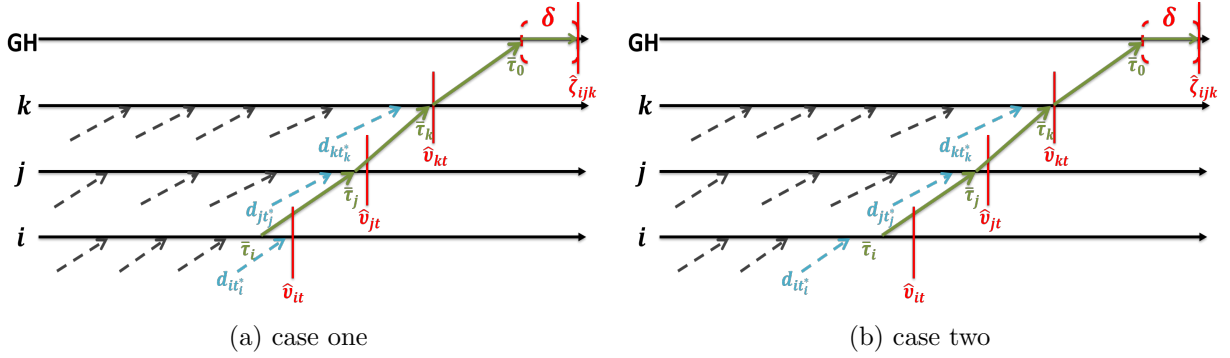


Figure 2 Stand-Alone Pickup Path Example

To illustrate, consider the two pickup paths in Figure 2. The pickup paths visit LHs i , j , and k at times $\bar{\tau}_i$ ($\bar{\tau}_i \geq \tau_i^*$), $\bar{\tau}_j$, and $\bar{\tau}_k$, respectively, and start unloading at the GH at time $\bar{\tau}_0$. For simplicity, we assume that loading at the LHs is instantaneous, that there is no waiting at any of LHs, and that the total demand at the LHs is less than the vehicle capacity. As before, the dashed arcs represent pickup demand becoming available at a LH. The skeleton demands $d_{it_i^*}$, $d_{jt_j^*}$, and $d_{kt_k^*}$ become available at t_i^* , t_j^* , and t_k^* , respectively. As the pickup path in Figure 2a visits LH i before its skeleton demand becomes available, but visits LH j and LH k after their skeleton demands become available, we have $\psi_{ip}^1 = 0$ and $\psi_{jp}^1 = \psi_{kp}^1 = 1$. Because the vehicle capacity is larger than the total demand, $\lceil d_{ip} \rceil = \lceil d_{jp} \rceil = \lceil d_{kp} \rceil = 1$ and first evaluation metric is $\psi_p^1 = \psi_{ip}^1 \times \lceil d_{ip} \rceil + \psi_{jp}^1 \times \lceil d_{jp} \rceil + \psi_{kp}^1 \times \lceil d_{kp} \rceil = 2$. The second evaluation metric is $\psi_p^2 = \frac{1}{3} \times ((\tau_\delta - \tau_{i0} - t_i^*) + (\tau_\delta - \tau_{j0} - t_j^*) + (\tau_\delta - \tau_{k0} - t_k^*))$. Now consider the pickup path in Figure 2b. Because this pickup path also visits LH i after its skeleton demand becomes available, $\psi_{ip}^1 = 1$ and the value of the first evaluation metric is $\psi_p^1 = 3$. The value of the second evaluation metric ψ_p^2 remains the same.

For convenience, let $\hat{P}_i^\delta \subseteq \hat{P}^\delta$ denote the set of enumerated pickup paths that visit LH i . Let binary variable z_p indicate whether pickup path $p \in \hat{P}^\delta$ is chosen ($z_p = 1$) or not ($z_p = 0$). The hierarchical optimization problem for choosing pickup paths that unload in interval δ is as follows:

$$\max \sum_{p \in \hat{P}^\delta} \psi_p^1 z_p \quad (\text{Phase 1})$$

$$\min \sum_{p \in \hat{P}^\delta} \psi_p^2 z_p \quad (\text{Phase 2})$$

$$\max \sum_{p \in \hat{P}^\delta} d_p z_p \quad (\text{Phase 3})$$

s. t.

$$\sum_{p \in \hat{P}^\delta} z_p \leq U^\delta; \quad (3a)$$

$$\sum_{p \in \hat{P}_i^\delta} z_p \leq 1 \quad \forall i \in N; \quad (3b)$$

$$z_p \in \{0, 1\} \quad \forall p \in \hat{P}^\delta. \quad (3c)$$

That is, the path-based model seeks to maximize the sum of the first evaluation metrics ψ_p^1 for chosen pickup paths in Phase 1, to minimize the sum of the second evaluation metrics ψ_p^2 for the chosen pickup paths while ensuring the first evaluation metric does not deteriorate, and to maximize the total demand served while ensuring that the first and second evaluation metrics do not deteriorate in Phase 3. Constraint (3a) ensures that at most U^δ pickup paths unload during time period δ . Constraints (3b) ensure that a LH is visited at most once by the chosen pickup paths. Constraints (3c) define the variables and their domains.

To keep the number of enumerated stand-alone pickup paths manageable, we have imposed a number of (artificial) restrictions during the enumeration. Next, we remove these restrictions and use local search to explore whether the chosen paths can be improved. To do so, we take the LHs visited on the chosen pickup paths and formulate and solve a *vehicle routing problem with time windows* to see if the same demands can be served using fewer paths and incurring lower travel costs. This may be possible because we remove the restriction on the number of LHs visited with a pickup path and we remove the restriction that travel times between LHs consecutively need to be relatively small.

Let N^δ denote the set of LHs visited by the chosen pickup paths, and let q'_i and w'_i denote the number of packages and the weight of the packages picked up at LH i , respectively. Furthermore, let e'_i and ℓ'_i denote the earliest time (all) these packages are available at LH i and the latest time (all) these packages can be at LH i , respectively. Let $|\Delta^{\hat{s}}|$ denote the length of the shift, and let Q ,

W , and τ^1 denote the maximum number of packages, the maximum weight, and the loading time of a vehicle, respectively. Let binary decision variable x_{ij} indicate whether a vehicle travels directly from hub i to hub j ($x_{ij} = 1$) or not ($x_{ij} = 0$), let continuous variable ρ_i indicate the arrival time at hub $i \in N^\delta \cup \{0'\}$ (before loading or unloading starts), and let continuous variables α_i and β_i indicate the number and the weight of packages on the vehicle before it starts loading or unloading at hub $i \in N^\delta \cup \{0'\}$, respectively. The hierarchical local search model is formulated as follows:

$$\min \sum_{i \in N^\delta} x_{0i} \quad (\text{Phase 1})$$

$$\min \sum_{i \in N^\delta} \sum_{j \in N^\delta \cup \{0'\}} c_{ij} x_{ij} \quad (\text{Phase 2})$$

s.t.

$$\sum_{j \in N^\delta \cup \{0'\}} x_{ij} = 1 \quad \forall i \in N^\delta; \quad (4a)$$

$$\sum_{j \in \{0\} \cup N^\delta} x_{ji} = 1 \quad \forall i \in N^\delta; \quad (4b)$$

$$\sum_{i \in N^\delta} x_{i0'} \leq U^\delta; \quad (4c)$$

$$\alpha_i + q'_i \leq \alpha_j + Q(1 - x_{ij}) \quad \forall i \in N^\delta, j \in N^\delta \cup \{0'\}; \quad (4d)$$

$$\beta_i + w'_i \leq \beta_j + W(1 - x_{ij}) \quad \forall i \in N^\delta, j \in N^\delta \cup \{0'\}; \quad (4e)$$

$$\rho_i + \tau^1 + \tau_{ij} \leq \rho_j + |\Delta^{\hat{s}}|(1 - x_{ij}) \quad \forall i \in N^\delta, j \in N^\delta \cup \{0'\}; \quad (4f)$$

$$\rho_i \geq t_i^* - |\Delta^{\hat{s}}|(1 - x_{0i}) \quad \forall i \in N^\delta; \quad (4g)$$

$$\rho_i \geq e'_i \quad \forall i \in N^\delta; \quad (4h)$$

$$\rho_i + \tau^1 \leq \ell'_i \quad \forall i \in N^\delta; \quad (4i)$$

$$\rho_{0'} = \tau_\delta; \quad (4j)$$

$$x_{ij} \in \{0, 1\} \quad \forall i \in \{0\} \cup N^\delta, j \in N^\delta \cup \{0'\}; \quad (4k)$$

$$0 \leq \alpha_i \leq Q \quad \forall i \in N^\delta \cup \{0'\}; \quad (4l)$$

$$0 \leq \beta_i \leq W \quad \forall i \in N^\delta \cup \{0'\}; \quad (4m)$$

$$\rho_i \geq 0 \quad \forall i \in N^\delta \cup \{0'\}. \quad (4n)$$

That is, the local search model seeks to minimize the total number of pickup paths in Phase 1, and to minimize the total travel cost of pickup paths while ensuring no more than the minimum number of pickup paths is used in Phase 2. Constraints (4a) and (4b) ensure that each LH is visited exactly once. Constraint (4c) enforces that the number paths does not exceed the unloading capacity at the GH. Constraints (4d) and (4e) ensure that a vehicle can only be loaded up to its

capacity. Constraints (4f) establish the relationship between the vehicle's arrival time at a hub and its immediate predecessor. Constraints (4g) ensure that pickup paths start at or after the critical time at the first LH. Constraints (4h) ensure that demands can only be loaded after they become available. Constraints (4i) ensure that vehicles serving demands depart no later than their latest departure time. Constraints (4j) ensure that the pickup paths start unloading at the right time. Constraints (4k), (4l), (4m) and (4n) impose the domains of decision variables. Observe that the solution from the path-based model provides an initial feasible solution for this local search model, which allows the model to be solved relatively quickly.

If after determining stand-alone pickup paths that unload during time interval δ_l , there are remaining unserved demands at the LHs, we seek to find stand-alone pickup paths that unload during the preceding time interval δ_{l-1} . The sets of back-haul and stand-alone pickup paths chosen for the shift will be merged with the partial multi-trip routes in the integration problem.

4.4. Integration problem

At the end of each shift, after the delivery and pickup paths for the shift have been chosen, we solve an integration problem. In the integration problem, we seek to extend the existing partial multi-trip routes with the paths for the shift. If necessary, new partial multi-trip routes are created. The goal is to end up with a minimum number of partial multi-trip routes.

Depth-first search is used to enumerate all feasible partial multi-trip routes that include at least one of the existing multi-trip routes or one of the delivery and pickup paths for the shift. Any flexibility in the start and end time of the existing partial routes and the delivery and pickup paths is taken into account during the enumeration. For a pickup path this means that we allow the unloading time interval to change if feasible. If the shift is a delivery shift, then the consistency requirements are respected during the enumeration.

Let R' denote the set of existing partial multi-trip routes, let P denote the set of delivery and pickup paths for the shift, and let R denote the set of enumerated partial multi-trip routes. For each partial route $r \in R$, let c_r denote the travel cost of the partial r and let $\tilde{\Delta}_r$ denote the set of feasible sets of time intervals during which the pickup paths in the partial route can unload at the GH (a partial route can include multiple pickup paths each unloading at the GH during a particular time interval $\delta \in \Delta^{\hat{S}}$).

Let $\tilde{\Delta}_\delta$ denote the set of feasible sets of time intervals that include δ . Let R_p denote the set of partial routes that include path $p \in P$. Let $R_{r'}$ denote the set of partial routes that include partial route $r' \in R'$. Let binary variable ξ_r indicate whether partial multi-trip route $r \in R$ is chosen ($\xi_r = 1$) or not ($\xi_r = 0$) and let binary variable θ_r^σ indicate whether the pickup paths in partial multi-trip route r use feasible set of unloading time intervals $\sigma \in \tilde{\Delta}_r$ ($\theta_r^\sigma = 1$) or not ($\theta_r^\sigma = 0$).

The hierarchical optimization model that determines the new set of partial multi-trip routes is as follows:

$$\begin{aligned}
& \min \sum_{r \in R} \xi_r && \text{(Phase 1)} \\
& \min \sum_{r \in R} c_r \xi_r && \text{(Phase 2)} \\
& \min \sum_{r \in R} \sum_{\sigma \in \tilde{\Delta}_r} \left(\sum_{\delta \in \sigma} \tau_\delta \right) \theta_r^\sigma && \text{(Phase 3)} \\
& \text{s.t.} \\
& \sum_{r \in R_p} \xi_r = 1 && \forall p \in P; && (5a) \\
& \sum_{r \in R_{r'}} \xi_r = 1 && \forall r' \in R'; && (5b) \\
& \sum_{\sigma \in \tilde{\Delta}_r} \theta_r^\sigma = \xi_r && \forall r \in R; && (5c) \\
& \sum_{r \in R} \sum_{\sigma \in \tilde{\Delta}_\delta} \theta_r^\sigma \leq U && \forall \delta \in \Delta^{\hat{S}}; && (5d) \\
& \theta_r^\sigma \in \{0, 1\} && \forall \sigma \in \tilde{\Delta}_r \quad \forall r \in R; && (5e) \\
& \xi_r \in \{0, 1\} && \forall r \in R. && (5f)
\end{aligned}$$

That is, the model seeks to minimize the number of partial multi-trip routes needed to perform the existing partial routes and delivery and pickup paths for the shift in Phase 1, to minimize the total travel cost of the chosen partial routes while ensuring a minimum number of partial routes is used in Phase 2, and to minimize the sum of start times of the unloading time intervals of the pickup paths in the chosen routes while ensuring a minimum number of partial routes of minimum travel cost is used in Phase 3. Constraints (5a) ensure that each path in the shift is included in exactly one partial route. Constraints (5b) ensure that each existing partial route is included in exactly one (new) partial route. Constraints (5c) ensure that the pickup paths of a chosen partial route unload using one of the feasible sets of unloading time intervals. Constraints (5d) ensure that the upper bound on the number of vehicles that can unload simultaneously is respected for each unloading time interval. Constraints (5e) and (5f) define the variables and their domains.

After processing all shifts we obtain a set of multi-trip routes that feasibly serve all delivery and pickup demands.

5. Computational Study

In this section, we first present an in-depth analysis of one of the most challenging city operations of this type in the SF Express service network. We provide details of the environment as well as

the schedules generated by the the decision framework. After that, we consider three other city operations, providing fewer specifics and focusing on differences and their impact.

The decision framework has been coded in Python and uses Gurobi 8.1.1 to solve integer programs. All experiments have been performed in a single thread of a dedicated Intel Xeon ES-2630 2.3GHz processor with 50GB RAM running Red Hat Enterprise Linux Server 7.4. It takes around 20 minutes of computing time to generate schedules for each of the four city operations.

In the first service area considered, referred to as instance I_1 , SF Express operates 1 GH and 192 LHs; the service area with its GH and LHs is shown in Figure 3a. The average and standard deviation of travel distances and travel times between the hubs are 53 and 35 kilometers, and 54 and 27 minutes, respectively. The capacities of the vehicles in the fleet, i.e., maximum number of packages (Q_m) and maximum weight of packages (W_m), and their loading time (τ_m^u) and unloading time (τ_m^l) can be found in Table 1. Figure 3b shows the largest vehicle type that each LH can accommodate.

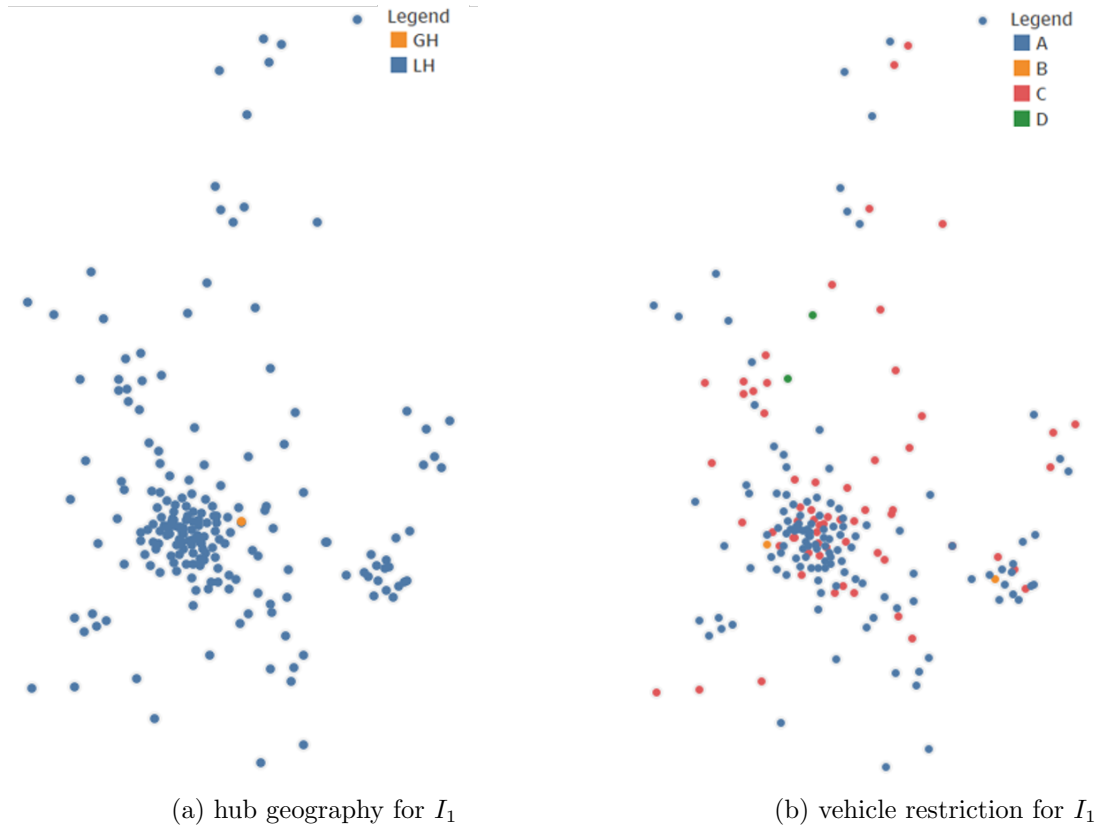


Figure 3 Hub Network for I_1

The demand for a representative day is captured by partitioning the day into 144 time intervals of 10 minutes. (The use of a coarser time discretization to define demand would not affect the

Table 1		Vehicle Information			
vehicle type	m	Q_m	W_m	τ_m^l	τ_m^u
A		800	1500	10	7
B		1000	1800	10	8
C		1200	2000	10	9
D		2000	4500	10	18

applicability of the proposed solution approach.) Each delivery demand is characterized by a time interval (one of the 144 time intervals in the planning horizon), a destination LH, the number of packages, and the weight of the packages. Each pickup demand is characterized by a time interval, an origin LH, the number of packages, and the weight of the packages. Figures 4a and 4b show the cumulative weight of the delivery demands arriving at the GH for a typical LH over time, as well as the cumulative weight of the pickup demands arriving at this LH for the GH over time, respectively.

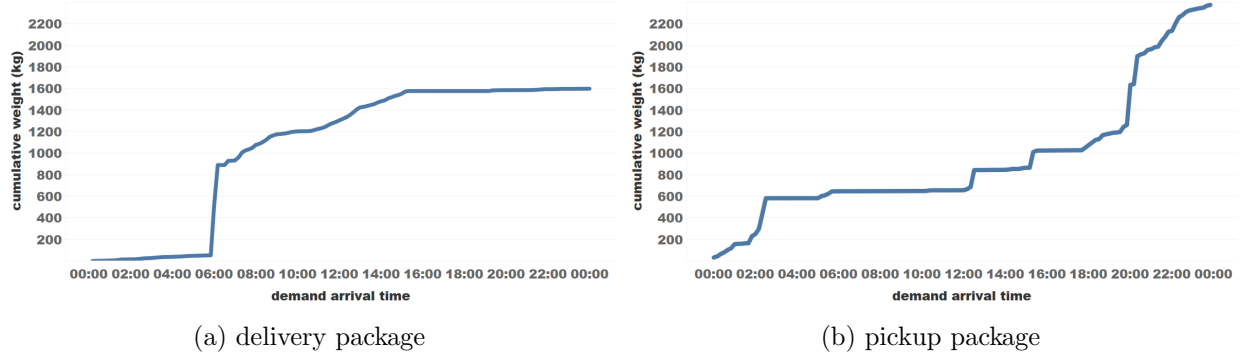


Figure 4 Cumulative Package Weight (kgs) VS. Demand Arrival Time Example

There are four delivery shifts and four pickup shifts, each with associated latest departure times at the origins (v_{it}, \hat{v}_{it}) and due times at the destinations ($\zeta_{it}, \hat{\zeta}_{it}$) for demands to be served in a shift, illustrated in Appendix B. Each delivery and pickup demand has to be served in a particular shift (based on the time interval during which the packages arrive). Demands that become available after the latest load start time of the last shift have to be served in the first shift (in practice, these demands are served in the first shift on the next day). Table 2 shows the number of LHs requiring service during each of the delivery and pickup shifts ($|N^s|$), as well as the total number of packages to be served (Q^s) and the total weight of the packages to be served. The weight of packages to be served at each of the LHs and for each delivery and pickup shift are shown in Appendix C. As expected, it shows that highest delivery weights are seen in the first delivery shift (representing mostly packages that have arrived at the GH from other cities during the night), and that highest

pickup weights are seen in the last pickup shift (representing mostly packages that will depart from the GH to other cities during the night). Package express carriers operating in a market with a smaller package volume or operating a gateway hub with larger loading and unloading capacities may not have to employ multiple shifts, in which case the proposed solution approach may not be suited.

Table 2 Shift Information for I_1

shift s	delivery				pickup			
	1st	2nd	3rd	4th	1st	2nd	3rd	4th
$ N^s $	192	132	192	192	142	191	168	189
Q^s	103,697	14,257	31,628	18,371	23,594	17,083	37,900	61,133
W^s	146,006	29,748	58,279	21,086	40,100	27,642	67,050	126,051

After unloading at the last LH in a delivery path a vehicle has to remain at the LH for at least 15 minutes before traveling to the first LH in a pickup path ($\phi = 15$). After completing a trip, a vehicle has to remain at the GH for at least 15 minutes before it can start its next trip ($\hat{\phi} = 15$). The GH has 11 unloading docks and, thus, can unload at most 11 vehicles simultaneously ($U = 11$). In practice, the number of loading docks at the GH is not restrictive. The maximum number of LHs visited in a delivery path is 3, i.e., $\kappa = 3$; as mentioned before, this is due to dedicating loading docks at the GH to a small number of LHs.

Instance characteristics and preliminary computational experiments were used to determine parameter values that ensure high solution quality. For example, when processing a pickup shift, we try not to create pickup paths with a long duration as this restricts the creation of candidate delivery paths with early start times in subsequent shifts. The number of LHs visited on a back-haul pickup path is limited to two in every pickup shift. When enumerating stand-alone pickup paths, the number of LHs visited is limited to three in the first and second pickup shifts, to one in the third pickup shift, and to two in the fourth pickup shift, and the travel times between LHs visited consecutively cannot exceed 25 minutes.

The LHs have one of two delivery patterns: either the LH requires deliveries in the first, third, and fourth delivery shift (60 LHs) or the LH requires deliveries in all four delivery shifts (132 LHs). After enumerating 15,213 candidate delivery groups, 74 delivery groups are chosen using our hierarchical optimization approach. The minimum number of vehicles required to serve delivery demand is 137 and the associated minimum number of trips is 370.

Table 3 presents critical statistics for the solutions produced by the multi-phase decision framework for each of the eight shifts: the number of new paths created ($|P|$), for the new paths created

the average vehicle utilization (μ^s), the average travel time (τ^s), the average travel distance (c^s), and the average number of visited LHs (n^s), and the number ($|R|$) and travel cost (C) of the partial multi-trip routes after integrating the new paths.

Table 3 Multi-Phase Decision Framework Results for I_1

phase	shift s	path type	$ P $	μ^s	τ^s	c^s	n^s	$ R $	C
1	1st delivery	delivery	169	78.10%	59.49	52.57	2.56	133	1.72×10^4
2	2nd delivery	delivery	51	39.08%	50.88	44.34	2.59	133	2.16×10^4
3	1st pickup	back-haul	0	—	—	—	—	133	2.21×10^4
		stand-alone	54	59.21%	58.28	51.78	2.74		
4	3rd delivery	delivery	76	55.49%	62.88	57.50	2.59	133	3.03×10^4
5	2nd pickup	back-haul	46	19.90%	60.50	61.53	1.65	133	3.52×10^4
		stand-alone	54	26.92%	59.69	58.39	2.26		
6	4th delivery	delivery	74	31.18%	62.82	57.43	2.59	137	4.32×10^4
7	3rd pickup	back-haul	84	51.55%	53.65	51.42	2.00	137	4.38×10^4
		stand-alone	14	77.88%	50.00	47.86	1.50		
8	4th pickup	back-haul	0	—	—	—	—	147	5.56×10^4
		stand-alone	147	64.81%	55.81	55.36	1.67		

We observe that even after processing the first two pickup shifts, the number of partial multi-trip routes is still equal to the number of partial multi-trip routes after processing the first delivery shift, which shows that the pickup paths created in these first two pickup shifts can easily be “sandwiched” between the delivery paths. It is also evident that the large number of packages to be picked up in the last pickup shift (and the large weight of these packages) in combination with the limited unloading capacity at the GH cause the number of required multi-trip routes to jump (from 137 to 147); there are 10 vehicles performing single-trip routes in the last pickup shift.

We “depict” the 147 multi-trip routes in Appendix D, where different activities are represented using different colors (the load start times at origins and the unload end times at destinations of all delivery and pickup paths are shown). Four vertical dashed reference lines indicate the latest unload end times at the GH for the pickup paths in the four pickup shifts. The multi-trip routes are given in order on non-decreasing end times. This figure also clearly demonstrates that the last pickup shift is the “bottleneck”. Consequently, to further reduce the number of multi-trip routes (required vehicles), changes or efficiency improvements have to be found that provide additional flexibility in the last pickup shift.

The dock utilization at the GH is shown in Figure 5, where we give the number of docks in use at different times during the planning period.

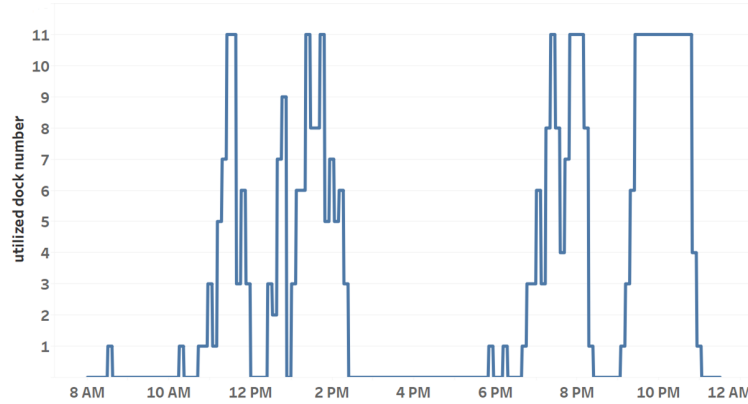


Figure 5 Number of Utilized Docks

We show the arrival times of delivery and pickup paths (ρ_{it} and $\hat{\rho}_{it}$) at the LHs during the planning period in Appendix E, where due times of delivery paths and latest departure times of pickup paths are also depicted for reference.

To assess the quality of the solution produced by the multi-phase decision framework for this instance, we compute a simple lower bound based on ideas introduced in Kontoravdis and Bard (1995). In the context of a vehicle routing problem with time windows, they propose to construct an *incompatibility* graph in which nodes represent customers and edges represent pairs of customers that cannot be visited in the same route. The size of a maximum clique in this graph provides a lower bound on the number of routes needed to serve all customers.

For each shift in our setting, we use a similar idea to compute a lower bound on the number of routes needed for serving the demand in the shift. Specifically, we construct an incompatibility graph in which nodes represent LHs and edges represent pairs of LHs for which the skeleton demand cannot be served on the same route (i.e., there will be an edge between nodes representing LH i and LH j if the skeleton demands $d_{it_i^*}$ and $d_{jt_j^*}$ cannot be served in the same route due to the available and latest departure times at origins and the due times at destinations).

Because skeleton demand at a LH represents only a fraction of the demand of that LH and because other restrictions are ignored, e.g., the limit on the number of vehicles that can simultaneously unload at the GH, the minimum idle time between consecutive paths, and the capacity of the vehicles performing trips, the size of the maximum clique in the incompatibility graph provides a (possibly weak) lower bound on the number of routes required to serve the demand in the shift. Table 4 reports the lower bounds obtained for each delivery and pickup shift in this manner. The maximum lower bound (over the eight shifts) is 129 vehicles, and, not surprisingly, is obtained in the last pickup shift.

For completeness sake, the available and latest departure times of the skeleton demands in the last shift are shown in Figure 6.

Table 4 Lower Bound: Maximum Clique

shift	delivery				pickup			
	1st	2nd	3rd	4th	1st	2nd	3rd	4th
lower bound	11	14	22	9	31	59	49	129

**Figure 6** Skeleton Demand Timings in the 4th Pickup Shift

Given that the solution produced by the multi-phase decision framework uses 147 multi-trip routes, which is less than 15% above the (possibly weak) lower bound of 129, we feel confident the solution produced is of high quality.

Next, we briefly summarize the characteristics of the three other city operations considered. The number of LHs ($|N|$), the average and standard deviation of travel distances (\bar{c}_{ij} , \tilde{c}_{ij}) and travel times ($\bar{\tau}_{ij}$, $\tilde{\tau}_{ij}$), the number of delivery shifts and pickup shifts ($|S|$, $|\hat{S}|$), and the number of unload docks at the GH (U) can be found in Table 5. For each of the three instances, the number of

Table 5 Basic characteristics for I_2, I_3 and I_4

instance	$ N $	\bar{c}_{ij}	\tilde{c}_{ij}	$\bar{\tau}_{ij}$	$\tilde{\tau}_{ij}$	$ S $	$ \hat{S} $	U
I_2	87	25	21	35	18	4	6	19
I_3	51	42	28	45	20	5	5	9
I_4	79	59	38	58	27	4	4	10

LHs requiring service during each of the delivery and pickup shifts and the total weight of the packages to be served are shown in Table 6. The service areas with the GH and LHs are shown in Appendix F. Except for the unloading time of the smallest vehicle type, which is 10 minutes, other aspects, such as demand partition, vehicle capacity, vehicle loading time, the minimum idle time between delivery and pickup path in the same trip, the minimum idle time between consecutive trips performed by the same vehicle, and the maximum number of LHs visited in a delivery path are the same as in I_1 .

Table 6 Shift information for I_2, I_3 , and I_4

instance	shift s	delivery					pickup					
		1st	2nd	3rd	4th	5th	1st	2nd	3rd	4th	5th	6th
I_2	$ N^s $	77	77	77	77	-	78	87	87	84	87	6
	W^s	157,123	19,109	27,348	35,128	-	124,953	29,671	25,438	9,484	13,939	833
I_3	$ N^s $	50	50	28	39	23	43	47	47	40	50	-
	W^s	107,024	19,910	14,057	12,491	4,509	148,106	30,651	20,065	5,071	14,281	-
I_4	$ N^s $	78	67	79	79	-	77	75	57	26	-	-
	W^s	97,808	21,230	27,728	11,226	-	160,139	44,081	7,879	18,676	-	-

There are a few notable differences compared to I_1 . There are more shifts in I_2 and I_3 compared to I_1 , which implies more frequent deliveries to a LH and more frequent pickups from a LH. Furthermore, the highest pickup weights in these three instances are seen in the first pickup shifts rather than in the last pickup shifts, and the difference between the latest departure time of pickup demands and the available time of skeleton delivery demands in adjacent shifts is much smaller, which reduces the opportunities to find pickup paths that can be linked to delivery paths.

Appendix G presents critical statistics for the solutions produced by the multi-phase decision framework for the three instances. As expected, the large weight of packages to be picked up in the first pickup shifts, the time conflicts between the first pickup shifts and preceding delivery shifts, as well as the limited unloading capacity at the GH cause the number of required multi-trip routes to jump after processing the first pickup shifts. For completeness sake, we also show the resulting multiple routes for the three instances in Appendix H, Appendix I and Appendix J, respectively.

Finally, we compare the schedules produced by the multi-phase solution approach (MSA) with the schedules used in practice by SF Express in terms of the number of vehicles used, $|R|$ (we only compare the number of vehicle to protect other more confidential information); see Table 7. Of course this is not a perfect, or even fair, comparison, but indicates the potential for cost reductions and the value that the use of the proposed technology might bring.

Table 7 Comparison with the schedules used in practice.

Instance	$ R $	Reduction
I_1	147	>25%
I_2	109	>20%
I_3	69	>15%
I_4	142	>25%

6. Final Remarks

Due to the scale and complexity of the city operations of package express carrier SF Express, the planning of these city operations is a time-consuming and challenging task. We have developed an effective optimization-based multi-phase heuristic solution approach seeking to minimize the number of vehicles required to time-feasibly serve demand (and to minimize the travel costs given the minimum required fleet).

Insight gained from the development of the decision framework is summarized below:

- When planning operations that take place across multiple shifts, it is useful to consider a time-decomposition approach that solves the planning problem in phases.
- Properly managing the interaction between the shifts (i.e., between the phases) is critical and challenging. For example, minimizing the number of vehicles in one shift may result in a larger than necessary fleet size for the entire planning horizon. To ensure global solution quality encouraging and discouraging the use of certain types of (delivery and pickup) paths in individual shifts is necessary.
- In this particular environment, the consistency requirement and the limited unloading capacity at the GH are the main challenges when trying to minimize the fleet size. Pre-defining candidate “delivery groups” of LHs (and a visit sequence within each group) and pre-defining time intervals for unloading operations help deal with these challenges.

Currently, SF Express also imposes consistency requirements in the creation of pickup paths in their planning model as this simplifies the problem complexity and allows for a set-partition-like formulation that integrates the decisions of (pickup and delivery) path selection, path connection, and LH grouping. Furthermore, the pickup (and delivery) demand of a LH in a shift is aggregated, which reduces the number of demands considered. As the consistency constraints on the pickup paths mimic what planners tend to do, the solutions tend to have patterns similar to those that have been used in practice, which facilitates adoption of solutions produced by the planning models. The planning models have been incorporated in the decision support systems used by regional planners to design and adjust the monthly plans of city operations. Based on the reports from planners from 10 regions, the use of the planning models has resulted in cost savings of more than \$35,000 per month since its introduction in some regions.

Observing its potential to further reduce operating costs (as shown in Table 7), the adoption and integration of the proposed optimization-based multi-phase heuristic solution approach into the SF Express planning system is under discussion.

The research reported in this paper is only one of many efforts towards providing SF Express with effective decision support in its planning and operations functions. There are many other challenges that can be addressed using optimization tools, which we plan to pursue in the near

future. Here we mention one. After reaching their origin gateway hub (city operations), inter-city packages with high service levels often need to be transported to their destination gateway hubs by a combination of ground transportation and air transportation, before being delivered at their final destination (city operations). This leads to an express shipment air service network design problem that consists of determining an integrated network of flights paths and vehicle routes that enables the cost-efficient overnight flow of express packages between origin and destination gateways.

Acknowledgments

This work would not have been possible without the assistance of the *Operations Planning Department* and *Data Science and Operations Research Center* at SF Express. We thank them not only for providing us with real-world instances, but also for the many informative and insightful discussions.

References

- M. Alinaghian and N. Shokouhi. Multi-depot multi-compartment vehicle routing problem, solved by a hybrid adaptive large neighborhood search. *Omega*, 76:85–99, 2018.
- C. Archetti and M. G. Speranza. The split delivery vehicle routing problem: A survey. In *The vehicle routing problem: Latest advances and new challenges*, pages 103–122. Springer, 2008.
- C. Archetti, D. Feillet, and M. G. Speranza. Complexity of routing problems with release dates. *European journal of operational research*, 247(3):797–803, 2015a.
- C. Archetti, O. Jabali, and M. G. Speranza. Multi-period vehicle routing problem with due dates. *Computers & Operations Research*, 61:122–134, 2015b.
- M. M. Baldi, D. Manerba, G. Perboli, and R. Tadei. A generalized bin packing problem for parcel delivery in last-mile logistics. *European Journal of Operational Research*, 274(3):990–999, 2019.
- J. F. Bard and A. I. Jarrah. Integrating commercial and residential pickup and delivery networks: A case study. *Omega*, 41(4):706–720, 2013.
- M. Battarra, M. Monaci, and D. Vigo. An adaptive guidance approach for the heuristic solution of a minimum multiple trip vehicle routing problem. *Computers & Operations Research*, 36(11):3041–3050, 2009.
- M. Bender, J. Kalcsics, and A. Meyer. Districting for parcel delivery services—a two-stage solution approach and a real-world case study. *Omega*, 96:102283, 2020.
- J. Caceres-Cruz, P. Arias, D. Guimarans, D. Riera, and A. A. Juan. Rich vehicle routing problem: Survey. *ACM Computing Surveys (CSUR)*, 47(2):1–28, 2014.
- D. Cattaruzza, N. Absi, D. Feillet, and D. Vigo. An iterated local search for the multi-commodity multi-trip vehicle routing problem with time windows. *Computers & Operations Research*, 51:257–267, 2014.
- D. Cattaruzza, N. Absi, and D. Feillet. The multi-trip vehicle routing problem with time windows and release dates. *Transportation Science*, 50(2):676–693, 2016a.
- D. Cattaruzza, N. Absi, and D. Feillet. Vehicle routing problems with multiple trips. *4OR*, 14(3):223–259, 2016b.
- T.-S. Chang and H.-M. Yen. City-courier routing and scheduling problems. *European Journal of Operational Research*, 223(2):489–498, 2012.
- P. Grangier, M. Gendreau, F. Lehuédé, and L.-M. Rousseau. The vehicle routing problem with cross-docking and resource constraints. *Journal of Heuristics*, pages 1–31, 2019.
- C. Groër, B. Golden, and E. Wasil. The consistent vehicle routing problem. *Manufacturing & service operations management*, 11(4):630–643, 2009.
- S. Irnich, M. Schneider, and D. Vigo. Chapter 9: Four variants of the vehicle routing problem. In *Vehicle Routing: Problems, Methods, and Applications, Second Edition*, pages 241–271. SIAM, 2014.

-
- Ç. Koç and G. Laporte. Vehicle routing with backhauls: Review and research perspectives. *Computers & Operations Research*, 91:79–91, 2018.
- G. Kontoravdis and J. F. Bard. A grasp for the vehicle routing problem with time windows. *ORSA journal on Computing*, 7(1):10–23, 1995.
- A. A. Kovacs, B. L. Golden, R. F. Hartl, and S. N. Parragh. Vehicle routing problems in which consistency considerations are important: A survey. *Networks*, 64(3):192–213, 2014.
- A. A. Kovacs, B. L. Golden, R. F. Hartl, and S. N. Parragh. The generalized consistent vehicle routing problem. *Transportation Science*, 49(4):796–816, 2015.
- E. Lam and P. Van Hentenryck. A branch-and-price-and-check model for the vehicle routing problem with location congestion. *Constraints*, 21(3):394–412, 2016.
- P. Munari and M. Savelsbergh. Compact formulations for split delivery routing problems. *Optimization Online 7500*, 2020.
- Z. Pei, X. Dai, Y. Yuan, R. Du, and C. Liu. Managing price and fleet size for courier service with shared drones. *Omega*, 104:102482, 2021.
- G. Perboli and M. Rosano. Parcel delivery in urban areas: Opportunities and threats for the mix of traditional and green business models. *Transportation Research Part C: Emerging Technologies*, 99:19–36, 2019.
- D. Reyes, A. L. Erera, and M. W. Savelsbergh. Complexity of routing problems with release dates and deadlines. *European Journal of Operational Research*, 266(1):29–34, 2018.
- Statistica. United Parcel Service of America, Inc.’s revenue and net income from FY 2009 to FY 2020. <https://www.statista.com/statistics/270108/ups-revenue-and-net-income/> (06-07-2021).
- I. Sungur, Y. Ren, F. Ordóñez, M. Dessouky, and H. Zhong. A model and algorithm for the courier delivery problem with uncertainty. *Transportation science*, 44(2):193–205, 2010.
- J. Van Belle, P. Valckenaers, and D. Cattrysse. Cross-docking: State of the art. *Omega*, 40(6):827–846, 2012.
- T. Vidal, T. G. Crainic, M. Gendreau, and C. Prins. Heuristics for multi-attribute vehicle routing problems: a survey and synthesis. *European Journal of Operational Research*, 231(1):1–21, 2013a.
- T. Vidal, T. G. Crainic, M. Gendreau, and C. Prins. A hybrid genetic algorithm with adaptive diversity management for a large class of vehicle routing problems with time-windows. *Computers and Operations Research*, 40(1):457–489, 2013b.
- N. Wassan, N. Wassan, G. Nagy, and S. Salhi. The multiple trip vehicle routing problem with backhauls: formulation and a two-level variable neighbourhood search. *Computers & Operations Research*, 78: 454–467, 2017.
- H. Wu, I. Herszterg, M. Savelsbergh, and Y. Huang. Service network design for same-day delivery with hub capacity constraints. *Optimization Online 7675*, 2020.

- S. Yan, J.-R. Lin, and C.-W. Lai. The planning and real-time adjustment of courier routing and scheduling under stochastic travel times and demands. *Transportation Research Part E: Logistics and Transportation Review*, 53:34–48, 2013.

Appendix A Determine Fleet Size for Serving Delivery Demand

We illustrate the computing the minimum number of vehicle required to serve the demands in a candidate group by means of the example in Figure 7. In the figure, the horizontal line represents the planning horizon of 12 time units at the GH and the arrival of delivery demands $\{d_{i1}, d_{i2}, \dots, d_{i12}\}$ for a LH i at the GH during each of the unit time intervals. There are two consecutive delivery shifts and the two dashed arrows represent the skeleton demands, d_{i6} and d_{i12} , each becoming available no later than the latest departure times v_{it}^1 and v_{it}^2 in their respective shifts.

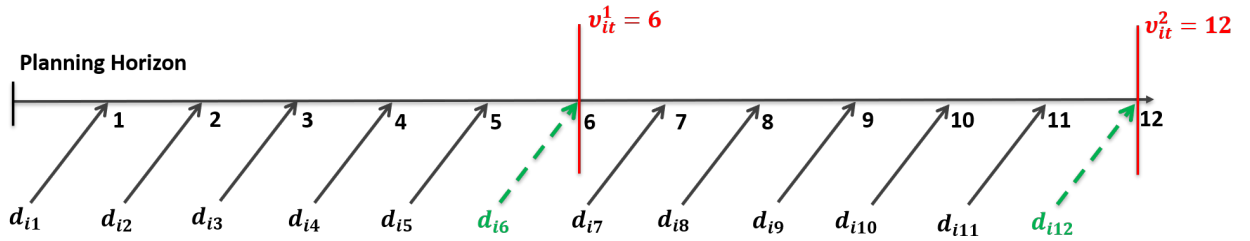


Figure 7 Delivery Demand Pattern Example

For simplicity, we assume that loading at the GH and unloading at the LH is instantaneous. Furthermore, let the latest departure time from the GH in a shift be equal to the arrival time of the skeleton demand. Consequently, the skeleton demands need to be loaded and dispatched as soon as they become available. LH i is the only hub in the delivery group candidate and the due time of each demand for LH i is 1.5 time units after the latest departure time (v_{it}^1 or v_{it}^2). An out-and-back delivery trip between the GH and the LH takes 3 time units, i.e., each leg takes 1.5 time units. The number of packages and the maximum weight a smallest vehicle can accommodate are 6 and 6 units, respectively. When the number of packages number and the weight of each demand equal to 2 and 2 units, respectively, then one vehicle is able to perform the 4 delivery trips needed to serve delivery demand for LH i . When the number of packages and the weight increase to 3 and 3 units for demands d_{i5} , d_{i6} , d_{i11} and d_{i12} , and decrease to 1 and 1 unit for demands d_{i1} , d_{i2} , d_{i7} and d_{i8} , and remain unchanged for other demands, then again 4 delivery trips are needed, but two vehicles are required, because the vehicle serving d_{i1} , d_{i2} , d_{i3} and d_{i4} (d_{i7} , d_{i8} , d_{i9} and d_{i10}) cannot be back at the GH before time 7 (time 13) and therefore cannot serve d_{i5} and d_{i6} (d_{i11} and d_{i12}) in its next trip as this would violate latest departure time constraint.

Algorithm 2 DETERMINE THE MINIMUM NUMBER OF VEHICLES (AND ASSOCIATED TRIPS)

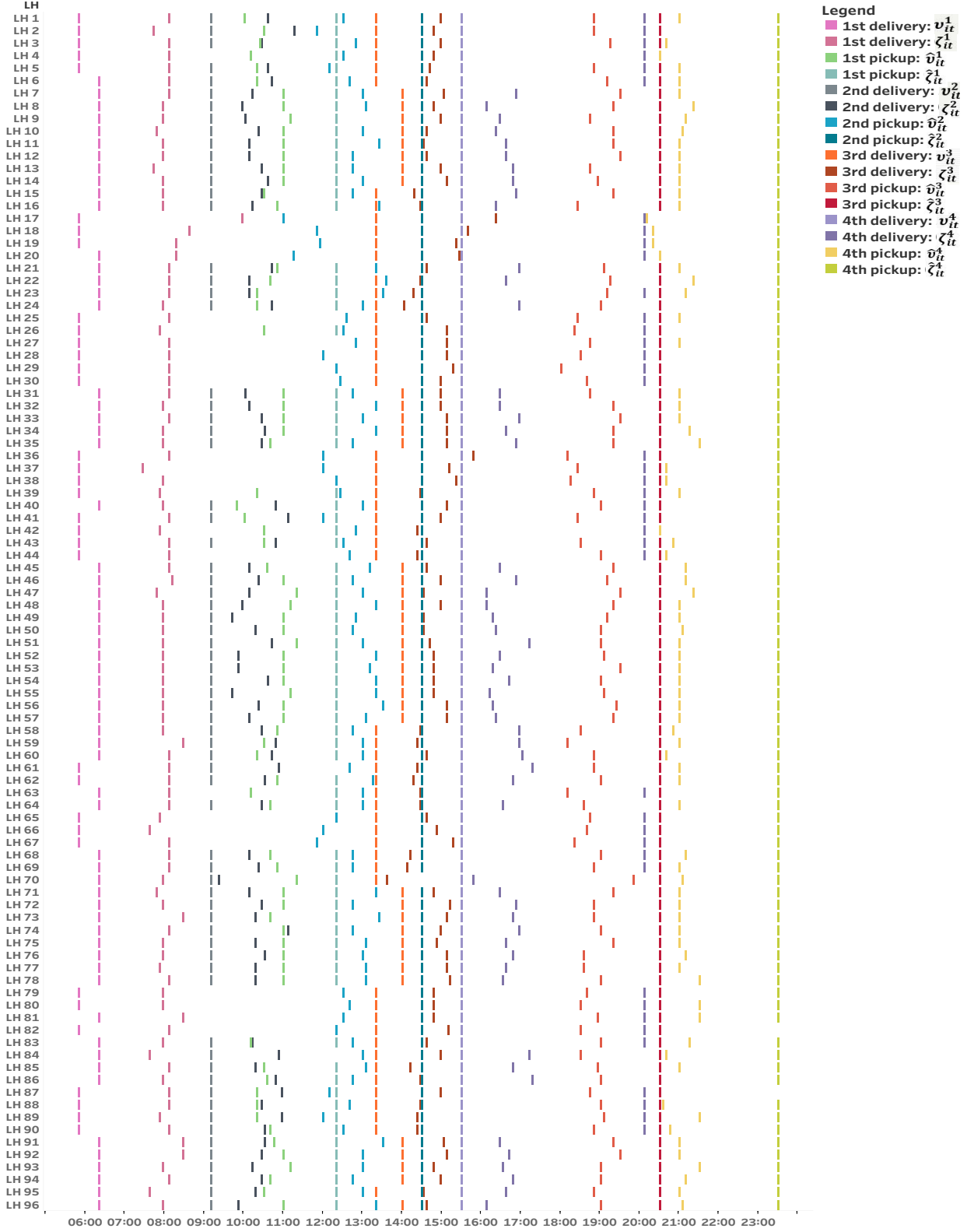
```

1: input: the delivery group candidate  $g$  and the corresponding delivery demand
2: output: number of tentative vehicles  $n_g^1$  and tentative trips  $n_g^2$ 
3:  $n_g^1 \leftarrow 0$ 
4:  $n_g^2 \leftarrow 0$ 
5: Sort associated delivery shifts with shift start times in non-increasing order
6: for  $s \in S$  do
7:   Sort associated delivery demand in shift  $s$  with available time in non-increasing order
8:   while all delivery demand in shift  $s$  has not been served do
9:     Construct a tentative delivery trip that loads demand in the sorted order, until all demand
       has been loaded, or until the vehicle becomes full
10:    if earliest end time of tentative trip +  $\hat{\phi} \leq$  latest depart time of existing vehicles then
11:      Assign the tentative trip to an existing vehicle
12:    else
13:      Assign the tentative trip to a new vehicle
14:       $n_g^1 \leftarrow n_g^1 + 1$ 
15:    end if
16:    Update route schedule for the assigned vehicle
17:     $n_g^2 \leftarrow n_g^2 + 1$ 
18:  end while
19: end for
20: return  $n_g^1$  and  $n_g^2$ 

```

Cases when delivery group candidates and the corresponding delivery demand patterns are different can be handled analogously following Algorithm 2.

Appendix B Shift Timing for I_1



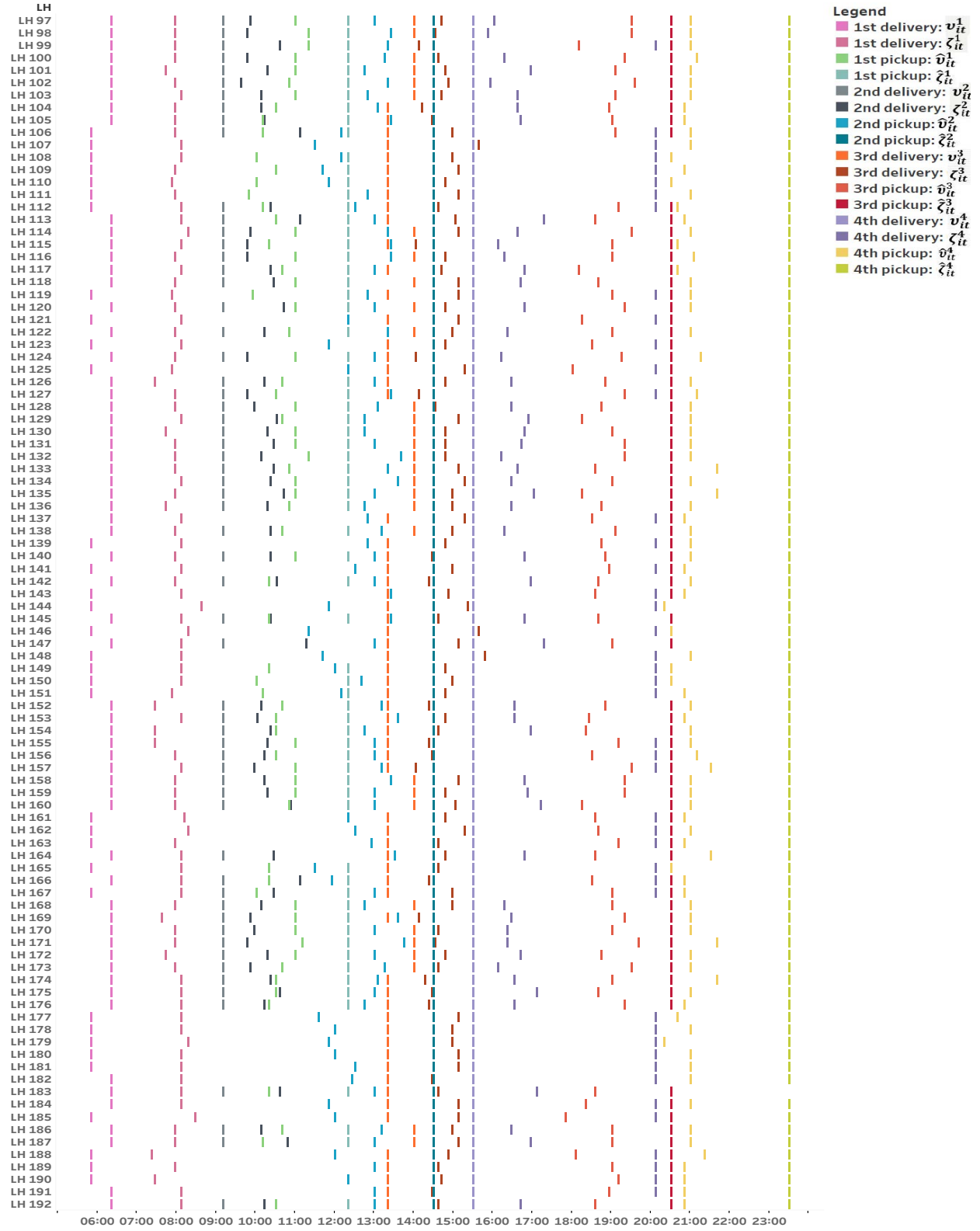


Figure 8 Delivery and Pickup Shift Timing for I_1

Appendix C Delivery and Pickup Demand Weight for I_1

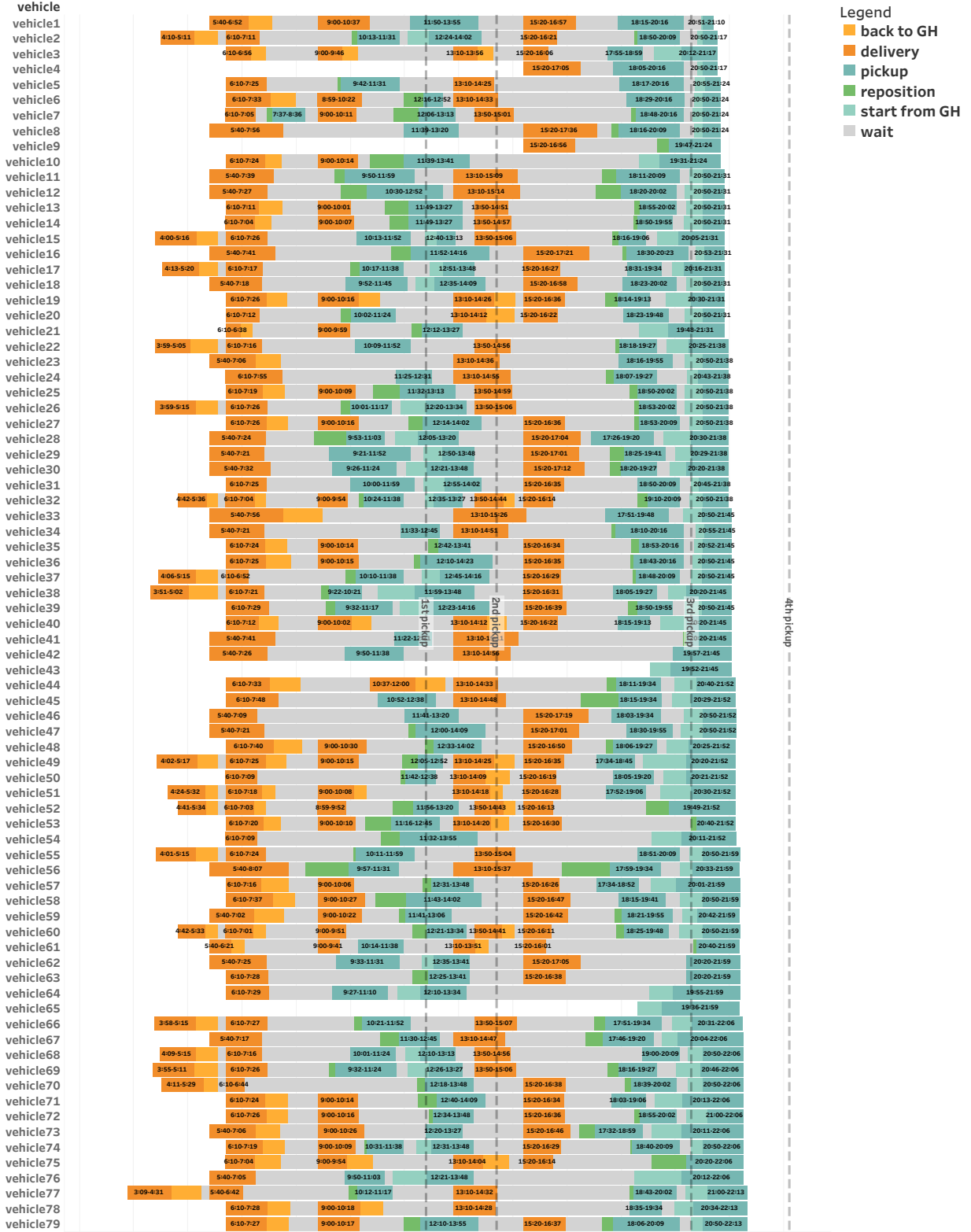


Figure 9 Delivery Demand Weight (kgs) for I_1



Figure 10 Pickup Demand Weight (kgs) for I_1

Appendix D Multiple-Trip Route Schedule for I_1



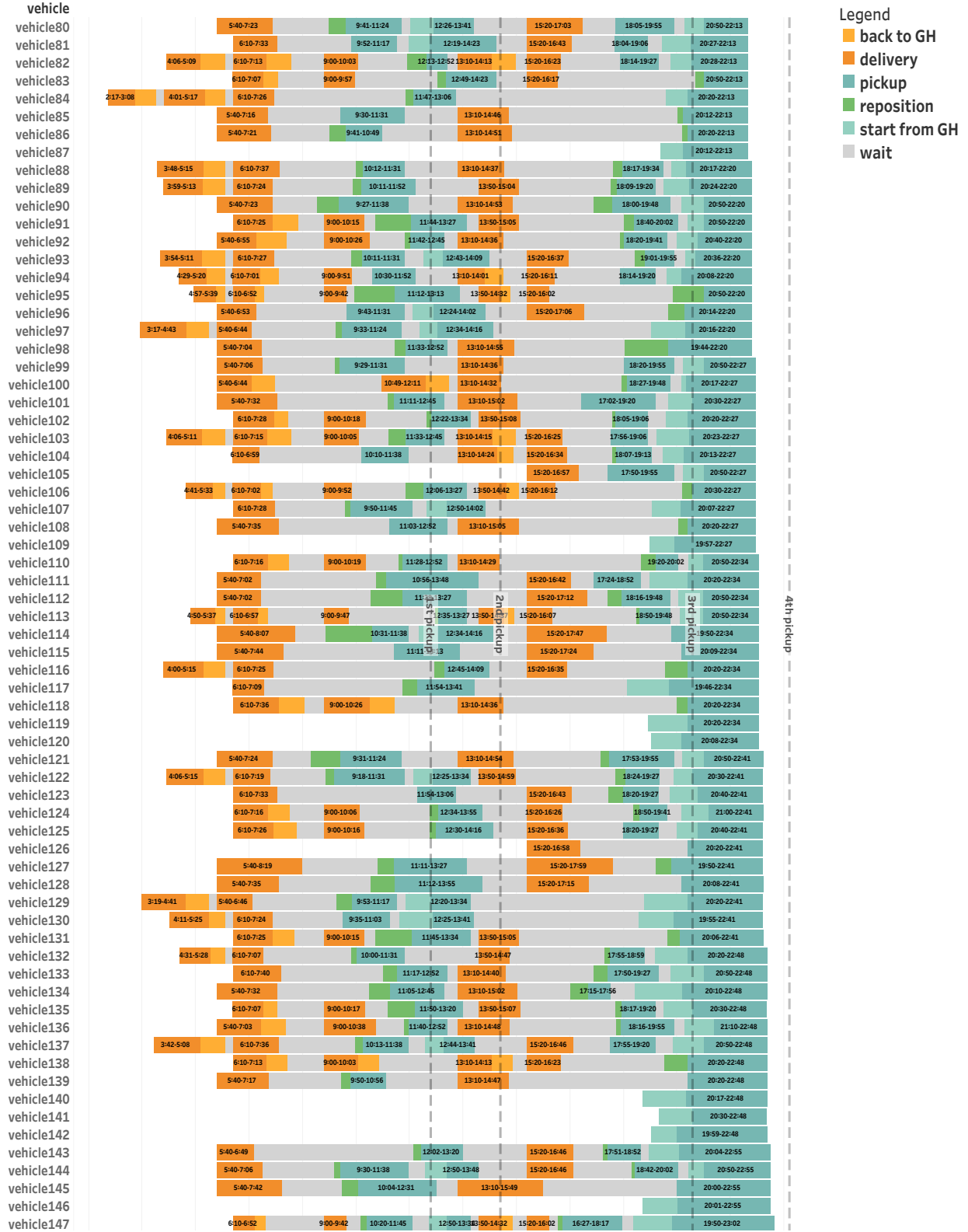
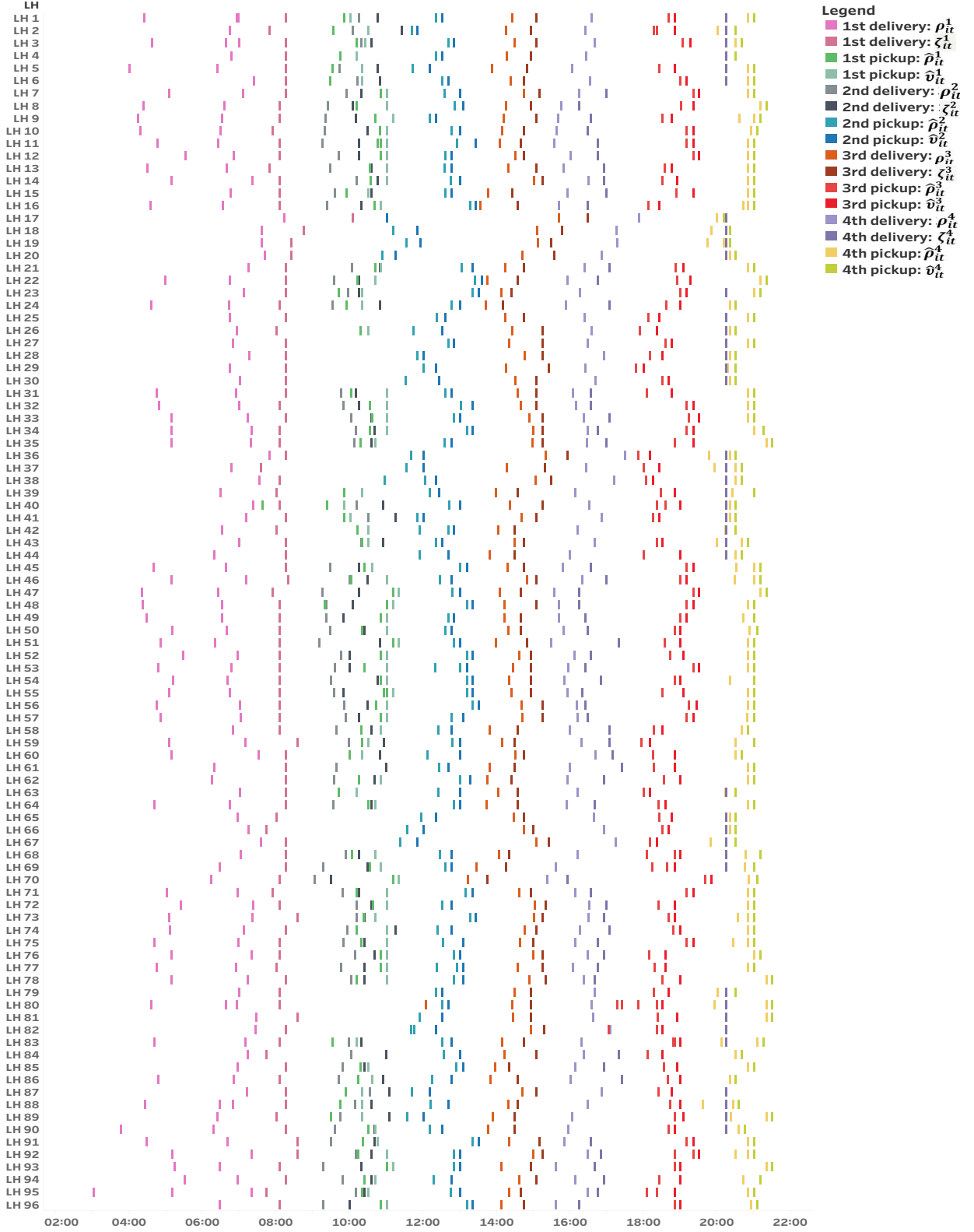
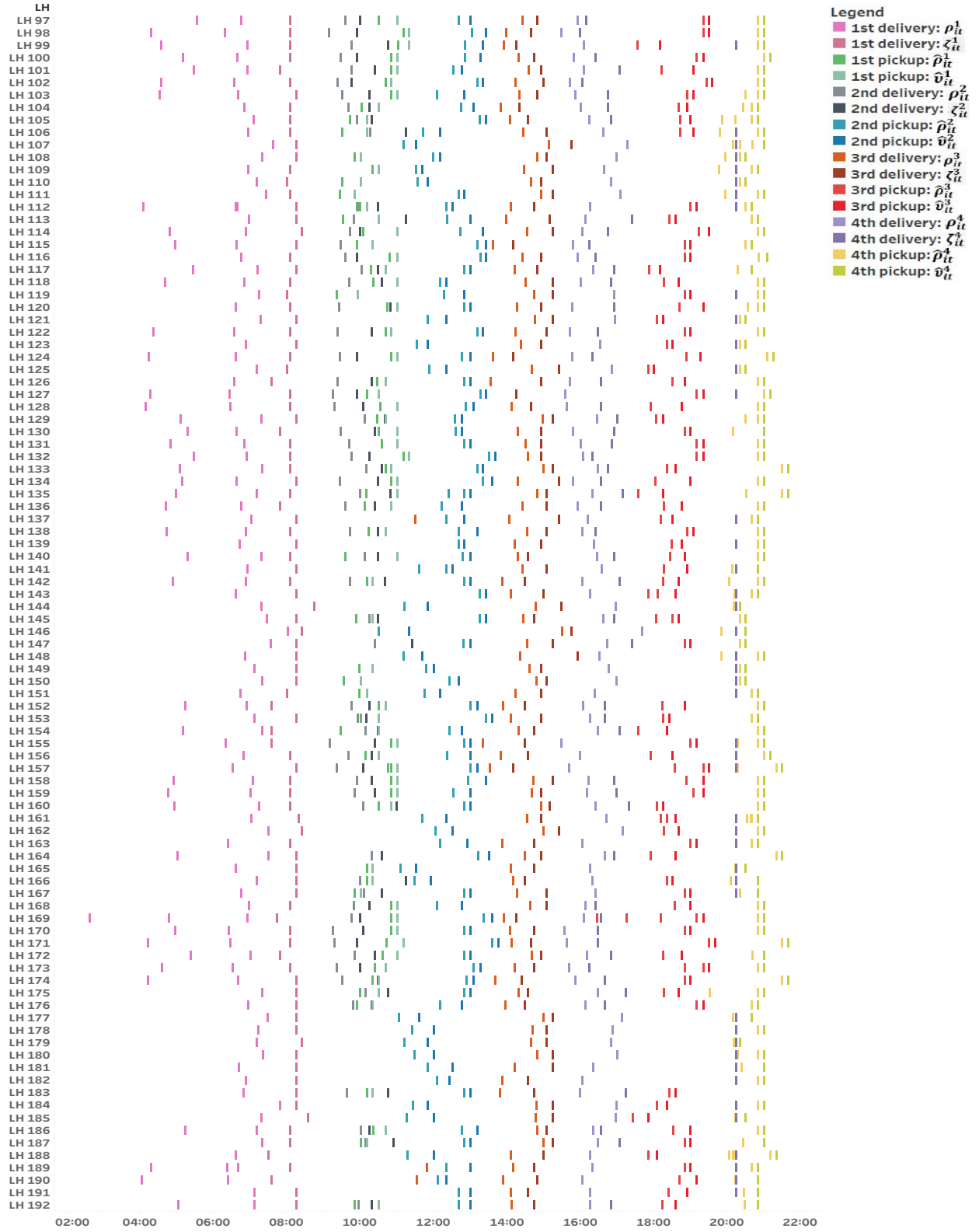


Figure 11 Multiple-Trip Route Schedule for I_1

Appendix E Delivery and Pickup Path Arrival Timing for I_1





(b) arrival timing for LH 97 ~ LH 192

Figure 12 Delivery and Pickup Path Arrival Timing for I_1

Appendix F Hub Geography for I_2 , I_3 , and I_4



Figure 13 Hub Geography for I_2

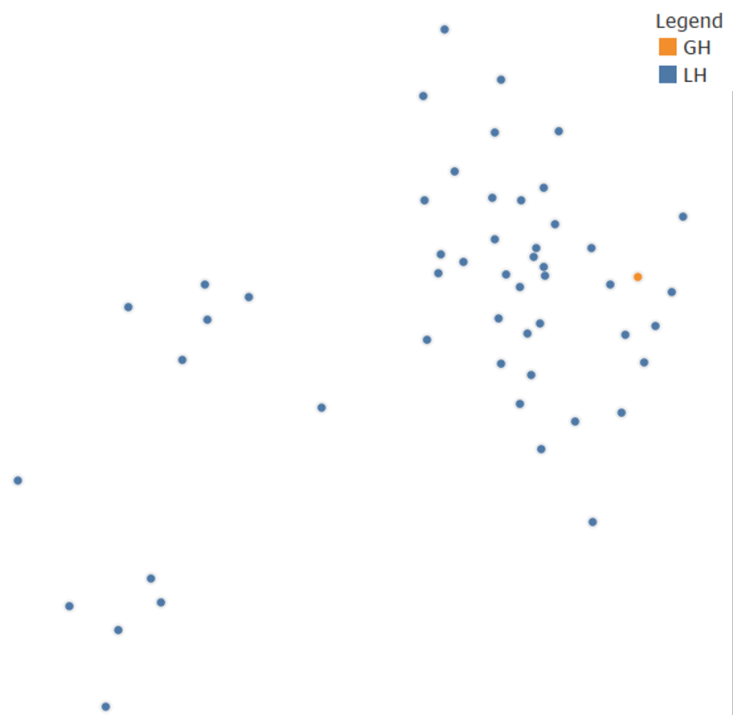
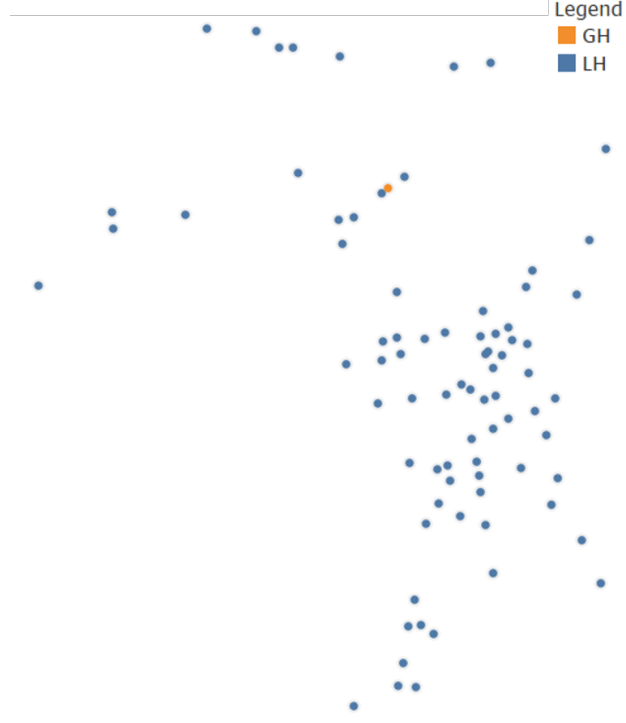


Figure 14 Hub Geography for I_3

Figure 15 Hub Geography for I_4

Appendix G Multi-Phase Decision Framework Results for I_2 , I_3 , and I_4

Table 8 Multi-Phase Decision Framework Results for I_2

phase	shift s	path type	$ P $	μ^s	τ^s	c^s	n^s	$ R $	C
1	1st delivery	delivery	114	91.89%	42.18	22.06	1.94	42	0.47×10^4
2	2nd delivery	delivery	36	35.39%	44.25	23.16	2.14	42	0.63×10^4
3	1st pickup	back-haul	2	66.80%	13.00	5.83	1.00	109	0.88×10^4
		stand-alone	102	80.15%	32.40	16.95	1.16		
4	3rd delivery	delivery	36	50.64%	44.25	23.16	2.14	109	1.03×10^4
5	2nd pickup	back-haul	10	20.37%	19.40	9.37	1.10	109	1.24×10^4
		stand-alone	74	23.98%	37.84	23.80	1.07		
6	4th delivery	delivery	36	65.05%	44.25	23.16	2.14	109	1.40×10^4
7	3rd pickup	back-haul	8	22.28%	21.63	10.01	1.25	109	1.58×10^4
		stand-alone	70	21.68%	35.94	21.61	1.14		
8	4th pickup	back-haul	0	—	—	—	—	109	1.74×10^4
		stand-alone	70	9.03%	33.90	19.45	1.23		
9	5th pickup	back-haul	0	—	—	—	—	109	2.06×10^4
		stand-alone	73	12.93%	36.79	22.05	1.21		
10	6th pickup	back-haul	0	—	—	—	—	109	2.08×10^4
		stand-alone	4	13.89%	50.00	26.09	1.50		

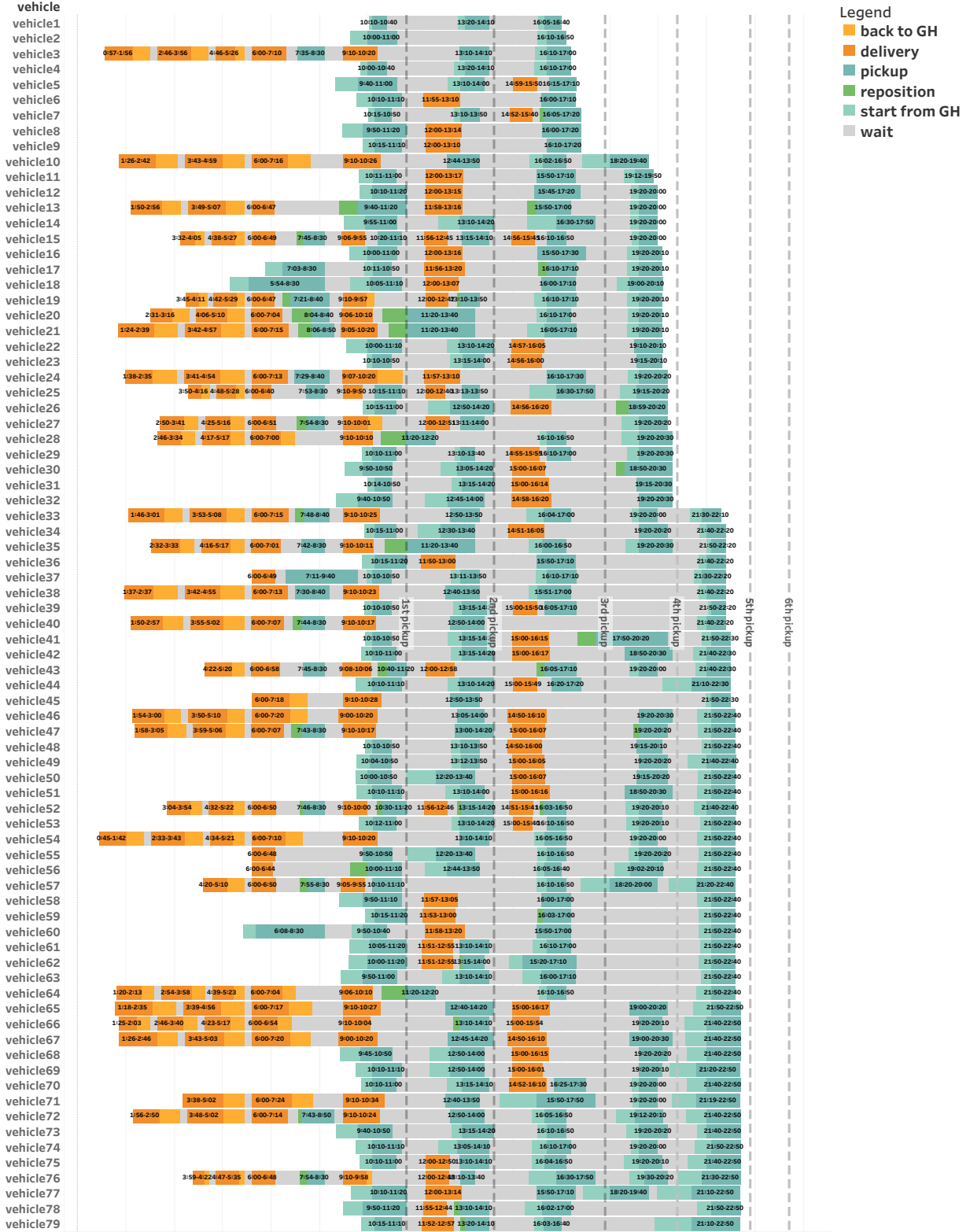
Table 9 Multi-Phase Decision Framework Results for I_3

phase	shift s	path type	$ P $	μ^s	τ^s	c^s	n^s	$ R $	C
1	1st delivery	delivery	90	79.28%	41.82	36.31	1.19	51	0.67×10^4
2	2nd delivery	delivery	46	28.86%	41.20	37.68	1.13	54	1.02×10^4
3	1st pickup	back-haul	0	—	—	—	—	69	1.27×10^4
		stand-alone	107	92.28%	35.86	29.51	1.14		
4	2nd pickup	back-haul	24	55.29%	54.63	44.92	1.75	69	1.45×10^4
		stand-alone	10	71.65%	66.40	65.64	1.70		
5	3rd delivery	delivery	25	37.49%	40.88	34.15	1.20	69	1.62×10^4
6	4th delivery	delivery	35	23.79%	38.69	35.86	1.11	69	1.87×10^4
7	5th delivery	delivery	22	13.66%	33.41	26.09	1.05	69	1.99×10^4
8	3rd pickup	back-haul	8	28.88%	32.38	31.78	1.25	69	2.14×10^4
		stand-alone	24	46.11%	49.50	41.50	1.71		
9	4th pickup	back-haul	0	—	—	—	—	69	2.14×10^4
		stand-alone	30	11.27%	40.17	30.89	1.40		
10	5th pickup	back-haul	0	—	—	—	—	69	2.48×10^4
		stand-alone	31	30.71%	69.61	62.80	2.03		

Table 10 Multi-Phase Decision Framework Results for I_4

phase	shift s	path type	$ P $	μ^s	τ^s	c^s	n^s	$ R $	C
1	1st delivery	delivery	83	78.56%	75.01	69.51	1.94	66	1.10×10^4
2	2nd delivery	delivery	38	37.25%	72.00	65.72	1.82	78	1.57×10^4
3	1st pickup	back-haul	31	97.28%	75.58	79.83	1.58	119	2.15×10^4
		stand-alone	78	98.21%	74.99	68.62	1.81		
4	3rd delivery	delivery	44	42.01%	72.93	69.15	1.80	142	2.72×10^4
5	4th delivery	delivery	44	17.01%	72.93	69.15	1.80	142	3.30×10^4
6	2nd pickup	back-haul	41	61.45%	76.66	75.36	1.83	142	3.36×10^4
		stand-alone	5	83.89%	117.20	125.48	2.40		
7	3rd pickup	back-haul	0	—	—	—	—	142	3.50×10^4
		stand-alone	45	11.67%	66.82	67.21	1.33		
8	4th pickup	back-haul	0	—	—	—	—	142	3.86×10^4
		stand-alone	31	40.16%	57.29	57.22	1.19		

Appendix H Multiple-Trip Route Schedule for I_2



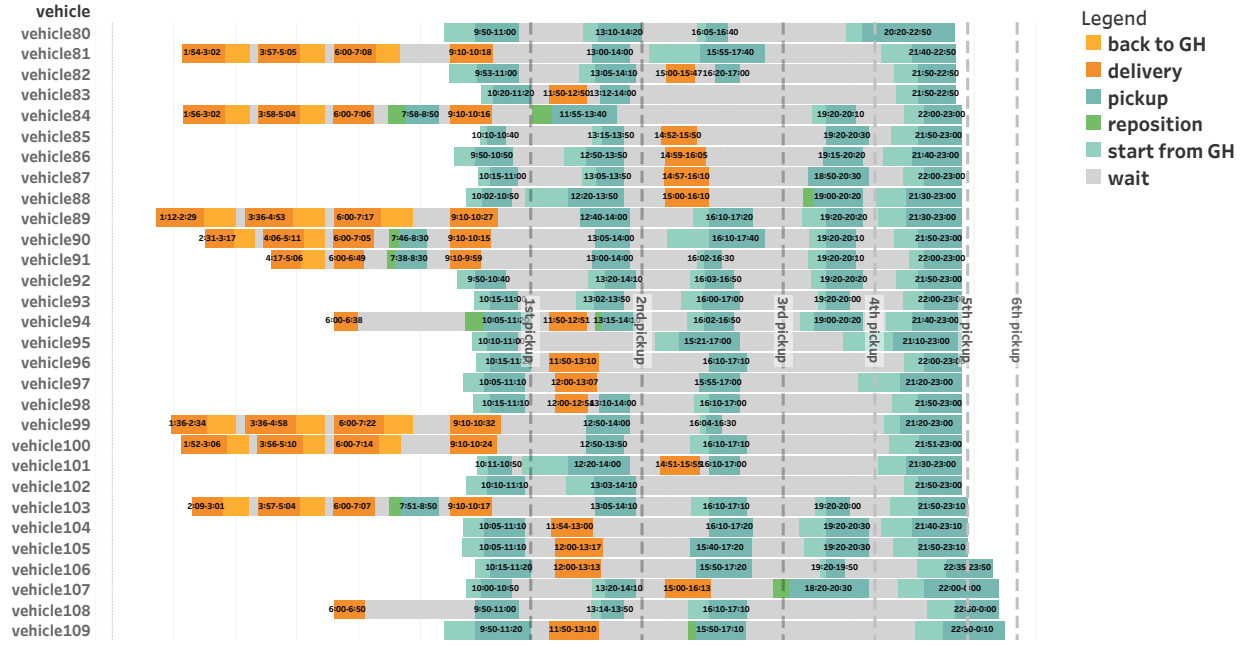


Figure 16 Multiple-Trip Route Schedule for I_2

Appendix I Multiple-Trip Route Schedule for I_3

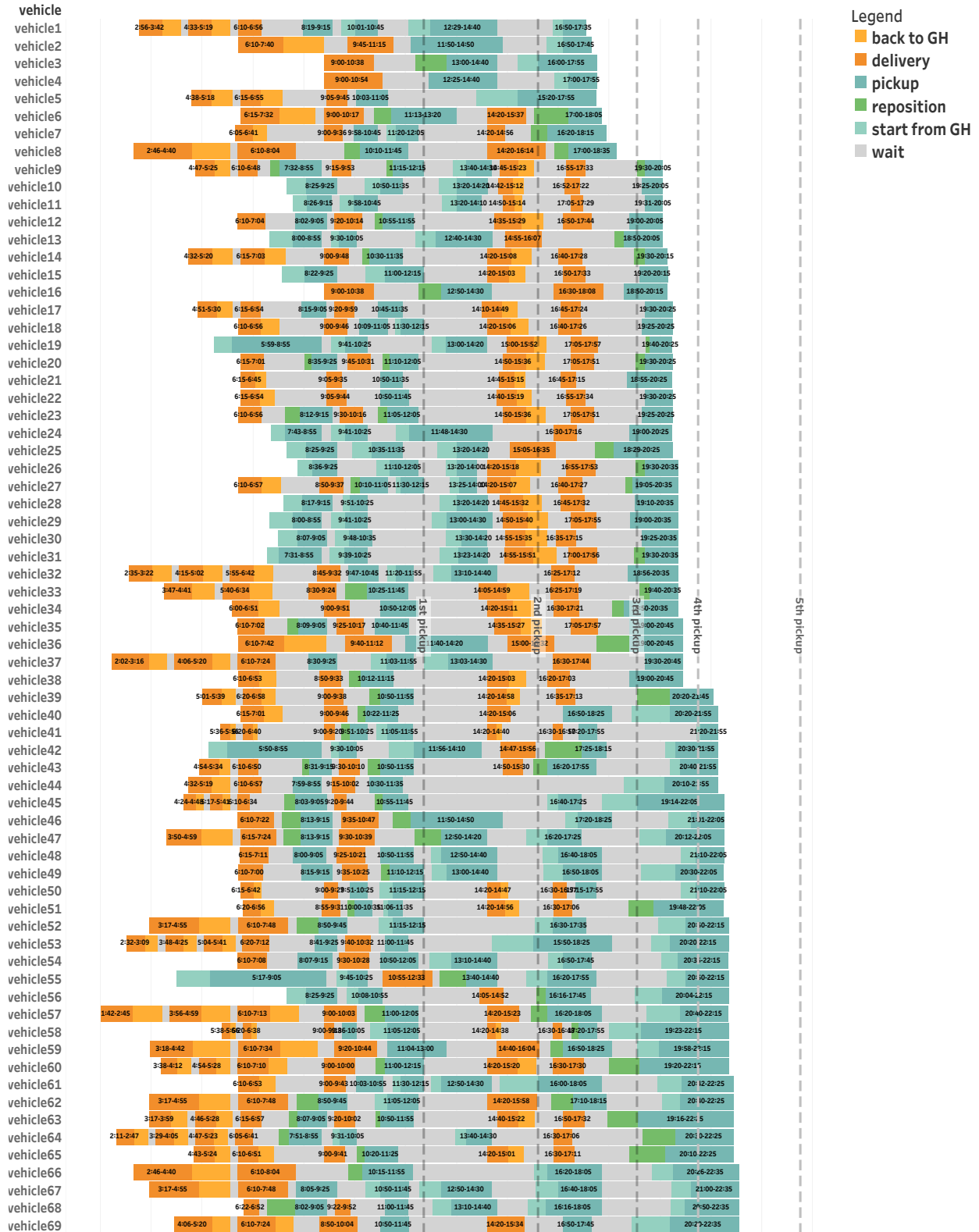
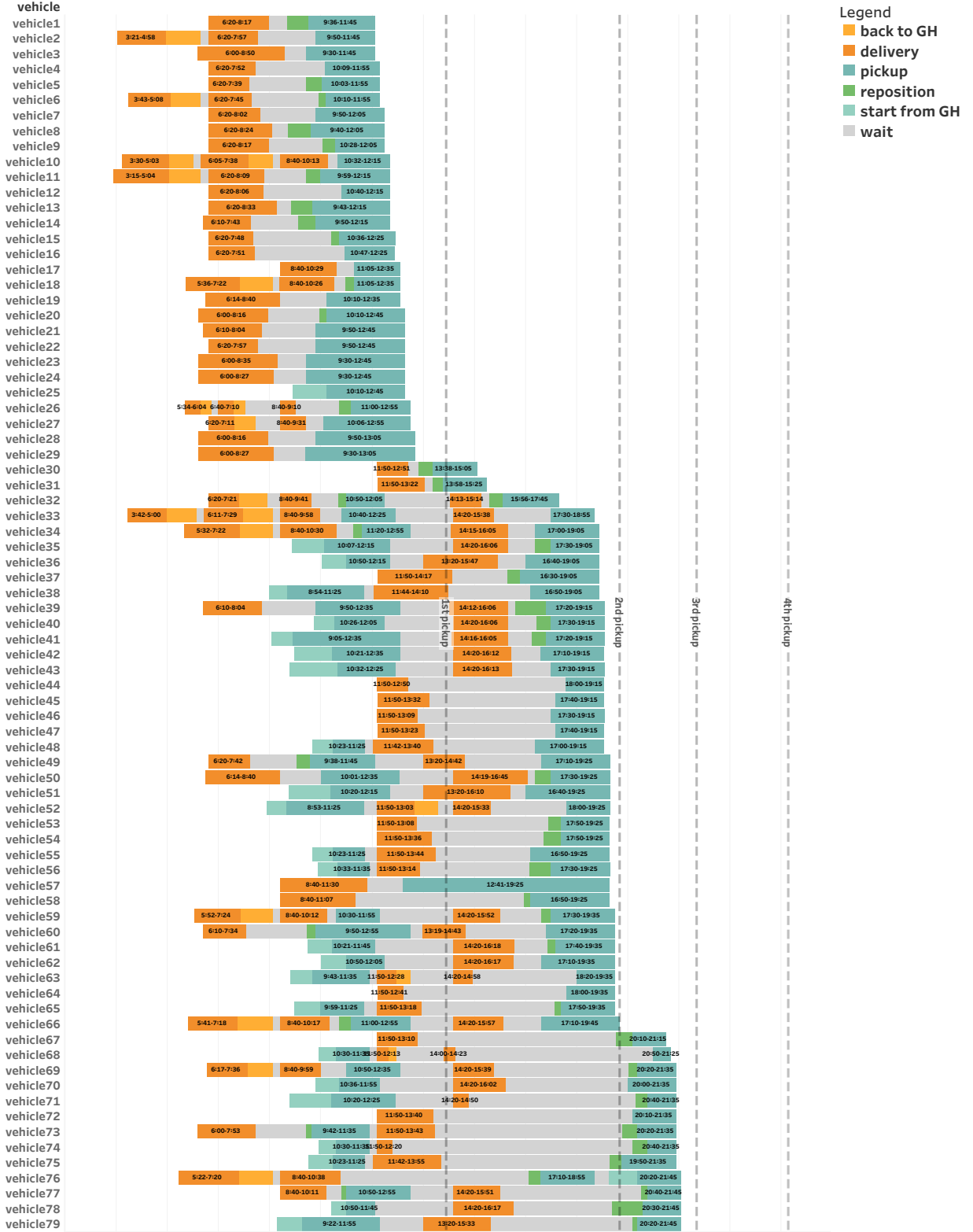
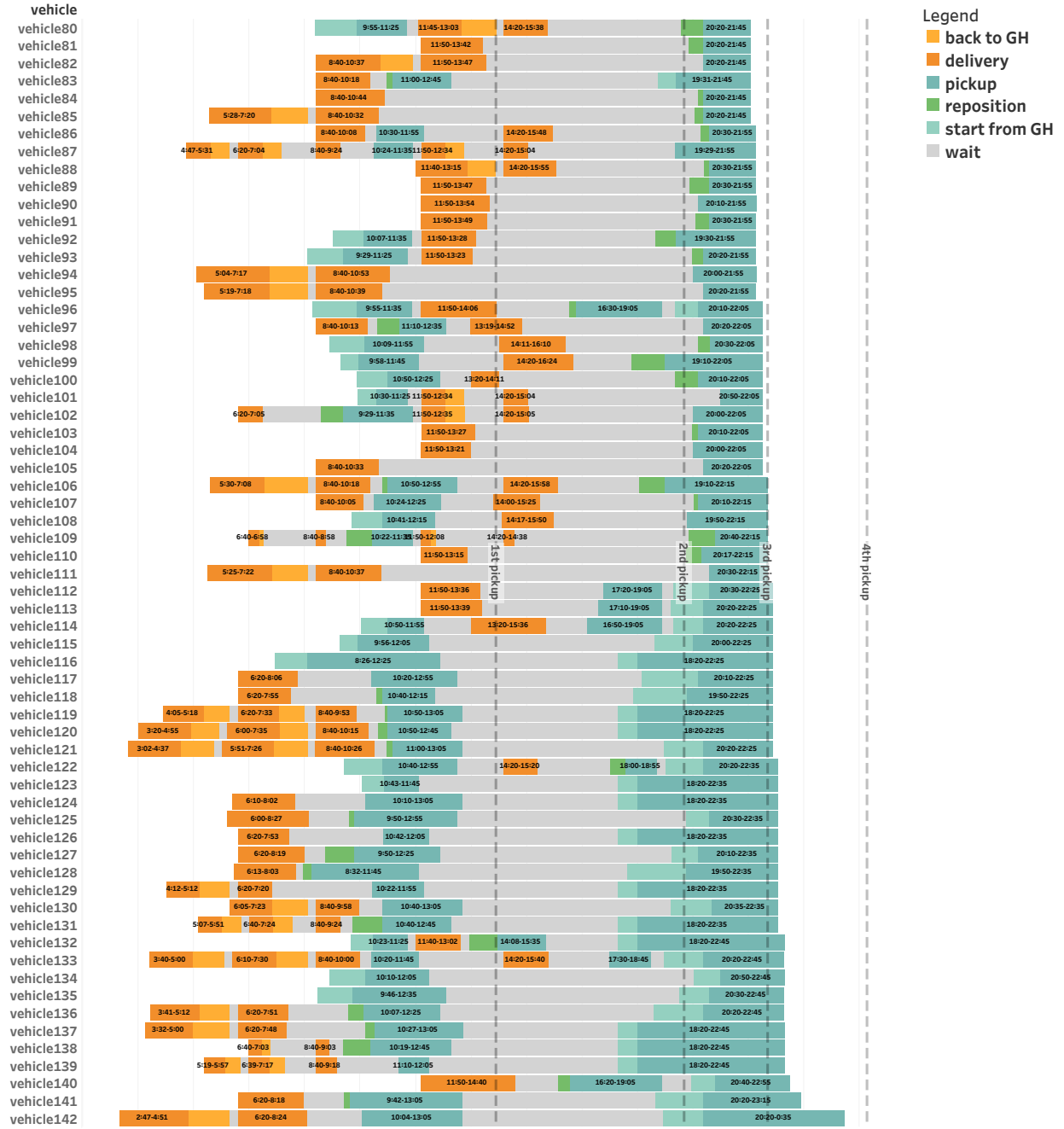


Figure 17 Multiple-Trip Route Schedule for I_3

Appendix J Multiple-Trip Route Schedule for I_4





(b) route schedule for vehicle 80 ~ vehicle 142

Figure 18 Multiple-Trip Route Schedule for I_4

Appendix K Notation

Table 11: Summary of Notation

$\{0\}$	gateway hub
N	set of local hubs
A	set of directed arcs
τ_{ij}	travel time of arc $a = (i, j)$
c_{ij}	travel cost of arc $a = (i, j)$
U	number of unloading docks at the GH
D	set of delivery demands
\hat{D}	set of pickup demands
T	set of end points of the intervals in the planning horizon
d_{it}	delivery demand for LH i that becomes available during the time interval ending at time t
q_{it}	number of packages for delivery demand d_{it}
w_{it}	weight of packages for delivery demand d_{it}
v_{it}	latest departure time for delivery demand d_{it}
ζ_{it}	due time for delivery demand d_{it}
\hat{d}_{it}	pickup demand that becomes available at LH i during the time interval that ends at time t
\hat{q}_{it}	number of packages for pickup demand \hat{d}_{it}
\hat{w}_{it}	weight of packages for pickup demand \hat{d}_{it}
\hat{v}_{it}	latest departure time for pickup demand \hat{d}_{it}
$\hat{\zeta}_{it}$	due time for pickup demand \hat{d}_{it}
S	set of delivery shifts
\hat{S}	set of pickup shifts
M	set of vehicle types
Q_m	maximum number of packages that a vehicle of type m can accommodate
W_m	maximum weight of packages that a vehicle of type m can accommodate
τ_m^l	loading time of a vehicle of type m
τ_m^u	unloading time of a vehicle of type m
N_m	set of LHs that can be visited by a vehicle of type m
ϕ	minimum units of time a vehicle has to remain at the last LH in delivery path before traveling to the first LH in the pickup path
$\hat{\phi}$	minimum units of time a vehicle has to remain at the GH after completing a trip before it can start a next trip
κ	maximum number of LHs visited in a delivery path
G	set of candidate delivery groups
G_i	set of candidate delivery group that include LH i
n_g^1	minimum number of vehicles required to serve the delivery demands for the LHs in group g
n_g^2	minimum number of trips required to serve the delivery demands for the LHs in group g
c_g	travel cost required to serve the delivery demands for the LHs in group g

y_g	binary decision variable representing whether group g is chosen or not
Δ	set of possible unload time intervals in the pickup shifts
$\Delta^{\hat{s}}$	set of possible unload time intervals in pickup shift \hat{s}
τ_i^*	critical time at LH i
$\hat{\kappa}_b^{\hat{s}}$	maximum number of LHs visited by a back-haul pickup path in pickup shift \hat{s}
$\hat{\kappa}_a^{\hat{s}}$	maximum number of LHs visited by a stand-alone path in pickup shift \hat{s}
P	set of delivery paths
\hat{P}	set of candidate back-haul pickup paths
n_b	number of LHs visited in candidate back-haul pickup path b
τ_δ	start time of interval δ
\hat{P}_d	set of candidate pickup paths that can serve as back-haul for delivery path d
$\Delta_b^{\hat{s}}$	set of time intervals in pickup shift \hat{s} during which candidate back-haul pickup path b can unload
\hat{P}_i	set of back-haul pickup paths that visit LH i
λ_b	binary variable indicating whether back-haul pickup path b is selected
η_{bd}^δ	binary variable indicating whether path b serves as back-haul for delivery path d and unloads during δ
U^δ	number of stand-alone pickup paths that can be accommodated in time interval δ
\hat{P}^δ	set of candidate stand-alone pickup paths that unload during time interval δ
d_p	size of the demands served by pickup path p
N^p	set of LHs visited by pickup path p
ψ_{ip}^1	parameter indicating whether pickup path p visits LH i at a desirable time
d_{ip}	size of the demands available at LH i at the time that pickup path p visits LH i
ψ_p^1	first evaluation metric
ψ_p^2	second evaluation metric
\hat{P}_i^δ	set of enumerated pickup paths that visit LH i and unload during time interval δ
z_p	binary variable indicating whether pickup path p is chosen
N^δ	set of LHs visited by the chosen pickup paths that unload during time interval δ
q_i'	number of the packages picked up at LH i
w_i'	weight of the packages picked up at LH i
e_i'	earliest time packages are available at LH i
ℓ_i'	latest time packages can be at LH i
$ \Delta^{\hat{s}} $	length of pickup shift \hat{s}
Q	maximum number of packages for (smallest) vehicle type
W	maximum weight for (smallest) vehicle type
τ^1	loading time for (smallest) vehicle type
x_{ij}	binary decision variable indicating whether a vehicle travels directly from hub i to hub j
ρ_i	continuous variable indicating the arrival time at hub i
α_i	continuous variable indicating the number of packages on the vehicle before it visits hub i
β_i	continuous variable indicating the weight of packages on the vehicle before it visits hub i

P	set of delivery and pickup paths for the shift
R'	set of existing partial multi-trip routes
R	set of enumerated partial multi-trip routes
R_p	set of enumerated partial routes that include path p
$R_{r'}$	set of enumerated partial routes that include existing partial route r'
c_r	travel cost of enumerated partial route r
$\tilde{\Delta}_r$	set of feasible sets of time intervals during which the pickup paths in the enumerated partial route r can unload
$\tilde{\Delta}_\delta$	set of feasible sets of time intervals that include δ
ξ_r	binary variable indicating whether enumerated partial multi-trip route r is chosen
θ_r^σ	binary variable indicating whether the pickup paths in enumerated partial multi-trip route r use the set of unloading time intervals σ