

The Continuous Time Inventory Routing Problem

Felipe Lagos Natasha Boland Martin Savelsbergh

H. Milton Stewart School of Industrial and Systems Engineering

Georgia Institute of Technology

Atlanta, Georgia 30332

Abstract

We consider a continuous time variant of the Inventory Routing Problem in which the maximum quantity that can be delivered at a customer depends on the customer's storage capacity and product inventory at the time of the delivery. We investigate critical components of a dynamic discretization discovery algorithm and demonstrate in an extensive computational study that these components are sufficient to produce provably high-quality, often optimal, solutions.

1 Introduction

The Inventory Routing Problem (IRP) integrates inventory management, vehicle routing, and delivery scheduling decisions. The IRP arises in the context of Vendor Managed Inventory (VMI), in which a supplier makes the replenishment decisions for products delivered to its customers. The variant of interest in this paper was introduced in the seminal paper by Bell et al. (1983). Critical characteristics of this variant are that only transportation costs are considered and that the system evolves in continuous time. That is, the amount of product that can be delivered at a customer at a particular point in time depends on the storage capacity and inventory at that point in time (which depends on the initial inventory, the product usage rate and the time elapsed since the start of the planning period, and the amount of product delivered since the start of the planning period). As a consequence, delivery times have to be scheduled carefully and vehicle travel times have to be accounted for accurately. This contrasts with the majority of the variants of the IRP considered in the literature, where the planning horizon is partitioned into periods and it is assumed that delivery routes take place at the start of the period, product consumption takes places at the end of the period, and that both happen instantaneously. For more comprehensive introductions to and discussions of the IRP, see Bertazzi et al. (2008) and Coelho et al. (2014).

The variant considered in this paper (and by (Bell et al. 1983)) is motivated by the IRP encountered by companies in the liquid gas industry, e.g., Air Liquide (www.airliquide.com), Air Products (www.airproducts.com), and Praxair (www.praxair.com). These companies produce liquefied gases, e.g., liquid oxygen, liquid nitrogen, or liquid argon, and install tanks

at their customers' premises and guarantee minimum product availability at any time, Customers use (consume) product at a certain rate (which can differ at different times of the day) often 24 hours per day (e.g., liquid oxygen in hospitals.) Thus, the amount of product that can be delivered to the tank changes at the same rate. The companies continuously monitor product usage and tank inventory levels so that they can produce cost-effective delivery schedules that meet their service commitments (i.e., the guaranteed minimum product availabilities). In practice, the companies tend to have customers that require multiple deliveries per day as well as customers that require only one or two deliveries per week. As a consequence, the use of a continuous time variant of the IRP is most appropriate in these settings, i.e., provides the most accurate representation of the system. Also relevant is the fact that the company contracts typically specify that the customers own/purchase the product upon delivery, which means that the companies do not have to consider product holding costs at the customer. This variant of the IRP has attracted attention in the past, e.g., Campbell et al. (1998), Campbell et al. (2002), and Campbell and Savelsbergh (2004a,b), and, more recently, was considered interesting and challenging enough to form the ROADEF/EURO 2016 Challenge (for more information, see www.roadef.org/challenge/2016/en/sujet.php) with real-life data provided by Air Liquide.

Solving instances of any variant of the IRP is challenging (Coelho and Laporte 2013). Integer programming techniques have been used for the “period” variants, e.g., branch-and-cut (Coelho and Laporte 2014, Avella et al. 2017) and branch-and-cut-and-price (Coelho and Laporte 2014, Desaulniers et al. 2015). The most advanced and successful of these can now solve instances with up to 50 customers and up to 5 vehicles. For the “continuous time” variant, no optimization algorithms exist, to the best of our knowledge, but lower bounding techniques have been developed in Song and Savelsbergh (2007).

As we shall find in this study, modeling inventory in continuous time and accurately modeling vehicle travel times makes for a particularly challenging IRP. Indeed, optimization problems over continuous time, in general, have been found to be difficult to solve to optimality. Compact models that use continuous variables to model time have weak linear programming (LP) relaxations. Their solution with current integer programming solver technology is limited to only small instances. Extended formulations, with binary variables indexed by time, have much stronger relaxations, but (tend to) have a huge number of variables. Such formulations rely on a *discretization* of time, which introduces approximation. Recently, Boland et al. (2017) introduced a dynamic discretization discovery algorithm for solving the continuous time service network design problem, which uses extended integer programming formulations. The key to the approach is that it discovers exactly which times are needed to obtain an optimal, continuous-time solution, in an efficient way, by solving a sequence of (small) integer programs. The integer programs are constructed as a function of a subset of times, with variables indexed by times in the subset. They are carefully designed to be tractable in practice, and to yield a lower bound (it is a cost minimization problem) on the optimal continuous-time value. Once the *right* (very small) subset of times is discovered, the resulting integer programming model yields the continuous-time optimal value.

In this paper, we explore and demonstrate the potential of dynamic discretization dis-

covery algorithms for the continuous time variant of the IRP, *CIRP*. The aim of our research is twofold. First, we want to develop an optimization algorithm for this important variant of the IRP, and by doing so hope to stimulate others to take up this challenge as well. Second, we want to demonstrate that dynamic discretization discovery algorithms can be developed and useful for problems other than service network design, and by doing so, again, hope to stimulate others to start using and advancing this approach.

Our contributions in this paper are both theoretical and algorithmic. We

- investigate the problem of minimizing the number of vehicles needed to obtain a feasible delivery plan, showing that it is strongly NP-hard, but that in the case of a single customer, it can be solved in pseudo-polynomial time,
- develop a mixed integer programming model for the CIRP over a given, uniform, discretization of time,
- prove that if the data for an instance is rational, then it has an optimal solution in which all delivery times at customers are rational, which yields, as a consequence, that the mixed integer programming model can, in theory, provide optimal solutions to the CIRP, by taking the discretization corresponding to these rationals,
- adapt the mixed integer program to yield lower bounds on the optimal CIRP value, which can be achieved in practice by using a sufficiently coarse discretization,
- propose model enhancements that strengthen both the resulting lower bound and the mixed integer programming formulation itself,
- develop two alternative approaches to finding feasible solutions, one using an adaptation of the mixed integer program and the other based on repairing solutions to the lower bound model, and
- carry out a computational study to assess the performance of these ideas, in practice.

Our study shows that with only a partial implementation of a dynamic discretization discovery algorithm, we have been able to optimally solve instances with up to 15 customers and have been able to obtain provably high-quality solutions for many others. Even though these results are notable, they also point to areas for further research and improvement.

The remainder of the paper is organized as follows. In Section 2, we formally introduce the continuous time inventory routing problem and some of the characteristics that distinguish it from period-based inventory routing problems. In Section 3, we highlight the challenges associated with determining the minimum number of vehicles required to produce a feasible delivery plan. In Section 4, we provide a mixed integer programming formulation for the continuous time inventory routing problem over a given discretization. In Section 5, we discuss how the mixed integer programming formulation can be modified to produce a lower bound on the cost of an optimal delivery plan. In Section 6, we outline two approaches for constructing feasible delivery plans. In Section 7, we present and discuss the results of an

extensive computational study. Finally, in Section 8, we describe the next steps towards a full-scale dynamic discretization discovery algorithm for the continuous time inventory routing problem.

2 The Continuous Time Inventory Routing Problem

We consider a vendor managed resupply environment in which a company manages the inventory of its customers, resupplying a single product from a single facility.

Each customer i in the set $N = \{1, \dots, n\}$ of customers has local storage capacity $C_i > 0$, uses product at a constant rate $\hat{u}_i > 0$, and has initial inventory $I_i^0 > 0$ at the start of the planning period. Note that a customer uses product at a given rate, i.e., a customer consumes a certain amount of product per unit of time. The planning horizon is $H > 0$. The company employs a fleet of m homogeneous vehicles, each with capacity $Q > 0$, to deliver product to its customers. A delivery route has to start and end at the company's facility and has to be completed before the end of the planning horizon. Unlike many inventory routing problem settings, here vehicle routes are not restricted to start and end in a single time period. Indeed, the setting we study here is based on continuous time, not subdivided into periods. Travel times $\hat{\tau}_{ij} > 0$ and travel costs $c_{ij} \geq 0$ between every pair of locations i and j , $i \neq j$, for $i, j \in N_0 = \{0, 1, \dots, n\}$, where 0 denotes the location of the company's facility, are known. Travel times are assumed to satisfy the triangle inequality. There is no inventory holding cost.

We allow a vehicle to wait at a customer location and to make multiple deliveries while at the customer's premises. This may be beneficial as it allows delivery of more than the customer's storage capacity without the need for an intervening trip to the depot. (We provide an illustration of this point later.) In practice, a customer may have sufficient space for several vehicles to wait at their premises, but usually at most one vehicle can deliver product at a time. Although the time needed for a vehicle to deliver product may depend on the quantity to be delivered, there is usually a substantial overhead time needed to engage and disengage the delivery equipment. Thus the delivery time can be reasonably well approximated by a constant (possibly customer-dependent) length of time. This situation can be modeled by the use of two locations for each customer, one for parking and one for making a delivery at the customer, with the latter constrained to allow at most one vehicle at a time. However, given the complexity of the ideas we wish to discuss in this paper, we make the simplifying assumption that all locations have the single-vehicle constraint: we assume that it is *not* possible for multiple vehicles to visit the same customer at the same time. We further assume that product delivery at a customer site is instantaneous. It is not difficult to extend the ideas we present here to account for multiple vehicles waiting at a customer and constant delivery time. We also assume that there is no cost for waiting. (In the liquid gas industry, drivers are salaried employees, so their time, for the purpose of this model, can be considered a sunk cost.) Finally, we can assume, without loss of generality, that customers are served by a finite set of deliveries, occurring at a finite number of time points during the planning period.

Vehicles are assumed to be at the company's facility at the start of the planning period and have to be back at the company's facility at the end of the planning period. However, vehicles can make multiple trips during the planning period. We assume that the loading of a vehicle at the company facility is instantaneous.

We assume that the company does not incur any holding cost for product, either at the company facility or at any of the customer sites. We also assume that the company facility always has sufficient product to supply customers; it does not have either production or storage capacity constraints.

The goal is to find a minimum cost delivery plan that ensures that none of the customers runs out of product during the planning period. A delivery plan specifies a set of vehicle itineraries, each of which consists of a sequence of trips/routes that start and end at the company facility within the time horizon, to be performed by a single vehicle. Each route specifies a departure time from the facility, a sequence of customer deliveries, and, for each, the delivery time and quantity delivered to the customer at that time. We refer to this problem as the Continuous Time Inventory Routing Problem (CIRP).

We start by presenting some observations about optimal solutions to instances of the CIRP.

- There exist instances of the CIRP in which it is beneficial to visit a customer more than once on a delivery route. Consider, for example, an instance with two customers, storage capacities $C_i = 2$ for $i = 1, 2$, usage rates $u_i = 1$ for $i = 1, 2$, initial inventories $I_1^0 = 1$ and $I_2^0 = 2$, a single vehicle with capacity $Q = 5$, travel times and costs $\hat{\tau}_{01} = c_{01} = \hat{\tau}_{12} = c_{12} = 1$ and $\hat{\tau}_{02} = c_{02} = 2$, and a time horizon $H = 4$. The optimal solution has a single route, of cost 4, visiting Customer 1 at time 1, delivering 2 units of product, visiting Customer 2 at time 2, delivering 2 units of product, and visiting Customer 1 again at time 3, delivering 1 unit of product; see Figure 1. This must be optimal, since any feasible solution must visit Customer 2 at least once, at a cost of 4.

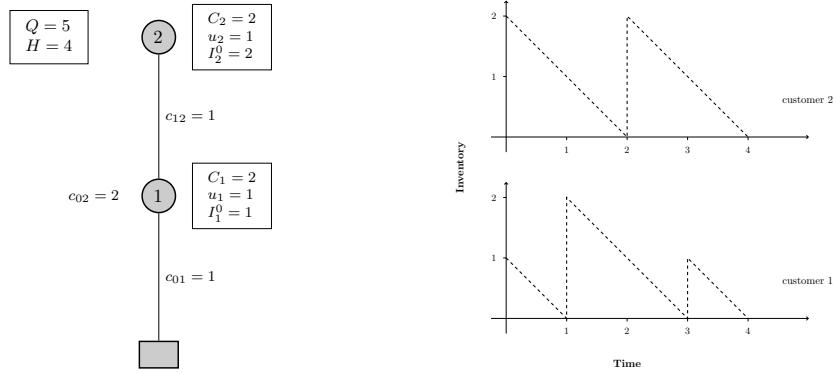


Figure 1: Instance for which it is optimal to visit a customer more than once on a route.

- A similar example, with only the single customer, Customer 1, illustrates that an optimal solution may require the vehicle to wait at the customer to make multiple

deliveries. With the same parameter values as above, but without Customer 2, any optimal solution is a single route, visiting Customer 1 at time 1 to deliver some product, for example, 2 units, and then waiting, for example, until time 2, to deliver the 1 unit remaining of the total 3 units required; see Figure 2.

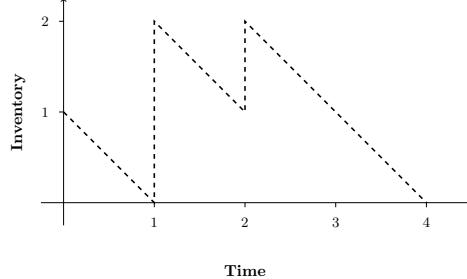


Figure 2: Instance for which it is optimal to waiting at a customer and make multiple deliveries.

- There exist instances of the CIRP in which all parameters are integer valued (i.e., capacities, usage rates, initial inventories, and travel times), but for which there exists no optimal solution that delivers product at customers only at integer times. Consider, for example, an instance with two customers, having storage capacities $C_1 = 7$, $C_2 = 5$, usage rates $\hat{u}_i = 2$, $i = 1, 2$, and initial inventories $I_1^0 = 3$, $I_2^0 = 5$. There is a single vehicle with capacity $Q = 12$. The travel times and costs are identical and symmetric: $\hat{\tau}_{01} = c_{01} = \hat{\tau}_{12} = c_{12} = \hat{\tau}_{02} = c_{02} = 1$. The time horizon is $H = 5$. The optimal solution has a single route of cost 3, visiting Customer 1 at time 1.5, delivering 7 units of product, and visiting Customer 2 at time 2.5, delivering 5 units; see Figure 3. This solution is optimal because any feasible solution must visit each customer at least once, and the cheapest way to do this is with a single route. There cannot be an optimal solution in which the deliveries take place only at integer times, since Customer 1 must have a delivery on or before time 1.5, when it runs out of product, and at time 1, Customer 1 only has capacity for 6 units of product. If the remaining 1 unit it requires is to be delivered without incurring extra cost, the vehicle must wait at Customer 1 until time 2 to deliver this unit, at which time it is too late to reach Customer 2 by time 2.5, when Customer 2 runs out of product. Thus the solution with delivery to Customer 1 and time 1.5 and Customer 2 and time 2.5 is the unique optimal solution.

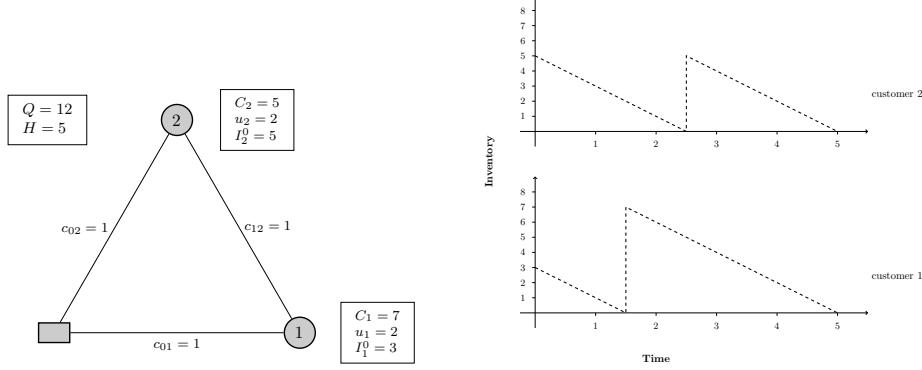


Figure 3: Instance with integer data for which the optimal solution has non-integer delivery times.

The last example illustrates the challenge in using discretization to approach the CIRP. An approximation in which vehicle departure and delivery times are restricted to integers would yield an optimal solution costing twice that of the true, continuous time optimal solution. Even an approximation with a discretization into periods of length 0.2, needing 15 time periods, still yields a factor of two error in the optimal value. To obtain the optimal continuous time solution in this case one may either use 6 periods of length 0.5, or 30 periods of length 0.1. This observation highlights another challenge: clearly the quality of the approximation from discretization is not monotonically improving in the granularity of the discretization.

3 Minimizing the number of vehicles

Considering the minimum number of vehicles required to produce a feasible delivery plan also reveals the complexity of the CIRP, as shown in the next two propositions.

Proposition 1. *The problem of finding the minimum number of vehicles required to produce a feasible delivery plan for a CIRP instance is strongly NP-hard.*

Proof. We show this with a reduction from the bin packing problem (BPP). Consider a BPP instance with a set of items $\{1, \dots, n\}$, each with size a_i for $i = 1, \dots, n$, and a set of bins $\{1, \dots, m\}$, each with capacity V . Without loss of generality, we may assume $2 \leq a_i \leq V$ and integer for $i = 1, \dots, n$.

Let the corresponding CIRP instance have time horizon $H = V$ and a fleet of homogeneous vehicles with capacity $Q = n$. For each item $i \in \{1, \dots, n\}$ in the BPP instance, there is a corresponding customer $i \in N$ in the CIRP instance. The travel time $\hat{\tau}_{0i}$ from the depot to customer i is $\frac{1}{2}a_i$ for $i \in N$. The travel time $\hat{\tau}_{ij}$ from customer i to customer j is $\hat{\tau}_{ij} = \hat{\tau}_{i0} + \hat{\tau}_{0j}$ for $i, j \in N, i \neq j$. Furthermore, for each customer $i \in N$, let the storage capacity be $C_i = H$, the initial inventory be $I_i^0 = H - 1$, and the usage rate be $u_i = 1$. Note that this implies that all customers have to be visited at least once.

Let m^* be the minimum number of vehicles required to produce a feasible delivery plan for the CIRP instance. Let $S_k \subseteq N$ be the set of customers visited by the k^{th} vehicle, $k = 1, \dots, m^*$. For each $k = 1, \dots, m^*$, let the items corresponding to the customers in S_k be assigned to Bin k . Because the total travel time for k^{th} vehicle is $\sum_{i \in S_k} 2\hat{\tau}_{0i} = \sum_{i \in S_k} a_i \leq H = V$, this assignment is feasible. Thus, there exists a solution to the bin packing problem with m^* bins. Conversely, let m^* be the number of bins in an optimal solution to the BPP instance and let S_k be the items assigned to Bin k for $k = 1, \dots, m^*$. The corresponding CIRP solution in which the customers in S_k are visited by vehicle k is feasible (note that by the definition of the travel times, the order in which customers are visited is immaterial). \square

It is not unexpected that finding the minimum number of vehicles required to produce a feasible delivery plan for a CIRP instance is NP-hard, but that finding the minimum number of vehicles required to produce a feasible delivery plan for a single-customer CIRP instance is also non-trivial may come as a surprise. Below, we provide a pseudo-polynomial time algorithm for finding the minimum number of vehicles required to produce a feasible delivery plan for a single-customer CIRP instance. It is still an open question whether a polynomial time algorithm exist, although we conjecture that it does.

In the remainder of this section, we use τ to denote the travel time from the depot to the (single) customer, u to denote the usage rate of the customer, and I to denote the initial inventory of the customer. If there is no limit on the number of vehicles available, then the problem has a feasible solution if and only if $C \geq I \geq \tau u$ and either $I \geq Hu$ or $H \geq 2\tau$.

Lemma 1. *If a single-customer CIRP instance has a feasible solution in which m vehicles make a total of n visits to the customer, then there is a feasible solution in which the vehicles make the trips in round-robin fashion, so Vehicle 1 makes trips $1, m+1, 2m+1, \dots$, Vehicle 2 makes trips $2, m+2, 2m+2, \dots$, etc. In general, Vehicle k makes the $(rm+k)$ th trip for $r = 0, 1, 2, \dots \lfloor (n-k)/m \rfloor$.*

Proof. Since there is only a single customer, all the trips are the same, depot-customer-depot, constituting a single visit to the customer, so we only have to assign the m vehicles to the n trips in the feasible solution. Furthermore, the vehicles are homogeneous, so any vehicle may be assigned to a trip without changing the quantity delivered on the trip, nor the time needed to complete it. (All vehicles require time τ to get to the customer and time τ to return from it.) Second, recall that no more than one vehicle can visit the customer at the same time, so no trip will arrive at the customer until after the preceding trip has departed, and trips are completely ordered in time.

Starting with the first trip in the solution, we may thus, without loss of generality, assign it to the first vehicle. Then we assign the second trip to the second vehicle, and so on. The first vehicle returns to the depot before the second vehicle returns (the second vehicle arrives at the customer only after the first vehicle finishes its delivery), the second vehicle returns to the depot before the third one, etc. Once we have assigned the first m trips, we consider a second trip for each vehicle, starting with the first vehicle. Since this vehicle was the first vehicle to return to the depot, it must be available for the $(m+1)$ th trip. Then, if needed, we assign a third trip, and so on until we have assigned all trips to vehicles. (We refer to

this process as “round-robin”). Thus, any feasible solution always has an ordered sequence of vehicles (from 1 to m) and each of them has a “similar” number of trips to perform. If m divides evenly into n , every vehicle performs exactly n/m trips; otherwise, $n \bmod m$ vehicles perform $\lceil n/m \rceil$ trips and the remainder perform $\lfloor n/m \rfloor$ trips. \square

Lemma 2. *Whether or not there is a feasible solution to a single-customer CIRP instance in which m vehicles make a total of n visits to the customer can be determined by solving a linear program.*

Proof. We can use the following linear program to determine whether a feasible solution with m vehicles making a total of n visits exists. By Lemma 1, we may assume that the vehicles make the visits in round-robin fashion: we let $\bar{m} = n \bmod m$ and let $\ell_v = \lfloor \frac{n}{m} \rfloor + 1$ if $v \leq \bar{m}$ and $\ell_v = \lfloor \frac{n}{m} \rfloor$ if $v > \bar{m}$, so that ℓ_v denotes the last visit for vehicle v ($v = 1, \dots, m$). Furthermore, let

- t_{vk}^- be the arrival time of the k^{th} visit of vehicle v ,
- t_{vk}^+ be the departure time of the k^{th} visit of vehicle v ,
- q_{vk} be the quantity delivered during the k^{th} visit by vehicle v ,
- I_{vk}^- be the inventory at the arrival time of the k^{th} visit of vehicle v ,
- I_{vk}^+ be the inventory at the departure time of the k^{th} visit of vehicle v , and
- ζ models the minimum time between visits by different vehicles at any customer.

Then the following linear program achieves our goal:

$$\begin{aligned}
& \max \zeta && (1a) \\
& t_{11}^- \geq \tau && \\
& t_{\bar{m}, \ell_{\bar{m}}}^+ \leq H - \tau && (1b) \\
& t_{vk}^- \geq t_{v-1, k}^+ + \zeta && v = 2, \dots, m, k = 1, \dots, \ell_v && (1c) \\
& t_{1k}^- \geq t_{m, k-1}^+ + \zeta && k = 2, \dots, \ell_1 && (1d) \\
& t_{vk}^- \geq t_{v, k-1}^+ + 2\tau && v = 1, \dots, m, k = 2, \dots, \ell_v && (1e) \\
& t_{vk}^+ \geq t_{vk}^- + \frac{\max\{I_{vk}^- + q_{vk} - C, 0\}}{u} && v = 1, \dots, m, k = 1, \dots, \ell_v && (1f) \\
& I_{vk}^+ = I_{vk}^- + q_{vk} - (t_{vk}^+ - t_{vk}^-)u && v = 1, \dots, m, k = 1, \dots, \ell_v && (1g) \\
& I_{vk}^- = I_{v-1, k}^+ + (t_{vk}^- - t_{v-1, k}^+)u && v = 1, \dots, m, k = 1, \dots, \ell_v && (1h) \\
& I_{1k}^- = I_{m, k-1}^+ + (t_{1k}^- - t_{m, k-1}^+)u && k = 2, \dots, \ell_1 && (1i) \\
& \sum_{v, k} q_{vk} \geq Hu - I && && (1j) \\
& I_{vk}^- \geq 0 && v = 1, \dots, m, k = 1, \dots, \ell_v. && (1k)
\end{aligned}$$

Constraint (1a) ensures that the vehicle making the first delivery can reach the customer before the start of the first delivery. Constraint (1b) ensures that the vehicle making the last delivery can return to the depot after completing the last delivery. Constraints (1c) and (1d) ensure that deliveries do not overlap, i.e., that the start of a delivery (which coincides with the arrival of a vehicle) does not happen until the preceding delivery has ended (which coincides with the departure of a vehicle). Constraints (1e) ensure the consecutive deliveries by the same vehicle properly account for the travel time to and from the depot. Constraints (1f) ensure that a vehicle does not depart from the customer until it was possible to transfer the entire delivery quantity into the customer's local storage. (These can easily be linearized.) Constraints (1g) are the inventory balance constraints associated with deliveries and account for any usage that may occur during a delivery (if the delivery is not instantaneous). Constraints (1h) and (1i) are the inventory balance constraints associated with the periods between consecutive deliveries. Constraints (1j) and (1k) ensure that the customer never runs out of product during the planning horizon. If the optimal objective value is positive, then there is a feasible solution, in which the requirement that no more than one vehicle is visiting a customer at a time is guaranteed by the objective; otherwise there can be no feasible solution. \square

Proposition 2. *Determining the minimum number of vehicles required to produce a feasible solution to a single-customer CIRP instance can be done in pseudo-polynomial time.*

Proof. Observe that the number of deliveries required at the customer is at least $\lceil (Hu - I)/Q \rceil$, which implies that at most $\bar{m} = \lceil (Hu - I)/Q \rceil$ vehicles are needed (each vehicle making a single delivery). Observe too that a vehicle can make at most $\lfloor H/2\tau \rfloor$ deliveries, which implies that at least $\underline{m} = \lceil \lceil (Hu - I)/Q \rceil / \lfloor H/2\tau \rfloor \rceil$ vehicles are needed. These observations show that we can enumerate the possible combinations of number of vehicles m (i.e., $\underline{m} \leq m \leq \bar{m}$) and number of deliveries n (i.e., $m \leq n \leq \bar{m} \lfloor H/2\tau \rfloor$ for a given number of vehicles m). For each combination, Lemma 2 shows that we can determine whether a feasible delivery schedule exists, by solving an LP with $\mathcal{O}(n)$ constraints and variables. This is pseudo-polynomial because the number of visits, n , depends polynomially on the planning horizon H (and thus is exponential in $\log(H)$). \square

Conjecture 1. *Determining the minimum number of vehicles required to produce a feasible solution to a single-customer CIRP instance can be done in polynomial time.*

4 Optimal delivery plans

As mentioned in the introduction, we will use partially time-expanded network formulations to solve the CIRP. An implicit underlying assumption is that there exists a discretization of time such that an optimal solution to a time-expanded network formulation using this discretization results in a continuous time optimal solution. In Section 6, we prove that when the parameters of an instance are rational numbers, an optimal solution involving only rational numbers exists, which in turn implies that such a discretization exists. Therefore,

consider a time discretization sufficiently fine that all time-based parameters (time horizon and travel times) are integer and so are all the times at which deliveries to and departures from a customer are made in an optimal solution. Specifically, suppose such a discretization is obtained by taking time intervals of length $\Delta > 0$ so that $H = T\Delta$ for some positive integer T and the discretization has T periods of length Δ . Under this discretization, the travel time from i to j is given by integer $\tau_{ij} = \hat{\tau}_{ij}/\Delta$ periods and the usage rate per period at customer i is given by $u_i = \hat{u}_i\Delta$. Under this discretization of time, we may safely restrict attention to solutions in which all deliveries to a customer occur and all vehicle departures occur at times in $\mathcal{T} = \{0, 1, \dots, T - 1\}$, where times are stated in units of periods of length Δ . We assume all travel times are non-negative, allowing travel time to be zero. In order to model waiting at a customer, and vehicles stationed at the depot between trips, we introduce waiting time $\tau_{ii} = 1$ for $i \in N_0$. We take $c_{ii} = 0$ for all $i \in N_0$.

We now construct a mixed integer linear programming formulation with vehicles and product routed in a time-expanded network, with timed node set $\mathcal{N}^{\mathcal{T}} = \{(0, T)\} \cup (N_0 \times \mathcal{T})$ and timed arc set

$$\mathcal{A}^{\mathcal{T}} = \{((i, s), (j, t)) \in \mathcal{N}^{\mathcal{T}} \times \mathcal{N}^{\mathcal{T}} : (i, j) \in N_0 \times N_0, s + \tau_{ij} = t\}.$$

For a given instance the network $(\mathcal{N}^{\mathcal{T}}, \mathcal{A}^{\mathcal{T}})$ may be reduced by preprocessing to eliminate nodes and arcs that cannot appear in any feasible vehicle route. We use the notation $\delta^+(i, s)$ to represent the set of customers $j \in N_0$ with $((i, s), (j, s + \tau_{ij})) \in \mathcal{A}^{\mathcal{T}}$. Similarly, $\delta^-(j, t)$ is used to represent the set of customers $i \in N_0$ with $((i, t - \tau_{ij}), (j, t)) \in \mathcal{A}^{\mathcal{T}}$.

For each $((i, t), (j, t + \tau_{ij})) \in \mathcal{A}^{\mathcal{T}}$, let binary variable x_{ij}^t indicate whether a vehicle travels from location i to j departing from i at time t or not, and let variable w_{ij}^t indicate the amount of product that is transported from i to j at time t (zero when no vehicle travels from i to j departing from i at time t). Let variable y_i^t indicate the quantity of product delivered to customer $i \in N$ at time $t \in \mathcal{T}$ and z_i^t indicate the inventory level at customer i at time t . If a delivery takes place at time t , then z_i^t indicates the inventory value *after* the delivery. The model includes the possibility of customer deliveries at time 0, since we allow travel times that are zero. It is assumed that the initial inventory at each customer is enough to sustain it until a vehicle can arrive, i.e., $I_i^0 \geq \tau_{0i}u_i$ for all $i \in N$; otherwise the problem is infeasible.

We can now define the formulation:

$$\begin{aligned} \min \quad & \sum_{(i,t) \in \mathcal{N}^{\mathcal{T}}} \sum_{j \in \delta^+(i,t)} c_{ij} x_{ij}^t \\ \text{s.t.} \quad & \sum_{j \in \delta^+(i,t)} x_{ij}^t = \sum_{j \in \delta^-(i,t)} x_{ji}^{t-\tau_{ji}} \quad (i,t) \in \mathcal{N}^{\mathcal{T}} \setminus \{(0,0), (0,T)\} \end{aligned} \quad (2a)$$

$$\sum_{i \in \delta^-(0,T)} x_{i,0}^{T-\tau_{i,0}} = m \quad (2b)$$

$$\sum_{j \in \delta^+(i,t)} x_{ij}^t \leq 1 \quad (i,t) \in \mathcal{N}^{\mathcal{T}} \quad (2c)$$

$$\sum_{j \in \delta^-(i,t)} w_{ji}^{t-\tau_{ji}} - \sum_{j \in \delta^+(i,t)} w_{ij}^t = y_i^t \quad (i,t) \in \mathcal{N}^{\mathcal{T}}, i \neq 0 \quad (2d)$$

$$0 \leq w_{ij}^t \leq Q x_{ij}^t \quad (i,t) \in \mathcal{N}^{\mathcal{T}}, j \in \delta^+(i,t) \quad (2e)$$

$$z_i^t = z_i^{t-1} + y_i^t - u_i \quad i \in N, t \in \mathcal{T} \setminus \{0\} \quad (2f)$$

$$z_i^0 = I_i^0 + y_i^0 \quad i \in N \quad (2g)$$

$$u_i \leq z_i^t \leq C_i \quad i \in N, t \in \mathcal{T} \quad (2h)$$

$$0 \leq y_i^t \quad i \in N, t \in \mathcal{T}$$

$$x_{ij}^t \in \{0, 1\} \quad ((i,t)(j, t + \tau_{ij})) \in \mathcal{A}^{\mathcal{T}}.$$

Constraints (2a) and (2b) ensure vehicle flow balance and ensure that all m vehicles are returned to the depot at the end of the planning horizon. Constraints (2c) together with the requirement that each x_{ij}^t variable is binary ensure that at most one vehicle can be visiting a customer at any one time. Constraints (2d) ensure product flow balance and enforce that product arriving in a vehicle at a customer is either delivered at that customer or remains on the vehicle. Constraints (2e) link the product flows to the vehicle flows. Constraints (2f) and (2g) model product usage at a customer and inventory balance. Constraints (2h) ensure that inventory at a customer is sufficient, after each delivery, to meet the customer demand in the coming period and never exceeds the local storage capacity. When all travel times are strictly positive, the time-expanded network is acyclic. As a consequence, there is no need to explicitly forbid subtours in the model, and the given vehicle flow balance constraints ensure that the number of vehicles that are away from the depot at any time does not exceed m . In the case that some travel times are zero, the situation is more complicated: we discuss this further in Section 5. The relationship between the model presented above and the model presented by Bell et al. (1983) is discussed in Appendix A.

This model has a nice structure, in the sense that for fixed x , the model is a network flow model. As a consequence, when all data is integer, and the problem is feasible, there must exist a solution in which all variables take on integer values.

Proposition 3. *For fixed x , the above model (2) in the w , y and z variables takes the form of a network flow problem.*

Proof. For fixed x , the w , y and z variables in any feasible solution to the above model are those that satisfy the constraints

$$\sum_{j \in N} w_{ji}^{t-\tau_{ji}} - \sum_{j \in N} w_{ij}^t = y_i^t \quad i \in N, t \in \mathcal{T} \quad (3a)$$

$$0 \leq w_{ij}^t \leq Qx_{ij}^t \quad i \in N_0, t \in \mathcal{T}, j \in \delta^+(i, t) \quad (3b)$$

$$z_i^t = z_i^{t-1} + y_i^t - u_i \quad i \in N, t \in \mathcal{T} \setminus \{0\} \quad (3c)$$

$$z_i^0 = I_i^0 + y_i^0 - u_i, \quad i \in N \quad (3d)$$

$$u_i \leq z_i^t \leq C_i \quad i \in N, t \in \mathcal{T} \quad (3e)$$

$$0 \leq y_i^t \quad i \in N, t \in \mathcal{T} \quad (3f)$$

These constraints can be shown to define a network flow polyhedron. The network has two nodes for each pair (i, t) with $i \in N_0$ and $t \in \mathcal{T}$. Let n_i^t denote the first and m_i^t denote the second such node. First nodes are linked by arcs in $N_0 \times N_0$, so there is a subnetwork with arcs of the form $(n_i^t, n_j^{t+\tau_{ij}})$ for $(i, j) \in N_0 \times N_0$, with capacity range $[0, Qx_{ij}^t]$ and flow on the arc given by the variable w_{ij}^t . The flow capacity constraints on these arcs are thus precisely constraints (3b). The first and second nodes for each pair (i, t) are linked by arcs of the form (n_i^t, m_i^t) , with flow lower bound zero, carrying flow given by variable y_i^t . First nodes are required to be transshipment nodes (have zero net outflow), so the flow conservation equation at nodes n_i^t are precisely the equations (3a). There are also arcs between the second nodes, of the form (m_i^t, m_i^{t+1}) , with capacity range $[u_i, C_i]$, carrying flow given by variable z_i^t . Thus the arc capacity constraints are precisely the constraints (3e). The nodes of the form m_i^t are required to have net inflow of u_i , if $t \in \mathcal{T} \setminus \{0\}$ and $u_i - I_i^0$ if $t = 0$. Thus the flow conservation constraints at second nodes of the form m_i^t are precisely (3c) and (3d). With the addition of appropriate dummy nodes and arcs to balance the flow, the constraints above clearly define a network flow polyhedron. \square

Unfortunately, the time-expanded network formulation (2) may be prohibitively large. Furthermore, while a *correct* discretization parameter, Δ , which guarantees that an optimal solution to (2) gives an optimal solution to the continuous time problem, must exist in theory, its value is, in practice, unknown. However, by selecting a (possibly incorrect) value of Δ , *a priori*, and adjusting time related parameters carefully, we can construct smaller, more manageable MIP formulations that provide either a lower or an upper bound on the optimal value of the original formulation. For example, if $H = 24$ and $\Delta = 2$, then the resulting formulation uses times $\mathcal{T} = \{0, 1, 2, \dots, 11\}$, stated in periods of length 2, and if $\Delta = 4$, the resulting formulation uses times $\mathcal{T} = \{0, 1, 2, \dots, 5\}$, stated in periods of length 4. When H is not divisible by Δ , then we may use $\mathcal{T} = \{0, 1, \dots, \lfloor \frac{H}{\Delta} \rfloor\}$. When using a given time discretization, the time related parameters need to be adjusted. For example, the usage rate \hat{u}_i at customer i has to be adjusted to $\Delta \hat{u}_i$, i.e., in each time interval of length Δ , $\Delta \hat{u}_i$ units of product are consumed. Adjusting the travel time is more involved as it requires making choices. The two natural choices are $\lfloor \hat{\tau}_{ij}/\Delta \rfloor$, which implies that the travel time may be decreased, and $\lceil \hat{\tau}_{ij}/\Delta \rceil$, which implies that the travel time may be increased. Depending

on this choice, the resulting formulation produces either a lower or an upper bound. The former is described in detail in Section 5.2 and the latter in Section 6.1.

Before doing so, we introduce some notation that will be useful in the remainder of the paper. Recall that a route, r , specifies a sequence of customer deliveries, starting and ending at the depot. If r specifies k deliveries to customers $i_j \in N$, $j = 1, \dots, k$, in the sequence i_1, \dots, i_k , the cost of the route, which we denote by c^r , is given by $c^r = \sum_{j=0}^k c_{i_j, i_{j+1}}$, where $i_0 = i_{k+1} = 0$. We will sometimes write $i \in r$ to indicate customer $i \in N$ is in route r . A route also specifies quantities delivered, say q_j^r is the quantity delivered to customer i_j at the j th delivery in route r , for $j = 1, \dots, k$. For the route to be feasible, it must be that $\sum_{j=1}^k q_j^r \leq Q$. Recalling that there may be more than one delivery to a customer in the same route, when the context ensures there is no chance of confusion, we also use $q_i^r = \sum_{j \in \{1, \dots, k\}: i_j = i} q_j^r$ to denote the total quantity delivered to customer $i \in r$ on route r . Naturally, for r a feasible route, $\sum_{i \in r} q_i^r \leq Q$ also.

5 Lower bounds

We first describe a simple lower bound that can be calculated without the need to solve an integer program, for instances in which the costs are symmetric and satisfy the triangle inequality.

5.1 A simple lower bound

Proposition 4. *Provided the costs $(c_{ij})_{i,j \in N_0}$ are symmetric and satisfy the triangle inequality, a lower bound on the optimal value of the CIRP is given by*

$$2 \sum_{i \in N} \left(\frac{H\hat{u}_i - I_i^0}{Q} \right) c_{0i}.$$

The proof relies on the following lemma.

Lemma 3. *Assume that the costs are symmetric and satisfy the triangle inequality, and consider a route r . Then for any $\lambda \geq 0$ such that $\sum_{i \in r} \lambda_i \leq 1$, it must be that*

$$c^r \geq 2 \sum_{i \in r} \lambda_i c_{0i}.$$

Proof. For any $\lambda \geq 0$ with $\sum_{i \in r} \lambda_i \leq 1$, it must be that $\sum_{i \in r} \lambda_i c_{0i} \leq \max_{i \in r} \{c_{0i}\}$. Since the costs are symmetric, twice the latter value gives the cost of visiting the customer in r that is farthest from the depot. Since the costs satisfy the triangle inequality, visiting one customer of the route is cheaper than visiting all of them. It follows that

$$c^r \geq 2 \max_{i \in r} \{c_{0i}\} \geq 2 \sum_{i \in r} \lambda_i c_{0i}$$

as required. \square

Proof of Proposition 4. During the planning horizon, customer $i \in N$ consumes $\hat{u}_i H$ units of product. Therefore, the amount delivered on the routes visiting customer i during the planning horizon needs to be at least $\hat{u}_i H - I_i^0$. Let R be the set of routes in an optimal solution and let q_i^r be the quantity delivered to customer $i \in N$ on route $r \in R$. Thus, $\sum_{r \in R: i \in r} q_i^r \geq \hat{u}_i H - I_i^0$ for all $i \in N$.

Next, we apply Lemma 3 using $\lambda_i = \frac{q_i^r}{Q}$ for customer $i \in r$ with $r \in R$. Since r is feasible, $\sum_{i \in r} q_i^r \leq Q$ so $\sum_{i \in r} \lambda_i \leq 1$. This implies that the optimal value of the CIRP, given by the sum of the costs of all routes, satisfies

$$\sum_{r \in R} c^r \geq \sum_{r \in R} 2 \sum_{i \in r} \frac{q_i^r}{Q} c_{0i} = 2 \sum_{i \in N} \frac{c_{0i}}{Q} \sum_{r \in R: i \in r} q_i^r \geq 2 \sum_{i \in N} \frac{c_{0i}}{Q} (\hat{u}_i H - I_i^0),$$

which completes the proof. \square

While this bound is very easy to calculate, we found that it is quite weak in practice. Hence, stronger lower bounds are required. The next section discusses an approach that, for more computational effort, can yield much stronger bounds.

5.2 A lower bound integer programming model

Our lower bound model is based on (2), in the sense that it uses the same variables and parameter names for a given discretization parameter, Δ . Specifically, $T = \lceil H/\Delta \rceil$, $u_i = \hat{u}_i \Delta$ for all $i \in N$, and $\tau_{ij} = \lfloor \hat{\tau}_{ij}/\Delta \rfloor$ for all $i, j \in N_0$, $i \neq j$. It is important to note that this may introduce travel times of length zero. We again take $\tau_{ii} = 1$ for all $i \in N_0$. To obtain a formulation that yields a lower bound on the optimal value of the continuous time problem, (2) must be modified. The modifications required are:

- the constraints $u_i \leq z_i^t \leq C_i$ in (2h) must be relaxed to

$$u_i \leq z_i^t \leq C_i + u_i \quad i \in N, t = 0, \dots, T-2 \quad (4a)$$

$$(H/\Delta - (T-1))u_i \leq z_i^{T-1} \leq C_i + (H/\Delta - (T-1))u_i \quad i \in N, \quad (4b)$$

where $(H/\Delta - (T-1))u_i$ is the number of units of product consumed by customer i in the part of the planning horizon from $\Delta(T-1)$ to H , and

- more than one vehicle at a customer at a time must be allowed, which can be effected by removing constraints (2c) and allowing each x_{ij}^t variable to be a non-negative integer, not necessarily binary.

Note that if Δ divides evenly into H , then $T = \lceil H/\Delta \rceil = H/\Delta$ and so $H/\Delta - (T-1) = 1$. Otherwise $0 \leq H/\Delta - (T-1) < 1$ and so the first modification is indeed a relaxation.

We call the resulting model the *Lower Bound Model (LBM)*.

5.3 Properties of the Lower Bound Model

We prove that the LBM indeed yields a lower bound on the value of the original, continuous time, problem (CIRP) by showing that any feasible solution for a continuous time instance can be mapped to a feasible solution for LBM having the same cost.

Proposition 5. *The optimal value of the Lower Bound Model is a lower bound on the optimal value of the CIRP.*

Proof. We will prove that any solution for the CIRP can be transformed into a feasible solution for LBM, without any additional cost. First, we recall that a CIRP solution consists of a finite set of deliveries to each customer, at a finite set of time points during the planning horizon, delivered by the m vehicles, each undertaking a sequence of routes, each of which starts and ends at the depot. The transformation “shifts” all deliveries made in the time interval $[\Delta t, \Delta(t+1))$ in the CIRP solution to occur at LBM time index $t \in \mathcal{T}$. Note that for any $s \in [0, H)$ it must be that $\lfloor s/\Delta \rfloor \in \mathcal{T}$. Now observe that if, in the CIRP solution, any vehicle moves from i to j departing at time $s \in [0, H)$, where $i, j \in N_0$, $i \neq j$, the time-expanded arc $((i, \lfloor s/\Delta \rfloor), (j, \lfloor s/\Delta \rfloor + \tau_{ij}))$ must exist in \mathcal{A}^T . This holds since $s + \hat{\tau}_{ij} \in [0, H]$ and

$$\lfloor s/\Delta \rfloor + \tau_{ij} = \lfloor s/\Delta \rfloor + \lfloor \hat{\tau}_{ij}/\Delta \rfloor \leq \lfloor (s + \hat{\tau}_{ij})/\Delta \rfloor \in \mathcal{T}$$

provided $s + \hat{\tau}_{ij} < H$. Note that it may be that $s + \hat{\tau}_{ij} = H$, in which case it must be that the vehicle is returning to the depot, so $j = 0$. In this case, $\lfloor s/\Delta \rfloor + \tau_{ij} \leq \lfloor H/\Delta \rfloor \in \{T-1, T\}$ and certainly $((i, \lfloor s/\Delta \rfloor), (0, \lfloor s/\Delta \rfloor + \tau_{ij})) \in \mathcal{A}^T$.

The transformation takes the variable x_{ij}^t in the LBM model to be the number of vehicles traveling from i to j departing at any time in $[\Delta t, \Delta(t+1))$ in the CIRP solution. Similarly, the w_{ij}^t variable in the LBM model will be taken to be the total units of product carried on any vehicle traveling from i to j , departing at any time in $[\Delta t, \Delta(t+1))$, in the CIRP solution. Feasibility of the CIRP solution ensures that LBM constraints (2a), (2b) and (2e) are satisfied, and it is now clear why, due to aggregation of multiple vehicle movements in the construction of the x_{ij}^t variables, these are relaxed in the LBM to allow non-binary integers and (2c) is omitted.

This construction of the x_{ij}^t variables ensures that the cost of the LBM solution is identical to the cost of the CIRP solution: each movement of a vehicle from i to j in the latter adds one to some x_{ij}^t variable, adding c_{ij} to the LBM objective function.

It remains to show that the customer inventory variables in the LBM can be set correctly. We first use the deliveries and inventory at each customer over the horizon $[0, H]$ in the CIRP solution to set the y_i^t and z_i^t variables in the LBM and show these are feasible in the LBM constraints (2f), (2g), (4a) and (4b). (Recall that (4a) and (4b) replace (2h) in the LBM.) We then explain why they are consistent with the vehicle routing variables defined above, ensuring (2d).

Given a CIRP solution, let J_i be a finite index set for the set of deliveries to customer $i \in N$, let $v_j^i \in [0, H]$ denote the time at which delivery $j \in J_i$ is made to customer i and let η_j^i denote the number of units of product delivered to customer i at this time.

From this data, the function $\hat{z}_i(s)$, representing the inventory of customer i at time Δs in the CIRP solution for each $s \in [0, H/\Delta]$, can readily be constructed. (We define $\hat{z}_i(s)$ to be the inventory at time Δs *excluding* any deliveries made at precisely this time.) Note that for feasibility of the CIRP solution, it must be that $0 \leq \hat{z}_i(s) \leq C_i$ for all $s \in [0, H/\Delta]$. Since the LBM “rounds down” travel times, to create a feasible solution for the LBM from the CIRP solution, we “shift” all deliveries made in the time interval $[\Delta t, \Delta(t+1))$ to occur at LBM time index $t \in \mathcal{T}$. It is thus helpful to define the index set of deliveries made in this interval: $J_i(t) = \{j \in J_i : \lfloor v_j^i / \Delta \rfloor = t\}$ for each $t \in \mathcal{T}$, $i \in N$. Thus in the CIRP solution, it must be that

$$\hat{z}_i(t) = \hat{z}_i(t-1) + \sum_{j \in J_i(t-1)} \eta_j^i - u_i, \quad \forall t \in \mathcal{T} \setminus \{0\}, i \in N, \quad (5)$$

and

$$\hat{z}_i(0) = I_i^0, \quad \forall i \in N. \quad (6)$$

Also, since the CIRP solution is feasible,

$$0 \leq \hat{z}_i(t) \leq C_i, \quad \forall t \in \mathcal{T}, i \in N. \quad (7)$$

and, since $0 \leq \hat{z}_i(H/\Delta) \leq C_i$, it must be that

$$0 \leq \hat{z}_i(T-1) + \sum_{j \in J_i(T-1)} \eta_j^i - (H/\Delta - (T-1))u_i \leq C_i, \quad \forall i \in N. \quad (8)$$

Construction of the LBM solution is completed by setting $y_i^t = \sum_{j \in J_i(t)} \eta_j^i$ and $z_i^t = \hat{z}_i(t) + y_i^t$ for each $t \in \mathcal{T}$, $i \in N$. Since $J_i(t)$ includes the index for any delivery made precisely at time Δt in the CIRP solution, we see that z_i^t is the inventory after any deliveries made at $t \in \mathcal{T}$ in the LBM solution. Clearly, for each $t \in \mathcal{T} \setminus \{0\}$, $i \in N$, we have, by (5), that

$$z_i^t = \hat{z}_i(t) + y_i^t = \hat{z}_i(t-1) + \sum_{j \in J_i(t-1)} \eta_j^i - u_i + y_i^t = \hat{z}_i(t-1) + y_i^{t-1} - u_i + y_i^t = z_i^{t-1} - u_i + y_i^t,$$

ensuring (2f) holds. Also, for each $i \in N$, we have, by (6), that

$$z_i^0 = \hat{z}_i(0) + y_i^0 = I_i^0 + y_i^0,$$

ensuring (2g) holds. Now for all $i \in N$ and $t = 0, 1, \dots, T-2$, we have that

$$z_i^t = \hat{z}_i(t) + y_i^t = \hat{z}_i(t) + \sum_{j \in J_i(t)} \eta_j^i = \hat{z}_i(t+1) + u_i,$$

by (5), and so $\hat{z}_i(t+1) = z_i^t - u_i$. Hence, by (7), it must be that for all $i \in N$ and $t = 0, 1, \dots, T-2$,

$$0 \leq \hat{z}_i(t+1) \leq C_i \iff 0 \leq z_i^t - u_i \leq C_i \iff u_i \leq z_i^t \leq C_i + u_i,$$

ensuring (4a) holds. Finally, for all $i \in N$,

$$z_i^{T-1} = \hat{z}_i(T-1) + y_i^{T-1} = \hat{z}_i(T-1) + \sum_{j \in J_i(T-1)} \eta_j^i,$$

and so by (8), it must be that

$$0 \leq z_i^{T-1} - ({}^H/\Delta - (T-1))u_i \leq C_i \iff ({}^H/\Delta - (T-1))u_i \leq z_i^{T-1} \leq C_i + ({}^H/\Delta - (T-1))u_i,$$

ensuring that (4b) holds.

To complete construction of the LBM feasible solution, we need to be sure that for any vehicle route making a delivery to customer i at time $s \in [0, H]$ in the CIRP solution, there is a corresponding route feasible to the LBM that delivers at time index $\lfloor s/\Delta \rfloor$. To see that this must be so, consider two consecutive customer deliveries in a CIRP vehicle route, or a departure from the depot followed by a customer delivery, or a customer delivery followed by departure from the depot. Suppose these two events occur at times $s_1, s_2 \in [0, H]$ with $s_1 \leq s_2$, and at locations $i_1, i_2 \in N_0$, respectively. Since the CIRP solution is feasible, we have that $s_1 + \hat{\tau}_{i_1 i_2} \leq s_2$ in the case $i_1 \neq i_2$ and $s_1 \leq s_2$ otherwise. In the former case, we get

$$s_1 + \hat{\tau}_{i_1 i_2} \leq s_2 \iff s_1/\Delta + \hat{\tau}_{i_1 i_2}/\Delta \leq s_2/\Delta \implies \lfloor s_1/\Delta \rfloor + \lfloor \hat{\tau}_{i_1 i_2}/\Delta \rfloor \leq \lfloor s_1/\Delta + \hat{\tau}_{i_1 i_2}/\Delta \rfloor \leq \lfloor s_2/\Delta \rfloor$$

and hence $\lfloor s_1/\Delta \rfloor + \tau_{i_1 i_2} \leq \lfloor s_2/\Delta \rfloor$. Thus the two events can occur consecutively in a LBM feasible route, at $\lfloor s_1/\Delta \rfloor, \lfloor s_2/\Delta \rfloor \in \mathcal{T}$, respectively. If $i_1 = i_2$, then obviously $s_1 \leq s_2$ implies $\lfloor s_1/\Delta \rfloor \leq \lfloor s_2/\Delta \rfloor$, so the two events can also occur consecutively in an LBM feasible route. \square

Having established that the LBM deserves its name, it is natural to ask whether or not its value is guaranteed to approach the CIRP value as Δ decreases towards zero. Unfortunately, the omission of (2c) from the LBM means that it may not. The LBM solution can have two vehicles visiting a customer at the same time, in order to move product from one vehicle to the other; doing so can reduce costs. This is a well known issue in split delivery vehicle routing problems, which makes them challenging to model. In split delivery routing, this issue is often resolved by a vehicle indexed formulation. The same device may be used here: there is a natural alternative form of the LBM based on routing variables indexed by vehicle. In practice, we found that the potential bound improvement from use of a vehicle indexed version of the LBM did not outweigh the extra computing time this much larger formulation required: even for small instances, solving the vehicle indexed formulation exactly was not possible in moderate time, and its best bound when stopped early did not match that of the LBM above, computed in the same time.

5.4 Strengthening the Lower Bound Model

5.4.1 Eliminating Depot Subtours

When Δ is relatively large, the LBM has another weakness. It occurs when for one or more pairs $(i, j) \in N_0 \times N_0$, we have $\tau_{ij} = \lfloor \hat{\tau}_{ij}/\Delta \rfloor = 0$, which induce cycles in the time-expanded

network (e.g., a cycle from i to j and back to i that takes up no time). The LBM allows a vehicle to traverse such a zero travel time cycle even if it is disconnected from the vehicle origin node, $(0, 0)$. For some $t \in \mathcal{T}$ and some set $S \subseteq N_0$ with $\tau_{ij} = 0$ for all distinct $i, j \in S$, the x_{ij}^t variables may induce a cycle even if $x_{hi}^{t-\tau_{hi}} = 0$ for all $i \in S$ and all $h \in \delta^-(i, t) \setminus S$. This violates the property of any CIRP feasible solution that, when transformed to an LBM solution with vehicle route variables x , the subgraph in $(\mathcal{N}^{\mathcal{T}}, \mathcal{A}^{\mathcal{T}})$ induced by x decomposes into paths (not necessarily simple) from $(0, 0)$ to $(0, T)$. In the presence of a zero travel time cycle, the LBM solution may violate this property. However, if the cycle does *not* include the depot, then the product flow variables, w_{ij}^t and the constraints (2d) and (2e) prevent the cycle from being used to deliver product to customers, so there is no incentive for such a cycle to appear in the LBM solution. On the other hand, if the cycle *does* include the depot, it can be used to deliver product to customers, without being part of a path connecting the origin node, $(0, 0)$, to the destination, $(0, T)$. Hence the vehicle traversing the cycle is omitted from the vehicle count constraint (2b). We call such a cycle in the LBM solution a *depot subtour* and illustrate its occurrence in Figure 4. The example shown in Figure 4 has two customers, $I_1^0 = 2$, $C_1 = 4$, $I_2^0 = 1$, $C_2 = 3$, $\hat{u}_i = 1$ for $i = 1, 2$, $c_{ij} = \hat{\tau}_{ij} = 1$ for all $i, j \in \{0, 1, 2\}, i \neq j$, a single vehicle, $Q = 6$, and planning horizon $H = 6$. An optimal CIRP solution is for the vehicle to use the time-expanded node sequence $(0, 0), (2, 1), (1, 2), (0, 3), (2, 4), (1, 5), (0, 6)$, visiting customer 2 then customer 1, returning to the depot, and repeating (shown on the left in Figure 4). On the first route, the vehicle delivers 3 units to both customers, and on the second route, delivers at least 2 units to customer 2 and at least 1 unit to customer 1. The cost is 6. If we take $\Delta = 2$ and solve the LBM, the optimal solution uses the time-expanded node sequence $(0, 0), (2, 0), (2, 1), (2, 2), (0, 2), (0, 3)$ and $(0, 1), (1, 1), (0, 1)$, the latter being a depot subtour (shown on the right in Figure 4). On the first sequence the vehicle delivers a total of 5 units to customer 2, delivering at least 1 unit on arrival. For example, it may deliver 2 units at time index 0, 2 units at time index 1 and the remaining 1 unit at time index 2. On the depot subtour, it delivers 4 units to customer 1. This meets customer demand with a cost of only 4. However, in the second period, at time index 1, the vehicle is apparently in two places at once: a phantom vehicle has been used on a depot subtour.

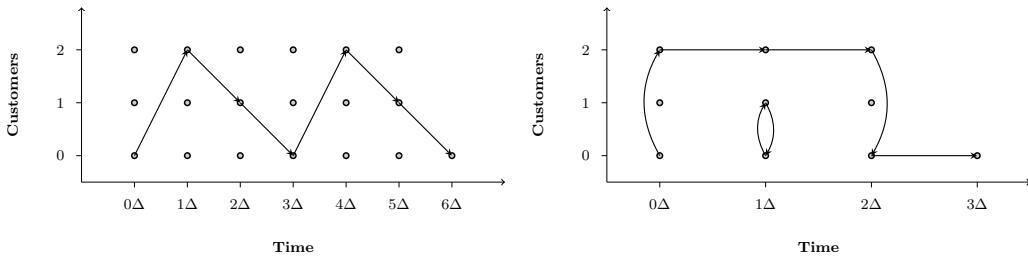


Figure 4: An example of a depot subtour.

Depot subtours can be avoided by the use of a vehicle indexed formulation. However, as discussed earlier, such a formulation is not as effective in practice. Alternatively, depot subtours can be avoided by ensuring that Δ is small enough that no zero travel time cycles

including the depot occur in the time-expanded network. However, this requirement may make it challenging to manage the size of the MIP formulation. Instead, we introduce auxiliary variables and constraints to eliminate depot subtours. The variable \tilde{w}_{ij}^t represents integer flow of a commodity that must be transported from node $(0, 0)$ to node $(0, T)$, with at least one unit of flow carried by every vehicle on every arc, and one unit of flow delivered to each location in N_0 per vehicle entering the location at any time $t \in \{1, \dots, T-1\}$. The following *depot subtour elimination constraints* are added to the LBM model:

$$\sum_{j \in \delta_t^-(i)} \tilde{w}_{ji}^{t-\tau_{ij}} - \sum_{j \in \delta_t^+(i)} \tilde{w}_{ij}^t = \sum_{j \in \delta_t^-(i)} x_{ji}^{t-\tau_{ji}} \quad \forall (i, t) \in \mathcal{N}^T \setminus \{(0, 0), (0, T)\} \quad (9a)$$

$$x_{ij}^t \leq \tilde{w}_{ij}^t \leq M_{ij}^t x_{ij}^t \quad \forall ((i, t), (j, t + \tau_{ij})) \in \mathcal{A}^T, t + \tau_{ij} \neq T, \quad (9b)$$

where M_{ij}^t is a large number. These constraints ensure that the x variables induce a subgraph in $(\mathcal{N}^T, \mathcal{A}^T)$ that can be decomposed into (possibly non-elementary) paths from $(0, 0)$ to $(0, T)$. In the example, we see that constraints (9a) require that $\tilde{w}_{01}^1 - \tilde{w}_{10}^1 = 1$ and $\tilde{w}_{10}^1 - \tilde{w}_{01}^1 = 1$, which is impossible. Thus the depot subtour $(0, 1), (1, 1), (0, 1)$ is eliminated.

Determining a valid choice for M_{ij}^t is not easy. It needs to be an upper bound on the number of times a vehicle can arrive at customer i at some time point t' in the set $\{t + \tau_{ij}, \dots, T-1\}$. In Appendix B, we suggest one approach to calculating a valid choice.

5.4.2 Valid Inequalities

We adapt several valid inequalities from the period inventory routing problem and suggest one additional class.

Period inventory routing problems take a time discretization as part of the problem description, with consumption of each customer in each time period a given parameter. Vehicles start and end routes within a single time period, and it is assumed that only the inventory at the end of each period, after any deliveries have been added and demand subtracted, must be nonnegative and no greater than the customer's storage capacity. In this context, Archetti et al. (2007) present several valid inequalities. We adapt three classes of valid inequality they present to our setting. In particular, we adapt inequalities (18) [Theorem 2], (20) [Theorem 4] and (22) [Theorem 6].

First, a lower bound on the total number of visits to a customer over the planning horizon is exploited. In the CIRP, we observe that customer $i \in N$ requires $\hat{u}_i H - I_i^0$ units of product in total to be delivered during the time, and thus requires at least $\left\lceil \frac{\hat{u}_i H - I_i^0}{Q} \right\rceil$ vehicle visits with an intervening return to the depot. Thus the inequalities

$$\left\lceil \frac{\hat{u}_i H - I_i^0}{Q} \right\rceil \leq \sum_{t \in \mathcal{T}} \sum_{j \in \delta^-(i, t) \setminus \{i\}} x_{ji}^{t-\tau_{ji}}, \quad i \in N. \quad (10)$$

must be valid for the LBM. Note that the right-hand side of this constraint excludes vehicles waiting at customer i , since a waiting vehicle will not have had an intervening return to the depot. In the case that $C_i < Q$, a stronger lower bound on the number of visits to customer

$i \in N$ is $\left\lceil \frac{\hat{u}_i H - I_i^0}{C_i} \right\rceil$, however in this case waiting vehicles must be counted. Thus, in this case, we obtain another class of inequalities,

$$\left\lceil \frac{\hat{u}_i H - I_i^0}{C_i} \right\rceil \leq \sum_{t \in \mathcal{T}} \sum_{j \in \delta^-(i,t)} x_{ji}^{t-\tau_{ji}}, \quad i \in N \text{ with } C_i < Q. \quad (11)$$

When $C_i < Q$, neither inequality from the class (10) or (11) may dominate the other; both may be useful.

Coelho and Laporte (2014) present inequalities similar to (10) and (11), but derive a lower bound on the number of visits to a customer that must occur in a given time interval. In our setting, their inequalities correspond to

$$\left\lceil \frac{u_i(t_2 - t_1 + 1) - C_i}{Q} \right\rceil \leq \sum_{t=t_1+1}^{t_2} \sum_{j \in \delta^-(i,t) \setminus \{i\}} x_{ji}^{t-\tau_{ji}}, \quad i \in N, 0 \leq t_1 < t_2 \leq T-1, \quad (12)$$

and,

$$\left\lceil \frac{u_i(t_2 - t_1 + 1) - C_i}{C_i} \right\rceil \leq \sum_{t=t_1+1}^{t_2} \sum_{j \in \delta^-(i,t)} x_{ji}^{t-\tau_{ji}}, \quad i \in N, 0 \leq t_1 < t_2 \leq T-1 \text{ with } C_i < Q, \quad (13)$$

where to be valid for LBM, we have to replace C_i with $C_i + u_i$ for $i \in N$.

The observation that the inventory on hand at the start of a period must be sufficient to sustain the customer until its next delivery can also be exploited. This needs to be done carefully in the LBM model, since the deliveries in a CIRP solution during the interval starting at time $t\Delta$ are “mapped to” deliveries at time point $t \in \mathcal{T}$. Thus, if there are no deliveries to customer i at any of time points $t_1 + 1, t_1 + 2, \dots, t_2$, in the LBM, the inventory after any delivery at t_1 must be sufficient to meet demand in the intervals starting at times $t_1\Delta, (t_1 + 1)\Delta, \dots, t_2\Delta$, i.e., for a time duration of $(t_2 - t_1 + 1)\Delta$. Thus inventory at i must be at least $\hat{u}_i(t_2 - t_1 + 1)\Delta = u_i(t_2 - t_1 + 1)$, and we have the valid inequality

$$z_i^{t_1} \geq u_i(t_2 - t_1 + 1) \left(1 - \sum_{t=t_1+1}^{t_2} \sum_{j \in \delta^-(i,t)} x_{ji}^{t-\tau_{ji}} \right), \quad i \in N, 0 \leq t_1 < t_2 \leq T-1. \quad (14)$$

Using ideas from Coelho and Laporte (2014), we can strengthen (14) to

$$z_i^{t_1} \geq u_i(t_2 - t_1 + 1) \left(1 - \min \left\{ \frac{\min\{C_i, Q\}}{u_i(t_2 - t_1 + 1)}, 1 \right\} \sum_{t=t_1+1}^{t_2} \sum_{j \in \delta^-(i,t)} x_{ji}^{t-\tau_{ji}} \right), \quad i \in N, 0 \leq t_1 < t_2 \leq T-1, \quad (15)$$

where, again, we have to replace C_i with $C_i + u_i$ for $i \in N$ to be valid for LBM.

Finally, we make use of the observation that whenever a vehicle arrives at a customer it must have departed the depot at some time sufficiently beforehand. To use this observation,

we must take care concerning the triangle inequality for travel times. In the context of our LBM, we note that, even though the original travel times, $\hat{\tau}$, are assumed to satisfy the triangle inequality, the travel times scaled to conform to a discretization with parameter Δ may not. For example, consider the case that $\hat{\tau}_{ij} = \hat{\tau}_{jk} = 3.2$ and $\hat{\tau}_{ik} = 6$, for three distinct customers i, j, k . The triangle inequality is satisfied here, as $\hat{\tau}_{ij} + \hat{\tau}_{jk} = 3.2 + 3.2 = 6.4 \geq 6 = \hat{\tau}_{ik}$. However, if we take $\Delta = 2$, then

$$\begin{aligned} t_{ij} + t_{jk} &= \lfloor \hat{\tau}_{ij}/\Delta \rfloor + \lfloor \hat{\tau}_{jk}/\Delta \rfloor = \lfloor 3.2/2 \rfloor + \lfloor 3.2/2 \rfloor = \lfloor 1.6 \rfloor + \lfloor 1.6 \rfloor = 1 + 1 \\ &= 2 \not\geq 3 = \lfloor 6/2 \rfloor = \lfloor \hat{\tau}_{ik}/\Delta \rfloor = t_{ik}. \end{aligned}$$

Thus we must adapt the statement of inequality to account for this. Letting τ_{0i} be the length of the shortest path from 0 to i in the complete network on nodes N_0 , with length of arc (j, k) taken to be τ_{jk} , we have that the class of constraints

$$\sum_{j \in \delta^-(i,t)} x_{ji}^{t-\tau_{ji}} \leq \sum_{s=0}^{t-\tau_{0i}} \sum_{j \in \delta^+(0,s), j \neq 0} x_{0j}^s, \quad (i,t) \in \mathcal{N}^T \quad (16)$$

is valid for the LBM.

6 Constructing feasible delivery plans

Next, we investigate optimization models that may produce feasible solutions to the continuous time problem, CIRP. We consider two alternative approaches. The first is to develop a different, upper bound, model, based on (2) with appropriate parameter choices and modifications. The second is based on the solution to the LBM, and attempts to adjust it to obtain a feasible CIRP solution.

6.1 An upper bound integer programming model

Like the LBM, our upper bound model is based on (2), using the same variables and parameter names for a give discretization parameter, Δ . As for the LBM, $T = \lceil H/\Delta \rceil$ and $u_i = \hat{u}_i \Delta$ for all $i \in N$, but the travel times are now rounded up, rather than down: $\tau_{ij} = \lceil \hat{\tau}_{ij}/\Delta \rceil$ for all $i, j \in N_0, i \neq j$. To obtain a formulation that yields an upper bound on the optimal value of the continuous time problem, (2) must be modified. The modifications required are:

- the constraints $u_i \leq z_i^t \leq C_i$ in (2h), for the case $t = T - 1$, must be replaced by

$$(H/\Delta - (T - 1))u_i \leq z_i^{T-1} \leq C_i, \quad \forall i \in N, \quad (17)$$

where $(H/\Delta - (T - 1))u_i$ is the number of units of product consumed by customer i in the part of the planning horizon from $(T - 1)\Delta$ to H , and

- some arcs must be removed, according to

$$\mathcal{A}^T := \mathcal{A}^T \setminus \{((i,t), (0,T)) : t\Delta + \hat{\tau}_{i0} > H\}. \quad (18)$$

Both are modifications to (2) only if Δ does not divide evenly into H . In this case, for the final time interval induced by the discretization, $[(T-1)\Delta, T\Delta]$, with $T\Delta > H$, the inventory on hand at each customer at the start of the interval only needs to be enough to supply the customer up to time H . Also, all vehicles must return to the depot by time H , which is modeled by the sink node $(0, T)$ in the discretized problem, and so arc $((i, t), (0, T)) \in \mathcal{A}^T$ should only be used if $\Delta t + \hat{\tau}_{i0} \leq H$.

We call the resulting model the *Upper Bound Model (UBM)*. That any feasible solution to the UBM is also a feasible solution to the CIRP is quite obvious. A vehicle visit to customer i (or the depot) at time point $t \in \mathcal{T}$ followed by a visit to customer j (or the depot) at time point $t + \tau_{ij} \in \mathcal{T}$ in the UBM corresponds to a vehicle visit to customer i at time $t\Delta \in [0, H]$ followed by travel to j , arriving at time $t\Delta + \hat{\tau}_{ij}$, and waiting at j for time $\tau_{ij}\Delta - \hat{\tau}_{ij} > 0$, by the definition of τ_{ij} . The inventory constraints ensure that the customer deliveries made at times $t\Delta$ for $t \in \mathcal{T}$ are sufficient to meet the customer demand over the whole planning period $[0, H]$. The proposition below follows.

Proposition 6. *If the Upper Bound Model is feasible, then it provides a feasible solution for the CIRP.*

We note that constraints similar to those for strengthening the LBM may be used to strengthen the UBM. Specifically, (10)–(13), and (15) may all be used as stated, while, since the triangle inequality for $\hat{\tau}$ implies the triangle inequality for τ in the UBM, (16) is applied simply as

$$\sum_{j \in \delta^-(i, t)} x_{ji}^{t-\tau_{ji}} \leq \sum_{s=0}^{t-\tau_{0i}} \sum_{j \in \delta^+(0, s)} x_{0j}^s, \quad (i, t) \in \mathcal{N}^T. \quad (19)$$

6.2 Converting solutions to LBM

In order to decide whether or not the solution to LBM is, in fact, an optimal to CIRP, we seek to convert the discrete time solution to LBM into a continuous time feasible solution of the same cost. If successful, then that solution must be an optimal continuous time solution.

Clearly any CIRP feasible solution that performs the same set of vehicle movements as that implied by the LBM solution will do. So a natural approach is to seek a CIRP feasible solution in which that occurs. We divide the process into two steps.

1. **Step 1: extracting delivery routes.** Recall that the LBM solution may not uniquely specify routes for each vehicle, since more than one vehicle may visit a customer at a time. Furthermore, the product quantities assigned to vehicle movements may involve product exchange at a common customer location and time, and so cannot be decomposed to match independent vehicle routes. Thus our first step is to seek a decomposition of the LBM solution into independent vehicle itineraries with associated product flows that provide inventory levels as close as possible to LBM feasibility.
2. **Step 2: revising customer visit times and quantities.** The first step yields a set of vehicle itineraries, each vehicle itinerary specifying a sequence of customer (and

depot) visits including the time of the visit and the quantity delivered. The time information implies a sequence of vehicle visits at a customer. In case of multiple visits at the same time, we impose an arbitrary order. Our second step is to revise the time of each visit and the quantity delivered during the visit, while preserving both the vehicle and customer visit sequences and ensuring each vehicle route's timing is feasible. The goal is to ensure that the first customer to run out of product does so as late as possible. If no customer runs out during the planning horizon, then a CIRP solution of the same cost has been found.

We accomplish the first step by solving a MIP and the second step by solving an LP or an IP. Details are found below.

Extracting delivery routes. As discussed earlier, the LBM solution may not uniquely specify routes (or itineraries) for each vehicle. Although this issue may be overcome with an alternative, vehicle indexed, formulation, we found that this approach did not perform well in practice. Instead, we propose a heuristic for solving the vehicle indexed formulation: first solve the LBM, and then, keeping some aspects of the LBM solution fixed, seek a “nearby” solution to the vehicle indexed formulation. Thus, given a solution to the LBM having vehicle movement variables x^* say, we seek a decomposition of it into independent vehicle itineraries with associated product flows.

To do so, we use a MIP model in which vehicle movement, product flow and delivery quantity variables, x_{ijv}^t , w_{ijv}^t and y_{iv}^t , respectively, are indexed by vehicle, $v \in V := \{1, \dots, m\}$. The vehicle movement variables are required to decompose those from the LBM solution in the sense that we require $\sum_{v \in V} x_{ijv}^t = x_{ij}^{*t}$ for all $((i, t), (j, t + \tau_{ij})) \in \mathcal{A}^T$. As a consequence, many vehicle movement variables, specifically x_{ijv}^t for which $x_{ij}^{*t} = 0$, can be eliminated in preprocessing, ensuring that the MIP is not difficult to solve in practice.

In other respects, the MIP is very similar to the LBM (adapted to use vehicle indexed product flow and delivery variables). However, since the LBM solution may implicitly require product exchange between vehicles that cannot be decomposed into product flows on independent routes, we cannot guarantee that LBM-feasible inventory levels at customers can be attained. We thus introduce slack and surplus variables, ξ_{it}^+ and ξ_{it}^- , respectively, on the inventory level for each customer $i \in N$ and time point $t \in \mathcal{T}$, and seek to minimize a weighted sum of these variables. Although any positive weights would achieve a solution that is in some sense “close to” that of the LBM, we put a higher weight on use of these variables at earlier time points. This is a heuristic designed to maximize the time period over which the resulting routes can feasibly supply customers in the CIRP solution we wish to attain after customer visit times and quantities are revised, subsequently. We call the

resulting MIP model the *vehicle itinerary extraction (VIE)* model:

$$\min \quad \sum_{t \in \mathcal{T}} (T - t) \sum_{i \in N} (\xi_{it}^+ + \xi_{it}^-)$$

$$\text{s.t.} \quad \sum_{j \in \delta^+(i, t)} x_{ijv}^t = \sum_{j \in \delta^-(i, t)} x_{jiv}^{t-\tau_{ji}} \quad (i, t) \in \mathcal{N}^{\mathcal{T}} \setminus \{(0, 0), (0, T)\}, v \in V \quad (20a)$$

$$\sum_{i \in \delta^-(0, T)} x_{i,0,v}^{T-\tau_{i,0}} = 1 \quad v \in V \quad (20b)$$

$$\sum_{v \in V} x_{ijv}^t = x_{ij}^{*t} \quad ((i, t), (j, t + \tau_{ij})) \in \mathcal{A}^{\mathcal{T}} \quad (20c)$$

$$\sum_{j \in \delta^-(i, t)} w_{jiv}^{t-\tau_{ji}} - \sum_{j \in \delta^+(i, t)} w_{ijv}^t = y_{iv}^t \quad (i, t) \in \mathcal{N}^{\mathcal{T}}, i \neq 0, v \in V \quad (20d)$$

$$0 \leq w_{ijv}^t \leq Q x_{ijv}^t \quad (i, t) \in \mathcal{N}^{\mathcal{T}}, j \in \delta^+(i, t), v \in V \quad (20e)$$

$$z_i^t = z_i^{t-1} + \sum_{v \in V} y_{iv}^t - u_i + \xi_{it}^+ - \xi_{it}^- \quad i \in N, t \in \mathcal{T} \setminus \{0\} \quad (20f)$$

$$z_i^0 = I_i^0 + \sum_{v \in V} y_{iv}^0 + \xi_{i0}^+ - \xi_{i0}^- \quad i \in N \quad (20g)$$

$$u_i \leq z_i^t \leq C_i + u_i \quad \forall i \in N, t = 0, \dots, T-2 \quad (20h)$$

$$u'_i \leq z_i^{T-1} \leq C_i + u'_i \quad \forall i \in N, \quad (20i)$$

$$y_{iv}^t \geq 0 \quad i \in N, t \in \mathcal{T}, v \in V$$

$$x_{ijv}^t \text{ integer} \quad ((i, t), (j, t + \tau_{ij})) \in \mathcal{A}^{\mathcal{T}}, v \in V$$

$$\xi_{it}^+, \xi_{it}^- \geq 0, \quad i \in N, t \in \mathcal{T},$$

where $u'_i = ({}^H/\Delta - (T-1))u_i$ for each $i \in N$. Note that this formulation does not require depot subtour elimination constraints, since we enforce that each route arrives at the depot exactly once.

Revising Customer Visit Times and Quantities. Consider the routes and customer visits specified by the solution to the VIE model. Let R denote the set of routes and let $\rho(r, k) \in N$ denote the k th customer visited in route $r \in R$, where n_r (with $n_r \geq 1$) indicates the number of customers visited on route r and $\rho(r, 0) = \rho(r, n_r + 1) = 0$ (the route starts and ends at the depot). For each customer $i \in N$, let n_i (with $n_i \geq 0$) denote the number of deliveries at customer i . Let $n_0 = 2 \sum_{r \in R} n_r$ denote the number of route departures and arrivals, which we will call *events*, at the depot. Let $\phi(r, k)$ for $k \in \{1, \dots, n_r\}$ denote the visit index in the visit sequence of the k th customer visited in route r . That is, if $\phi(r, k) = \ell$, then the ℓ th delivery at customer $\rho(r, k)$ is the k th delivery performed by route r . (If a customer is visited by more than one vehicle at the same time, we arbitrarily order these visits in the customer visit sequence.) For $k = 0$, $\phi(r, 0) = \ell$ indicates that the departure of route r from the depot is the ℓ th event at the depot. Similarly, for $k = n_r + 1$, $\phi(r, n_r + 1) = \ell$ indicates that the arrival of route r at the depot is the ℓ th event at the depot. To account

for the fact that a vehicles can perform multiple routes in its itinerary, let $r_1^v, r_2^v, \dots, r_{n_v}^v$ denote the routes performed by vehicle v , where n_v denotes the number of routes performed by vehicle v .

We now construct a linear programming (LP) model to decide (revise) the time of each delivery to a customer, and the quantity to be delivered at that time, while preserving the visit sequence at each customer, the customer sequence on each route and the route sequence in each itinerary. Preserving these sequences, which are encoded in the route indices and the $\rho(\cdot, \cdot)$ and $\phi(\cdot, \cdot)$ functions, enables the timing and quantity decisions to be made using an LP, without the need for binary variables. Naturally, it may be that no feasible CIRP solution using these sequences exists. The LP may be infeasible; solving it is a primal heuristic, and may fail. The LP is constructed so that if it is feasible, one of two cases must occur: (1) all its feasible solutions require deliveries by more than one vehicle to a customer at the same time, or (2) a feasible solution in which all vehicle delivery times at a customer are distinct. In the former case, there, again, cannot be a feasible CIRP solution using these sequences. In the latter case, a feasible CIRP solution has been found, and, since the cost of any solution is purely the sum of the route costs, which are preserved by the model, its solution must be optimal for the CIRP. The LP is constructed as follows.

Let variable t_i^ℓ indicate the time of the ℓ th visit to customer i , while q_i^ℓ indicates the quantity delivered in that visit (and $t_i^0 = 0$). Inventory variables, I_i^ℓ , denote the inventory at customer i immediately after the ℓ th delivery. (As before, I_i^0 is denotes the initial inventory at i). Let variable t_0^ℓ denote the time of the ℓ th event at the depot. Finally, let variable ζ denote the minimum time between deliveries at any customer. We define the LP as follows:

$$\begin{aligned}
& \max && \zeta \\
\text{s.t.} & t_{\rho(r,k)}^{\phi(r,k)} + \hat{\tau}_{\rho(r,k)\rho(r,k+1)} \leq t_{\rho(r,k+1)}^{\phi(r,k+1)} && r \in R, k = 0, \dots, n_r \quad (21a) \\
& t_0^{\phi(r_j^v, n_{r_j^v} + 1)} \leq t_0^{\phi(r_{j+1}^v, 0)} && v \in V, j = 1, \dots, n_v - 1 \quad (21b) \\
& \sum_{k=1}^{n_r} q_{\rho(r,k)}^{\phi(r,k)} \leq Q && r \in R \quad (21c) \\
& t_i^\ell \geq t_i^{\ell-1} + \zeta && i \in N, \ell = 1, \dots, n_i \quad (21d) \\
& I_i^\ell = I_i^{\ell-1} - \hat{u}_i (t_i^\ell - t_i^{\ell-1}) + q_i^\ell && i \in N, \ell = 1, \dots, n_i \quad (21e) \\
& I_0^i \geq \hat{u}_i H && i \in N \text{ and } n_i = 0 \quad (21f) \\
& I_i^{n_i} - \hat{u}_i (H - t_i^{n_i}) \geq 0 && i \in N \text{ and } n_i > 0 \quad (21g) \\
& q_i^\ell \leq I_i^\ell \leq C_i && i \in N, \ell = 1, \dots, n_i \quad (21h) \\
& 0 \leq q_i^\ell && i \in N, \ell = 1, \dots, n_i \\
& 0 \leq t_i^\ell \leq H && i \in N, \ell = 1, \dots, n_i.
\end{aligned}$$

Constraints (21a) and (21b) ensure that for each of the vehicles the visit times at customers properly account for travel times between locations, and, in case a vehicle performs multiple routes, for travel times to and from the depot in between consecutive routes in its itinerary.

Constraints (21c) ensure that the delivery quantities on a route do not exceed the vehicle capacity. Constraints (21d) ensure that the visit times at customers occur in the same order as in the solution to the lower bound model, and that consecutive deliveries occur at least ζ units of time apart. Constraints (21e) ensure that the inventory at customers is accurately modeled. Constraints (21f)– (21h) together ensure that inventory at each customer is sufficient to meet demand at all points in time. Constraints (21f) and (21g) ensure that inventory after the last delivery is sufficient to meet demand until the end of the planning horizon, while (21h) guarantee that customer i does not run out of product in period $[t_i^{\ell-1}, t_i^\ell]$ for $\ell = 1, \dots, n_i$. Constraints (21h) also ensure that inventory does not exceed capacity at any customer. If the LP is feasible and has optimal value $\zeta^* > 0$, then the LP solution provides a feasible solution to the CIRP with vehicle movement cost the same as the cost of x^* used in the VIE model. Hence, if x^* is an optimal solution to the LBM, the LP solution also gives an optimal solution to the CIRP.

Theorem 1. *Let the travel times, the storage and vehicle capacities, the initial inventories and the usage rates of an instance of the CIRP be rational. Then, if the CIRP instance is feasible, it has an optimal solution that is rational.*

Proof. Since we have rational parameters, we know that the LP model (21) always has a rational optimal solution when it is feasible. The model finds the best visit times and delivery quantities for a given set of vehicle itineraries and customer visit sequences. In particular, if we take the set of vehicle itineraries and customer visit sequences to be those of an optimal solution to the CIRP, then we will get a rational solution. \square

Note that this result implies that there exists a discretization of time such that an optimal solution to a time-expanded network formulation using this discretization results in a continuous time optimal solution, which is alluded to in Section 4.

A more powerful model is obtained when we only require that the sequence in which customers are visited in a route is maintained, i.e., when we allow the sequence of routes visiting a customer to change, when we allow the sequence in which routes are performed by a vehicle to change, and when we allow routes to be reassigned to a different vehicle. Doing so, however, requires the introduction of binary variables. Specifically, we let binary variable x_{rk}^ℓ indicate whether the k^{th} visit of route r , which is a visit to customer $\rho(r, k)$, is the ℓ^{th} visit at customer $\rho(r, k)$, and let $y_{r\bar{r}}$ indicate that route r and route \bar{r} are performed by the same vehicle and that route \bar{r} is performed immediately after route r by that vehicle.

The visit times have to be viewed from two perspectives: a customer perspective and a route perspective. Let t_i^ℓ indicate the time of the ℓ^{th} visit to customer i and let \bar{t}_r^k indicate the time of the k^{th} visit of route r . Similarly, let q_i^ℓ indicate the quantity delivered in the ℓ^{th} visit to customer i and \bar{q}_r^k indicate the quantity delivered in the k^{th} visit of route r . We need to ensure that these time are consistent with the decisions we make for the routes. We

do so with the following constraints:

$$\begin{aligned}
-H(1 - x_{rk}^\ell) &\leq t_{\rho(r,k)}^\ell - \bar{t}_r^k \leq H(1 - x_{rk}^\ell) & r \in R, k = 1, \dots, n_r, \ell = 1, \dots, n_{\rho(r,k)}, \\
-Q(1 - x_{rk}^\ell) &\leq q_{\rho(r,k)}^\ell - \bar{q}_r^k \leq Q(1 - x_{rk}^\ell) & r \in R, k = 1, \dots, n_r, \ell = 1, \dots, n_{\rho(r,k)}, \\
\sum_{\ell=1}^{n_{\rho(r,k)}} x_{rk}^\ell &= 1 & r \in R, k = 1, \dots, n_r, \\
\sum_{r \in R} \sum_{k=1, \dots, n_r : \rho(r,k)=i} x_{rk}^\ell &= 1 & i \in N, \ell = 1, \dots, n_i.
\end{aligned}$$

Ensuring that each route is assigned to one of the m vehicles, that the routes assigned to a vehicle are performed in sequence, and that a route departs from the depot only after the immediately preceding route has returned to the depot is enforced by the following constraints

$$\begin{aligned}
\bar{t}_r^{n_r+1} &\leq \bar{t}_{\bar{r}}^0 + H(1 - y_{r\bar{r}}) & r \in R, \bar{r} \in R, r \neq \bar{r}, \\
\sum_{\bar{r} \in R} y_{r\bar{r}} &\leq 1 & r \in R, \\
\sum_{\bar{r} \in R} y_{\bar{r}r} &\leq 1 & r \in R, \\
\sum_{r, \bar{r} \in R} y_{r\bar{r}} &\geq |R| - m,
\end{aligned}$$

where the last constraint ensures that at most m routes have no predecessor, which implies that the routes are assigned to no more than m vehicles.

The complete *route-preserving only* (RPO) model can be found in Appendix C. Similar to the previous LP model, if the RPO model is feasible and has an optimal solution with positive value, then, since the cost of any solution is purely the sum of the route costs, which are preserved by both the VIE model and the RPO model, the solution to the latter must be optimal for the CIRP. Computational experiments revealed that even though RPO is an integer program, on instances of interest, it can often find a feasible solution in a short amount of time. Therefore, in our computational study, we have used RPO to try and extract a feasible delivery schedule from a solution to LBM.

7 A computational study

The purpose of our computational study is two-fold. First, it is to establish the potential of dynamic discretization discovery algorithms in contexts other than service network design (Boland et al. (2017)). Second, and certainly not less important, it is to demonstrate that optimal solutions, or at least provably high-quality solutions, can be obtained for non-trivial instances of one of the most challenging, but practically relevant, variants of the inventory routing problem.

7.1 Instances

We define two base instances and then alter those in a controlled way to create additional instances. Based on preliminary experiments, we have seen that the largest instances we can solve in a reasonable amount of time have 15 customers, therefore the base instances have 15 customers. The customers in the two base instances have the same usage rates and storage capacities, but differ in their locations, from which travel times (and costs) are derived.

One of the base instances has customers located in the area $[-5, 5] \times [-5, 5]$ with the company's facility located in the center, so the depot has coordinates $(x_0, y_0) = (0, 0)$. Customer locations are chosen uniformly at random with the given area. The other base instance has customers located in clusters, with locations chosen uniformly at random within three subregions of the area: $[2, 3] \times [2, 3]$, $[2, 3] \times [-2, -3]$, and $[-2, -3] \times [-2, -3]$.

The base usage rate, denoted by u_i^{base} for customer $i \in N$, is a randomly generated integer between 4 and 12 (each equally likely). The customer's storage capacity is an integer that depends on the customer's usage rate: for each $i \in N$, $C_i = f \times u_i$, where f is a randomly generated integer between 8 and 14 (each equally likely). The locations, base usage rates, and storage capacities of the 15 customers are given in Table 1. The travel time, $\hat{\tau}_{ij}$, from location i to j , with $i, j = 0, 1, \dots, 15$, is taken to be the Euclidean distance, $\|(x_i, y_i) - (x_j, y_j)\|$, rounded to two decimal places, and the cost, c_{ij} , is set equal to the travel time. We note that in all instances, the triangle inequality still holds after rounding the travel times, and that there is sufficient time for a vehicle to reach each customer before it runs out of product.

We generate instances by considering different numbers of customers $n = |N|$. The smallest instance has 5 customers, then 7, 10, 12, and the largest instance has 15 customers. Each instance uses the first n customers in Table 1. So for example, a *clustered instance* with 5 customers takes the first 5 customers using the location coordinates of fourth and fifth columns, while a *non-clustered instance* with 12 customers takes the first 12 customers using the location coordinates of the second and third columns.

Table 1: Customer data for the instances.

i	Customer locations				C_i	Three usage rate cases					
	non-clustered		clustered			u_i^{base}	r_i	$1.1u_i^{base}$	r_i	$1.2u_i^{base}$	r_i
1	2.22	-1.53	2.18	2.77	132	11	2	12.1	2	13.2	3
2	3.16	3.67	2.42	-2.46	44	4	2	4.4	2	4.8	2
3	2.68	-2.41	2.98	2.55	64	8	2	8.8	3	9.6	3
4	-1.56	2.12	2.43	-2.75	65	5	1	5.5	2	6.0	2
5	-1.60	0.45	2.59	2.70	120	12	2	13.2	3	14.4	3
6	3.86	3.98	2.40	-2.97	140	10	2	11.0	2	12.0	2
7	-3.45	4.32	2.54	2.03	81	9	2	9.9	2	10.8	3
8	1.23	0.39	2.76	-2.49	44	4	2	4.4	2	4.8	2
9	3.68	-3.73	2.18	2.34	88	11	2	12.1	3	13.2	3
10	-1.86	-1.96	2.49	-2.01	70	7	2	7.7	2	8.4	2
11	3.92	1.97	-2.52	-2.06	100	10	2	11.0	2	12.0	3
12	-1.69	2.82	-2.85	-2.68	112	8	1	8.8	2	9.6	2
13	3.53	1.36	-2.75	-2.17	132	11	2	12.1	2	13.2	3
14	-2.46	-3.05	-2.48	-2.90	44	4	2	4.4	2	4.8	2
15	-2.66	1.59	-2.74	-2.53	77	6	1	6.6	2	7.2	2

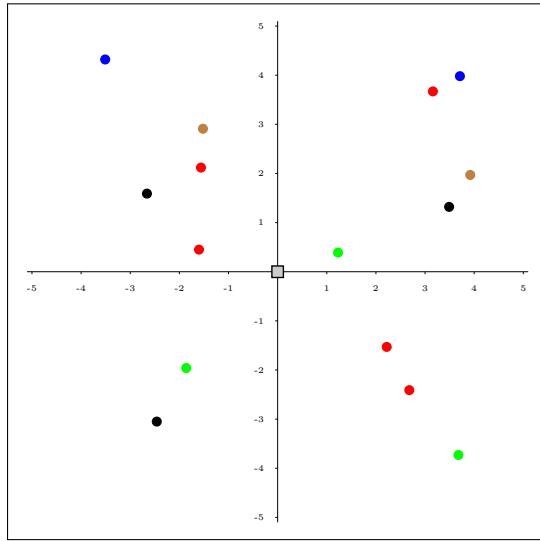


Figure 5: Random customers

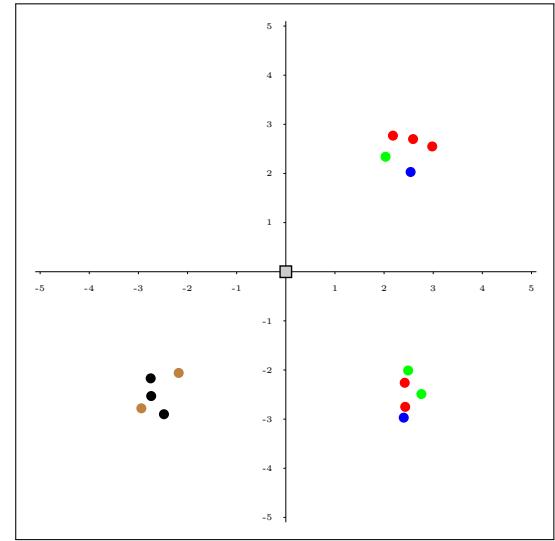


Figure 6: Clustered customers

Figures 5 and 6 plot customers and depot positions. Customers 1 to 5 are shown in red, Customers 6 and 7 are shown in blue, Customers 8 to 10 are shown in green, Customers 11 and 12 are shown in brown, and, finally, Customers 13 to 15 are shown in black.

To decide the planning horizon, H , we first observe that the travel time from the depot to any customer and back cannot exceed 15, since $\sqrt{5^2 + 5^2} \approx 7.07$. Thus if the horizon is at least 15, we can be sure there is time to serve each customer once in the planning horizon and return the vehicle to the depot. Also, for simplicity, we set the initial inventory of each customer equal to its storage capacity, so all customers start at full capacity at the start of the planning horizon. From the choice of $C_i = fu_i^{base}$ for $f \leq 14$, we see that provided the horizon is more than 14, each customer will require at least one visit during the planning horizon, using the base data. Since we want a horizon in which some customers will be visited several times, we choose $H = 18$ for all instances. (This is also a choice that enables several alternative integer time discretizations.)

We alter base instances to obtain a larger set of instances by scaling up the usage rates, using the scaling factors 1.1 and 1.2. In other words, we have instances with $\hat{u}_i = u_i^{base}$ for all $i \in N$, $\hat{u}_i = 1.1u_i^{base}$ for all $i \in N$ and $\hat{u}_i = 1.2u_i^{base}$ for all $i \in N$, for each of the clustered and non-clustered customer locations and each number of customers. The scaled usage rate cases are shown in the ninth and eleventh columns of Table 1. We also observe that the number of visits to customer i over the time horizon must be at least $r_i(\hat{u}_i) = \lceil (\hat{u}_i H - I_i^0) / \min\{C_i, Q\} \rceil$. These lower bounds on the number of visits to a customer, in each of the three usage rate cases, are shown in the eighth, tenth and twelfth columns of Table 1, headed r_i . So for the base case, most customers require at least two visits and some require at least one. For the first scaling of usage rates, most customers require at least two visits, while a few require three visits. For the second scaling of usage rates, two or three visits are required, at least, mixed about half and half.

We take the vehicle capacity to be the expected value of the customer capacity, which is the expected value of the base customer usage rate multiplied by the midpoint of the random multiplier range, i.e., $Q = 8 \times 11 = 88$. To investigate the impact of the vehicle capacity, we introduce two variations, one in which we take the vehicle capacity to be $0.75 \times Q = 66$ and one in which we take the vehicle capacity to be $1.25 \times Q = 110$. When we report results, we refer to these three variants as $Q1$ ($Q = 66$), $Q2$ ($Q = 88$), and $Q3$ ($Q = 110$).

Thus, we have $2 \times 3 \times 3 \times 5 = 90$ instances, for the two types of customer location, the three usage rate scalings, the three vehicle capacities, and the five different numbers of customers. Instances will be identified and referenced using a 4-tuple (customer location type, number of customers, usage rate scaling factor, and vehicle capacity scaling factor), e.g., (R,7,U2,Q1), abbreviated as R7U2Q1, indicates an instance with random customer locations, 7 customers, usage rate scaling U2, and vehicle capacity scaling Q1. All instances can be found at <https://github.com/felipelagos/cirplib>.

One of the challenges associated with the LBM is that travel times may be rounded down to zero. In Table 2, we show the fraction of travel times in an instance that are rounded to zero for different values Δ (i.e., $\Delta = H/2k$ for $k = 1, \dots, 9$ and $\Delta = H/6k$ for $k = 4, \dots, 10$), where, for convenience, we also report the resulting number of time points in the discretization (H/Δ). Observe that for the random instances and $\Delta \leq H/30$, all the travel times are positive, which implies that depot subtour elimination constraints are no longer needed.

Length Δ (H/Δ)	Clustered					Random				
	5	7	10	12	15	5	7	10	12	15
9.00 (2)	100.00	100.00	100.00	100.00	100.00	100.00	96.43	96.36	97.44	97.50
4.50 (4)	60.00	60.71	60.00	46.15	40.00	46.67	35.71	43.64	43.59	42.50
3.00 (6)	26.67	32.14	36.36	28.21	25.83	33.33	25.00	25.45	24.36	23.33
2.25 (8)	26.67	32.14	36.36	26.92	25.00	20.00	14.29	12.73	12.82	12.50
1.80 (10)	26.67	32.14	36.36	26.92	25.00	20.00	14.29	10.91	8.97	10.00
1.50 (12)	26.67	32.14	36.36	26.92	25.00	6.67	7.14	5.45	5.13	5.83
1.29 (14)	26.67	32.14	36.36	26.92	25.00	6.67	7.14	3.64	3.85	5.00
1.12 (16)	26.67	32.14	36.36	26.92	25.00	6.67	7.14	3.64	3.85	3.33
1.00 (18)	26.67	32.14	36.36	25.64	24.17	6.67	7.14	3.64	3.85	3.33
0.75 (24)	20.00	25.00	29.09	20.51	19.17	0.00	3.57	1.82	1.28	0.83
0.60 (30)	20.00	17.86	21.82	15.38	14.17	0.00	0.00	0.00	0.00	0.00
0.50 (36)	20.00	14.29	14.55	10.26	10.00	0.00	0.00	0.00	0.00	0.00
0.43 (42)	20.00	14.29	10.91	7.69	6.67	0.00	0.00	0.00	0.00	0.00
0.38 (48)	6.67	7.14	5.45	3.85	4.17	0.00	0.00	0.00	0.00	0.00
0.33 (54)	6.67	7.14	3.64	2.56	2.50	0.00	0.00	0.00	0.00	0.00
0.30 (60)	6.67	7.14	3.64	2.56	1.67	0.00	0.00	0.00	0.00	0.00

Table 2: Percentage (%) of zero travel times for LBM for a given discretization length.

Proposition 1 establishes that finding the minimum number of vehicles required to guarantee the existence of a feasible delivery plan is strongly NP-hard. Therefore, to determine the number of vehicles available in an instance, we use a modified version of UBM. Instead of minimizing the total cost of the routes, we minimize the number of vehicles m . We run the UBM with a time limit of 2 hours for each of two values of Δ , $H/9$ and $H/18$, recording the best feasible solution found within the time limit. Note that in all cases a feasible solution was found. We then take the minimum of the number of vehicles used in the two solutions. The resulting number of vehicles for each instance can be found in Tables 3 and 4.

Customers		Usage Level								
		U1			U2			U3		
		Capacity			Capacity			Capacity		
Customers	Q1	Q2	Q3	Q1	Q2	Q3	Q1	Q2	Q3	
	5	3	3	3	4	3	3	4	4	3
	7	4	3	3	5	4	4	6	5	4
	10	6	5	4	7	6	5	8	7	6
	12	7	6	5	8	7	6	10	8	7
	15	8	7	6	9	8	7	11	9	8

Table 3: Minimum number of vehicles for the clustered instance.

		Usage Level								
		U1			U2			U3		
		Capacity			Capacity			Capacity		
		Q1	Q2	Q3	Q1	Q2	Q3	Q1	Q2	Q3
Customers	5	3	3	3	4	3	3	4	3	3
	7	5	4	3	6	5	4	6	5	5
	10	7	6	5	8	7	6	9	8	7
	12	8	7	6	9	8	7	11	9	8
	15	9	8	7	11	9	8	13	11	9

Table 4: Minimum number of vehicles for the random instance.

7.2 Experiments

As mentioned above, the purpose of our experiments is to establish the potential of dynamic discretization discovery algorithms and to demonstrate that optimal solutions, or at least provably high-quality solutions, can be obtained for instances of CIRP.

In this proof-of-concept study, we simply experiment with different discretizations, i.e., with different values of Δ , and analyze the results. In future research, we will focus on dynamically discovering (location-dependent) discretizations. More specifically, here we solve LBM for $\Delta = {}^H/k$ for $k = 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 12, 14, 16, 18, 24, 30, 36, 42, 48, 54, 60$, and UBM for $\Delta = {}^H/k$ for $k = 9, 10, 12, 14, 16, 18, 24, 30, 36, 42, 48, 54, 60$ (i.e., only finer discretizations). After solving LBM, we use the route-preserving only (RPO) model to try and convert the solution into a CIRP feasible solution. Each model is solved with a time limit of 2 hours.

The lower bound for an instance is the maximum value of best bound over all values of Δ . The upper bound for an instance is the minimum value among all known CIRP feasible solutions (found either when determining the number of vehicles for the instance, or by the UBM, or by the RPO model applied to output of the LBM, for some value of Δ). Tables 5 and 6 show the resulting optimality gap for each of the instances. When the upper bound is associated with the feasible solution obtained when determining the number of vehicles for the instance, the value of the gap is presented in parentheses. When this happens, neither the UBM nor the RPO model produced a feasible solution.

		Usage Level								
		U1			U2			U3		
		Capacity			Capacity			Capacity		
Customers	5	0.00	0.00	0.00	0.00	1.16	0.00	1.94	0.00	0.00
	7	1.12	4.43	0.94	0.00	1.75	1.44	0.00	0.00	1.82
	10	0.00	2.14	2.47	0.71	2.09	3.29	0.84	1.25	2.39
	12	0.00	0.87	2.24	3.13	1.78	2.15	0.51	8.54	6.82
	15	0.00	3.04	7.02	(13.59)	3.21	5.94	3.65	7.74	(24.17)

Table 5: Best optimality gap (%) for clustered instances.

		Usage Level								
		U1			U2			U3		
		Capacity			Capacity			Capacity		
Customers	5	0.00	0.00	8.63	0.00	0.25	0.29	0.00	2.55	0.00
	7	0.00	0.00	8.80	0.00	0.00	0.66	0.00	0.04	3.57
	10	4.15	0.00	5.17	0.00	3.77	0.00	2.32	3.05	6.13
	12	8.54	0.01	5.26	(11.09)	7.46	9.49	0.01	14.38	(24.16)
	15	(18.36)	1.47	(17.62)	(15.62)	(20.25)	(34.39)	7.06	(26.04)	(18.19)

Table 6: Best optimality gap (%) for random instances.

In Figure 7, we summarize these results by means of a histogram that shows the percentage of instances for which a certain optimality gap was achieved. The histogram demon-

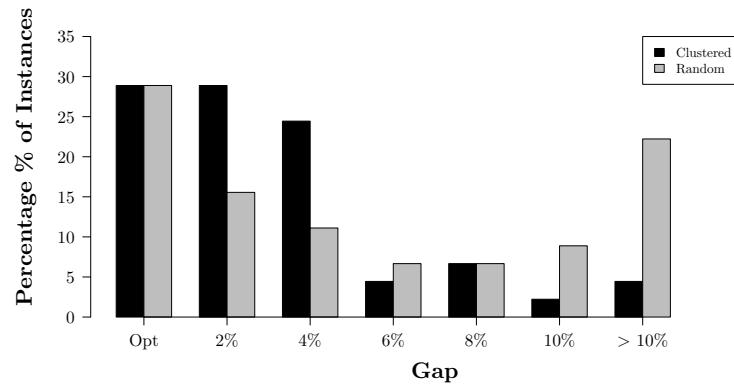


Figure 7: Histogram of instances achieving a certain optimality gap.

strates that for most instances, high-quality solutions are obtained, especially for clustered

instances; one clustered instance with 15 customers was solved to optimality. Twelve out of 90 instances have an optimality gap of more than 10%; most of them random instances with 15 customers.

It is not surprising that when neither the UBM nor the RPO model finds a feasible solution, the resulting gap is large (when we determine the number of vehicles for an instance, we minimize the number of vehicles and do not consider costs). To assess the impact of the available number of vehicles on the ability to obtain low cost solutions, we solved the largest instances with random customer locations assuming one more vehicle was available. The results can be found in Table 7. We see that for all instances a feasible solution was obtained, and that except for Instance R15U2Q2 these solutions are either optimal or close to optimal.

		Usage Level								
		U1			U2			U3		
		Capacity			Capacity			Capacity		
		Q1	Q2	Q3	Q1	Q2	Q3	Q1	Q2	Q3
Customers	15	2.05	0.01	0.12	1.62	14.12	0.98	0.72	2.18	9.45

Table 7: Best optimality gap (%) for random instances with one more vehicle than the best known minimum.

In Figure 8, we focus on the impact of the discretization (i.e., the value of Δ/H) on the bounds for a few select instances. For each value of Δ/H , we show the value of the bound on the cost obtained by solving LBM (“Lower Bound” in the legend), the cost of the feasible delivery schedule extracted by RPO from the solution to LBM, if any (“LBM Feas” in the legend), and the cost of the feasible delivery schedule obtained when solving UBM, if any (“UBM Feas” in the legend). In Appendix D, we provide, for four of these instances, detailed solution statistics (column headings are self-explanatory) for all values of H/Δ .

The most striking observation is that for all these instances, the best lower bound is obtained for the largest value of Δ , i.e., for the discretization with the fewest number of time points. This is both disappointing, because one would expect that a finer discretization should lead to a better bound, but also encouraging, because it suggests that with carefully chosen time points, it may be possible to get good bounds also for larger instances. We also observe that if successful, the RPO model produces high-quality solutions, often optimal solutions, and that the UBM produces many feasible solution, but that their quality is not always high.

To highlight the fact that these instances are non-trivial, we investigate the optimal solution for Instance R7U2Q1. The vehicle itineraries can be found in Table 8 and the customer inventory profiles can be found in Figure 9. We see that three vehicles make multiple trips during the planning horizon and that one vehicle delivers product at four customers on a single trip. Furthermore, we see that all customers receive multiple deliveries during the planning horizon and that one customer receives two consecutive deliveries from a vehicle that waits at its premises. We note that an alternative optimal solution exists

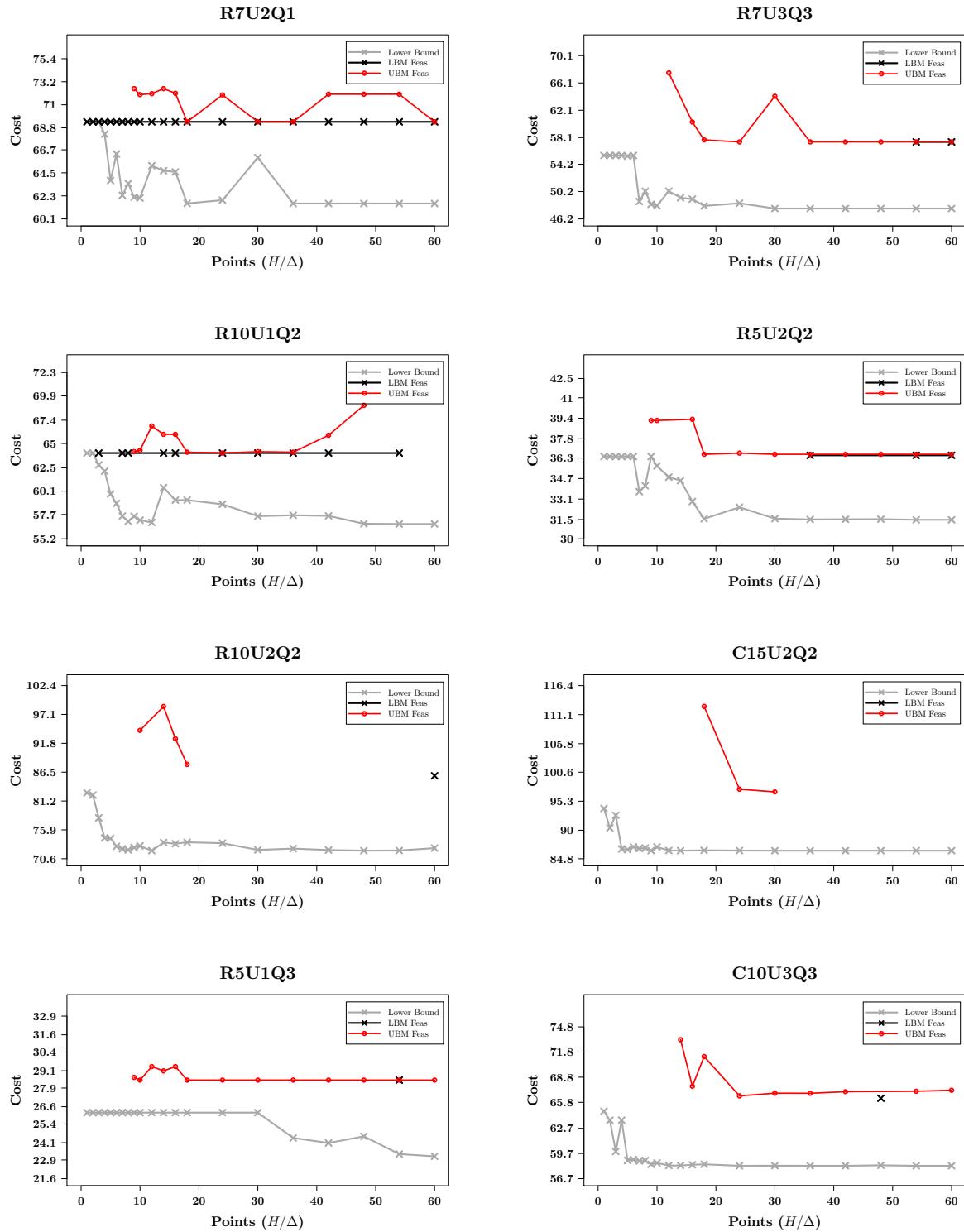


Figure 8: Lower and upper bound value for different values of H/Δ .

Table 8: Vehicle itineraries in the optimal solution to Instance R7U2Q1.

Vehicle	Route			Visit 1			Visit 2			Visit 3		
	t_0	t_{n+1}		i	t_i	q_i	i	t_i	q_i	i	t_i	q_i
1	1	0.00	8.15	1	5.45	66.00	1	12.74	19.80	4	14.59	11.66
	2	8.15	18.00	3	11.75	46.20						
2	1	0.00	18.00	4	4.06	22.34	7	7.00	31.20	4	14.59	11.66
	2	5.57	18.00	5	3.91	51.60						
3	1	0.00	5.57	6	11.01	58.00	5	13.78	55.76	5	13.78	55.76
	2	9.79	18.00	3	6.19	54.46						
4	1	0.00	18.00	2	8.72	35.20	5	13.78	55.76	5	13.78	55.76
	2	9.79	18.00	5	13.00	10.24						
6	1	0.00	18.00	7	9.82	66.00						

that combines the two deliveries into a single delivery (at the time of the first visit). This highlights one of the challenges in solving CIRP instances. There may be many alternative solutions with the same cost.

We presented (and used) a number of valid inequalities to strengthen the linear programming relaxations of both the LBM and the UBM. To show the importance of using these valid inequalities, in Figure 10, we show for Instance R7U2Q1 the best lower and upper bound value for different values of H/Δ when solving the LBM and the UBM with and without the valid inequalities (a $^+$ in the legend is used to indicate that the results are obtained when valid inequalities are added to the formulation).

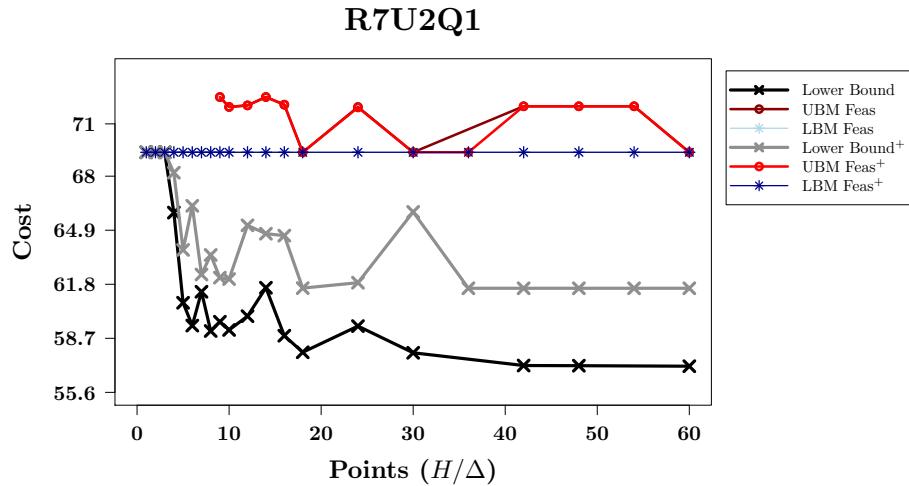


Figure 10: Assessing the value of the valid inequalities for Instance R7U2Q1.

We observe that incorporating the valid inequalities results in improved lower and upper

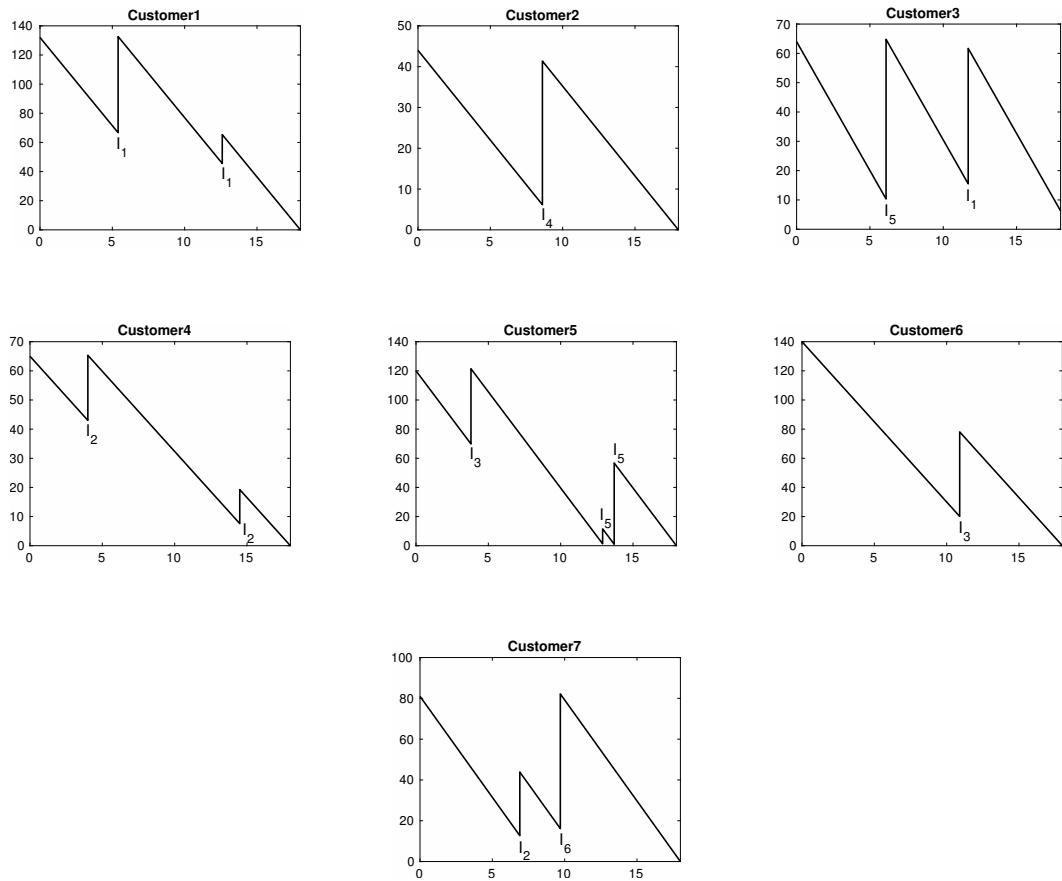


Figure 9: Customer inventory profiles in the optimal solution to Instance R7U2Q1.

bounds for many values of H/Δ . Furthermore, although not visible in the figure, two more feasible delivery schedules were extracted by RPO when starting from solutions to LBM with valid inequalities. (We note that for this particular instance, an optimal solution was found even without using any valid inequalities.)

8 Discussion and future research

In this paper, we have demonstrated that proven optimal solutions to instances of the continuous time inventory routing problem can be obtained using relatively simple time discretization ideas in combination with sophisticated integer programming models. This achievement relies on an integer program that provides a lower bound on the optimal solution value, an integer program that extracts a set of delivery routes from a solution to the lower bound model, and an integer program that seeks to manipulate a set of extracted delivery routes so as to construct a feasible continuous-time solution.

The computational study has revealed that the integer program that provides a lower bound quickly becomes very difficult to solve as the number of time points in the discretization increases. The best lower bounds are typically found for very coarse time discretizations; for finer time discretizations, the integer program does not solve to optimality within the given time limit and only the best bound can be used. Thus, at the moment, the best strategy for obtaining a lower bound for an instance of the CIRP is to solve the integer program that provides a lower bound for a few values of $\Delta = H/k$ with $k \leq 4$.

This, of course, is not entirely satisfactory, which is why we are currently pursuing a full-fledged dynamic discretization discovery algorithm. The solution of the integer program that provides a lower bound for a specific of Δ , can be viewed as a single iteration of a dynamic discretization discovery algorithm. What is missing is a component that analyzes why a set of extracted delivery routes cannot be converted to a feasible (and therefore optimal) continuous-time solution, and uses the results of that analysis to refine the time discretization.

To ensure that a computationally efficient dynamic discretization discovery algorithm results, it is necessary that non-uniform time discretizations can be handled, i.e., the set of time points associated with a location in the partially time-expanded network can depend on the location and the time between two consecutive time points associated with a location can vary. Fortunately, it is not too difficult to extend the lower and upper bound models presented in Sections 5.2 and 6.1 to handle non-uniform discretizations.

Thus, to develop a full-fledged dynamic discretization discovery algorithm that can handle larger CIRP instances, what remains is the design and implementation of a component that analyzes why a set of extracted delivery routes cannot be converted to a feasible continuous-time solution, and uses the results of that analysis to identify time points that can be added to the set of time points at one or more locations and that ensure that the solution to the integer program that produces a lower bound improves. This is easier said than done, and is the focus of our current research.

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant No. 1662848.

References

C. Archetti, L. Bertazzi, G. Laporte, and M. G. Speranza. A branch-and-cut algorithm for a vendor-managed inventory-routing problem. *Transportation Science*, 41(3):382–391, 2007.

P. Avella, M. Boccia, and L. Wolsey. Single-period cutting planes for inventory routing problems. *Transportation Science*, 52(3):497–508, 2017.

W. Bell, L. Dalberto, M. Fisher, A. Greenfield, R. Jaikumar, P. Kedia, R. Mack, and P. Prutzman. Improving the distribution of industrial gases with an on-line computerized routing and scheduling optimizer. *Interfaces*, 13(6):4–23, 1983.

L. Bertazzi, M. Savelsbergh, and M. G. Speranza. Inventory routing. In B. Golden, S. Raghavan, and E. Wasil, editors, *The Vehicle Routing Problem: Latest Advances and New Challenges*, pages 49–72. Springer US, Boston, MA, 2008.

N. Boland, M. Hewitt, L. Marshall, and M. Savelsbergh. The continuous-time service network design problem. *Operations Research*, 65(5):1303–1321, 2017.

A. Campbell and M.W.P. Savelsbergh. A decomposition approach for the inventory routing problem. *Transportation Science*, 38:488–502, 2004a.

A. Campbell and M.W.P. Savelsbergh. Delivery volume optimization. *Transportation Science*, 38: 210–223, 2004b.

A. Campbell, L. Clarke, A. Kleywegt, and M.W.P. Savelsbergh. The inventory routing problem. In T. Crainic and G. Laport, editors, *Fleet Management and Logistics*, pages 95–113. Kluwer Academic Publishers, Norwell, MA, 1998.

A. Campbell, L. Clarke, and M.W.P. Savelsbergh. Inventory routing in practice. In P. Toth and D. Vigo, editors, *The Vehicle Routing Problem*, pages 309–330. SIAM monographs on discrete mathematics and applications, 2002.

L. C. Coelho and G. Laporte. The exact solution of several classes of inventory-routing problems. *Computers & Operations Research*, 40(2):558 – 565, 2013.

L. C. Coelho and G. Laporte. Improved solutions for inventory-routing problems through valid inequalities and input ordering. *International Journal of Production Economics*, 155:391 – 397, 2014. Celebrating a century of the economic order quantity model.

L C Coelho, J F Cordeau, and Gilbert Laporte. Thirty years of inventory routing. *Transportation Science*, 48(1):1–19, 2014.

G. Desaulniers, J. Rakke, and L. C. Coelho. A branch-price-and-cut algorithm for the inventory-routing problem. *Transportation Science*, 50:1060–1076, 2015.

J.-H. Song and M. Savelsbergh. Performance measurement for inventory routing. *Transportation Science*, 41(1):44–54, 2007.

Appendix A – Relation to the model of Bell et al. (1983)

In the model presented in Bell et al. (1983), the customer inventory levels are managed by introducing lower and upper bounds on the (cumulative) delivery quantity based on customer usage rates. This is equivalent to managing inventory levels imposing inventory flow conservation. Indeed, in Bell et al. (1983), the following constraints ensure that the total delivery to customer i during the time interval from 0 to t is feasible,

$$d_{it} \leq \sum_{k \in R_i} \sum_{v \in V_k} \sum_{\tau \in T_k: \tau \leq t - tc_{ik}} x_{ik\tau v} \leq D_{it}, \quad \text{for all } t \text{ and } i \in A_1,$$

where d_{it} is the lower limit on the amount that must be delivered to customer i during the interval from 0 to t , D_{it} is the upper limit on the amount that can be delivered to customer i during the interval from 0 to t , and $x_{ik\tau v}$ is a variable that represents the amount delivered to customer i on route k by vehicle v starting at time τ (and R_i the set of routes that visit i , V_k the set of vehicles that can drive route k , T_k the set of time points at which route k can start, tc_{ik} the time after the start of route k until delivery at customer i is completed, and A_1 the set of customers that can receive any number of deliveries). Note that $\sum_{k \in R_i} \sum_{v \in V_k} \sum_{\tau \in T_k, \tau \leq t - tc_{ik}} x_{ik\tau v}$ represents the total delivery to customer i up to time t .

In our setting, the total delivery to customer i from 0 to t is the following,

$$\sum_{s \leq t} y_i^s = \sum_{0 < s \leq t} (z_i^s - z_i^{s-1} + u_i) + (z_i^0 - I_i^0) = z_i^t + u_i(t-1) - I_i^0.$$

Using the inventory bounds, $u_i \leq z_i^t \leq C_i$, we have

$$u_i t - I_i^0 \leq \sum_{s \leq t} y_i^s \leq C_i - I_i^0 + u_i(t-1),$$

where $u_i t - I_i^0$ is the lower limit on the amount that must be delivered to i (i.e., d_{it}), and $C_i - I_i^0 + u_i(t-1)$ is the upper limit (i.e., D_{it}).

Thus, the model presented in (Bell et al. 1983) has “substituted out” the inventory balance constraints.

Appendix B – Calculating big-M values for depot sub-tour elimination

Here we consider some possible ways to determine the LBM parameter, M_{ij}^t .

If all (original) travel times between pairs of customers are positive, say $\min_{i,j \in N_0, i \neq j} \hat{\tau}_{ij} =: \epsilon > 0$, then the number of vehicle arrivals from another customer (or the depot) in the time remaining after the arrival at j (following a departure from i at time $t\Delta$), may be at most $(H - (t\Delta + \hat{\tau}_{ij}))/\epsilon$. Also, a vehicle can arrive by waiting, of which there can be at most

$T - (t + \tau_{ij})$ cases. However, if $\Delta < \epsilon$, then depot subtours cannot occur, and otherwise, the upper bound obtained by “using” an interval of length Δ to move between customers taking time ϵ is greater than 1, which is all that can be “used” by waiting, so $(H - (t\Delta + \hat{\tau}_{ij}))/\epsilon$ is a valid upper bound on the number of arrivals. Thus we may take $M_{ij}^t = 1 + \lfloor (H - (t\Delta + \hat{\tau}_{ij}))/\epsilon \rfloor$. (The extra 1 is to count the arrival at j after departure from i at time point t .)

Otherwise, if there are co-located customers, for example, so $\hat{\tau}_{ij} = 0$ for some pair $i \neq j$, then if all original parameters are integer, we may use the observation that the exact MIP model has integer solutions, to see that all customer visits must deliver at least 1 unit of product. So we may take $M_{ij}^t = 1 + \sum_{k \in N} \hat{u}_k (H - (t\Delta + \hat{\tau}_{ij}))$, which is an upper bound on the number of units of product that can be delivered in the time remaining.

Appendix C – Route-preserving only (RPO) model

max

s.t.

$$\begin{aligned}
& \zeta \\
& \bar{t}_r^k + \tau_{\rho(r,k)\rho(r,k+1)} \leq \bar{t}_r^{k+1} & r \in R, k = 0, \dots, n_r \\
& \bar{t}_r^{n_r+1} \leq \bar{t}_{\bar{r}}^0 + H(1 - y_{r\bar{r}}) & r \in R, \bar{r} \in R, r \neq \bar{r}, \\
& \sum_{\bar{r} \in R} y_{r\bar{r}} \leq 1 & r \in R, \\
& \sum_{\bar{r} \in R} y_{\bar{r}r} \leq 1 & r \in R, \\
& \sum_{r, \bar{r} \in R} y_{r\bar{r}} \geq |R| - m, \\
& \sum_{k=1}^{n_r} \bar{q}_r^k \leq Q & r \in R \\
& t_i^\ell \geq t_i^{\ell-1} + \zeta & i \in N_0, \ell = 1, \dots, n_i \\
& -H(1 - x_{rk}^{\rho(r,k)\ell}) \leq t_{\rho(r,k)}^\ell - \bar{t}_r^k \leq H(1 - x_{rk}^{\rho(r,k)\ell}) & r \in R, k = 1, \dots, n_r, \ell = 1, \dots, n_{\rho(r,k)}, \\
& -Q(1 - x_{rk}^{\rho(r,k)\ell}) \leq q_{\rho(r,k)}^\ell - \bar{q}_r^k \leq Q(1 - x_{rk}^{\rho(r,k)\ell}) & r \in R, k = 1, \dots, n_r, \ell = 1, \dots, n_{\rho(r,k)} \\
& \sum_{\ell=1}^{n_{\rho(r,k)}} x_{rk}^{\rho(r,k)\ell} = 1 & r \in R, k = 1, \dots, n_r \\
& \sum_{r \in R} \sum_{k=1, \dots, n_r : \rho(r,k)=i} x_{rk}^{\rho(r,k)\ell} = 1 & i \in N, \ell = 1, \dots, n_i \\
& I_i^\ell = I_i^{\ell-1} - u_i (t_i^\ell - t_i^{\ell-1}) + q_i^\ell & i \in N, \ell = 1, \dots, n_i \\
& I_0^i \geq u_i H & i \in N \text{ and } n_i = 0 \\
& I_i^{n_i} - u_i (H - I_i^{n_i}) \geq 0 & i \in N \text{ and } n_i > 0 \\
& q_i^\ell \leq I_i^\ell \leq C_i & i \in N, \ell = 1, \dots, n_i \\
& q_i^\ell \geq 0 & i \in N, \ell = 1, \dots, n_i \\
& t_i^\ell \geq 0 & i \in N, \ell = 1, \dots, n_i
\end{aligned}$$

Appendix D – Detailed computational results for a few instances

Note that the time limit is set to two hours of cpu time, using the equivalent in terms of “ticks” (see CPLEX manual for details). However, we report the observed wall clock time (which can be much more than the cpu time). Note too that UBM was only run for values of $H/\Delta \geq 9$. Finally, we indicate that no feasible solution was found with a dash (-); if no feasible solution to LBM is found, then it is not possible to run RPO. Recall that when the objective function value for RPO is positive, a feasible solution with value equal to the value of the solution to LBM has been constructed.

Instance	H/Δ	LBM			RPO			UBM		
		solution time	bound value	optimality gap	solution time	solution value	solution time	bound value	optimality gap	
R5U2Q2	1	1.86	36.42	0.00	0.01	-	-	-	-	-
	2	12.91	36.42	0.00	0.01	-	-	-	-	-
	3	10.17	36.42	0.00	0.09	-	-	-	-	-
	4	535.68	36.42	0.00	0.13	-	-	-	-	-
	5	1997.23	36.42	0.00	0.07	-	-	-	-	-
	6	8752.82	36.42	0.00	8.99	-	-	-	-	-
	7	TL	33.68	8.13	0.45	-	-	-	-	-
	8	TL	34.14	6.67	2.16	-	-	-	-	-
	9	1751.35	36.42	0.00	0.05	-	-	10.20	39.23	0.00
	10	TL	35.68	2.09	1.72	-	-	154.56	39.23	0.00
	12	TL	34.81	4.63	2.68	-	-	6.93	-	-
	14	TL	34.54	5.46	0.16	-	-	15.86	-	-
	16	TL	32.91	10.67	14.82	-	-	48.40	39.32	0.00
	18	TL	31.56	15.41	19.13	-	-	70.37	36.59	0.00
	24	TL	32.46	12.21	8270.25	-	-	308.10	36.68	0.00
	30	TL	31.57	15.35	24.17	-	-	7312.31	36.59	0.00
	36	TL	31.51	15.88	0.01	0.43	17030.43	36.59	0.00	-
	42	TL	31.52	15.81	TL	-	-	TL	36.59	6.73
	48	TL	31.53	15.50	TL	-	-	TL	36.59	13.68
	54	TL	31.48	15.99	0.02	0.15	-	TL	36.59	8.59
	60	TL	31.48	15.99	127.35	0.29	-	TL	36.59	15.84

R10U1Q2	Instance	H/Δ	LBM			RPO			UBM		
			solution time	bound value	optimality gap	solution time	solution value	solution time	solution value	optimality gap	
R10U1Q2	1	26.61	64.02	0.00	0.02	RPO	RPO	UBM	UBM	UBM	
	2	482.30	64.02	0.00	0.02						
	3	TL	62.80	1.94	6.55		0.73				
	4	TL	62.16	3.00	5.92		-				
	5	TL	59.81	7.04	0.04		-				
	6	TL	58.83	8.82	297.88		-				
	7	TL	57.54	11.26	0.84		0.76				
	8	TL	56.99	12.34	78.54		0.12				
	9	TL	57.51	11.33	37.16		-	305.50	64.16	0.00	
	10	TL	57.11	12.10	11820.51		-	3157.92	64.30	0.00	
	12	TL	56.88	12.55	651.73		-	TL	66.80	8.00	
	14	TL	60.46	5.89	0.02		0.79	1913.11	65.94	0.00	
	16	TL	59.18	8.18	0.02		0.84	TL	65.94	9.36	
	18	TL	59.18	8.19	398.85		-	TL	64.11	5.60	
	24	TL	58.75	8.98	3.63		0.50	TL	64.02	8.92	
	30	TL	57.53	11.28	0.11		0.40	TL	64.16	9.69	
	36	TL	57.62	11.10	0.06		0.50	TL	64.11	11.30	
	42	TL	57.56	11.22	0.08		0.41	TL	65.85	14.97	
	48	TL	56.75	-	-		-	TL	68.93	18.89	
	54	TL	56.72	12.88	0.23		0.42	TL	-	-	
	60	TL	56.72	-	-		-	TL	-	-	

R10U2Q2	Instance	H/Δ		LBM bound value	LBM optimality gap	RPO solution time	RPO solution value	UBM solution time	UBM solution value	UBM optimality gap
		1	2							
	1	8.60	82.71	0.00	0.01					
	2	TL	82.28	0.52	0.00					
	3	TL	78.10	5.91	0.01					
	4	TL	74.43	11.12	51.30					
	5	TL	74.36	11.22	68.61					
	6	TL	72.87	13.50	93.59					
	7	TL	72.41	14.23	0.01					
	8	TL	72.21	14.54	1.62					
	9	TL	72.66	13.83	8.82					
	10	TL	72.94	13.40	629.83					
	12	TL	72.08	14.74	44.13					
	14	TL	73.57	12.43	3486.23					
	16	TL	73.35	12.77	TL					
	18	TL	73.62	12.35	TL					
	24	TL	73.45	12.61	TL					
	30	TL	72.22	14.53	TL					
	36	TL	72.46	-	-					
	42	TL	72.19	-	-					
	48	TL	72.08	-	-					
	54	TL	72.11	-	-					
	60	TL	72.55	18.31	0.07	0.30	TL	-	-	-

	C15U2Q2	Instance		LBM solution time	LBM bound value	LBM optimality gap	RPO solution time	RPO solution value	UBM solution time	UBM solution value	UBM optimality gap
		H/Δ	1								
	1	18.60	93.98	0.00	0.21						
	2	TL	90.42	3.93	0.02						
	3	TL	92.73	1.35	120.94	-					
	4	TL	86.58	8.55	2248.46	-					
	5	TL	86.47	8.68	80.82	-					
	6	TL	86.95	8.09	1998.54	-					
	7	TL	86.67	8.43	2403.24	-					
	8	TL	86.76	8.33	3383.00	-					
	9	TL	86.28	8.92	TL	-	5.62				
	10	TL	86.94	8.10	TL	-	391.61				
	12	TL	86.30	9.15	TL	-	3.22				
	14	TL	86.27	8.94	TL	-	TL				
	18	TL	86.33	9.28	TL	-	TL				
	24	TL	86.29	9.78	TL	-	TL	97.49		6.21	
	30	TL	86.27	10.36	TL	-	TL	97.00		12.44	
	36	TL	86.27	-	-	-	TL	-		-	
	42	TL	86.27	-	-	-	TL	-		-	
	48	TL	86.27	-	-	-	TL	-		-	
	54	TL	86.27	-	-	-	TL	-		-	
	60	TL	86.27	-	-	-	TL	-		-	