Titre: Title:	The Inventory routing problem with demand moves
Auteurs: Authors:	Annelieke C. Baller, Said Dabia, Guy Desaulniers, & Wout E. H. Dullaert
Date:	2021
Type:	Article de revue / Article
Référence: Citation:	Baller, A. C., Dabia, S., Desaulniers, G., & Dullaert, W. E. H. (2021). The Inventory routing problem with demand moves. SN Operations Research Forum, 2(1), 6 (61 pages). https://doi.org/10.1007/s43069-020-00042-z

Document publié chez l'éditeur officiel
Document issued by the official publisher

Titre de la revue: Journal Title:	SN Operations Research Forum (vol. 2, no. 1)
Maison d'édition: Publisher:	Springer Nature
URL officiel: Official URL:	https://doi.org/10.1007/s43069-020-00042-z
Mention légale: Legal notice:	This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommonshorg/licenses/by/4.0/.

ORIGINAL RESEARCH

SN

The Inventory Routing Problem with Demand Moves

Annelieke C. Baller[1] · Said Dabia[1] (ID) **· Guy Desaulniers[2,3] · Wout E. H. Dullaert[1]**

© The Author(s) 2021

Abstract

In the Inventory Routing Problem, customer demand is satisfied from inventory which is replenished with capacitated vehicles. The objective is to minimize total routing and inventory holding cost over a time horizon. If the customers are located relatively close to each other, one has the opportunity to satisfy the demand of a customer by inventory stored at another nearby customer. In the optimization of the customer replenishments, this option can be included to lower total costs. This is for example the case for ATMs in urban areas where an ATM-user that wants to withdraw money could be redirected to another ATM. To the best of our knowledge, the possibility of redirecting end-users is new to the operations research literature and has not been implemented, but is being considered, in the industry. We formulate the Inventory Routing Problem with Demand Moves in which demand of a customer can (partially) be satisfied by the inventory of a nearby customer at a service cost depending on the quantity and the distance. We propose a branch-price-and-cut solution approach which is evaluated on problem instances from the literature. Cost improvements over the classical IRP of up to 10% are observed with average savings around 3%.

1 Introduction

The Inventory Routing Problem (IRP) combines the optimization of inventory management and routing of the replenishments for a set of customers. This problem is relevant in vendor-managed inventory settings, in which a supplier (the vendor) takes both the routing and replenishment decisions. The customers need to be served over

---

Extended author information available on the last page of the article.

a given time horizon and the vendor needs to decide when to replenish each customer, how much to deliver, and how to combine the visits in feasible vehicle routes while minimizing total routing and inventory holding costs. Each customer faces a certain demand per period which must be satisfied from the customer's inventory, this demand is composed of demands of multiple end-users. If the customers are located relatively close to each other, one may have the opportunity to satisfy a part of the demand of a customer by another nearby customer by redirecting some of the end-users. This is for example the case when considering ATMs in urban areas where ATMs are often located in close proximity. This provides the opportunity to redirect ATM-users ("end-users") who want to withdraw money from one ATM to another in case of a cash shortage, hence, to move demand between ATMs ("customers"). The same principle could also be applied to, e.g., urban bike sharing systems. Our business partner considers this a realistic option to reduce their ATM replenishment costs. Moreover, in the future, it might be possible to provide ATM-users with information upfront via a mobile application which can increase customer service by avoiding visits to out-of-service ATMs. Besides that, it is also an option to give the user a small discount if they withdraw cash from certain ATMs. Note that, for example in the Netherlands, a user can withdraw money at every ATM without transaction costs, even if an ATM is not owned by his own bank but by a competing bank. Hence, stimulating a user to withdraw at a certain ATM does not result in transaction costs for the user.

The possibility of redirecting end-users can be incorporated in the optimization of the customer replenishments to reduce total costs. We define the Inventory Routing Problem with Demand Moves (IRPDM) in which demand of a customer can (partially) be satisfied by another customer. We assume that a service cost is incurred by the vendor for each demand move. This cost can, for example, reflect a loss in service experienced by the end-user or the actual cost of the discount provided to the end-users. Hence, the IRPDM consists of deciding on the timing and quantity of deliveries to each customer, both to satisfy its own demand and potential demand moved from other customers to this customer, deciding on the vehicle routes to perform the replenishments and on demand moves between customers. The objective of the IRPDM is to minimize the total routing, inventory holding and service costs.

The contributions of this paper can be summarized as follows. We introduce the notion of demand moves and define the IRPDM. We solve the problem with a branch-price-and-cut (BPC) solution method based on the approach by Desaulniers et al. [1] for the IRP. Valid Inequalities (VIs) from the IRP literature are not directly applicable to the IRPDM. Non-trivial modifications of these inequalities are proposed to ensure that they capture the effect of the demand moves in the IRPDM. Experiments on synthetic IRP instances from the literature illustrate the performance of the proposed solution approach and offer insight in the benefits of allowing demand moves in inventory routing problems. Moreover, sensitivity analysis, for example on the service costs, is conducted to derive managerial insights.

The paper is organized as follows. Section 2 provides an overview of related literature. In Section 3, we formally describe the problem and we present a mathematical programming formulation for the IRPDM. Section 4 describes the solution method and contains an extensive description of the VIs. The results of the computational

experiments are detailed in Section 5. Finally, Section 6 discusses conclusions and directions for future research.

2 Literature

Inspired by a real-life case on ATM replenishment, this paper contributes to a recent stream of papers on cash supply chains. Van Anholt et al. [2] propose a three-step heuristic to solve a combined inventory management and routing problem for so-called Recirculation ATMs (RATM). At an RATM, a user can both withdraw and deposit money; hence, the IRP solutions contain both delivery and pick-up activities. Money that is picked up at one ATM can be used for a replenishment of another ATM. Hence, transshipments performed by a vehicle are included in the model. Instances are based on real-life data with up to 200 customers and one vehicle per depot for a planning horizon of 6 days. Larrain et al. [3] consider an IRP which allows stock-outs and the replenishment policy consists of swapping new cassettes of a chosen amount for the old cassettes that can still contain bank notes. The authors propose a matheuristic to solve instances with up to 60 locations, 3 vehicles, and up to 18 periods (in at most 6 days). Geismar et al. [4] provide a literature overview on currency supply chains by reviewing studies that look into the supply chain from the supply side (national banks), the demand side (commercial banks and ATM networks), and the private sector logistics providers' side. In their analysis of ATM replenishment-related literature, Geismar et al. [4] mention the study by Van Anholt et al. [2] on RATMs and suggest as future research to investigate possible incentives to rebalance RATM inventories by steering users to a certain RATM (either withdraw from a full RATM or deposit at an empty RATM). As incentive they suggest a premium for making a deposit at a certain RATM and these premiums can be reviewed online by the user. In this paper, we investigate the IRPDM to examine possible supply chain savings when implementing a similar system for ATMs.

The IRPDM is related to the IRP with Transshipment (IRPT) introduced by Coelho et al. [5]. In the IRPT, one can move goods from an origin customer or depot to a destination customer in order to redistribute merchandise between stores of the same chain to cover unexpected demand variations, redistribute inventory to reduce handling costs, and in case storage capacity is limited at certain locations. It is assumed that these transshipments are performed by an outsourced carrier. Coelho et al. [5] propose an ILP formulation for this problem and develop an adaptive large neighborhood search heuristic for the single vehicle case. The heuristic is tested on instances from the literature with one vehicle, with up to 30 customers and 6 periods and with up to 50 customers and 3 periods. The heuristic's stopping criterion is 25,000 iterations or 1 h of computation time (which was reached for some of the largest instances). Coelho and Laporte [6] use a branch-and-cut (BC) algorithm without problem specific VIs for the IRPT, solving instances up to 30 customers with 6 periods and 50 customers with 3 periods with a maximum running time of 12 hours per instance. Lefever et al. [7] propose an improved formulation for the IRPT which is solved by BC and uses two problem specific VIs. Compared with [6], the computation times are lower and two more instances with 6 periods are solved.

On the one hand, for certain features, the IRPDM can be seen as a special case of the IRPT. First, in the IRPT, it is possible to transship goods and store the goods at the destination customer to be used during multiple periods. In the IRPDM, the goods are not transshipped, but the demand is moved. Therefore, a demand move in one direction, is equivalent to a transshipment of goods in the other direction if these transshipped goods are immediately consumed. Second, in the IRPDM, demand moves to the depot are not possible while in the IRPT, goods can be moved directly from the depot to a customer.

On the other hand, the IRPDM contains some features that generalize the IRPT. First, we handle the multiple vehicle case, while in [5] and [6], only the single vehicle case is considered. Second, in the IRPDM, a large distance between customers would make a demand move impractical. Therefore, we restrict demand moves for each customer to a small subset of neighboring customers in close proximity and this subset can be different for each customer. In contrast, both in [5] and [6], the set of customers from which goods can be transshipped is limited to a depot and a subset of the customers, and this set is fixed for all customers. Hence, there is a set of "source" locations from which goods can be transshipped to any other customer. This can be modeled as a special case in the framework of neighbors that we define for the IRPDM. Although the limitation to a general set of source locations is modeled in [5], this feature does not seem to be used in the computational experiments. Concluding, in the IRPDM, any customer can be a "source" location, not only a predetermined subset of the customers, and in the IRPDM, the "source" customers can be different for each customer.

For extensive surveys on variants and solution methods for the IRP, we refer to [8] and [9]. Next to solution methods mentioned in these surveys, Desaulniers et al. [1] present a new formulation for the IRP that performs better for instances with multiple vehicles. In [1], a BPC algorithm is proposed to solve the IRP. In the model, columns represent a combination of a route and a so-called route delivery pattern (RDP) specifying the quantity delivered to each customer along the route. In the master problem, the optimal combinations of routes and RDPs are selected to minimize total routing and inventory holding costs. In the pricing problem (PP), routes and associated RDPs are generated based on the dual variables retrieved from the master problem. To model demand moves in the IRP, we extend the IRP formulation as introduced by Desaulniers et al. [1]. The main differences are the handling of initial inventory at the customers at the beginning of the planning horizon, the introduction of the neighboring customers, and the non-trivial adjustments to the VIs. Section 3 provides more details on the IRP formulation by Desaulniers et al. [1] and the extension for the IRPDM.

3 Problem Description and Formulation

In the IRPDM, a supplier replenishes inventory for a set of customers over a certain planning horizon. The supplier has an initial inventory at the beginning of the

planning horizon and a known quantity becomes available in each period. A given number of vehicles with a load capacity restriction is available to perform the replenishments. A customer can be serviced at most once per period, i.e., split deliveries are not allowed. Each customer has an initial inventory at the beginning of the planning horizon, faces a given demand per period, and has an inventory capacity that must be respected. Via demand moves, part of a customer's demand can be satisfied by another nearby customer. Note that we consider moving a part of the demand of the customer (the ATM) which implies that in practice, the demand of several end-users of the customer is moved. For each demand move, a cost is incurred that depends on the quantity moved and the distance between the customers involved. The other costs consist of routing costs for the distance traveled by the vehicles and inventory holding costs at the supplier and the customers. The objective of the IRPDM is to minimize the routing, inventory holding, and demand move costs. The inventory holding costs are charged on the quantity in stock at the end of each period assuming the following order of events in a period: increase inventory at the supplier, delivery of goods to the customers, consumption, inventory calculation. This order of events coincides with most literature on the IRP [10].

More formally, a single supplier, denoted by 0, needs to serve a set of customers N over a time horizon $P = \{1, 2, \ldots, \rho\}$. A fictitious period $\rho + 1$ is considered to handle end inventories. At each discrete time moment $p \in P$, a quantity d_0^p becomes available at the supplier 0 and each customer $i \in N$ faces a demand d_i^p. A homogeneous fleet of K vehicles with capacity Q is available to deliver the goods to the customers. For each customer $i \in N$, a holding capacity C_i needs to be respected and backlogging is not allowed. A customer $i \in N$ (the supplier 0) has an initial inventory I_i^0 (I_0^0) and a unit holding cost h_i (h_0). A Maximum Level inventory policy is adopted for the customers which means the delivery quantity can be chosen freely as long as inventory capacity is respected. The distance between the depot and each customer, and between all customers is given and denoted by c_{ij}. Each customer can redirect (part of) its demand to another customer. Therefore, for each customer $i \in N$, a set of neighboring customers $\mathcal{N}_i$ is established; the demand of the customers in $\mathcal{N}_i$ can be satisfied by i via demand moves. So, if a demand move from i to j is possible, this means that $i \in \mathcal{N}_j$, i.e., j can satisfy demand of i. If a demand move takes place, a cost m_{ij} is charged per unit moved and per unit distance between i and j. Adding these costs in the objective is similar to incorporating costs of, for example, backlogging [11] and lost sales [3] in IRP settings.

In most IRP formulations, the model involves variables indicating the quantity delivered in a period to a certain customer. Inventory balance constraints keep track of the use of the inventory to satisfy the demand in the different periods. In contrast, in the formulation for the IRP proposed by Desaulniers et al. [1], the model uses variables indicating for which periods the delivered quantities are dedicated. The authors use the fact that there always exists an optimal solution that respects the first-in, first-out (FIFO) principle. Hence, it is possible, given the period of delivery, to determine the periods to which a delivered quantity is assigned. Moreover, if there is initial inventory, FIFO implies that this inventory is used to cover the demand

in the first periods of the planning horizon. Therefore, so-called residual demands $\bar{d}_i^p$ can be calculated. In periods for which the initial inventory cannot cover the demand, customers have a positive residual demand. Constraints that make sure all demand is covered are only needed for periods with a positive residual demand. In the IRPDM, initial inventory can also be used to satisfy moved demand of another customer. Therefore, we cannot use residual demands as in [1], but we have to model the use of the initial demand explicitly. This also implies that these constraints are needed for all customers and all periods.

Given the FIFO rule, let $I_i^{0,s} = \max\{0, I_i^0 - \sum_{\ell=1}^s d_i^\ell\}$ be the quantity remaining from initial inventory at customer i at the end of period s if initial inventory is only used to satisfy demand of customer i itself. Let P_{ijp}^+ denote the subset of periods associated with subdeliveries delivered in period $p \in P$ to customer $i \in N$ dedicated to customer $j \in \mathcal{N}_i \cup \{i\}$. The deliveries for periods P_{ijp}^+ and customers $j \in \mathcal{N}_i \cup \{i\}$ are so-called subdeliveries. Note that a subdelivery can be zero, then no delivery is made for the corresponding period and customer. The set P_{ijp}^+ can be determined as follows:

$$
P_{ijp}^+ = \begin{cases}
\left\{ s \in \{p, p+1, \ldots, \rho+1\} \mid \sum_{l=p}^{s-1} d_i^l < C_i \right\} & \text{if } i = j \text{ and } \mathcal{N}_i \neq \emptyset \\[2ex]
\left\{ s \in \{p, p+1, \ldots, \rho+1\} \mid \right. \\
\quad \left(s \in P, \bar{d}_i^s > 0 \text{ and } (s = p \text{ or } \sum_{l=p}^{s-1} d_i^l + I_i^{0,s-1} < C_i) \right) \\
\quad \text{or } \left. \left(s = \rho+1 \text{ and } \sum_{l=p}^{s-1} d_i^l + I_i^{0,s-1} < C_i \right) \right\} & \text{if } i = j \text{ and } \mathcal{N}_i = \emptyset \\[2ex]
\left\{ s \in \{p, p+1, \ldots, \rho\} \mid \sum_{l=p}^{s} d_i^l < C_i \right\} & i \neq j.
\end{cases}
$$

The set P_{ijp}^+ should be large enough such that possible solutions for the IRP are not excluded, which can be derived as follows. If $i = j$ and $\mathcal{N}_i \neq \emptyset$, there are neighbors for which the initial inventory can satisfy the demand. Set P_{ijp}^+ is then largest if all periods are included such that inventory capacity of the customer is not exceeded by the total delivery made. If $i = j$ and $\mathcal{N}_i = \emptyset$, all initial inventory is used to satisfy demand of customer i itself, and subdeliveries are only needed for periods with a positive residual demand, and such that inventory capacity is not exceeded. If $i \neq j$, because of the FIFO principle, demand of customer i needs to be satisfied from inventory before satisfying (part of) the demand of a neighbor j. This means that a subdelivery for a customer j in a period s is possible if the total demand of customer i up to and including period s does not exceed C_i. For ease of notation, denote $P_{ip}^+ = P_{iip}^+$, and introduce $P_{ip}^{+\ell}$ denoting the latest period in set P_{ip}^+.

Let u_{ijp}^s denote the upper bound on the quantity of a subdelivery in period $s \in P_{ijp}^+$. For the visited customer i $u_{ip}^s := u_{iip}^s$ is computed as follows:

$$
u_{ip}^s = u_{iip}^s = \begin{cases}
\min\{d_i^s, C_i - I_i^0\} & \text{if } s = p = 0 \\
\min\{d_i^s, C_i\} & \text{if } s = p \neq 0 \\
C_i - \sum_{\ell=p}^{s-1} d_i^\ell & \text{if } s = \rho+1 \\
\min\{d_i^s, C_i - \sum_{\ell=p}^{s-1} d_i^\ell\} & \text{otherwise}
\end{cases}
$$

and the upper bound u_{ijp}^s for a neighboring customer $j \in \mathcal{N}_i$ is given by:

$$
u_{ijp}^s = \begin{cases}
\min\left\{ d_j^s, C_i - I_i^0 - \bar{d}_i^s \right\} & \text{if } s = p = 0 \\
\min\left\{ d_j^s, C_i - d_i^s \right\} & \text{if } s = p \neq 0 \\
0 & \text{if } s = \rho + 1 \\
\min\left\{ d_j^s, C_i - \sum_{\ell=p}^{s} d_i^\ell \right\} & \text{otherwise.}
\end{cases}
$$

Note that delivering goods for the fictitious ending period will be in inventory at the same customer $i \in N$, no matter whether these goods are dedicated to the customer itself or one of its neighbors. Therefore, without changing the solutions, we set the upper bound for the quantity dedicated to each neighbor to 0 for this fictitious period. Also note that it is never optimal to have an incoming demand move and outgoing demand move in the same period. Therefore, first the customer's own demand needs to be satisfied before using goods in inventory to satisfy the demand of a neighbor. This influences the upper bound on the delivered quantity that is dedicated to a neighbor. The set of feasible routes is denoted by R. For each route, $r \in R$, N_r indicates the set of visited customers and A_r the set of arcs used in the route. Let a_{ri} be equal to 1 if customer $i \in N$ is visited in route $r \in R$ and 0 otherwise. An RDP w corresponding to period p details the quantities $q_{wij}^s \in [0, u_{ijp}^s]$ delivered to customer $i \in N_r$ dedicated to satisfy the demand of customer $j \in \mathcal{N}_i \cup \{i\}$ in period $s \in P_{ijp}^+$. As in [1], $q_{wij}^s = 0$ corresponds to a *zero subdelivery*, $q_{wij}^s = u_{ijp}^s$ to a *full subdelivery*, and a *partial subdelivery* if $0 < q_{wij}^s < u_{ijp}^s$. An extreme RDP contains at most one partial subdelivery. A set of extreme RDPs W_r^p is associated with each route $r \in R$ in period $p \in P$. Note that with a convex combination of multiple extreme RDPs any combination of delivered quantities can be constructed. The total quantity delivered in RDP $w \in W_r^p$ is denoted by $q_w = \sum_{i \in N_r} \sum_{j \in \mathcal{N}_i \cup \{i\}} \sum_{s \in P_{ijp}^+} q_{wij}^s$.

Given a route $r \in R$ and an extreme RDP $w \in W_r^p$, we can identify the quantity $\hat{b}_{wi}^s$ delivered to customer $i \in N$ that will be in inventory at the end of period $p \le s \le P_{ip}^{+\ell}$. Compared with [1], we use $\hat{b}_{wi}^s$ instead of b_{wi}^s to indicate that in our case deliveries dedicated to neighboring customers are also included. Let $c_{rw} = \sum_{(i,j) \in A_r} c_{ij} + \sum_{i \in N_r} \sum_{s \in P_{ip}^+} h_i \hat{b}_{wi}^s$ be the costs associated with route r and RDP w in which the first term is the routing costs and the second term is the inventory holding costs of all units delivered to the visited customers. Note that a unit dedicated to satisfy the demand of a neighboring customer stays in inventory at the customer until consumption. Denote by $P_{ijs}^- = \{ p \in P \mid s \in P_{ijp}^+ \}$ the set of periods in which a subdelivery can be made to customer $i \in N$ to satisfy the demand of customer $j \in \mathcal{N}_i \cup \{i\}$ in period $s \in P$. Use P_{is}^- to represent the union of sets P_{ijs}^- over all $j \in \mathcal{N}_i \cup \{i\}$.

To model the IRPDM, we use the following decision variables. Continuous variables $y_{rw}^p \in [0, 1]$ are indicating the proportion of route $r \in R$ with RDP $w \in W_r^p$ in period $p \in P$. To clarify some of the notation, Figure 1 illustrates a variable y_{rw}^p for a route 0-1-2-0 and RDP w in a period p, corresponding delivery quantities q_{wij}^p for the visited customers (solid arrows) and the involved demand move (dotted arrow).

Nonnegative variables I_0^p indicate the inventory level at the supplier at the end of period $p \in P$. Nonnegative, integer variables z_{ij}^p indicate the quantity out of initial inventory at customer $i \in N$ used to satisfy demand of customer $j \in \mathcal{N}_i \cup \{i\}$ in period $p \in P$.

To comply with the FIFO principle, we prevent a demand move from i to j if customer i still has inventory left. Therefore, we introduce binary decision variables v_i^s. This variable will be equal to 1 if there is a positive inventory level at customer $i \in N$ at the end of period $s \in P$ and 0 otherwise. Hence, if $v_i^s = 0$, a demand move from customer i to j ($i \in \mathcal{N}_j$) in period s is possible. If $v_i^s = 1$, customer i still has inventory left that should be used first and a demand move is definitely not possible.

We can now formulate the IRPDM as follows:

$$\min \quad \sum_{p\in P}\sum_{r\in R}\sum_{w\in W_r^p} c_{rw} y_{rw}^p + \sum_{p\in P} h_0 I_0^p + \sum_{p\in P}\sum_{i\in N}\left(I_i^0 - \sum_{s\leq p}\sum_{j\in\mathcal{N}_i\cup\{i\}} z_{ij}^p\right) h_i$$

$$+ \sum_{p\in P}\sum_{r\in R}\sum_{w\in W_r^p}\sum_{i\in N_r}\sum_{j\in\mathcal{N}_i}\sum_{s\in P_{ijp}^+} m_{ij} q_{wij}^s y_{rw}^p + \sum_{p\in P}\sum_{i\in N}\sum_{j\in\mathcal{N}_i} m_{ij} z_{ij}^p \tag{1a}$$

$$\text{s.t.} \quad I_0^{p-1} + d_0^p - \sum_{r\in R}\sum_{w\in W_r^p} q_w y_{rw}^p = I_0^p, \qquad\qquad \forall p \in P, \tag{1b}$$

$$\sum_{i:j\in\mathcal{N}_i\cup\{j\}}\sum_{p\in P_{ijs}^-}\sum_{r\in R}\sum_{w\in W_r^p} q_{wij}^s y_{rw}^p + \sum_{i:j\in\mathcal{N}_i\cup\{j\}} z_{ij}^s = d_j^s, \qquad \forall j \in N, \forall s \in P, \tag{1c}$$

$$I_i^0 - \sum_{p\leq s}\sum_{j\in\mathcal{N}_i\cup\{i\}} z_{ij}^p + \sum_{p\in P_{is}^-}\sum_{r\in R}\sum_{w\in W_r^p} \hat{b}_{wi}^s y_{rw}^p$$

$$+ \sum_{j\in\mathcal{N}_i}\sum_{p\in P_{ijs}^-}\sum_{r\in R}\sum_{w\in W_r^p} q_{wij}^s y_{rw}^p + d_i^s \leq C_i, \qquad \forall i \in N, \forall s \in P, \tag{1d}$$

$$\sum_{r\in R}\sum_{w\in W_r^p} a_{ri} y_{rw}^p \leq 1, \qquad\qquad \forall i \in N, \forall p \in P, \tag{1e}$$

$$\sum_{r\in R}\sum_{w\in W_r^p} y_{rw}^p \leq K, \qquad\qquad \forall p \in P, \tag{1f}$$

$$\sum_{p\in P}\sum_{j\in\mathcal{N}_i\cup\{i\}} z_{ij}^p \leq I_i^0, \quad \forall i \in N, \tag{1g}$$

$$I_i^0 - \sum_{p\leq s}\sum_{j\in\mathcal{N}_i\cup\{i\}} z_{ij}^p + \sum_{p\in P_{is}^-}\sum_{r\in R}\sum_{w\in W_r^p} \hat{b}_{wi}^s y_{rw}^p \leq \left(C_i - d_i^s\right) v_i^s, \qquad \forall i \in N, \forall s \in P, \tag{1h}$$

$$\sum_{i:j\in\mathcal{N}_i}\sum_{p\in P_{ijs}^-}\sum_{r\in R}\sum_{w\in W_r^p} q_{wij}^s y_{rw}^p + \sum_{i:j\in\mathcal{N}_i} z_{ij}^p \leq d_j^s\left(1 - v_j^s\right), \qquad \forall j \in N, \forall s \in P, \tag{1i}$$

$$0 \leq I_0^p \leq C_0, \qquad\qquad \forall p \in P, \tag{1j}$$

$$z_{ij}^p \in \mathbb{N}, \qquad\qquad \forall i \in N, \forall j \in N, \forall p \in P, \tag{1k}$$

$$y_{rw}^p \geq 0, \qquad\qquad \forall p \in P, \forall r \in R, \forall w \in W_r^p, \tag{1l}$$

$$\sum_{w\in W_r^p} y_{rw}^p \in \{0,1\}, \qquad\qquad \forall p \in P, \forall r \in R, \tag{1m}$$

$$v_i^s \in \{0,1\}, \qquad\qquad \forall i \in N, \forall s \in P. \tag{1n}$$

The objective function (1a) minimizes the total routing, inventory holding, and demand move costs. Note that the demand move costs consist of the moves satisfied by initial inventory and by deliveries made during the planning period. Constraints
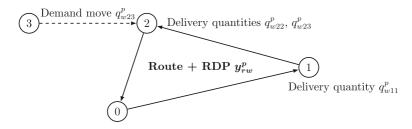
Fig. 1 Visualization of a route and RDP variable, delivery quantities, and a demand move

(1b) balance the inventory level at the supplier between periods. Constraints (1c) make sure that, for each customer $j \in N$, all demand is satisfied by deliveries to customer j itself, to one of the customers i for which $j \in \mathcal{N}_i$, or from initial inventory. Constraints (1d) are the capacity constraints at the customers. Note that the inventory level at customer i in a period is highest after the deliveries and before consumption. The highest inventory level is thus equal to the remaining initial inventory, plus past deliveries (dedicated to i or to $j \in \mathcal{N}_i$) that are not consumed yet at the end of the period, plus the demand satisfied for other customers $j \in \mathcal{N}_i$ in this period and the demand at i in this period. Split deliveries are prevented by constraints (1e) and the number of used vehicles per period is limited by constraints (1f). Constraints (1g) make sure that the amount used from initial inventory does not exceed the actual initial inventory. In (1h), the left hand side is equal to the ending inventory at customer i in period s. If the ending inventory is positive, variable v_i^s must be equal to 1. Note that in this case, constraints (1d) are more restrictive than (1h). If $v_i^s = 0$ then there cannot be any ending inventory. By constraints (1i), a demand move from j to i can only occur if $v_j^s = 0$ which implies that there cannot be any ending inventory in the same period.

Note that the maximum number of periods in which demand can be satisfied from initial inventory is limited, for example, if the demand is the same every period, the number of periods is $\lceil I_i^0/d_i \rceil$. Hence, the variables z_{ij}^p only need to be introduced for those periods.

3.1 Limiting the Moved Demand

In the current formulation of the problem, it is possible that one customer is never replenished by a vehicle, but that all of its demand is satisfied from the customer's initial inventory and via demand moves. In practice, this might not be desirable, for example for an ATM that is closest to a hospital. Therefore, as a general rule, we limit the quantity that is satisfied by another customer via demand moves to a certain percentage θ of the demand. The left hand side of the constraint should be the quantity of the demand of customer $j \in N$ in period $s \in P$ satisfied via demand moves, either via a delivery to another customer i such that $j \in \mathcal{N}_i$ or from initial inventory. The right hand side should limit the quantity to θd_j^s. The left hand side of this constraint

is identical to the left hand side of constraint (1i), hence, we can merge the two types of constraints as follows:

$$\sum_{i:j\in\mathcal{N}_i}\sum_{p\in P_{ijs}^-}\sum_{r\in R}\sum_{w\in W_r^p} q_{wij}^s y_{rw}^p + \sum_{i:j\in\mathcal{N}_i} z_{ij}^p \leq \theta d_j^s\left(1-v_j^s\right) \quad \forall j\in N, \forall s\in P \quad (2)$$

3.2 Using Initial Inventory for Demand Moves

In Section 4, we will present a BPC algorithm to solve the IRPDM. Existing cuts for the IRP are no longer valid and need to be adjusted to handle demand moves. Because initial inventory can be used to satisfy moved demand, it is not straightforward how the cuts can be properly adjusted. Therefore, we first study a simplified variant of the problem, in which initial inventory can only be used to satisfy demand of the customer itself. In the remainder of the paper, we consider this variant of the problem, unless indicated otherwise. In formulation (1a)–(1n), this variant can be modeled by setting $z_{ij}^p = 0$ for all $i \neq j$ or by considering a formulation with residual demands as in [1]. Moreover, we need to adjust the calculations of P_{ijp}^+ and u_{ijp}^s since goods only need to be delivered for periods with a positive residual demand. The set of periods to which a delivery in period p to customer i can be dedicated is given by:

$$P_{ijp}^+ = \begin{cases} \left\{s\in\{p,p+1,\ldots,\rho+1\}|\left(s\in P, \bar{d}_i^s > 0 \text{ and } (s=p \text{ or } \sum_{l=p}^{s-1}d_i^l + I_i^{0,s-1} < C_i)\right)\right. \\ \left. \text{or } \left(s=\rho+1 \text{ and } \sum_{l=p}^{s-1}d_i^l + I_i^{0,s-1} < C_i\right)\right\} & \text{if } i=j \\ \left\{s\in\{p,p+1,\ldots,\rho\}|\bar{d}_j^s > 0 \text{ and } \sum_{l=p}^s d_i^l + I_i^{0,s} < C_i\right\} & \text{otherwise} \end{cases}$$

An upper bound u_{ip}^s on the quantity dedicated to each period $s\in P_{ijp}^+$ is computed as follows for the visited customer:

$$u_{ip}^s = u_{iip}^s = \begin{cases} \min\left\{\bar{d}_i^s, C_i - I_i^{0,s-1}\right\} & \text{if } s=p \\ C_i - \sum_{\ell=p}^{s-1}d_i^\ell - I_i^{0,s-1} & \text{if } s=\rho+1 \\ \min\left\{\bar{d}_i^s, C_i - \sum_{\ell=p}^{s-1}d_i^\ell - I_i^{0,s-1}\right\} & \text{otherwise} \end{cases}$$

and the upper bound u_{ijp}^s for a neighboring customer $j\in\mathcal{N}_i$ is given by:

$$u_{ijp}^s = \begin{cases} \min\left\{\bar{d}_j^s, C_i - I_i^{0,s-1} - \bar{d}_i^s\right\} & \text{if } s=p=1 \\ 0 & \text{if } s=\rho+1 \\ \min\left\{\bar{d}_j^s, C_i - \sum_{\ell=p}^{s-1}d_i^\ell - I_i^{0,s-1} - \bar{d}_i^s\right\} & \text{otherwise} \end{cases}$$

4 Solution Method

A BPC method is used to solve model (1a)–(1n). Column generation is applied to the master problem consisting of the linear relaxation of (1a)–(1l) and (1n) to compute lower bounds within a branch-and-bound algorithm. Columns represent a route and an extreme RDP, and these are generated by the PP. This solution method can be applied to both the case in which the initial inventory can be used to satisfy moved demand and the case in which initial inventory cannot be used for this purpose. VIs are added dynamically to the master problem to tighten the linear relaxations. The

VIs are based on inequalities proposed in the literature for the IRP, we adjust these for the case in which initial inventory cannot be used to satisfy moved demand. An integer feasible solution is found by branching on the appropriate variables. Below, we describe the column generation process, the PP, the VIs, and the branching procedure.

4.1 Column Generation

Column generation is an iterative procedure that solves a linear program (LP). The procedure to solve the linear relaxation of (1a)–(1l) and (1n) starts with an LP with a limited set of variables y_{rw}^p, which is called the restricted master problem (RMP). Then, new variables are added which are found by solving one or more PPs and with these new variables, the RMP is resolved. The PPs generate negative reduced cost variables y_{rw}^p (also called columns) with respect to the dual values of the current RMP. This process continues until the PPs do not generate new variables.

Initially, artificial columns with very high costs are added to guarantee a feasible solution for the RMP, such that dual values can be retrieved to be used in the PP. To obtain a better initial solution, an additional set of columns is computed in the following greedy way. Consider, for each period p, the customers $\mathcal{S}$ that have residual demand in this period, if there are none, consider the customers with residual demand in period $p+1$. For each customer, consider the RDP with a full delivery in period p (or $p+1$) and zero deliveries for other periods. Create a route to visit the customers in $\mathcal{S}$ by applying the nearest neighbor heuristic starting at the depot and adding customers as long as vehicle capacity is not violated. Each customer that is added to the route is marked as visited. If no customers in $\mathcal{S}$ can be added anymore without violating vehicle capacity, the route is finished. If there are still unvisited customers in $\mathcal{S}$, create another route.

4.2 Pricing Problem

For the IRPDM, there is a PP for each period $p \in P$ in the planning horizon. A column generated by the PP corresponds to a delivery route $r \in R$ and an extreme RDP $w \in W_r^p$ that are feasible with respect to the constraints of the problem. Hence, the PP consists of a routing part and a delivery part which results in solving an Elementary Shortest Path Problem with Resource Constraints (ESPPRC) combined with the linear relaxation of a knapsack problem. After providing more explanation on the PP, the details on solving the ESPPRC will be discussed in Section 4.2.1. The PP for the IRPDM is an extension of the one for the IRP in [1].

Associate dual variables π_p^{1b}, π_{is}^{1c}, π_{is}^{1d}, π_{ip}^{1e}, π_p^{1f}, π_{is}^{1h}, and π_{js}^{1i} with constraints (1b)–(1f) and (1h)–(1i) respectively. The reduced cost of a variable y_{rw}^p is given by:

$$\bar{c}_{rw}^p = c_{rw} + \sum_{i \in N_r} \sum_{j \in \mathcal{N}_i} \sum_{s \in P_{ijp}^+} m_{ij} q_{wij}^s + q_w \pi_p^{1b} - \sum_{i \in N_r} \sum_{j \in \mathcal{N}_i \cup \{i\}} \sum_{s \in P_{ijp}^+} q_{wij}^s \pi_{js}^{1c} - \sum_{i \in N_r} \sum_{s=p}^{P_{ip}^{+\ell}} \hat{b}_{wi}^s \pi_{is}^{1d}$$

$$- \sum_{i \in N_r} \sum_{j \in \mathcal{N}_i} \sum_{s \in P_{ijp}^+} q_{wij}^s \pi_{is}^{1d} - \sum_{i \in N_r} \pi_{ip}^{1e} - \pi_p^{1f} - \sum_{i \in N_r} \sum_{s=p}^{P_{ip}^{+\ell}} \hat{b}_{wi}^s \pi_{is}^{1h} - \sum_{i \in N_r} \sum_{j \in \mathcal{N}_i} \sum_{s \in P_{ijp}^+} q_{wij}^s \pi_{js}^{1i} \quad (3)$$

in which $c_{rw} = \sum_{(i,j)\in A_r} c_{ij} + \sum_{i\in N_r} \sum_{s=p}^{P_{ip}^{+\ell}} h_i \hat{b}_{wi}^s$ are the routing and inventory holding costs for a route r and RDP w.

For the routing part of the problem, define a graph $G^P = (V^P, A^P)$ in which V^P is the set of nodes and A^P is the set of arcs with arc travel costs c_{ij}, $(i, j) \in A^P$. The set of nodes includes nodes corresponding to the customers v^i, and to a depot source node v^S and sink node v^E. Set A^P contains all arcs between the customers $(i, j) \in N \times N$, $i \neq j$, all arcs from the source node (v^S, i), $i \in N$ and all arcs entering the sink node (i, v^E), $i \in N$. In the ESPPRC, define the cost of an arc to be:

$$\bar{c}_{ij} = \begin{cases} c_{ij} - \pi_p^{1f} & \text{if } i = v^S \\ c_{ij} - \pi_{ip}^{1e} & \text{otherwise} \end{cases} \quad \forall (i, j) \in A^P. \tag{4}$$

For the delivery part of the problem, a linear relaxation of a knapsack problem needs to be solved with the extra feature that the delivery quantity for a customer consists of goods to satisfy the demand of the customer itself and of its neighbors. Therefore, introduce two sets of variables. First, associate with each customer $i \in N$ and period $s \in P_{is}^+$ the variable $\xi_i^s \in [0, u_{ip}^s]$ specifying the quantity delivered to customer i that is dedicated to satisfy the demand of customer i in period s if $s \in P$ or to the end inventory if $s = \rho + 1$. Second, associate with each customer $i \in N$, each of its neighbors $j \in \mathcal{N}_i$ and each period $s \in P_{ijp}^+$ variable $\psi_{ij}^s \in [0, u_{ijp}^s]$ specifying the quantity delivered to customer i dedicated to satisfy the demand of customer j in period s. As indicated before, $\rho + 1 \notin P_{ijp}^+$ for $j \in \mathcal{N}_i$. Given a route $r \in R$, variables ξ_i^s, $s \in P_{is}^+$, and ψ_{ij}^s, $s \in P_{ijp}^+$, must be 0 for customers $i \in N \setminus N_r$. Moreover, it must hold that $\sum_{i\in N_r} \left(\sum_{s\in P_{ip}^+} \xi_i^s + \sum_{j\in \mathcal{N}_i} \sum_{s\in P_{ijp}^+} \psi_{ij}^s \right) \leq Q$ to respect vehicle capacity. Given the conditions above, $(\xi_i^s)_{i\in N, s\in P_{ip}^+}$ and $(\psi_{ij}^s)_{i\in N, j\in \mathcal{N}_i, s\in P_{ijp}^+}$ define an RDP w associated with route r. The reduced cost can be rewritten as follows:

$$\bar{c}_{rw}^p = \sum_{(i,j)\in A_r} \bar{c}_{ij} + \sum_{i\in N_r} \sum_{s\in P_{ip}^+} \xi_i^s \left(\pi_p^{1b} - \pi_{is}^{1c} + \sum_{p\leq t<s} \left(h_i - \pi_{it}^{1d} - \pi_{it}^{1h} \right) \right)$$

$$+ \sum_{i\in N_r} \sum_{j\in \mathcal{N}_i} \sum_{s\in P_{ijp}^+} \psi_{ij}^s \left(m_{ij} + \pi_p^{1b} - \pi_{js}^{1c} - \pi_{is}^{1d} - \pi_{js}^{1i} + \sum_{p\leq t<s} \left(h_i - \pi_{it}^{1d} - \pi_{it}^{1h} \right) \right) \tag{5}$$

An extreme RDP has at most one partial subdelivery; hence, in an extreme RDP at most one variable ξ_i^s or ψ_{ij}^s can take a value in the open interval $]0u_{ip}^s[$ and $]0u_{ijp}^s[$, respectively.

4.2.1 Labeling Algorithm

Labeling algorithms have been proposed in the literature to solve the PPs of a wide variety of routing problems [12]. To solve the PP of the IRPDM, we propose a labeling algorithm in which a label represents both a partial route (path) and an associated extreme RDP. The labeling algorithm starts with a label at the source node v^S in the graph G^P, this label is extended to subsequent nodes if such extensions are feasible. An extension to the sink node v^E results in a route with corresponding extreme RDP. During the algorithm, a dominance rule can be used to discard labels that will not result in the optimal solution of the PP.

An extreme RDP consists of full subdeliveries, zero subdeliveries and at most one partial subdelivery. During the execution of the labeling algorithm, the quantity delivered in the partial subdelivery is unknown, because this quantity can depend on the other deliveries made. When a label is extended to the sink node v^E, the size of the partial subdelivery is determined. Following [1], we keep track of the possible contribution of the partial subdelivery to the reduced costs.

A label L_i corresponding to a partial route ending in node i with associated RDP w contains the following elements:

- T_i^{cost} : Reduced cost of the route/RDP (r, w), excluding the dual contribution of the partial subdelivery if $i \neq v^E$.
- T_i^{loadF} : Total quantity delivered along (r, w), the quantity of full subdeliveries only if $i \neq v^E$.
- $T_i^{\text{cust}_k}$: Binary value indicating whether or not customer $k \in N$ has been visited in the route.
- T_i^{part} : Binary value indicating whether or not RDP w contains a partial subdelivery.
- T_i^{rateP} : Unit rate of contribution of the partial subdelivery to the reduced costs, if applicable.
- T_i^{maxP} : Maximum quantity that can be delivered in the partial subdelivery, if applicable.

Therefore, the label is denoted by $L_i = \left(T_i^{\text{cost}}, T_i^{\text{loadF}}, \left(T_i^{\text{cust}_k} \right)_{k \in N}, T_i^{\text{part}}, T_i^{\text{rateP}}, T_i^{\text{maxP}} \right)$. There are three subdelivery types: a full (F), partial (P), and zero (Z) subdelivery. An extreme RDP consists of the subdelivery types for the visited customers and their neighbors for the periods in P_{ip}^+ and P_{ijp}^+, respectively, which we call a customer delivery pattern (CDP). For example, for a visit to customer i with one neighbor j, the CDP FF–P means that full subdeliveries are made for the two periods in P_{ip}^+ for customer i and that a partial subdelivery is made to satisfy a demand move from customer j in the single period in P_{ijp}^+. A CDP can contain at most one partial subdelivery, since an RDP can contain at most one, and the full subdeliveries cannot exceed vehicle capacity Q. For each customer $i \in N$ and period $p \in P$, we determine a list of feasible CDPs Γ_{ip} that we consider in the labeling algorithm at a node corresponding to customer i in period p. To make this list as short as possible, which will speed up the labeling algorithm, the list can be filtered to exclude CDPs that do not comply with the FIFO rule. For example, for customer i without neighbors, a CDP FPF cannot be optimal, and hence, this CDP is excluded from the list. Note that the FIFO rule can only be applied to the part of the CDP that indicate the subdeliveries for the visited customer itself and cannot be applied to the part of the CDP indicating the deliveries for the neighboring customers. For example, a CDP FFF-FPF is feasible with respect to the FIFO rule and can be in the optimal solution if the neighboring customer receives a delivery in the second period of P_{ijp}^+.

To express the resource extension functions, define binary parameters f_γ^s (respectively, $f_{\gamma j}^s$) which is equal to 1 if CDP $\gamma \in \Gamma_{ip}$ contains a full subdelivery for period

s for the visited customer (respectively, neighbor j). Similarly, define g_γ^s (respectively, $g_{\gamma j}^s$) which is equal to 1 for a partial subdelivery in period s for the visited customer (respectively, neighbor j). Now, we can define for each CDP $\gamma \in \Gamma_{ip}$ the cost $\tau_\gamma^{\text{cost}}$, the load of the full deliveries $\tau_\gamma^{\text{loadF}}$, an indicator whether there is a partial subdelivery in the CDP $\tau_\gamma^{\text{part}}$, the rate of the partial delivery $\tau_\gamma^{\text{rateP}}$, and the maximum size of the partial delivery $\tau_\gamma^{\text{maxP}}$, which are defined as follows:

$$\tau_\gamma^{\text{cost}} = \sum_{s \in P_{ip}^+} f_\gamma^s u_{ip}^s \left(\pi_p^{\text{1b}} - \pi_{is}^{\text{1c}} + \sum_{p \leq t < s} \left(h_i - \pi_{it}^{\text{1d}} - \pi_{it}^{\text{1h}} \right) \right) +$$

$$\sum_{j \in \mathcal{N}_i} \sum_{s \in P_{ijp}^+} f_{\gamma j}^s u_{ijp}^s \left(m_{ij} + \pi_p^{\text{1b}} - \pi_{js}^{\text{1c}} - \pi_{is}^{\text{1d}} - \pi_{js}^{\text{1i}} + \sum_{p \leq t < s} \left(h_i - \pi_{it}^{\text{1d}} - \pi_{it}^{\text{1h}} \right) \right)$$

$$\tau_\gamma^{\text{loadF}} = \sum_{s \in P_{ip}^+} f_\gamma^s u_{ip}^s + \sum_{j \in \mathcal{N}_i} \sum_{s \in P_{ijp}^+} f_{\gamma j}^s u_{ijp}^s$$

$$\tau_\gamma^{\text{part}} = \sum_{s \in P_{ip}^+} g_\gamma^s + \sum_{j \in \mathcal{N}_i} \sum_{s \in P_{ijp}^+} g_{\gamma j}^s$$

$$\tau_\gamma^{\text{rateP}} = \sum_{s \in P_{ip}^+} g_\gamma^s \left(\pi_p^{\text{1b}} - \pi_{is}^{\text{1c}} + \sum_{p \leq t < s} \left(h_i - \pi_{it}^{\text{1d}} - \pi_{it}^{\text{1h}} \right) \right) +$$

$$\sum_{j \in \mathcal{N}_i} \sum_{s \in P_{ijp}^+} g_{\gamma j}^s \left(m_{ij} + \pi_p^{\text{1b}} - \pi_{js}^{\text{1c}} - \pi_{is}^{\text{1d}} - \pi_{js}^{\text{1i}} + \sum_{p \leq t < s} \left(h_i - \pi_{it}^{\text{1d}} - \pi_{it}^{\text{1h}} \right) \right)$$

$$\tau_\gamma^{\text{maxP}} = \sum_{s \in P_{ip}^+} g_\gamma^s (u_{ip}^s - 1) + \sum_{j \in \mathcal{N}_i} \sum_{s \in P_{ijp}^+} g_{\gamma j}^s (u_{ijp}^s - 1)$$

Any CDP with a partial subdelivery for which $\tau_\gamma^{\text{rateP}} \geq 0$ can be discarded, since replacing the partial subdelivery with a zero subdelivery provides a solution with at most the same reduced cost.

The resource extension functions are defined as follows. Assume we have a label $L_i = \left(T_i^{\text{cost}}, T_i^{\text{loadF}}, \left(T_i^{\text{cust}_k} \right)_{k \in N}, T_i^{\text{part}}, T_i^{\text{rateP}}, T_i^{\text{maxP}} \right)$ corresponding to a node $i \neq v^E$ and the label is extended along an arc $(i, j) \in A^p$ $(j \neq v^E)$, for every CDP in Γ_{jp}. Let $\gamma \in \Gamma_{jp}$ be one of those CDPs. The extended label is given by $L_j = \left(T_j^{\text{cost}}, T_j^{\text{loadF}}, \left(T_j^{\text{cust}_k} \right)_{k \in N}, T_j^{\text{part}}, T_j^{\text{rateP}}, T_j^{\text{maxP}} \right)$ with:

$$T_j^{\text{cost}} = T_i^{\text{cost}} + \bar{c}_{ij} + \tau_\gamma^{\text{cost}} \tag{6}$$

$$T_j^{\text{loadF}} = T_i^{\text{loadF}} + \tau_j^{\text{loadF}} \tag{7}$$

$$T_j^{\text{cust}_k} = \begin{cases} T_i^{\text{cust}_k} + 1 & \text{if } j = k \\ T_i^{\text{cust}_k} & \text{otherwise,} \end{cases} \quad \forall k \in N \tag{8}$$

$$T_j^{\text{part}} = T_i^{\text{part}} + \tau_\gamma^{\text{part}} \tag{9}$$

$$T_j^{\text{rateP}} = T_i^{\text{rateP}} + \tau_\gamma^{\text{rateP}} \tag{10}$$

$$T_j^{\text{maxP}} = \begin{cases} \min\{\tau_\gamma^{\text{maxP}}, Q - T_i^{\text{loadF}} - \tau_\gamma^{\text{loadF}}\} & \text{if } \tau_\gamma^{\text{part}} = 1 \\ \min\{T_i^{\text{maxP}}, Q - T_i^{\text{loadF}} - \tau_\gamma^{\text{loadF}}\} & \text{otherwise.} \end{cases} \tag{11}$$

The resulting label is feasible if $T_j^{\text{loadF}} \leq Q$, $T_j^{\text{cust}_k} \leq 1$ for all $k \in N$, and $T_j^{\text{part}} \leq 1$. When extending to the sink node $j = v^E$, the cost computation differs to account for the partial subdelivery:

$$T_j^{\text{cost}} = \begin{cases} T_i^{\text{cost}} + \bar{c}_{ij} + T_i^{\text{maxP}} T_i^{\text{rateP}} & \text{if } T_i^{\text{rateP}} < 0 \\ T_i^{\text{cost}} + \bar{c}_{ij} & \text{otherwise.} \end{cases} \tag{12}$$

The number of labels can become very large; therefore, a dominance rule is used to reduce the number of labels. The dominance rule introduced for the IRP by Desaulniers et al. [1] still holds for the IRPDM:

Definition 1 A label $L_1 = \left(T_1^{\text{cost}}, T_1^{\text{loadF}}, \left(T_1^{\text{cust}_k} \right)_{k \in N}, T_1^{\text{part}}, T_1^{\text{rateP}}, T_1^{\text{maxP}} \right)$ is said to dominate a label $L_2 = \left(T_2^{\text{cost}}, T_2^{\text{loadF}}, \left(T_2^{\text{cust}_k} \right)_{k \in N}, T_2^{\text{part}}, T_2^{\text{rateP}}, T_2^{\text{maxP}} \right)$ if both labels L_1 and L_2 are associated with the same vertex and the following conditions are satisfied:

(a) $T_1^{\text{loadF}} \leq T_2^{\text{loadF}}$;
(b) $T_1^{\text{cust}_k} \leq T_2^{\text{cust}_k}$;
(c) $T_1^{\text{part}} \leq T_2^{\text{part}}$;
(d) $T_1^{\text{cost}} - T_1^{\text{maxP}} T_1^{\text{rateP}} \leq T_2^{\text{cost}} - T_2^{\text{maxP}} T_2^{\text{rateP}}$;
(e) $T_1^{\text{cost}} - \left(T_2^{\text{loadF}} - T_1^{\text{loadF}} \right) T_1^{\text{rateP}} \leq T_2^{\text{cost}}$;
(f) $T_1^{\text{cost}} - \left(T_2^{\text{loadF}} + T_2^{\text{maxP}} - T_1^{\text{loadF}} \right) T_1^{\text{rateP}} \leq T_2^{\text{cost}} - T_2^{\text{maxP}} T_2^{\text{rateP}}$.

4.2.2 Heuristic Labeling Algorithms

Before applying the exact labeling algorithm described above, two heuristic labeling algorithms are applied. First, for each route/RDP combination in the current RMP solution, optimize the CDPs for the given route with respect to the current dual variables. To optimize the CDPs, the labeling algorithm is solved with only the arcs in the given route. Second, the labeling algorithm is performed on a graph that contains only a subset $\hat{A}^p$ of the arcs A^p for each period $p \in P$. The arcs are selected by the procedure proposed in [13]. Arcs that do not belong to the κ least reduced cost out of an origin node, or into a destination node, are removed. To compute the reduced cost of an arc A, the average cost over all possible CDPs for the destination node is computed (or similarly, the average cost over the CDPs of the origin node of an arc) and added to the reduced cost. In this calculation, we assume that no quantity is delivered in the partial deliveries. Then, for each node, the κ arcs with the lowest reduced cost, both incoming and outgoing, are kept in the graph. We set a dynamic value for κ, which starts at 1 and is incremented by 2 if no columns are generated in every subproblem for $\kappa = 1$.

4.2.3 Acceleration Techniques

Next to the heuristic labeling procedures, the following acceleration techniques are implemented to speed up the column generation procedure.

First, the list of CDPs Γ_{ip} associated with a customer $i \in N$ in period $p \in P$ can be established once before the solution procedure starts. The costs and values associated with each CDP $\gamma \in \Gamma_{ip}$ need to be updated at each iteration with the corresponding dual variables. Before the (heuristic) labeling algorithm solves the PP, we filter the list of CDPs by applying the dominance conditions as in Definition 1, except for condition (b), in which all T values are replaced by the current τ values of the CDPs.

Second, ng-path relaxation is applied as defined in [14]. This relaxation of the PP allows for cycles in the paths. To apply ng-paths, define for each node $v \in V^p$ in network $G^p = (V^p, A^p)$ a subset of customers NG_v. Let NG_v contain v itself and a subset of vertices that are closest to v such that $|NG_v| = b$. Here, b is a predefined parameter (which is set to 7 in our experiments). An ng-path can contain a sequence of visits $v - v_1 - v_2 - \ldots - v$ only if at least one node $w \notin NG_v$ is visited in between two visits of v. The labeling algorithm is adjusted to accommodate ng-paths as explained in [15].

Finally, since constraints (1e), which limit the number of visits to a customer in one period to one, are numerous and often not binding in the optimal solution, these constraints are relaxed first and added only if violated in a BC form. Moreover, Desaulniers et al. [1] showed that for the IRP some holding capacity constraints (equivalent to 1d) are redundant with the constraints equivalent to 1c and 1e. However, for the IRPDM, it is not possible to establish a comparable statement. Hence, all capacity constraints (1d) are now present in the master problem. Yet, it is likely that for each customer this constraint is only binding in one or two periods. Therefore, we add the holding capacity constraints (1d) also in a dynamic way similar to constraints 1e.

4.3 Valid Inequalities

Next to the heuristic labeling described in Section 4.2.2 and the acceleration techniques described in Section 4.2.3, VIs are implemented to strengthen linear relaxations of the problem and hence, to speed up the solution method. Only one family of VIs that was proposed for the IRP can immediately be applied to the IRPDM. For the variant of the IRPDM in which initial inventory cannot be used to satisfy moved demand, existing VIs can be adjusted, although the adjustments are not trivial. For the variant of the IRPDM in which initial inventory can be used to satisfy moved demand, it is not clear whether or how some of the existing IRP VIs can be adjusted. Therefore, we restrict ourselves to developing VIs for the variant in which initial inventory cannot be used to satisfy moved demand.

First, in Section 4.3.1, we describe a family of VIs proposed in [1] for the IRP and we argue why this family of inequalities is also valid for the IRPDM. Second, we propose a generalization of the first family of inequalities in Section 4.3.2. Third, Sections 4.3.3 to 4.3.6 propose VIs for the IRPDM that are derived from VIs for the IRP. Finally, in Section 4.3.7, we elaborate why the VIs in Sections 4.3.3 to 4.3.6 need structural changes for the variant of the IRPDM in which initial inventory can be used to satisfy moved demand.

4.3.1 Valid Inequalities on the Minimum Number of Routes per Time Interval

In the IRP, given the total quantity that must be delivered and the vehicle capacity, one can compute the minimum number of vehicle routes needed to deliver all goods. So, if one adds up the residual demand of all customers up to period $\rho \in P$, a lower bound can be established on the number of routes to fulfill the total residual demand. This also holds for the number of routes needed up to a certain period $\ell \in P$. We denote these inequalities as Route Inequalities (RIs). In the IRPDM, both in case initial inventory can and cannot be used to satisfy moved demand, the total quantity that needs to be delivered by vehicles remains the same as in the IRP. Therefore, these inequalities can be applied to the IRP and both variants of the IRPDM without adjustments. A lower bound on the number of routes is given by $lb_\ell^R = \left\lceil \sum_{i \in N} \sum_{s=1}^{\ell} \bar{d}_i^s / Q \right\rceil$ and the following VIs hold:

$$\sum_{p=1}^{\ell} \sum_{r \in R} \sum_{w \in W_r^p} y_{rw}^p \geq lb_\ell^R, \quad \forall \ell \in P \tag{13}$$

Let π_ℓ^{13}, $\ell \in P$ be the dual variables associated with VIs (13). The reduced cost is adjusted the same way as in [1]:

$$\bar{c}_{ij} = \begin{cases} c_{ij} - \pi_p^{1f} - \sum_{\ell=p}^{\rho} \pi_\ell^{13} & \text{if } i = v^S \\ c_{ij} - \pi_{ip}^{1e} & \text{otherwise,} \end{cases} \quad \forall (i, j) \in A^P. \tag{14}$$

4.3.2 Generalized Valid Inequalities on the Minimum Number of Routes per Time Interval

The cuts in Section 4.3.1 can be generalized to time intervals $[\ell, \ell']$ where $\ell, \ell' \in P$ are such that $\ell' > \ell$. For example, suppose there is one customer and it has the following residual demands: 25, 40, 40, 40, 40, 40 for the six periods in the planning horizon, the vehicle capacity is $Q = 100$, and inventory capacity at the customer is $C = 80$. Then, at the end of period 4, there can be at most an inventory of 40; hence, at least one vehicle must visit this customer in the interval [5, 6]. All residual demands for this customer can be covered with 3 vehicles, but if, in a fractional solution, this customer receives a visit of one vehicle in periods 1, 3, and 4, and of 0.4 vehicle in period 5, the inequality for interval [5, 6] will be violated. We denote these inequalities as Generalized Route Inequalities (GRIs). If $\ell = 1$, the inequalities are the same as in Section 4.3.1.

Define a new lower bound $lb_{ll'}^{\bar{R}} = \left\lceil \frac{\sum_{i \in N} \left(\sum_{p=\ell}^{\ell'} d_i^p - \left(C_i - d_i^{\ell-1} \right) \right)}{Q} \right\rceil$. The numerator now accounts for the maximum possible inventory level at the end of period $\ell - 1$ at each customer. Note that the fraction can be rounded up because all terms are known at the beginning of the planning horizon. We propose the following generalized VIs:

$$\sum_{p=\ell}^{\ell'} \sum_{r \in R} \sum_{w \in W_r^p} y_{rw}^p \geq lb_{ll'}^{\bar{R}}, \quad \forall \ell, \ell' \in P \tag{15}$$

Associating dual variables $\pi_{\ell\ell'}^{15}$, $\ell, \ell' \in P$ with these inequalities, then the modified arc reduced costs $\bar{c}_{ij}$ become:

$$\bar{c}_{ij} = \begin{cases} c_{ij} - \pi_p^{1f} - \sum_{\ell=1}^{p} \sum_{\ell'=p}^{\rho} \pi_{\ell\ell'}^{15} & \text{if } i = v^S \\ c_{ij} - \pi_{ip}^{1e} & \text{otherwise,} \end{cases} \quad \forall (i,j) \in A^p. \quad (16)$$

4.3.3 Valid Inequalities on the Minimum Number of Visits per Customer

For the IRP, given the residual demand at a customer over periods 1 to $\ell \in P$, the inventory capacity at the customer and the vehicle capacity, one can compute how many visits are at least needed to satisfy a customer's demand [16, 17]. In the IRPDM, demand at a customer i cannot only be satisfied via deliveries by a vehicle but also via other customers $j : i \in \mathcal{N}_j$. Hence, a delivery to such a customer j should also be counted as a "visit" to customer i. Note that C_j can also decrease the number of visits needed for customer i, if residual demand is satisfied via a customer j. Therefore, define $C_i^{\max} = \max_{j:i \in \mathcal{N}_j \cup \{i\}}\{C_j\}$. Then the minimum number of visits needed to a customer between periods 1 and ℓ is given by $lb_{i\ell}^V = \left\lceil \frac{\sum_{p=1}^{\ell} \bar{d}_i^p}{\min\{Q, C_i^{\max}\}} \right\rceil$. The following VIs hold:

$$\sum_{p=1}^{\ell} \sum_{r \in R} \sum_{w \in W_r^p} \left(a_{ri} + \sum_{j:i \in \mathcal{N}_j} a_{rj} \right) y_{rw}^p \geq lb_{i\ell}^V \quad \forall i \in N, \ \forall \ell \in P \quad (17)$$

Associate dual variables $\pi_{i\ell}^{17}$, $i \in N$, $\ell \in P$ with the VIs. In the PP for period p, the arc reduced costs are adjusted as follows:

$$\bar{c}_{ij} = \begin{cases} c_{ij} - \pi_p^{1f} & \text{if } i = v^S \\ c_{ij} - \pi_{ip}^{1e} - \sum_{\ell=p}^{\rho} \pi_{i\ell}^{17} - \sum_{j \in \mathcal{N}_i} \sum_{\ell=p}^{\rho} \pi_{j\ell}^{17} & \text{otherwise} \end{cases} \quad \forall (i,j) \in A^p. \quad (18)$$

Lefever et al. [7] adjust a different version of these VIs for the IRPT in which the right-hand side contains the variable representing the inventory level. Lefever et al. [7] include transshipments by adding the fraction of transshipped demand to the left-hand side. Since variables are included in these terms, rounding is not possible.

4.3.4 Valid Inequalities on the Minimum Number of Subdeliveries per Demand

Inequalities on the minimum number of subdeliveries per demand (MNSDIs) for the IRP were proposed by Desaulniers et al. [1] based on the idea of Desaulniers [18] on similar inequalities for the Split Delivery Vehicle Routing Problem (SDVRP). The idea is that the residual demand $\bar{d}_i^s$ of customer $i \in N$ in period $s \in P$ can be fulfilled via one subdelivery of size $\bar{d}_i^s$, or via at least two smaller subdeliveries in different periods. We extend the inequalities to incorporate demand moves.

A given residual demand $\bar{d}_i^s > 0$ can, in the IRPDM, be fulfilled in four different ways: (i) either by performing one subdelivery to customer i in a period $p \in P_{is}^-$, (ii) at least two subdeliveries to customer i in different periods $p \in P_{is}^-$, (iii) one subdelivery to a customer $j : i \in \mathcal{N}_j$ in a period $p \in P_{jis}^-$ or (iv) at least two

subdeliveries to a customer j in different periods $p \in P^-_{jis}$. Define a^S_{ijw} (respectively, a^M_{ijw}) as a binary parameter equal to 1 if $a_{ri} = 1$ and $\bar{d}^s_j$ (respectively, less than $\bar{d}^s_j$) units are delivered in the subdelivery to customer i dedicated to customer $j \in \mathcal{N}_i \cup \{i\}$ and period $s \in P^+_{ijp}$ in RDP w, 0 otherwise. Define $a^S_{wi} = a^S_{wij}$ if $i = j$, and similarly for a^M_{wi}. The MNSDIs can be stated as follows:

$$\sum_{j: i \in \mathcal{N}_j \cup \{i\}} \sum_{p \in P^-_{jis}} \sum_{r \in R} \sum_{w \in W^p_r} \left(2a^S_{wji} + a^M_{wji} \right) y^p_{rw} \geq 2, \quad \forall i \in N, \, \forall s \in P : \bar{d}^s_i > 0 \quad (19)$$

Define π^{19}_{is}, $i \in N$, $s \in P$ as the dual variables of VIs (19). To take the dual variables into account in the PP for period $p \in P$, modify parameters τ^{cost}_γ as follows:

$$\tau^{cost}_\gamma = \sum_{s \in P^+_{ip}} \left[f^s_\gamma u^s_{ip} \left(\pi^{1b}_p - \pi^{1c}_{is} + \sum_{p \leq t < s} \left(h_i - \pi^{1d}_{it} - \pi^{1h}_{it} \right) \right) - (1 + f^s_\gamma) \pi^{19}_{is} \right]$$
$$+ \sum_{j \in \mathcal{N}_i} \sum_{s \in P^+_{ijp}} \left[f^s_{\gamma j} u^s_{ijp} \left(m_{ij} + \pi^{1b}_p - \pi^{1c}_{js} - \pi^{1d}_{is} - \pi^{1i}_{js} + \sum_{p \leq t < s} \left(h_i - \pi^{1d}_{it} - \pi^{1h}_{it} \right) \right) \right.$$
$$\left. - (1 + f^s_{\gamma j}) \pi^{19}_{js} \right] \quad (20)$$

4.3.5 Multiperiod Capacitated Subtour Inequalities

Avella et al. [19] formulate Multiperiod Capacitated Subtour Inequalities (MCSIs) for the IRP. The MCSIs exploit that over a given set of subsequent periods p_1 to p_2, one can determine the minimum vehicle flow needed to satisfy the demand of a subset of customers $S \subseteq N$. Before deriving the MCSIs for the IRPDM, we will rewrite the VIs of [19] for the IRP in the terminology of our paper.

Let $(E : F)$ denote the set of arcs $(i, j) \in A$ for which $i \in E$ and $j \in F$ with A the complete set of arcs. Suppose there is a subset of customers $S \subseteq N$ and a time interval $[p_1, p_2]$ in which $p_1 \leq p_2$, $p_1, p_2 \in P$. Define a_{rij} to be a binary parameter indicating whether arc (i, j) is traversed in route $r \in R$. The following inequalities hold for the IRP:

$$\sum_{t=p_1}^{p_2} \sum_{r \in R} \sum_{w \in W^t_r} \sum_{\substack{i,j: \\ i \in N \cup \{v^S\} \setminus S \\ \text{and } j \in S}} a_{rij} y^t_{rw} \geq \left\lceil \frac{\sum_{i \in S} \left(\sum_{t=p_1}^{p_2} d^t_i - \left(C_i - d^{p_1 - 1}_i \right) \right)}{Q} \right\rceil,$$

$$\forall S \subseteq N, \, \forall p_1, p_2 \in P \quad (21)$$

The left hand side computes the vehicle flow into $S \subseteq N$ during the periods p_1 to p_2. In the nominator of the right hand side, for each customer in S, we add up the demand over the periods in the time interval, minus the largest possible inventory at the end of period $p_1 - 1$. The largest possible ending inventory at a customer i is equal to the holding capacity C_i minus the demand in period $p_1 - 1$. Note that for $p_1 = 1$, the

right hand side can be improved to $\left\lceil \frac{\sum_{i \in S} \sum_{t=1}^{p_2} \bar{d}_i^t}{Q} \right\rceil$ since there is no delivery possible before this time period and the remaining (residual) demand is known.

Avella et al. [19] introduce a quadratic program to solve the separation problem for this family of inequalities, which is rewritten as follows for the IRP:

$$\min \sum_{t=p_1}^{p_2} \sum_{r \in R} \sum_{w \in W_r^t} \sum_{(i,j) \in A} a_{rij} \bar{y}_{rw}^t (1-\alpha_i)\alpha_j - \gamma \tag{22}$$

$$\text{s.t. } Q\gamma \leq \sum_{i \in N} \left(\sum_{t=p_1}^{p_2} d_i^t - \left(C_i - d_i^{p_1-1} \right) \right) \alpha_i + Q - \epsilon \tag{23}$$

$$\alpha_i \in \{0, 1\} \qquad\qquad\qquad \forall i \in N \tag{24}$$

$$\gamma \in \mathbb{Z} \tag{25}$$

in which $\bar{y}_{rw}^t$ are the values of the current fractional solution, $\alpha_i = 1$ if $i \in S$ and 0 otherwise for $i \in N$, and ϵ is a very small positive constant. γ represents the value of the right hand side of (21). Solutions with a negative objective correspond to violated cuts.

In the IRPDM, the demand of a customer $j \in S$ cannot only be satisfied by vehicles going into the set S (first term of (26)) but also by deliveries to a customer $i \in N \setminus S$ for which $j \in \mathcal{N}_i$ (second term of (26)). Note that if there are customers $j, k \in S$ and $k \in \mathcal{N}_j$, flow into j does not have to contribute twice to the flow into S to account for a possible demand move. We therefore adjust the MCSIs as follows for the IRPDM:

$$\sum_{t=p_1}^{p_2} \sum_{r \in R} \sum_{w \in W_r^t} \sum_{\substack{i,j: \\ i \in N \cup \{v^S\} \setminus S \\ \text{and } j \in S}} a_{rij} y_{rw}^t + \sum_{i \in N \setminus S} \sum_{j \in S \cap \mathcal{N}_i} \sum_{\substack{1 \leq t \leq p_2: \\ P_{ijt}^+ \cap [p_1, p_2] \neq \emptyset}} \sum_{r \in R} \sum_{w \in W_r^t} a_{ri} y_{rw}^t$$

$$\geq \left\lceil \frac{\sum_{i \in S} \left(\sum_{t=p_1}^{p_2} d_i^t - \left(C_i - d_i^{p_1-1} \right) \right)}{Q} \right\rceil, \quad \forall S \subseteq N, \forall p_1, p_2 \in P \tag{26}$$

The second term represents deliveries to customers $i \in N \setminus S$ for which $j \in S \cap \mathcal{N}_i$, but only for the periods $1 \leq t \leq p_2$ in which the subdelivery periods P_{ijt}^+ have any overlap with the interval $[p_1, p_2]$ under consideration. Again, for $p_1 = 1$, the right hand side can be improved to $\left\lceil \frac{\sum_{i \in S} \sum_{t=1}^{p_2} \bar{d}_i^t}{Q} \right\rceil$.

To separate the inequalities for the IRPDM, the programs (22)–(25) can still be used, but with the following extended objective function:

$$\min \sum_{t=p_1}^{p_2} \sum_{r \in R} \sum_{w \in W_r^t} \sum_{(i,j) \in A} a_{rij} \bar{y}_{rw}^t (1-\alpha_i)\alpha_j$$

$$+ \sum_{i \in N} \sum_{j \in \mathcal{N}_i} \sum_{\substack{1 \leq t \leq p_2: \\ P_{ijt}^+ \cap [p_1, p_2] \neq \emptyset}} \sum_{r \in R} \sum_{w \in W_r^t} a_{ri} \bar{y}_{rw}^t (1-\alpha_i)\alpha_j - \gamma \tag{27}$$

Associate dual variables $\pi_{S\ell\ell'}^{26}$, $S \subseteq N$, $\ell, \ell' \in P$ with inequalities (26). In the subproblem for period $p \in P$, the reduced costs are adjusted as follows:

$$\bar{c}_{ij} = \begin{cases} c_{ij} - \pi_p^{1f} - \sum_{S\subseteq N} \sum_{\ell=1}^{p} \sum_{\ell'=p}^{\rho} \bar{z}_{ij} \pi_{S\ell\ell'}^{26} & \text{if } i = v^S \\ c_{ij} - \pi_{ip}^{1e} - \sum_{S\subseteq N} \sum_{\ell=1}^{p} \sum_{\ell'=p}^{\rho} \bar{z}_{ij} \pi_{S\ell\ell'}^{26} & \forall (i,j) \in A^p. \\ \quad - \sum_{S\subseteq N} \sum_{k\in\mathcal{N}_i} \sum_{\ell=1}^{p} \sum_{\ell'=p}^{\rho} \bar{z}_{ik} \hat{z}_{p\ell\ell'} \pi_{S\ell\ell'}^{26} & \text{otherwise,} \end{cases}$$

$$(28)$$

with $\bar{z}_{ij}$ a parameter equal to 1 if customer $i \in N \cup \{v^S\} \setminus S$ and $j \in S$, and $\hat{z}_{p\ell\ell'}$ equal to 1 if $P_{ijp}^+ \cap [\ell, \ell'] \neq \emptyset$.
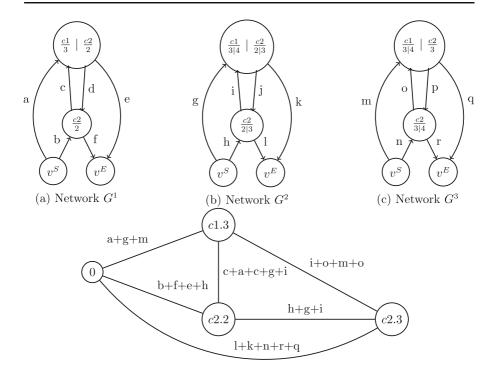
4.3.6 Capacity Inequalities

The Capacity Inequalities (CIs) were introduced for the Capacitated Vehicle Routing Problem (CVRP) [20]. For the CVRP, given a subset of customers $U \subseteq N$ and a lower bound $\kappa(U)$ on the number of vehicles required to service these customers given the vehicle capacity, the total flow of vehicles incident to subset U must be at least $2\kappa(U)$. Desaulniers et al. [1] propose an adaptation of these VIs for the IRP. Instead of a subset of customers, the authors use subsets of positive residual demands to define the CIs. For the CVRP, a graph depicting the flow between customers is used for separating the VIs. For the IRP, an auxiliary graph is used which depicts the flow between consecutive residual demands assuming the FIFO principle.

For the IRPDM in which initial inventory cannot be used to satisfy moved demand, we extend the notion of the flow between residual demands. We will use a similar auxiliary graph as in [1]; however, the underlying structure of the graph per period changes. Define the set of residual demands $RD = \{(i, s) \in N \times P \mid \bar{d}_i^s > 0\}$ and the auxiliary graph $G^* = (V^*, E^*)$. Node set V^* contains a depot node 0 and a node for each residual demand in RD. The edge set E^* contains the following types of edges. First, an edge is present between the depot node and each residual demand node. Secondly, edges are present between consecutive nodes that correspond to the same customer, i.e., an edge exists between (i, s) and $(i, s + 1)$. Third, there is an edge between nodes (i, s) and (i', s'), $i \neq i'$, if there exists a period $p \in P$ such that s is the latest period in $P_{ip}^+ \cap P$ and s' is the earliest period in $P_{i'p}^+ \cap P$. Until now, this definition is the same as in [1]. Additionally, for the IRPDM, an edge exists between (i, s) and (i', s'), $i \neq i'$, if i is in $\mathcal{N}_k$ for some $k \neq i, i'$ and there is a period $p \in P$ such that s is the latest period in P_{kip}^+ and s' is the earliest period in $P_{i'p} \cap P$. Edges that do not have any weight can be discarded.

For the weights on the edges, for a given (fractional) solution, we look into a network per period $G^p = (V^p, A^p)$. For the IRP, a node in this network for a given period p represents a customer and the periods for which a subdelivery can be made P_{ip}^+. For the IRPDM, a node represents both a customer and its neighbors, and the periods for which a subdelivery can be made for these customers in period p, i.e., the periods in P_{ip}^+ and P_{ijp}^+ for all $j \in \mathcal{N}_i$. To illustrate the structure of the auxiliary graph for the IRPDM, consider the example in Fig. 2.

Consider the example with $N = \{c1, c2\}$ and $P = \{1, 2, 3\}$. Customer $c1$ has a positive residual demand in period 3 and customer $c2$ has a positive residual demand

(a) Network G^1 (b) Network G^2 (c) Network G^3

(d) Auxiliary graph G^*

Fig. 2 Example for CIs

in periods 2 and 3. Moreover, customer $c2$ is a neighbor of customer $c1$, i.e., $\mathcal{N}_{c1} = \{c2\}$. The nodes in the networks in Fig. 2a–c represent both the customer and their neighbor, if applicable. Period $4 = \rho + 1$ is the fictitious period and recall that this period is never included in the delivery periods for a neighbor. Figure 2d gives the corresponding auxiliary graph G^* in which customer i and period s are indicated by $ci.s$.

To illustrate the association of the arcs with the edges, consider as an example the edge between $c1.3$ and $c2.2$. This edge is present because the latest period in $P^+_{c2,1}$ is 2 and the first period in $P^+_{c1,1}$ is 3. The edge represents the flow between the residual demands and can be computed by summing the flow on arc $(c2, c1) \in A^1$ (arc c), the incoming arcs of node $c1 \in V^1$ (arcs a and c) and the incoming arcs of node $c1 \in V^2$ (arcs g and i). The last two sets are added since customer $c2$ is a neighbor of customer $c1$, the last period of $P^+_{c1,c1,1} \cap P$ and $P^+_{c1,c1,2} \cap P$ is period 3, and the first period of $P^+_{c1,c2,1}$ and $P^+_{c1,c2,2}$ is period 2. Note that arc c is added twice to this edge flow; counting arcs twice for one edge was not possible in the auxiliary graph for the IRP but is now necessary to account for the demand moves.

Since we only change the underlying auxiliary graph for the IRPDM, the VIs as defined in [1] and the impact on the reduced cost remain the same and are not repeated here for conciseness. They use three separation heuristics for the CIs for

the IRP. The first one is the separation routine of the CVRPSEP package [21] developed by Lysgaard et al. [22] which is followed by a filter to incorporate that, for the IRP, the flow through a node can be higher than one. Second, current routes/RDPs with exactly one partial subdelivery in a current solution are considered to construct subsets $U \subseteq N$ on which the VI is evaluated. Finally, a route-based connected component heuristic is applied which was proposed by Archetti et al. [23] for the SDVRP with Time Windows. For details on the separation heuristics, which are also used for the IRPDM, we refer to [1].

4.3.7 IRPDM in Which Initial Inventory Can Satisfy Moved Demand

The inequalities presented in Sections 4.3.3 to 4.3.6 cannot be adjusted without changing their structure and effectiveness for the variant of the IRPDM in which initial inventory at a customer can be used to satisfy the demand of another customer via a demand move. For the inequalities in Sections 4.3.3 to 4.3.5, two main reasons preventing effective adjustments are (1) the "flow" resulting from the use of initial inventory should be accounted for in the left hand side and (2) residual demand can no longer be used in the right hand side of the inequalities. For the CIs in Section 4.3.6, similar to the other inequalities, the residual demands can no longer be used to construct the auxiliary graph. Hereafter, we discuss the two main reasons in more detail by reflecting on the VIs on the minimum number of visits per customer (Section 4.3.3).

(1) If it is possible to satisfy a demand move from initial inventory, potentially all demand at a customer j is satisfied from the initial inventory of customers j and i : $j \in \mathcal{N}_i$. In that case, no visits by a vehicle (to j or i) are needed to satisfy the demand of j. The idea of considering the use of initial inventory of i to satisfy demand of j as a "visit" could be applied to adjust the VI in two ways. A first approach could be to add the initial inventory variables to the left hand side; however, it would not be counting visits, but units of goods. One could divide by the demand and round up, but this would be non-linear. A second approach could be to add supplementary binary variables, which are equal to 1 if initial inventory is used to satisfy a demand move. However, these binary variables must be added to the master problem and moreover, a set of big-M constraints is needed to make sure the binary variables have the correct value.

(2) In the right hand side of the inequality, residual demands $\bar{d}_j$ can no longer be used since initial inventory of customer j can be used to satisfy demand of a customer $k \in \mathcal{N}_j$. Hence, not all initial inventory is necessarily available to satisfy demand of customer j itself. In the right hand side of inequality (17), we could therefore incorporate the variables that represent the use of initial inventory to satisfy moved demand. The disadvantage of that is that decision variables are included in the fraction and rounding the fraction would make the inequality non-linear.

Concluding, both on the left and the right hand side of the VI, the structure has to be changed to handle the possibility of using initial inventory to satisfy moved demand, making the inequalities weaker. A similar reasoning can be followed for the VIs in Sections 4.3.4 and 4.3.5. Therefore, the inequalities cannot be adjusted for this variant of the IRPDM without changing their structure and effectiveness.

4.4 Branching

To find a feasible solution to the problem, seven types of branching decisions are evaluated if a fractional solution of the linear relaxation is computed. The branching decisions are defined on the following variables:

1. The total number of routes over all periods $\left(\sum_{p \in P} \sum_{r \in R} \sum_{w \in W_r^p} y_{rw}^p \right)$.
2. The number of routes per period $p \in P$ $\left(\sum_{r \in R} \sum_{w \in W_r^p} y_{rw}^p \right)$.
3. The number of visits per customer $i \in N$ $\left(\sum_{p \in P} \sum_{r \in R} \sum_{w \in W_r^p} a_{ri} y_{rw}^p \right)$.
4. v_i^s variables.
5. z_{ij}^p variables.
6. The flow through each customer vertex $i \in N$ in each period $p \in P$ $\left(\sum_{r \in R} \sum_{w \in W_r^p} a_{ri} y_{rw}^p \right)$.
7. The flow on each edge $< i, j >$ in each period $p \in P$ which is equal to the sum of the flows on the corresponding arcs (i, j) and (j, i) in A^p $\left(\sum_{r \in R} \sum_{w \in W_r^p} (a_{rij} + a_{rji}) y_{rw}^p \right)$.

Compared with the solution method proposed by Desaulniers et al. [1] for the IRP, we added three types of variables to branch on 3, 4, and 5. Types 4, 5, and 7 are sufficient to guarantee an optimal integer solution. For a discussion on the arc and edge flows, we refer to [1]. Branching decisions are imposed in the model by adding a constraint, except for setting the flow on an edge to zero for which both corresponding arcs are removed from the arc set A^p. Adding an extra constraint to the master problem implies an adjustment in the reduced costs; specifics are omitted here for conciseness.

The next steps are followed to decide which branching decision is imposed. Compute the values of the variables for each type of decisions 1 to 7 and select for each type the candidate variable with a value closest to 0.5. If the candidates for types 6 and 7 have fractional values between 0.25 and 0.75, then branch on the variable with the value closest to 0.5 out of these two (at equality, select the type 3 decision). If there are no type 6 or 7 variables to branch on, if there is a candidate variable of type 1 or 2, select the candidate with the value closest to 0.5. If no candidate exists in the previous types, branch on the candidate variable of type 3 if one exists. Otherwise, choose the candidate variable of type 4 or 5 of which the value is closest to 0.5 to branch on.

A local-depth first search approach as described in [1] is applied to select the next node in the branch-and-bound tree to explore.

5 Computational Experiments

To assess the impact of including the demand moves in the IRP, we performed computational experiments using the described BPC algorithm that was implemented using C++ and the Gencol library. CPLEX 12.6 is used to solve all restricted master problems during the solution procedure. These experiments are run on an Intel Core

i7-4770 processor at 3.40 GHz, with 8 cores and 16 GB RAM. For all tests, only a single core is used and a time limit of 2 h is imposed for each instance. To evaluate the benefits of demand moves, the IRPDM results are compared to the IRP by using the solution values obtained by Coelho and Laporte [17] and Desaulniers et al. [1] (see instances and results in [24]).

To design our test set, we use the benchmark instances proposed by Archetti et al. [16] for the IRP. The time horizon in these instances is either 3 or 6 periods, instances have a multiple of five customers, and there is one vehicle with a given capacity. Moreover, an instance contains for each customer the location, the initial inventory level, the maximum inventory capacity, the demand, and the inventory holding rate. For the depot, the quantity becoming available is given instead of the demand and there is no maximum on the inventory. There are two levels for the inventory holding rate. Based on that, there are four classes of instances denoted by their inventory holding rate level (High or Low) and planning horizon (3 or 6 periods), resulting in classes H3, H6, L3, and L6. The instances originally include a single vehicle, but the instances have been used for the multi-vehicle case by dividing the vehicle capacity by the chosen number of vehicles. Details on the instances are available in [16].

To incorporate demand moves, we determine for each customer $i \in N$ the closest customer $j \in N \setminus \{i\}$ that is within 150 units of distance and set $\mathcal{N}_i = \{j\}$. All VIs are added in a dynamic way in each node of the branch-and-bound tree. The CIs and MCSIs are only added in nodes that are at most at depth two in the tree. The costs for demand moves are set to $m_{ij} = 0.01$ per unit of goods and per unit of distance between locations i and j (following [5]) and there is no limit on the amount of demand that can be moved, unless indicated otherwise.

In Section 5.1, we present results to assess the effectiveness of the new GRIs (Section 4.3.2), the MCSIs (Section 4.3.5), and the CIs (Section 4.3.6). Thereafter, generating results for the IRPDM with the most efficient settings, Section 5.2 compares the solution values of the IRPDM with the solution values of the IRP. In Section 5.2.1, the cost of a demand move is set to $m_{ij} = 0.05$ and $m_{ij} = 0.1$ to assess the impact of changing this cost. In practice, it might not be desirable that all demand of a customer is satisfied by another customer as described in Section 3.1. Hence, Section 5.2.2 reports the effect of limiting the percentage of demand that can be moved to 25%, 50%, and 75% of the demand of one customer in each period.

5.1 Effectiveness of Valid Inequalities

We assess the effectiveness of the GRIs, MCSIs, and CIs by solving the IRPDM with different combinations of VIs. The first setting includes the CIs and the GRIs, while the second setting includes the MCSIs and the GRIs. For the third and fourth settings, both the CIs and MCSIs are included. Additionally, setting three includes the GRIs, while setting four uses the original RIs. The remaining VIs are used in all settings. For each setting, the IRPDM is solved for a subset of the instances. The algorithm is tested on the instances with 3 and 4 vehicles ("K"), with 5 and 10 customers for 3-period horizon, and with 5 customers for 6-period horizon, for both high and low inventory holding costs. Hence, for each H3/L3 class ("Class"), there are 10 instances, and for each H6/L6 class, there are 5 instances in this subset.

Table 1 reports for each class and number of vehicles the average integrality gap at the root node before adding VIs ("Gap$_r$"), which is the same for all settings. Thereafter, the table compares for each setting the number of instances solved to optimality ("Opt"), the average running time of instances solved to optimality ("T(s)"), the average integrality gap at the root node after adding VIs ("Gap"), and the average number of CIs ("CI") and MCSI ("MCSI") inequalities added during the execution of the algorithm, respectively. Only the instances solved to optimality are considered when computing the averages. The integrality gap is computed as $(\overline{z} - \underline{z})/\overline{z}$ with $\underline{z}$ the lower bound computed at the root node of the branch-and-bound tree and $\overline{z}$ the optimal value. The row "overall" shows the overall average, overall minimum, or maximum reported in each of columns.

Table 1 shows that with all four settings, the same number of instances can be solved. Moreover, the CIs and MCSIs are effective since the integrality gap is approximately halved by adding these VIs. This can be observed from the overall integrality gaps of 2.1% and 1.9% for settings 1 and 2 respectively, compared to the gap of 4.2% in the root node before adding VIs. In tests with the RIs (without the CIs and MCSIs), still 55 instances can be solved to optimality, but the average running time is more than 2.5 times as high and the integrality gap is more than twice as high as in setting 4. Details on these tests are not reported, but are available on request.

The results indicate that the MCSIs are slightly more effective than the CIs since setting 2 gives both a lower average computation time and lower integrality gap after adding the VIs than setting 1. Combining these types of inequalities in settings 3 and 4 increases the efficiency since the running time goes down, even tough the integrality gap is the same as for settings 2 and 3. In settings 3 and 4, the average number of identified CIs is much lower than in setting 1, but since the MCSIs are slightly more effective, the average computation time is still lower in settings 3 and 4 than in setting 1. Comparing settings 3 and 4 shows that the integrality gaps are the same, which implies that using the GRIs does not seem to improve the solution method for the IRPDM. Note that the average number of CIs and MCSIs differs slightly between settings 3 and 4. Although the generalization of the route inequalities is not effective for the IRPDM, this cannot immediately be concluded for other problems. Based on these observations, setting 4 will be used for the remainder of the experiments.

5.2 Comparing IRP and IRPDM

To evaluate the benefit of exploiting demand moves in the IRP, the solutions of the IRPDM are compared to the solutions of the IRP. We look into the (percentage) cost improvement that is achieved for the test instances, and examine the number of moves and their size that actually take place in the IRPDM solutions. The solutions for the IRP are collected from [24].

Table 2 shows the obtained results. For each class of instances ("Class"), fleet size ("K"), and number of customers ("N"), Table 2 first shows the number of instances that are solved to optimality for both the IRP and IRPDM ("Opt."). Secondly, the average computation time for the BC IRP algorithm by Coelho and Laporte [17]

Table 1 Effectiveness of GRIs, MCSIs, and CIs

Class	K	Gap_r	1. CIs - GRIs				2. MCSIs - GRIs				3. CIs and MCSIs - GRIs					4. CIs and MCSIs - RIs				
			Opt.	$T(s)$	Gap	CI	Opt.	$T(s)$	Gap	MCSI	Opt.	$T(s)$	Gap	CI	MCSI	Opt.	$T(s)$	Gap	CI	MCSI
H3	3	2.9	10	19	1.3	6.7	10	12	1.0	4.4	10	14	1.0	1.5	4.2	10	14	1.0	1.5	4.2
H3	4	3.9	10	16	1.6	8.0	10	12	1.4	4.9	10	11	1.4	1.3	4.9	10	11	1.4	1.3	4.9
H6	3	2.7	4	1006	2.5	1.3	4	669	2.3	4.8	4	711	2.3	0.3	4.3	4	698	2.3	0.5	4.3
H6	4	2.4	5	1821	2.0	6.8	5	1475	1.7	3.0	5	1348	1.7	2.6	3.6	5	1333	1.7	2.6	3.6
L3	3	5.2	10	20	2.3	5.5	10	20	1.8	4.6	10	19	1.8	1.8	4.5	10	19	1.8	1.8	4.5
L3	4	6.7	10	19	2.6	7.7	10	16	2.4	4.9	10	15	2.4	1.8	4.8	10	15	2.4	1.8	4.8
L6	3	4.1	4	2414	3.7	1.8	4	1894	3.5	4.0	4	1885	3.5	0.3	4.0	4	1834	3.5	0.3	4.0
L6	4	3.4	4	1238	2.4	7.8	4	1299	2.4	2.5	4	1174	2.4	5.8	3.0	4	1168	2.4	4.0	3.3
Overall		4.2	57	500	2.1	6.2	57	411	1.9	4.4	57	393	1.9	1.8	4.3	57	387	1.9	1.7	4.4

Table 2 Comparison of solution values IRP and IRPDM

Class	K	N	IRP			IRPDM					
			Opt.	T(s)-BC	T(s)-BPC	T(s)	Av. Impr. (%)	Max. Impr. (%)	Min. Impr. (%)	Av. Nr. of DMs	Av. Sz. of DM
H3	3	5	5/5	0.6	0.0	0.2	3.35	4.10	1.49	2.2	39.9
H3	3	10	5/5	8.2	4.9	26.9	1.43	3.56	0.16	2.6	37.4
H3	3	15	3/5	22.0	8.2	279.9	0.58	0.93	0.26	2.0	12.3
H3	3	20	1/5	26.0	1.1	2402.9	0.08	0.08	0.08	2.0	16.5
H3	4	5	5/5	0.8	0.0	3.8	3.43	5.55	0.86	2.8	28.8
H3	4	10	5/5	110.4	4.3	18.4	1.22	2.99	0.00	3.2	39.5
H3	4	15	1/5	84.0	33.5	7155.5	0.78	0.78	0.78	3.0	26.0
H3	5	5	5/5	0.6	0.0	0.4	5.00	6.07	3.75	2.0	30.0
H3	5	10	5/5	264.8	1.5	778.8	2.22	4.01	1.49	3.6	33.9
H3	5	15	1/5	2252.0	7.6	5860.0	0.55	0.55	0.55	2.0	36.0
H3	5	20	1/5	2918.0	3.4	4858.2	0.92	0.92	0.92	2.0	15.0
Overall H3			*37/60*	*196.6*	*3.3*	*682.7*	*2.36*	*6.07*	*0.00*	*2.6*	*31.6*
H6	3	5	4/5	34.8	490.7	697.6	1.83	3.04	0.00	4.8	54.2
H6	4	5	5/5	37.6	110.2	1332.7	2.47	4.29	0.46	5.6	41.7
H6	5	5	3/5	110.0	102.6	1188.9	2.99	4.17	1.04	5.3	37.1
Overall H6			*12/15*	*54.8*	*235.1*	*1085.0*	*2.38*	*4.29*	*0.00*	*5.3*	*43.8*
L3	3	5	5/5	0.6	0.0	0.3	5.25	6.98	2.12	2.2	39.9
L3	3	10	5/5	8.4	4.9	37.8	2.66	6.47	0.24	2.4	42.0
L3	3	15	3/5	18.3	9.8	379.3	1.24	1.97	0.57	1.7	13.2

Table 2 (continued)

| | | | IRP | | IRPDM | | | | | |
Class	K	N	Opt.	T(s)-BC	T(s)-BPC	T(s)	Av. Impr. (%)	Max. Impr. (%)	Min. Impr. (%)	Av. Nr. of DMs	Av. Sz. of DM
L3	3	20	1/5	11.0	0.9	737.5	0.50	0.50	0.50	2.0	16.5
L3	4	5	5/5	0.8	0.0	4.1	5.09	7.22	1.18	3.2	26.4
L3	4	10	5/5	111.2	6.2	26.7	2.13	5.25	0.00	3.8	34.8
L3	4	15	1/5	91.0	45.8	3454.7	1.80	1.80	1.80	7.0	14.3
L3	5	5	5/5	0.8	0.0	0.5	7.0	10.02	5.50	2.0	29.6
L3	5	10	5/5	262.2	3.9	713.8	3.80	6.87	2.33	3.4	35.0
L3	5	15	1/5	2037.0	14.1	1821.5	1.76	1.76	1.76	2.0	36.0
L3	5	20	1/5	3975.0	3.0	5899.1	2.14	2.14	2.14	3.0	30.7
Overall L3			*37/60*	*218.6*	*4.5*	*458.6*	*3.80*	*10.02*	*0.00*	*2.8*	*31.7*
L6	3	5	4/5	46.3	1169.5	1833.9	2.96	5.06	0.14	5.5	39.2
L6	4	5	4/5	127.3	549.6	1168.3	4.11	6.37	0.89	6.3	34.9
L6	5	5	2/5	110.5	0.6	216.2	3.55	5.26	1.83	4.0	41.0
Overall L6			*10/15*	*91.5*	*687.8*	*1244.1*	*3.54*	*6.37*	*0.14*	*5.5*	*37.8*

("T(s)-BC"), for the BPC IRP algorithm by Desaulniers et al. [1] ("T(s)-BPC"), and for the IRPDM ("T(s)") are reported. Thereafter, the average ("Av. Impr"), maximum ("Max. Impr"), and minimum ("Min. Impr") percentage cost improvement of the IRPDM over the IRP are stated. Finally, the average number of demand moves ("Av. Nr. of DMs") and the average size of a demand move are reported ("Av. Sz. of DM").

We run the algorithm on instances with up to 25 customers for a 3-period horizon and up to 10 customers for a 6-period horizon. All detailed results are reported in Appendix 1. Two instances can be solved for the IRPDM while no feasible solution for the IRP exists. Hence, these instances are not included in the results since there are no IRP results to compare with. For instances with a 6-period horizon, we can only solve instances with five customers to optimality. This is limited, however, note that for the IRP not all instances with a 6-period horizon and ten customers have been solved to optimality with BPC in state-of-the-art literature [1] and the proposed IRPDM is a more complicated problem.

The average computation times show that the IRPDM instances require more time to be solved than the IRP with the BPC solution method. This can be expected since the BPC solution method for the IRPDM is an extension of the one for the IRP. Compared to the BC method for the IRP, solving the IRP is in general easier. The higher computation times are caused by having a more extensive master problem which includes additional binary variables and new constraints. Also, the capacity constraints are not redundant with the other constraints in the master problem which was the case for the IRP [1]. Moreover, the number of CDPs per customer increases substantially since the patterns include the deliveries dedicated to the neighboring customers. Consequently, the number of labels is greater which slows down the PP. However, for some instances, the BPC method for the IRPDM solves them more quickly, for example class H3 with 4 vehicles and 10 customers.

The average cost improvement is around 2.35% for instance classes with high holding costs and above 3.5% for instance classes with low inventory holding costs, respectively. In general, it can be observed that the average improvements are higher for low inventory holding costs while the average number and the average size of the demand moves do not differ much. This can be explained by the fact that for high holding costs, the routing costs are a smaller part of the solution value than for low holding costs. Since the demand moves decrease the routing costs and increase the holding costs, the savings yielded by including demand moves are larger when holding costs are lower.

The maximum improvements go up to 10% and for only three classes of instances, the minimum cost improvement equals zero. A zero cost improvement means that solving the IRPDM results in the same solution as the IRP, i.e., exploiting demand moves does not result in a cost saving. Looking into the detailed results in Appendix 1 shows that out of 96 instances, only three instances do not result in a cost improvement. The number of demand moves per instance is 2.6 and 2.8 for short time horizons, and 5.3 and 5.5 for long horizons, respectively. This implies that there is on average one demand move per period of the planning horizon. The higher number of moves for longer horizons can be explained by the fact that in a longer planning

horizon, there are more opportunities to incorporate a demand move for multiple periods. Note that the percentage cost improvement is not higher for a longer planning horizon than for a shorter planning horizon. If a demand move takes place, the number of units moved is quite substantial with averages between 30 and 45 units, which is approximately between half and three-quarters of the average demand (the demand is between 10 and 100).

5.2.1 Impact of the Demand Move Costs

In the previous experiments, the service fee incurred for a demand move (per unit of goods and per unit of distance) is set to $m = 0.01$. This section examines the impact of this parameter on the IRPDM solutions by solving the IRPDM for different values of m. For a subset of instances (instance sizes 5, 10 and 15 for 3-period horizon and sizes 5 and 10 for 6-period horizon), the IRPDM is solved for $m = 0.005$, $m = 0.05$ and $m = 0.1$ as well. The latter two values were also tested by Coelho et al. [5] for the IRPT. Table 3 reports the average cost improvement over the IRP ("Av. (%)"), the maximum cost improvement ("Max. (%)"), the average number of demand moves ("Av. Nr."), and the average size of the demand move ("Av. Sz.") per instance class and fleet size. Only instances that are solved to optimality for all parameter values of m are considered (the number of instances solved is indicated in column "Opt."); therefore, averages can differ marginally from the reported results in Table 2 for $m = 0.01$. Detailed results can be found in Appendix 2.

The results in Table 3 show that the improvement over the IRP by using demand moves diminishes if the cost of demand move increases, which can be expected. Increasing the value of m from $m = 0.01$ to $m = 0.05$ results in average improvements that are approximately a factor five lower, as illustrated in Fig. 3. The average number of demand moves decreases more rapidly as planning horizons become longer. Increasing the value of m to 0.1 results in very few demand moves, and hence, a very minor cost improvement of only 0.1% and 0.2% on average for high and low inventory holding costs, respectively. Moreover, if a demand move takes place, the number of units moved is very limited and approximately half of the size for $m = 0.05$.

Lowering m from $m = 0.01$ to $m = 0.005$ leads to higher improvements, as can be expected. Note that mainly for 6-period horizon instances, the number of demand moves increases which results in an average cost improvement twice the improvement for $m = 0.01$. The average size of the demand moves does not change substantially for this change in demand move cost m.

Overall, it can be observed that the value of m has a larger impact for instances with a 6-period horizon. Figure 3 shows that when m is increased, the average cost improvement decreases faster for a longer than for a shorter planning horizon. Also the number of demand moves declines faster for a longer planning horizon, starting at averages of well above one move per period for $m = 0.005$, but reducing to almost zero for $m = 0.05$ and $m = 0.1$.

Table 3 Impact of move cost m

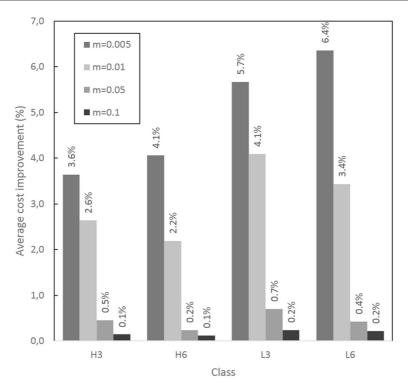| Class	K	Opt.	m = 0.005				m = 0.01				m = 0.05				m = 0.1			
			Av. (%)	Max. (%)	Av. Nr.	Av. Sz.	Av. (%)	Max. (%)	Av. Nr.	Av. Sz.	Av. (%)	Max. (%)	Av. Nr.	Av. Sz.	Av. (%)	Max. (%)	Av. Nr.	Av. Sz.
H3	3	12	3.0	6.2	2.5	32.2	2.1	4.1	2.3	34.4	0.4	1.7	0.3	6.0	0.2	0.7	0.3	3.7
H3	4	10	3.5	6.6	3.4	40.8	2.3	5.5	3.0	33.6	0.4	2.0	0.4	4.8	0.2	1.6	0.3	2.0
H3	5	10	4.5	7.9	3.1	33.4	3.6	6.1	2.8	31.9	0.6	1.6	0.8	12.2	0.0	0.4	0.2	6.0
Overall H3		*32*	*3.6*	*7.9*	*3.0*	*35.3*	*2.6*	*6.1*	*2.7*	*33.4*	*0.5*	*2.0*	*0.5*	*8.3*	*0.1*	*1.6*	*0.3*	*3.6*
H6	3	4	3.9	5.8	7.3	50.8	1.8	3.0	4.8	54.2	0.4	1.5	0.3	6.0	0.3	1.1	0.3	6.0
H6	4	4	3.3	6.2	5.8	39.1	2.2	4.3	5.0	44.2	0.2	0.3	0.5	16.0	0.0	0.0	0.3	2.0
H6	5	1	7.6	7.6	12.0	45.3	3.8	3.8	5.0	34.0	0.0	0.0	0.0	-	0.0	0.0	0.0	-
Overall H6		*9*	*4.1*	*7.6*	*7.1*	*45.0*	*2.2*	*4.3*	*4.9*	*46.7*	*0.2*	*1.5*	*0.3*	*12.7*	*0.1*	*1.1*	*0.2*	*4.0*
L3	3	12	5.0	10.5	2.3	35.8	3.5	7.0	2.2	36.6	0.7	3.1	0.3	6.0	0.3	1.4	0.3	3.7
L3	4	11	5.3	8.6	4.3	34.4	3.4	7.2	3.8	28.5	0.5	3.1	0.4	4.8	0.3	2.5	0.3	2.0
L3	5	10	6.8	12.7	3.1	33.6	5.5	10.0	2.7	32.3	1.0	2.5	1.0	11.7	0.1	0.7	0.1	6.0
Overall L3		*33*	*5.7*	*12.7*	*3.2*	*34.7*	*4.1*	*10.0*	*2.9*	*32.7*	*0.7*	*3.1*	*0.5*	*8.3*	*0.2*	*2.5*	*0.2*	*3.3*
L6	3	4	6.1	9.5	7.0	51.4	3.0	5.1	5.5	39.2	0.6	2.4	0.3	6.0	0.4	1.7	0.3	6.0
L6	4	3	5.3	9.0	6.3	34.0	3.5	6.4	6.0	34.6	0.4	0.7	0.7	16.0	0.0	0.0	0.0	-
L6	5	1	10.5	10.5	11.0	49.4	5.3	5.3	5.0	33.4	0.0	0.0	0.0	-	0.0	0.0	0.0	-
Overall L6		*8*	*6.4*	*10.5*	*7.3*	*44.6*	*3.4*	*6.4*	*5.6*	*36.7*	*0.4*	*2.4*	*0.4*	*12.7*	*0.2*	*1.7*	*0.1*	*6.0*

Fig. 3 Average cost improvement for *m*-values by class

5.2.2 Impact of Limit on Moved Demand

As discussed in Section 3.1, the IRPDM allows that all demand of one customer is moved to another customer (after using the initial inventory). This could imply that some customers are never replenished by a vehicle. From a service point of view, this might be unacceptable. In this section, we therefore analyze the impact on the solutions when the moved demand per customer per period is limited to a given percentage, as discussed in Section 3.1. We solve the IRPDM for a maximum of 25%, 50%, and 75%, additionally to the results already obtained for 100% (which allows moving all demand). The same instances are used as in Section 5.2.1 and Table 4 reports similar information as Table 3. Only instances solved to optimality for all settings are taken into account; therefore, the averages for 100% can deviate slightly from the reported results in Table 2. Appendix 2 reports the detailed results.

Table 4 shows that the average number of units moved decreases if the maximum demand moved becomes smaller. For example, for class H3, the average size is only 6 units if the limit is 25% compared to 32.3 units if there is no maximum. It is interesting to observe that the average number of units declines stronger for a shorter planning horizon than for a longer planning horizon. For instance, for class H6, the average number of units is 15.6 for a 25% limit, which is much larger than the 6

Table 4 Impact of maximum on moved demand

Class	K	Opt.	Max. 25%				Max. 50%				Max. 75%				Max. 100%			
			Av. (%)	Max. (%)	Av. Nr.	Av. Sz.	Av. (%)	Max. (%)	Av. Nr.	Av. Sz.	Av. (%)	Max. (%)	Av. Nr.	Av. Sz.	Av. (%)	Max. (%)	Av. Nr.	Av. Sz.
H3	3	13	0.3	2.6	0.3	4.3	0.5	2.6	0.5	10.6	0.8	3.2	1.0	23.5	2.0	4.1	2.3	32.6
H3	4	10	0.4	2.4	0.4	2.5	0.7	2.9	0.7	21.0	0.8	2.9	0.8	25.5	2.3	5.5	3.0	33.6
H3	5	9	1.0	3.2	1.1	9.4	1.2	3.8	1.4	12.4	2.1	4.7	1.7	22.1	3.8	6.1	2.9	30.6
Overall H3		*32*	*0.5*	*3.2*	*0.6*	*6.0*	*0.8*	*3.8*	*0.8*	*14.0*	*1.2*	*4.7*	*1.1*	*23.6*	*2.6*	*6.1*	*2.7*	*32.3*
H6	3	3	0.8	2.0	1.3	16.5	0.8	2.0	1.0	23.5	0.9	2.0	2.0	21.1	1.9	3.0	4.7	49.0
H6	4	3	0.8	1.4	0.7	16.0	0.9	1.4	0.7	18.5	1.0	1.7	1.3	23.3	2.4	4.3	6.0	31.6
H6	5	1	1.1	1.1	4.0	13.0	1.1	1.1	2.0	27.5	2.5	2.5	4.0	36.5	3.8	3.8	5.0	34.0
Overall H6		*7*	*0.9*	*2.0*	*1.4*	*15.6*	*0.9*	*2.0*	*1.0*	*22.3*	*1.2*	*2.5*	*2.0*	*25.1*	*2.4*	*4.3*	*5.3*	*37.8*
L3	3	12	0.6	4.6	0.3	3.7	0.9	4.6	0.4	9.0	1.3	5.6	0.9	27.6	3.1	7.0	1.8	34.7
L3	4	11	0.6	3.6	0.5	4.2	1.0	3.6	0.8	18.5	1.1	3.6	0.9	22.7	3.4	7.2	3.8	28.5
L3	5	9	1.7	4.8	1.0	9.4	2.1	6.1	1.2	12.6	3.4	6.6	1.4	21.8	5.8	10.0	2.8	31.0
Overall L3		*32*	*0.9*	*4.8*	*0.6*	*6.5*	*1.3*	*6.1*	*0.8*	*13.6*	*1.8*	*6.6*	*1.1*	*24.4*	*4.0*	*10.0*	*2.8*	*31.6*
L6	3	2	0.5	0.9	1.5	14.0	0.6	1.0	1.0	21.0	0.7	1.3	2.5	18.6	2.3	4.4	3.5	34.9
L6	4	3	1.2	2.1	0.7	16.0	1.3	2.1	0.7	18.5	1.5	2.8	1.3	23.3	3.5	6.4	6.0	34.6
L6	5	1	1.8	1.8	4.0	10.7	1.9	1.9	2.0	21.5	3.7	3.7	4.0	35.8	5.3	5.3	5.0	33.4
Overall L6		*6*	*1.1*	*2.1*	*1.5*	*14.2*	*1.1*	*2.1*	*1.0*	*20.1*	*1.6*	*3.7*	*2.2*	*23.9*	*3.4*	*6.4*	*5.0*	*34.5*

units for class H3 while the differences between the classes are small if there is no limit imposed (32.3 vs. 37.8 units). The same observation holds for classes with low holding costs.

Furthermore, the results show that the difference in cost improvement is small between a limit of 25% and 50%. Figure 4 shows that the largest difference can be observed between maxima of 75% and 100% (i.e., no limit). Restricting the demand moved to 75% of the demand per customer per period approximately halves the percentage cost improvement over the IRP. As an example, consider instance class L3 which has an average cost improvement of 4.0% if there is no limit, and only 1.5% in case of a maximum of 75%. This shows that if there is any restriction on the amount of demand that can be moved, a large share of the potential cost improvement is lost. This can be explained by the fact that in case of a limit, some customers must be served by a vehicle while their demand would have been moved and no visit would be required if there was no limitation on the moved demand. Therefore, routing costs increase and the improvement over the IRP is lower. The number of required replenishments is also enhanced by the limitation that demand can only be moved if there is no inventory left. Hence, replenishing a customer once and spreading this inventory over multiple periods combined with moving some demand every period is not possible; instead, replenishments with a vehicle are necessary.
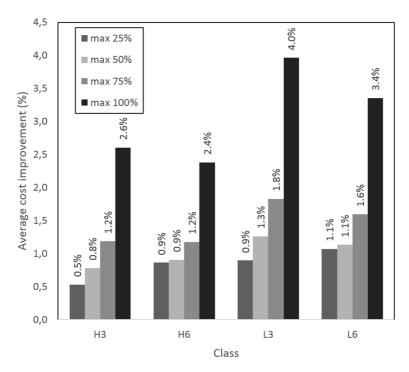
Fig. 4 Average cost improvement for maximum demand move by class

Although limiting the demand that can be moved, to 25% for example, clearly results in lower cost improvements over the IRP than imposing no limit. From a service perspective, this can still be preferable. Even with moving a very limited amount of goods, we still find average cost improvements between 0.5 and 1.1%, and up to 4.8% maximally, which can be substantial in practice.

6 Conclusion

In this paper, we introduced the Inventory Routing Problem with Demand Moves (IRPDM). This problem is an extension of the IRP with the addition that a customer can satisfy (part of) the demand of another customer. Although originally inspired by redirecting ATM-users to nearby ATMs, the IRPDM can prove useful to a variety of settings, such as bicycle sharing systems where users can collect or return a bicycle at several stations. We formulate a mathematical model for the IRPDM as an extension of the IRP formulation of Desaulniers et al. [1] and we develop a BPC solution method including non-trivially adjusted VIs stemming from the IRP.

The IRPDM is solved on IRP benchmark instances from the literature [16] and the performance of three types of VIs is analyzed. The tests show that MCSIs (see [19] for the IRP) adjusted for the IRPDM are more effective than adjusted CIs (see [1] for the IRP), and that using both these types of inequalities results in the best performance of the algorithm. To assess the impact of allowing for demand moves in the IRP, we compare the solutions of the IRPDM to those of the IRP. Moreover, we analyze the average number and size of demand moves to develop management insights.

Cost improvements of up to 10% are achieved for a demand move cost of $m = 0.01$ per unit of demand and unit of distance and if there is no limit on the moved demand. Moreover, it is observed that there is on average approximately one demand move per day, which implies that these improvements are achieved without a large change in the solutions compared to the IRP. The designed algorithm can solve instances with up to twenty customers, three periods, and five vehicles to optimality, which is limited. It must be noted that the IRPDM is much more difficult than the IRP, for which instances up to fifty customers can be solved to optimality with a state-of-the-art BPC method [1]. Sensitivity analysis on both the demand move costs and the maximum on the moved demand per customer per period is performed. Varying the demand move costs shows that the impact of increasing the costs is larger for a longer planning horizon than for a shorter planning horizon on both the percentage cost improvement over the IRP and the number of demand moves performed. Limiting the demand that can be moved per period of one customer to 75% of its demand already has a considerable impact on the cost improvement over the IRP compared with the cost improvement if there is no limit. The percentage cost improvement is approximately halved in case of 75% compared to 100%. Even by allowing only 25% of the demand to be moved, we observe cost improvements up to 4.8% and around 1% on average compared to the classical IRP.

In this paper, we limit ourselves to the case in which initial inventory can only be used to satisfy demand of the customer itself for algorithmic reasons. An extension

would be to develop an exact solution method that does accommodate satisfying moved demand with the initial inventory. A challenge can especially be found in the design of VIs for this problem as discussed in Section 4.3.7. Moreover, the results show that allowing for demand moves can lead to significant cost savings. Therefore, the design of a heuristic solution method for the IRPDM capable of solving larger scale IRPDM instances is an interesting future research direction. A helpful insight obtained in this paper which can be used in the development of heuristics is that the number of demand moves taking place in optimal solutions is rather limited. Finally, in our model, we only consider a load capacity constraint on the vehicles. In practice, the number of ATMs that can be served by one vehicle in one period is often limited by time, which is now not considered in the IRPDM. Hence, demand moves can also be useful if there is insufficient vehicle *time* capacity to replenish all ATMs in a certain area. It would be interesting to investigate the impact of allowing demand moves if the number of customers that can be replenished is further limited, for example, by limiting the number of customers served per vehicle.

Compliance with Ethical Standards

Conflict of Interest The authors declare that they have no conflict of interest.

Appendix 1: Results per instance

Tables 5, 6, and 7 show detailed results on the instances used for Section 5.2 when $m = 0.01$ and no maximum on the moved demand. The fleet size differs per table ($K = 3,4,5$). The three tables report per instance the computation time for the IRPDM if the instance is solved within 2 h of running time and whether the instance was solved to optimality (*y/n*). Thereafter, the upper bound ("UB"), the root lower bound ("LB_{root}"), and the lower bound after adding valid inequalities ("LB_{cuts}") is given. The size of the tree ("Tree"), the number of added CIs ("CI") and MCSIs ("MCSI") are then given, followed by the percentage cost improvement over the IRP ("Impr."), the number of demand moves ("Nr.DM"), and the average size of the demand moves ("Sz.DM") of the solution. Finally, the best upper bound found for the IRP is reported ("UB"), retrieved from [24] and an indication ("Opt.") whether the instance was solved to optimality (*y/n*). The instance is indicated in the following format $C\_Hp\_Nc\_Kv\_i$ with C the level of holding costs, p the number of periods, c the number of customers, v the number of vehicles, and i the index of the instance. If no solution is available because the instance is not solved to optimality, a dash is filled out. We only include instances for which at least the root node is solved. For two IRP instances, there is no feasible solution possible, and their upper bounds are therefore unknown (Unk).

Table 5 Detailed results on K3 instances

Instance	IRPDM											IRP	
	$T(s)$	Opt.	UB	LB$_{root}$	LB$_{cuts}$	Tree	CI	MCSI	Impr.	Nr.DM	Sz.DM	UB	Opt.
High_H3_N5_K3_1	0.1	y	2212.2	2212.2	2212.2	7	0	0	3.76	1	50.0	2298.7	y
High_H3_N5_K3_2	0.4	y	2334.5	2081.6	2299.1	65	0	2	1.49	1	14.0	2369.9	y
High_H3_N5_K3_3	0.1	y	4019.3	3996.5	3996.5	15	2	0	4.10	3	40.3	4191.3	y
High_H3_N5_K3_4	0.4	y	2772.3	2719.6	2720.6	32	2	2	3.54	4	14.3	2874.1	y
High_H3_N5_K3_5	0.0	y	2561.5	2561.5	2561.5	4	0	0	3.83	2	81.0	2663.7	y
High_H3_N10_K3_1	49.7	y	5497.4	5377.8	5427.6	79	7	10	0.16	1	34.0	5506.1	y
High_H3_N10_K3_2	52.2	y	5680.7	5522.0	5628.4	177	4	4	1.09	1	44.0	5743.3	y
High_H3_N10_K3_3	7.9	y	4735.3	4623.3	4710.2	28	0	9	1.51	2	22.5	4808.1	y
High_H3_N10_K3_4	21.9	y	5145.5	4952.2	5060.0	56	0	9	3.56	7	32.3	5335.3	y
High_H3_N10_K3_5	3.0	y	5182.1	5038.1	5106.4	26	0	6	0.81	2	54.0	5224.5	y
High_H3_N15_K3_1	77.3	y	6184.9	6074.0	6184.6	12	6	6	0.93	2	18.5	6242.9	y
High_H3_N15_K3_2	607.8	y	6038.6	5969.1	6034.7	31	1	5	0.54	2	10.0	6071.3	y
High_H3_N15_K3_3	154.6	y	6908.1	6895.3	6908.1	8	0	5	0.26	2	8.5	6926.2	y
High_H3_N15_K3_4	7270.0	n	-	5471.7	5544.2	12	0	9	-	-	-	5705.2	y
High_H3_N15_K3_5	7277.4	n	-	5554.8	5583.2	124	9	10	-	-	-	5967.3	y
High_H3_N20_K3_1	7201.0	n	-	7730.8	7909.2	1	6	8	-	-	-	8165.4	y
High_H3_N20_K3_2	7485.9	n	-	7172.0	7278.9	3	0	12	-	-	-	7499.5	y
High_H3_N20_K3_3	2402.9	y	7833.9	7788.9	7824.5	13	0	4	0.08	2	16.5	7840.5	y

Table 5 (continued)

Instance	IRPDM											IRP	
	$T(s)$	Opt.	UB	LB_{root}	LB_{cuts}	Tree	CI	MCSI	Impr.	Nr.DM	Sz.DM	UB	Opt.
High_H3_N20_K3_4	7200.8	n	-	7598.0	7687.4	16	0	12	-	-	-	7919.1	y
High_H3_N20_K3_5	7200.5	n	-	8964.2	9030.6	1	0	11	-	-	-	9149.2	y
High_H3_N25_K3_4	7201.9	n	-	8804.3	8854.9	4	0	8	-	-	-	9049.1	y
Low_H3_N5_K3_1	0.0	y	1335.0	1335.0	1335.0	5	0	0	6.68	1	50.0	1430.5	y
Low_H3_N5_K3_2	0.7	y	1549.2	1286.0	1511.0	93	1	3	2.12	1	14.0	1582.7	y
Low_H3_N5_K3_3	0.1	y	2835.1	2800.0	2800.0	15	2	0	5.42	3	40.3	2997.4	y
Low_H3_N5_K3_4	0.5	y	2175.7	2124.7	2129.0	33	3	2	5.04	4	14.3	2291.2	y
Low_H3_N5_K3_5	0.0	y	1408.0	1408.0	1408.0	4	0	0	6.98	2	81.0	1513.8	y
Low_H3_N10_K3_1	92.0	y	2725.9	2605.6	2646.4	135	9	10	0.24	1	34.0	2732.6	y
Low_H3_N10_K3_2	48.6	y	3395.0	3230.8	3344.8	177	2	4	2.17	1	44.0	3470.2	y
Low_H3_N10_K3_3	9.3	y	2574.4	2458.5	2546.2	27	0	9	2.83	2	22.5	2649.3	y
Low_H3_N10_K3_4	33.2	y	2977.3	2783.2	2893.9	83	0	9	6.47	7	32.3	3183.4	y
Low_H3_N10_K3_5	5.7	y	2422.4	2278.6	2334.6	53	1	8	1.57	1	77.0	2461.1	y
Low_H3_N15_K3_1	165.2	y	2728.9	2603.9	2724.1	12	3	10	1.97	2	18.5	2783.8	y
Low_H3_N15_K3_2	851.8	y	2725.2	2664.8	2721.1	34	3	5	1.18	2	10.0	2757.8	y
Low_H3_N15_K3_3	120.8	y	3055.3	3044.7	3055.3	10	0	6	0.57	1	11.0	3072.8	y

Table 5 (continued)

| Instance | IRPDM | | | | | | | | | | | IRP | |
	T(s)	Opt.	UB	LB_{root}	LB_{cuts}	Tree	CI	MCSI	Impr.	Nr.DM	Sz.DM	UB	Opt.
Low_H3_N15_K3_4	7201.5	n	-	2636.5	2712.7	29	1	8	-	-	-	2886.3	y
Low_H3_N15_K3_5	7201.1	n	-	2839.0	2871.2	90	17	10	-	-	-	3260.6	y
Low_H3_N20_K3_1	7200.5	n	-	3148.3	3326.8	3	1	17	-	-	-	3605.7	y
Low_H3_N20_K3_2	8942.2	n	-	2585.0	2697.1	3	1	11	y-	-	-	2908.5	y
Low_H3_N20_K3_3	737.5	y	3049.4	3008.3	3045.0	14	0	6	0.50	2	16.5	3064.8	y
Low_H3_N20_K3_4	7201.1	n	-	3754.7	3843.1	18	4	9	-	-	y-	4088.9	y
Low_H3_N20_K3_5	7200.7	n	-	3938.7	4056.7	7	2	15	-	-	-	4124.2	y
Low_H3_N25_K3_4	7200.8	n	y-	3398.2	3451.4	8	3	10	-	-	-	3659.1	y
Low_H3_N25_K3_5	7200.7	n	-	3943.1	4035.1	1	0	4	-	-	-	4120.3	y
High_H6_N5_K3_1	38.5	y	7259.1	7166.0	7233.3	1425	0	3	0.00	0	-	7259.1	y
High_H6_N5_K3_2	1394.6	y	6331.5	6149.5	6162.7	32258	1	7	3.04	8	42.1	6530.4	y
High_H6_N5_K3_3	7200.1	n	9876.2	9267.5	9510.2	89083	7	6	-	3	12.0	9862.9	y
High_H6_N5_K3_4	20.1	y	6242.1	5982.5	6020.5	525	1	6	2.56	6	55.8	6405.7	y
High_H6_N5_K3_5	1337.1	y	5871.2	5739.8	5739.8	18955	0	1	1.70	5	64.6	5972.8	y
High_H6_N10_K3_1	7200.5	n	-	11095.7	11098.5	1551	0	8	-	-	-	11440.9	y
High_H6_N10_K3_2	7200.3	n	-	11039.7	11123.5	2559	0	18	-	-	-	11584.8	n

Table 5 (continued)

| Instance | IRPDM | | | | | | | | | | | | IRP | |
	$T(s)$	Opt.	UB	LB_{root}	LB_{cuts}	Tree	CI	MCSI	Impr.	Nr.DM	Sz.DM	UB	Opt.
High_H6_N10_K3_3	7200.3	n	-	9752.7	9825.1	2572	3	16	-	-	-	9982.7	y
High_H6_N10_K3_4	7200.7	n	-	10551.6	10585.4	845	5	13	-	-	-	11168.5	y
High_H6_N10_K3_5	7200.3	n	10604.7	10339.6	10365.3	3749	0	11	-	2	34.0	10702.1	y
Low_H6_N5_K3_1	73.5	y	4650.5	4559.5	4630.1	2442	0	3	0.14	1	14.0	4657.2	y
Low_H6_N5_K3_2	2003.8	y	4007.3	3834.0	3842.8	45914	1	7	5.06	8	42.1	4221.0	y
Low_H6_N5_K3_3	7200.1	n	7729.6	7114.2	7356.3	86419	2	7	-	0	-	7703.5	y
Low_H6_N5_K3_4	16.5	y	4298.7	4042.1	4077.4	447	0	6	4.37	6	55.8	4495.3	y
Low_H6_N5_K3_5	5241.9	y	3792.0	3656.1	3656.1	67053	0	0	2.25	7	44.7	3879.1	y
Low_H6_N10_K3_1	7200.1	n	-	6757.0	6759.4	1658	1	5	-	-	-	7138.6	n
Low_H6_N10_K3_2	7200.4	n	-	7725.4	7807.8	2670	1	12	-	-	-	8276.2	n
Low_H6_N10_K3_3	7200.8	n	-	5910.4	5980.6	2339	4	14	-	-	-	6185.3	y
Low_H6_N10_K3_4	7200.4	n	-	6860.1	6900.0	1040	2	14	-	-	-	7483.3	y
Low_H6_N10_K3_5	7200.4	n	5774.0	5474.0	5503.1	4470	0	7	-	2	34.0	5824.1	y

Table 6 Detailed results on K4 instances

Instance	IRPDM											IRP	
	$T(s)$	Opt.	UB	LB_{root}	LB_{cuts}	Tree	CI	MCSI	Impr.	Nr.DM	Sz.DM	UB	Opt.
High_H3_N5_K4_1	0.0	y	2395.9	2383.7	2395.8	6	0	1	3.08	1	50.0	2472.1	y
High_H3_N5_K4_2	18.9	y	2577.2	2328.6	2397.3	3251	1	3	0.86	2	12.0	2599.6	y
High_H3_N5_K4_3	0.1	y	4557.1	4307.6	4534.9	10	1	1	5.21	3	27.7	4807.5	y
High_H3_N5_K4_4	0.1	y	3032.2	3012.0	3023.0	4	0	2	5.55	4	14.0	3210.2	y
High_H3_N5_K4_5	0.0	y	2755.8	2755.8	2755.8	4	0	0	2.43	4	40.5	2824.5	y
High_H3_N10_K4_1	7.3	y	6021.1	5816.4	6004.3	36	2	7	0.00	0	-	6021.1	y
High_H3_N10_K4_2	19.2	y	6416.6	6095.8	6370.9	217	0	6	1.88	4	59.5	6539.3	y
High_H3_N10_K4_3	4.6	y	5109.4	4960.4	5079.4	41	1	9	0.34	2	220	5127.0	y
High_H3_N10_K4_4	40.4	y	5660.1	5432.9	5569.3	218	2	12	2.99	7	33.3	5834.7	y
High_H3_N10_K4_5	20.7	y	5594.2	5285.4	5448.8	201	6	8	0.89	3	43.3	5644.4	y
High_H3_N15_K4_1	7201.3	n	6931.3	6376.2	6461.4	1192	11	14	-	0	-	6611.3	y
High_H3_N15_K4_2	7204.8	n	-	6299.8	6370.1	446	2	8	-	-	-	6705.9	y
High_H3_N15_K4_3	7201.8	n	-	7249.7	7452.8	470	3	9	-	-	-	7607.7	y
High_H3_N15_K4_4	7155.5	y	5970.6	5822.5	5915.0	307	0	4	0.78	3	26.0	6017.5	y
High_H3_N15_K4_5	7217.2	n	-	5923.9	6168.8	516	9	12	-	-	-	6375.4	y

Table 6 (continued)

Instance	IRPDM											IRP	
	$T(s)$	Opt.	UB	LB_{root}	LB_{cuts}	Tree	CI	MCSI	Impr.	Nr.DM	Sz.DM	UB	Opt.
High_H3_N20_K4_1	7469.9	n	-	8246.4	8459.9	20	4	12	-	-	-	8717.8	y
High_H3_N20_K4_2	7287.6	n	-	7357.5	7459.6	3	1	14	-	-	-	7710.9	y
High_H3_N20_K4_3	7201.9	n	-	8144.6	8290.0	114	1	15	-	-	-	8414.1	y
High_H3_N20_K4_4	7255.0	n	-	8131.7	8350.9	28	2	14	-	-	-	8589.3	y
High_H3_N20_K4_5	7204.3	n	-	9569.2	9697.8	68	5	11	-	-	-	9782.6	y
High_H3_N25_K4_1	7201.9	n	-	8826.2	9031.2	1	0	18	-	-	-	9287.9	y
High_H3_N25_K4_2	7202.9	n	-	9982.4	10114.5	14	2	15	-	-	-	10264.0	n
High_H3_N25_K4_3	7200.8	n	-	10763.7	10814.6	1	0	3	-	-	-	11026.8	y
High_H3_N25_K4_4	7216.4	n	-	9127.3	9201.5	32	6	11	-	-	-	9436.9	y
High_H3_N25_K4_5	7201.4	n	-	11633.4	11711.7	5	0	21	-	-	-	11806.0	y
Low_H3_N5_K4_1	0.1	y	1518.7	1507.2	1518.6	12	0	1	5.17	1	50.0	1601.6	y
Low_H3_N5_K4_2	20.4	y	1791.3	1535.1	1607.4	3502	0	3	1.18	3	8.0	1812.7	y
Low_H3_N5_K4_3	0.1	y	3360.9	3112.8	3341.5	13	1	1	6.74	3	27.7	3603.7	y
Low_H3_N5_K4_4	0.1	y	2441.5	2420.8	2436.3	11	0	2	7.22	4	14.0	2631.6	y

Table 6 (continued)

Instance	IRPDM											IRP	
	$T(s)$	Opt.	UB	LB_{root}	LB_{cuts}	Tree	CI	MCSI	Impr.	Nr.DM	Sz.DM	UB	Opt.
Low_H3_N5_K4_5	0.0	y	1590.2	1590.2	1590.2	4	0	0	5.14	5	32.4	1676.4	y
Low_H3_N10_K4_1	16.1	y	3261.9	3046.2	3234.5	86	0	7	0.00	0	-	3261.9	y
Low_H3_N10_K4_2	12.4	y	4131.6	3811.3	4087.2	124	5	4	3.26	6	35.2	4271.0	y
Low_H3_N10_K4_3	7.2	y	2947.0	2793.6	2913.7	67	2	12	0.73	3	25.3	2968.7	y
Low_H3_N10_K4_4	55.2	y	3490.0	3253.6	3398.9	249	1	12	5.25	8	29.1	3683.2	y
Low_H3_N10_K4_5	42.6	y	2832.0	2509.5	2675.2	374	9	6	1.39	2	49.5	2872.1	y
Low_H3_N15_K4_1	7201.6	n	3349.4	2912.6	3003.1	1573	11	13	-	2	45.5	3166.4	y
Low_H3_N15_K4_2	7200.6	n	3309.8	2996.2	3064.8	655	3	8	-	5	24.4	3396.7	y
Low_H3_N15_K4_3	7200.9	n	-	3404.9	3610.8	293	4	8	-	-	-	3757.4	y
Low_H3_N15_K4_4	3454.7	y	3142.5	2979.5	3084.4	293	1	5	1.80	7	14.3	3200.2	y
Low_H3_N15_K4_5	7201.2	n	3851.9	3208.1	3451.6	562	23	9	-	4	24.8	3671.1	y
Low_H3_N20_K4_1	7338.8	n	-	3663.6	3891.6	20	4	16	-	-	-	4148.0	y
Low_H3_N20_K4_2	7200.7	n	-	2756.8	2860.5	32	5	17	-	-	-	3128.1	y
Low_H3_N20_K4_3	7248.7	n	-	3366.8	3508.9	24	1	23	-	-	-	3645.5	y
Low_H3_N20_K4_4	7201.3	n	-	4288.9	4516.5	84	7	16	-	-	-	4787.9	n
Low_H3_N20_K4_5	7210.8	n	-	4542.4	4672.2	84	11	15	-	-	-	4764.8	y
Low_H3_N25_K4_1	7215.1	n	-	3457.6	3554.8	1	0	3	-	-	-	3949.7	y
Low_H3_N25_K4_2	7233.3	n	-	4219.2	4353.8	16	17	8	-	-	-	4502.9	n
Low_H3_N25_K4_3	7200.7	n	-	4422.7	4530.4	1	0	8	-	-	-	4687.6	y
Low_H3_N25_K4_4	7204.9	n	-	3727.9	3804.8	34	6	15	-	-	-	4044.8	y

Table 6 (continued)

| Instance | IRPDM | | | | | | | | | | | IRP | |
	T(s)	Opt.	UB	LB$_{root}$	LB$_{cuts}$	Tree	CI	MCSI	Impr.	Nr.DM	Sz.DM	UB	Opt.
Low_H3_N25_K4_5	7202.3	n	-	4519.2	4587.5	12	0	20	-	-	-	4672.8	y
High_H6_N5_K4_1	23.9	y	8074.3	7904.6	8058.6	986	1	8	0.46	2	50.0	8111.3	y
High_H6_N5_K4_2	3045.6	y	7136.7	6871.6	6879.0	91012	9	4	3.71	8	31.4	7411.3	y
High_H6_N5_K4_3	112.0	y	11179.6	10957.5	11027.7	3394	0	1	2.58	9	12.8	11475.3	y
High_H6_N5_K4_4	12.7	y	6686.1	6568.3	6580.4	675	3	4	4.29	7	32.1	6985.4	y
High_H6_N5_K4_5	3469.1	y	6904.5	6743.1	6792.3	71871	0	1	1.31	2	82.0	6996.4	y
High_H6_N10_K4_1	7200.1	n	-	12431.1	12519.0	4868	0	14	-	-	-	12801.4	n
High_H6_N10_K4_2	7200.1	n	-	12727.7	12771.1	7152	3	14	-	-	-	13190.4	n
High_H6_N10_K4_3	7200.3	n	-	10633.1	10815.1	7520	2	22	-	-	-	11067.9	y
High_H6_N10_K4_4	7200.4	n	-	11652.0	11760.7	4105	3	17	-	-	-	12323.9	y
High_H6_N10_K4_5	7200.1	n	-	11046.8	11097.2	9728	1	12	-	-	-	11471.5	y
Low_H6_N5_K4_1	21.6	y	5457.4	5295.4	5451.7	885	7	8	0.89	4	56.7	5506.2	y
Low_H6_N5_K4_2	4518.2	y	4814.3	4568.9	4576.1	136841	5	2	6.08	7	35.9	5125.8	y
Low_H6_N5_K4_3	116.2	y	9030.5	8806.9	8877.3	3424	0	1	3.11	7	15.0	9320.7	y
Low_H6_N5_K4_4	17.1	y	4761.9	4644.5	4650.4	908	4	2	6.37	7	32.1	5085.8	y
Low_H6_N5_K4_5	7200.1	n	4825.1	4660.4	4713.6	148679	0	1	-	3	53.7	4913.4	y
Low_H6_N10_K4_1	7200.1	n	-	8090.4	8180.9	4490	0	17	-	-	-	8421.9	n
Low_H6_N10_K4_2	7200.1	n	-	9431.4	9467.5	8072	1	12	-	-	-	9875.1	n
Low_H6_N10_K4_3	7200.2	n	-	6800.1	6970.1	7256	3	21	-	-	-	7255.6	y
Low_H6_N10_K4_4	7200.2	n	-	7966.3	8081.3	4876	2	19	-	-	-	8645.1	y
Low_H6_N10_K4_5	7200.2	n	-	6187.5	6232.1	10028	1	10	-	-	-	6604.9	y

Table 7 Detailed results on K5 instances

Instance	IRPDM											IRP	
	T(s)	Opt.	UB	LB$_{root}$	LB$_{cuts}$	Tree	CI	MCSI	Impr.	Nr.DM	Sz.DM	UB	Opt.
High_H3_N5_K5_1	0.0	y	2480.8	2434.5	2480.8	5	0	1	3.75	1	30.0	2577.5	y
High_H3_N5_K5_2	1.7	y	2675.2	2543.8	2570.5	358	1	2	4.70	1	28.0	2807.3	y
High_H3_N5_K5_3	0.2	y	4858.0	4744.5	4827.4	25	1	3	5.87	2	32.5	5160.9	y
High_H3_N5_K5_4	0.1	y	3720.3	3647.8	3710.4	7	3	1	4.60	2	21.0	3899.7	y
High_H3_N5_K5_5	0.0	y	2974.7	2974.7	2974.7	4	0	0	6.07	4	38.5	3166.8	y
High_H3_N10_K5_1	3235.5	y	6392.1	6239.7	6275.8	18095	1	6	1.61	2	44.0	6496.7	y
High_H3_N10_K5_2	294.4	y	6857.0	6667.9	6772.3	1704	1	7	1.49	4	51.7	6960.9	y
High_H3_N10_K5_3	2.3	y	5467.2	5285.2	5438.9	17	6	18	1.62	5	15.2	5557.2	y
High_H3_N10_K5_4	361.3	y	6056.0	5853.4	5924.1	1667	12	10	4.01	6	27.3	6308.8	y
High_H3_N10_K5_5	0.6	y	5634.9	5499.8	5621.4	11	0	6	2.37	1	31.0	5771.5	y
High_H3_N15_K5_1	5860.0	y	6985.1	6718.4	6900.3	803	0	14	0.55	2	36.0	7023.7	y
High_H3_N15_K5_2	7201.1	n	7067.4	6603.7	6942.2	1199	3	6	-	4	23.0	7194.3	y
High_H3_N15_K5_3	7211.3	n	7960.6	7680.2	7825.7	578	2	14	-	3	20.7	7981.7	y
High_H3_N15_K5_4	7200.5	n	-	6152.6	6332.3	228	2	5	-	-	-	6388.1	y
High_H3_N15_K5_5	7221.9	n	-	6383.7	6678.1	514	10	14	-	-	-	6963.6	y
High_H3_N20_K5_1	4858.2	y	8883.2	8708.2	8832.9	69	13	14	0.92	2	15.0	8965.2	y
High_H3_N20_K5_2	7227.2	n	-	7583.2	7676.6	18	10	15	-	-	-	7910.5	y
High_H3_N20_K5_3	7201.3	n	-	8461.8	8654.6	243	9	16	-	-	-	8787.5	y
High_H3_N20_K5_4	7267.5	n	-	8647.4	8840.9	202	7	14	-	-	-	9047.3	y
High_H3_N20_K5_5	7202.5	n	-	10215.0	10381.4	110	11	22	-	-	-	10523.8	y

Table 7 (continued)

Instance	IRPDM											IRP	
	T(s)	Opt.	UB	LB_{root}	LB_{cuts}	Tree	CI	MCSI	Impr.	Nr.DM	Sz.DM	UB	Opt.
High_H3_N25_K5_1	7233.7	n	-	9142.5	9257.3	19	13	15	-	-	-	9473.2	y
High_H3_N25_K5_2	7204.6	n	-	10503.0	10614.4	24	2	17	-	-	-	10757.7	y
High_H3_N25_K5_3	7202.3	n	-	11345.9	11481.0	9	1	12	-	-	-	11576.3	y
High_H3_N25_K5_4	7202.7	n	-	9437.5	9513.7	90	11	14	-	-	-	9719.3	y
High_H3_N25_K5_5	7211.2	n	-	12215.8	12362.3	14	2	25	-	-	-	12451.2	y
Low_H3_N5_K5_1	0.0	y	1606.8	1556.6	1606.8	4	0	1	6.05	1	30.0	1710.3	y
Low_H3_N5_K5_2	2.2	y	1885.7	1755.7	1781.7	480	3	3	6.63	1	28.0	2019.6	y
Low_H3_N5_K5_3	0.2	y	3656.1	3555.5	3625.8	34	1	3	7.81	2	30.5	3965.8	y
Low_H3_N5_K5_4	0.2	y	3136.5	3056.3	3124.8	25	1	3	5.50	2	21.0	3319.0	y
Low_H3_N5_K5_5	0.0	y	1807.2	1807.2	1807.2	5	0	0	10.02	4	38.5	2008.5	y
Low_H3_N10_K5_1	2979.2	y	3623.8	3475.3	3514.2	18209	4	6	2.82	2	44.0	3728.8	y
Low_H3_N10_K5_2	229.5	y	4572.5	4385.6	4489.5	1832	1	7	2.33	4	51.7	4681.3	y
Low_H3_N10_K5_3	3.3	y	3313.5	3119.2	3271.2	39	8	19	2.58	5	15.2	3401.2	y
Low_H3_N10_K5_4	356.3	y	3871.4	3673.5	3740.3	1827	10	9	6.87	5	32.8	4157.0	y
Low_H3_N10_K5_5	0.8	y	2862.0	2729.3	2850.7	14	0	9	4.41	1	31.0	2993.9	y
Low_H3_N15_K5_1	1821.5	y	3517.8	3251.5	3437.4	718	2	16	1.76	2	36.0	3580.8	y
Low_H3_N15_K5_2	7201.1	n	3736.5	3302.7	3642.2	1102	3	6	-	4	20.0	3889.4	y
Low_H3_N15_K5_3	7201.1	n	4073.0	3845.3	3989.4	694	5	8	-	2	16.5	4133.6	y
Low_H3_N15_K5_4	7201.2	n	-	3311.0	3489.9	153	1	5	-	-	-	3572.5	y

Table 7 (continued)

| Instance | IRPDM | | | | | | | | | | | IRP | |
	T(s)	Opt.	UB	LB$_{root}$	LB$_{cuts}$	Tree	CI	MCSI	Impr.	Nr.DM	Sz.DM	UB	Opt.
Low_H3_N15_K5_5	7200.6	n	-	3663.7	3967.9	634	4	13	-	-	-	4256.2	y
Low_H3_N20_K5_1	5899.1	y	4310.5	4137.9	4287.6	76	9	18	2.14	3	30.7	4404.8	y
Low_H3_N20_K5_2	7268.9	n	-	2989.8	3081.3	26	7	12	-	-	-	3344.1	y
Low_H3_N20_K5_3	7207.0	n	-	3686.7	3874.7	272	6	22	-	-	-	4016.5	y
Low_H3_N20_K5_4	7203.6	n	-	4802.3	4999.5	90	5	18	-	-	-	5215.8	y
Low_H3_N20_K5_5	7231.0	n	-	5185.1	5361.8	54	25	20	-	-	-	5506.0	y
Low_H3_N25_K5_1	7220.6	n	-	3759.8	3880.7	12	20	21	-	-	-	4095.2	y
Low_H3_N25_K5_2	7233.8	n	-	4739.7	4851.6	61	12	8	-	-	-	5009.8	y
Low_H3_N25_K5_3	7226.8	n	-	5009.1	5130.1	8	1	12	-	-	-	5229.3	y
Low_H3_N25_K5_4	7203.3	n	-	4033.6	4122.3	116	12	20	-	-	-	4378.8	y
Low_H3_N25_K5_5	7272.3	n	-	5109.3	5242.0	28	0	25	-	-	-	5314.1	y
High_H6_N5_K5_1	443.4	y	8948.5	8834.4	8909.3	21623	8	7	1.04	3	48.7	9043.0	y
High_H6_N5_K5_2	3116.5	y	7952.5	7626.7	7677.8	128143	4	3	4.17	8	28.5	8298.9	y
High_H6_N5_K5_3	7200.1	n	13332.6	12885.3	13005.0	194178	0	6	-	2	10.5	13399.1	y
High_H6_N5_K5_4	6.7	y	7872.5	7631.1	7853.6	479	0	11	3.75	5	34.0	8179.6	y
High_H6_N5_K5_5	181.8	y	7918.0	7860.5	7860.5	6611	0	0	-	9	15.2	Unk	n
High_H6_N10_K5_1	7200.1	n	-	13622.8	13729.7	8772	0	11	-	-	-	14146.1	n

Table 7 (continued)

| Instance | IRPDM | | | | | | | | | | | IRP | |
	$T(s)$	Opt.	UB	LB_{root}	LB_{cuts}	Tree	CI	MCSI	Impr.	Nr.DM	Sz.DM	UB	Opt.
High_H6_N10_K5_2	7200.1	n	-	14378.7	14435.0	13973	0	11	-	-	-	14949.5	n
High_H6_N10_K5_3	7200.1	n	-	11606.2	11779.0	11015	1	18	-	-	-	12054.3	y
High_H6_N10_K5_4	7200.1	n	-	12854.9	13001.2	8236	10	25	-	-	-	13754.9	y
High_H6_N10_K5_5	7200.2	n	-	11709.5	11746.9	14993	2	13	-	-	-	12068.5	y
Low_H6_N5_K5_1	427.5	y	6327.4	6224.6	6297.7	20562	9	6	1.83	3	48.7	6445.7	y
Low_H6_N5_K5_2	7200.2	n	5692.2	5317.8	5374.3	288586	3	3	-	7	24.0	6009.4	y
Low_H6_N5_K5_3	7200.1	n	11157.6	10773.0	10892.4	202314	0	5	-	7	12.0	11282.7	y
Low_H6_N5_K5_4	4.9	y	5954.7	5727.7	5952.0	344	0	10	5.26	5	33.4	6285.0	y
Low_H6_N5_K5_5	1581.8	y	5830.2	5784.2	5784.2	49449	0	0	-	10	13.7	Unk	n
Low_H6_N10_K5_1	7200.1	n	-	9289.8	9414.6	9312	0	15	-	-	-	9914.7	n
Low_H6_N10_K5_2	7200.1	n	-	11080.0	11138.7	14965	0	10	-	-	-	11633.8	n
Low_H6_N10_K5_3	7200.1	n	-	7785.3	7954.8	14293	0	16	-	-	-	8239.6	y
Low_H6_N10_K5_4	7200.1	n	-	9175.4	9325.4	10152	5	21	-	-	-	10093.5	n
Low_H6_N10_K5_5	7200.3	n	8122.8	6858.9	6895.3	15403	6	13	-	5	18.8	7214.5	y

Appendix 2: Results on impact of *m* and maximum on moved demand

Tables 8, 9, and 10 show detailed results on the instances used for Sections 5.2.1 and 5.2.2. The fleet size differs per table ($K = 3,4,5$). The three tables show for each instance details on the time ("$T(s)$"), the upper bound ("UB"), and the best lower bound ("LB_{best}"). The details are shown for different values of m (0.01, 0.005, 0.05, 0.1) and varying maxima on the moved demand (100%, 25%, 50%, 75%). Remind that instances with up to 15 customers and horizon three, and with five customers and horizon six are tested. The formatting of the instance number is the same as in Appendix 1.

Table 8 Detailed results impact demand move cost and maximum on moved demand, K3 instances

Instance	m = 0.01&Max.100%			m = 0.005&Max.100%			m = 0.05&Max.100%			m = 0.1&Max.100%			m = 0.01&Max.25%			m = 0.01&Max.50%			m = 0.01&Max.75%		
	T(s)	UB	LB_{best}	T(s)	UB	LB_{best}	T(s)	UB	LB_{best}	T(s)	UB	LB_{best}	T(s)	UB	LB_{best}	T(s)	UB	LB_{best}	T(s)	UB	LB_{best}
High_H3_N5_K3_1	0	2212.2	2212.2	0	2186.7	2186.7	0	2298.7	2298.7	0	2298.7	2298.7	0	2298.7	2298.7	0	2287.9	2287.9	0	2287.9	2287.9
High_H3_N5_K3_2	0	2334.5	2334.5	1	2328.3	2328.3	0	2369.9	2369.9	0	2369.9	2369.9	2	2369.9	2369.9	0	2334.5	2334.5	0	2334.5	2334.5
High_H3_N5_K3_3	0	4019.3	4019.3	0	3930.3	3930.3	0	4121.4	4121.4	0	4191.3	4191.3	1	4191.3	4191.3	1	4191.3	4191.3	1	4157.4	4157.4
High_H3_N5_K3_4	0	2772.3	2772.3	0	2736.4	2736.4	0	2852.2	2852.2	1	2859.2	2859.2	1	2846.6	2846.6	1	2846.6	2846.6	1	2846.6	2846.6
High_H3_N5_K3_5	0	2561.5	2561.5	0	2507.1	2507.1	0	2617.4	2617.4	0	2647.1	2647.1	0	2593.6	2593.6	0	2593.6	2593.6	0	2578.4	2578.4
High_H3_N10_K3_1	50	5497.4	5497.4	99	5471.3	5471.3	4	5506.1	5506.1	2	5506.1	5506.1	48	5506.1	5506.1	106	5506.1	5506.1	56	5497.4	5497.4
High_H3_N10_K3_2	52	5680.7	5680.7	48	5669.9	5669.9	27	5743.3	5743.3	8	5743.3	5743.3	119	5743.3	5743.3	326	5743.3	5743.3	40	5680.7	5680.7
High_H3_N10_K3_3	8	4735.3	4735.3	22	4724.3	4724.3	77	4808.1	4808.1	8	4808.1	4808.1	135	4808.1	4808.1	761	4808.1	4808.1	34	4778.1	4778.1
High_H3_N10_K3_4	22	5145.5	5145.5	22	5034.4	5034.4	4875	5335.3	5335.3	857	5335.3	5335.3	6240	5335.3	5335.3	7009	5301.6	5301.6	3000	5275.9	5275.9
High_H3_N10_K3_5	3	5182.1	5182.1	5	5121.7	5121.7	1	5224.5	5224.5	1	5224.5	5224.5	5	5224.5	5224.5	9	5224.5	5224.5	32	5224.5	5224.5
High_H3_N15_K3_1	77	6184.9	6184.9	200	6172.4	6172.4	40	6196.5	6196.5	60	6199.6	6199.6	156	6194.1	6194.1	144	6194.1	6194.1	99	6194.1	6194.1
High_H3_N15_K3_2	608	6038.6	6038.6	-	-	6030.1	104	6071.3	6071.3	27	6071.3	6071.3	902	6071.3	6071.3	1264	6071.3	6071.3	635	6071.3	6071.3
High_H3_N15_K3_3	155	6908.1	6908.1	269	6903.9	6903.9	99	6926.2	6926.2	27	6926.2	6926.2	244	6924.9	6924.9	263	6924.9	6924.9	389	6924.9	6924.9
High_H3_N15_K3_4	-	-	5558.5	-	-	5545.6	-	6820.6	5606.4	-	-	5603.9	-	-	5595.7	-	-	5604.1	-	-	5585.6
High_H3_N15_K3_5	-	-	5684.6	-	-	5651.8	-	-	5795.0	-	6154.8	5802.7	-	-	5732.8	-	-	5665.1	-	-	5705.3
Low_H3_N5_K3_1	0	1335.0	1335.0	0	1309.5	1309.5	0	1430.5	1430.5	0	1430.5	1430.5	0	1430.5	1430.5	0	1417.1	1417.1	1	1417.1	1417.1

Table 8 (continued)

Instance	$m = 0.01$&Max.100%			$m = 0.005$&Max.100%			$m = 0.05$&Max.100%			$m = 0.1$&Max.100%			$m = 0.01$&Max.25%			$m = 0.01$&Max.50%			$m = 0.01$&Max.75%		
	$T(s)$	UB	LB_{best}	$T(s)$	UB	LB_{best}	$T(s)$	UB	LB_{best}	$T(s)$	UB	LB_{best}	$T(s)$	UB	LB_{best}	$T(s)$	UB	LB_{best}	$T(s)$	UB	LB_{best}
Low_H3_N5_K3_2	1	1549.2	1549.2	1	1542.9	1542.9	0	1582.7	1582.7	0	1582.7	1582.7	2	1582.7	1582.7	0	1549.1	1549.1	1	1549.1	1549.1
Low_H3_N5_K3_3	0	2835.1	2835.1	0	2746.1	2746.1	0	2927.6	2927.6	0	2997.4	2997.4	1	2997.4	2997.4	1	2997.4	2997.4	1	2973.1	2973.1
Low_H3_N5_K3_4	1	2175.7	2175.7	0	2139.8	2139.8	1	2269.2	2269.2	1	2275.6	2275.6	1	2263.6	2263.6	1	2263.6	2263.6	2	2263.6	2263.6
Low_H3_N5_K3_5	0	1408.0	1408.0	0	1354.6	1354.6	0	1467.5	1467.5	1	1497.2	1497.2	0	1443.7	1443.7	1	1443.7	1443.7	0	1428.5	1428.5
Low_H3_N10_K3_192	192	2725.9	2725.9	124	2697.4	2697.4	6	2732.6	2732.6	3	2732.6	2732.6	74	2732.6	2732.6	255	2732.6	2732.6	78	2725.9	2725.9
Low_H3_N10_K3_249	249	3395.0	3395.0	121	3384.2	3384.2	45	3470.2	3470.2	21	3470.2	3470.2	166	3470.2	3470.2	475	3470.2	3470.2	30	3395.0	3395.0
Low_H3_N10_K3_39	39	2574.4	2574.4	34	2563.4	2563.4	16	2649.3	2649.3	7	2649.3	2649.3	151	2649.3	2649.3	929	2649.3	2649.3	91	2625.9	2625.9
Low_H3_N10_K3_433	433	2977.3	2977.3	20	2866.2	2866.2	1631	3183.4	3183.4	1473	3183.4	3183.4	-	3201.4	3158.1	3612	3131.8	3131.8	1920	3109.7	3109.7
Low_H3_N10_K3_56	56	2422.4	2422.4	10	2363.0	2363.0	2	2461.1	2461.1	1	2461.1	2461.1	8	2461.1	2461.1	27	2461.1	2461.1	91	2457.9	2457.9
Low_H3_N15_K3_1165	1165	2728.9	2728.9	162	2716.3	2716.3	41	2740.8	2740.8	69	2743.9	2743.9	157	2738.4	2738.4	340	2738.4	2738.4	272	2738.4	2738.4
Low_H3_N15_K3_2852	2852	2725.2	2725.2	-	2720.6	2719.6	128	2757.8	2757.8	26	2757.8	2757.8	845	2757.8	2757.8	1076	2757.8	2757.8	681	2757.8	2757.8
Low_H3_N15_K3_3121	3121	3055.3	3055.3	455	3051.6	3051.6	38	3072.8	3072.8	23	3072.8	3072.8	212	3072.8	3072.8	220	3072.8	3072.8	317	3072.8	3072.8
Low_H3_N15_K3_4	-	-	2740.0	-	-	2702.5	-	-	2801.6	-	-	2822.1	-	-	2797.2	-	-	2780.7	-	-	2751.7
Low_H3_N15_K3_5	-	-	2963.7	-	-	2921.3	-	-	3065.4	-	-	3088.4	-	-	2988.4	-	-	2995.8	-	-	2971.4

Table 8 (continued)

Instance	$m = 0.01\&Max.100\%$			$m = 0.005\&Max.100\%$			$m = 0.1\&Max.100\%$			$m = 0.05\&Max.100\%$			$m = 0.01\&Max.25\%$			$m = 0.01\&Max.50\%$			$m = 0.01\&Max.75\%$		
	$T(s)$	UB	LB_{best}	$T(s)$	UB	LB_{best}	$T(s)$	UB	LB_{best}	$T(s)$	UB	LB_{best}	$T(s)$	UB	LB_{best}	$T(s)$	UB	LB_{best}	$T(s)$	UB	LB_{best}
High_H6_N5_K3_1	39	7259.1	7259.1	18	7214.1	7214.1	34	7259.1	7259.1	35	7259.1	7259.1	42	7259.1	7259.1	45	7259.1	7259.1	40	7259.1	7259.1
High_H6_N5_K3_2	139	56331.5	56331.5	86	6148.5	6148.5	999	6433.4	6433.4	2387	6460.1	6460.1	2454	6399.4	6399.4	3478	6399.4	6399.4	6561	6399.4	6399.4
High_H6_N5_K3_3	-	9876.2	9825.0	-	9904.0	9760.7	542	9862.9	9862.9	440	9862.9	9862.9	3224	9848.8	9848.8	3597	9848.8	9848.8	4633	9848.8	9848.8
High_H6_N5_K3_4	20	6242.1	6242.1	9	6045.2	6045.2	21	6405.7	6405.7	20	6405.7	6405.7	44	6378.2	6378.2	70	6377.5	6377.5	166	6364.8	6364.8
High_H6_N5_K3_5	1337	5871.2	5871.2	320	5764.6	5764.6	1520	5972.8	5972.8	1410	5972.8	5972.8	5020	5951.0	5951.0	-	5946.3	5938.0	5236	5914.4	5914.4
Low_H6_N5_K3_1	74	4650.5	4650.5	28	4595.4	4595.4	98	4657.2	4657.2	93	4657.2	4657.2	49	4650.5	4650.5	61	4650.5	4650.5	61	4650.5	4650.5
Low_H6_N5_K3_2	2004	4007.3	4007.3	147	3821.6	3821.6	2320	4120.6	4120.6	5502	4147.3	4147.3	5335	4088.8	4088.8	-	4125.9	4082.8	-	4111.4	4068.7
Low_H6_N5_K3_3	-	7729.6	7673.1	-	7746.0	7610.0	559	7703.5	7703.5	474	7703.5	7703.5	2573	7685.2	7685.2	3431	7685.2	7685.2	4357	7685.2	7685.2
Low_H6_N5_K3_4	17	4298.7	4298.7	10	4102.3	4102.3	22	4495.3	4495.3	23	4495.3	4495.3	31	4454.8	4454.8	75	4452.3	4452.3	203	4437.9	4437.9
Low_H6_N5_K3_5	5242	3792.0	3792.0	1496	3688.1	3688.1	4909	3879.1	3879.1	4571	3879.1	3879.1	-	3880.6	3840.4	-	3871.5	3821.5	-	3833.8	3805.9

Table 9 Detailed results impact demand move cost and maximum on moved demand, K4 instances

Instance	m = 0.01&Max.100%			m = 0.005&Max.100%			m = 0.05&Max.100%			m = 0.1&Max.100%			m = 0.01&Max.25%			m = 0.01&Max.50%			m = 0.01&Max.75%		
	$T(s)$	UB	LB_{best}	$T(s)$	UB	LB_{best}	$T(s)$	UB	LB_{best}	$T(s)$	UB	LB_{best}	$T(s)$	UB	LB_{best}	$T(s)$	UB	LB_{best}	$T(s)$	UB	LB_{best}
High_H3_N5_K4_1	0	2395.9	2395.9	0	2370.4	2370.4	0	2421.7	2421.7	0	2431.9	2431.9	0	2413.5	2413.5	0	2413.5	2413.5	0	2413.5	2413.5
High_H3_N5_K4_2	19	2577.2	2577.2	14	2557.3	2557.3	5	2599.6	2599.6	5	2599.6	2599.6	16	2599.6	2599.6	18	2599.6	2599.6	25	2599.6	2599.6
High_H3_N5_K4_3	0	4557.1	4557.1	0	4496.4	4496.4	1	4775.7	4775.7	1	4807.5	4807.5	2	4794.2	4794.2	0	4668.0	4668.0	1	4668.0	4668.0
High_H3_N5_K4_4	0	3032.2	3032.2	0	2997.0	2997.0	0	3210.2	3210.2	0	3210.2	3210.2	0	3210.2	3210.2	1	3210.2	3210.2	4	3210.2	3210.2
High_H3_N5_K4_5	0	2755.8	2755.8	0	2701.4	2701.4	0	2824.5	2824.5	0	2824.5	2824.5	0	2824.5	2824.5	0	2824.5	2824.5	1	2806.8	2806.8
High_H3_N10_K4_1	7	6021.1	6021.1	9	6002.9	6002.9	2	6021.1	6021.1	2	6021.1	6021.1	6	6021.1	6021.1	4	6021.1	6021.1	8	6021.1	6021.1
High_H3_N10_K4_2	19	6416.6	6416.6	11	6325.1	6325.1	21	6485.3	6485.3	80	6492.6	6492.6	21	6479.4	6479.4	72	6478.5	6478.5	213	6474.2	6474.2
High_H3_N10_K4_3	5	5109.4	5109.4	7	5092.9	5092.9	2	5127.0	5127.0	2	5127.0	5127.0	7	5127.0	5127.0	8	5127.0	5127.0	9	5127.0	5127.0
High_H3_N10_K4_4	40	5660.1	5660.1	17	5545.4	5545.4	20	5827.9	5827.9	18	5832.6	5832.6	118	5824.2	5824.2	35	5784.2	5784.2	97	5784.2	5784.2
High_H3_N10_K4_5	21	5594.2	5594.2	22	5496.1	5496.1	9	5644.4	5644.4	6	5644.4	5644.4	18	5644.4	5644.4	40	5636.6	5636.6	66	5636.6	5636.6
High_H3_N15_K4_1	-	6931.3	6555.0	-	6928.5	6515.1	-	6625.6	6593.5	-	6660.0	6601.4	-	-	6535.4	-	6984.5	6563.1	-	-	6558.6
High_H3_N15_K4_2	-	-	6486.8	-	6578.1	6418.6	-	6705.0	6612.1	-	6793.2	6649.6	-	-	6590.6	-	-	6563.9	-	-	6549.6
High_H3_N15_K4_3	-	-	7522.0	-	-	7492.9	-	7779.6	7581.8	-	7618.4	7594.3	-	-	7540.1	-	7824.4	7531.3	-	-	7521.6
High_H3_N15_K4_4	715	5970.6	5970.6	-	-	5942.0	1359	6017.5	6017.5	614	6017.5	6017.5	-	6160.3	6003.4	-	-	5996.2	-	-	6002.3

Table 9 (continued)

Instance	m = 0.01&Max.100%		m = 0.005&Max.100%		m = 0.05&Max.100%		m = 0.1&Max.100%		m = 0.01&Max.25%		m = 0.01&Max.50%		m = 0.01&Max.75%	
	T(s)UB	LB$_{best}$	T(s)UB	LB$_{best}$	T(s) UB	LB$_{best}$	T(s) UB	LB$_{best}$	T(s)UB	LB$_{best}$	T(s)UB	LB$_{best}$	T(s)UB	LB$_{best}$
High_H3_N15_K4_5	–	6284.6	–	6247.0	1058 6351.5	6351.5	3464 6370.5	6370.5	–	6329.3	–	6323.6	–	6293.7
Low_H3_N5_K4_1	0 1518.7	1518.7	0 1493.2	1493.2	0 1552.0	1552.0	0 1562.2	1562.2	0 1543.8	1543.8	0 1543.8	1543.8	0 1543.8	1543.8
Low_H3_N5_K4_2	20 1791.3	1791.3	16 1770.5	1770.5	7 1812.7	1812.7	6 1812.7	1812.7	19 1812.7	1812.7	18 1812.7	1812.7	25 1812.7	1812.7
Low_H3_N5_K4_3	0 3360.9	3360.9	0 3300.2	3300.2	1 3571.9	3571.9	1 3603.7	3603.7	3 3603.7	3603.7	0 3473.6	3473.6	1 3473.6	3473.6
Low_H3_N5_K4_4	0 2441.5	2441.5	0 2406.3	2406.3	0 2631.6	2631.6	0 2631.6	2631.6	1 2631.6	2631.6	1 2631.6	2631.6	5 2631.6	2631.6
Low_H3_N5_K4_5	0 1590.2	1590.2	0 1536.5	1536.5	0 1676.4	1676.4	0 1676.4	1676.4	1 1676.4	1676.4	2 1676.4	1676.4	2 1658.4	1658.4
Low_H3_N10_K4_1	16 3261.9	3261.9	13 3235.5	3235.5	3 3261.9	3261.9	3 3261.9	3261.9	11 3261.9	3261.9	16 3261.9	3261.9	17 3261.9	3261.9
Low_H3_N10_K4_2	12 4131.6	4131.6	10 4046.2	4046.2	62 4216.2	4216.2	147 4223.5	4223.5	48 4206.0	4206.0	215 4204.3	4204.3	454 4198.3	4198.3
Low_H3_N10_K4_3	7 2947.0	2947.0	5 2924.4	2924.4	4 2968.7	2968.7	4 2968.7	2968.7	12 2968.7	2968.7	20 2968.7	2968.7	25 2968.7	2968.7
Low_H3_N10_K4_4	55 3490.0	3490.0	22 3375.2	3375.2	28 3670.1	3670.1	46 3674.8	3674.8	613 3666.4	3666.4	34 3615.7	3615.7	161 3615.7	3615.7
Low_H3_N10_K4_5	43 2832.0	2832.0	41 2728.8	2728.8	7 2872.1	2872.1	5 2872.1	2872.1	19 2872.1	2872.1	57 2870.8	2870.8	105 2870.8	2870.8
Low_H3_N15_K4_1	– 3349.4	3092.0	– 3417.4	3056.3	– 3172.9	3131.4	– 3169.7	3141.7	– 3188.9	3099.2	– 3475.1	3101.8	–	3093.5
Low_H3_N15_K4_2	– 3309.8	3186.2	–	3111.3	– 3402.9	3313.9	– 3588.4	3353.7	–	3255.0	–	3241.4	– 3694.7	3240.1
Low_H3_N15_K4_3	–	3666.4	–	3645.0	– 3794.3	3740.4	– 3760.4	3746.9	–	3705.1	–	3677.1	–	3672.4

Table 9 (continued)

Instance	m = 0.01&Max.100%			m = 0.005&Max.100%			m = 0.05&Max.100%			m = 0.1&Max.100%			m = 0.01&Max.25%			m = 0.01&Max.50%			m = 0.01&Max.75%		
	T(s)	UB	LB$_{best}$	T(s)	UB	LB$_{best}$	T(s)	UB	LB$_{best}$	T(s)	UB	LB$_{best}$	T(s)	UB	LB$_{best}$	T(s)	UB	LB$_{best}$	T(s)	UB	LB$_{best}$
Low_H3_N15_K4_4	34	4355	3142.5	3440	3105.7	3105.7	1375	3200.2	3200.2	567	3200.2	3200.2	3103	3180.0	3180.0	7108	3180.0	3180.0	5550	3180.0	3180.0
Low_H3_N15_K4_5	-	3851.9	3579.2	-	-	3542.6	1403	3643.1	3643.1	3671	3661.4	3661.4	-	3622.2	-	-	3600.9	-	-	-	3588.1
High_H6_N5_K4.1	24	8074.3	8074.3	9	7965.5	7965.5	350	8111.3	8111.3	349	8111.3	8111.3	530	8111.3	8111.3	1122	8111.3	8111.3	1953	8111.3	8111.3
High_H6_N5_K4.2	3046	7136.7	7136.7	3385	7018.1	7018.1	-	7440.9	7345.5	-	7444.2	7351.5	-	7469.9	7249.5	-	7211.2	-	-	7177.2	-
High_H6_N5_K4_3	112	11179.6	11179.6	141	11101.5	11101.5	457	11440.2	11440.2	692	11474.2	11474.2	885	11344.9	11344.9	605	11319.0	11319.0	1131	11319.0	11319.0
High_H6_N5_K4.4	13	6686.1	6686.1	78	6553.2	6553.2	53	6961.9	6961.9	86	6985.4	6985.4	106	6887.0	6887.0	145	6887.0	6887.0	301	6864.6	6864.6
High_H6_N5_K4_5	3469	6904.5	6904.5	5340	6850.3	6850.3	707	6996.4	6996.4	603	6996.4	6996.4	-	6999.2	6974.9	-	6998.9	6941.2	-	6974.2	6932.9
Low_H6_N5_K4_1	22	5457.4	5457.4	13	5341.7	5341.7	740	5506.2	5506.2	753	5506.2	5506.2	1003	5506.2	5506.2	2014	5506.2	5506.2	5291	5506.2	5506.2
Low_H6_N5_K4_2	4518	4814.3	4814.3	5453	4695.7	4695.7	-	5130.5	5016.8	-	5108.7	5022.2	-	4915.8	-	-	4874.4	-	-	4842.0	-
Low_H6_N5_K4_3	116	9030.5	9030.5	179	8959.8	8959.8	542	9284.4	9284.4	945	9320.7	9320.7	912	9189.1	9189.1	678	9160.7	9160.7	1379	9160.7	9160.7
Low_H6_N5_K4_4	17	4761.9	4761.9	107	4629.0	4629.0	83	5052.2	5052.2	144	5085.8	5085.8	171	4977.3	4977.3	247	4977.3	4977.3	369	4945.8	4945.8
Low_H6_N5_K4_5	-	4825.1	4805.5	-	4773.5	4740.1	-	4913.4	4913.2	5862	4913.4	4913.4	-	4901.4	4854.8	-	4867.4	4826.2	-	4954.0	4814.3

Table 10 Detailed results impact demand move cost and maximum on moved demand, K5 instances

Instance	$m=0.01$&Max.100%			$m=0.005$&Max.100%			$m=0.05$&Max.100%			$m=0.1$&Max.100%			$m=0.01$&Max.25%			$m=0.01$&Max.50%			$m=0.01$&Max.75%		
	$T(s)$	UB	LB_{best}	$T(s)$	UB	LB_{best}	$T(s)$	UB	LB_{best}	$T(s)$	UB	LB_{best}	$T(s)$	UB	LB_{best}	$T(s)$	UB	LB_{best}	$T(s)$	UB	LB_{best}
High_H3_N5_K5_1	0	2480.8	2480.8	0	2465.2	2465.2	0	2536.8	2536.8	1	2567.4	2567.4	0	2512.3	2512.3	0	2480.8	2480.8	0	2480.8	2480.8
High_H3_N5_K5_2	2	2675.2	2675.2	2	2658.9	2658.9	1	2774.9	2774.9	4	2807.3	2807.3	8	2807.3	2807.3	11	2801.9	2801.9	2	2675.2	2675.2
High_H3_N5_K5_3	0	4858.0	4858.0	0	4810.8	4810.8	1	5117.5	5117.5	2	5160.9	5160.9	6	5145.4	5145.4	12	5145.4	5145.4	1	4971.0	4971.0
High_H3_N5_K5_4	0	3720.3	3720.3	0	3695.7	3695.7	1	3899.7	3899.7	0	3899.7	3899.7	1	3899.7	3899.7	1	3899.7	3899.7	4	3899.7	3899.7
High_H3_N5_K5_5	0	2974.7	2974.7	0	2917.4	2917.4	2	3136.6	3136.6	4	3166.8	3166.8	0	3064.1	3064.1	1	3064.1	3064.1	3	3064.1	3064.1
High_H3_N10_K5_1	3233	6392.1	6392.1	5977	6332.1	6332.1	450	6496.7	6496.7	350	6496.7	6496.7	-	-	-	-	6487.9	6466.6	3419	6429.8	6429.8
High_H3_N10_K5_2	294	6857.0	6857.0	736	6790.7	6790.7	21	6960.9	6960.9	17	6960.9	6960.9	234	6960.9	6960.9	953	6960.9	6960.9	651	6920.2	6920.2
High_H3_N10_K5_3	2	5467.2	5467.2	3	5437.4	5437.4	6	5534.8	5534.8	16	5556.7	5556.7	226	5535.6	5535.6	6	5503.3	5503.3	7	5503.3	5503.3
High_H3_N10_K5_4	361	6056.0	6056.0	832	5974.6	5974.6	97	6308.8	6308.8	641	6308.8	6308.8	805	6282.2	6282.2	1249	6282.2	6282.2	3857	6282.2	6282.2
High_H3_N10_K5_5	1	5634.9	5634.9	1	5596.0	5596.0	3	5713.6	5713.6	10	5771.5	5771.5	1	5656.4	5656.4	1	5656.4	5656.4	1	5656.4	5656.4
High_H3_N15_K5_1	5860	6985.1	6985.1	-	7609.5	6918.1	-	7084.1	6966.7	-	7068.7	6974.7	-	7486.4	6997.3	6000	7018.5	7018.5	-	7025.7	7013.3
High_H3_N15_K5_2	-	7067.4	7019.4	-	7000.4	6931.6	4189	7118.8	7118.8	-	7218.3	7163.1	-	7191.1	7116.4	3871	7085.8	7085.8	-	-	7061.5
High_H3_N15_K5_3	-	7960.6	7905.5	-	-	7862.2	754	7949.7	7949.7	2446	7966.5	7966.5	6510	7936.2	7936.2	-	-	7930.4	-	-	7920.3
High_H3_N15_K5_4	-	-	6361.7	-	-	6642.0	6214	6388.1	6388.1	2137	6388.1	6388.1	2157	6388.1	6388.1	4405	6388.1	6388.1	-	-	6381.4
High_H3_N15_K5_5	-	-	6771.3	-	-	6713.9	-	-	6867.0	-	7076.0	6889.9	-	-	6803.3	-	-	6781.4	-	-	6773.0

Table 10 (continued)

Instance	$m = 0.01$&Max.100%			$m = 0.005$&Max.100%			$m = 0.05$&Max.100%			$m = 0.1$&Max.100%			$m = 0.01$&Max.25%			$m = 0.01$&Max.50%			$m = 0.01$&Max.75%		
	T(s)	UB	LB_{best}	T(s)	UB	LB_{best}	T(s)	UB	LB_{best}	T(s)	UB	LB_{best}	T(s)	UB	LB_{best}	T(s)	UB	LB_{best}	T(s)	UB	LB_{best}
Low_H3_N5_K5_1	0	1606.8	1606.8	0	1591.0	1591.0	0	1667.1	1667.1	1	1697.7	1697.7	0	1642.6	1642.6	0	1606.8	1606.8	0	1606.8	1606.8
Low_H3_N5_K5_2	2	1885.7	1885.7	1	1870.5	1870.5	1	1985.4	1985.4	3	2019.6	2019.6	8	2019.6	2019.6	12	2013.9	2013.9	1	1885.7	1885.7
Low_H3_N5_K5_3	0	3656.1	3656.1	1	3611.8	3611.8	2	3934.0	3934.0	2	3965.8	3965.8	7	3947.7	3947.7	15	3947.7	3947.7	1	3768.2	3768.2
Low_H3_N5_K5_4	0	3136.5	3136.5	0	3111.9	3111.9	1	3319.0	3319.0	0	3319.0	3319.0	1	3319.0	3319.0	2	3319.0	3319.0	6	3319.0	3319.0
Low_H3_N5_K5_5	0	1807.2	1807.2	0	1753.6	1753.6	3	1976.7	1976.7	6	2008.5	2008.5	1	1912.1	1912.1	2	1912.1	1912.1	4	1912.1	1912.1
Low_H3_N10_K5_1	2979	3623.8	3623.8	4869	3563.5	3563.5	699	3728.8	3728.8	540	3728.8	3728.8	5056	3728.8	3728.8	-	3713.6	3709.3	3696	3660.9	3660.9
Low_H3_N10_K5_2	230	4572.5	4572.5	584	4506.2	4506.2	25	4681.3	4681.3	18	4681.3	4681.3	262	4681.3	4681.3	1130	4681.3	4681.3	876	4647.0	4647.0
Low_H3_N10_K5_3	3	3313.5	3313.5	4	3283.7	3283.7	10	3377.7	3377.7	40	3401.2	3401.2	721	3381.3	3381.3	7	3341.9	3341.9	25	3341.9	3341.9
Low_H3_N10_K5_4	356	3871.4	3871.4	820	3790.1	3790.1	127	4139.2	4139.2	192	4157.0	4157.0	393	4101.6	4101.6	1442	4101.6	4101.6	3556	4101.6	4101.6
Low_H3_N10_K5_5	1	2862.0	2862.0	1	2831.7	2831.7	3	2937.9	2937.9	9	2993.9	2993.9	0	2877.1	2877.1	1	2877.1	2877.1	1	2877.1	2877.1
Low_H3_N15_K5_1	1822	3517.8	3517.8	-	3495.2	3457.2	2020	3576.1	3576.1	5691	3579.5	3579.5	5673	3559.0	3559.0	-	3584.7	3555.0	-	3584.5	3553.2
Low_H3_N15_K5_2	-	3736.5	3720.5	-	3705.0	3625.4	5861	3811.3	3811.3	-	3889.4	3852.9	6455	3807.2	3807.2	2882	3778.2	3778.2	-	-	3762.2
Low_H3_N15_K5_3	-	4073.0	4065.6	-	4076.4	4032.3	670	4098.4	4098.4	695	4115.0	4115.0	-	-	4080.3	-	-	4077.9	-	-	4076.1

Table 10 (continued)

Instance	$m = 0.01$&Max.100%			$m = 0.005$&Max.100%			$m = 0.05$&Max.100%			$m = 0.1$&Max.100%			$m = 0.01$&Max.25%			$m = 0.01$&Max.50%			$m = 0.01$&Max.75%		
	T(s)	UB	LB$_{best}$	T(s)	UB	LB$_{best}$	T(s)	UB	LB$_{best}$	T(s)	UB	LB$_{best}$	T(s)	UB	LB$_{best}$	T(s)	UB	LB$_{best}$	T(s)	UB	LB$_{best}$
Low_H3_N15_K5_4-	-	-	3513.8	-	-	3491.0	6145	3572.5	3572.5	5338	3572.5	3572.5	-	-	3570.4	-	-	3569.5	-	-	3546.4
Low_H3_N15_K5_5-	-	-	4062.6	-	-	4009.5	-	-	4145.3	-	4431.1	4158.6	-	-	4089.5	-	-	4068.8	-	-	4054.6
High_H6_N5_K5_1	443	8948.5	8948.5	232	8874.1	8874.1	6744	9030.1	9030.1	-	9043.7	9034.2	-	8998.6	8998.4	1256	8966.2	8966.2	902	8956.4	8956.4
High_H6_N5_K5_2	3117	7952.5	7952.5	3094	7839.5	7839.5	1434	8176.1	8176.1	-	8249.5	8211.7	-	8069.5	8069.5	-	8015.2	8015.2	6789	7989.1	7989.1
High_H6_N5_K5_3	-	13332.6	13254.1	-	13207.6	13172.1	327	13399.1	13399.1	257	13399.1	13399.1	-	13928.7	13351.4	-	13438.0	13321.6	-	-	13294.7
High_H6_N5_K5_4	7	7872.5	7872.5	2	7558.1	7558.1	768	8179.6	8179.6	692	8179.6	8179.6	412	8091.8	8091.8	1864	8087.1	8087.1	94	7975.6	7975.6
High_H6_N5_K5_5	182	7918.0	7918.0	725	7866.6	7866.6	749	8086.4	8086.4	713	8102.1	8102.1	130	7965.1	7965.1	730	7965.1	7965.1	1831	7965.1	7965.1
Low_H6_N5_K5_1	428	6327.4	6327.4	261	6253.0	6253.0	-	6422.0	6415.5	-	6445.8	6414.1	-	6412.8	6378.5	1495	6347.1	6347.1	2056	6345.9	6345.9
Low_H6_N5_K5_2	-	5692.2	5653.3	-	5613.0	5538.8	-	5982.1	5875.3	-	6023.5	5878.3	-	-	5728.8	-	-	5672.8	-	-	5656.0
Low_H6_N5_K5_3	-	11157.6	11106.8	-	11120.0	11022.6	1141	11282.7	11282.7	872	11282.7	11282.7	-	-	11203.5	-	-	11174.1	-	-	11140.8
Low_H6_N5_K5_4	5	5954.7	5954.7	1	5622.2	5622.2	1389	6285.0	6285.0	326	6285.0	6285.0	-	6169.0	6169.0	1557	6168.4	6168.4	102	6054.4	6054.6
Low_H6_N5_K5_5	1582	5830.2	5830.2	4029	5778.9	5778.9	3036	5999.5	5999.5	2726	6015.9	6015.9	1117	5879.5	5879.5	4470	5879.5	5879.5	-	5881.2	5873.3

References

1. Desaulniers. G, Rakke JG, Coelho LC (2016) A branch-price-and-cut algorithm for the inventory-routing problem. Transp. Sci. 50(3):1060–1076
2. Van Anholt RG, Coelho LC, Laporte G, Vis IFA (2016) An inventory-routing problem with pickups and deliveries arising in the replenishment of automated teller machines. Transp. Sci. 50(3):1077–1091
3. Larrain H, Coelho LC, Cataldo A (2017) A variable MIP neighborhood descent algorithm for managing inventory and distribution of cash in automated teller machines. Computers & Operations Research 85:22–31
4. Geismar HN, Sriskandarajah C, Zhu Y (2017) A review of operational issues in managing physical currency supply chains. Prod. Oper. Manag. 26(6):976–996
5. Coelho LC, Cordeau J-F, Laporte G (2012) The inventory-routing problem with transshipment. Computers & Operations Research 39(11):2537–2548
6. Coelho LC, Laporte G (2013) The exact solution of several classes of inventory-routing problems. Computers & Operations Research 40(2):558–565
7. Lefever W, Aghezzaf E-H, Hadj-Hamou K, Penz B (2018) Analysis of an improved branch-and-cut formulation for the inventory-routing problem with transshipment. Computers & Operations Research 98:137–148
8. Andersson H, Hoff A, Christiansen M, Hasle G, Løkketangen A (2010) Industrial aspects and literature survey: Combined inventory management and routing. Computers & Operations Research 37(9):1515–1536
9. Coelho LC, Cordeau J-F, Laporte G (2014) Thirty years of inventory routing. Transp. Sci. 48(1):1–19
10. Archetti C, Bianchessi N, Irnich S, Speranza MG (2014) Formulations for an inventory routing problem. Int. Trans. Oper. Res. 21(3):353–374
11. Abdelmaguid TF, Dessouky MM, Ordóñez F (2009) Heuristic approaches for the inventory-routing problem with backlogging. Computers & Industrial Engineering 56(4):1519–1534
12. Irnich S, Desaulniers G (2005) Shortest path problems with resource constraints. In: Desaulniers G, Desrosiers J, Solomon MM (eds) Column generation, pp 33–65. Springer
13. Desaulniers G, Lessard F, Hadjar A (2008) Tabu search, partial elementarity, and generalized k-path inequalities for the vehicle routing problem with time windows. Transp. Sci. 42(3):387–404
14. Baldacci R, Mingozzi A, Roberti R (2011) New route relaxation and pricing strategies for the vehicle routing problem. Oper. Res. 59(5):1269–1283
15. Desaulniers G, Madsen OliBG, Ropke S (2014) The vehicle routing problem with time windows. In: Toth P, Vigo D (eds) Vehicle routing: Problems, methods, and applications, 2nd edition, pp 119–159. MOS-SIAM series on optimization (SIAM, Philadelphia)
16. Archetti C, Bertazzi L, Laporte G, Speranza MG (2007) A Branch-and-Cut algorithm for a vendor-managed inventory-routing problem. Transp. Sci. 41(3):382–391
17. Coelho LC, Laporte G (2014) Improved solutions for inventory-routing problems through valid inequalities and input ordering. Int. J. Prod. Econ. 155:391–397
18. Desaulniers G (2010) Branch-and-price-and-cut for the split-delivery vehicle routing problem with time windows. Oper. Res. 58(1):179–192
19. Avella P, Boccia M, Wolsey LA (2018) Single-period cutting planes for inventory routing problems. Transp. Sci. 52(3):497–508

20. Laporte G, Nobert Y, Desrochers M (1985) Optimal routing under capacity and distance restrictions. Oper. Res. 33(5):1050–1073
21. Lysgaard J (2004) Cvrpsep. https://econ.au.dk/research/researcher-websites/jens-lysgaard/cvrpsep/, Accessed: 29 August 2020
22. Lysgaard J, Letchford AN, Eglese RW (2004) A new branch-and-cut algorithm for the capacitated vehicle routing problem. Math. Program. 100(2):423–445
23. Archetti C, Bouchard M, Desaulniers G (2011) Enhanced branch and price and cut for vehicle routing with split deliveries and time windows. Transp. Sci. 45(3):285–298
24. Coelho LC (2021) Problem instances inventory routing. http://www.leandro-coelho.com/instances/inventory-routing/, Accessed: 4 December 2017

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Affiliations

Annelieke C. Baller[1] · Said Dabia[1] ⬤ · Guy Desaulniers[2,3] · Wout E. H. Dullaert[1]

Annelieke C. Baller
annelieke.baller@ortec.com; a.c.baller@vu.nl

Guy Desaulniers
guy.desaulniers@gerad.ca

Wout E.H. Dullaert
wout.dullaert@vu.nl

[1] Department of Operations Analytics, Vrije Universiteit Amsterdam, 1081 HV, Amsterdam, The Netherlands

[2] GERAD, Montréal, Québec, H3T 2A7, Canada

[3] Department of Mathematics and Industrial Engineering, Polytechnique Montréal (Québec), Montréal, Québec, H3C 3A7, Canada