

University of Groningen

A simultaneous facility location and vehicle routing problem arising in health care logistics in the Netherlands

Veenstra, Marjolein; Roodbergen, Kees Jan; Coelho, Leandro C.; Zhu, Stuart X.

Published in:
European Journal of Operational Research

DOI:
[10.1016/j.ejor.2018.01.043](https://doi.org/10.1016/j.ejor.2018.01.043)

IMPORTANT NOTE: You are advised to consult the publisher's version (publisher's PDF) if you wish to cite from it. Please check the document version below.

Document Version
Publisher's PDF, also known as Version of record

Publication date:
2018

[Link to publication in University of Groningen/UMCG research database](#)

Citation for published version (APA):

Veenstra, M., Roodbergen, K. J., Coelho, L. C., & Zhu, S. X. (2018). A simultaneous facility location and vehicle routing problem arising in health care logistics in the Netherlands. *European Journal of Operational Research*, 268(2), 703-715. <https://doi.org/10.1016/j.ejor.2018.01.043>

Copyright

Other than for strictly personal use, it is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license (like Creative Commons).

The publication may also be distributed here under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license. More information can be found on the University of Groningen website: <https://www.rug.nl/library/open-access/self-archiving-pure/taverne-amendment>.

Take-down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from the University of Groningen/UMCG research database (Pure): <http://www.rug.nl/research/portal>. For technical reasons the number of authors shown on this cover page is limited to 10 maximum.



Innovative Applications of O.R.

A simultaneous facility location and vehicle routing problem arising in health care logistics in the Netherlands



Marjolein Veenstra^{a,1}, Kees Jan Roodbergen^{a,*}, Leandro C. Coelho^{a,b}, Stuart X. Zhu^a

^a University of Groningen, Faculty of Economics and Business, Department of Operations, the Netherlands

^b Canada Research Chair in Integrated Logistics, CIRRELT, Université Laval, Canada

ARTICLE INFO

Article history:

Received 28 July 2016

Accepted 23 January 2018

Available online 3 February 2018

Keywords:

Routing

Facility location

Health care logistics

Hybrid heuristic

ABSTRACT

This paper introduces a simultaneous facility location and vehicle routing problem that arises in health care logistics in the Netherlands. In this problem, the delivery of medication from a local pharmacy can occur via lockers, from where patients that are within the coverage distance of a locker can collect their medication, or by home delivery. The aim of the problem is to determine which lockers from a set of potential locker locations to open and to generate routes that visit the opened lockers and routes that visit the patients that are not covered by the opened lockers, while minimizing the routing costs and the opening costs of the lockers. We formally define this problem and solve it by applying a branch-and-bound algorithm to this mathematical formulation. Moreover, we propose a fast hybrid heuristic to solve the problem. Extensive computational results are given on a randomly generated instance set and an instance set inspired by practice from Alliance Healthcare Netherlands, an industrial partner. Our results indicate that our heuristic is able to consistently outperform a commercial solver applied to the mathematical formulation, and that its solutions are extremely robust. We provide important business insights on several parameters of the problem.

© 2018 Elsevier B.V. All rights reserved.

1. Introduction

Home delivery of medication is an increasing industry that helps alleviate the burden at hospitals, while still guaranteeing that patients receive proper care as required by the hospitals. In the Netherlands, Alliance Healthcare Netherlands (AHN) is a company that offers such services to hospitals. AHN runs various activities, including a wholesale operation for medicines and 60 pharmacies (under the consumer brand name Boots). AHN is required by its contract to a number of hospitals to supply patients with repeated medication after they are discharged. Note that patients are not free to choose a pharmacy and must receive the medication from the hospital contractor.

Within this context, a local pharmacy acts as a depot from where the medications are distributed to patients, and where a fleet of vehicles is available. Lockers can be opened at a set of

potential locations. We consider a set of patients that need repeated medication. When a locker is opened within a prespecified distance to a patient, their medication will be distributed via the locker. All other patients will be visited by a vehicle and receive their medications at home. Routes for replenishing lockers and routes for delivering medication to patients are separated. This separation of routes is due to several practical considerations. First, it must be guaranteed that lockers are replenished before a given time of the day. This is more difficult to achieve when combining patients in the same route, since the interaction time with the patients upon delivery is difficult to predict. Second, the ideal delivery period differs between lockers (before people return home from work) and home delivery (after people return home from work). Third, drivers that perform home delivery must have better interpersonal skills. Generally, the same reasons also cause a significant preference of the logistics company for delivering to lockers rather than to patients. The goal of the problem is to determine which lockers to open, and to generate the vehicle routes that visit the lockers and the vehicle routes that visit the patients. The objective is to minimize total costs, consisting of routing costs and opening costs of the lockers.

Fig. 1 shows an example derived from data of AHN, our industrial partner (for details we refer to Section 4.2). The figure depicts a depot (marked by a square), a set of patients (marked with

* Corresponding author at: University of Groningen, Faculty of Economics and Business, Department of Operations, The Netherlands.

E-mail addresses: mveenstramarjolein@gmail.com (M. Veenstra), k.j.roodbergen@rug.nl (K.J. Roodbergen), leandro.coelho@cirrelt.ca (L.C. Coelho), x.zhu@rug.nl (S.X. Zhu).

¹ New Nexus Business Intelligence, the Netherlands.

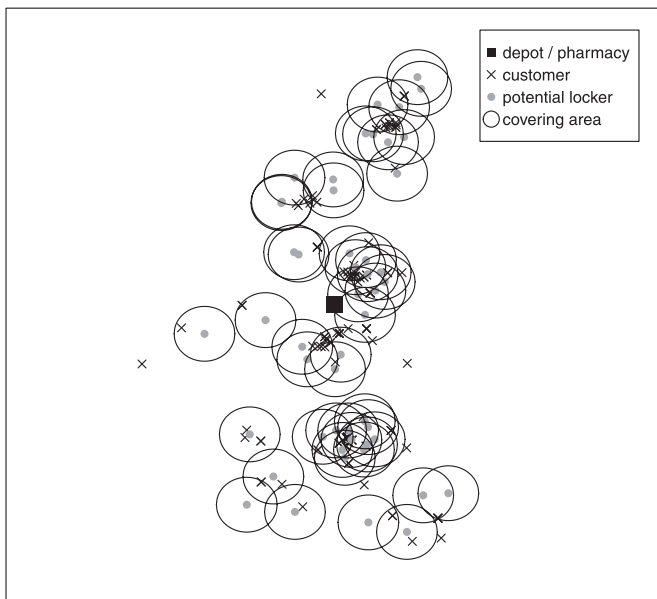


Fig. 1. Example of an instance based on data from AHN.

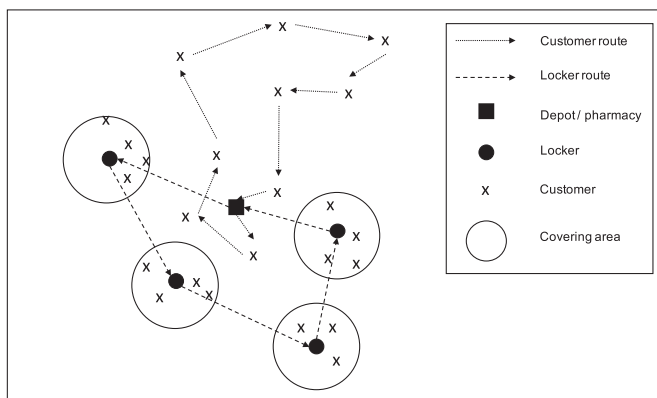


Fig. 2. Example of a solution to the problem.

crosses), and a set of potential locker locations (marked by gray circles) with the area they cover (marked by the larger circles around them). A solution to a smaller instance could be represented as in Fig. 2.

The problem setting as described before also arises in the location of pick-up lockers for e-commerce operations. However, to the best of our knowledge, this configuration has not been studied before. Nevertheless, related literature exists, namely the location-routing problem (LRP), the two-echelon location-routing problem (2E-LRP), and a number of covering problems. In the classical LRP (for surveys we refer to Albareda-Sambola, 2015; Drexler & Schneider, 2015; Lopes, Ferreira, Santos, & Barreto, 2013; Nagy & Salhi, 2007; Prodhon & Prins, 2014; Schneider & Drexler, 2017), the objective is to open some depots, assign customers to the opened depots, and design vehicle routes from the opened depots to the customers such that the total costs consisting of the depot costs, the fixed costs per vehicle used and the routing costs are minimized. A special case of the LRP that incorporates subcontracting options is introduced by Stenger, Schneider, Schwind, and Vigo (2012). In this problem, self-operated and subcontracted depots can be opened. The customers that are assigned to the subcontracted depots are served by an external carrier and therefore are not explicitly routed. The subcontracted depots show a similarity

to the opened lockers in our problem, namely, patients assigned to them are excluded from the routing decisions. However, two main features differentiate our problem from that of Stenger et al. (2012). First, patients can only be assigned to an opened locker within the coverage distance. Second, we have to create routes visiting all opened lockers for replenishments, whereas the subcontracted depots of Stenger et al. (2012) and Stenger et al. (2013) are not routed at all. The authors propose a hybrid heuristic for their problem, which is based on simulated annealing for the location decisions and variable neighborhood search for the routing decisions. Similar subcontracting issues have also arisen in the context of the VRP (Zäpfel & Bögl, 2008). The 2E-LRP (see Cuda, Guastaroba, & Speranza, 2015 for a survey), integrates the location of depots and satellites (intermediate facilities), the construction of vehicle routes from depots to satellites, and the construction of vehicle routes from the satellites to the customers.

The property of covering imposes that certain locations do not have to be visited as long as they are within a prescribed distance from another location that is visited. Covering exists in several routing problems, among these are the covering salesman problem (see Golden, Naji-Azimi, Raghavan, Salari, & Toth, 2012 and Salari & Naji-Azimi, 2012 for heuristics), and the covering tour problem (see Hachicha, Hodgson, Laporte, & Semet, 2000 and Allahyari, Salari, & Vigo, 2015 for heuristics, Tricoire, Graf, & Gutjahr, 2012 for a branch-and-cut algorithm, and Gendreau, Laporte, & Semet, 1997 and Hà, Bostel, Langevin, & Rousseau, 2013 for both a heuristic and an exact algorithm). Another research on covering is that of Naji-Azimi, Renaud, Ruiz, and Salari (2012) in which satellite distribution centers are located using a covering tour approach. In this problem, satellite distribution centers must be located such that they cover all customers. The satellite distribution centers can be compared to the lockers in our problem. Different from our problem, all customers need to be covered in Naji-Azimi et al. (2012) and consequently no routes visiting the remaining customers need to be generated. The authors provide a mathematical formulation for the problem and develop a multi-start heuristic.

Other problems combining routing and covering are the close-enough traveling salesman problem (CETSP) and the close-enough vehicle routing problem (CEVRP). In the CETSP a route needs to be designed to minimize the travel distance under the restriction that each customer is within a certain radius of the route (see Dong, Yang, & Chen, 2007; Gulczynski, Heath, & Price, 2006, and Mennell, 2009 for heuristics, and Yuan, Orłowska, & Sadiq, 2007 for an exact method). The CEVRP is an extension of the CETSP that considers multiple routes (see Mennell, 2009 for a mathematical formulation and a heuristic). Close-enough problems also exist in the arc routing version of the problem (see Ávila, Corberán, Plana, & Sanchis, 2016; Hà, Bostel, Langevin, & Rousseau, 2014, and Drexler, 2014).

The vehicle routing-allocation problem (VRAP) (Beasley & Nascimento, 1996) is also of interest. In the VRAP, not all customers are visited by the vehicle: some might not be visited at all, or allocated to one of the visited customer nodes. An application to postal service locating post boxes in order to maximize user convenience is presented in Labbé and Laporte (1986), and another one to food distribution in Solak, Scherrer, and Ghoniem (2014).

The goal of our paper is to introduce, model, and solve the simultaneous facility location and vehicle routing problem as previously described using a commercial solver applied to the mathematical formulation and a heuristic approach. We mathematically model and solve this new problem arising in the health care sector in the Netherlands both exactly and by means of a fast and efficient heuristic. We run several computational experiments on random instances and on instances created from data of AHN. Moreover, we also adapt instances from the classical LRP to assess the

performance of our methods. We show that our hybrid heuristic is fast and finds (near-)optimal solutions.

The remainder of this paper is organized as follows. In Section 2, we formally define and formulate the problem. A hybrid heuristic is proposed in Section 3. In Section 4, extensive computational experiments are reported on several instances, and managerial insights are given. Conclusions are given in Section 5.

2. Problem statement and mathematical formulation

The problem studied in this paper can be defined on a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{A})$, where \mathcal{V} is the set of nodes and \mathcal{A} is the set of arcs. Let $\mathcal{V} = \{0\} \cup \mathcal{P} \cup \mathcal{L}$, where $\{0\}$ is the depot, \mathcal{P} denotes the set of patients, and \mathcal{L} is the set of potential locker locations. Two sets of vehicles \mathcal{K} and \mathcal{M} are based at the depot. The coverage distance of a locker $j \in \mathcal{L}$ is given by r_j and a fixed opening cost F_j is associated with it. The cost, length and travel time of arc $(i, j) \in \mathcal{A}$ are denoted by c_{ij} , d_{ij} and t_{ij} . The cost for the vehicles to the patients are penalized by a factor ϕ for the reasons noted in Section 1. Each patient that is within the coverage distance of an opened locker is deemed served. The remaining patients need to be visited by the vehicles from the set \mathcal{K} . The opened lockers need to be visited by the vehicles from the set \mathcal{M} . A maximum route duration T_1 (T_2) is given for the vehicles of the set \mathcal{K} (\mathcal{M}). The route duration of a vehicle consists of the total of travel time and service time. The service time associated with node $i \in \mathcal{V}$ is given by s_i , where $s_i > 0$ for $i \in \mathcal{P} \cup \mathcal{L}$ and $s_0 = 0$. A vehicle route is feasible if it starts and ends at the depot and does not exceed the maximum route duration constraint.

Three types of decisions must be made, namely, (1) whether locker $j \in \mathcal{L}$ will be opened, (2) the construction of the patient routes, i.e., the vehicle routes that visit the patient nodes that are not assigned to a locker, and (3) the construction of the locker routes, i.e., the vehicle routes that visit the opened lockers. The objective of the problem is to minimize the opening costs of the lockers and routing costs.

The problem can be modeled as follows. Let binary variables w_i be equal to one if and only if node $i \in \mathcal{P}$ is assigned to the depot, i.e., the patient needs to be visited by a vehicle, z_i equal to one if and only if node $i \in \mathcal{P}$ is assigned to a locker, u_{ij} equal to one if and only if patient $i \in \mathcal{P}$ is assigned to locker $j \in \mathcal{L}$, and v_i equal to one if and only if locker $i \in \mathcal{L}$ is opened. For the patient routes, we use binary variables x_{ij}^k equal to one if and only if vehicle $k \in \mathcal{K}$ travels from node i to j , $i, j \in \mathcal{P} \cup \{0\}$, y_i^k equal to one if and only if vehicle $k \in \mathcal{K}$ visits node $i \in \mathcal{P} \cup \{0\}$, and integer variables h_{ij}^k reflecting the load on vehicle $k \in \mathcal{K}$ while traversing arc (i, j) , $i, j \in \mathcal{P} \cup \{0\}$. Similar variables are used for the locker routes, namely binary variables p_{ij}^k equal to one if and only if vehicle $k \in \mathcal{M}$ travels from node i to j , $i, j \in \mathcal{L} \cup \{0\}$, q_i^k equal to one if and only if vehicle $k \in \mathcal{M}$ visits node $i \in \mathcal{L} \cup \{0\}$, and integer variables g_{ij}^k which reflect the load on vehicle $k \in \mathcal{M}$ while traversing arc (i, j) , $i, j \in \mathcal{L} \cup \{0\}$. The problem can be formulated as follows:

$$\min \sum_{i \in \mathcal{P} \cup \{0\}} \sum_{j \in \mathcal{P} \cup \{0\}} \sum_{k \in \mathcal{K}} \phi c_{ij} x_{ij}^k + \sum_{i \in \mathcal{L} \cup \{0\}} \sum_{j \in \mathcal{L} \cup \{0\}} \sum_{k \in \mathcal{M}} c_{ij} p_{ij}^k + \sum_{i \in \mathcal{L}} F_i v_i \tag{1}$$

$$\text{subject to} \quad \sum_{j \in \mathcal{L}} u_{ij} = z_i \quad i \in \mathcal{P} \tag{2}$$

$$w_i + z_i = 1 \quad i \in \mathcal{P} \tag{3}$$

$$u_{ij} \leq v_j \quad i \in \mathcal{P}, j \in \mathcal{L} \tag{4}$$

$$u_{ij} d_{ij} \leq r_j v_j \quad i \in \mathcal{P}, j \in \mathcal{L} \tag{5}$$

$$r_j v_j \leq d_{ij} + z_i M \quad i \in \mathcal{P}, j \in \mathcal{L} \tag{6}$$

$$\sum_{j \in \mathcal{P} \cup \{0\}} x_{ij}^k + \sum_{j \in \mathcal{P} \cup \{0\}} x_{ji}^k = 2y_i^k \quad i \in \mathcal{P}, k \in \mathcal{K} \tag{7}$$

$$\sum_{j \in \mathcal{P}} x_{0j}^k + \sum_{j \in \mathcal{P}} x_{j0}^k = 2y_0^k \quad k \in \mathcal{K} \tag{8}$$

$$\sum_{k \in \mathcal{K}} y_i^k = w_i \quad i \in \mathcal{P} \tag{9}$$

$$\sum_{j \in \mathcal{P} \cup \{0\}} x_{ij}^k = \sum_{j \in \mathcal{P} \cup \{0\}} x_{ji}^k \quad i \in \mathcal{P}, k \in \mathcal{K} \tag{10}$$

$$\sum_{j \in \mathcal{P}} \sum_{k \in \mathcal{K}} h_{0j}^k = \sum_{i \in \mathcal{P}} w_i \tag{11}$$

$$\sum_{i \in \mathcal{P} \cup \{0\}} h_{ij}^k - \sum_{i \in \mathcal{P} \cup \{0\}} h_{ji}^k = y_j^k \quad j \in \mathcal{P}, k \in \mathcal{K} \tag{12}$$

$$h_{ij}^k \leq (|P| - 1) x_{ij}^k \quad i \in P, j \in P \cup \{0\}, k \in K \tag{13}$$

$$h_{0j}^k \leq |P| x_{0j}^k \quad j \in P \cup \{0\}, k \in K \tag{14}$$

$$\sum_{i \in \mathcal{P} \cup \{0\}} \sum_{j \in \mathcal{P} \cup \{0\}} t_{ij} x_{ij}^k + \sum_{i \in \mathcal{P} \cup \{0\}} \sum_{j \in \mathcal{P} \cup \{0\}} s_j x_{ij}^k \leq T_1 \quad k \in \mathcal{K} \tag{15}$$

$$\sum_{j \in \mathcal{L} \cup \{0\}} p_{ij}^k + \sum_{j \in \mathcal{L} \cup \{0\}} p_{ji}^k = 2q_i^k \quad i \in \mathcal{L}, k \in \mathcal{M} \tag{16}$$

$$\sum_{j \in \mathcal{L}} p_{0j}^k + \sum_{j \in \mathcal{L}} p_{j0}^k = 2q_0^k \quad k \in \mathcal{M} \tag{17}$$

$$\sum_{k \in \mathcal{M}} q_i^k = v_i \quad i \in \mathcal{L} \tag{18}$$

$$\sum_{j \in \mathcal{L} \cup \{0\}} p_{ij}^k = \sum_{j \in \mathcal{L} \cup \{0\}} p_{ji}^k \quad i \in \mathcal{L}, k \in \mathcal{M} \tag{19}$$

$$\sum_{j \in \mathcal{L}} \sum_{k \in \mathcal{M}} g_{0j}^k = \sum_{i \in \mathcal{L}} v_i \tag{20}$$

$$\sum_{i \in \mathcal{L} \cup \{0\}} g_{ij}^k - \sum_{i \in \mathcal{L} \cup \{0\}} g_{ji}^k = q_j^k \quad j \in \mathcal{L}, k \in \mathcal{M} \tag{21}$$

$$g_{ij}^k \leq (|L| - 1) p_{ij}^k \quad i \in L, j \in L \cup \{0\}, k \in M \tag{22}$$

$$g_{0j}^k \leq |L| p_{0j}^k \quad j \in L \cup \{0\}, k \in M \tag{23}$$

$$\sum_{i \in \mathcal{L} \cup \{0\}} \sum_{j \in \mathcal{L} \cup \{0\}} t_{ij} p_{ij}^k + \sum_{i \in \mathcal{L} \cup \{0\}} \sum_{j \in \mathcal{L} \cup \{0\}} s_j p_{ij}^k \leq T_2 \quad k \in \mathcal{M} \tag{24}$$

$$z_i, w_i \in \{0, 1\} \quad i \in \mathcal{P} \tag{25}$$

$$u_{ij} \in \{0, 1\} \quad i \in \mathcal{P}, j \in \mathcal{L} \quad (26) \quad \sum_{k \in \mathcal{K}} x_{0i}^k \leq \sum_{k \in \mathcal{K}} y_i^k \quad i \in \mathcal{P} \quad (37)$$

$$v_i \in \{0, 1\} \quad i \in \mathcal{L} \quad (27) \quad x_{ij}^k + x_{ji}^k \leq 1 \quad i, j \in \mathcal{P}, k \in \mathcal{K} \quad (38)$$

$$x_{ij}^k \in \{0, 1\} \quad i, j \in \mathcal{P} \cup \{0\}, k \in \mathcal{K} \quad (28) \quad g_{i0}^k = 0 \quad i \in \mathcal{L}, k \in \mathcal{M} \quad (39)$$

$$y_i^k \in \{0, 1\} \quad i \in \mathcal{P} \cup \{0\}, k \in \mathcal{K} \quad (29) \quad g_{ii}^k = 0 \quad i \in \mathcal{L} \cup \{0\}, k \in \mathcal{M} \quad (40)$$

$$h_{ij}^k \geq 0 \quad i, j \in \mathcal{P} \cup \{0\}, k \in \mathcal{K} \quad (30) \quad \sum_{k \in \mathcal{M}} p_{i0}^k \leq \sum_{k \in \mathcal{M}} q_i^k \quad i \in \mathcal{L} \quad (41)$$

$$p_{ij}^k \in \{0, 1\} \quad i, j \in \mathcal{L} \cup \{0\}, k \in \mathcal{M} \quad (31) \quad \sum_{k \in \mathcal{M}} p_{0i}^k \leq \sum_{k \in \mathcal{M}} q_i^k \quad i \in \mathcal{L} \quad (42)$$

$$q_i^k \in \{0, 1\} \quad i \in \mathcal{L} \cup \{0\}, k \in \mathcal{M} \quad (32) \quad p_{ij}^k + p_{ji}^k \leq 1 \quad i, j \in \mathcal{L}, k \in \mathcal{M}. \quad (43)$$

$$g_{ij}^k \geq 0 \quad i, j \in \mathcal{L} \cup \{0\}, k \in \mathcal{M}. \quad (33)$$

The objective function (1) minimizes the total of the travel costs for the patient and the locker routes, and the costs for the opened lockers. Constraints (2)–(6) correspond to the opening of the lockers and the assignment of patients to the lockers or the depot. Constraints (2) ensure that if a patient is assigned to a locker, it is assigned to only one locker. Constraints (3) ensure that a patient is either assigned to the depot or to one of the lockers. Due to constraints (4) a patient can only be assigned to a locker if it is opened. Constraints (5) ensure that a patient can only be assigned to a locker within its coverage distance. Due to constraints (6), a patient is forced to be assigned to a locker if it is within the coverage distance of at least one opened locker. Constraints (7)–(15) correspond to the patient routes. Constraints (7) ensure that if a vehicle visits a patient, it enters and leaves its node exactly once, whereas, if the vehicle does not visit a patient, its node is not visited. Constraints (8) ensure that only vehicles that are used can visit patients. Due to constraints (9), if a patient is assigned to the depot then exactly one vehicle visits its node, otherwise no vehicle visits it. Constraints (10) are the flow conservation constraints. Constraints (11)–(14) ensure that the solution is connected. More specifically, constraint (11) ensures that the total load leaving the depot is equal to the number of patients assigned to it. Constraints (12) ensure that if a patient is assigned to a given vehicle, the difference between the load of the vehicle arriving at and leaving from its node is exactly one. Constraints (15) ensure that the route duration is respected. Constraints (16)–(24) are similar to (7)–(15) and apply for the locker routes. Constraints (25)–(33) define the nature and the range of the variables. Parameter M in constraints (6) can be set as small as possible as:

$$M = \begin{cases} \max_{j \in \mathcal{L}} r_j & \text{if } \exists i \in \mathcal{P}, j \in \mathcal{L} \text{ s.t. } d_{ij} < r_j, \\ \max_{j \in \mathcal{L}} r_j - \min_{i \in \mathcal{P}, j \in \mathcal{L} | d_{ij} < r_j} d_{ij} & \text{otherwise.} \end{cases}$$

The following valid inequalities can be imposed to strengthen the formulation:

$$h_{i0}^k = 0 \quad i \in \mathcal{P}, k \in \mathcal{K} \quad (34)$$

$$h_{ii}^k = 0 \quad i \in \mathcal{P} \cup \{0\}, k \in \mathcal{K} \quad (35)$$

$$\sum_{k \in \mathcal{K}} x_{i0}^k \leq \sum_{k \in \mathcal{K}} y_i^k \quad i \in \mathcal{P} \quad (36)$$

Constraints (34)–(38) apply for the patient routes. Constraints (34) enforce the load on a vehicle to be zero if it enters the depot, while constraints (35) enforce the load on a vehicle to equal zero for each arc $(i, i), i \in \mathcal{P} \cup \{0\}$. Constraints (36) and (37) impose that if a patient is not visited by a vehicle, the arc from the patient toward the depot and the arc from the depot toward the patient, respectively, are not traversed by a vehicle. Constraints (38) exclude an arc between two patients to be traversed by a vehicle if its opposite arc is traversed by that vehicle. Constraints (39)–(43) are similar to (34)–(38) and apply for the locker routes.

3. Solution algorithms

We now describe the solution algorithms used to solve the problem at hand. First, in Section 3.1 we describe a hybrid VNS heuristic. Then, we also propose a streamlined version of the VNS for benchmark purposes is presented in Section 3.2.

3.1. Hybrid VNS heuristic

In this section, we describe the hybrid heuristic we propose to simultaneously solve the facility location and vehicle routing problem as formalized in Section 2. The general idea behind this heuristic is that it iteratively modifies a solution by changing the set of opened lockers, updates the routes accordingly and applies a variable neighborhood search (VNS) procedure to improve the routes. The structure of the heuristic is given in Algorithm 1, which belongs to the class of hierarchical class of location-routing

```

1 Construct initial solution  $s$ ;
2 while stopping criterion is not met do
3    $s' \leftarrow s$ ;
4   Change the set of opened lockers in  $s'$ ;
5   Update the routes in  $s'$ ;
6   Apply the VNS Procedure to the routes in  $s'$ ;
7   if Accept( $s', s$ ) then
8      $s \leftarrow s'$ ;
9   end
10 end
Result:  $s_{BEST}$ ;

```

Algorithm 1: Structure of the Hybrid Heuristic.

heuristics, as the routing solution is updated within each tentative locational move. The heuristic starts with an initial solution (line

1). While the stopping criterion is not met, the following steps are executed in each iteration. The set of opened lockers is changed (line 4). Then, the patient and locker routes are updated accordingly (line 5). A VNS procedure is applied to the patient and locker routes (line 6). The acceptance of the solution is based on simulated annealing (line 7). When the stopping criterion is met, the heuristic outputs the best found solution.

In Section 3.1.1 we describe the construction of the initial solution. Section 3.1.2 details the step in which the set of opened lockers is changed. The phase in which the routes are updated is described in Section 3.1.4. The VNS procedure is presented in Section 3.1.5. The acceptance and stopping criteria are given in Section 3.1.6. Section 3.1.7 discusses feasibility issues.

3.1.1. Initial solution

The initial solution is constructed as follows. It starts with \mathcal{M} empty locker routes and \mathcal{K} empty patient routes. Each locker $i \in \mathcal{L}$ is opened with probability ρ , $0 < \rho < 1$. The opened lockers are iteratively inserted in a random sequence. Each locker is inserted at its best position in one of the locker routes, i.e., in the position that yields the smallest increase in objective value. Let \mathcal{P} be the set of patients that are not within the covering range of any of the opened lockers. Each patient $p \in \mathcal{P}$ is inserted in one of the patient routes at its best position, where the sequence of patients is randomly determined. The VNS procedure (described in Section 3.1.5) is applied to the locker and patient routes.

3.1.2. Changing the set of opened lockers

In this step the set of opened lockers is changed based on one of the following three procedures: (1) opening a locker, (2) closing a locker, or (3) moving a locker. The three different procedures will be explained in detail in the next paragraphs. The idea of opening, closing, and moving a locker is derived from Nagy and Salhi (1996). The first procedure is applied if all lockers in the current solution are closed, whereas the second procedure is applied if all lockers in the current solution are opened. If the former two situations do not apply, either procedure 1, 2, or 3 is invoked with a probability equal to δ_1 , δ_2 , and δ_3 , respectively, where $0 < \delta_1, \delta_2, \delta_3 < 1$ and $\delta_1 + \delta_2 + \delta_3 = 1$.

Opening a locker. In this procedure one locker in the set \mathcal{C} of closed lockers is opened. Opening a locker is based on one of the following three methods: (1) random, (2) based on cost reduction, or (3) based on the number of covered patients. The probability of using methods 1, 2, and 3 is η_1 , η_2 , and η_3 , respectively, where $0 < \eta_1, \eta_2, \eta_3 < 1$ and $\eta_1 + \eta_2 + \eta_3 = 1$. Method 1 randomly selects one locker from the set \mathcal{C} . Method 2 selects a locker with a bias toward lockers with high cost reductions as follows. The cost reduction of opening currently closed locker $c \in \mathcal{C}$ is calculated as $r_c = R_s - R_c - F_c$, where R_c is the total routing costs with locker c inserted at its best position on a locker route and removing from the patient routes those that are within its coverage range; R_s is the routing costs of the current solution; and F_c is the opening cost of locker c . Let L be the array of closed lockers sorted by non-increasing cost reduction. Locker $c = L[\lceil y^{p_1} |\mathcal{C}| \rceil]$ is selected to open, where p_1 is a user-defined parameter and y is randomly drawn from the interval $[0, 1)$. Method 3 computes, for each closed locker, the number of patients that are covered by each closed locker but are not covered by any of the remaining opened lockers. Then, the locker is selected similarly to Method 2, where L is now the array of closed lockers sorted by non-increasing number of affected patients. The method uses parameter p_2 instead of p_1 .

Closing a locker. In this procedure, one locker in the set \mathcal{O} of opened lockers is closed. Closing a locker is based on one of the following three methods: (1) random, (2) based on cost reduction,

or (3) based on the number of covered patients. These methods are similar to opening a locker, where in Method 2 the cost reduction of closing a currently opened locker $o \in \mathcal{O}$ is calculated as $r_o = R_s - R_o + F_o$, where R_o is the routing costs when removing locker o and inserting the patients that are within the coverage range of locker o and not within the coverage range of any of the remaining opened lockers, where the patients are iteratively inserted at their best position, based on a random sequence. In Method 3 the array of closed lockers is sorted by non-decreasing number of patients only covered by the closed locker. The probability of using methods 1, 2, and 3 is γ_1 , γ_2 , and γ_3 , respectively, where $0 < \gamma_1, \gamma_2, \gamma_3 < 1$ and $\gamma_1 + \gamma_2 + \gamma_3 = 1$.

3.1.3. Moving a locker

Let \mathcal{O} and \mathcal{C} be the set of opened and closed lockers, respectively. With a probability equal to θ , a locker $o \in \mathcal{O}$ to close is selected first, after which a locker $c \in \mathcal{C}$ to open is randomly selected with equal probability such that locker c is within the coverage radius of locker o . With probability $1 - \theta$, the procedure first selects a locker to open and then selects a locker to close, also within its coverage radius. In both cases, if no locker exists within the coverage radius, a random locker is selected to be closed (opened).

3.1.4. Post-optimization of the routes

After the set of opened lockers is changed, the routes are updated accordingly. If a locker is closed in the current iteration it is removed from its route and the patients that are within its coverage radius and not within the coverage radius of any of the remaining opened lockers are inserted in the patient routes at their best positions, which is done iteratively in a random sequence. If a locker is opened in the current iteration it is inserted at its best position in one of the locker routes, and the patients within its coverage radius are removed from their routes. If a locker is opened and another one is closed in the current iteration, all removal operations occur before the insertion operations begin.

3.1.5. VNS procedure

The basic idea of VNS is proposed by Mladenović and Hansen (1997). In the VNS procedure different local search operators are applied to the vehicle routes. The structure of our VNS procedure is given in Algorithm 2. The procedure starts with a set of routes,

```

1 R: Set of routes;
2 improveoverall ← 1;
3 while improveoverall = 1 do
4   improveoverall ← 0;
5   for k ∈ {1, ..., 6} do
6     improve ← 1;
7     while improve = 1 do
8       improve ← 0; R' ← R;
9       R' ← Operatork(R');
10      if cost(R) > cost(R') then
11        improve ← 1; improveoverall ← 1; R ← R';
12      end
13    end
14  end
15 end

```

Algorithm 2: Structure of the VNS procedure.

i.e., all locker routes or all patient routes. The VNS procedure applies a local search operator until no improvement is found and then continues with the next operator. If all operators are applied and an improvement is found the procedure restarts, otherwise it stops. The six local search operators are used in the following

order: 1-0-Exchange, 1-1-Exchange, Intra-2-Opt, Inter-2-Opt, Intra-3-Opt, Inter-3-Opt'. In Algorithm 2, these operators are referred to as $Operator_k$, $k \in \{1, \dots, 6\}$. The 1-0-Exchange, 1-1-Exchange and k -Opt operators are commonly applied in the literature (see, e.g., Zachariadis & Kiranoudis, 2010, and Kindervater & Savelsbergh, 1997).

The 1-0-Exchange operator iteratively removes a location and reinserts it at its best position, where the sequence of locations chosen at each iteration is randomly determined.

For the 1-1-Exchange operator, the sequence in which the locations are examined is random. The position of a current location i is interchanged with the position of the location j that yields the largest decrease in objective value (if any).

The Intra-2-Opt operator is applied to each route separately. For each combination of two arcs in the same route, the operator computes the decrease in objective value when removing the arcs and reconnecting the resulting paths. If there is at least one combination of arcs that results in a decrease in objective value, the two arcs that yield the largest decrease in objective value are removed and the resulting paths are reconnected.

The Inter-2-Opt operator is similar to the Intra-2-Opt operator, but here the two arcs come from different routes.

The Intra-3-Opt operator computes the decrease in objective value when removing three arcs from the same route and reconnecting the resulting paths in a different way such that the largest decrease in objective value is obtained. If there is at least one combination of arcs that results in a decrease in objective value, the three arcs that yield the largest decrease in objective value are removed and the resulting paths are reconnected.

The Inter-3-Opt' operator is derived from the Inter-3-Opt operator and computes the decrease in objective value when removing a partial path and inserting it in the best position of another route, where the size of the partial path is larger than one. The partial path that leads to the largest decrease in objective value (if any) is inserted at the best position of another route.

3.1.6. Acceptance and stopping criteria

In each iteration, a new solution s' is accepted if its objective value $f(s')$ is smaller than the objective value $f(s)$ of the current solution s . If $f(s') \geq f(s)$, the acceptance of the new solution s' is based on a simulated annealing acceptance criterion (see Kirkpatrick, Gelatt Jr., & Vecchi, 1983), where the probability of acceptance is equal to $e^{-(f(s')-f(s))/T}$, where $T > 0$ is the temperature at the given iteration. At the first iteration, the temperature is set to $T = T_{start}$. In the subsequent iterations, the temperature decreases according to the formula $T = \kappa \cdot T$, where $0 < \kappa < 1$ is the cooling rate. The algorithm stops when it has run for N_{iter} iterations.

3.1.7. Feasibility

We allow infeasible solutions that violate the maximum route duration constraints to be accepted during the search of the routing heuristic. In order to steer the solution toward feasibility, a penalty is incurred and added to the objective value if one of the maximum route duration constraints is violated. For each route R , the penalty $p(R)$ is computed as $p(R) = p_{time} \cdot \max\{0, T_R - \bar{T}_R\}$, where p_{time} is a constant penalty for each time unit exceeding the maximum route duration, T_R is the total time of route R and \bar{T}_R is the maximum route duration of the vehicle corresponding to route R . At the end of the heuristic, the solution is checked for feasibility.

3.2. Fast VNS heuristic

In this section we describe how we derived a faster version of the VNS algorithm just described to compare the quality of the solution and the time required to obtain them.

Here, we focus on quickly changing locker locations and on obtaining routes. The idea is not to focus on improving routing decision stemming from bad location decisions. Hence, for every location decision, we use only one operator to create and modify the vehicles routes, namely the 1-0 Exchange operator. Only if a solution is accepted, the full VNS procedure is applied to it.

Algorithm 3 describes the pseudocode of this heuristic.

```

1 Construct initial solution  $s$ ;
2 while stopping criterion is not met do
3    $s' \leftarrow s$ ;
4   Change the set of opened lockers in  $s'$ ;
5   Update the routes in  $s'$ ;
6   Apply the 1-0 Exchange operator in  $s'$ ;
7   if Accept( $s', s$ ) then
8     Apply the VNS Procedure to the routes in  $s'$ ;
9      $s \leftarrow s'$ ;
10  end
11 end
Result:  $s_{BEST}$ ;

```

Algorithm 3: Structure of the Fast VNS heuristic.

4. Computational experiments

The mathematical model in Section 2 was solved using CPLEX 12.6.1, and the hybrid heuristic was coded in Java. All computations were executed on machines equipped with two Intel Westmere EP X5650 six-core processors running at 2.667 GHz, and with up to 48 GB of RAM installed per node running the Scientific Linux 6.3. In Section 4.1 we describe the parameter setting of the hybrid heuristic. In Section 4.2 we describe the two data sets used to test the solution methods. Computational results of the branch-and-bound algorithm applied to the integer formulation given in Section 2 and results of the hybrid heuristic are discussed in detail in Section 4.3.

4.1. Parameter setting

We created a set of 240 random instances to tune the parameters of the hybrid heuristic. The instances in this set contain up to 100 patient nodes and 50 potential locker locations. We started the parameter tuning by setting the parameters to values that seemed reasonable based on our experience. We iteratively tested a range of values for each parameter and updated the value of the parameter if it led to an improvement. We continued until we could not find a significant improvement for any of the parameters, evaluating the trade-off between solution quality and computational time. The final values for the parameters in the heuristic are as follows. The number of iterations N_{iter} is equal to 50000, the start temperature T_{start} is such that a solution 5% worse than the initial solution is accepted in the first iteration with probability 0.2. The cooling rate κ is set to 0.999875716 which implies that the temperature at the last iteration is equal to $0.002 \cdot T_{start}$. The probability of opening a locker in the initial solution is $\rho = 0.1$. The probability δ_1 of opening a locker, δ_2 of closing a locker, and δ_3 of moving a locker are set to 0.45, 0.5 and 0.05, respectively. The probabilities for the different methods for opening or closing a locker are given by $(\eta_1, \eta_2, \eta_3, \gamma_1, \gamma_2, \gamma_3) = (0.01, 0.94, 0.05, 0.01, 0.94, 0.05)$. The performance of the heuristic strongly depends on the choice of the values for the parameters in the set $(\delta_2, \delta_3, \eta_1, \eta_2, \eta_3, \gamma_1, \gamma_2, \gamma_3)$. The parameters p_1 and p_2 within the methods for opening or closing a locker are set to 10 and 3, respectively. The parameter θ in the procedure of moving a locker is set to 0.7. The parameter p_{time} is instance-dependent and equal to $50 \cdot (x_{max} - x_{min} + y_{max} - y_{min})$.

where x_{max} and x_{min} are the values of the largest and smallest x-coordinate, respectively, and y_{max} and y_{min} are the values of the largest and smallest y-coordinate, respectively.

4.2. Description of the data sets

We created three sets of instances to test the branch-and-bound and hybrid heuristic algorithms. The first set of instances is randomly generated, the second set of instances is based on data from AHN, and the third set of instances are modified from a set of benchmarks for the LRP (see Appendix). In the first two cases, no patients exist within the coverage radius of the depot since these patients naturally served by the pharmacy.

The first set of instances is randomly generated and consists of 117 instances. The number of potential locker locations is up to 50 and the number of patients is up to 150. For each instance the number of vehicles in the sets \mathcal{K} and \mathcal{M} are equal to 3 and 2, respectively. The feasibility and difficulty heavily depends on the choice of the parameters. We have chosen the parameters such that the instances are feasible and that the instances rightly reflect the structure of the problem faced by our industrial partner. The x-coordinates and y-coordinates of the patients and lockers are randomly generated in the range $[0, 100]$, whereas those of the depot are random in the range $[25, 75]$. The distances are computed based on the Euclidean measure and rounded to the nearest integer. We set the travel time and the travel cost equal to the travel distance. The penalty factor ϕ is set to 10. The service time is equal for all patients and is an integer drawn from the interval $[1, 5]$. The service time for a locker is then set to five times the service time of a patient. Let N_p be the nearest-neighbor heuristic cost for visiting all patients starting from and ending at the depot (without considering the maximum route duration), and let N_l be the corresponding cost for visiting all locker locations. Then, we set T_1 , the maximum route duration of the vehicles in the set \mathcal{K} , equal to the nearest integer of $2/3 \cdot N_p + s_p \cdot |\mathcal{P}|$, where s_p denotes the service time for the patients, and similarly, we set T_2 , the maximum route duration of the vehicles in the set \mathcal{M} , equal to the nearest integer of $2/3 \cdot N_l + s_l \cdot |\mathcal{L}|$, where s_l denotes the service time for the lockers. The opening cost of a locker F_i differs between the lockers and is equal to the nearest integer of $\pi \cdot N_p / |\mathcal{L}|$, where π is a random value in the range $[1, 4]$. The coverage distance r_j is the same for all potential lockers and is a random integer in the set $\{10, 11, \dots, 20\}$.

The second set of instances is derived from location data of AHN. The original data set consists of one depot location (a pharmacy) and the locations of the patients that this pharmacy has served over a period of time. We created a set of 117 instances, which we refer to as the AHN instances. For these instances, we focused on an area of 20000×20000 meters, with the depot located in the center. Within this area, we randomly selected a set of patients from the original data set. We created instances with up to 150 patients and 50 potential locker locations. The lockers are randomly located in the aforementioned area, where we ensured that at least one patient is within the coverage distance of each potential locker location. For each instance, the number of vehicles in the sets \mathcal{K} and \mathcal{M} are equal to 3 and 2, respectively, and the maximum route duration for each vehicle is equal to 4 hours. The service time of a patient equals 180 seconds and the service time of a locker equals 360 seconds. The travel cost is equal to the travel distance in meters and the travel time is computed based on an average speed of 50 kilometers per hour. The penalty factor ϕ is set to 10. The opening cost of a locker is set to 20000 and the coverage distance of a locker is set to 1000 meters.

4.3. Comparative results

We start our analyses by comparing the performance of the full VNS algorithm and of its streamlined version, as described in Section 3. These results are presented in Table 1 and show that the CPU time gain of the streamlined version of the VNS is negligible, whereas the quality of the solution deteriorates. For this reason, the remainder of the results described in the paper will focus on the full version of the hybrid VNS heuristic.

The branch-and-bound algorithm applied to the integer program given in Section 2 was run with a time limit of 7200 seconds. The hybrid heuristic was run ten times for each instance. In Tables 2 and 3, the computational results are given for the instances containing up to 100 patients in the set of random instances and the set of AHN instances, respectively. The tables contain for each instance $|\mathcal{P}|$, the number of patients and $|\mathcal{L}|$, the number of potential locker locations. For the branch-and-bound algorithm we report for each instance \bar{z} , the value of the best integer solution found, \underline{z} the best lower bound, and $Gap(\%)$, the percentage gap between \bar{z} and \underline{z} , computed as $(\bar{z} - \underline{z}) / \bar{z} \cdot 100$, and $Time(s)$, the total computation time in seconds. For the hybrid heuristic, we report for each instance $Minimum$, the minimum objective value over ten runs, $Gap(\%)$ the percentage gap between $Minimum$ and \bar{z} , computed as $(Minimum - \bar{z}) / \bar{z} \cdot 100$, $Average$, the average objective value over ten runs, $Gap(\%)$ the percentage gap between $Average$ and \bar{z} , computed as $(Average - \bar{z}) / \bar{z} \cdot 100$, and $Time(s)$, the average running time (in seconds) over the 10 runs.

For the random instances with up to 100 patients, the branch-and-bound algorithm finds an optimal solution for 46 out of 72 instances within the predefined time limit. The average gap between \bar{z} and \underline{z} is 13.24% and the average computation time is equal to 3476.8 seconds. Out of the 46 instances for which an optimal solution is proven, the hybrid heuristic finds it for 41 instances in all ten runs and it finds the optimal solution for 44 instances in at least one run. The average gap between $Minimum$ and \bar{z} is -4.18% and between $Average$ and \bar{z} is -4.16%, which shows that our heuristic is very robust and stable. This implies that the average solution found over ten runs by the hybrid heuristic improves the best solution found by the branch-and-bound algorithm within the predefined time limit on average with 4.16%. The average computation time for the hybrid solution is 77.2 seconds and is therefore significantly less than the computation time for the branch-and-bound algorithm.

For the AHN instances with up to 100 patients, the branch-and-bound algorithm finds an optimal solution for 44 out of 72 instances within the predefined time limit. The hybrid heuristic finds the optimal solution for all of these instances in at least one run and for all but one of the instances in all ten runs. The average gap between \bar{z} and \underline{z} is 9.61%. The average computation time is 3421.6 seconds. The average gap between $Minimum$ ($Average$) and \bar{z} is -3.64% (-3.63%). The average computation time for the hybrid solution is 68.9 seconds and is therefore significantly less than the computation time for the branch-and-bound algorithm.

4.4. Sensitivity analysis

In order to evaluate how the various opening and closing locker procedures affect the performance of our hybrid heuristic, we have modified it to a version we call *random heuristic*. In this version the procedure of opening and the procedure of closing a locker is only based on the random method, i.e., $\eta_3 = \gamma_3 = 1$.

We have then evaluated all instances in the sets of random and AHN instances, where the instances are grouped based on the number of potential locker locations. Table 4 reports comparative results for the hybrid heuristic and the random heuristic. For each group, the table reports $|\mathcal{L}|$, the number of

Table 1
Computational results for the random instances.

Instance	\mathcal{P}	\mathcal{L}	Fast VNS heuristic		Hybrid heuristic	
			Average	Time (s)	Average	Time (s)
R1	30	10	3884.0	10.3	3884.0	10.4
R2	30	15	2897.0	11.1	2897.0	11.1
R3	30	20	2050.0	11.2	2050.0	11.3
R4	30	25	3496.0	12.0	3496.0	12.0
R5	30	30	3841.0	14.8	3841.0	15.0
R6	30	35	2767.0	16.9	2767.0	17.0
R7	30	40	2131.1	15.9	2129.0	15.9
R8	30	45	1016.5	17.0	1012.4	17.2
R9	30	50	460.2	17.9	455.0	18.2
R10	40	10	4388.0	19.5	4388.0	19.7
R11	40	15	3874.0	19.1	3874.0	19.4
R12	40	20	3531.0	16.6	3531.0	16.7
R13	40	25	4495.0	21.4	4495.0	21.5
R14	40	30	4402.0	25.3	4399.0	25.7
R15	40	35	3967.2	26.2	3964.0	27.0
R16	40	40	2794.0	26.1	2789.0	27.5
R17	40	45	2647.1	24.0	2641.0	24.9
R18	40	50	3397.5	29.1	3390.0	29.8
R19	50	10	5260.0	31.1	5259.0	31.2
R20	50	15	4769.0	33.8	4768.0	34.0
R21	50	20	5059.5	32.1	5058.0	32.3
R22	50	25	2355.0	30.2	2351.0	30.5
R23	50	30	4044.0	37.8	4040.0	38.2
R24	50	35	3966.0	37.7	3959.0	38.3
R25	50	40	3705.0	34.8	3697.0	35.7
R26	50	45	4128.1	41.0	4120.0	42.7
R27	50	50	2722.0	37.0	2715.2	38.1
R28	60	10	6317.2	58.7	6314.5	58.9
R29	60	15	4298.5	42.1	4289.0	42.9
R30	60	20	5251.1	53.0	5245.0	54.1
R31	60	25	2557.0	39.4	2540.0	41.0
R32	60	30	3863.5	47.1	3854.0	49.0
R33	60	35	2971.4	42.6	2960.0	45.4
R34	60	40	4842.2	70.2	4830.0	71.5
R35	60	45	3574.5	53.0	3562.0	55.4
R36	60	50	2689.0	45.8	2677.0	48.1
R37	70	10	5610.0	80.1	5605.0	81.8
R38	70	15	5168.0	71.8	5157.0	73.4
R39	70	20	5456.2	72.3	5447.0	74.9
R40	70	25	4750.4	53.8	4738.0	56.6
R41	70	30	5101.0	70.2	5083.0	73.6
R42	70	35	1748.0	60.5	1732.0	63.3
R43	70	40	4066.5	70.3	4053.0	74.6
R44	70	45	3379.0	69.8	3365.0	73.2
R45	70	50	2305.5	73.4	2287.0	76.1
R46	80	10	6253.0	114.5	6244.0	118.2
R47	80	15	5184.0	86.3	5162.0	92.6
R48	80	20	6022.0	99.2	5997.0	104.2
R49	80	25	4344.0	92.8	4323.0	96.8
R50	80	30	4178.0	85.2	4155.0	90.7
R51	80	35	6205.0	119.2	6181.0	123.3
R52	80	40	4094.0	103.2	4065.0	108.4
R53	80	45	2461.0	94.5	2424.0	99.3
R54	80	50	3656.0	113.8	3623.0	118.6
R55	90	10	7458.5	170.1	7453.0	172.5
R56	90	15	5941.0	137.5	5919.0	139.0
R57	90	20	4699.0	108.6	4668.0	111.2
R58	90	25	6029.0	132.8	6003.0	136.7
R59	90	30	4658.2	129.2	4638.0	130.5
R60	90	35	4899.0	157.5	4867.0	160.0
R61	90	40	3705.0	109.8	3681.0	110.2
R62	90	45	2590.0	132.4	2553.0	133.9
R63	90	50	1608.0	120.8	1586.0	124.8
R64	100	10	7062.0	215.7	7034.0	218.8
R65	100	15	7424.0	214.8	7389.0	216.9
R66	100	20	5832.5	185.7	5790.0	186.1
R67	100	25	5761.0	177.9	5720.0	178.3
R68	100	30	3845.0	136.8	3798.0	139.1
R69	100	35	3541.0	162.1	3481.6	163.6
R70	100	40	3672.0	148.5	3621.0	149.1
R71	100	45	4690.0	210.4	4621.0	211.4
R72	100	50	946.0	144.3	873.0	146.9

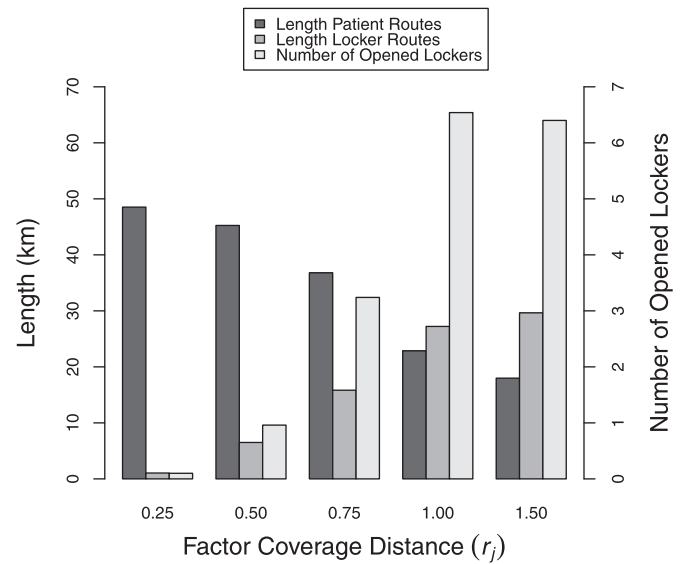


Fig. 3. The effect of changing the value for r_j .

potential locker locations, and for both the hybrid heuristic and the random heuristic it reports $Gap(\%)$, the average percentage gap to the best known solutions. For each instance, the gap is computed by $(z_i - z_{best})/z_{best} \cdot 100$, where z_i , $i \in \{hybrid, random\}$ is the average solution value over ten runs of heuristic i and z_{best} is the solution value of the best solution found over ten runs of the hybrid heuristic and ten runs of the random heuristic. The running times of the two heuristics are quite similar.

The results in Table 4 show the effect of incorporating more sophisticated methods for opening and closing lockers. The hybrid heuristic as proposed in this paper, performs on average 0.02% worse compared to the best found solutions. On the other hand, the random heuristic that only incorporates a random method for opening and closing a locker performs on average 7.18% worse compared to the best found solutions. Moreover, while the performance of the hybrid heuristic is quite stable when increasing the number of potential locker locations, the gap with respect to the best known solutions increases with an increasing number of potential locker locations for the random heuristic.

Furthermore, we have formally assessed the stability of our hybrid heuristic by calculating the *coefficient of variation* (CV), which is a dimensionless and normalized measure of dispersion of a distribution. The CV is defined as the ratio of the standard deviation (s) to the mean (\bar{X}), $CV = s/\bar{X}$. The lower the CV value, the more stable and robust the algorithm is. The CV for the AHN and random instances, computed over the 10 runs reported in the previous tables, is not larger than 0.013 for each instance, which shows how strong and stable our algorithm is.

4.5. Managerial insights

In this section, we show the effects of changing the parameter values r_j , F_j , and ϕ , which have a deep effect on the quality of service and offer many managerial insights. We have created a data set consisting of a subset of 50 instances from the AHN instance set. For these instances, we changed the values of the parameters r_j , F_j , and ϕ by multiplying them by 0.25, 0.50, 0.75, 1.00, and 1.50 for the parameters r_j and F_j , and 0.20, 0.40, 0.60, 0.80, and 1.00 for the parameter ϕ . Note that the base case corresponds to a factor of 1.00. For each instance and each parameter value, we run the heuristic ten times and use the result with the smallest objective. The results are shown in Figs. 3–5, where the average length of

Table 2
Computational results for the random instances.

Instance	P	C	Branch-and-bound				Hybrid heuristic				
			\bar{z}	\underline{z}	Gap (%)	Time (s)	Minimum	Gap (%)	Average	Gap (%)	Time (s)
R1	30	10	3884	3884.0	0.00	44.5	3884	0.00	3884.0	0.00	10.4
R2	30	15	2897	2897.0	0.00	59.8	2897	0.00	2897.0	0.00	11.1
R3	30	20	2050	2050.0	0.00	12.6	2050	0.00	2050.0	0.00	11.3
R4	30	25	3496	3496.0	0.00	100.4	3496	0.00	3496.0	0.00	12.0
R5	30	30	3841	3841.0	0.00	189.8	3841	0.00	3841.0	0.00	15.0
R6	30	35	2767	2767.0	0.00	248.7	2767	0.00	2767.0	0.00	17.0
R7	30	40	2129	2129.0	0.00	471.3	2129	0.00	2129.0	0.00	15.9
R8	30	45	1012	1012.0	0.00	162.3	1012	0.00	1012.4	0.04	17.2
R9	30	50	455	455.0	0.00	251.1	455	0.00	455.0	0.00	18.2
R10	40	10	4388	4388.0	0.00	70.5	4388	0.00	4388.0	0.00	19.7
R11	40	15	3874	3874.0	0.00	206.7	3874	0.00	3874.0	0.00	19.4
R12	40	20	3531	3531.0	0.00	102.3	3531	0.00	3531.0	0.00	16.7
R13	40	25	4495	4495.0	0.00	117.8	4495	0.00	4495.0	0.00	21.5
R14	40	30	4399	4399.0	0.00	302.1	4399	0.00	4399.0	0.00	25.7
R15	40	35	3964	3964.0	0.00	337.4	3964	0.00	3964.0	0.00	27.0
R16	40	40	2789	2789.0	0.00	466.5	2789	0.00	2789.0	0.00	27.5
R17	40	45	2641	2641.0	0.00	498.9	2641	0.00	2641.0	0.00	24.9
R18	40	50	3390	3390.0	0.00	649.9	3390	0.00	3390.0	0.00	29.8
R19	50	10	5259	5259.0	0.00	386.9	5259	0.00	5259.0	0.00	31.2
R20	50	15	4768	4768.0	0.00	265.0	4768	0.00	4768.0	0.00	34.0
R21	50	20	5058	5058.0	0.00	549.3	5058	0.00	5058.0	0.00	32.3
R22	50	25	2351	2351.0	0.00	25.9	2351	0.00	2351.0	0.00	30.5
R23	50	30	4040	4040.0	0.00	582.7	4040	0.00	4040.0	0.00	38.2
R24	50	35	3959	3959.0	0.00	566.2	3959	0.00	3959.0	0.00	38.3
R25	50	40	3697	3697.0	0.00	1045.5	3697	0.00	3697.0	0.00	35.7
R26	50	45	4120	4120.0	0.00	821.9	4120	0.00	4120.0	0.00	42.7
R27	50	50	2715	2715.0	0.00	4286.4	2715	0.00	2715.2	0.01	38.1
R28	60	10	6170	6170.0	0.00	2022.6	6245	1.22	6314.5	2.34	58.9
R29	60	15	4289	4289.0	0.00	891.9	4289	0.00	4289.0	0.00	42.9
R30	60	20	5245	5245.0	0.00	1227.9	5245	0.00	5245.0	0.00	54.1
R31	60	25	2540	2540.0	0.00	653.2	2540	0.00	2540.0	0.00	41.0
R32	60	30	3854	3854.0	0.00	1969.3	3854	0.00	3854.0	0.00	49.0
R33	60	35	2960	2960.0	0.00	1226.3	2960	0.00	2960.0	0.00	45.4
R34	60	40	4830	4830.0	0.00	1839.2	4830	0.00	4830.0	0.00	71.5
R35	60	45	3580	3290.3	8.80	7200.0	3562	-0.50	3562.0	-0.50	55.4
R36	60	50	2677	2677.0	0.00	5198.4	2677	0.00	2677.0	0.00	48.1
R37	70	10	5605	5605.0	0.00	3144.9	5605	0.00	5605.0	0.00	81.8
R38	70	15	5157	5157.0	0.00	4954.3	5157	0.00	5157.0	0.00	73.4
R39	70	20	5447	5447.0	0.00	2466.0	5447	0.00	5447.0	0.00	74.9
R40	70	25	4738	4738.0	0.00	5254.2	4738	0.00	4738.0	0.00	56.6
R41	70	30	5083	5083.0	0.00	2159.2	5083	0.00	5083.0	0.00	73.6
R42	70	35	1732	1732.0	0.00	36.9	1732	0.00	1732.0	0.00	63.3
R43	70	40	5250	3170.0	65.61	7200.0	4053	-22.80	4053.0	-22.80	74.6
R44	70	45	3767	1940.4	94.14	7200.0	3365	-10.67	3365.0	-10.67	73.2
R45	70	50	2298	2264.9	1.46	7200.0	2286	-0.52	2287.0	-0.48	76.1
R46	80	10	6244	6244.0	0.00	4068.2	6244	0.00	6244.0	0.00	118.2
R47	80	15	5472	4846.1	12.92	7200.0	5162	-5.67	5162.0	-5.67	92.6
R48	80	20	6086	5742.3	5.99	7200.0	5997	-1.46	5997.0	-1.46	104.2
R49	80	25	4323	4250.5	1.70	7200.0	4323	0.00	4323.0	0.00	96.8
R50	80	30	4145	4145.0	0.00	6707.3	4155	0.24	4155.0	0.24	90.7
R51	80	35	6226	6042.1	3.04	7200.0	6181	-0.72	6181.0	-0.72	123.3
R52	80	40	4065	3963.9	2.55	7200.0	4065	0.00	4065.0	0.00	108.4
R53	80	45	2424	2424.0	0.00	2794.1	2424	0.00	2424.0	0.00	99.3
R54	80	50	3623	3419.7	5.94	7200.0	3623	0.00	3623.0	0.00	118.6
R55	90	10	8550	7253.3	17.88	7200.0	7453	-12.83	7453.0	-12.83	172.5
R56	90	15	6962	5738.9	21.31	7200.0	5919	-14.98	5919.0	-14.98	139.0
R57	90	20	5075	4347.7	16.73	7200.0	4668	-8.02	4668.0	-8.02	111.2
R58	90	25	7301	5775.9	26.40	7200.0	6003	-17.78	6003.0	-17.78	136.7
R59	90	30	5260	3922.9	34.08	7200.0	4638	-11.83	4638.0	-11.83	130.5
R60	90	35	6093	4032.9	51.08	7200.0	4867	-20.12	4867.0	-20.12	160.0
R61	90	40	6153	2239.9	174.70	7200.0	3681	-40.18	3681.0	-40.18	110.2
R62	90	45	2593	1677.0	54.62	7200.0	2553	-1.54	2553.0	-1.54	133.9
R63	90	50	1586	1586.0	0.00	3387.5	1586	0.00	1586.0	0.00	124.8
R64	100	10	10888	6852.8	58.88	7200.0	7034	-35.40	7034.0	-35.40	218.8
R65	100	15	17775	6712.0	164.83	7200.0	7389	-58.43	7389.0	-58.43	216.9
R66	100	20	5871	5631.2	4.26	7200.0	5790	-1.38	5790.0	-1.38	186.1
R67	100	25	7510	5097.0	47.34	7200.0	5720	-23.83	5720.0	-23.83	178.3
R68	100	30	4000	3554.1	12.55	7200.0	3798	-5.05	3798.0	-5.05	139.1
R69	100	35	3477	3371.3	3.14	7200.0	3477	0.00	3481.6	0.13	163.6
R70	100	40	3633	2744.9	32.35	7200.0	3621	-0.33	3621.0	-0.33	149.1
R71	100	45	5031	3847.7	30.75	7200.0	4621	-8.15	4621.0	-8.15	211.4
R72	100	50	873	873.0	0.00	308.1	873	0.00	873.0	0.00	146.9
Average					13.24	3476.8		-4.18		-4.16	77.2

Table 3
Computational results for the AHN instances.

Instance	$ \mathcal{P} $	$ \mathcal{L} $	Branch-and-bound				Hybrid heuristic				
			\bar{z}	z	Gap (%)	Time (s)	Minimum	Gap (%)	Average	Gap (%)	Time (s)
AHN1	30	10	402805	402805.0	0.00	123.5	402805	0.00	402805.0	0.00	9.7
AHN2	30	15	320809	320809.0	0.00	43.0	320809	0.00	320809.0	0.00	10.4
AHN3	30	20	376827	376827.0	0.00	57.6	376827	0.00	376827.0	0.00	10.6
AHN4	30	25	263178	263178.0	0.00	10.6	263178	0.00	263178.0	0.00	13.6
AHN5	30	30	280986	280986.0	0.00	69.4	280986	0.00	280986.0	0.00	14.3
AHN6	30	35	217404	217404.0	0.00	7.9	217404	0.00	217404.0	0.00	14.5
AHN7	30	40	259006	259006.0	0.00	22.5	259006	0.00	259006.0	0.00	16.2
AHN8	30	45	289541	289541.0	0.00	192.2	289541	0.00	289541.0	0.00	18.9
AHN9	30	50	256455	256455.0	0.00	296.2	256455	0.00	256455.0	0.00	18.1
AHN10	40	10	443380	443380.0	0.00	265.7	443380	0.00	443380.0	0.00	16.4
AHN11	40	15	420794	420794.0	0.00	203.6	420794	0.00	420794.0	0.00	16.6
AHN12	40	20	398636	398636.0	0.00	147.0	398636	0.00	398636.0	0.00	13.8
AHN13	40	25	332717	332717.0	0.00	166.5	332717	0.00	332717.0	0.00	20.5
AHN14	40	30	295215	295215.0	0.00	141.4	295215	0.00	295215.0	0.00	20.8
AHN15	40	35	310224	310224.0	0.00	93.3	310224	0.00	310224.0	0.00	23.1
AHN16	40	40	258299	258299.0	0.00	124.3	258299	0.00	258299.0	0.00	20.3
AHN17	40	45	246563	246563.0	0.00	344.9	246563	0.00	246563.0	0.00	21.3
AHN18	40	50	255968	255968.0	0.00	594.7	255968	0.00	255968.0	0.00	26.4
AHN19	50	10	536155	536155.0	0.00	659.5	536155	0.00	536155.0	0.69	24.2
AHN20	50	15	483182	483182.0	0.00	343.6	483182	0.00	483182.0	0.00	18.4
AHN21	50	20	308951	308951.0	0.00	455.6	308951	0.00	308951.0	0.00	23.3
AHN22	50	25	427465	427465.0	0.00	192.5	427465	0.00	427465.0	0.00	26.5
AHN23	50	30	444803	440315.9	1.02	7200.0	444803	0.00	444803.0	0.00	27.2
AHN24	50	35	338337	338337.0	0.00	3451.6	338337	0.00	338337.0	0.00	26.9
AHN25	50	40	369609	369609.0	0.00	554.4	369609	0.00	369609.0	0.00	24.5
AHN26	50	45	281806	281806.0	0.00	483.6	281806	0.00	281806.0	0.00	37.9
AHN27	50	50	255322	255322.0	0.00	336.3	255322	0.00	255322.0	0.00	36.9
AHN28	60	10	510723	510723.0	0.00	786.4	510723	0.00	510723.0	0.00	47.9
AHN29	60	15	387533	387533.0	0.00	658.3	387533	0.00	387533.0	0.00	34.9
AHN30	60	20	420748	420748.0	0.00	1003.0	420748	0.00	420748.0	0.00	25.8
AHN31	60	25	526724	526724.0	0.00	3515.5	526724	0.00	526724.0	0.00	40.5
AHN32	60	30	461333	431363.1	6.95	7200.0	461333	0.00	461333.0	0.00	45.8
AHN33	60	35	324238	324238.0	0.00	411.2	324238	0.00	324238.0	0.00	48.0
AHN34	60	40	330589	330589.0	0.00	890.9	330589	0.00	330589.0	0.00	62.4
AHN35	60	45	326513	326513.0	0.00	501.3	326513	0.00	326513.0	0.00	46.5
AHN36	60	50	230152	230152.0	0.00	313.7	230152	0.00	230152.0	0.00	43.0
AHN37	70	10	554804	554804.0	0.00	1577.3	554804	0.00	554804.0	0.00	90.9
AHN38	70	15	530990	516662.6	2.77	7200.0	530990	0.00	530990.0	0.00	71.6
AHN39	70	20	542751	542751.0	0.00	3133.8	542751	0.00	542751.0	0.00	48.6
AHN40	70	25	447599	447599.0	0.00	1755.0	447599	0.00	447599.0	0.00	56.4
AHN41	70	30	494524	494524.0	0.00	2451.3	494524	0.00	494524.0	0.00	70.4
AHN42	70	35	578068	578068.0	0.00	2670.0	578068	0.00	578068.0	0.00	66.6
AHN43	70	40	561772	544812.2	3.11	7200.0	556447	-0.95	556447.0	-0.95	81.8
AHN44	70	45	380021	380021.0	0.00	1668.1	380021	0.00	380021.0	0.00	83.5
AHN45	70	50	302074	302074.0	0.00	463.6	302074	0.00	302074.0	0.00	75.5
AHN46	80	10	548988	487482.2	12.62	7200.0	547449	-0.28	547449.0	-0.28	114.7
AHN47	80	15	446764	425069.0	5.10	7200.0	446764	0.00	446764.0	0.00	123.0
AHN48	80	20	450842	432862.9	4.15	7200.0	450842	0.00	450842.0	0.00	71.3
AHN49	80	25	450048	408033.4	10.30	7200.0	426489	-5.23	426489.0	-5.23	80.5
AHN50	80	30	496847	426685.8	16.44	7200.0	480258	-3.34	480258.0	-3.34	69.6
AHN51	80	35	485028	485028.0	0.00	4725.1	485028	0.00	485028.0	0.00	105.2
AHN52	80	40	471189	471189.0	0.00	3664.5	471189	0.00	471189.0	0.00	113.3
AHN53	80	45	623723	492884.7	26.55	7200.0	549577	-11.89	549577.0	-11.89	83.8
AHN54	80	50	1126,035	478220.6	135.46	7200.0	575217	-48.92	575217.0	-48.92	114.0
AHN55	90	10	560816	516511.6	8.58	7200.0	548753	-2.15	548753.0	-2.15	137.2
AHN56	90	15	512996	376683.2	36.19	7200.0	467334	-8.90	467334.0	-8.90	116.6
AHN57	90	20	620039	588981.6	5.27	7200.0	617562	-0.40	617562.0	-0.40	103.8
AHN58	90	25	647540	546252.0	18.54	7200.0	594855	-8.14	594855.0	-8.14	95.3
AHN59	90	30	462562	417127.1	10.89	7200.0	441713	-4.51	441713.0	-4.51	123.1
AHN60	90	35	422362	383535.5	10.12	7200.0	420702	-0.39	420702.0	-0.39	116.3
AHN61	90	40	573543	518992.3	10.51	7200.0	553063	-3.57	553063.0	-3.57	113.9
AHN62	90	45	632222	595328.2	6.20	7200.0	617045	-2.40	617045.0	-2.40	134.8
AHN63	90	50	653711	470398.9	38.97	7200.0	579540	-11.35	579540.0	-11.35	163.7
AHN64	100	10	860960	592047.2	45.42	7200.0	656683	-23.73	656683.0	-23.73	204.9
AHN65	100	15	582832	539966.7	7.94	7200.0	564899	-3.08	564946.0	-3.07	172.7
AHN66	100	20	1142970	608207.6	87.92	7200.0	647355	-43.36	647355.0	-43.36	128.8
AHN67	100	25	661951	535501.9	23.61	7200.0	595560	-10.03	595560.0	-10.03	116.7
AHN68	100	30	593085	559529.2	6.00	7200.0	590941	-0.36	590941.0	-0.36	150.6
AHN69	100	35	450905	450905.0	0.00	5184.3	450905	0.00	450905.0	0.00	156.8
AHN70	100	40	934901	612546.5	52.63	7200.0	647879	-30.70	647879.0	-30.70	155.6
AHN71	100	45	716290	411011.1	74.28	7200.0	532112	-25.71	532112.0	-25.71	149.9
AHN72	100	50	589691	474491.8	24.28	7200.0	514801	-12.70	514801.0	-12.70	150.1
Average					9.61	3421.6		-3.64		-3.63	68.9

Table 4
Average gap with best known solutions over all instances.

$ \mathcal{C} $	Hybrid heuristic Gap (%)	Random heuristic Gap (%)
10	0.13	0.17
15	0.00	0.40
20	0.00	1.97
25	0.00	3.91
30	0.00	5.09
35	0.01	9.63
40	0.00	11.43
45	0.00	14.15
50	0.01	17.87
Average	0.02	7.18

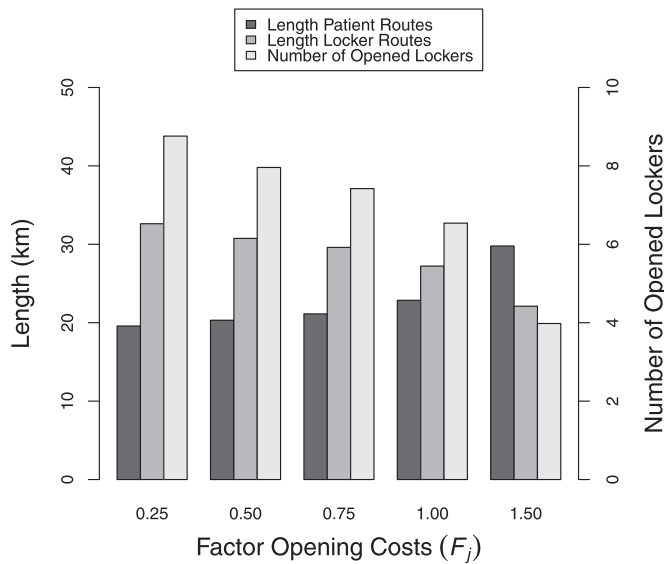


Fig. 4. The effect of changing the value for F_j .

the locker and patient routes and the average number of opened lockers over all instances are reported for the different parameter values.

New regulations may impose a change in the coverage distance. As can be seen in Fig. 3, if the coverage distance decreases, the number of opened lockers decreases and thereby the length of the locker routes decreases, whereas the length of the patient routes increases. Thus, a decrease in coverage distance reduces the number of opened lockers and thereby increases the number of home deliveries. On the other hand, an increase in coverage distance results in less home deliveries and more patients served by the lockers. This effect occurs since a larger coverage distance implies that more patients are within a locker’s range, which in turn implies that a locker is more cost-effective.

Fig. 4 shows the effects of the opening costs of the lockers. The number of opened lockers and thereby the length of the locker routes decrease when the opening costs increase, whereas the length of the patient routes increase. We see the opposite effect when the opening costs decrease. Hence, decreasing the opening costs for the lockers results in more opened lockers and thereby less home deliveries.

The effect of the value for the parameter ϕ is shown in Fig. 5. The value of ϕ reflects the resistance of a company against home delivery. The higher the value of ϕ , the higher the stimulus toward delivery through lockers. Thus, with an increasing value of ϕ , the number of opened lockers is increasing and thereby the length of

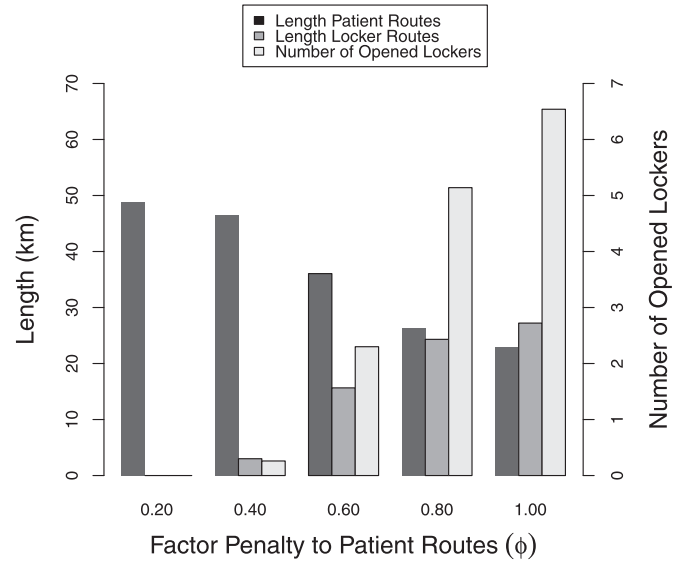


Fig. 5. The effect of changing the value for ϕ .

Table 5
Cost break down for the AHN instances.

Instance	% of cost		
	Locker location	Customer route	Locker route
AHN1	0.10	0.86	0.04
AHN2	0.31	0.59	0.10
AHN3	0.48	0.42	0.11
AHN4	0.84	0.00	0.16
AHN5	0.85	0.00	0.15
AHN5	0.85	0.00	0.15
AHN7	0.85	0.00	0.15
AHN8	0.55	0.31	0.14
AHN9	0.86	0.00	0.14
AHN10	0.05	0.95	0.00
AHN11	0.24	0.68	0.08
AHN12	0.05	0.92	0.03
AHN13	0.36	0.56	0.08
AHN14	0.61	0.27	0.12
AHN15	0.77	0.10	0.13
AHN16	0.85	0.00	0.15
AHN17	0.89	0.00	0.11
AHN18	0.86	0.00	0.14
AHN19	0.07	0.90	0.03
AHN20	0.04	0.95	0.01
AHN21	0.52	0.40	0.09
AHN22	0.14	0.80	0.06
AHN23	0.18	0.77	0.05
AHN24	0.41	0.49	0.10
AHN25	0.27	0.68	0.05
AHN26	0.85	0.00	0.15
AHN27	0.86	0.00	0.14
AHN28	0.08	0.89	0.03
AHN29	0.15	0.76	0.09
AHN30	0.05	0.92	0.03
AHN31	0.04	0.96	0.00
AHN32	0.22	0.72	0.07
AHN33	0.86	0.00	0.14
AHN34	0.85	0.00	0.15
AHN35	0.86	0.00	0.14
AHN36	0.87	0.00	0.13
AHN37	0.07	0.90	0.03
AHN38	0.08	0.89	0.03
AHN39	0.07	0.91	0.02
AHN40	0.09	0.88	0.03
AHN41	0.36	0.56	0.08
AHN42	0.21	0.76	0.03
AHN43	0.29	0.67	0.05

(continued)

Table 5 (continued)

Instance	% of cost		
	Locker location	Customer route	Locker route
AHN44	0.26	0.67	0.06
AHN45	0.86	0.00	0.14
AHN46	0.11	0.86	0.03
AHN47	0.22	0.69	0.08
AHN48	0.09	0.88	0.03
AHN49	0.09	0.89	0.01
AHN50	0.17	0.78	0.05
AHN51	0.25	0.68	0.07
AHN52	0.59	0.29	0.12
AHN53	0.18	0.78	0.03
AHN54	0.21	0.74	0.06
AHN55	0.11	0.88	0.01
AHN56	0.09	0.89	0.02
AHN57	0.10	0.89	0.02
AHN58	0.07	0.92	0.02
AHN59	0.23	0.71	0.07
AHN60	0.29	0.65	0.07
AHN61	0.25	0.70	0.05
AHN62	0.23	0.72	0.05
AHN63	0.17	0.79	0.04
AHN64	0.12	0.84	0.03
AHN65	0.11	0.86	0.03
AHN66	0.09	0.89	0.02
AHN67	0.00	1.00	0.00
AHN68	0.17	0.78	0.05
AHN69	0.44	0.46	0.10
AHN70	0.19	0.77	0.05
AHN71	0.11	0.86	0.03
AHN72	0.19	0.75	0.06
Average	0.35	0.58	0.07

the locker routes is increasing, whereas fewer patients need home delivery and thus the length of the patient routes is decreasing.

We also analyze the breakdown of costs for all AHN instances. By taking the average values over 10 runs for each instance, we compute the percentage of each cost element in comparison with the total cost. These are detailed as follows. In Table 5 we present the average cost of opening lockers, the cost of routing vehicles to customer locations, and the cost of routing vehicles to locker locations. The costs for opening lockers occur at the tactical level and constitute fixed costs. The cost for routing occur at the operational and are variable costs as these may change with changes in demand. The results in Table 5 show that the cost of locating lockers represents roughly one third of the total cost. The fact that routing costs dominate location costs is typical for this application since lockers are quite cheap compared to the facilities in other LRP ap-

plications. The general concern in LRP applications on the robustness of location decisions is therefore not pressing in our application. Furthermore, a locker could also be moved to another location after its first installation, which is not usually possible in other LRP applications. This can be done at low cost since the vast majority of the cost is in the locker system itself, not in its installation on site. We can therefore use the best available point estimate of demand for the first installation of the network, as we present in this paper. Future demand changes cannot largely distort this decision making as the configuration can be adapted at a later point in time to compensate for this. This is an important aspect to consider for AHN; if demand changes, locker locations should be reconsidered and changed accordingly.

5. Conclusions

In this paper, we have introduced, modeled, and solved a simultaneous facility location and vehicle routing problem that arises in health care logistics in the Netherlands, and can also be applied to other areas, such as the location of pick-up lockers for e-commerce operations. In this problem, the distribution of medication to a set of patients can occur via locker locations, from where patients within the coverage distance of the locker can collect their medication, or by home delivery. The goal of this integrated problem is to determine which lockers in a set of potential locker locations to open, to generate vehicle routes that visit the opened lockers and to generate the vehicle routes that visit the patients that are not within the coverage distance of one of the opened lockers, such that the total travel and locker costs are minimized. We are the first to formally define this practice-inspired problem and we give a mathematical formulation that can be solved by a branch-and-bound algorithm. Moreover, we propose a hybrid heuristic to solve the problem. We propose two sets of benchmark instances, a randomly generated set and an instance set based on data from our industrial partner AHN. We show that our branch-and-bound algorithm is able to solve instances with up to 100 patients and 50 potential locker locations to optimality within a time limit of 7200 s.

Our hybrid heuristic algorithm was able to consistently outperform CPLEX using only a fraction of the running time. Our solutions were up to 4.18% better for the random set and 3.64% better on the real AHN set (average over the best of 10 runs per instance) than the branch-and-bound solution obtained within a time limit of 7200 seconds. The heuristic could be improved by applying known computer science techniques, such as parallelism and faster programming languages. Moreover, we have shown that our heuristic is very stable, capable of yielding the same (best)

Table 6

Computational results for additional instances.

Instance	P	L	Branch-and-bound				Hybrid heuristic				
			\bar{z}	z	Gap (%)	Time (s)	Minimum	Gap (%)	Average	Gap (%)	Time (s)
Christofides69	50	4	3387	3387.0	0.00	342.9	3387	0.00	3387.0	0.00	54.0
Christofides69	75	9	3354	3354.0	0.00	1416.9	3354	0.00	3354.0	0.00	94.0
Christofides69	100	9	4072	3555.1	14.54	7200.0	3890	-4.47	3890.0	-4.47	194.6
Daskin95	88	7	460	460.0	0.00	18.4	460	0.00	460.0	0.00	86.8
Daskin95	150	9	228182	121516.2	87.78	7200.0	174948	-23.33	174948.0	-23.33	456.7
Gaskell67	21	4	2518	2518.0	0.00	6.3	2518	0.00	2518.0	0.00	7.3
Gaskell68	22	4	5400	5400.0	0.00	44.4	5400	0.00	5400.0	0.00	6.7
Gaskell69	29	4	4710	4056.1	16.12	7200.0	4710	0.00	4710.0	0.00	13.7
Gaskell70	32	4	4090	4090.0	0.00	328.6	4090	0.00	4124.0	0.83	15.8
Gaskell71	36	4	2670	2670.0	0.00	58.6	2670	0.00	2670.0	0.00	13.5
Min92	27	4	23,952	23952.0	0.00	90.3	23,952	0.00	24064.0	0.47	6.6
Min93	134	7	-	29931.3	-	7200.0	35933	-	35933.0	-	513.1
Or76	117	13	-	58066.2	-	7200.0	73323	-	73323.0	-	172.3
Perl84	55	14	2528	2528.0	0.00	1677.7	2528	0.00	2528.0	0.00	25.9
Perl85	85	6	2065	2065.0	0.00	1754.2	2065	0.00	2065.0	0.00	94.9
Average					9.11	2782.5		-2.14		-2.04	

results consistently over the 10 runs per instance. We have also shown that the incorporation of sophisticated methods for opening a locker and closing a locker has a large impact on the results of our hybrid heuristic. Results show that when the opening cost of the lockers decrease or the coverage distance of the lockers increases, the number of opened lockers increases and thereby the number of home deliveries decreases.

Acknowledgments

This project was funded by the Dutch Institute for Advanced Logistics (Dinalog) and partly funded by the Natural Sciences and Engineering Research Council of Canada (NSERC) under grant 2014-05764, Netherlands Organisation for Scientific Research under grants 040.03.021, 040.21.001, Royal Netherlands Academy of Arts and Sciences under grant 530-4CDI18, and National Natural Science Foundation of China under grants 71571125, 7151130129, 71711530046. This support is greatly acknowledged. We thank Alliance Healthcare Netherlands for providing data. We also thank an associate editor and two anonymous referees for their valuable comments on an earlier version of this paper.

Appendix

An additional set of instances are modified from a set of benchmarks for the LRP. These specify the x-coordinates and y-coordinates of the patients and the depots. For each instance, we adopted the coordinates, randomly selected one of the depots to serve as the depot for our instance and we marked the remaining depots as potential locker locations. This results in a set of 15 instances with up to 14 potential locker locations and up to 150 patients. We adopted the opening cost of a depot as the opening cost of the corresponding locker. The remaining parameters are generated similarly to the random instances. The generation of the service time and the coverage distance is slightly changed such that the proportion to the distances is the same as for the random instances.

The results for these instances are given in Table 6. The format of the table is identical to that of Tables 2 and 3. The results in this table are comparable to the results for the random and AHN instances. Since the set of potential locker location \mathcal{L} is relatively small, these instances are somewhat less interesting and are therefore incorporated in the Appendix. Unfortunately, as far as we are aware, there are no benchmark instances for simultaneous routing and location problems that consider larger sets of locker locations.

References

- Albareda-Sambola, M. (2015). Location-routing and location-arc routing. In G. Laporte, S. Nickel, & F. Saldanha da Gama (Eds.), *Location science* (pp. 399–418). Springer. chapter 15.
- Allahyari, S., Salari, M., & Vigo, D. (2015). A hybrid metaheuristic algorithm for the multi-depot covering tour vehicle routing problem. *European Journal of Operational Research*, 242(3), 756–768.
- Ávila, T., Corberán, A., Plana, I., & Sanchis, J. M. (2016). A new branch-and-cut algorithm for the generalized directed rural postman problem. *Transportation Science*, 50(2), 750–761.
- Beasley, J. E., & Nascimento, E. M. (1996). The vehicle routing-allocation problem: A unifying framework. *Top*, 4(1), 65–86.
- Cuda, R., Guastaroba, G., & Speranza, M. G. (2015). A survey on two-echelon routing problems. *Computers & Operations Research*, 55, 185–199.
- Dong, J., Yang, N., & Chen, M. (2007). Heuristic approaches for a TSP variant: The automatic meter reading shortest tour problem. In E. K. Baker, A. Joseph, A. Mehrotra, & M. A. Trick (Eds.), *Extending the horizons: Advances in computing, optimization, and decision technologies* (pp. 145–163). Springer US.
- Drexl, M. (2014). On the generalized directed rural postman problem. *Journal of the Operational Research Society*, 65(8), 1143–1154.
- Drexl, M., & Schneider, M. (2015). A survey of variants and extensions of the location-routing problem. *European Journal of Operational Research*, 241(2), 283–308.
- Gendreau, M., Laporte, G., & Semet, F. (1997). The covering tour problem. *Operations Research*, 45(4), 568–576.
- Golden, B. L., Naji-Azimi, Z., Raghavan, S., Salari, M., & Toth, P. (2012). The generalized covering salesman problem. *INFORMS Journal on Computing*, 24(4), 534–553.
- Gulczynski, D. J., Heath, J. W., & Price, C. C. (2006). The close enough traveling salesman problem: A discussion of several heuristics. In F. B. Alt, M. C. Fu, & B. L. Golden (Eds.), *Perspectives in operations research: Papers in honor of Saul Gass' 80th Birthday* (pp. 271–283). Springer US.
- Hà, M. H., Bostel, N., Langevin, A., & Rousseau, L.-M. (2013). An exact algorithm and a metaheuristic for the multi-vehicle covering tour problem with a constraint on the number of vertices. *European Journal of Operational Research*, 226(2), 211–220.
- Hà, M. H., Bostel, N., Langevin, A., & Rousseau, L.-M. (2014). Solving the close-enough arc routing problem. *Networks*, 63(1), 107–118.
- Hachicha, M., Hodgson, M. J., Laporte, G., & Semet, F. (2000). Heuristics for the multi-vehicle covering tour problem. *Computers & Operations Research*, 27(1), 29–42.
- Kindervater, G. A. P., & Savelsbergh, M. W. P. (1997). Vehicle routing: handling edge exchanges. In E. Aarts, & J. K. Lenstra (Eds.), *Local search in combinatorial optimization* (pp. 337–360). Chichester: Wiley.
- Kirkpatrick, S., Gelatt, C. D., Jr., & Vecchi, M. P. (1983). Optimization by simulated annealing. *Science*, 220, 671–680.
- Labbé, M., & Laporte, G. (1986). Maximizing user convenience and postal service efficiency in post box location. *Belgian Journal of Operations Research, Statistics and Computer Science*, 26, 21–35.
- Lopes, R. B., Ferreira, C., Santos, B. S., & Barreto, S. (2013). A taxonomical analysis, current methods and objectives on location-routing problems. *International Transactions in Operational Research*, 20(6), 795–822.
- Mennell, W. K. (2009). *Heuristics for solving three routing problems: Close-enough traveling salesman problem, close-enough vehicle routing problem, sequence-dependent team orienteering problem*. University of Maryland Ph.D. thesis.
- Mladenović, N., & Hansen, P. (1997). Variable neighborhood search. *Computers & Operations Research*, 24(11), 1097–1100.
- Nagy, G., & Salhi, S. (1996). Nested heuristic methods for the location-routing problem. *Journal of the Operational Research Society*, 47(9), 1166–1174.
- Nagy, G., & Salhi, S. (2007). Location-routing: Issues, models and methods. *European Journal of Operational Research*, 177(2), 649–672.
- Naji-Azimi, Z., Renaud, J., Ruiz, A., & Salari, M. (2012). A covering tour approach to the location of satellite distribution centers to supply humanitarian aid. *European Journal of Operational Research*, 222(3), 596–605.
- Prodhon, C., & Prins, C. (2014). A survey of recent research on location-routing problems. *European Journal of Operational Research*, 238(1), 1–17.
- Salari, M., & Naji-Azimi, Z. (2012). An integer programming-based local search for the covering salesman problem. *Computers & Operations Research*, 39(11), 2594–2602.
- Schneider, M., & Drexl, M. (2017). A survey of the standard location-routing problem. *Annals of Operations Research*, 259(1–2), 1–26.
- Solak, S., Scherrer, C., & Ghoniem, A. (2014). The stop-and-drop problem in nonprofit food distribution networks. *Annals of Operations Research*, 221(1), 407–426.
- Stenger, A., Schneider, M., Schwind, M., & Vigo, D. (2012). Location routing for small package shippers with subcontracting options. *International Journal of Production Economics*, 140(2), 702–712.
- Stenger, A., Vigo, D., Enz, S., & Schwind, M. (2013). An adaptive variable neighborhood search algorithm for a vehicle routing problem arising in small package shipping. *Transportation Science*, 47(1), 64–80.
- Tricoire, F., Graf, A., & Gutjahr, W. J. (2012). The bi-objective stochastic covering tour problem. *Computers & Operations Research*, 39(7), 1582–1592.
- Yuan, B., Orlowska, M., & Sadiq, S. (2007). On the optimal robot routing problem in wireless sensor networks. *IEEE Transactions on Knowledge and Data Engineering*, 19(9), 1252–1261.
- Zachariadis, E. E., & Kiranoudis, C. T. (2010). A strategy for reducing the computational complexity of local search-based methods for the vehicle routing problem. *Computers & Operations Research*, 37(12), 2089–2105.
- Zäpfel, G., & Bögl, M. (2008). Multi-period vehicle routing and crew scheduling with outsourcing options. *International Journal of Production Economics*, 113(2), 980–996.