



Contents lists available at ScienceDirect

## European Journal of Operational Research

journal homepage: [www.elsevier.com/locate/ejor](http://www.elsevier.com/locate/ejor)

Production, Manufacturing, Transportation and Logistics

## Minimizing late deliveries in a truck loading problem

Mario C. Vélez-Gallego<sup>a,\*</sup>, Alejandro Teran-Somohano<sup>b</sup>, Alice E. Smith<sup>b</sup><sup>a</sup> Departamento de Ingeniería de Producción, Universidad EAFIT, Medellín, Colombia<sup>b</sup> Department of Industrial and Systems Engineering, Auburn University, Auburn, AL, USA

## ARTICLE INFO

## Article history:

Received 4 February 2019

Accepted 31 March 2020

Available online xxx

## Keywords:

Logistics

Distribution

Integer programming

Supply chain management

Transportation

## ABSTRACT

In this work we address the problem of determining the optimal types of products and their quantities that should be loaded on a fleet of heterogeneous one or two level trucks so that the weighted sum of delivered products is maximized, with the weights being an exponential function of the lateness computed for each unit shipped. We propose a mixed integer linear formulation followed by a two-phase solution approach. During phase one the formulation is solved aiming at maximizing the weighted sum of products delivered, whereas for the second phase, the formulation is solved aiming at minimizing the number of trucks, while ensuring that the objective function attained in phase one is not compromised. A greedy heuristic is also proposed in order to better quantify the advantages of adopting the proposed exact approach with respect to solution quality. To assess the performance of the proposed approach we used two sets of test instances, a large set of randomly generated instances that resemble the practical application that motivated this research, and a small set of real instances provided by the company. The results of our computational experiments suggest that the proposed exact solution approach is very effective in solving realistic sized instances.

© 2020 Elsevier B.V. All rights reserved.

## 1. Introduction

The work discussed in this paper was motivated by a motorcycle assembly company that needs to determine, on a daily basis, which motorcycles to load onto a heterogeneous fleet of trucks for distribution. The motorcycles are distributed along a series of pre-determined routes with the goal of meeting the due dates of customer orders to the highest extent possible. The company sells its products through a network of dealers scattered over a wide geographical area. As these dealers place orders dynamically over time requesting one or several stock keeping unites (i.e., SKUs) in different quantities, an expected arrival date (i.e., due date) is defined for each SKU. In order to fulfill these orders, the company has defined a set of fixed routes, all starting at its centralized depot. Each dealer in the distribution network belongs to an external company, and a company may own one or several dealers. For the remainder of this paper, the companies that own one or more dealers will be referred to as customers of the motorcycle assembly company. A typical route visits several cities and includes all dealers located in the cities along its path, so that each dealer is assigned to a single route. As an example, Fig. 1 presents an instance with three routes, three customers, and 13 dealers color-coded depending on the cus-

tomers that owns these dealers. Every day a fleet of heterogeneous trucks is loaded and dispatched to deliver the products to the dealers. The trucks are assigned to the routes depending on the total demand in each route, the number of trucks available, the on-hand inventory of each SKU, and the credit availability for each customer. The latter criterion refers to the fact that customers pay for the motorcycles several days or even weeks after they arrive to the dealers. The financial department of the company has defined a credit line for each customer (i.e., a maximum dollar amount that each customer can owe to the company at any point in time), and thus, the number of motorcycles that can be delivered to the dealers of a certain customer sometimes is limited by the credit line available. This credit line varies over time depending on the payments made by the customer, and on the dollar value of the motorcycles shipped to the dealers of a given customer (note that the credit line is assigned to a customer, not to a specific dealer). Once a truck is assigned to one of the routes, it visits all cities in the route delivering products to those dealers. The fleet of trucks belongs to an external supplier that charges the motorcycle assembly company a per-unit rate that depends on the size of the SKU and the distance between the depot and the dealer to which the SKU has to be delivered. This implies that the decision on the number of SKUs to load on a given truck, and on which truck to assign to a given route, should be made based on service considerations; this is, on minimizing late deliveries. A list of the most important assumptions considered in this work follows:

\* Corresponding author.

E-mail addresses: [marvelez@eafit.edu.co](mailto:marvelez@eafit.edu.co) (M.C. Vélez-Gallego), [ateran@auburn.edu](mailto:ateran@auburn.edu) (A. Teran-Somohano), [smithae@auburn.edu](mailto:smithae@auburn.edu) (A.E. Smith).<https://doi.org/10.1016/j.ejor.2020.03.083>

0377-2217/© 2020 Elsevier B.V. All rights reserved.

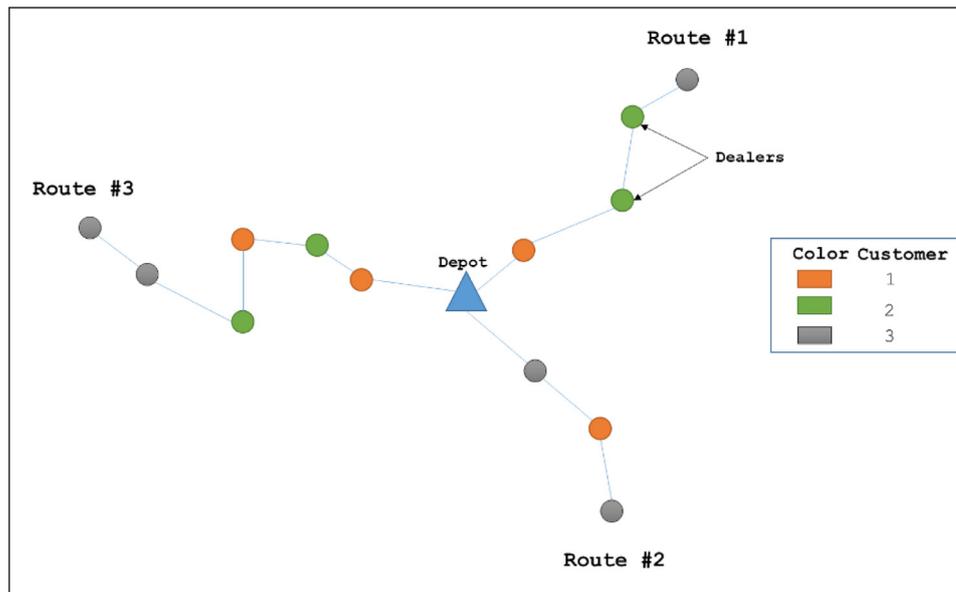


Fig. 1. Problem overview.

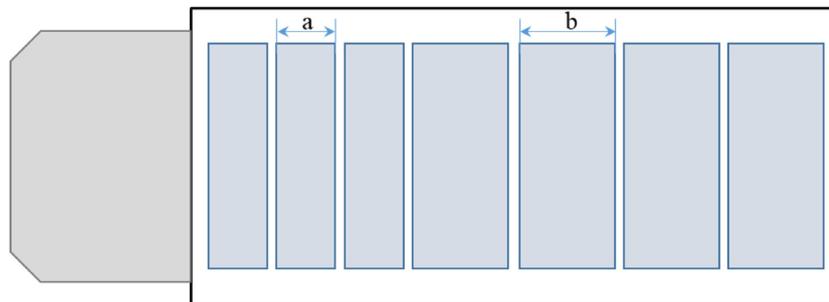


Fig. 2. Top view of a loaded truck.

- We assume that all the trucks depart from a single depot, and the time that a truck spends traveling from the depot to the first dealer, and the travel time between every pair of dealers on its route are deterministic and known in advance.
- The unloading time at the dealer's sites is negligible. As a truck requires several days to complete a route, the unloading times are negligible compared to the travel times between cities.
- The routes are assumed to be pre-defined and fixed. One reason for making this assumption is that the cities where the dealers are located are grouped in clusters, and there is only one highway that connects the depot with all the cities within the cluster, leaving no decision regarding for routing.
- A single dimension is considered for the motorcycle SKUs (i.e., the width). This assumption reflects the fact that in the problem that motivated this research, there is a single feasible orientation to load the SKUs in any truck, regardless of its capacity. As an example, Fig. 2 depicts the top view of a truck loaded with three units of an SKU of width  $a$ , and four units of another SKU of width  $b$ .
- The problem in the company that motivated this research is solved on a daily basis. We assume that it is not possible to plan ahead for several days of operation as it is not feasible to accurately forecast some of the parameters needed to solve the problem (e.g., the on-hand inventory for each SKU at the end of the day, or the number and capacity of the trucks that will become available at the end of the day, and the payments made by the customers, among others).

- Early deliveries are desirable and, thus, not penalized.

To solve this problem we propose a mixed integer linear programming (MILP) formulation followed by a two-phase solution approach. During phase one the formulation is solved aiming at maximizing the weighted sum of products delivered, whereas for the second phase, the formulation is solved aiming at minimizing the number of trucks, while ensuring that the objective function attained in phase one is not compromised. One might interpret the problem as if there is a trade-off between units shipped and number of vehicles used. However, it is also true that all SKUs must be shipped at some point in time. By shipping less units to save, say one truck, the decision maker would be deferring the shipment of these units for later, thus affecting customer service. This is the reason behind modeling the problem as a single customer service objective problem, leaving the number of trucks used in the solution as a subordinate metric. We follow the proposed two-phase approach to ensure that the optimum of customer service is attained while using the least number of trucks. The model ensures that the trucks are used efficiently is by means of the constraint that imposes a minimum capacity usage on the trucks. A greedy heuristic is also proposed as a means for assessing the impact of implementing the proposed exact approach. The remainder of this paper is organized as follows. A literature review is presented in Section 2. A formal description of the problem along with a mixed integer linear formulation for the problem, and a proposed two-phase solution approach is described in Section 3. A

greedy heuristic for solving the problem is described in Section 4, whereas a summary of the results of a computational experiment carried out to assess the effectiveness of the proposed approach is presented in Section 5. Conclusions are presented in Section 6.

## 2. Literature review

The problem addressed in this work is as a variant of the so-called vehicle loading problem (VLP), which is in turn related to two types of well-known logistics problems: (1) the container loading problem, and (2) the vehicle routing problem.

### 2.1. The container loading problem

The container loading problem (CLP) consists in determining the optimal combination and placement of items (the cargo) inside one or more larger containers, with respect to some objective. In general, the cargo items are restricted to rectangular shapes (Bortfeldt & Wäscher, 2013) as it is common to pack items within boxes or pallets before loading. However, some work has been done in loading non-rectangular but regularly shaped items (Fraser & George, 1994; Stoyan & Yaskov, 2012), as well as irregularly shaped items such as furniture (Egeblad, Garavelli, Lisi, & Pisinger, 2010) and automobiles (Liu, Smith, & Qian, 2016).

The primary type of constraint is that imposed by the container's capacity, either in terms of volume (Brown, Ellis, Graves, & Ronen, 1987; Yüceer & Özakça, 2010) or weight (Dereli & Das, 2010), or its dimensions (Che, Huang, Lim, & Zhu, 2011; Respen & Zufferey, 2017; Romão, dos Santos, & Arroyo, 2012; Tian, Zhu, Lim, & Wei, 2016). Other constraints include restrictions on weight distribution (Makarem & Haraty, 2010), on cargo orientation (Bortfeldt, Gehring, & Mack, 2003), loading order and priority (Ren, Tian, & Sawaragi, 2011; Respen & Zufferey, 2017; Tian et al., 2016), load stability and balance (Alonso, Alvarez-Valdes, Iori, & Parreño, 2019; Ramos, Silva, & Oliveira, 2018), among others. The problem can be formulated for a single container or truck (Bortfeldt et al., 2003; Respen & Zufferey, 2017; Yüceer & Özakça, 2010), or for several (Che et al., 2011; Liu et al., 2016; Toffolo, Esprit, Wauters, & Berghe, 2017). In the latter case, containers can be either homogeneous (Thapatsuwan, Pongcharoen, Hicks, & Chainate, 2012) or heterogeneous (Che et al., 2011; Liu et al., 2016). Some of the objectives found in the literature include maximizing capacity utilization (Chunyu, Jinying, & Xiaobo, 2010; Dereli & Das, 2010; Ramos et al., 2018), minimizing cost (Che et al., 2011; Tian et al., 2016), minimizing the number of containers required (Toffolo et al., 2017), maximizing profit (Liu et al., 2016), maximizing space utilization, (Che et al., 2011; Tian et al., 2016), and maximizing replenishment time (Yüceer & Özakça, 2010). Since the CLP is an NP-hard problem, exact methods are limited in their application (Respen & Zufferey, 2017), especially when realistic constraints are included in the model. Hence, heuristics (Egeblad et al., 2010; Toffolo et al., 2017), meta-heuristics (Bortfeldt et al., 2003), and hybrid methods (Chunyu et al., 2010; Dereli & Das, 2010; Romão et al., 2012) are the most common means for solving it. A detailed review of the literature concerning the CLP can be found in Bortfeldt and Wäscher (2013). A review covering 3D loading problems in particular can be found in Zhao, Bennell, Bektać, and Dowsland (2016).

### 2.2. The vehicle routing problem

The vehicle routing problem (VRP) seeks to find the optimal set of routes used by a fleet of vehicles to satisfy the needs of a series of customers. The problem is commonly modeled as a network construction problem, where nodes can represent either depots (starting points) or delivery sites (terminal points)

(Braekers, Ramaekers, & Nieuwenhuyse, 2016). The model determines which edges should be present in the network. Many real-life complexities have been incorporated into the problem such as time-dependent travel times (Balseiro, Loiseau, & Ramonet, 2011), time windows for delivery (Braaten, Gjønnnes, Hvattum, & Tirado, 2017; Campelo, Neves-Moreira, Amorim, & Almada-Lobo, 2018; Cornillier, Boctor, & Renaud, 2012), deterministic vs. stochastic demands, fleets of homogenous (Kim, Kim, & Shim, 2002) or heterogeneous (Jiang, Ng, Poh, & Teo, 2014; Leung, Zhang, Zhang, Hua, & Lim, 2013) vehicles, delivery and pick up sites (Zachariadis, Tarantilis, & Kiranoudis, 2016), and so on. The most common objective is the minimization of transportation costs (Hokama, Miyazawa, & Xavier, 2016; Kim et al., 2002; Leung et al., 2013; Reil, Bortfeldt, & Mönch, 2018; Ronen & Goodhart, 2008) or distances (Campelo et al., 2018), though other objectives have been used such as maximizing revenue (Cornillier et al., 2012), number of customers visited (Jiang et al., 2014), as well as other factors, such as inventory holding costs (Kim & Kim, 2000). Like the CLP, different solution methods have been used to solve the VRP, with meta-heuristics being the dominant ones. See Braekers et al. (2016) for a recent survey of the available literature.

VRP formulations are increasingly taking loading restrictions into consideration, thus creating a hybrid of the VRP with the CLP. The same objectives and constraints used for the CLP are included in these formulations (Leung et al., 2013; Reil et al., 2018; Wei, Zhang, Zhang, & Lim, 2015; Zachariadis et al., 2016). A valuable review of the literature on VRP with loading constraints can be found in Pollaris, Braekers, Caris, Janssens, and Limbourg (2015).

### 2.3. The truck loading problem

The work we present in this paper belongs to a class of problems, hereafter referred to as the truck loading problem (TLP), that do not fall into either the CLP or the VRP as presented above. Within this class we include problems that unlike the VRP do not seek to design the routes to be followed by the trucks, and do not fall into the category of CLP, as loading containers (i.e. trucks) without exceeding some space or weight constraints is only one of the several dimensions the problem considers. In the earliest work we are aware of in this area, Milosavljević, Teodorović, Papić, and Pavković (1996) proposed a model based on fuzzy logic to assign a fleet of heterogeneous trucks to a set of transportation requests. A similar theme was followed by Brown and Ronen (1997), who focused on selecting whether orders should be consolidated into trucks for shipping without concern for routes or specific loading patterns. The same problem addressed in Milosavljević et al. (1996) was considered by Vukadinović, Teodorović, and Pavković (1999), who proposed a neuro-fuzzy approach to tackle the problem. Cordeau, Dell'Amico, Falavigna, and Iori (2015) solve a routing problem for deliveries to automotive dealers by considering distance traveled, fixed costs and service costs using a heterogeneous truck fleet. They constrain the solutions based on restrictions on loading the vehicles onto the trucks and also penalize solutions with late deliveries. This is done over a planning horizon of several days and is solved with a heuristic approach. To the best of our knowledge, the works in the literature that are closest to our work are Yüceer and Özakça (2010), and Liu et al. (2016). In Yüceer and Özakça (2010), the authors address the problem of determining which SKUs should be loaded onto a single truck and in what quantity so that the replenishment time of each product at each destination is maximized. This work considers a single truck divided into several compartments, all of them with a different capacity in terms of space; and several destination points along a single fixed route. The work addressed by Liu et al. (2016) considers a heterogeneous fleet of trucks that needs to be assigned to a single destination point with the objective

**Table 1**  
Main differences to related works in the literature.

	Yüceer and Özakça (2010)	Liu et al. (2016)	Present work
Fleet	Single truck	Heterogeneous	Heterogeneous
Truck levels	Single	Multiple	Multiple
Number of trucks used	Not applicable	Not considered	Considered as a subordinate metric
Routes	Single	Single	Multiple
Packing dimensions	One	Two	One
Demand due dates	Not considered	Not considered	Considered
On-hand inventory	Not considered	Considered	Considered
Objective	Replenishment time	Profit	Timely delivery of products

maximizing the total profit. This work devotes a fair amount of attention to the geometry of both the transporting trucks, which may have one or two levels, and the product being transported, which are automobiles of different geometries. In our work, the objective pursued is the timely delivery of motorcycles to dealers, with no regard to transportation costs, with the number of trucks used in the solution treated only as a subordinate metric. The proposed approach in our work is novel in considering multiple trucks and routes simultaneously, along with considering both on-hand inventory and customer credit lines as part of the decision-making process. A summary of the most relevant differences between these two works and the present work is presented in Table 1.

**3. Mathematical formulation**

Formally, we are given a set of distributors or dealers scattered over a wide geographical area. Each dealer has been previously assigned to one of several routes that are periodically visited by a fleet of trucks in order to fulfill orders placed by the dealers along the route. We let  $S$  be the set of SKUs,  $R$  the set of fixed routes and  $C$  the set of customers. Each dealer belongs to a customer, and it is possible that several dealers, even if assigned to different routes, belong to the same customer. We let  $E_{rc}$  be the set of dealers along route  $r \in R$  that belong to customer  $c \in C$ . We let  $T$  be the set of trucks and  $L_t$  the set of levels or decks of truck  $t \in T$ . To solve the problem above, the following mixed integer linear programming formulation is developed.

Sets

- $S$  SKU's
- $R$  Routes
- $C$  Customers
- $E_{rc}$  Dealers along route  $r \in R$  that belong to customer  $c \in C$
- $T$  Trucks
- $L_t$  Levels of truck  $t \in T$

Parameters

- $q_{tl}$  Capacity of the  $l$ th level of truck  $t \in T$
- $\rho_t$  Fraction of the capacity of truck  $t \in T$  that must be used if assigned to a route
- $I_i$  Inventory on hand of SKU  $i \in S$
- $p_i$  Price of SKU  $i \in S$
- $f_i$  Size of SKU  $i \in S$
- $\mu_c$  Maximum amount of money that can be assigned to customer  $c \in C$
- $d_{irk}$  Demand of SKU  $i \in S$  by the  $k$ th dealer along route  $r \in R$
- $e_{rk}$  Time required by a truck to reach the  $k$ th dealer along route  $r \in R$
- $\tau_{irk}$  Time before the demand of SKU  $i \in S$  at the  $k$ th dealer along route  $r \in R$  is due
- $\lambda$  A real number strictly greater than one (i.e.  $\lambda > 1$ )

Decision variables

$x_{irtl}$  := Units of SKU  $i \in S$  to be loaded onto the  $l$  – th level of truck  $t \in T$  for delivery along route  $r \in R$

$z_{irk}$  := Units of SKU  $i \in S$  to be delivered to the  $k$  – th dealer along route  $r \in R$

$y_{rt}$  :=  $\begin{cases} 1 & \text{If truck } t \in T \text{ is assigned to route } r \in R \\ 0 & \text{Otherwise} \end{cases}$

Objective functions

$$\text{Maximize } \sum_{i \in S} \sum_{r \in R} \sum_{c \in C} \sum_{k \in E_{rc}} \lambda^{e_{rk} - \tau_{irk}} \cdot z_{irk} \tag{1}$$

$$\text{Minimize } \sum_{r \in R} \sum_{t \in T} y_{rt} \tag{2}$$

S.t:

$$\sum_{i \in S} \sum_{l \in L_t} f_i \cdot x_{irtl} \leq q_{tl} \cdot y_{rt} \quad \forall r \in R, t \in T, l \in L_t \tag{3}$$

$$\sum_{i \in S} f_i \cdot x_{irtl} \geq \sum_{l \in L_t} \rho_t \cdot q_{tl} \cdot y_{rt} \quad \forall r \in R, t \in T \tag{4}$$

$$\sum_{t \in T} \sum_{l \in L_t} x_{irtl} \leq \sum_{c \in C} \sum_{k \in E_{rc}} d_{irk} \quad \forall i \in S, r \in R \tag{5}$$

$$z_{irk} \leq d_{irk} \quad \forall i \in S, r \in R, c \in C, k \in E_{rc} \tag{6}$$

$$\sum_{t \in T} \sum_{l \in L_t} x_{irtl} = \sum_{c \in C} \sum_{k \in E_{rc}} z_{irk} \quad \forall i \in S, r \in R \tag{7}$$

$$\sum_{r \in R} \sum_{t \in T} \sum_{l \in L_t} x_{irtl} \leq I_i \quad \forall i \in S \tag{8}$$

$$\sum_{i \in S} \sum_{r \in R} \sum_{k \in E_{rc}} p_i \cdot z_{irk} \leq \mu_c \quad \forall c \in C \tag{9}$$

$$\sum_{r \in R} y_{rt} \leq 1 \quad \forall t \in T \tag{10}$$

$$y_{rt} \in \{0, 1\} \quad \forall r \in R, t \in T \tag{11}$$

$$x_{irtl} \in \mathbb{Z}^+ \quad \forall i \in S, r \in R, t \in T, l \in L_t \tag{12}$$

$$z_{irk} \in \mathbb{Z}^+ \quad \forall i \in S, r \in R, c \in C, k \in E_{rc} \tag{13}$$

The primary objective function in expression (1) aims at maximizing the weighted sum of the number of SKUs shipped to the dealers, with the weights being an exponential function of the lateness associated with each SKU. The reason for using an exponential function is twofold: first, it ensures that the weights are always positive, regardless if the lateness is positive or negative;

and second, it assigns significantly higher weights to SKUs with larger lateness values. A secondary objective function in expression (2) aims at minimizing the number of trucks sent to distribute the product, and it is used to ensure that the number of trucks used while maximizing expression (1) is kept at its minimum. Constraints in expression (3) limit the total product being loaded to a level of a truck to its capacity, whereas constraints in expression (4) ensure that if a truck is to be assigned to a route, it has to be loaded to a certain fraction of its capacity. These constraints prevent the model from sending a truck to a route with only a small fraction of its capacity being used. The constraints in expression (5) limit the number of SKUs loaded in a truck to the total demand of such SKU in the route to which the truck was assigned to, whereas expression (6) limits the number of units of a given SKU delivered to a particular dealer to the units demanded by such dealer. Expression (7) is a balance equation needed so that all the SKUs loaded on a truck are to be delivered to one or more dealers in the route. Expression (8) limits the number of units shipped of a given SKU in all the trucks to the on-hand inventory available for that SKU, whereas constraints in expression (9) impose a limit on the dollar amount released to a given customer. Finally, the constraints in expression (10) assure that a truck can only be assigned to at most one route, while expressions (11), (12) and (13) define the domains of the decision variables.

A two-phase solution approach

In order to solve an instance of the problem at hand, our proposed two-phase solution approach works as follows. First, the MILP formulation is solved using expression (1) as the objective function. Once a solution for this first phase is reached with an objective function value of say,  $\alpha^*$ ; phase two consists of solving the MILP formulation again, this time using expression (2) as the objective function. In the latter phase, besides the set of constraints described earlier in this section, the constraint in expression (14) is included in the formulation. The rationale behind using this two-phase approach is to guarantee that the objective function in expression (1) is maximized using the least number of trucks possible.

$$\sum_{i \in S} \sum_{j \in J} \sum_{c \in C} \sum_{k \in E_{jc}} \lambda^{e_{jk} - \tau_{ijk}} \cdot z_{ijk} \geq \alpha^* \tag{14}$$

4. A greedy heuristic

The problem that motivated this research is currently solved on a daily basis following a trial-and-error approach implemented on a spreadsheet. The process of finding a feasible solution requires the efforts of three people and takes no less than three hours. The heuristic described below is far more sophisticated than the current solution approach, and thus, will be used later in this work as an upper bound on the performance of the current spreadsheet approach. This implies that the savings of using our two-phase approach when compared to the greedy heuristic described below should be regarded as a conservative metric. To describe the greedy heuristic we use the same notation defined to describe the MILP formulation in Section 3. Some additional notation is now introduced. We let  $W_r$ , computed as in expression (15), be the value of objective function (1) attributable to route  $r$  if all the unassigned SKUs in the route are shipped. In a similar fashion, the total truck space required to ship all the demand in route  $r$  (i.e.,  $S_r$ ) yet to be assigned to a truck, is defined as in expression (16). In both expressions, we let  $u_{ijk}$  be an upper bound on the number of units of SKU  $i$  that can be shipped to the  $k$ -th dealer in route  $r$ , and is calculated as in expression (17), where  $Q$  is the maximum truck capacity, and  $c_f^k$  is the index of the customer to

which the  $k$ -th dealer along route  $r$  belongs to. We use expressions (15), (16), and (17) to calculate the ratio  $\gamma_r = \frac{W_r}{S_r}$  for every route  $r \in R$ . The value of  $\gamma_r$  captures the potential contribution to the objective function of route  $r$  per unit of truck space, and it is used as a means to prioritize the routes while executing the greedy heuristic: a large  $\gamma_r$  value indicates that the corresponding route is a good candidate for being assigned to a truck. The greedy heuristic is described in Algorithm 1.

Algorithm 1

```

1:  $\bar{q}_t \leftarrow q_t \forall t \in T, \text{Solution} \leftarrow \emptyset$ 
2: while True do
3:   if  $\max_{r \in R} \{\gamma_r\} = 0$  then
4:     return Solution
5:   else
6:      $\hat{r} \leftarrow \operatorname{argmax}_{r \in R} \{\gamma_r\}$ 
7:     for  $t \in T$  do
8:        $\delta_{i\hat{r}k} \leftarrow d_{i\hat{r}k} \forall i \in S, c \in C, k \in E_{\hat{r}c}$ 
9:        $\bar{\mu}_{c_f^k} \leftarrow \mu_{c_f^k} \forall c \in C, k \in E_{\hat{r}c}$ 
10:       $\bar{I}_i \leftarrow I_i \forall i \in S$ 
11:      repeat
12:         $\lambda_{i\hat{r}k}^t \leftarrow \min \left\{ \delta_{i\hat{r}k}, \bar{I}_i, \left\lfloor \frac{\bar{q}_t - \bar{q}_t}{\delta_{i\hat{r}k} \cdot f_i} \right\rfloor, \left\lfloor \frac{\bar{\mu}_{c_f^k}}{\delta_{i\hat{r}k} \cdot p_i} \right\rfloor \right\} \quad \forall i \in S, c \in C, k \in E_{\hat{r}c}$ 
13:         $[\hat{i}, \hat{k}] \leftarrow \operatorname{argmax}_{i \in S, c \in C, k \in E_{\hat{r}c}} \{2^{e_{jk} - \tau_{ijk}} \cdot \lambda_{i\hat{r}k}^t\}$ 
14:         $\delta_{i\hat{r}k} \leftarrow \delta_{i\hat{r}k} - \lambda_{i\hat{r}k}^t$ 
15:         $\bar{I}_i \leftarrow \bar{I}_i - \lambda_{i\hat{r}k}^t$ 
16:         $\bar{\mu}_{c_f^k} \leftarrow \bar{\mu}_{c_f^k} - p_i \cdot \lambda_{i\hat{r}k}^t$ 
17:         $\bar{q}_t \leftarrow \bar{q}_t - f_i \cdot \lambda_{i\hat{r}k}^t$ 
18:      until  $\lambda_{i\hat{r}k}^t = 0$ 
19:      end for
20:       $\Omega_{\hat{r}} \leftarrow \{t \in T \mid \frac{q_t - \bar{q}_t}{q_t} \geq \rho_t\}$ 
21:      if  $\|\Omega_{\hat{r}}\| = 0$  then
22:        return Solution
23:      else
24:         $\hat{t} \leftarrow \operatorname{argmax}_{t \in \Omega_{\hat{r}}} \{q_t\}$ 
25:        for all  $i \in S, c \in C, k \in E_{\hat{r}c} \mid \lambda_{i\hat{r}k}^{\hat{t}} > 0$  do
26:          Solution  $\leftarrow$  Solution  $\cup \{[\hat{t}, \hat{r}, i, k, \lambda_{i\hat{r}k}^{\hat{t}}]\}$ 
27:           $d_{i\hat{r}k} \leftarrow d_{i\hat{r}k} - \lambda_{i\hat{r}k}^{\hat{t}}$ 
28:           $I_i \leftarrow I_i - \lambda_{i\hat{r}k}^{\hat{t}}$ 
29:           $\mu_{c_f^k} \leftarrow \mu_{c_f^k} - p_i \cdot \lambda_{i\hat{r}k}^{\hat{t}}$ 
30:        end for
31:         $T \leftarrow T \setminus \{\hat{t}\}$ 
32:      end if
33:    end if
34:  end while

```

$$W_r = \sum_{i \in S} \sum_{c \in C} \sum_{k \in E_{rc}} \lambda^{e_{jk} - \tau_{irk}} \cdot u_{irk} \tag{15}$$

$$S_r = \sum_{i \in S} \sum_{c \in C} \sum_{k \in E_{rc}} f_i \cdot u_{irk} \tag{16}$$

$$u_{irk} = \min \left\{ I_i, \left\lfloor \frac{Q}{d_{irk} \cdot f_i} \right\rfloor, \left\lfloor \frac{\mu_{c_f^k}}{d_{irk} \cdot p_i} \right\rfloor \right\} \tag{17}$$

From a general point of view, the greedy heuristic repeats the following four steps:

1. Identify  $\hat{r}$ , the index of the route with the highest  $\gamma$  ratio (see line 6).

**Table 2**  
Parameters and values for test instance generation.

Parameter	Values
SKUs	[100, 200]
Routes	[10, 20]
Trucks	[5, 10, 15, 20, 25]
Customers	[5, 10, 15]

- Virtually load each available truck  $t$  by following a greedy strategy (see lines 11–18) that iteratively identifies the combination of SKU ( $\hat{i}$ ), dealer ( $\hat{k}$ ), and quantity ( $\lambda_{i\hat{k}}^t$ ) that contributes the most to the objective function (lines 12 and 13). Once identified, the algorithm loads  $\lambda_{i\hat{k}}^t$  units of SKU  $\hat{i}$  onto truck  $t$  destined for the  $\hat{k}$ th dealer in route  $\hat{r}$  (see line 12). The unfilled demand, the on-hand inventory, the credit line of the corresponding customer, and the available space in (the copy of) truck  $t$  are updated (see lines 14–17). The algorithm continues to follow this strategy until all available trucks are virtually loaded with as many units as possible.
- Define  $\Omega_{\hat{r}}$  as the set of candidate trucks that can be assigned to route  $\hat{r}$ . Here, truck  $t$  belongs to  $\Omega_{\hat{r}}$  if the truck meets the minimum truck capacity utilization constraint represented by  $\rho_t$  (see line 20).
- Identify, among the trucks in  $\Omega_{\hat{r}}$ , the index of the truck that contributes the most to the objective function if assigned to route  $\hat{r}$  (i.e.  $\hat{t}$  in line 24). Assign truck  $\hat{t}$  to route  $\hat{r}$ , and physically load it as it was loaded in the previous step (lines 25–30). Eliminate truck  $\hat{t}$  from the set of available trucks (lines 31).

This algorithm stops either because there are no feasible routes (line 4), or because there are no trucks available (line 22). The former happens when it is not possible to load a truck that meets its minimum capacity constraint and assign it to a route, or because there is not enough demand, inventory, or credit line.

## 5. Computational results

Two sets of instances were used to evaluate the performance of the proposed two-phase exact approach. A first set of 300 instances was generated randomly based on the loading operation that inspired this research, whereas a second set of five real instances, corresponding to five consecutive days of operation, was provided by the company. Four parameters were considered to generate the first set of test instances: (1) the number of SKUs, (2) the number of routes, (3) the number of trucks, and (4) the number of customers. A table with the different values considered for each parameter is presented in Table 2. For each combination of the values of these parameters (i.e. 60), five independent instances were generated, for a total of 300. The reader may refer to <https://github.com/mcvelezg/mldiatlp/> to access the test instances we generated, and the complete results of the computational experiments.

The procedure we followed to generate each instance is now explained. For each route in the instance, the number of dealers in the route was sampled from a discrete uniform distribution between 20 and 30. Now, for each dealer in the route two parameters were generated randomly: (1) the index of the customer to which the dealer belongs to, and (2) the time it takes for a truck to travel from the depot to its location. The index of the customer was sampled from a discrete uniform distribution between one and the number of customers in the instance. As per the time required for a truck to reach the dealer, we proceeded as follows. The time to reach the first dealer in the route was sampled from a discrete uniform distribution between zero and one; whereas the time to reach any other dealer was computed as the time required to reach its immediate predecessor in the route plus a

**Table 3**  
Truck sizes.

Truck size	Number of levels	Capacity Level 1	Capacity Level 2
Small	1	20	–
Medium	2	18	17
Large	2	35	30

sample from a discrete uniform distribution between zero and one. To clarify, if dealer A comes immediately before dealer B in a given route (say  $A \rightarrow B$ ), the travel time to reach B is computed as the travel time to reach A plus a non-negative number (either 0 or 1). As these travel times represent days, an additional travel time of zero means that both dealers (A and B) are in the same city and thus served within the same day, whereas an additional travel time of one (day) means that it takes the truck one day to reach B after visiting A. The maximum dollar amount that can be released to each customer (i.e., the value of the parameter  $\mu_c$ ) was sampled from a discrete uniform distribution between \$10,000 and \$20,000. For the capacity of the trucks, we considered three sizes as described in Table 3. These sizes, number of levels and level capacities correspond to the real application that motivated this research. The size of each truck in each instance was randomly chosen from this set of sizes.

The data related to the SKUs in the instance was generated as follows. For each SKU  $i \in S$ , three independent quantities were randomly generated: (1) the on-hand inventory  $I_i$ , (2) the price  $p_i$ , and (3) the size  $f_i$ . The first two parameters were sampled from a discrete uniform distribution in the intervals [1, 20] and [150, 500], respectively. As per the sizes, the values were randomly drawn from the set {1, 1.5, 2}. To generate demand at the dealer level we proceeded as follows. First, we randomly generated the number of SKUs demanded by each dealer by sampling a discrete uniform distribution between one and ten. Second, we selected at random the corresponding number of SKUs from the pool of SKUs in the instance. Once the group of SKUs demanded by a given dealer is generated, we continued to randomly sample from discrete uniform distributions two quantities for each SKU: (1) the number of units requested, and (2) the number of time units remaining until the demand is due. The former was sampled between one and ten, whereas the latter was sampled from the interval [−5, 10] to account for the fact that some demands are already overdue at the time the problem is to be solved.

The second set of instances was obtained from the company that motivated this research and corresponds to five consecutive days of operation. The solutions obtained and implemented by the decision makers were also provided by the company, and will be used later in this section for assessing the effectiveness of the proposed solution approach. The following are the characteristics of this set of instances: 10 routes, 127 SKUs, 230 dealers in total, and the number of trucks between 15 and 20. The trucks in these instances are as described in Table 3. Both sets of test instances as well as detailed results are available at <https://github.com/mcvelezg/mldiatlp/>.

### Two-phase approach

The formulation described in Section 3 was coded in Xpress Mosel Version 4.8.2 and each instance was solved using the Gurobi Optimizer 8.1.1 as the commercial solver. In both phases, the solver was allowed to run for a maximum of one hour on a machine with 64 gigabyte of memory and four Intel Core i7 processors running at 2.9 gigahertz under Windows 10 at 64 bits. For phase one, we set the relative optimality tolerance at  $10^{-3}$ , whereas for phase two we used an absolute optimality tolerance of 0.99 to take advantage of the integrality of the objective function (i.e.,

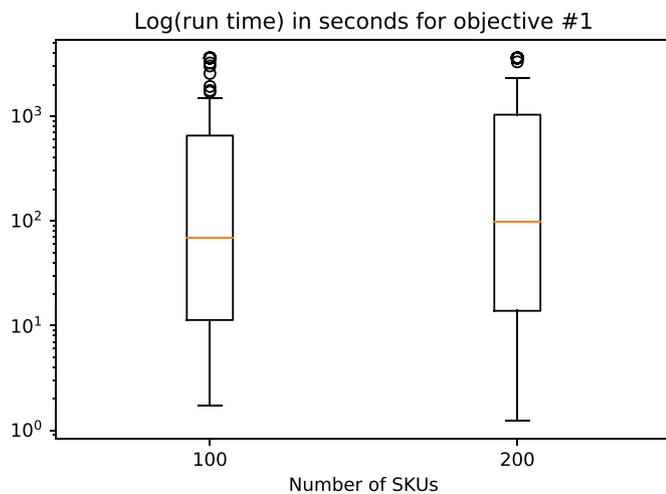


Fig. 3. Run time vs. number of SKUs.

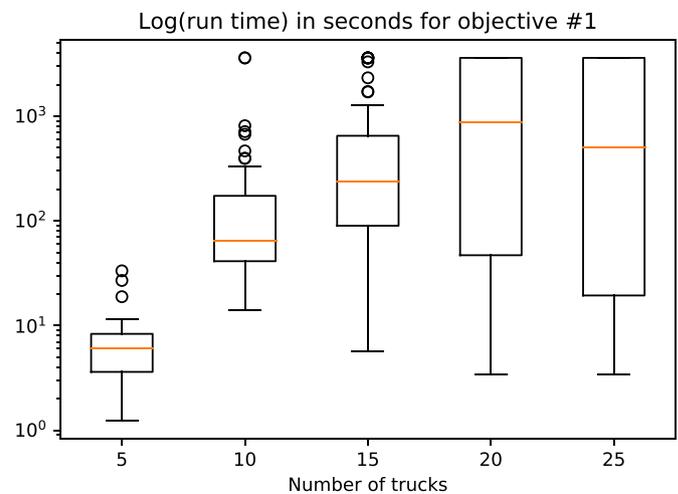


Fig. 5. Run time vs. number of vehicles.

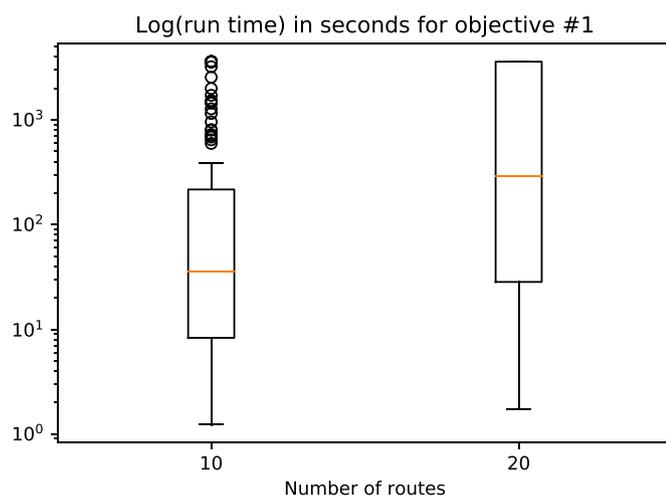


Fig. 4. Run time vs. number of routes.

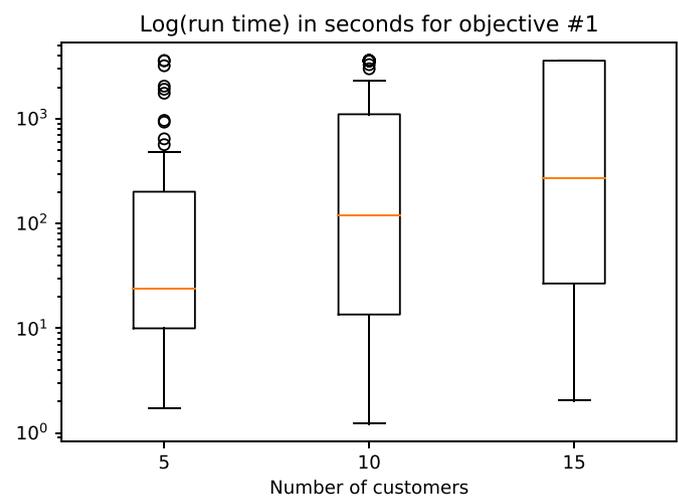


Fig. 6. Run time vs. number of customers.

number of trucks). If the solver could not find an optimal solution within the allotted time when solving phase-one formulation, the best integer solution found so far was transferred to phase two. As per phase two, if the solver could not converge to an optimal solution within the allotted time, the best integer solution was reported. In both phases, if the solver cannot prove optimality within the allotted time of one hour, the best linear relaxation was also kept for further analysis. To conduct the computational experiments the value of the parameter  $\lambda$  (see expressions (1) and (2)) was set to 2.0 for the first set of instances, and to 1.1 for the second set. The reason for using different values for this parameter is that in the real instances provided by the company there were demands greatly overdue. If a  $\lambda$  value of 2 was used in these cases, the value of the objective function would become too high and might have numerical instability issues within the solver.

After completing the computational experiments for the randomly generated set of instances, the solver was able to find 248 out of 300 optimal solutions for the phase-one formulation. For the 52 remaining instances for which optimality could not be proved, the average optimality gap after one hour was 0.0437%. Figs. 3–6 present the box-plots of the run times of phase one discriminated by the number of SKUs, routes, trucks, and customers, respectively. For these figures we used a log scale to better present the results. From Fig. 3, we can conclude that the run times seem to be unaffected by the number of SKUs in the instance. On the other hand, from Figs. 4 and 6 it is observed that as the numbers

of routes and customers increase, so does the median and the variance of the run times. Regarding the effect of the number of trucks on the run times, the box plots in Fig. 5 show that the median run times increase steadily between 5 and 20 trucks.

Once a solution to the phase-one formulation has been reached, the value of the objective function in the best solution is used as a parameter in the phase-two formulation (i.e.,  $\alpha^*$  in expression (14)). The results obtained after solving the phase-two formulation for the randomly generated instances are now presented. Here the solver was able to find an optimal solution in 274 out of 300 instances. For the 22 remaining instances, the average optimality gap was 11.6%. As in phase one, the box-plots of the run times (in log scale) required to solve the phase-two formulation, discriminated by the number of SKUs, routes, trucks, and customers, are presented in Figs. 7–10, respectively. It can be observed from the figures that the number of SKUs has no clear impact on the computational times; however, as the number of routes and trucks increase, so does the median and variance of the computational times, as shown in Figs. 8 and 9. The effect of the number of customers, on the other hand, seems to have no impact on the computational times, as seen in Fig. 10.

Fig. 11 presents the box-plots of the run times after combining (i.e., adding) the times required by both phases to terminate, discriminated by the number of trucks in the instance. Overall, the median of the combined run time was 168.96 seconds.

**Table 4**  
Results summary for real instances.

Instance	Available trucks	Company's Solution		Phase 1		Phase 2
		Objective 1	Objective 2	Objective 1	Objective 2	Objective 2
1	17	36,848.40	8	79,335.24	17	17
2	19	3175.81	4	44,531.32	16	14
3	15	3586.19	4	10,585.77	8	7
4	20	3817.48	7	9890.99	10	8
5	17	8020.23	8	13,627.78	13	11

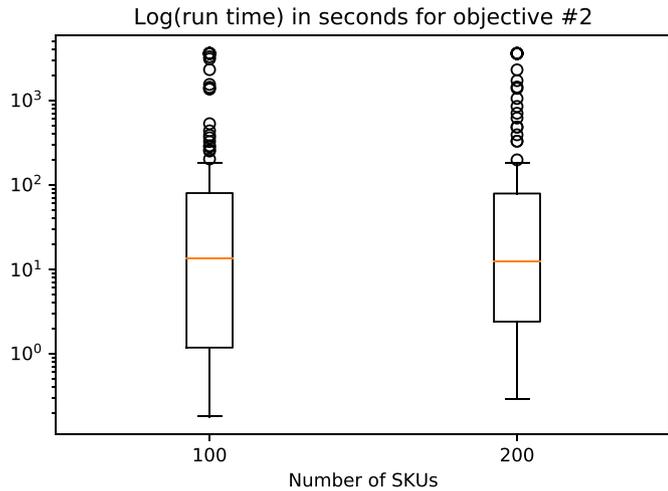


Fig. 7. Run time vs. number of SKUs.

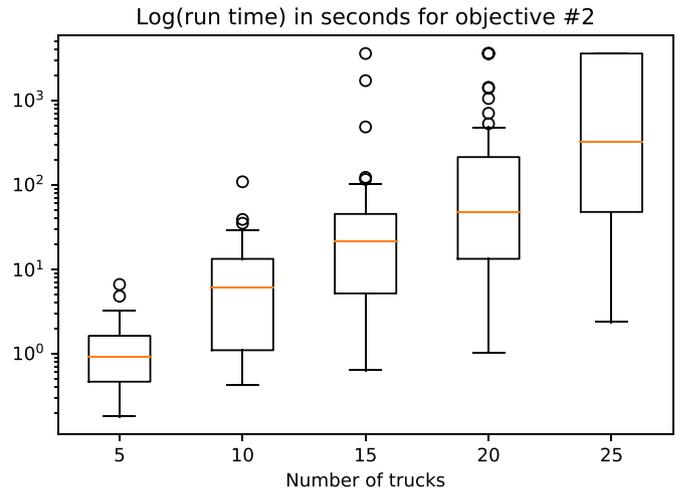


Fig. 9. Run time vs. number of trucks.

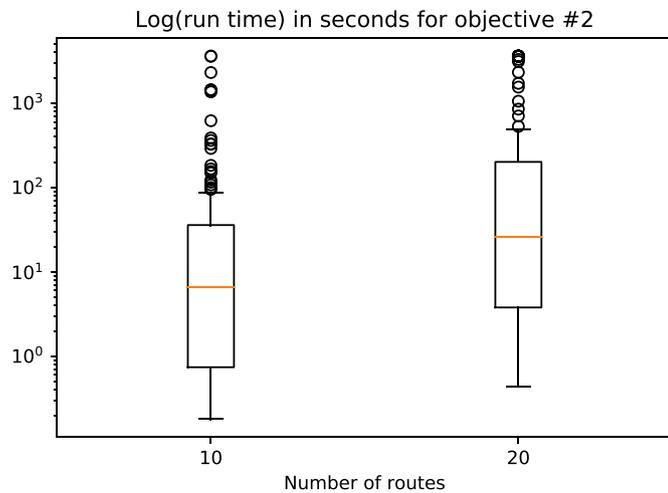


Fig. 8. Run time vs. number of routes.



Fig. 10. Run time vs. number of customers.

We now explore the effectiveness of solving phase-two formulation. We do so by quantifying the number of trucks reduced in phase two with respect to the solution found in phase one. Fig. 12 presents the number of trucks reduced discriminated by the number of trucks in the instance. As expected, as the number of trucks increases, the number of trucks reduced after solving phase-two formulation also increases. Overall, the results summarized in Fig. 12 support the need of phase two in our approach, as significant reductions in the number of trucks used in the optimal solution of phase one can be attained.

With respect to the real instances provided by the company, Table 4 presents a comparison between the solutions implemented by the company, and the solutions obtained after each phase of the two-phase approach. In all cases the company's solution uses

significantly fewer trucks than the number recommended by the best solution at the expense of a vastly lower value of the objective function. It is also worth noting that in four out of the five instances, the number of trucks used is reduced from phase one to phase two. The proposed exact approach improves the company's solution (objective 1) between 41.15% and 92.87%, although the company's solution uses between 12.5% and 71.43% fewer trucks when compared to the optimal solution. It is interesting that the company uses fewer trucks but consider how they approach this problem. It is a combinatorial problem and they usually stop when they find a feasible solution, rather than trying to optimize customer service. Because of their ad hoc approach to solving the problem, they typically begin and end with fewer trucks to make the problem more tractable for the human brain.

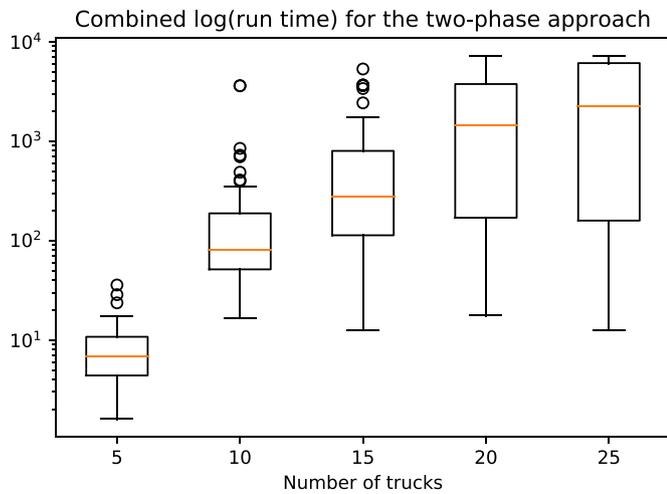


Fig. 11. Combined run time in the two-phase approach.

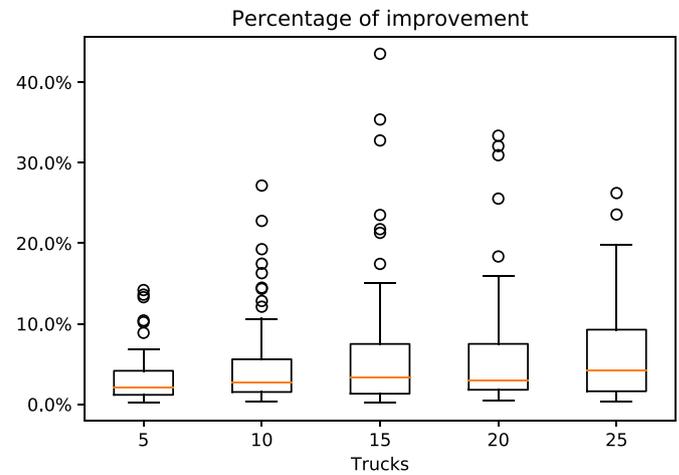


Fig. 13.  $\Delta_j$  values vs. number of trucks.

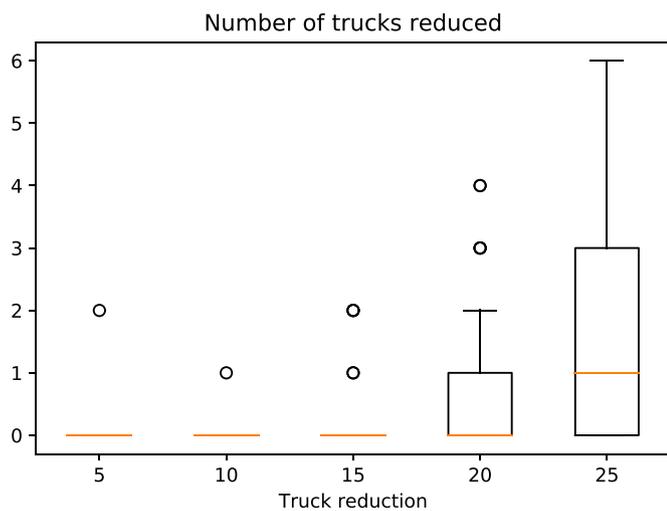


Fig. 12. Trucks reduced in phase two.

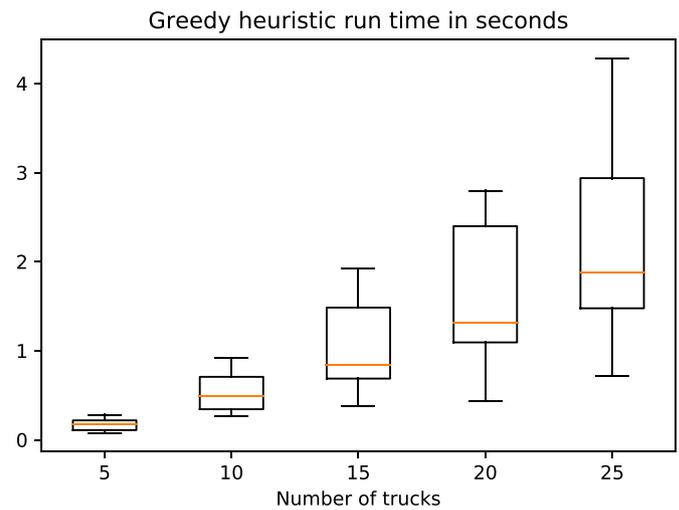


Fig. 14. Run times of the greedy heuristic.

Greedy heuristic

We now analyze the solution quality and run time of the greedy heuristic. For each instance  $j$ , the difference  $\Delta_j$  between the value of the objective function (1) obtained after solving the phase-one formulation (i.e.,  $Z_j$ ), and the corresponding value found by means of the greedy heuristic (i.e.,  $G_j$ ), was calculated as a percentage, using expression (18). Fig. 13 presents the box-plots for the  $\Delta_j$  values, discriminated by the number of trucks in the instance.

$$\Delta_j = \frac{Z_j - G_j}{G_j} \cdot 100\% \quad (18)$$

Overall, the average improvement of the two-phase approach relative to the greedy heuristic was 5.60%, with a maximum of 43.53%. Since objective function (1) is the primary objective of this work, the results summarized in Fig. 13 point out the potential improvement in terms of customer service level that can be attained if the proposed two-phase solution approach is implemented instead of the greedy heuristic. As per the run time required by the greedy heuristic to terminate, Fig. 14 presents the box-plots of the run times (in seconds) discriminated by the number of trucks in the instance. In all cases, the greedy heuristic took less than five seconds to terminate.

Finally, the greedy heuristic, on average, tends to use less trucks when compared to the optimal solutions obtained by means of

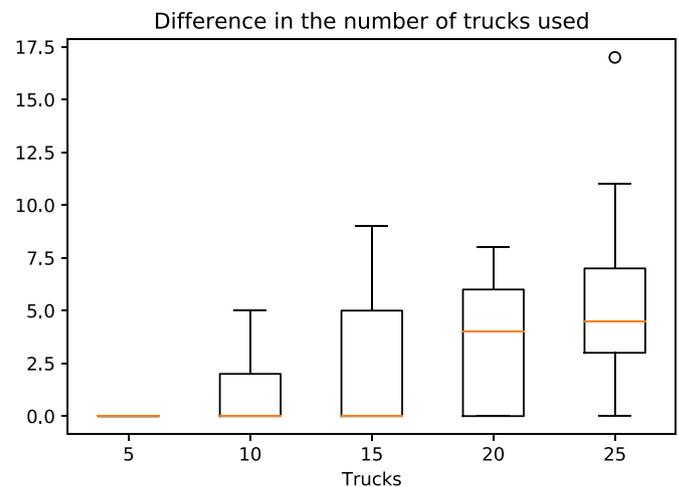


Fig. 15. Difference in the number of trucks used: Greedy vs. two-phase approach.

the proposed two-phase approach. Fig. 15 presents the difference between the number of trucks used in the optimal solution, and the trucks used in the solution found by the greedy heuristic. In the figure, a positive difference means that the optimal solution used more trucks than the greedy heuristic. However, since adding a truck is not a concern of the company, the improvement in customer service identified by the optimal method compared with

the heuristic comes at no additional cost. A repository with the test instances, the solution files produced by the solver, and a spreadsheet file summarizing the results of the computational experiments is available at <https://github.com/mcvelezg/mldiatlp/>.

## 6. Conclusions

This paper has developed an exact approach and a heuristic to load both one and two level trucks to deliver motorcycles on routes that span multiple cities. The primary objective is to satisfy both customer demand and timing of delivery. We consider both random instances and actual problems from the company that inspired the work. The exact approach uses a hierarchical structure to optimize customer satisfaction then considers the number of trucks used, which each truck has a minimum utilization. Results greatly improve the solutions which the company identifies with their current manual approach. After extensive computational experiments carried out on both real-life and randomly generated instances, the exact two-phase approach showed to be effective in solving to optimality 240 out of the 300 instances tested. The solutions found by means of the two-phase approach were, on average, 5.60% better than those found using the greedy heuristic. With respect to computational cost, the two-phase approach took an average of about 20 minutes to terminate, with approximately 60% of the instances solved within five minutes. The heuristic is quick and offers almost optimal solutions in most of the cases, even for realistic sized problems. However, the computational effort of the exact approach is still very modest and well within the capabilities of the company. Our work has not considered the routing decision because in the country considered, the inter-city road network is very limited and does not offer viable options. Our approach herein could be furthered by integrating routing decisions along with the loading decisions for those countries or regions with routing alternatives.

## References

- Alonso, M., Alvarez-Valdes, R., Iori, M., & Parreño, F. (2019). Mathematical models for multi container loading problems with practical constraints. *Computers & Industrial Engineering*, 127, 722–733.
- Balseiro, S., Loiseau, I., & Ramonet, J. (2011). An ant colony algorithm hybridized with insertion heuristics for the time dependent vehicle routing problem with time windows. *Computers & Operations Research*, 38(6), 954–966.
- Bortfeldt, A., Gehring, H., & Mack, D. (2003). A parallel tabu search algorithm for solving the container loading problem. *Parallel Computing*, 29(5), 641–662.
- Bortfeldt, A., & Wäscher, G. (2013). Constraints in container loading A state-of-the-art review. *European Journal of Operational Research*, 229(1), 1–20.
- Braaten, S., Gjønnnes, O., Hvattum, L. M., & Tirado, G. (2017). Heuristics for the robust vehicle routing problem with time windows. *Expert Systems with Applications*, 77, 136–147.
- Braekers, K., Ramaekers, K., & Nieuwenhuysse, I. V. (2016). The vehicle routing problem: State of the art classification and review. *Computers & Industrial Engineering*, 99, 300–313.
- Brown, G. G., Ellis, C. J., Graves, G. W., & Ronen, D. (1987). Real-time, wide area dispatch of mobil tank trucks. *Interfaces*, 17(1), 107–120.
- Brown, G. G., & Ronen, D. (1997). Consolidation of customer orders into truckloads at a large manufacturer. *Journal of the Operational Research Society*, 48(8), 779–785.
- Campelo, P., Neves-Moreira, F., Amorim, P., & Almada-Lobo, B. (2018). Consistent vehicle routing problem with service level agreements: A case study in the pharmaceutical distribution sector. *European Journal of Operational Research*, 273(1), 131–145.
- Che, C. H., Huang, W., Lim, A., & Zhu, W. (2011). The multiple container loading cost minimization problem. *European Journal of Operational Research*, 214(3), 501–511.
- Chunyu, R., Jinying, S., & Xiaobo, W. (2010). Study on hybrid genetic algorithm for multi-vehicle and multi-cargo loading problem. In *Proceedings of 2010 IEEE international conference on intelligent computing and intelligent systems* (pp. 554–558). Xiamen, China
- Cordeau, J.-F., Dell'Amico, M., Falavigna, S., & Iori, M. (2015). A rolling horizon algorithm for auto-carrier transportation. *Transportation Research Part B: Methodological*, 76, 68–80.
- Cornillier, F., Bockor, F., & Renaud, J. (2012). Heuristics for the multi-depot petrol station replenishment problem with time windows. *European Journal of Operational Research*, 220(2), 361–369.
- Derehi, T., & Das, G. S. (2010). A hybrid simulated annealing algorithm for solving multi-objective container-loading problems. *Applied Artificial Intelligence*, 24(5), 463–486.
- Egeblad, J., Garavelli, C., Lisi, S., & Pisinger, D. (2010). Heuristics for container loading of furniture. *European Journal of Operational Research*, 200(3), 881–892.
- Fraser, H. J., & George, J. A. (1994). Integrated container loading software for pulp and paper industry. *European Journal of Operational Research*, 77(3), 466–474.
- Hokama, P., Miyazawa, F. K., & Xavier, E. C. (2016). A branch-and-cut approach for the vehicle routing problem with loading constraints. *Expert Systems with Applications*, 47, 1–13.
- Jiang, J., Ng, K. M., Poh, K. L., & Teo, K. M. (2014). Vehicle routing problem with a heterogeneous fleet and time windows. *Expert Systems with Applications*, 41(8), 3748–3760.
- Kim, J.-U., & Kim, Y.-D. (2000). A lagrangian relaxation approach to multi-period inventory/distribution planning. *Journal of the Operational Research Society*, 51(3), 364–370.
- Kim, J.-U., Kim, Y.-D., & Shim, S.-O. (2002). Heuristic algorithms for a multi-period multi-stop transportation planning problem. *Journal of the Operational Research Society*, 53(9), 1027–1037.
- Leung, S. C., Zhang, Z., Zhang, D., Hua, X., & Lim, M. K. (2013). A meta-heuristic algorithm for heterogeneous fleet vehicle routing problems with two-dimensional loading constraints. *European Journal of Operational Research*, 225(2), 199–210.
- Liu, J., Smith, A. E., & Qian, D. (2016). The vehicle loading problem with a heterogeneous transport fleet. *Computers & Industrial Engineering*, 97, 137–145.
- Makarem, O. C., & Haraty, R. A. (2010). Smart container loading. *Journal of Computational Methods in Sciences and Engineering*, 10(2), 231–245.
- Milosavljević, N., Teodorović, D., Papić, V., & Pavković, G. (1996). A fuzzy approach to the vehicle assignment problem. *Transportation Planning and Technology*, 20(1), 33–47.
- Pollaris, H., Braekers, K., Caris, A., Janssens, G. K., & Limbourg, S. (2015). Vehicle routing problems with loading constraints: State-of-the-art and future directions. *OR Spectrum*, 37(2), 297–330.
- Ramos, A. G., Silva, E., & Oliveira, J. F. (2018). A new load balance methodology for container loading problem in road transportation. *European Journal of Operational Research*, 266(3), 1140–1152.
- Reil, S., Bortfeldt, A., & Mönch, L. (2018). Heuristics for vehicle routing problems with backhauls, time windows, and 3d loading constraints. *European Journal of Operational Research*, 266(3), 877–894.
- Ren, J., Tian, Y., & Sawaragi, T. (2011). A tree search method for the container loading problem with shipment priority. *European Journal of Operational Research*, 214(3), 526–535.
- Respen, J., & Zufferey, N. (2017). Metaheuristics for truck loading in the car production industry. *National Transactions in Operational Research*, 24, 277–301.
- Romão, O. C., dos Santos, A. G., & Arroyo, J. E. C. (2012). A two phase hybrid method to support 2d and 3d vehicle loading. In *Proceedings of the twelfth international conference on intelligent systems design and applications (ISDA)* (pp. 320–325). Kochi, India
- Ronen, D., & Goodhart, C. A. (2008). Tactical store delivery planning. *Journal of the Operational Research Society*, 59(8), 1047–1054.
- Stoyan, Y., & Yaskov, G. (2012). Packing congruent hyperspheres into a hypersphere. *Journal of Global Optimization*, 52(4), 855–868.
- Thapatsuwon, P., Pongcharoen, P., Hicks, C., & Chainate, W. (2012). Development of a stochastic optimisation tool for solving the multiple container packing problems. *International Journal of Production Economics*, 140(2), 737–748.
- Tian, T., Zhu, W., Lim, A., & Wei, L. (2016). The multiple container loading problem with preference. *European Journal of Operational Research*, 248(1), 84–94.
- Toffolo, T. A., Esprit, E., Wauters, T., & Berghe, G. V. (2017). A two-dimensional heuristic decomposition approach to a three-dimensional multiple container loading problem. *European Journal of Operational Research*, 257(2), 526–538.
- Vukadinović, K., Teodorović, D., & Pavković, G. (1999). An application of neurofuzzy modeling: The vehicle assignment problem. *European Journal of Operational Research*, 114(3), 474–488.
- Wei, L., Zhang, Z., Zhang, D., & Lim, A. (2015). A variable neighborhood search for the capacitated vehicle routing problem with two-dimensional loading constraints. *European Journal of Operational Research*, 243(3), 798–814.
- Yüceer, Ü., & Özakça, A. (2010). A truck loading problem. *Computers & Industrial Engineering*, 58(4), 766–773.
- Zachariadis, E. E., Tarantilis, C. D., & Kiranoudis, C. T. (2016). The vehicle routing problem with simultaneous pick-ups and deliveries and two-dimensional loading constraints. *European Journal of Operational Research*, 251(2), 369–386.
- Zhao, X., Bennell, J. A., Bektaç, T., & Dowsland, K. (2016). A comparative review of 3d container loading algorithms. *International Transactions in Operational Research*, 23(1–2), 287–320.