CIRRELT

Centre interuniversitaire de recherche
sur les réseaux d'entreprise, la logistique et le transport

Interuniversity Research Centre
on Enterprise Networks, Logistics and Transportation

The Value of Integrating Loading and Routing

Jean-François Côté
Gianfranco Guastaroba
Maria Grazia Speranza

July 2015

CIRRELT-2015-31

Document de travail également publié par la Faculté des sciences de l'administration de l'Université Laval, sous le numéro FSA-2015-008.

Bureaux de Montréal :
Université de Montréal
Pavillon André-Aisenstadt
C.P. 6128, succursale Centre-ville
Montréal (Québec)
Canada H3C 3J7
Téléphone : 514 343-7575
Télécopie : 514 343-7121

Bureaux de Québec :
Université Laval
Pavillon Palasis-Prince
2325, de la Terrasse, bureau 2642
Québec (Québec)
Canada G1V 0A6
Téléphone : 418 656-2073
Télécopie : 418 656-2624

www.cirrelt.ca

The Value of Integrating Loading and Routing

Jean-François Côté[1,2,*], Gianfranco Guastaroba[2], Maria Grazia Speranza[2]

1 Interuniversity Research Centre on Enterprise Networks, Logistics and Transportation (CIRRELT) and Department of Operations and Decision Systems, 2325 de la Terrasse, Université Laval, Québec, Canada G1V 0A6

2 Department of Economics and Management, C. Da s. Chiara 50, University of Brescia, Brescia, Italy

Abstract. Location-routing, inventory-routing, multi-echelon routing, routing problems with loading constraints are classes of problems that are receiving increasing attention in the scientific community. Problems in these classes generalize classical vehicle routing problems enlarging the decision space to optimize a broader system. The resulting problems are computationally harder to solve but offer opportunities to achieve remarkable additional savings. In this paper we address the issue of quantifying the potential benefit deriving from tackling such complex problems instead of sequentially solving the individual problems they integrate. To this aim, we consider as a proof of concept the Capacitated Vehicle Routing Problem (CVRP) with Two-dimensional Loading constraints (2L-CVRP), a variant of the CVRP where rectangular-shaped items have to be delivered to customers and loading constraints have to be satisfied. We consider the 2L-CVRP in an integrated manner and compare the solutions with those obtained from two not integrated approaches based on addressing sequentially and therefore separately, the routing and the loading problems. The importance of an integrated approach for the 2L-CVRP is validated through the study of the worst-case performance of the not integrated approaches, and conducting computational experiments on benchmark and new instances.

Keywords. Integration in logistics, Vehicle Routing Problems, loading constraints, orthogonal packing, worst-case analysis.

---

* Corresponding author: Jean-Francois.Cote@cirrelt.ca

Dépôt légal – Bibliothèque et Archives nationales du Québec
Bibliothèque et Archives Canada, 2015

1 Introduction

In recent years, the study of more realistic and involved variants than the classical Capacitated Vehicle Routing Problem (CVRP) is attracting an increasing academic attention. This growing body of literature is stimulated by the desire of bridging the gap between academic problems and real-world applications, on the one side, and the recent advances in optimization methods and computer capabilities, on the other side, that are making it possible to jointly solve strongly interdependent problems that have been, until recently, addressed independently. *Integrated vehicle routing problems* is the term increasingly used to denote the class of problems where the Vehicle Routing Problem (VRP) arises in combination with other optimization problems within the broader context of logistics and transportation (Bektaş *et al.* [6]). Some examples of problems in this class are the location-routing problems where decisions of location and routing are jointly optimized (see the surveys by Prodhon and Prins [34] and by Drexl and Schneider [21]); the production-routing problems which jointly optimizes production, inventory, and routing decisions (see the survey by Adulyasak *et al.* [1]); the inventory-routing problems which combines routing and inventory management problems (see the survey by Coelho *et al.* [14]); the multi-echelon routing problems that study how to optimally route freight in distribution systems comprising several levels (see the survey by Cuda *et al.* [18] on two-echelon routing problems); and routing problems with loading constraints where the routing of vehicles and the loading of goods onto them are simultaneously optimized (see the survey by Iori and Martello [27]). A remarkable number of surveys, such as those cited above, as well as special issues in international journals, including the recent one edited by Bektaş *et al.* [6], that recently appeared in the literature testify the increasing attention that this research area is attracting among academics.

As integrated vehicle routing problems combine optimization problems that are usually $\mathcal{NP}$-hard by themselves, the prevailing attitude among operations researchers has been, until recently, to tackle each problem independently, at the expense of global optimization. In fact, solving each problem independently, even by means of an exact method, and then combining the partial solutions obtained, typically leads to a sub-optimal solution for the integrated (global) problem. On the other side, combining two, or more, hard problems causes a significant increase of the computational burden required, but tends to provide considerably better solutions than solving optimally each problem independently, often even if the integrated problem is solved with a heuristic. It is important to motivate an integrated approach, quantifying the magnitude of the benefits that can be achieved addressing the integrated problem directly instead of tackling each problem independently. To the best of our knowledge, only few papers appeared in the literature along this line of research. Salhi and Rand [36] show that ignoring routing aspects when location decisions are taken leads to sub-optimal solutions. Chandra and Fisher [12] provide a computational study aimed at investigating the potential benefits of coordinating production, inventory and routing decisions. Their findings show that a cost reduction ranging from 3% to 20% can be achieved integrating the above decisions within a single model rather than sequentially solving the separate problems. Bertazzi and Speranza [8] show an illustrative example that motivates the need of combining routing and inventory management problems, whereas Archetti and Speranza [4] present a computational study that focuses on the analysis of the benefits of an integrated policy applied to an inventory-routing problem.

The VRP is one of the most important and investigated class of combinatorial optimization problems. It calls for the determination of an optimal set of routes to be performed by a fleet of vehicles to serve a given set of customers (e.g., see Toth and Vigo [37]). In the CVRP, which is the simplest and most studied member of the class of VRPs, customer demands are deterministic, known in advance, and cannot be split. The vehicles are all identical, based at a single depot, and have a maximum capacity. In the CVRP, the demand of each customer is expressed by a single value, usually representing the total weight of the items to be transported. Thus, a solution is feasible for the CVRP if the sum of the demands of the customers assigned to each vehicle does not exceed its capacity. Nevertheless, in many real-world freight transportation applications it cannot be neglected that the items are characterized not only by a weight but also by a shape. Examples of these applications include the transportation of large and heavy items, such as furniture, household appliances, and some mechanical components (e.g., see Wang *et al.* [38]). In these situations, a solution that is feasible for the CVRP may prove to be infeasible in practice since it is impossible to determine a feasible loading pattern to allocate all the items within the loading area of the vehicles. These loading issues are closely related to multi-dimensional packing problems, especially extensions of the classical (one dimensional) Bin Packing Problem (BPP). Several operational restrictions often complicate the problem further (see Iori *et al.* [28]). Among other restrictions, since the size (and weight) of the items in the related applications is usually large, load rearrangements at a customer location can be complicated, overly time consuming, or even impossible. Additionally, it may be necessary to impose that the unloading of the items of a customer must not be blocked by any item belonging to a customer to be visited later along the route. The latter requirement (called 'Last-In First-Out' (LIFO) or, sometimes, 'sequential loading' or 'rear loading' constraint) is introduced to model, among other situations, the impossibility for a forklift truck to perform substantial lateral shifts while unloading an item. As a consequence, the sequence of customers to be visited in a vehicle route has to be designed to avoid unnecessary unloading and rearrangements operations. All the former observations motivate the growing interest that routing problems with loading constraints is attracting among academics and practitioners.

In this paper we consider the *CVRP with Two-dimensional Loading constraints*, henceforth referred to as *2L-CVRP*, that is a variant of the CVRP where rectangular-shaped items have to be transported and loading constraints have to be satisfied. The assumption that characterizes the 2L-CVRP is that the items cannot be stacked on top of each other. The latter assumption differentiates the 2L-CVRP from the other major class of routing problems with loading constraints, the CVRP with three-dimensional loading constraints where some items can be superposed (e.g., see Gendreau *et al.* [24]). The 2L-CVRP models applications concerning the transportation of heavy or fragile items, such as refrigerators or pieces of catering equipment, such as food trolleys (Iori and Martello [27]), or when customer orders are loaded onto pallets which cannot be stacked on top of each other. In the 2L-CVRP the demand of each customer is composed of a set of rectangular-shaped items such that, for each item, its weight, shape and orientation are given. A fleet of identical vehicles based at a single depot is available to deliver these items. The vehicles have a given weight capacity and a rectangular loading area that can be accessed only from one side (we assume that vehicles are rear-loaded). The 2L-CVRP calls for the determination of a minimum-cost set of routes to be traveled by the given fleet of vehicles to serve the customers, subject to the following set of constraints:

a) *Weight capacity constraints*: the total weight of the items loaded onto a vehicle cannot exceed its capacity;

b) *Classical BPP constraints*: there must exist a non-overlapping loading pattern of all the items into the loading area of the vehicles;

c) *Item clustering constraints*: all the items of a given customer must be assigned to the same vehicle;

d) *Item orientation constraints*: each item has a fixed orientation and cannot be rotated;

e) *Orthogonality constraints*: each item has to be loaded with its edges parallel to those of the vehicle;

f) *LIFO constraints*: items of the current customer must be directly available by pulling them out from the rear doors without moving any item of other customers.

The 2L-CVRP reduces to the classical CVRP by assigning to each customer a single item having both sizes equal to 1, and by setting the dimension of the loading area of each vehicle equal to the total number of customers (Iori *et al.* [28]). The literature on the 2L-CVRP is still rather limited. Iori *et al.* [28] propose the first exact algorithm for a 2L-CVRP where single-customer routes are not allowed and all the available vehicles have to be routed. The solution method proposed is based on a Branch-and-Cut (B&C) algorithm that iteratively calls an inner branch-and-bound for the solution of the loading sub-problem. Computational results are given for instances generated from benchmark test problems for the CVRP that comprise up to 35 customers and 114 items. The only other exact method that we are aware of has been proposed in Côté *et al.* [15] for a stochastic variant of the 2L-CVRP, where the size and weight of some items are not known with certainty when the routes are planned. The authors show that their method can also be applied to solve the deterministic counterpart of the stochastic 2L-CVRP, i.e., the 2L-CVRP addressed in Iori *et al.* [28], reducing to a B&C. To this aim, they conduct some computational experiments on the set of benchmark instances introduced in [28] and a subset of those proposed in [25]. The results indicate that the algorithm introduced in Côté *et al.* [15] is the current state-of-the-art exact method for the solution of the 2L-CVRP studied in Iori *et al.* [28], being able to solve to optimality instances with up to 71 customers and 226 items. On the other side, a larger number of authors propose heuristics for the 2L-CVRP as defined above. These algorithms are often tested to solve also slightly different variants, obtained removing or replacing some of its constraints. Among the most important variants we mention the one where the LIFO constraints are removed (the unrestricted 2L-CVRP), and the one where items can be rotated (the non-oriented 2L-CVRP). Gendreau *et al.* [25] design a tabu search algorithm for the 2L-CVRP. An ant colony optimization algorithm is proposed in Fueller *et al.* [23], whereas Zachariadis *et al.* [39] develop a guided tabu search. More recently, Duhamel *et al.* [22] design a multi-start evolutionary local search, whereas Leung *et al.* [31] implement an extended version of the guided tabu search proposed in [39]. Variants of the 2L-CVRP with a heterogeneous fleet of vehicles are addressed in Leung *et al.* [30] and in Dominguez *et al.* [20].

The packing problem associated with the loading of items inside a vehicle is formally known as the *Two-dimensional Orthogonal Packing Problem with Unloading Constraints* (2OPPUL). Given a sequence of customers, the 2OPPUL calls for the determination of a feasible packing of the items that do not violate the above set of constraints. Most heuristic algorithms proposed for the solution of the 2L-CVRP use simple heuristics such as Bottom-Left, Bottom-Left Fill, and Touching Perimeter heuristics for tackling the packing problems (see Zachariadis *et al.* [39]). In da Silveira *et al.* [19] a GRASP heuristic for the strip packing problem with unloading constraints is proposed. In this variant, a strip of fixed width and infinite height is given and the objective of the problem is to minimize the height of the occupied area. To the best of our knowledge, only two exact methods are available in the literature to tackle the 2OPPUL. Specifically, Iori *et al.* [28] design a branch-and-bound algorithm based on the exact method for the strip packing problem described in Martello *et al.* [33]. More recently, Côté *et al.* [16] describe a Benders decomposition approach which can solve instances with up to 52 items. Relaxing the LIFO constraints, the problem reduces to a *Two-dimensional Orthogonal Packing Problem* (2OPP), that is found as a sub-problem of the *Two-dimensional Strip Packing Problem* (2SPP) and of the two-dimensional bin packing problem. Recent algorithms for these problems can be found in [2, 3, 7, 10, 11, 13, 17, 33]. We refer the interested reader to the surveys by Bortfeldt and Wäscher [9] and by Iori and Martello [27] for a literature review on the three-dimensional case.

Contributions of the paper. The contributions of this paper are the following. We mentioned above that exact algorithms for a 2L-CVRP where single-customer routes are not allowed and all the vehicles available have to be routed have been proposed in Iori *et al.* [28] and in Côté *et al.* [15]. In the present paper, we consider a more general definition for the 2L-CVRP where single-customer routes are possible, and some vehicles can be unused if this is cost-effective. We describe a mathematical formulation for this version of the 2L-CVRP, and we solve it by means of the exact method described in Côté *et al.* [15] after some minor adaptations. We propose two not integrated approaches for the solution of the 2L-CVRP based on considering sequentially, and therefore separately, the routing and the loading aspects of the problem. The first approach tries to mimic the possible behavior of a logistic operator, whereas the second is more sophisticated and is based on the solution of a routing problem with profits. We provide evidence of the importance of an integrated approach for the 2L-CVRP, studying the worst-case performance of the not integrated approaches, as well as reporting on extensive computational experiments conducted on benchmark and new instances.

Structure of the paper. The structure of the paper is as follows. Section 2 provides a formal description of the 2L-CVRP, as well as of the approach used to solve it in an integrated manner. Two not integrated solution approaches for the 2L-CVRP are detailed in Section 3, whereas section 4 gives an overview of the methodology implemented to address the packing problems that are generated during the execution of all the approaches. Section 5 studies the worst-case performance of the proposed not integrated approaches. Computational experiments are presented and discussed in Section 6. Finally, Section 7 draws some conclusions and discusses future research directions.

2 The Integrated Problem

The 2L-CVRP considered in this paper can be described as follows. Let $G = (V, E)$ be a complete undirected and weighted graph. Set $V = \{0, 1, 2, ..., n\}$ is a set of vertices with cardinality $|V| = n+1$ ($|B|$ denotes the cardinality of set B), where vertex 0 corresponds to the depot, and set $C = V\setminus\{0\}$ denotes a set of $|C| = n$ customers. Set $E = \{\{i, j\} : i, j \in V, i < j\}$ is a set of edges, and a non-negative traveling cost c_{ij} is associated with each edge $\{i, j\} \in E$. We assume that the traveling costs c_{ij} satisfy the triangle inequality.

Each customer $j \in C$ has a known and deterministic demand comprising m_j two-dimensional items, each one having a specific width and height denoted as w^l_j and h^l_j ($l = 1, \ldots, m_j$), respectively. Henceforth, the total area covered by the m_j items associated with customer $j \in C$ is denoted as $a_j = \sum_{l=1}^{m_j} w^l_j h^l_j$, whereas q_j indicates their total weight (recall that the demand of each customer cannot be split among different vehicles).

A fleet of K_{max} homogeneous and capacitated vehicles is available at the depot to serve all the customers. Vertex 0 is the starting and ending point of any route. Each vehicle has a maximum weight capacity Q, and a rectangular loading area that is accessible from the back for loading/unloading operations. The loading area of each vehicle has a given width and height denoted as W and H, respectively, such that the total area available in each vehicle to carry the items is $A = WH$. The 2L-CVRP calls for the determination of a minimum-cost set of no more than

K_{max} routes to serve the demand of the customers, such that the constraints a)-f) described in the Introduction are not violated. Let $\mathcal{R}_{inf}$ be the set composed of all routes that do not satisfy the loading constraints b)-f).

We model the 2L-CVRP using two-index vehicle flow variables as main decision variables, inspired by classical formulations for the CVRP. Let $x_{ij} \in \{0,1\}$, with $1 \leq i < j$, be a binary variable that takes value 1 if edge $\{i,j\} \in E$ is traversed by a vehicle, and 0 otherwise. In order to allow single-customer routes, we introduce the integer variables $x_{0j} \in \{0,1,2\}$, with $j \in C$, that represent the number of times that each edge $\{0,j\}$ incident to the depot is traversed. The possibility to use less than K_{max} vehicles is modeled as follows. Let K_{min} be a lower bound on the number of vehicles required to serve all the customers in set C. Then, we introduce the binary variable $z_k \in \{0,1\}$, with $k = K_{min}, K_{min}+1, \ldots, K_{max}$, that takes values 1 if k vehicles are routed, and 0 otherwise. The 2L-CVRP can be cast as the following Integer Linear Programming (ILP) model:

$$\min \quad z = \sum_{i \in V \setminus \{n\}} \sum_{j \in V : j > i} c_{ij} x_{ij} \tag{1}$$

subject to

$$\sum_{j \in C} x_{0j} = \sum_{k=K_{min}}^{K_{max}} 2k z_k \tag{2}$$

$$\sum_{k=K_{min}}^{K_{max}} z_k = 1 \tag{3}$$

$$\sum_{i \in V : i < j} x_{ij} + \sum_{h \in V : h > j} x_{jh} = 2 \qquad j \in C \tag{4}$$

$$\sum_{i \in S} \sum_{j \in S : j > i} x_{ij} \leq |S| - \left\lceil \max \left\{ \frac{\sum_{j \in S} a_j}{A}, \frac{\sum_{j \in S} q_j}{Q} \right\} \right\rceil \qquad S \subseteq C, 2 \leq |S| \leq n \tag{5}$$

$$\sum_{(i,j) \in R} x_{ij} \leq |R| - 1 \qquad R \in \mathcal{R}_{inf} \tag{6}$$

$$x_{ij} \in \{0,1\} \qquad 1 \leq i < j \leq n \tag{7}$$

$$x_{0j} \in \{0,1,2\} \qquad j \in C \tag{8}$$

$$z_k \in \{0,1\} \qquad k = K_{min}, K_{min}+1, \ldots, K_{max}. \tag{9}$$

Objective function (1) minimizes the total traveling cost. Constraints (2), along with constraints (3) and (9), impose that the number of vehicles used in any feasible solution is at least equal to K_{min} and at most equal to K_{max}. The minimization of the objective function leads to the selection of the most appropriate number of vehicles to use in the interval $[K_{min}, K_{max}]$. The degree constraints (4) state that exactly two edges incident to each vertex associated with a customer must be selected. Constraints (5) are the subtour elimination and rounded-capacity constraints. Each route that does not satisfy the loading constraints, i.e., each route in set $\mathcal{R}_{inf}$, is forbidden by constraint (6). In these inequalities, hereafter referred to as infeasible path constraints, route R is defined as the set of edges traversed in an infeasible route belonging to set $\mathcal{R}_{inf}$. Finally, constraints (7), (8), and (9) define the decision variables.

The problem tackled in this paper generalizes the problem addressed in Iori *et al.* [28] and in

Côté *et al.* [15]. The problem we consider reduces to the problem studied in the latter papers by setting $K_{min} = K_{max}$, and defining x_{0j} as binary variables. The exact algorithm described in Côté *et al.* [15] has been adapted to solve optimization model (1)-(9). As mentioned above, this exact algorithm was originally proposed to solve the stochastic 2L-CVRP and reduces to a B&C when applied to its deterministic counterpart. We briefly summarize the structure of the algorithm and refer to [15] for more details.

At the beginning of the algorithm, the subtour and rounded-capacity inequality (5) and infeasible path constraints (6) are relaxed. The resulting problem (1)-(4) and (7)-(9) is then solved by means of a Mixed Integer Linear Programming (MILP) general-purpose solver. At each node of the branching tree, the algorithm first checks for the presence of violated subtour and rounded-capacity inequalities using the CVRPSEP package described in [32]. If any violated inequality is found, it is added to the model and the node is solved again. Otherwise, the following two alternatives may occur:

1. the solution is fractional, and then the algorithm branches on fractional variables;

2. the solution is integer, and then the algorithm proceeds as follows. First, each route in the solution is considered as an unordered set S of customers. The packing problem associated with an unordered set of customers is a 2OPP (see the Introduction) and is solved using the methods described in the following Section 4. If the 2OPP is infeasible, the route violates the loading constraints, and the following inequality is added to the model:

$$\sum_{i \in S} \sum_{j \in S : j > i} x_{ij} \leq |S| - 1. \tag{10}$$

   This inequality is a stronger form of the infeasible path constraint (6) as it forbids any path visiting all the customers in S. After that all the inequalities found of type (10) have been added to the model, the node is solved again. On the other hand, if the 2OPP is feasible, the 2OPPUL associated with each path is solved using the methods described in Section 4 and constraints (6) are added, if necessary. If no violated constraint is found, the solution is feasible.

The introduction of single-customer routes requires only marginal modifications of the B&C described above. On the other hand, the lower bound K_{min} on the number of vehicles is computed as the maximum value among the following lower bounds:

1. $\left\lceil \frac{\sum_{i \in C} q_i}{Q} \right\rceil$;

2. two lower bounds based on the solution of a relaxation of the 2SPP. As mentioned in the Introduction, in the 2SPP a given set of two-dimensional items have to fit inside a strip of infinite height and fixed width. The 2SPP aims at minimizing the height of the occupied area. Dividing the resulting height by H, i.e., the height of each vehicle, and rounding up this value gives a bound on the required number of vehicles. Instead of solving the 2SPP, we solve the linear relaxation of the Gilmore-Gomory formulation (see [26]) of the cutting stock problem on the height and the width;

3. we solve the linear relaxation of the following formulation:

$$
\begin{aligned}
\min \ & \sum_{\mathrm{K} \in \mathcal{K}} \xi_{\mathrm{K}} & \\
\text{subject to} \ & \sum_{\mathrm{K} \in \mathcal{K}} b_{i\mathrm{K}} \xi_{\mathrm{K}} \geq 1 & i \in C \\
& \xi_{\mathrm{K}} \in \{0,1\} & \mathrm{K} \in \mathcal{K},
\end{aligned}
$$

where $\mathcal{K} = \{S \subseteq C \mid \sum_{i \in S} q_i \leq Q, \sum_{i \in S} a_i \leq A\}$, i.e., it contains sets of customers whose sums of demand weights and areas are smaller than the weight capacity and loading area available. Parameter $b_{i\mathrm{K}} \in \{0,1\}$ is equal to 1 if customer i is present in set K, and 0 otherwise. The above model is solved by column generation where the sub-problem is a knapsack problem with 2 constraints that is modeled as a MILP and solved using a general-purpose solver.

Parameter K_{max} is set equal to the number of vehicles indicated in the original instance.

Henceforth, we will say that the problem is solved with an *Integrated Solution Approach* (*ISA*) when the optimization model (1)-(9) is used.

3 Not Integrated Solution Approaches

The broad idea of a not integrated solution approach for the 2L-CVRP is to address separately the CVRP and the loading problem, instead of tackling the problem as a whole. We consider two *Not Integrated Solution Approaches* (*NISA*). Both approaches follow the same general scheme: use a two-phase algorithm where the CVRP and the loading constraints are sequentially taken into consideration. The general scheme followed by the two NISAs is sketched in Algorithm 1.

The first phase (hereafter called *Phase 1*) is common to both NISAs, and it mainly consists in solving a relaxation of optimization model (1)-(9) where the infeasible path constraints (6) are neglected. The resulting optimization model is a formulation for the CVRP that is solved along the lines described earlier for the ISA. Indeed, the subtour and rounded-capacity inequalities (5) are first relaxed. Then, problem (1)-(4) and (7)-(9) is solved by means of a general-purpose solver. At each node of the branching tree, the algorithm checks for the presence of violated subtour and rounded-capacity inequalities using the CVRPSEP package described in [32]. Any violated inequality found is added to the model, and the node is solved again. Otherwise, two alternatives may occur:

1. the solution is fractional, and then the algorithm branches on fractional variables;

2. the solution is integer, and then the solution found is feasible.

Once the above CVRP model is solved and an optimal solution is found, the loading feasibility of each route is assessed using the procedures described in Section 4. Each route that violates a loading constraint is hereafter called *infeasible route*. Let $\mathcal{R}'_{inf}$ denote the set containing all the infeasible routes in the CVRP solution.

The second phase (henceforth referred to as *Phase 2*) is different between the two NISAs and is therefore described separately below.

Algorithm 1 General Scheme of the Not Integrated Solution Approaches.

/* *Phase 1.* */

1. Solve a CVRP.
2. Identify infeasible routes for the 2L-CVRP.

/* *Phase 2.* */

1. Remove infeasible customers from the infeasible routes to make them feasible.
2. Determine new routes to serve the removed customers.

3.1 NISA: Phase 2

In this section, we first provide a general description of Phase 2, which is valid for both NISAs. Subsequently, we detail separately the structure of each NISA.

The general idea of Phase 2 is the following. Firstly, a subset of the customers that are visited in each infeasible route $R' \in \mathcal{R}'_{inf}$ is identified, such that their removal makes the route feasible for the loading constraints. These customers are hereafter referred to as *infeasible customers*. Let C_{inf} be the set of all the infeasible customers for all the infeasible routes in $\mathcal{R}'_{inf}$. Subsequently, each infeasible customer is removed from the corresponding infeasible route, along with the edges that connect the customer with its two adjacent vertices in the route. The edge incident to the latter two vertices is introduced to replace the removed edges. Finally, a set of routes is determined to serve the infeasible customers in set C_{inf}.

NISA with Nearest Customers

In general terms, the first approach, henceforth referred to as the *NISA with Nearest Customers* (*NISA-NC*), begins with removing from each infeasible route an adequate number of customers starting from the vertex nearest to the depot. Then, optimization model (1)-(9) is solved restricted to set C_{inf}. The basic rationale of this approach is to create a set of infeasible customers that are, hopefully, located in proximity to the depot. As a consequence, the routes to be determined have to visit vertices that are likely to be close to each other.

More formally, at the beginning of Phase 2 set C_{inf} is empty, i.e., $C_{inf} = \emptyset$. Each infeasible route $R' \in \mathcal{R}'_{inf}$ is modified as follows. Between the two vertices adjacent to the depot, the nearest to the depot, say i', is removed and added to set C_{inf}. Let R'' denote route R' where the two edges incident to node i' are replaced with the edge connecting the depot with the successor of node i' in the route. The feasibility of route R'' is then checked. If R'' is not feasible, we remove the successor of node i' in route R', and iterate the process until the route becomes feasible. Conversely, if R'' is feasible, another infeasible route from set $\mathcal{R}'_{inf}$, if any, is considered. Once all infeasible routes have been made feasible, optimization model (1)-(9) is solved restricted to set C_{inf}. Parameter K_{min} is computed as described in Section 2, whereas parameter K_{max} is computed as follows. Each instance is solved by means of the Adaptive Large Neighborhood Search heuristic (ALNS) introduced in [35], where we set the maximum number of iterations equal to 15000. The value K_{max} is then set equal to the number of vehicles used in the heuristic solution plus 2. We add 2 because in preliminary experiments we found that for some instances a better objective function could be achieved adding few vehicles to those selected in the solution found by the heuristic.

The rationale of the NISA-NC is to mimic the behavior of a logistic operator that, once a set of routes is determined solving a CVRP, loads each vehicle starting from the customer that, between the two adjacent to the depot, is the farthest. This will be the last customer visited in the route. If the route is infeasible, the items of some customers will not be loaded on that vehicle.

NISA with Optimization Model

In the second approach, hereafter referred to as the *NISA with Optimization Model* (*NISA-OM*), the identification of the infeasible customers and the subsequent determination of the routes to serve them are simultaneously carried out solving an optimization model, as detailed below. The rationale of the NISA-OM is to solve the second phase of a not integrated solution approach with a sophisticated method. Eventually, we will show that the benefits achieved using an integrated approach are remarkable, even if a sophisticated method is used to address both phases of a NISA.

We model the problem of identifying and routing the infeasible customers as a VRP with profits (see Archetti *et al.* [5] for a recent overview on this class of problems). In contrast with the classical VRP where it is mandatory to serve all the given customers, in a VRP with profits the customers to serve have to be determined among those belonging to a given set. A profit is in general associated with each customer that makes the visit of such a customer more or less attractive.

To model the problem considered in this section as a VRP with profits, we introduce the concept of *options*. Particularly, for each infeasible route $R' \in \mathcal{R}'_{inf}$ we identify a set of options $\mathcal{O}_{R'}$, where each option $o \in \mathcal{O}_{R'}$ represents a subset of the customers visited in route R' that, if removed, makes the resulting route, say R'', feasible for the loading constraints. Intuitively speaking, choosing option $o \in \mathcal{O}_{R'}$ indicates that the customers composing that option have been identified as infeasible customers. Therefore, exactly one option has to be chosen for each infeasible route in set $\mathcal{R}'_{inf}$, and a minimum-cost set of routes has to be determined to serve the customers composing the selected options.

The set of options for each infeasible route is generated as follows. Let R' be an infeasible route. Note that the graph is undirected and, therefore, route R' can be traveled in two directions. The algorithm starts choosing one of the two directions, and then removes one customer at a time from the beginning of the route, until the resulting tour is feasible for the loading constraints. Let η_1 be the number of customers removed. Then, the same procedure is repeated considering the opposite direction. Let η_2 be the number of customers removed. Note that, given the heterogeneity of the customer demands, η_1 and η_2 can be different. In the computational experiments, these two numbers are often equal and, if different, they differ in most of the cases by just one unit (e.g., $\eta_1 = \eta_2+1$). The algorithm evaluates each possible subset of customers whose cardinality is between 1 and the maximum among η_1 and η_2. The algorithm checks if, after removing those customers, the resulting route is feasible or not. If the route is feasible, then this subset of customers is an option and is added to set $\mathcal{O}_{R'}$.

Each option $o \in \mathcal{O}_{R'}$, with $R' \in \mathcal{R}'_{inf}$, is associated with a *profit*. This profit represents the savings, in terms of traveling costs, resulting from the removal of the customers composing option o from route R'. In other words, let $c(R')$ denote the cost of a generic infeasible route $R' \in \mathcal{R}'_{inf}$, and let R'' be the route obtained removing from R' the customers in option o. Then, profit p_o associated with option o is computed as $p_o = c(R') - c(R'')$.

We model the problem of identifying the infeasible customers and determining how to serve them, henceforth referred to as the *Selective CVRP with Two-dimensional Loading constraints* (*2L-SCVRP*), making use of a further set of binary variables, in addition to variables x_{ij}, x_{0j}, and z_k as defined in Section 2. Let $y_o \in \{0, 1\}$ be a binary variable that takes value 1 if option $o \in \mathcal{O}_{R'}$, with $R' \in \mathcal{R}'_{inf}$, is chosen, and 0 otherwise. Furthermore, in order to avoid an unnecessarily large number of variables and constraints, the following optimization model considers only the set of n' customers served in the infeasible routes, denoted hereafter as C', and, consequently, set $V' = C' \cup \{0\}$ as the set of vertices, with cardinality $|V'| = n' + 1$. Correspondingly, K'_{min} and K'_{max} denote the lower and upper bounds, respectively, on the number of vehicles required to serve all the customers in set C'. K'_{min} is set equal to 1, whereas K'_{max} is computed along the lines described above for the NISA-NC. The 2L-SCVRP can be cast as the following ILP model:

$$\min \quad w = \sum_{i \in V' \setminus \{n'\}} \sum_{j \in V' : j > i} c_{ij} x_{ij} - \sum_{R' \in \mathcal{R}'_{inf}} \sum_{o \in \mathcal{O}_{R'}} p_o y_o \tag{11}$$

subject to

$$\sum_{j \in C'} x_{0j} = \sum_{k = K'_{min}}^{K'_{max}} 2k z_k \tag{12}$$

$$\sum_{k = K'_{min}}^{K'_{max}} z_k = 1 \tag{13}$$

$$\sum_{i \in V' : i < j} x_{ij} + \sum_{h \in V' : h > j} x_{jh} = 2 y_o \qquad j \in o, \quad o \in \mathcal{O}_{R'}, \quad R' \in \mathcal{R}'_{inf} \tag{14}$$

$$\sum_{o \in \mathcal{O}_{R'}} y_o = 1 \qquad R' \in \mathcal{R}'_{inf} \tag{15}$$

$$\sum_{j \in S} \sum_{i \notin S : i < j} x_{ij} + \sum_{j \in S} \sum_{h \notin S : h > j} x_{jh} \geq 2 \sum_{o \in \mathcal{O}_{R'} : o \bigcap S = \emptyset} y_o \qquad S \subset V', \quad 0 \in S, \quad R' \in \mathcal{R}'_{inf} \tag{16}$$

$$\sum_{i \in S} \sum_{j \in S : j > i} x_{ij} \leq |S| - \left\lceil \max \left\{ \frac{\sum_{j \in S} a_j}{A}, \frac{\sum_{j \in S} q_j}{Q} \right\} \right\rceil \qquad S \subseteq C', 2 \leq |S| \leq n \tag{17}$$

$$\sum_{(i,j) \in R} x_{ij} \leq |R| - 1 \qquad R \in \mathcal{R}_{inf} \tag{18}$$

$$x_{ij} \in \{0, 1\} \qquad 1 \leq i < j \leq n' \tag{19}$$

$$x_{0j} \in \{0, 1, 2\} \qquad j \in C' \tag{20}$$

$$z_k \in \{0, 1\} \qquad k = K'_{min}, K'_{min} + 1, \ldots, K'_{max} \tag{21}$$

$$y_o \in \{0, 1\} \qquad o \in \mathcal{O}_{R'}, \quad R' \in \mathcal{R}'_{inf}. \tag{22}$$

Objective function (11) aims at minimizing the difference between the total cost of routing the infeasible customers and the total profit associated with the options selected. Constraints (12), (13), (17), and (18) have the same meaning of constraints (2), (3), (5), and (6), respectively. Constraint (14) ensures that if option $o \in \mathcal{O}_{R'}$ is chosen for infeasible route $R' \in \mathcal{R}'_{inf}$, then each vertex j included in that option is incident to exactly two edges in any feasible solution. For each infeasible route $R' \in \mathcal{R}'_{inf}$, constraint (15) imposes that exactly one option is selected, i.e., each infeasible route has to be made feasible removing the customers included in one option. Constraints (16) force each subset of customers corresponding to a selected option to be reachable from vertex 0 by means of two edge-disjoint paths. Note that constraints (16) are redundant given constraints (17), but are stronger than the latter when $\left\lceil \max \left\{ \frac{\sum_{j \in S} a_j}{A}, \frac{\sum_{j \in S} q_j}{Q} \right\} \right\rceil = 1$. Finally, constraints (19), (20), (21) and (22) define the decision variables.

Following the classification of the VRPs with profits provided in Archetti *et al.* [5], the above

2L-SCVRP can be classified as a capacitated profitable tour problem with multiple vehicles.

4 Packing problems

In this section we discuss the methodology designed to solve the packing problems that are generated during the execution of the algorithms described above. Different approaches are used to quickly detect if a route is feasible or infeasible. As detailed in the following, the 2OPP and the 2OPPUL are sequentially considered. The 2OPP is considered first because is a simpler problem to handle than the 2OPPUL (the former is a relaxation of the latter obtained removing the unloading constraints).

The procedures listed below are called sequentially until the infeasibility of the 2OPP (and, consequently, of the 2OPPUL) is proven:

1. a simple lower bound is obtained by summing up the areas of all items corresponding to the customers visited in the route. This route is infeasible if the sum exceeds the loading area;

2. the more sophisticated lower bound L_{dff}^{BM} on the required height of the loading area, taken from Boschetti and Montaletti [10], is then used. If this bound is larger than the height H of the loading area, the route is infeasible. It should be noted that only the first three dual feasible functions are used here (see [10] for more details). The fourth one, which proved to be time consuming and not really effective during preliminary tests, has been disregarded;

3. another lower bound on the required height of the loading area is obtained by invoking the alternating constructive procedure reported in Alvarez-Valdes *et al.* [2];

4. two additional lower bounds on the height and width of the loading area are based on the Gilmore-Gomory formulation (see [26]) of the cutting stock problem. They correspond to L_3^H and L_3^W detailed in Côté *et al.* [16]. If these bounds are larger than H and W, respectively, the route is infeasible;

5. the One-dimensional Contiguous Bin Packing problem (1CBP), a tight relaxation of the 2OPP, is finally solved with the branch-and-bound algorithm in Côté *et al.* [17]. If the 1CBP has no feasible solution, the route is infeasible.

If, after carrying out the procedures described above, the 2OPP has not been proven to be infeasible, we consider the 2OPPUL and apply sequentially the following procedures to determine whether a feasible solution exists:

1. the 2OPPUL is first solved by means of an approximate method, namely a variant of the heuristic reported in Leung *et al.* [29], originally developed for the 2SPP. The heuristic is a two-phase algorithm, where a solution is first constructed and then improved with a simulated annealing algorithm. In our procedure, the original construction heuristic is replaced by the Bottom-Left and Max-Touching Parameter heuristics described in Dominguez *et al.* [20] to address the unloading constraints. If a feasible packing is found with this heuristic, the route is feasible;

2. the lower bound L_2 for the 2OPPUL, reported in Côté *et al.* [16], is used to estimate the required area. If the value of L_2 exceeds the loading area A, the route is infeasible;

3. the branch-and-bound algorithm described in Boschetti and Montaletti [10], originally developed for the 2SPP, has been adapted to the 2OPPUL. It is applied with the following additional fathoming criterion: if an item does not fit in any position among a set of previously calculated positions, the current partial solution cannot lead to any feasible solution, and then the node can be fathomed. In practice, this algorithm can often find feasible solutions very quickly. We allow the generation of a maximum of 1000000 nodes in the branching tree before stopping the algorithm. If the algorithm returns a feasible packing, the route is feasible. If the algorithm ends without finding any feasible packing, two situations may occur. If the algorithm stops because it explored the maximum number of nodes allowed without finding a feasible packing, the packing problem is not proved to be infeasible and the algorithm performs the procedure described below. Conversely, if the algorithm ends without the need to explore all the nodes allowed, the packing problem is proved to be infeasible and, consequently, the route is infeasible;

4. the exact algorithm designed in Côté *et al.* [16] for solving the 2OPPUL is finally applied. It is based on a mathematical formulation for the 1CBP where some constraints are added to satisfy the unloading requirements. In practice, this algorithm proved to be very good in detecting infeasibility in short computation times.

After running the procedures described above, the algorithm has determined if the packing problem is feasible or infeasible and, if feasible, the corresponding solution.

5 Worst-Case Analysis

In this section we study the worst-case performance of the not integrated approaches described in Section 3. To the sake of clarity, the analysis is provided for the NISA-NC. Nevertheless, the same analysis applies to the NISA-OM, after some straightforward adjustments that are mentioned at the end of this section.

To simplify the exposition, we assume that each customer demands exactly 1 item.

Henceforth, z^* denotes the cost of an optimal solution of the 2L-CVRP, whereas we refer to the cost of the best solution found by the NISA-NC as z^H_{NISA}. We first show that this not integrated approach finds an optimal solution for the 2L-CVRP whenever it is applied to solve an instance comprising at most 2 customers. Then, we prove that the worst-case ratio is bounded by 2 in the general case that considers $n \geq 3$, and, finally, that this bound is tight.

Theorem 1 *If $n \leq 2$, then $z^H_{NISA} = z^*$.*

Proof. The claim is trivial for $n = 1$ since the cycle (0,1,0) is the optimal route for both the CVRP and the 2L-CVRP.

For $n = 2$, two cases can occur.

`Case 1.` If the cycle (0,1,2,0) is feasible for the 2L-CVRP, then it also feasible (and optimal) for the CVRP, and $z^H_{NISA} = z^*$.

`Case 2.` If the cycle (0,1,2,0) is not feasible for the 2L-CVRP, then the optimal solution for the latter problem consists of the two cycles (0,1,0) and (0,2,0). On the other side, cycle (0,1,2,0) can be either not feasible for the CVRP, or feasible for the CVRP but not feasible when the loading constraints are considered. In both cases, the aforementioned two cycles (0,1,0) and (0,2,0) form the solution found applying the not integrated approach. □

We now prove that the worst-case ratio for the NISA-NC is bounded by 2. Let us denote the cost of an optimal solution of the CVRP obtained in Phase 1 as z_{P1}, whereas let z_{P2} denote the optimal cost of the routes traveled to serve the infeasible customers identified in Phase 2. Trivially, the cost of the solution found using the NISA-NC is less or equal than the sum of the former two costs, i.e., $z^H_{NISA} \leq z_{P1} + z_{P2}$, where the equality holds if the CVRP solution is feasible for the loading constraints, i.e., $z_{P2} = 0$.

Theorem 2 *If $n \geq 3$, then $\frac{z^H_{NISA}}{z^*} \leq 2$, and this bound is tight.*

Proof. We can write the following chain of inequalities
$\frac{z^H_{NISA}}{z^*} \leq \frac{z_{P1}+z_{P2}}{z^*} \leq \frac{z^*+z^*}{z^*} = 2,$
where the first inequality is due to the remark reported above, whereas the second inequality is due to the observation that both z_{P1} and z_{P2} are lower bounds to the optimal cost of the 2L-CVRP.

To prove that the bound is tight, see the instance shown in Figure 1. Graph G is depicted in Figure 1(a). Figure 1(b) shows the size and orientation of each item demanded by the customers, whereas Figure 1(c) illustrates the loading area of a vehicle. Items are loaded from the top of the picture. To the sake of simplicity, we assume that the total weight of the items is not binding.

We first consider the NISA-NC. Figure 2(a) shows an optimal route for the CVRP of the instance depicted in Figure 1. Let us choose customer 1 as the first customer to be visited in the route (the opposite case, i.e., when customer 3 is chosen, is symmetric). The route in Figure 2(a) corresponds to an infeasible loading pattern, as shown in Figure 2(b). Let customer 1 be the vertex to remove (customers 1 and 3 are both located at the same distance from the depot). A feasible solution for the 2L-CVRP obtained applying the NISA-NC is illustrated in Figure 2(c). Its cost is $z^H_{NISA} = 4M + 2\varepsilon$. Note that an alternative solution having the same cost can be obtained removing customer 3 instead of customer 1. A feasible solution for the 2L-CVRP is shown in Figure 3. Its cost is $z^* = 2M + 3\varepsilon$.

□

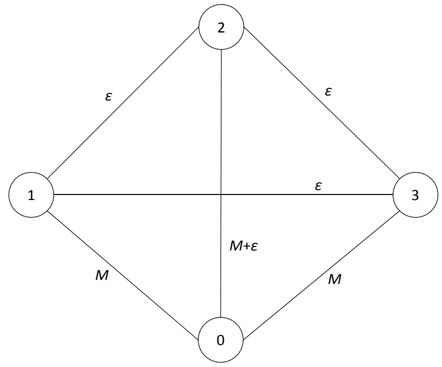
The proofs of Theorem 1 1 and of the bound in 2 hold also for the NISA-OM. The instance depicted in Figure 1 can be used to prove that the worst-case ratio of 2 is tight also for the NISA-OM. Given the optimal route for the CVRP shown in Figure 2(a), three options are available that consist in removing customer 1, or customer 2, or customer 3, respectively. The profit associated with each of the former options is 0, ε, and 0, respectively, whereas the corresponding values of objective function (11) are $2M$, $2M + \varepsilon$, and $2M$, i.e., either the first or the third option is chosen leading to the same set of solutions found using the NISA-NC.

6 Experimental analysis

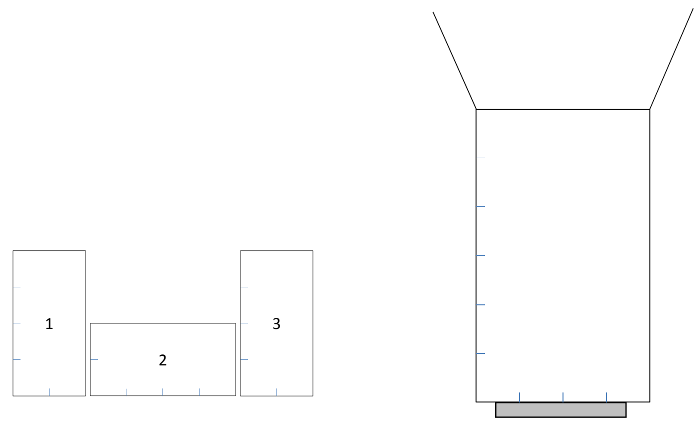
This section is devoted to the presentation and discussion of the computational experiments. They were conducted on an Intel 2.67 GHz processor running Scientific Linux 6.3 as Operating System. The algorithms were coded in C++, and the B&C algorithm was embedded within the framework provided by the CPLEX 12.6.0.1 solver using the default parameters. In Section 6.1 we describe the instances we tested, whereas Section 6.2 provides detailed computational results.

6.1 Testing Environment

In the computational experiments we used two data sets. The first data set comprises a subset of the benchmark instances for the 2L-CVRP, whereas the second one is composed of test problems



(a) Graph G with 3 customers.



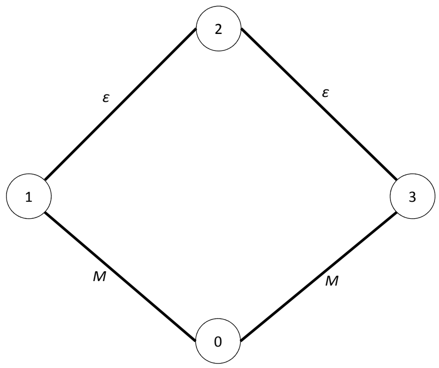
(b) Size and orientation of each item demanded by the customers.

(c) Loading area of a vehicle.

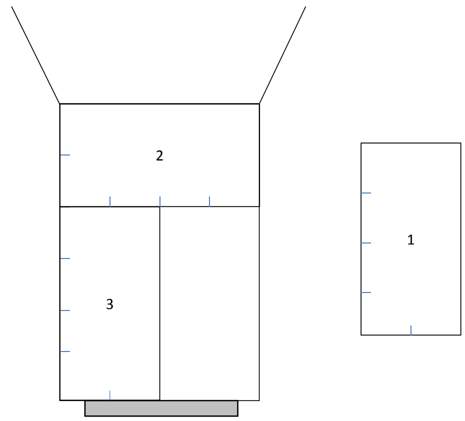
Figure 1: Instance with $n = 3$ where distances satisfy the triangle inequality and $\frac{z^H_{NISA}}{z^*} \to 2$ for $\varepsilon \to 0$.

that we generated as detailed in the following. Altogether, the different approaches were compared on 148 instances, ranging from small-scale (i.e., 15 customers and 24 items) to instances with 75 customers and 202 items.

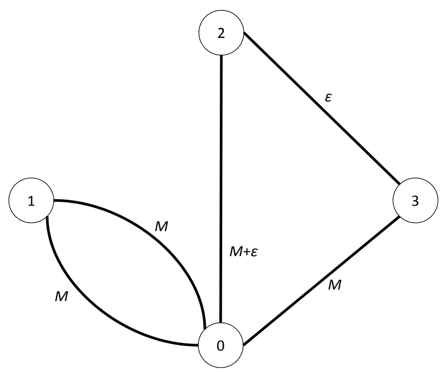
The set of benchmark instances for the 2L-CVRP (publicly available at `http://www.or.deis.unibo.it/research.html`) currently includes 180 instances. A subset of these instances were first tested in Iori *et al.* [28], where test problems comprising up to 35 customers and 114 items were solved to proven optimality by means of a B&C algorithm. The remaining benchmark instances were first solved in Gendreau *et al.* [25] by means of a tabu search heuristic able to solve instances with up to 255 customers and 786 items. Since large-scale instances for the 2L-CVRP can be currently solved only by means of a heuristic algorithm, we limited the experiments to the *benchmark instances* that include no more than 75 customers and 202 items. In order to make the paper self-contained, we briefly describe the characteristics of the benchmark instances, and refer the reader to Iori *et al.* [28] for any further detail. The benchmark instances were generated by extending to the 2L-CVRP



(a) Optimal route for the CVRP, $z_{P1} = 2M + 2\varepsilon$.



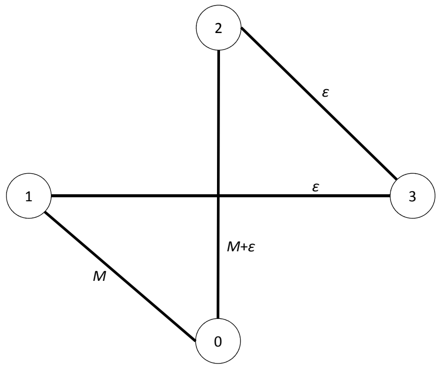
(b) Infeasible loading pattern corresponding to the route in Figure 2(a).



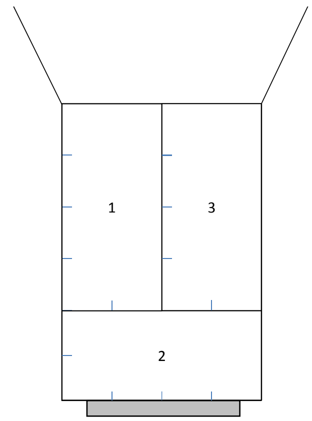
(c) Feasible solution for the 2L-CVRP.

Figure 2: Feasible solution for the 2L-CVRP of the instance in Figure 1 obtained applying the NISA-NC, $z^H_{NISA} = 4M + 2\varepsilon$.

classical instances for the CVRP. In each 2L-CVRP instance, the coordinates of each vertex, the total weight q_j associated with each customer $j \in C$, as well as the maximum weight capacity Q of the vehicles, are those of the corresponding CVRP instance. The traveling cost c_{ij} associated with each edge $\{i, j\} \in E$ is an integer value computed by rounding down to the next integer the Euclidean distance between vertex i and vertex j. From each CVRP instance, 5 2L-CVRP instances were created according to 5 different modes used to generate the items demanded by each customer. This gives rise to 5 different classes of instances, out of which we did not consider those belonging to Class 1 (as called in [28]) since they correspond to the original CVRP instances (i.e., the loading constraints are not binding) and, therefore, do not provide any insight on the benefits of integration. As for the remaining four classes (from Class 2 to Class 5), the size of the loading area of each vehicle is computed setting $W = 20$ and $H = 40$. The number of items associated with each customer (i.e., parameter m_j), the shape of each item (i.e., vertical, homogeneous, or horizontal), as well as its height and width, were randomly generated within the intervals reported in Table 1. The subset of benchmark instances that we considered in the computational experiments consists of 84 instances.



(a) Optimal route for the 2L-CVRP.



(b) Loading pattern for the solution in Figure 3(a).

Figure 3: Feasible solution for the 2L-CVRP of the instance in Figure 1, $z^* \leq 2M + 3\varepsilon$.

Class	m_j	Vertical Height	Vertical Width	Homogeneous Height	Homogeneous Width	Horizontal Height	Horizontal Width
2	[1, 2]	[.4H, .9H]	[.1H, .2H]	[.2H, .5H]	[.2H, .5H]	[.1H, .2H]	[.4H, .9H]
3	[1, 3]	[.3H, .8H]	[.1H, .2H]	[.2H, .4H]	[.2H, .4H]	[.1H, .2H]	[.3H, .9H]
4	[1, 4]	[.2H, .7H]	[.1H, .2H]	[.1H, .4H]	[.1H, .4H]	[.1H, .2H]	[.2H, .7H]
5	[1, 5]	[.1H, .6H]	[.1H, .2H]	[.1H, .3H]	[.1H, .3H]	[.1H, .2H]	[.1H, .6H]

Table 1: Intervals used to generate the benchmark instances (from Iori *et al.* [28]).

In order to validate the importance of an integrated approach for the 2L-CVRP testing a large set of instances, we created an additional set of test problems that comprises 64 instances. These instances, referred in the following as the *new instances*, are derived from the benchmark instances for the 2L-CVRP selecting a subset of the customers from the instances having from 75 to 255 customers. The number of selected customers ranges from 22 to 71, and correspond to the first customers listed in the instances.

Finally, we set a maximum computing time equal to 14400 seconds (i.e., 4 hours) for the ISA, as well as for both the NISA-NC and the NISA-OM.

6.2 Computational Results

In this section we provide and comment the foremost outcomes of the computational experiments. As the largest-scale instances that we tested were not solved to proven optimality by the ISA and, for some of them, both NISAs reached the time threshold before terminating the computation, we decided to organize the exposition of the experimental results as follows. Table 2 summarizes a set of statistics that were computed over only the instances solved to proven optimality by the ISA within the time limit. On the other hand, Table 3 considers all the instances such that both NISAs found a feasible solution for the 2L-CVRP within the time limit (this condition happened for both NISAs on approximately the same subset of instances). Finally, the detailed computational results for each of the benchmark and of the new instances can be found in Tables 4 and 5, respectively.

In Table 2 we present the results for the benchmark and the new instances, and, within each of these two groups, we summarize the results for each of the four classes of instances that we mentioned earlier. Specifically, in Table 2 we report, for each class, the number of instances tested (column with header *# Tot. Inst.*) and the number of instances solved to proven optimality by the

Class	# Tot. Inst.	# Opt. Sol.	ISA %weight	ISA %area	ISA CPU (sec.)	NISA-NC z^{UB} Gap %	NISA-NC Worst Gap %	NISA-NC %weight	NISA-NC %area	NISA-NC CPU (sec.)	NISA-OM z^{UB} Gap %	NISA-OM Worst Gap %	NISA-OM %weight	NISA-OM %area	NISA-OM CPU (sec.)
Benchmark instances															
2	21	12	77.97%	62.12%	1127.4	10.81%	23.00%	67.08%	52.46%	17.9	7.81%	13.10%	68.48%	54.00%	32.9
3	21	13	76.10%	66.86%	159.0	9.64%	24.11%	68.18%	58.96%	11.4	7.96%	22.84%	68.18%	58.96%	38.1
4	21	13	74.31%	69.10%	88.2	6.74%	15.79%	67.81%	61.51%	33.7	5.90%	15.79%	67.42%	60.81%	52.6
5	21	14	79.82%	63.20%	735.0	3.18%	21.63%	77.04%	59.84%	34.3	3.12%	21.63%	77.04%	59.84%	48.6
Average			*77.09%*	*65.34%*	*519.9*	*7.45%*	*21.11%*	*70.22%*	*58.33%*	*24.6*	*6.11%*	*18.50%*	*70.44%*	*58.51%*	*43.4*
New instances															
2	16	4	62.46%	70.71%	840.7	6.79%	12.90%	54.98%	62.89%	36.8	3.36%	6.36%	57.90%	65.79%	71.1
3	16	6	60.58%	75.38%	588.2	8.89%	14.22%	54.15%	68.03%	15.5	6.86%	11.85%	53.90%	68.22%	67.8
4	16	8	54.48%	82.96%	1968.6	11.01%	34.21%	48.45%	72.36%	10.3	7.73%	11.83%	48.87%	73.58%	80.6
5	16	13	57.00%	84.99%	1965.9	7.57%	27.53%	52.40%	76.48%	298.2	6.83%	21.91%	52.40%	76.48%	529.5
Average			*57.75%*	*80.76%*	*1554.8*	*8.61%*	*24.79%*	*52.05%*	*72.03%*	*135.5*	*6.62%*	*15.36%*	*52.49%*	*72.75%*	*265.1*

Table 2: ISA optimal solutions: A summary of the computational results for Classes 2-5.

ISA (column # *Opt. Sol.*). The latter figure represents the number of instances considered when computing the statistics described below. The following three columns in the table are devoted to the ISA. Columns *%weight* and *%area* give the average load factors of the vehicles routed in terms of the weight capacity used and the loading area occupied, respectively. Column *CPU (sec.)* shows the average computing time in seconds. The subsequent 5 columns illustrate the average results for the NISA-NC. In addition to the three statistics described for the ISA, we report two further figures. Column z^{UB} *Gap %* refers to the average error of the best solution value found by the NISA-NC compared to the optimal solution value found by the ISA. The error for each instance is computed as $100\frac{z^H_{NISA}-z^{UB}}{z^{UB}}$, where z^H_{NISA} denotes the cost of the best feasible solution found by the NISA-NC and z^{UB} refers to the value of the optimal solution determined using the ISA. The error computed for each instance is then averaged over all the instances considered for a given class to obtain statistic z^{UB} Gap %. Statistic *Worst Gap %* shows the worst error computed out of all the selected instances in the class. The last 5 columns show the same set of statistics for the NISA-OM, with the only difference that here z^H_{NISA} refers to the cost of the best feasible solution found by the NISA-OM.

The figures reported in Table 2 show that the error committed adopting one of the proposed NISAs is, on average, large. Indeed, considering the NISA-NC, statistic 'z^{UB} Gap %' is on average equal to 7.45% for the benchmark instances, whereas it is approximately 8.6% for the new instances. The same statistic takes a rather large value, even if slightly smaller, also for the NISA-OM (approximately 6.1% and 6.6% for the benchmark and the new instances, respectively). This result indicates that even a heuristic that does not provide very accurate results but that addresses the 2L-CVRP as an integrated problem, say one that finds on average solutions that are within 3-4% from the optimum, can halve the error committed by the two NISAs. The worst error committed using the NISA-NC is always larger than 12.90% and, often, larger than 20% (see the figures reported in column 'Worst Gap %'). The values concerning the worst error improve using the NISA-OM, sometimes quite significantly (e.g., see the figures referred to Class 2 for the benchmark instances where the worst error is reduced by approximately 10%), but remain remarkably large. Addressing the 2L-CVRP by means of the integrated approach often provides solutions where a smaller number of vehicles is routed compared to those found using the proposed NISAs. To this aim, one can compare in Tables 4 and 5 the number of vehicles routed in the best solution found by the ISA (column # *Veh.*) with the additional number of vehicles routed in each solution computed with the two NISAs (column *Add. Veh.*). As a consequence, the average load factors computed for the solutions obtained using the ISA are considerably better than those concerning the NISAs, both in terms of weight and area. Indeed, adopting the ISA instead of the NISA-NC, the average improvement achieved for the benchmark instances is approximately equal to 7% for both load factors, while for the new instances it is almost equal to 6% for statistic '%weight' and larger than 8.7% for statistic '%area'. Almost negligible improvements were found for both load factors using the NISA-OM instead of the NISA-NC. As expected, solving the 2L-CVRP by means of the ISA requires considerably larger computing efforts than using any NISA. The new instances resulted to be the most challenging in terms of average CPU times spent. As far as these instances are considered, both NISAs found a solution within few seconds, whereas the ISA took, on average, approximately 26 minutes to solve to optimality an instance. Nevertheless, it is worth noting that the ISA solved to proven optimality within fractions of a second most of the benchmark instances comprising up to 25 customers (see Table 4).

Table 3 summarizes the results computed over all the instances where both NISAs found a feasible solution for the 2L-CVRP within the time limit. In other words, compared to the results

Class	# Tot. Inst.	# Solv. NISA	ISA Opt. Gap %	ISA Worst Gap %	ISA %weight	ISA %area	ISA CPU (sec.)	NISA-NC z^{UB} Gap %	NISA-NC Worst Gap %	NISA-NC %weight	NISA-NC %area	NISA-NC CPU (sec.)	NISA-OM z^{UB} Gap %	NISA-OM Worst Gap %	NISA-OM %weight	NISA-OM %area	NISA-OM CPU (sec.)
Benchmark instances																	
2	21	16	1.34%	6.47%	73.10%	67.20%	4440.1	13.29%	28.29%	61.64%	55.58%	17.2	8.91%	15.82%	63.78%	58.09%	348.2
3	21	17	0.90%	6.23%	71.00%	70.97%	3504.8	10.33%	24.11%	62.90%	61.91%	12.1	8.32%	22.84%	63.28%	62.44%	760.3
4	21	18	1.14%	8.83%	68.07%	74.12%	4056.7	7.74%	15.79%	61.60%	65.82%	38.8	6.82%	15.79%	61.32%	65.32%	117.5
5	21	19	0.71%	6.46%	72.64%	69.09%	4326.4	3.78%	21.63%	69.66%	64.89%	752.4	3.67%	21.63%	69.66%	64.89%	811.0
Average			*1.01%*	*7.02%*	*71.17%*	*70.41%*	*4083.5*	*8.56%*	*22.25%*	*64.11%*	*62.28%*	*221.1*	*6.81%*	*19.09%*	*64.62%*	*62.85%*	*514.6*
New instances																	
2	16	10	3.85%	10.86%	50.39%	75.69%	8963.3	12.54%	25.74%	42.97%	64.22%	37.0	7.45%	19.82%	45.41%	67.74%	409.9
3	16	12	2.58%	6.66%	49.35%	79.39%	7486.5	10.98%	26.08%	42.83%	68.60%	92.3	8.25%	15.31%	43.72%	71.07%	236.2
4	16	14	1.91%	6.99%	47.03%	82.98%	7284.6	11.55%	34.21%	42.45%	74.18%	30.2	7.71%	11.83%	43.18%	76.34%	169.8
5	16	14	0.04%	0.55%	57.45%	85.36%	2853.9	7.80%	27.53%	52.84%	76.95%	357.3	7.05%	21.91%	52.54%	76.53%	590.5
Average			*1.94%*	*5.88%*	*51.18%*	*81.33%*	*6428.2*	*10.56%*	*28.69%*	*45.55%*	*71.62%*	*138.0*	*7.60%*	*17.09%*	*46.38%*	*73.41%*	*351.5*

Table 3: NISA solved instances: A summary of the computational results for Classes 2-5.

described above, some instances that were not solved to proven optimality by the ISA are also included. In addition to the statistics described above, this table reports, for each class, the number of instances considered while computing the statistics (*# Solv. NISA*) and two further performance measures for the ISA. Specifically, statistic *Opt. Gap %* shows the average optimality gap computed, for each instance, as $100\frac{z^{UB}-z^{LB}}{z^{LB}}$, where z^{UB} and z^{LB} are the best upper and lower bounds, respectively, found by the ISA within the time limit. Statistic *Worst Gap %* reported in the fifth column indicates the worst optimality gap computed out of all the instances considered in the class.

The importance of adopting an integrated approach to address the 2L-CVRP is even more evident from the figures reported in Table 3. Indeed, although the computation of the statistics reported in this table also considers some instances that were not solved to proven optimality by the ISA, if compared to the figures shown in Table 2 the average errors for both NISAs increase in almost all classes of instances, and for the new instances in particular. Additionally, the worst errors also deteriorate significantly, especially for the NISA-NC where statistic 'Worst Gap %' takes values larger than 25% in five classes out of eight. The comments about the remaining statistics are similar to those reported above for Table 2. Given these results, we feel we can draw the following conclusions. On the one hand, even when the solution found by the ISA is not proven to be optimal, it is a high-quality solution and, likely, very close to the optimum (or the optimum itself). In these situations, the ISA probably tends to spend an excessive amount of computing time to prove the optimality of the solution found. On the other hand, the magnitude of the benefits achieved using an integrated approach is quite remarkable on all the instances where the NISAs have been able to find a feasible solution for the 2L-CVRP.

As mentioned above, Tables 4 and 5 report the detailed computational results for each of the benchmark and of the new instances, respectively. Note that in columns 'CPU (sec.)' we indicated with *T.L.* whenever the corresponding approach did not terminate the computation within the time threshold. It is worth highlighting that in the large majority of these cases, for both NISAs the time limit was reached while still solving a CVRP. On the other side, in all the instances that were not solved by the NISA-OM, but were by the NISA-NC, the time limit was reached by the former approach while solving the 2L-SCVRP. As a consequence, for all these instances the NISAs did not find any feasible solution for the 2L-CVRP within the time limit allowed (*n/a* indicates this occurrence). This remark gives rise to another crucial concern about the use of a not integrated approach. When one of the problems that is addressed independently is computationally hard to solve, as for the CVRP, and limited computing resources are available (not only in terms of time but also memory) it may happen that a not integrated approach is not able to find any feasible solution for the integrated problem. This situation can occur even for some instances for which an integrated approach, which searches from the beginning for a solution of the integrated problem, obtains a feasible solution, sometimes without proving its optimality.

7 Conclusions

A growing body of literature focuses on the study of integrated vehicle routing problems, where classical vehicle routing problems are considered in combination with other optimization problems. This recent trend is motivated by the desire of bridging the gap between academic research and real-world applications, on the one side, and by the advances in optimization methods and computer capabilities, on the other side. The integrated problems are, however, computationally very hard and, to motivate the use of an integrated approach, the savings that can be achieved by tackling

Instance Details			ISA				NISA-NC			NISA-OM		
				Opt.	#	CPU	z^{UB}	Add.	CPU	z^{UB}	Add.	CPU
Name	n	m	z^{UB}	**Gap %**	**Veh.**	**(sec.)**	**Gap %**	**Veh.**	**(sec.)**	**Gap %**	**Veh.**	**(sec.)**
0102	15	24	277	0.00%	4	0.3	6.14%	0	1.1	6.14%	0	4.4
0103	15	31	280	0.00%	3	0.9	10.71%	+1	1.3	2.86%	+1	9.4
0104	15	37	288	0.00%	4	0.4	6.94%	+1	1.3	6.25%	+1	5.4
0105	15	45	273	0.00%	3	0.5	0.00%	0	1.3	0.00%	0	1.3
0202	15	25	337	0.00%	6	0.2	8.31%	+1	1.3	7.72%	+1	5.2
0203	15	31	345	0.00%	6	0.6	7.25%	+1	1.4	6.38%	+1	5.1
0204	15	40	326	0.00%	6	0.1	0.00%	0	0.8	0.00%	0	1.1
0205	15	48	326	0.00%	6	0.1	0.00%	0	0.9	0.00%	0	0.9
0302	20	29	396	0.00%	5	172.1	21.97%	+2	8.0	11.87%	+1	17.7
0303	20	46	387	0.00%	5	5.5	10.59%	+1	3.1	10.59%	+1	52.0
0304	20	44	360	0.00%	5	2.9	15.28%	0	2.5	6.11%	0	28.5
0305	20	49	351	0.00%	4	0.3	0.00%	0	1.4	0.00%	0	1.6
0402	20	32	434	0.00%	6	0.9	7.14%	+1	1.7	7.14%	+1	4.2
0403	20	43	432	0.00%	7	0.8	7.64%	0	1.8	4.86%	0	8.5
0404	20	50	433	0.00%	7	0.7	0.00%	0	2.0	0.00%	0	2.8
0405	20	62	423	0.00%	6	0.1	0.00%	0	1.5	0.00%	0	1.5
0502	21	31	380	0.00%	4	0.3	7.37%	+1	1.8	7.11%	+1	19.0
0503	21	37	373	0.00%	4	0.1	12.33%	+1	1.6	12.33%	+1	6.2
0504	21	41	377	0.00%	4	0.2	3.18%	+1	1.9	2.92%	+1	6.3
0505	21	57	367	0.00%	4	0.2	0.00%	0	1.5	0.00%	0	1.6
0602	21	33	491	0.00%	6	0.4	12.02%	+1	2.1	9.57%	+1	6.8
0603	21	40	492	0.00%	7	0.9	7.93%	0	1.8	7.93%	0	7.0
0604	21	57	489	0.00%	6	0.3	0.00%	0	1.6	0.00%	0	1.7
0605	21	56	488	0.00%	6	0.2	0.00%	0	1.7	0.00%	0	1.7
0702	22	32	724	0.00%	5	87.2	12.43%	+1	4.4	7.60%	+1	39.2
0703	22	41	693	0.00%	4	7.8	8.51%	+1	3.9	7.50%	+1	55.3
0704	22	51	693	0.00%	4	2.6	8.23%	+1	2.6	8.23%	+1	51.6
0705	22	55	647	0.00%	4	12.0	6.03%	+1	7.0	6.03%	+1	30.1
0802	22	29	716	0.00%	4	119.3	15.50%	+2	6.3	7.68%	+1	29.9
0803	22	42	730	0.00%	5	6.3	7.67%	+1	2.6	7.67%	+1	32.0
0804	22	48	687	0.00%	4	3.8	14.12%	+1	3.6	14.12%	+1	49.1
0805	22	52	605	0.00%	3	237.9	10.58%	+1	304.2	9.75%	+1	346.7
0902	25	40	598	0.00%	8	0.3	8.19%	+1	2.3	8.19%	+1	3.5
0903	25	61	601	0.00%	8	0.7	4.83%	+1	2.5	4.83%	+1	6.0
0904	25	63	609	0.00%	9	3.4	6.24%	0	2.5	6.24%	0	6.6
0905	25	91	595	0.00%	8	0.2	0.00%	0	1.9	0.00%	0	3.0
1002	29	43	687	0.00%	6	12900.0	23.00%	+1	6.5	13.10%	+1	79.5
1003	29	49	616	0.00%	5	60.8	23.70%	0	7.0	15.75%	0	155.3
1004	29	72	703	0.00%	6	11.7	15.79%	+1	4.0	15.79%	+2	33.4
1005	29	86	675	0.00%	6	9735.3	21.63%	0	8.6	21.63%	0	101.1
1102	29	43	708	6.47%	6	*T.L.*	17.37%	+1	7.5	15.82%	+1	157.5
1103	29	62	705	0.00%	6	1885.5	24.11%	+1	7.4	22.84%	+1	52.9
1104	29	74	781	2.40%	7	*T.L.*	9.60%	0	12.3	9.60%	0	97.8
1105	29	91	613	0.00%	6	78.6	0.00%	0	5.4	0.00%	0	7.5
1202	30	50	591	0.00%	10	102.4	7.61%	+1	38.6	7.61%	+1	42.2
1203	30	56	587	0.00%	10	25.6	0.00%	0	51.0	0.00%	0	45.3
1204	30	82	598	0.00%	10	721.7	5.35%	+1	225.7	5.35%	+1	231.6
1205	30	101	587	0.00%	10	31.4	0.00%	0	23.4	0.00%	0	23.5
1302	32	44	2656	7.35%	6	*T.L.*	30.23%	+1	29.4	*n/a*	*n/a*	*T.L.*
1303	32	56	2472	0.20%	6	*T.L.*	8.54%	0	6.7	6.63%	0	136.3
1304	32	78	2595	0.00%	6	64.9	5.82%	+1	6.2	5.05%	+1	97.6
1305	32	102	2333	0.78%	5	*T.L.*	2.79%	+1	165.0	2.79%	+1	269.2
1402	32	47	1071	5.41%	5	*T.L.*	28.29%	+2	10.1	13.63%	+1	3792.1
1403	32	57	1023	4.49%	5	*T.L.*	13.10%	+1	5.8	12.02%	+1	69.2
1404	32	65	965	0.39%	5	*T.L.*	15.75%	+1	20.6	14.09%	+1	226.0
1405	32	87	907	6.46%	5	*T.L.*	3.64%	0	882.8	3.64%	0	499.3
1502	32	48	1028	3.99%	5	*T.L.*	19.26%	+2	23.3	11.96%	+1	171.4
1503	32	59	1165	6.23%	6	*T.L.*	18.28%	+2	16.0	10.30%	+1	221.7
1504	32	84	1226	7.01%	6	*T.L.*	8.81%	+1	17.3	8.48%	+1	192.3
1505	32	114	1143	0.00%	6	87.9	6.30%	+1	30.4	6.30%	+1	68.0
1602	35	56	682	0.00%	11	145.7	0.00%	0	141.1	0.00%	0	143.4
1603	35	74	682	0.00%	11	71.4	0.00%	0	63.3	0.00%	0	60.7
1604	35	93	691	0.00%	11	333.9	6.66%	+1	182.8	6.66%	+1	167.7
1605	35	114	682	0.00%	11	105.2	0.00%	0	90.5	0.00%	0	91.3
1702	40	60	847	3.30%	15	*T.L.*	*n/a*	*n/a*	*T.L.*	*n/a*	*n/a*	*T.L.*
1703	40	73	839	2.21%	15	*T.L.*	*n/a*	*n/a*	*T.L.*	*n/a*	*n/a*	*T.L.*
1704	40	96	839	2.01%	15	*T.L.*	*n/a*	*n/a*	*T.L.*	*n/a*	*n/a*	*T.L.*
1705	40	127	839	2.07%	15	*T.L.*	*n/a*	*n/a*	*T.L.*	*n/a*	*n/a*	*T.L.*
1802	44	66	1032	5.59%	9	*T.L.*	18.02%	+1	19.5	7.46%	+1	1055.7
1803	44	87	1082	4.32%	9	*T.L.*	10.44%	+2	28.4	8.87%	+2	12001.9
1804	44	112	1110	1.93%	9	*T.L.*	9.19%	+2	22.8	8.74%	+2	326.7
1805	44	122	910	0.92%	8	*T.L.*	1.32%	0	1656.3	1.32%	0	1746.8
1902	50	82	773	15.01%	10	*T.L.*	14.75%	+2	589.9	*n/a*	*n/a*	*T.L.*
1903	50	103	782	11.42%	11	*T.L.*	*n/a*	*n/a*	*T.L.*	*n/a*	*n/a*	*T.L.*
1904	50	134	779	8.83%	11	*T.L.*	8.34%	+1	188.2	5.13%	+1	589.3
1905	50	157	634	0.79%	8	*T.L.*	7.41%	+1	1992.3	6.62%	+1	2119.2
2002	71	104	526	17.20%	14	*T.L.*	*n/a*	*n/a*	*T.L.*	*n/a*	*n/a*	*T.L.*
2003	71	151	518	10.71%	14	*T.L.*	*n/a*	*n/a*	*T.L.*	*n/a*	*n/a*	*T.L.*
2004	71	178	527	6.30%	15	*T.L.*	*n/a*	*n/a*	*T.L.*	*n/a*	*n/a*	*T.L.*
2005	71	226	460	4.62%	12	*T.L.*	12.17%	+1	9119.8	11.74%	+1	10094.5
2102	75	114	1037	18.93%	14	*T.L.*	*n/a*	*n/a*	*T.L.*	*n/a*	*n/a*	*T.L.*
2103	75	164	1121	15.08%	17	*T.L.*	*n/a*	*n/a*	*T.L.*	*n/a*	*n/a*	*T.L.*
2104	75	168	965	10.24%	13	*T.L.*	*n/a*	*n/a*	*T.L.*	*n/a*	*n/a*	*T.L.*
2105	75	202	859	6.70%	11	*T.L.*	*n/a*	*n/a*	*T.L.*	*n/a*	*n/a*	*T.L.*
Average (NISA solved[a])				*1.01%*	*6.40*	*4083.5*	*8.56%*	*+0.71*	*221.1*	*6.81%*	*+0.66*	*514.6*
Average (Total[b])				*2.37%*	*7.54*	*5799.8*			*2251.0*			*2832.8*

a: the average is computed over the instances solved by both NISAs.
b: the average is computed over all the instances.

Table 4: Benchmark instances: A comparison of the different approaches.

Instance Details			ISA				NISA-NC			NISA-OM		
Name	n	m	z^{UB}	Opt. Gap %	# Veh.	CPU (sec.)	z^{UB} Gap %	Add. Veh.	CPU (sec.)	z^{UB} Gap %	Add. Veh.	CPU (sec.)
2102_22	22	38	388	0.00%	5	225.7	5.67%	0	2.9	1.80%	0	45.9
2103_22	22	47	382	0.00%	5	32.2	11.78%	+1	8.6	4.45%	0	85.3
2104_22	22	49	351	0.00%	5	2.0	3.42%	0	2.8	3.42%	0	36.1
2105_22	22	58	316	0.00%	3	22.4	6.33%	+1	320.5	6.33%	+1	405.3
2202_22	22	34	380	0.00%	5	148.9	1.32%	+1	3.9	1.32%	+1	57.0
2203_22	22	42	350	0.00%	5	3.5	6.86%	0	2.3	6.86%	+1	43.3
2204_22	22	58	390	0.00%	5	5.9	10.00%	+1	2.9	7.95%	+1	22.0
2205_22	22	67	336	0.00%	4	5.8	0.00%	0	4.6	0.00%	0	4.1
2302_25	25	37	403	0.00%	5	559.2	12.90%	+1	6.2	3.97%	0	41.6
2303_25	25	50	422	0.00%	5	7.1	14.22%	+1	4.6	11.85%	+1	58.8
2304_25	25	61	401	0.00%	5	7.9	5.99%	+1	5.3	5.99%	+1	54.8
2305_25	25	76	384	0.00%	5	33.3	0.00%	0	10.9	0.00%	0	18.0
2402_29	29	48	544	10.86%	8	*T.L.*	19.30%	+2	156.3	9.74%	+1	214.9
2403_29	29	61	491	0.00%	7	1858.9	7.33%	+1	14.1	5.09%	+1	49.2
2404_29	29	77	486	0.00%	7	1179.3	5.35%	0	5.2	5.35%	0	68.2
2405_29	29	84	458	0.00%	6	14.8	0.00%	0	10.9	0.00%	0	12.0
2502_29	29	45	529	6.44%	7	*T.L.*	15.12%	1	11.6	6.05%	+1	40.9
2503_29	29	57	452	0.00%	6	63.2	9.51%	0	7.1	9.29%	0	110.3
2504_29	29	72	484	0.00%	6	14.4	10.33%	+1	5.9	10.33%	+1	113.3
2505_29	29	91	430	0.00%	5	17.5	*n/a*	*n/a*	*T.L.*	*n/a*	*n/a*	*T.L.*
2602_30	30	47	338	2.35%	6	*T.L.*	25.74%	+2	7.7	19.82%	+1	1204.9
2603_30	30	59	320	2.96%	6	*T.L.*	11.88%	+1	23.3	9.06%	0	172.5
2604_30	30	80	393	6.99%	7	*T.L.*	19.08%	+1	15.6	9.16%	0	193.2
2605_30	30	99	316	0.00%	6	854.3	27.53%	+1	69.4	20.25%	+1	233.1
2702_32	32	51	544	5.30%	7	*T.L.*	13.42%	+1	18.7	8.82%	+1	295.9
2703_32	32	71	555	6.30%	7	*T.L.*	12.61%	+2	41.8	11.89%	+2	192.4
2704_32	32	92	571	4.80%	8	*T.L.*	11.91%	+1	15.5	7.88%	+1	104.8
2705_32	32	105	488	0.00%	6	668.4	7.17%	+1	587.8	7.17%	+1	734.7
2802_32	32	52	928	7.46%	7	*T.L.*	20.91%	+1	22.9	*n/a*	*n/a*	*T.L.*
2803_32	32	66	836	4.16%	6	*T.L.*	26.08%	+2	11.4	15.31%	+1	93.1
2804_32	32	78	871	0.00%	6	10288.4	34.21%	+2	10.7	11.83%	+1	131.3
2805_32	32	98	712	0.00%	5	6118.1	22.47%	+1	43.2	21.91%	+1	590.8
3502_32	32	49	110	0.00%	8	2429.0	7.27%	+1	134.2	6.36%	+1	139.9
3503_32	32	65	110	0.00%	8	1564.2	3.64%	+1	56.2	3.64%	+1	59.9
3504_32	32	86	110	0.00%	8	4146.7	9.09%	+1	41.0	7.27%	+1	64.5
3505_32	32	99	108	0.00%	8	70.5	0.00%	0	55.5	0.00%	0	53.2
2902_35	35	59	478	6.58%	7	*T.L.*	15.90%	+2	15.5	8.37%	+1	1816.9
2903_35	35	63	426	4.88%	7	*T.L.*	14.55%	+1	13.7	*n/a*	*n/a*	*T.L.*
2904_35	35	85	428	4.33%	7	*T.L.*	15.42%	+1	18.4	9.58%	0	495.1
2905_35	35	114	432	0.00%	6	1905.6	14.12%	+1	21.7	13.43%	+1	199.9
3002_35	35	52	571	7.01%	7	*T.L.*	8.76%	+1	13.0	8.23%	+1	241.0
3003_35	35	69	575	4.80%	8	*T.L.*	9.57%	+1	16.0	7.13%	+1	109.4
3004_35	35	76	528	0.00%	6	104.6	9.66%	+1	8.4	9.66%	+1	154.3
3005_35	35	100	505	0.00%	6	1816.7	3.56%	0	83.9	3.56%	0	344.6
3102_40	40	63	702	11.41%	9	*T.L.*	16.38%	+2	1069.3	*n/a*	*n/a*	*T.L.*
3103_40	40	87	672	6.09%	9	*T.L.*	6.85%	+1	750.7	6.70%	+1	998.9
3104_40	40	100	620	1.83%	8	*T.L.*	9.35%	0	62.3	5.16%	0	231.6
3105_40	40	127	588	0.00%	7	318.3	4.59%	+1	48.9	4.59%	+1	79.0
3202_44	44	65	723	11.43%	9	*T.L.*	14.25%	+2	39.5	*n/a*	*n/a*	*T.L.*
3203_44	44	94	715	6.66%	9	*T.L.*	11.47%	+2	171.3	7.69%	+1	860.7
3204_44	44	112	703	4.00%	9	*T.L.*	10.24%	+1	96.9	7.97%	+1	496.7
3205_44	44	143	633	0.00%	8	8572.2	6.79%	+1	2596.8	6.79%	+1	4010.8
3302_50	50	75	681	13.06%	10	*T.L.*	*n/a*	*n/a*	*T.L.*	*n/a*	*n/a*	*T.L.*
3303_50	50	101	738	12.57%	11	*T.L.*	*n/a*	*n/a*	*T.L.*	*n/a*	*n/a*	*T.L.*
3304_50	50	118	670	4.79%	10	*T.L.*	7.61%	0	132.2	6.42%	0	211.7
3305_50	50	132	567	0.00%	7	5156.9	5.82%	+1	22.0	4.76%	+1	198.1
3402_71	71	110	270	14.59%	15	*T.L.*	*n/a*	*n/a*	*T.L.*	*n/a*	*n/a*	*T.L.*
3403_71	71	151	266	10.91%	15	*T.L.*	*n/a*	*n/a*	*T.L.*	*n/a*	*n/a*	*T.L.*
3404_71	71	175	263	8.09%	15	*T.L.*	15.21%	+2	11473.4	*n/a*	*n/a*	*T.L.*
3405_71	71	215	212	0.55%	12	*T.L.*	10.85%	+1	1126.1	9.91%	+2	1382.8
3602_71	71	104	241	22.27%	12	*T.L.*	*n/a*	*n/a*	*T.L.*	*n/a*	*n/a*	*T.L.*
3603_71	71	143	283	30.28%	15	*T.L.*	*n/a*	*n/a*	*T.L.*	*n/a*	*n/a*	*T.L.*
3604_71	71	179	273	15.91%	14	*T.L.*	*n/a*	*n/a*	*T.L.*	*n/a*	*n/a*	*T.L.*
3605_71	71	221	222	9.83%	12	*T.L.*	*n/a*	*n/a*	*T.L.*	*n/a*	*n/a*	*T.L.*
Average (NISA solved[a])				*1.94%*	*6.64*	*6428.2*	*10.56%*	*+0.90*	*138.0*	*7.60%*	*+0.72*	*351.5*
Average (Total[b])				*4.21%*	*7.63*	*7940.9*			*2332.1*			*3432.0*

a: the average is computed over the instances solved by both NISAs.
b: the average is computed over all the instances.

Table 5: New instances: A comparison of the different approaches.

an integrated problem directly rather than decomposing it and addressing each problem separately must be assessed.

In this paper we have considered the capacitated vehicle routing problem with two-dimensional loading constraints, which is an integrated problem where the capacitated vehicle routing problem is combined with the problem of finding a feasible loading pattern for a set of rectangular-shaped items. We have proposed a solution approach that addresses the integrated problem by means of an exact algorithm and compared such approach with two not integrated approaches that consider the routing and the loading aspects of the problem sequentially. We have shown that the cost of solution obtained with a not integrated approach may be as large as twice the cost of an optimal integrated solution. We have also shown empirically the importance of the integration for this problem. The computational results have shown that the integrated problem provides better solutions, both in terms of total cost, number of vehicles routed and loading factors achieved. In particular, on the instances where an optimal solution is found for the integrated approach, the average cost increase of the two not integrated approaches is 7.45% and 6.11%, respectively, for the benchmark instances, whereas it is 8.61% and 6.62% for the new test problems.

The results obtained suggest that it is worthwhile to jointly tackle strongly interdependent problems that have been, until recently, addressed separately and that this is true even in the cases the integrated problems cannot be solved exactly, provided the error generated by a heuristic remains smaller than the cost increase of a not integrated approach. The line of research proposed in this paper can be extended to other integrated problems to motivate their scientific study and to evaluate the potential benefit of their implementation in practice.

References

[1] Y. Adulyasak, J.-F. Cordeau, and R. Jans. The production routing problem: A review of formulations and solution algorithms. *Computers & Operations Research*, 55(0):141–152, 2015.

[2] R. Alvarez-Valdes, F. Parreño, and J.M. Tamarit. A branch and bound algorithm for the strip packing problem. *OR Spectrum*, 31(2):431–459, 2009.

[3] Y. Arahori, T. Imamichi, and H. Nagamochi. An exact strip packing algorithm based on canonical forms. *Computers & Operation Research*, 39(12):2991–3011, 2012.

[4] C. Archetti and M.G. Speranza. The value of integration in logistics. In *Proceedings of the 28th European Conference on Modelling & Simulation, ECMS 2014*, pages 691–697, 2014.

[5] C. Archetti, M.G. Speranza, and D. Vigo. Vehicle routing problems with profits. In P. Toth and D. Vigo, editors, *Vehicle Routing: Problems, Methods, and Applications, 2nd Edition*, pages 273–297. SIAM, 2014.

[6] T. Bektaş, G. Laporte, and D. Vigo. Integrated vehicle routing problems. *Computers & Operations Research*, 55(0):126, 2015.

[7] G. Belov and H. Rohling. LP bounds in an interval-graph algorithm for orthogonal-packing feasibility. *Operations Research*, 61(2):483–497, 2013.

[8] L. Bertazzi and M.G. Speranza. Inventory routing problems: An introduction. *EURO Journal on Transportation and Logistics*, 1(4):307–326, 2012.

[9] A. Bortfeldt and G. Wäscher. Container loading problems - A state-of-the-art review. *European Journal of Operational Research*, 229(1):1–20, 2013.

[10] M.A. Boschetti and L. Montaletti. An exact algorithm for the two-dimensional strip-packing problem. *Operations Research*, 58(6):1774–1791, 2010.

[11] A. Caprara and M. Monaci. Bidimensional packing by bilinear programming. *Mathematical Programming*, 118(1):75–108, 2009.

[12] P. Chandra and M.L. Fisher. Coordination of production and distribution planning. *European Journal of Operational Research*, 72(3):503–517, 1994.

[13] F. Clautiaux, J. Carlier, and A. Moukrim. A new exact method for the two-dimensional orthogonal packing problem. *European Journal of Operational Research*, 183(3):1196–1211, 2007.

[14] L.C. Coelho, J.-F. Cordeau, and G. Laporte. Thirty years of inventory routing. *Transportation Science*, 48(1):1–19, 2013.

[15] J.-F. Côté, M. Gendreau, and J.-Y. Potvin. The vehicle routing problem with stochastic two-dimensional items. *Technical report, CIRRELT-2013-84*, 2013. Submitted.

[16] J.-F. Côté, M. Gendreau, and J.-Y. Potvin. An exact algorithm for the two-dimensional orthogonal packing problem with unloading constraints. *Operations Research*, 62(5):1126–1141, 2014.

[17] J.-F. Côté, M. Iori, and M. Dell'Amico. Combinatorial benders' cuts for the strip packing problem. *Operations Research*, 62(3):643–661, 2014.

[18] R. Cuda, G. Guastaroba, and M.G. Speranza. A survey on two-echelon routing problems. *Computers & Operations Research*, 55(0):185–199, 2015.

[19] J.L.M. da Silveira, F.K. Miyazawa, and E.C. Xavier. Heuristics for the strip packing problem with unloading constraints. *Computer & Operations Research*, 40(4):991–1003, 2013.

[20] O. Dominguez, A.A. Juan, B. Barrios, J. Faulin, and A. Agustin. Using biased randomization for solving the two-dimensional loading vehicle routing problem with heterogeneous fleet. *Annals of Operations Research*, In Press. DOI: 10.1007/s10479-014-1551-4.

[21] M. Drexl and M. Schneider. A survey of variants and extensions of the location-routing problem. *European Journal of Operational Research*, 241(2):283–308, 2015.

[22] C. Duhamel, P. Lacomme, A. Quilliot, and H. Toussaint. A multi-start evolutionary local search for the two-dimensional loading capacitated vehicle routing problem. *Computers & Operations Research*, 38(3):617–640, 2011.

[23] G. Fuellerer, K.F. Doerner, R.F. Hartl, and M. Iori. Ant colony optimization for the two-dimensional loading vehicle routing problem. *Computers & Operations Research*, 36(3):655–673, 2009.

[24] M. Gendreau, M. Iori, G. Laporte, and S. Martello. A tabu search algorithm for a routing and container loading problem. *Transportation Science*, 40(3):342–350, 2006.

[25] M. Gendreau, M. Iori, G. Laporte, and S. Martello. A tabu search heuristic for the vehicle routing problem with two-dimensional loading constraints. *Networks*, 51(1):4–18, 2008.

[26] P.C. Gilmore and R.E. Gomory. A linear programming approach to the cutting-stock problem. *Operations Research*, 9(6):849–859, 1961.

[27] M. Iori and S. Martello. Routing problems with loading constraints. *TOP*, 18(1):4–27, 2010.

[28] M. Iori, J.-J. Salazar-González, and D. Vigo. An exact approach for the vehicle routing problem with two-dimensional loading constraints. *Transportation Science*, 41(2):253–264, 2007.

[29] C.H.S. Leung, D. Zhang, and M.S. Kwang. A two-stage intelligent search algorithm for the two-dimensional strip packing problem. *European Journal of Operational Research*, 215(1):57–69, 2011.

[30] S.C.H. Leung, Z. Zhang, D. Zhang, X. Hua, and M.K. Lim. A meta-heuristic algorithm for heterogeneous fleet vehicle routing problems with two-dimensional loading constraints. *European Journal of Operational Research*, 225(2):199–210, 2013.

[31] S.C.H. Leung, X. Zhou, D. Zhang, and J. Zheng. Extended guided tabu search and a new packing algorithm for the two-dimensional loading vehicle routing problem. *Computers & Operations Research*, 38(1):205–215, 2011.

[32] J. Lysgaard, A.N. Letchford, and R.W. Eglese. A new branch-and-cut algorithm for the capacitated vehicle routing problem. *Mathematical Programming*, 100(2):423–445, 2004.

[33] S. Martello, M. Monaci, and D. Vigo. An exact approach to the strip-packing problem. *INFORMS Journal on Computing*, 15(3):310–319, 2003.

[34] C. Prodhon and C. Prins. A survey of recent research on location-routing problems. *European Journal of Operational Research*, 238(1):1–17, 2014.

[35] S. Røpke and D. Pisinger. An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. *Transportation Science*, 40(4):455–472, 2006.

[36] S. Salhi and G.K. Rand. The effect of ignoring routes when locating depots. *European Journal of Operational Research*, 39(2):150–156, 1989.

[37] P. Toth and D. Vigo, editors. *Vehicle Routing: Problems, Methods, and Applications, 2nd Edition*. SIAM, Philadelphia, PA, 2014.

[38] F. Wang, Y. Tao, and N. Shi. A survey on vehicle routing problem with loading constraints. In *International Joint Conference on Computational Sciences and Optimization, CSO 2009*, pages 602–606, 2009.

[39] E.E. Zachariadis, C.D. Tarantilis, and C.T. Kiranoudis. A guided tabu search for the vehicle routing problem with two-dimensional loading constraints. *European Journal of Operational Research*, 195(3):729–743, 2009.