This is the peer reviewd version of the followng article:

Routing problems with loading constraints / Iori, Manuel; S., Martello. - In: TOP. - ISSN 1134-5764. - STAMPA. - 18:1(2010), pp. 4-27. [10.1007/s11750-010-0144-x]

Terms of use:

The terms and conditions for the reuse of this version of the manuscript are specified in the publishing policy. For all terms of use and more information see the publisher's website.

27/07/2024 16:34

(Article begins on next page)

Routing problems with loading constraints

Manuel Iori [†], Silvano Martello [‡]

[†] *Dipartimento di Scienze e Metodi dell'Ingegneria, Università di Modena e Reggio Emilia, Via Amendola 2, 42100 Reggio Emilia, Italy*
`manuel.iori@unimore.it`

[‡] *Dipartimento di Elettronica, Informatica e Sistemistica, Università degli Studi di Bologna, Viale Risorgimento 2, 40136 Bologna, Italy*
`silvano.martello@unibo.it`

Abstract

We consider difficult combinatorial optimization problems arising in transportation logistics when one is interested in optimizing both the routing of vehicles and the loading of goods into them. The separate problems (routing and loading) are already $\mathcal{NP}$-hard, and very difficult to solve in practice. A fortiori their combination is extremely challenging and stimulating. Although the specific literature is still quite limited, a first attempt to a systematic view of this field can be useful both to academic researchers and to practitioners. We review vehicle routing problems with two- and three-dimensional loading constraints. Other combinations of routing and special loading constraints arising from industrial applications are also considered.

Keywords: Vehicle routing, Loading, Two-dimensional packing, Three-dimensional packing, Traveling salesman, Pickup&delivery.

1 Introduction

Two central issues in transportation logistics concern the routing of vehicles and the loading of goods into them. Most optimization problems arising in these two areas are $\mathcal{NP}$-hard, and extremely difficult to solve in practice. For this reason, traditionally these two research areas have been handled separately, at the expense of the overall optimization. Only in recent years algorithms combining these two issues have been proposed in the literature. Combining two difficult problems leads to a considerable increase of difficulty, but on the other hand it allows a better solution of the corresponding logistic targets. Although the specific literature is still quite limited (a brief overview can be found in Wang, Tao and Shi [90]), we believe that a first attempt to a systematic view of this field can be useful both to academic researchers and to practitioners. In the next two sections we briefly review the basic issues in the areas of routing and loading, respectively. In Section 4 we consider the capacitated vehicle routing problem with two-dimensional loading constraints, and in Section 5 the capacitated vehicle routing problem with three-dimensional loading constraints. Another three-dimensional routing problem with special loading constraints is examined in Section 6. In Section 7 we discuss traveling salesman problems with pickup&delivery and loading constraints. A miscellaneous of other problems that combine routing and loading is finally provided in Section 8.

2 Routing problems

The prototype problem in the routing area is the famous *Traveling Salesman Problem* (TSP). Given a set of cities, along with the cost of traveling between each pair of cities, the TSP is to find a minimum cost tour of all the cities. The problem is $\mathcal{NP}$-hard in the strong sense, and its optimal solution is a classical challenge in combinatorial optimization. The *Symmetric Traveling Salesman Problem* (STSP) can be modeled by a *graph* $G = (V, E)$, where $V = \{0, 1, \ldots, n\}$ is the *vertex* set, $E = \{(i, j) : i, j \in V\}$ is the *edge* set, and c_{ij} $(i, j \in V)$ is the *cost* of traveling along edge (i, j) (in either direction). The *Asymmetric Traveling Salesman Problem* (ATSP) can be modeled by a *digraph* $G = (V, A)$, where $A = \{(i, j) : i, j \in V\}$ is the *arc* set, and c_{ij} $(i, j \in V)$ is the *cost* of traveling along arc (i, j) (from i to j). In most practical contexts it is assumed that the costs satisfy the *triangle inequality* $c_{ij} \leq c_{ik} + c_{kj}\ \forall\ i, j, k \in V$, which is easily imposed by defining each cost c_{ij} as the cost of the shortest path from i to j. The TSP is probably the most extensively studied problem in combinatorial optimization. We refer the reader to the most important books that treat this subject (Lawler, Lenstra, Rinnooy Kan and Shmoys [62], Reinelt [81], Gutin and Punnen [55], and Applegate, Bixby, Chvátal and Cook [1]), as well as to the recent articles by D'Ambrosio, Lodi and Martello [35] and Letchford and Lodi [63].

The natural extension of the TSP to real-world transportation issues is the *Capacitated Vehicle Routing Problem* (CVRP). We assume that a central depot is located at vertex 0, where a fleet of K identical vehicles is available, while n customers are located at vertices $1, 2, \ldots, n$. All vehicles have the same capacity D, and each customer has a demand d_i (with $0 \leq d_i \leq D$ for $i = 1, 2, \ldots, n$). The CVRP is to find a set of at most K circuits (called *routes*), each visiting the depot and a subset of customers, such that: (i) each customer is visited by exactly one vehicle; (ii) each vehicle performs at most one route; (iii) the sum of the demands on each route is not greater than D; (iv) the sum of the costs of the traveled edges/arcs is a minimum.

The CVRP has been deeply investigated since the seminal work by Dantzig and Ramser [36]. We will not review here the huge literature that exists on this problem. The reader is referred to the books by Toth and Vigo [87] and Golden, Raghavan and Wasil [54], as well as to the recent surveys by Cordeau, Laporte, Savelsbergh and Vigo [30] and Baldacci, Toth and Vigo [6, 7]. We just mention here that excellent results have been recently obtained by Fukasawa, Longo, Lysgaard, Poggi de Aragão, Resi, Uchoa and Werneck [49] (branch-and-cut-and-price), and by Baldacci, Christofides and Mingozzi [4] and Baldacci and Mingozzi [5] (set partitioning formulation).

Vehicle routing problems have great relevance in real world distribution systems, where the costs associated with operating vehicles and crews form an important component of the total costs: even small percentage savings can thus result in considerable total savings. The use of the CVRP for real-world problems can however be limited by the existence of many additional constraints that are not captured by the model. In the *Distance-Constrained CVRP* each edge or arc has an associated traveling time, and an upper bound is imposed on the total traveling time of each vehicle. In the *CVRP with time windows* each customer has an associated time interval in which the visit must start. In the *CVRP with backhauls*, in addition to the set of customers who have to receive a given quantity of product (*linehaul customers*), there is a set of customers where a given quantity of product has to be picked up (*backhaul customers*), and in each route all linehaul customers must be visited before all backhaul customers. The CVRP with backhauls is a member of a huge area of *CVRPs with pickup&delivery*, that includes many variants: customers may be associated to both product to deliver and product to be picked up, the product picked up at a customer may be delivered to another customer or has to go to the depot, and so on. All these problems may be further generalized to combined constraints and to variants such as, e.g., multiple depots, multiple periods, multiple vehicle types. The reader is referred again to the books and surveys [87, 54, 30, 6, 7] mentioned above, and, for the area of pickup&delivery routing problems, to the web page maintained at the University of La Laguna, see `http://webpages.ull.es/users/hhperez/PDsite/index.html`.

3 Loading constraints

In the CVRP the demand of a customer is expressed by a value that represents the total weight of the items to be delivered, while in real-world instances demands consist of sets of items which are characterized not only by a weight but also by a shape. A more detailed modeling of such characteristics can involve additional issues such as, e.g., handling loading and unloading operations, dealing with fragile items, operating with automatic forks, and so on. A first complication of the basic models arises when we want to ensure that the transported items can be feasibly allocated within the vehicle loading space. A second possible complication may come from the fact that the unloading operations should be performed without reshuffling the transported items, hence they are affected by the order in which the items are loaded and/or by the items position within the loading area. Problems related to the former complication are briefly reviewed below, while details on the latter are discussed in the specific sections.

In some transportation applications one has to handle rectangular-shaped items that cannot be stacked one on top of the other (because of their fragility or weight). This happens, for example, when the transported items are large kitchen appliances, such as refrigerators, or pieces of catering equipment, such as food trolleys. In such cases, the CVRP must include, besides the classical weight constraints, additional constraints to reflect two-dimensional loading aspects.

In other real-world contexts the customer demands consist of sets of three-dimensional rectangular boxes of given size and weight that can be (partially) superposed. For such cases, constraints imposing a feasible packing of the goods in the loading space have to be added to the weight constraints.

The loading issues above are closely related to multi-dimensional packing problems, which arise as extensions of the classical (one dimensional) bin packing (see, e.g., Martello and Toth [70], Coffman, Garey and Johnson [25], Coffman, Galambos, Martello and Vigo [24]). A considerable number of multi-dimensional packing problems has been studied in combinatorial optimization, and there is a huge literature on this subject. As will be seen in the next sections, the main problems that have been addressed in conjunction with routing issues are:

- *Two-Dimensional Bin Packing Problem* (2BPP): Pack a given set of rectangles into the minimum number of large identical rectangles (*bins*). Exact approaches for the 2BPP are generally based on branch-and-bound techniques and are able to solve instances with up to 100 items (although some instances with 20 items remain unsolved). Exact algorithms and lower bounds for the 2BPP have been proposed by Martello and Vigo [71], Fekete, Schepers and van der Veen [45], Boschetti and Mingozzi [12, 13], Pisinger and Sigurd [79], and Caprara and Monaci [17]. A closely related problem, known as the *orthogonal packing problem*, was effectively solved by Clautiaux, Jouglet, Carlier and Moukrim [23].

- *Two-Dimensional Strip Packing Problem* (2SPP): Pack a given set of rectangles into an open-ended rectangle of given width and infinite height (*strip*) so as to minimize the overall height at which the strip is used. A survey on this problem has been recently produced by Riff, Bonnaire and Neveu [82]. Recent effective algorithms not included in such survey are those by Caprara and Monaci [17] and Kenmochi, Imamichi, Nonobe, Yagiura and Nagamochi [60]. In addition, many heuristic and metaheuristic approaches have been tested on the 2BPP and the 2SPP.

- *Three-Dimensional Bin Packing Problem* (3BPP): Pack a given set of rectangular boxes into the minimum number of large identical three-dimensional boxes. Exact algorithms for the 3BPP have been given by Martello, Pisinger and Vigo [68] (see also [38]) and by Martello, Pisinger, Vigo, den Boef and Korst [69]. Lower bounds have been proposed, among others, by Boschetti [11], and effective heuristics by Faroe, Pisinger and Zachariasen [44] and by Crainic, Perboli and Tadei [34].

- *Three-Dimensional Strip Packing Problem* (3SPP): Pack a given set of rectangular boxes into an open-ended three-dimensional strip of given width and depth and infinite height so as to minimize the overall height at which the strip is used. A heuristic algorithm for the 3SPP was recently proposed by Bortfeldt and Mack [10].

The items may have a fixed orientation, parallel to the edges of the bins, or they can be rotated (usually by 90°). A typology of packing problems was proposed by Dyckhoff and Finke [41], and later updated by Wäscher, Haußner and Schumann [91]. The website of ESICUP, `http://www.fe.up.pt/~esicup` provides a repository of articles organized according to such typology.

In Sections 4 and 5 we review recent results obtained on combinations of the CVRP with two- and three-dimensional loading constraints. As the CVRP is a generalization of the TSP, it is $\mathcal{NP}$-hard in the strong sense, and very difficult to solve in practice. The same holds for all the packing problems above, which are generalizations of the (strongly $\mathcal{NP}$-hard) bin packing problem. Even deciding whether a given set of two-dimensional rectangles can be packed into a given two-dimensional bin is strongly $\mathcal{NP}$-complete. A fortiori the combinations of these two areas are extremely challenging. We will see that they can sometimes be solved exactly for small-size instances, and that powerful metaheuristic algorithms are able to produce solutions of good quality for instances of realistic size.

4 The capacitated vehicle routing problem with two-dimensional loading constraints

As in the classical CVRP, we are given a complete undirected graph $G = (V, E)$, where V is the set of the $n + 1$ vertices corresponding to the depot (vertex 0) and to the customers (vertices $1, 2, \ldots, n$), and E is the set of the edges (i, j), each having an associated cost c_{ij}. Each customer must be served by a single vehicle. There are K identical vehicles, each having a weight capacity D and a rectangular loading surface of width W and height H. The demand of customer i ($i = 1, 2, \ldots, n$) consists of m_i items of total weight d_i: item $I_{i\ell}$ ($\ell = 1, 2, \ldots, m_i$) has width $w_{i\ell}$ and height $h_{i\ell}$. In the version considered here, the items have fixed orientation, i.e., they must be packed with their w-edge (resp. h-edge) parallel to the W-edge (resp. H-edge) of the loading surface.

When a vehicle k is assigned a route that includes a set $S(k) \subseteq \{1, 2, \ldots, n\}$ of customers, the two following constraints must be satisfied:

- the total weight $\sum_{i \in S(k)} d_i$ must not exceed the vehicle capacity D;
- there must exist a feasible (non-overlapping) loading of all the items requested by the customers of $S(k)$ into the $W \times H$ loading area.

The *Capacitated Vehicle Routing Problem with Two-Dimensional Loading Constraints* (2L-CVRP) is to find a partition of the customers into no more than K subsets and, for each subset, a route starting and ending at the depot such that both conditions above hold, and the total cost of the edges in all the routes is a minimum. An example involving 8 customers and 3 vehicles is depicted in Figure 1 (taken from [53]).

The above problem is also known as the *Unrestricted* 2L-CVRP. Another version arises when the demanded items have great weight, size or fragility, so moving them inside the vehicle is unpractical. In the resulting problem, known as the *Sequential* 2L-CVRP, it is then additionally imposed that, for each vehicle k, the demanded items are allocated so that, when a customer is visited, his demanded items can be downloaded through a

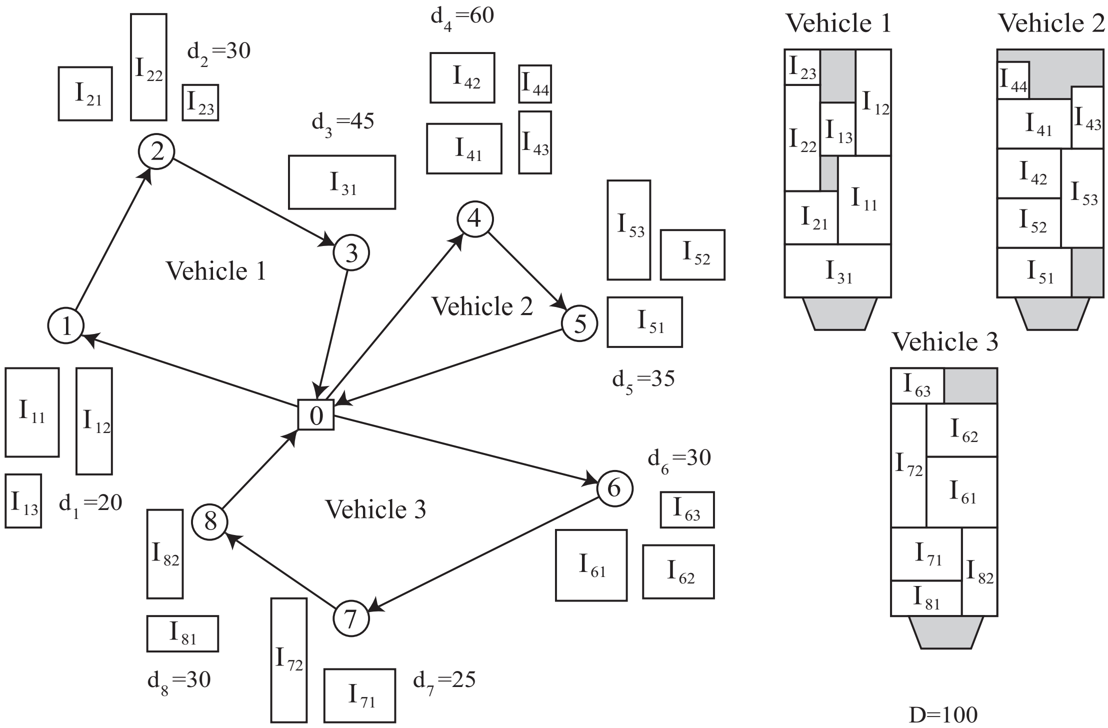

Figure 1: *Example of the 2L-CVRP.*

sequence of straight movements (one per item) parallel to the H-edge of the loading area. This implies that the strip going from the w-edge of any item demanded by the customer to the rear of the vehicle must be free of items demanded by customers visited later on in the route. In the transportation literature this constraint is denoted as "sequential loading", or "rear loading", or "LIFO (Last-In First-Out) policy".

The solution depicted in Figure 1 satisfies the sequential loading constraint. Consider, e.g., Vehicle 1: customer 1 is first visited, and items I_{12}, I_{13} and I_{11} can be consecutively unloaded, with no need of moving the other transported items.

With respect to the multi-dimensional packing problems introduced in Section 3, determining the minimum number of vehicles involves the solution to a 2BPP with additional constraints. A first constraint, common to both versions of the problem, imposes that all items demanded by a customer are packed in the same bin. For the sequential 2L-CVRP, a second constraint restricts the feasible patterns to those allowing the requested sequential operations.

Iori, Salazar González and Vigo [59] presented an exact algorithm for the solution of the sequential 2L-CVRP with integer edge costs, and two additional restrictions sometimes encountered in the literature: all K vehicles must be used, and no one-customer route is allowed. The algorithm is based on a branch-and-cut approach, making use of both classical valid inequalities from the CVRP literature and specific inequalities associated with infeasible loading sequences. The model handled by the algorithm initially disregards the loading constraints. Whenever it produces an integer solution satisfying the weight constraint for all routes, the feasibility of the corresponding loading patterns is evaluated through heuristics, possibly followed by a nested branch-and-bound procedure (following the approach introduced by Martello and Vigo [71]). If all loadings are feasible, the incumbent solution is updated. Otherwise new cuts are added to the current model in order to forbid infeasible routes. The algorithm was evaluated on benchmark instances from the CVRP literature (see Iori [57] for more details on the experiments), showing a satisfactory behavior for small-size instances.

In order to deal with more realistic, larger sized instances, heuristic algorithms have been proposed. Among the various metaheuristic approaches proposed for the CVRP (see Cordeau, Gendreau, Hertz, Laporte and Sormany [27] for a recent survey), very good results were obtained through Tabu search (see Cordeau and Laporte [29]). On the basis of this observation, Gendreau, Iori, Laporte and Martello [53] developed Tabu search algorithms for the 2L-CVRP, both for the sequential and the unrestricted version. The general approach can accept moves producing infeasible routes in the following sense. For the sequential 2L-CVRP, the generated routes must satisfy the constraint on sequential loading but can have a total weight exceeding D and/or can require a loading surface of height exceeding H. Moves leading to such infeasibilities are assigned a *penalty* proportional to the level of the constraint violation. Moves are evaluated by considering the edge costs, and possibly penalized by infeasible loadings. The routing aspect of the problem is handled through an adaptation of Taburoute, a Tabu search heuristic developed by Gendreau, Hertz and Laporte [50] for the CVRP. Note that weight infeasibilities are immediate to check, while the loading ones require the solution of an $\mathcal{NP}$-hard problem. The latter issue was thus handled through a heuristic algorithm which outputs, for each vehicle, a two-dimensional loading pattern having width W and a (possibly infeasible) height, to be tested against the available height H. This is obtained by heuristically solving a 2SPP problem for the unrestricted case, or a modified 2SPP problem for the sequential case, through iterated calls to modified versions of the greedy heuristic developed by Iori, Martello and Monaci [58].

Some improvements were proposed to the above Tabu search approach. An *Ant Colony Optimization* (ACO) algorithm was proposed by Fuellerer, Doerner, Hartl and Iori [47]. The algorithm is based on the ACO approach developed by Reimann, Doerner and Hartl [80] for the CVRP, modified and extended through the addition of loading techniques. The algorithm searches the space of routing solutions, and loading feasibility is checked through lower bounds, heuristics and a truncated branch-and-bound. The ant colony is initialized with a population of ants, each of which searches for a low-cost feasible solution through a generalization of the classical savings algorithm by Clarke and Wright [22]. The decision on combining customers into a unique route is based on a probabilistic rule that takes into account both savings and pheromone information.

Another improvement was proposed by Zachariadis, Tarantilis and Kiranoudis [93], who developed a guided Tabu search algorithm. The routing aspects of the problem are guided by a classical Tabu search strategy in which the objective function is conveniently altered to increase diversification. The feasibility of the produced loadings is tested through a number of heuristics. The information on the feasibility or infeasibility of the generated routes is stored in special data structures to avoid useless re-executions of the loading heuristics.

We present a computational analysis of the three metaheuristics discussed, performed on randomly generated instances. The graph and the weights demanded by the customers come from CVRP benchmarks (described in [87], and downloadable from `http://www.or.deis.unibo.it/research.html`). For each CVRP network, five 2L-CVRP instances were generated according to the following five classes:

Class 1: each customer was assigned a single item, with unit width and unit height (pure CVRP instances);

Classes 2 – 5: the number m_i of items demanded by each customer i was uniformly randomly generated in different intervals. Each item was then randomly assigned, with equal probability, one of three possible shapes (vertical, homogeneous or horizontal), and its width and height were generated accordingly.

Detailed information on the test bed generation is provided in Table 1. For each instance, the number of vehicles was determined by heuristically solving a 2BPP with additional constraints. For all classes, the loading area had sizes $H = 40$ and $W = 20$. All instances are available on line at `http://www.or.deis.unibo.it/research.html`.

The computational experiments on the resulting 180 instances were executed by the various authors on similar computers, although with different CPU speeds. Table 2 gives the average values obtained for each set of five instances. The first two columns give a CVRP instance identifier, and the number of customers. For each metaheuristic three information are provided: average solution value z, average elapsed CPU time when the final incumbent solution was found (sec_h), and average total CPU time spent (sec_{tot}). The last line of the table reports the average values over all instances. It turns out that the ACO

Table 1: 2L-CVRP test bed generation.

		Vertical		*Homogeneous*		*Horizontal*	
Class	m_i	$h_{i\ell}$	$w_{i\ell}$	$h_{i\ell}$	$w_{i\ell}$	$h_{i\ell}$	$w_{i\ell}$
1	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$	$[1,1]$
2	$[1,2]$	$\left[\frac{4H}{10},\frac{9H}{10}\right]$	$\left[\frac{W}{10},\frac{2W}{10}\right]$	$\left[\frac{2H}{10},\frac{5H}{10}\right]$	$\left[\frac{2W}{10},\frac{5W}{10}\right]$	$\left[\frac{H}{10},\frac{2H}{10}\right]$	$\left[\frac{4W}{10},\frac{9W}{10}\right]$
3	$[1,3]$	$\left[\frac{3H}{10},\frac{8H}{10}\right]$	$\left[\frac{W}{10},\frac{2W}{10}\right]$	$\left[\frac{2H}{10},\frac{4H}{10}\right]$	$\left[\frac{2W}{10},\frac{4W}{10}\right]$	$\left[\frac{H}{10},\frac{2H}{10}\right]$	$\left[\frac{3W}{10},\frac{8W}{10}\right]$
4	$[1,4]$	$\left[\frac{2H}{10},\frac{7H}{10}\right]$	$\left[\frac{W}{10},\frac{2W}{10}\right]$	$\left[\frac{H}{10},\frac{4H}{10}\right]$	$\left[\frac{W}{10},\frac{4W}{10}\right]$	$\left[\frac{H}{10},\frac{2H}{10}\right]$	$\left[\frac{2W}{10},\frac{7W}{10}\right]$
5	$[1,5]$	$\left[\frac{H}{10},\frac{6H}{10}\right]$	$\left[\frac{W}{10},\frac{2W}{10}\right]$	$\left[\frac{H}{10},\frac{3H}{10}\right]$	$\left[\frac{W}{10},\frac{3W}{10}\right]$	$\left[\frac{H}{10},\frac{2H}{10}\right]$	$\left[\frac{W}{10},\frac{6W}{10}\right]$

Table 2: Aggregate results on the 2L-CVRP with sequential loading (averages over five instances per line).

		Tabu search [53] (Pentium IV, 1.7 GHz)			ACO [47] (Pentium IV, 3.2 GHz)			Guided Tabu Search [93] (Pentium IV, 2.4 GHz)		
I	n	z	sec_h	sec_{tot}	z	sec_h	sec_{tot}	z	sec_h	sec_{tot}
1	15	295.01	2.6	9.2	291.83	5.5	7.2	299.12	2.9	4.6
2	15	343.18	0.4	3.5	343.22	0.3	0.6	344.36	1.6	2.6
3	20	380.19	3.8	18.9	378.17	2.3	3.1	386.36	2.6	4.6
4	20	440.91	1.4	17.0	439.07	2.3	3.0	441.87	2.4	3.9
5	21	382.30	4.1	27.6	381.63	9.7	12.8	392.14	5.6	9.5
6	21	501.40	5.1	19.5	499.77	3.3	4.3	503.20	2.9	4.7
7	22	691.23	15.5	53.0	678.22	9.2	11.7	692.78	14.6	25.8
8	22	691.89	32.8	83.7	684.37	9.6	16.1	686.29	17.4	26.5
9	25	620.77	10.6	40.0	614.88	4.0	4.9	619.29	8.6	14.2
10	29	679.68	43.5	179.6	668.48	47.8	50.1	689.45	23.5	41.3
11	29	719.76	99.0	199.4	690.22	55.1	57.8	715.52	44.0	77.1
12	30	627.59	58.8	99.5	615.90	6.5	7.4	624.90	66.4	106.8
13	32	2550.89	49.0	312.8	2480.04	58.9	61.8	2544.74	40.5	72.2
14	32	1048.72	146.0	439.5	1007.92	154.9	163.1	1041.76	86.7	143.8
15	32	1160.25	165.4	313.4	1145.96	133.9	138.8	1154.90	34.6	61.9
16	35	703.60	28.0	157.2	701.09	6.6	8.3	705.77	33.4	50.9
17	40	865.72	88.9	226.2	864.92	4.1	7.2	864.81	49.9	84.9
18	44	1037.65	566.5	1167.8	1003.84	285.2	292.6	1027.97	214.0	328.0
19	50	746.91	365.2	1521.5	728.89	118.6	122.1	745.67	292.8	466.8
20	71	513.84	808.9	3370.3	484.23	1057.4	1073.8	510.17	440.8	747.0
21	75	1025.79	1702.2	3561.2	987.54	1027.3	1037.3	1022.58	759.3	1303.7
22	75	1052.39	1573.8	3461.8	1018.76	726.8	738.9	1051.02	1184.3	1998.8
23	75	1121.18	675.8	3600.0	1051.16	1197.3	1206.5	1088.81	919.8	1461.3
24	75	1208.52	2642.5	3324.6	1134.90	312.6	323.4	1172.36	1303.7	2096.4
25	100	1350.56	2336.5	3600.1	1309.98	2424.5	2454.9	1349.11	1988.7	3389.2
26	100	1341.30	1554.6	3600.3	1306.24	2370.4	3558.1	1344.68	1115.5	1936.5
27	100	1439.37	1308.2	3600.0	1341.25	1536.8	1570.3	1390.20	818.6	1292.3
28	120	2502.48	2576.9	3600.1	2417.89	8349.4	8714.5	2476.66	1541.4	2622.8
29	134	2296.03	1162.5	3600.2	2131.54	8180.4	8837.5	2206.22	1257.7	2179.6
30	150	1873.27	2021.4	3600.2	1734.46	8267.0	8720.3	1832.96	1229.5	1980.6
31	199	2366.54	2102.2	3600.5	2219.34	8512.6	8747.4	2327.74	1681.6	2748.7
32	199	2354.60	2305.2	3600.6	2191.97	8687.9	8745.8	2235.70	2528.1	4313.5
33	199	2360.74	2221.2	3600.6	2245.46	8631.9	8742.7	2317.97	2367.0	4104.9
34	240	1408.64	2184.4	3601.0	1160.98	8645.4	8771.6	1186.77	3674.1	4596.2
35	252	1786.93	2223.1	3600.2	1465.85	8822.0	8942.2	1515.69	3291.8	4313.2
36	255	1693.10	2626.3	3600.9	1603.86	8978.6	9011.7	1610.60	2825.2	4732.2
AVG		1171.75	936.5	1817.0	1111.77	2462.4	2560.3	1142.22	829.8	1315.2

algorithm improves the solution quality, with respect to the original Tabu search, by 5% on average, while for the guided Tabu search the improvement is limited to 2.5%. The guided Tabu search improves the solutions for the larger instances, while for the smaller instances it is often outperformed by the original Tabu search. By considering the different CPU speeds, the ACO algorithm requires a considerable increase in the computational effort (around 2-3 times), while the guided Tabu search takes more or less the same time as the original Tabu search. Concerning the values of sec_h, it can be observed that the ACO algorithm finds the final incumbent solution in a CPU time very close to the total CPU time spent, and about 5 times larger than that of the Tabu search approaches.

On similar instances, the exact algorithm by Iori, Salazar González and Vigo [59] solved to optimality all instances with $n \leq 25$ (in less than one CPU hour per instance), and 37 % of those with $25 < n \leq 35$ (with time limit set to one CPU day per instance), failing for the larger instances. The largest instance solved to optimality had 35 customers and 114 items, while the smallest unsolved instance had 29 customers and 43 items. The instances of Classes 4 and 5 turned out to be the most challenging ones for the exact algorithm. This is probably explained by the fact that such instances include a large number of relatively small items per vehicle, hence the enumerative phase of the packing subproblems tend to degenerate to almost complete enumeration.

Computational experiments reported in [53] show that the inclusion of the unrestricted loading constraint considerably worsens the solution values of the CVRP, on average by about 50%. A relatively smaller further increase (on average by 4%) is produced by the constraint on sequential loading. Analogous results can be found in [47] and [93].

5 The capacitated vehicle routing problem with three-dimensional loading constraints

The 3L-CVRP is a natural extension to three dimensions of the two-dimensional case discussed in the previous section. To our knowledge, the first contribution in this direction is the metaheuristic algorithm proposed by Gendreau, Iori, Laporte and Martello [51], who extended their approach [53] to a three-dimensional case arising from real-world applications. In the 3L-CVRP, for each vehicle one has to solve a three-dimensional packing problem, which consists in finding a non-overlapping packing of a set of rectangular boxes into a rectangular container. Figure 2 depicts a solution to a simple problem with two vehicles and five customers. In addition to the standard weight and packing constraints (see Section 3), other operational constraints are frequently encountered in real-world applications. We assume that the boxes can be rotated by 90° on the horizontal plane, while upside-down rotations are not allowed. Some of the goods may be fragile, in which case it is requested that no non-fragile item be placed over a fragile one. In addition, when boxes are stacked, the supporting surface must be large enough to guarantee the stability of the load. It is also frequently requested that the loading of each vehicle allows sequential unloading (see Section 4). An example of three-dimensional loading for Vehicle 1 of Figure

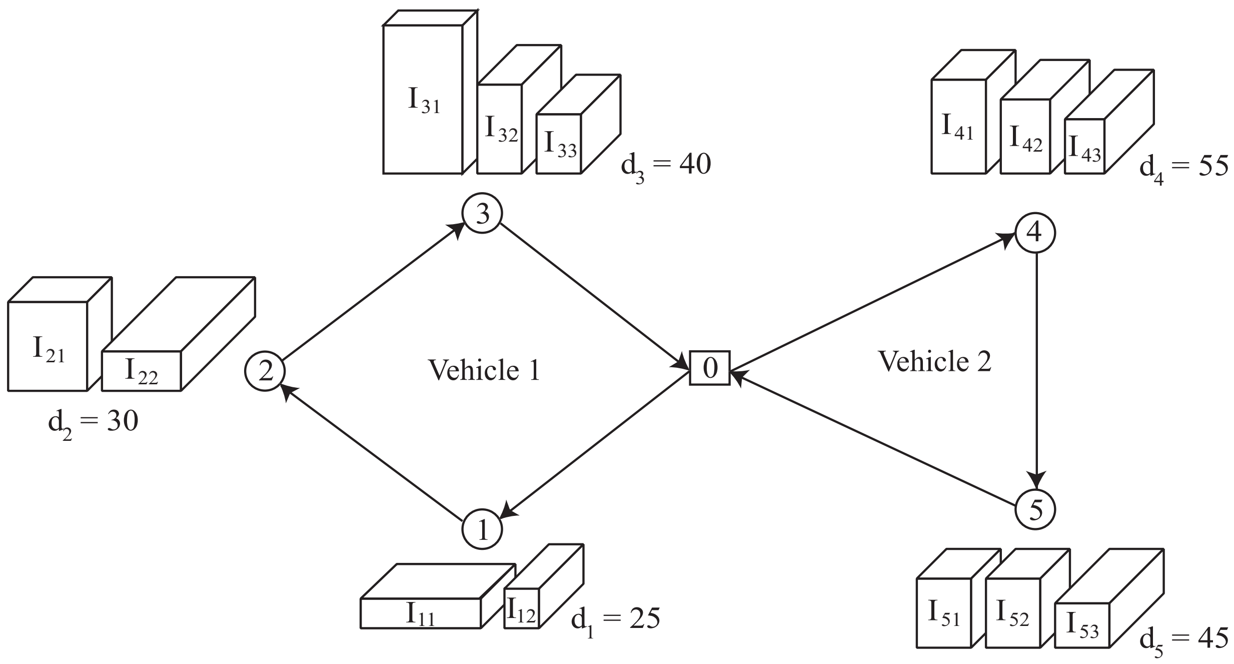

Figure 2: *Example of the 3L-CVRP.*

2 is shown in Figure 3 (taken from [51]), where it is assumed that the vehicle is unloaded in the direction of the z axis.

For the 3L-CVRP, each of the K vehicles has a weight capacity D and a three-

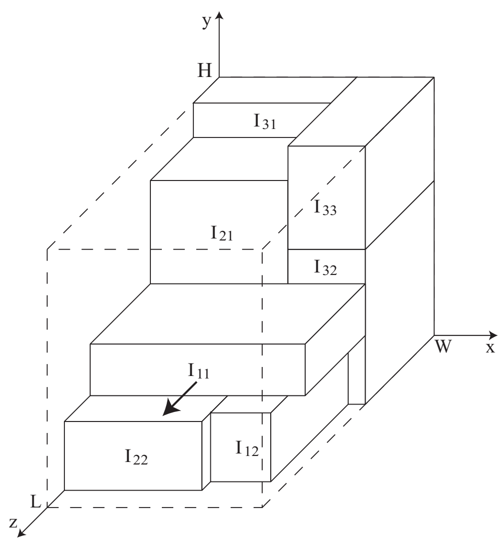

Figure 3: *A possible three-dimensional loading for Vehicle 1 of Figure 2.*

dimensional rectangular loading space defined by width W, height H and length L. Each customer i ($i = 1, 2, \ldots, n$) requires a set of m_i three-dimensional items $I_{i\ell}$ ($\ell = 1, 2, \ldots, m_i$) having width $w_{i\ell}$, height $h_{i\ell}$ and length $l_{i\ell}$, whose total weight is d_i. The items have a fixed orientation with respect to the height, but they can be rotated by 90° on the w-l plane. In addition, each item $I_{i\ell}$ has a *fragility* flag $f_{i\ell}$, equal to 1 if $I_{i\ell}$ is fragile, and to 0 otherwise: no non-fragile item can be placed over a fragile item, while fragile items can be stacked one over the other as well as over non-fragile items.

When one wants to place an item $I_{i\ell}$ over other items, the corresponding supporting surface has to be considered. Let $\overline{h}$ be the height at which the bottom of $I_{i\ell}$ is placed, and $\overline{A}$ the supporting area of the bottom of $I_{i\ell}$ that lays on items having their top at height $\overline{h}$: the packing is feasible only if such area is no less than a given threshold percentage a of the base of the item, i.e., if $\overline{A} \geq a\, w_{ik}\, l_{ik}$. We finally impose that the loading of each vehicle obeys the following sequential constraint. When customer i is visited, it must be possible to unload all items I_{ik} of his demand through a sequence of straight movements (one per item) parallel to the L-edge (see again Figure 3). In other words, no item demanded by a customer visited later may be placed over $I_{i\ell}$ or between $I_{i\ell}$ and the rear of the vehicle.

The 3L-CVRP consists in finding a set of at most K routes such that:

- each customer is served by exactly one vehicle;
- no vehicle carries a total weight exceeding D;
- for each vehicle there is an orthogonal three-dimensional loading of the transported items, which satisfies the constraints described above (fixed vertical orientation, fragility, supporting area and sequential loading);
- the solution cost is a minimum.

To our knowledge, no exact algorithm exists for this problem, and the only results are on extensions to this case of metaheuristic algorithms developed for the 2L-CVRP, and discussed in Section 4.

Gendreau, Iori, Laporte and Martello [51], used the general frame of their two-dimensional approach [53] (see Section 4), but introduced a novel way for determining the vehicle loads. Instead of solving a (possibly modified) 2SPP problem through iterated calls to a greedy heuristic, they developed an "inner" Tabu search algorithm (with its own Tabu list) to solve a modified 3SPP. Whenever a neighbor is explored by the "outer" (main) Tabu search, and a customer is moved from one route to another one, the loadings for the updated sets of transported items are computed by iteratively invoking the following inner algorithm:

(i) the inner neighborhood modifies the sequences adopted for the items loading;

(ii) two greedy heuristics are attempted in order to pack the items of the new sequence by minimizing the used length.

Once the inner algorithm has been executed a convenient number of times, if the smallest obtained length exceeds the vehicle length, the solution passed to the outer algorithm is penalized accordingly. The two greedy heuristics executed at Step (ii) were obtained by modifying two algorithms for two-dimensional packing problems: the classic *bottom-left* algorithm by Baker, Coffman and Rivest [3], and the effective *touching perimeter* algorithm by Lodi, Martello and Vigo [64].

Fuellerer, Doerner, Hartl and Iori [48] generalized their ACO approach [47] to the three-dimensional case by checking loading feasibility through the above mentioned modified heuristics from [3] and [64].

Tarantilis, Zachariadis and Kiranoudis [86] similarly adapted to the three-dimensional case their guided Tabu search [93]. They also considered a variant of the problem in which the size and weight of the transported items are not excessive, hence they can be manually unloaded, so some unloading operations become less restrictive.

In Table 3 we present a computational analysis of the above metaheuristics on 27 randomly generated instances, obtained from classic CVRP instances. The loading space was set to $W = 25$, $H = 30$ and $L = 60$. For each customer, the number of requested items was uniformly randomly generated between 1 and 3. For each item, width, height and length were uniformly randomly generated in the intervals $[0.2W, 0.6W]$, $[0.2H, 0.6H]$ and $[0.2L, 0.6L]$, respectively. For each instance, the number of vehicles was defined so as to ensure the existence of at least one feasible solution. This was obtained by heuristically solving, for each instance, a 3BPP, conveniently modified so as to take into account the additional loading constraints of the 3L-CVRP. The threshold percentage for the minimum supporting surface was set to 0.75. All instances are available on line at `http://www.or.deis.unibo.it/research.html`.

The first two columns of Table 3 give the instance identifier and the number of customers. For each instance and for each metaheuristic the entries give the solution value z, the elapsed CPU time when the final solution was found (sec_h), and the total CPU time (sec_{tot}). The last line of the table reports the average values over all instances. The ACO approach improved the solution quality by 8% with respect to the original Tabu search, the guided Tabu search by 4%. The total CPU times of the ACO and the guided Tabu search algorithms are roughly equivalent, while the CPU time spent to obtain the incumbent solution is larger for the former algorithm.

Additional computational experiments on the effect of the various loading constraints (reported in the original papers) show that the removal of one or more constraints produces an improvement of the solution value, as it can be expected, although relatively limited (roughly ranging between 3% and 15%). Gendreau, Iori, Laporte and Martello [51] also report computational results on challenging real-world 3L-CVRP instances provided by an Italian furniture company.

Table 3: Results on the 3L-CVRP.

		Tabu search [53] (Pentium IV, 3 GHz)			ACO [47] (Pentium IV, 3.2 GHz)			Guided Tabu Search [93] (Pentium IV, 2.8 GHz)		
I	n	z	sec_h	sec_{tot}	z	sec_h	sec_{tot}	z	sec_h	sec_{tot}
1	15	316.32	129.5	1800.0	305.35	11.2	12.0	321.47	7.8	13.2
2	15	350.58	5.3	1800.0	334.96	0.1	0.6	334.96	7.2	11.5
3	20	447.73	461.1	1800.0	409.79	88.5	121.8	430.95	352.6	540.6
4	20	448.48	181.1	1800.0	440.68	3.9	5.4	458.04	204.0	323.5
5	21	464.24	75.8	1800.0	453.19	22.7	30.9	465.79	61.3	99.6
6	21	504.46	1167.9	1800.0	501.47	17.5	18.4	507.96	768.8	1212.4
7	22	831.66	181.1	1800.0	797.47	51.4	67.4	796.61	241.5	364.8
8	22	871.77	156.1	1800.0	820.67	56.2	78.6	880.93	140.0	230.0
9	25	666.10	1468.5	1800.0	635.50	15.3	16.3	642.22	604.7	982.2
10	29	911.16	714.0	3600.0	841.12	241.2	246.7	884.74	803.1	1308.4
11	29	819.36	396.4	3600.0	821.04	172.4	199.8	873.43	308.5	522.5
12	30	651.58	268.1	3600.0	629.07	46.2	48.2	624.24	180.8	294.6
13	32	2928.34	1639.1	3600.0	2739.80	235.4	308.8	2799.74	1309.5	2193.1
14	32	1559.64	3451.6	3600.0	1472.26	623.8	642.8	1504.44	2678.1	4581.3
15	32	1452.34	2327.4	3600.0	1405.48	621.0	656.8	1415.42	1466.3	2528.3
16	35	707.85	2550.3	3600.0	698.92	12.8	14.8	698.61	2803.2	4256.5
17	40	920.87	2142.5	3600.0	870.33	11.8	14.9	872.79	1208.6	2096.0
18	44	1400.52	1452.9	3600.0	1261.07	2122.2	2209.8	1296.59	1300.9	2275.2
19	50	871.29	1822.3	7200.0	781.29	614.3	623.6	818.68	1438.4	2509.0
20	71	732.12	790.0	7200.0	611.26	3762.3	3901.0	641.57	1284.8	1940.9
21	75	1275.20	2370.3	7200.0	1124.55	5140.0	5180.6	1159.72	1704.8	2823.4
22	75	1277.94	1611.3	7200.0	1197.43	2233.6	2290.3	1245.35	1663.5	2685.6
23	75	1258.16	6725.6	7200.0	1171.77	3693.4	3727.6	1231.92	3048.2	4659.1
24	75	1307.09	6619.3	7200.0	1148.70	1762.8	1791.5	1201.96	2876.8	4854.1
25	100	1570.72	5630.9	7200.0	1436.32	8619.7	8817.1	1457.46	3432.0	5725.8
26	100	1847.95	4123.7	7200.0	1616.99	6651.2	6904.3	1711.93	3974.8	6283.1
27	100	1747.52	7127.2	7200.0	1573.50	10325.8	10483.9	1646.44	5864.2	9915.7
AVG		1042.26	2058.9	4266.7	966.66	1746.6	1793.1	997.18	1471.6	2415.9

6 The multi-pile vehicle routing problem

The multi-pile vehicle routing problem, introduced by Doerner, Fuellerer, Gronalt, Hartl and Iori [39], arose from a real-world transportation situation faced by a company delivering timber products. Each customer requires a mix of three-dimensional products that may belong to two categories:

- *short* chipboards of various types; and
- *long* chipboards.

All chipboards of the same category requested by a customer are preventively palletized, thus producing *short* and *long* pallets, called *items* in the following. All items have the same width W, equal to the vehicle width, and variable length and height. While the height can take any value between 1 and H (the vehicle height), the lengths can only take the value L (i.e., the vehicle length) for the long items, or the value $L/3$ for the short ones. The items can be stacked one on top of the other, producing *piles*. The length of the vehicle loading area is subdivided into three sectors of size $L/3$, and each item occupies a certain height of one or three sectors. The loading area is accessed through a rolling shutter placed on the side of the vehicle, so all piles can be accessed independently. The loading of each vehicle must obey a sequential constraint: when customer i is visited, it must be possible to unload all items of his demand without moving other items, i.e., such items must be on top of the piles. The weight capacity of the K vehicles is inessential in this case. Each customer i ($i = 1, 2, \ldots, n$) requires a set of m_i items $I_{i\ell}$ ($\ell = 1, 2, \ldots, m_i$) having fixed width W, height $h_{i\ell}$ and length $l_{i\ell} \in \{L/3, L\}$. The items can be placed one over the other, regardless of any supporting surface (see Section 5), as, in the loading phase, "holes" are filled with bulk material. The *Multi-Pile Vehicle Routing Problem* (MP-VRP) is to find a set of at most K routes such that:

- each customer is served by exactly one vehicle;
- for each vehicle there is a feasible sequential loading of the transported items;
- the solution cost is a minimum.

An example with 5 customers and 2 vehicles is depicted in Figure 4. A possible loading for Vehicle 1 is shown in Figure 5, where the dashed areas represent bulk material.

Doerner, Fuellerer, Gronalt, Hartl and Iori [39] proposed two metaheuristic approaches for the MP-VRP: a Tabu search algorithm and an ACO algorithm, conceptually analogous to those presented by Gendreau, Iori, Laporte and Martello [53, 52] and by Fuellerer, Doerner, Hartl and Iori [47], respectively.

Tricoire, Doerner, Hartl and Iori [88] presented a combination of *Variable Neighborhood Search* (VNS) and branch-and-cut for solving the problem, either exactly or heuristically. In the VNS phase the neighborhood is obtained through *cross-exchange* (see Taillard, Badeau, Gendreau, Guertin and Potvin [85]). The branch-and-cut algorithm is based on the classical two-index model of the CVRP, tightened by families of valid inequalities (see Lysgaard, Letchford and Eglese [66]). The overall resulting algorithm iteratively solves the subproblem of finding a feasible loading (if any) for a single vehicle and a given set of items. Lower bounds for such subproblem are obtained by relaxing it to the parallel processor scheduling problem known as $\mathrm{P}||\mathrm{C}_{\max}$, and computing the lower bounds proposed by Hochbaum and Shmoys [56] and by Dell'Amico and Martello [37], while the exact solution is obtained through dynamic programming.

Table 4 computationally compares the three metaheuristics above on 21 randomly generated instances obtained from the classic CVRP instances proposed by Christofides, Mingozzi and Toth [21], and available on line at `http://www.univie.ac.at/bwl/prod/`

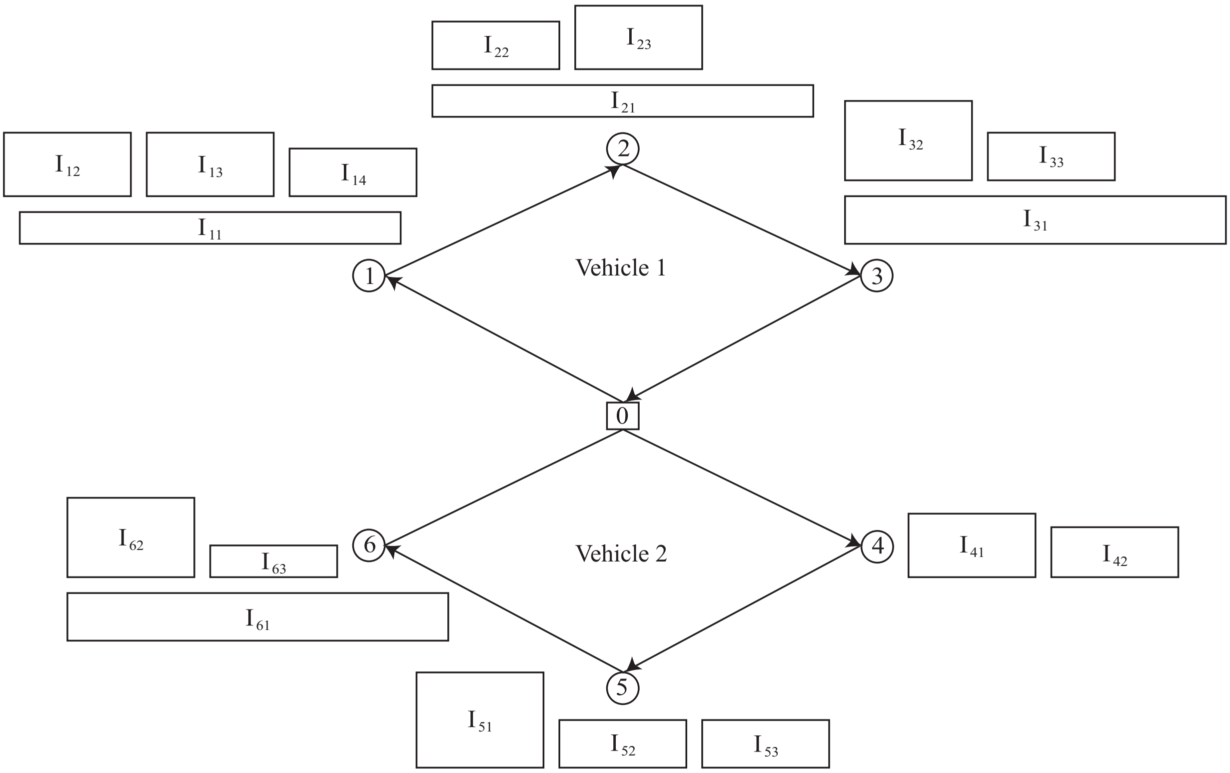

Figure 4: *Example of MP-VRP.*

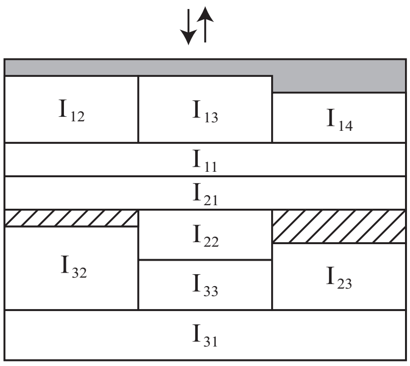

Figure 5: *A feasible loading for Vehicle 1 of Figure 4.*

`research/VRPandBPP/`. The entries give the same information as for Table 3. The results show a clear superiority of the VNS approach. The exact approach of [88] could solve to optimality instances with up to 38 customers. Results on the heuristic solution of real-world instances are also reported in [39] and in [88], showing, in this case too, a clear superiority of the VNS approach.

Table 4: Results on the MP-VRP.

		Tabu search [39] (Pentium IV, 2.6 GHz)			ACO [39] (Pentium IV, 2.6 GHz)			VNS [88] (Pentium IV, 3.2 GHz)		
I	n	z	sec_h	sec_{tot}	z	sec_h	sec_{tot}	z	sec_h	sec_{tot}
1	50	594.06	12.4	2966.9	594.56	4.9	12.3	591.80	90	< 1800
2	50	620.91	312.7	2229.2	622.82	5.9	11.5	640.07	76	< 1800
3	50	636.95	1261.8	1809.0	638.97	4.9	11.4	634.46	67	< 1800
4	75	990.51	119.1	2599.8	981.07	31.8	73.7	986.12	408	< 1800
5	75	912.62	3409.6	3594.9	915.34	30.6	72.7	901.01	283	< 1800
6	75	920.61	336.6	2694.8	917.84	29.6	72.4	892.11	409	< 1800
7	100	1209.46	3481.8	4885.6	1208.72	144.6	364.3	1197.44	402	< 1800
8	100	1247.54	2740.1	3585.3	1242.87	139.4	349.2	1222.37	468	< 1800
9	100	1196.15	3576.6	3994.0	1187.49	130.8	366.8	1162.49	604	< 1800
10	150	1672.70	2660.6	7200.0	1660.55	1599.5	3978.3	1635.78	470	< 1800
11	150	1603.09	4925.4	7200.0	1575.28	1513.0	3998.5	1562.83	470	< 1800
12	150	1592.68	4902.5	7092.1	1583.65	1269.7	3940.5	1547.90	563	< 1800
13	199	2107.49	1717.4	7200.1	2085.68	5501.5	7200.0	2049.25	658	< 1800
14	199	1879.00	6611.1	7200.0	1863.42	5416.6	7200.0	1839.54	925	< 1800
15	199	2042.28	4282.6	7200.0	1999.74	5090.5	7200.0	1965.85	506	< 1800
16	120	2292.03	522.9	6406.4	2269.56	837.0	1466.8	2254.28	739	< 1800
17	120	2122.34	1784.2	7200.0	2107.66	719.5	1327.1	2102.64	606	< 1800
18	120	2237.86	2855.0	4927.7	2195.66	666.5	1368.7	2183.12	794	< 1800
19	100	1154.31	1944.3	5918.0	1153.45	150.2	302.9	1136.61	345	< 1800
20	100	1237.43	148.6	4088.1	1248.83	157.5	303.5	1228.61	463	< 1800
21	100	1183.18	3859.6	4052.3	1182.92	109.8	274.8	1165.13	630	< 1800
AVG		1402.53	2450.7	4954.5	1392.19	1121.6	1899.8	1376.16	475	

7 Traveling salesman problems with pickup&delivery and loading constraints

In the *Traveling Salesman Problem with Pickup&Delivery* (TSPPD) a single vehicle must visit a set of customers, each associated with an origin location where some items must be picked up, and a destination location where such items must be delivered. The problem consists of determining a shortest Hamiltonian cycle through all locations while ensuring that the pickup of any given request is performed before the corresponding delivery.

Generalization of this problem in which loading aspects play a relevant role are reviewed in the next sections. The reader interested in general routing problems with pickup&delivery is referred to the surveys by Berbeglia, Cordeau, Gribkovskaia and Laporte [9] (who also propose a three-field classification) and Parragh, Doerner and Hartl [75, 76], as well as to the branch-and-cut algorithm for the TSPPD recently presented by Dumitrescu, Ropke, Cordeau and Laporte [40].

7.1 The traveling salesman problem with pickup&delivery and LIFO loading

The variant of the TSPPD in which both pickups and deliveries must be performed in Last-In First-Out (LIFO) order is known as the *TSPPD with LIFO Loading* (TSPPDL). The problem naturally arises in the routing of vehicles that have a single access point (usually the rear) for loading and unloading the transported items. Avoiding load rearrangements is particularly relevant when rear-loading vehicles transport large, heavy or fragile items, or hazardous materials. Another application arising in industrial contexts is the routing of automated guided vehicles (AGVs) which typically use a stack to move items between different workstations.

To the best of our knowledge, the first contribution on the TSPPDL is the one by Ladany and Mehrez [61], who studied a real-world delivery problem arising in the transportation of milk containers. A similar problem was later investigated by Pacheco [74], who proposed a heuristic algorithm based on classical TSP neighboring procedures. The TSPPDL was also recently studied by Carrabs, Cordeau and Laporte [19], who proposed new local search operators and a VNS heuristic, reporting results on instances with up to 375 customers.

Coming to exact approaches, a branch-and-bound algorithm was introduced by Carrabs, Cerulli and Cordeau [18]. The algorithm makes use of additive lower bounds produced by the classical TSP relaxations based on assignment problems and shortest spanning r-arborescences. The algorithm could solve to optimality many instances with 15 customers and some instances with 21 customers. Slightly better results were obtained by Cordeau, Iori, Laporte and Salazar González [28] through a branch-and-cut algorithm based on families of valid inequalities and on tailored branching strategies.

7.2 The traveling salesman problem with pickup&delivery and FIFO loading

The variant of the TSPPD in which both pickups and deliveries must be performed in First-In First-Out (FIFO) order is known as the *TSPPD with FIFO Loading* (TSPPDF). The TSPPDF arises, for example, in *fair* dial-a-ride systems, i.e., when the passengers resent another passenger being picked up after them but dropped off before them. Other potential industrial applications may arise in the management of automatic guided vehicles that load items on one end and unload them at the other end.

The TSPPDF was recently introduced by Erdogan, Cordeau and Laporte [42], who proposed an integer linear programming (ILP) formulation of the problem with a polynomial number of variables and constraints. Using CPLEX branch-and-bound, they were able to solve to optimality instances with 12 customers within 4 CPU hours. For other instances, involving 37 or more customers, even the linear programming relaxation of the problem could not be solved within the time limit, hence local search heuristics were adopted.

Carrabs, Cerulli and Cordeau [18] extended their additive branch-and-bound algorithm to the FIFO case, solving most instances with up to 13 customers and some instances with 15, 17 and 19 customers. Their algorithm strongly benefits from elimination rules that gradually remove from the graph arcs that are incompatible with previous branching decisions.

A branch-and-cut algorithm for the TSPPDF was recently proposed by Cordeau, Dell'Amico and Iori [26]. The algorithm, which makes use of valid inequalities, specially tailored branching strategies and fathoming criteria, could solve to optimality instances involving up to 25 customers.

7.3 The double traveling salesman problem with multiple stacks

The double traveling salesman problem with multiple stacks (DTSPMS) is a pickup&delivery problem in which all pickups must be completed before any delivery can be made. The problem originates from a real-life application where a rear-loaded container, structured in a number of stacks, is used to transport pallets from a set of pickup customers to a set of delivery customers. Pickups and deliveries are performed in two separate routes. The problem is to find the two routes and the stacking plan that minimize the total transportation costs.

The problem was introduced by Petersen and Madsen [78], who presented an ILP model and simple metaheuristic algorithms. A VNS algorithm for the DTSPMS was proposed by Felipe, Ortuño and Tirado [46]. Lusby, Larsen, Ehrgott and Ryan [65] presented an exact algorithm, which generates the k best solutions for each of the two separate routes, and looks for the lowest cost pair (if any) that allows a feasible stacking plan. Petersen, Archetti and Speranza [77] proposed several branch-and-cut algorithms, one of which, based on the separation of infeasible path constraints, clearly outperforms the others.

8 Miscellaneous

The problems we have considered so far consist of a CVRP with the addition of constraints on the loading. Other variants of routing problems, especially arising from industrial contexts, include a number of additional real-world constraints, among which loading is present at a certain extent. In the next sections we briefly comment on recent results in such area, restricting ourselves to the cases in which the loading aspect is particularly relevant.

8.1 Vehicle routing problems with multi-compartment loading

Routing-loading problems arising in the shipping industry and in the land delivery of petroleum products frequently impose the transportation of various products in separate compartments or tanks. It is usually imposed, especially in the case of land delivery, that, whenever a client is visited, the entire contents of the requested tanks are emptied.

In its basic version, the land delivery problem, known as *Petrol Station Replenishment Problem* (PSRP) (see Cornillier, Boctor, Laporte and Renaud [31]) calls for the delivery of petroleum products to petrol stations by means of an heterogeneous fleet of compartmented tank vehicles. The objective is to maximize the total profit, which is given by the sales revenue minus the routing cost.

Already in the early Eighties, Brown and Graves [16] developed an automated, real-time dispatch system for the daily operations at Chevron USA, involving dispatching from more than 80 bulk terminals on a fleet of more than 300 vehicles in approximately 2600 loads per day. They introduced an integer programming model and a heuristic algorithm based on a network structure representing the matching of orders and vehicles. Brown, Ellis, Graves and Ronen [14] later derived another real-time dispatch system for Mobil Oil Corporation. A study to redesign the distribution network in the Netherlands of a large oil company was presented by van der Bruggen and R. Gruson and M. Salomon" [89]. The operational environments existing in dispatching petroleum products, and the operations research tools used by oil companies up to the mid Nineties were examined by Rosen [83].

Avella, Boccia and Sforza [2] considered the case of a company that delivers different types of fuel from a single depot to a set of fuel pumps. They formulated the problem through set partitioning, and proposed an exact branch-and-price algorithm combined with a fast heuristic used to produce an initial feasible solution and a starting set of columns.

Cornillier, Boctor, Laporte and Renaud [32] address the basic version of the PSRP by means of an exact algorithm, tested on randomly generated instances and on a real-world case arising in Eastern Quebec. The algorithm is based on the decomposition of the original problem into a routing component and a loading component. The routing component is modeled through a classical set partitioning formulation, solved through column generation, while the feasibility of each route is tested through the solution of an associated loading subproblem. The computational results indicate the effectiveness of the algorithm, which the authors report to be regularly used by the transportation company that provided the test case. The same authors addressed two generalizations of the PSRP:

- for the case where the deliveries of petrol products to the stations can be postponed or anticipated, [31] gives a multi-phase heuristic algorithm, followed by two local search procedures. For each potential route, the loading problem is solved by a greedy heuristic followed by a simple improvement phase;

- for the case where the deliveries to the stations have to obey given time windows, [33] proposes two constructive heuristics based on a pre-selection of given subsets of arcs and/or routes and on the decomposition of the geographical space into sectors.

Fagerholt and Christiansen [43] studied a multi-ship pickup&delivery problem faced by a producer of mineral fertilizers that adopts ships characterized by a flexible way of partitioning the loading space in compartments. They developed a set partitioning approach consisting of two phases. In the first phase, a restricted number of feasible candidate routes for each ship is heuristically generated, while the second phase produces an overall feasible solution through set partitioning. A similar set partitioning approach had been used by Brown, Goodman and Wood [15] to determine the annual planning schedule for naval combatants of the Atlantic fleet so as to optimize the fleet combat readiness during peacetime.

8.2 Other routing and loading problems

Malapert, Guerét, Jussien, Langevin and L.-M. Rousseau [67] considered a variant of the 2L-CVRP in which additional pickup&delivery constraints (see Section 2) are present, and proposed a Constraint Programming model, based on a scheduling approach to handle the loading aspects.

Three-dimensional container loading problems with multi-drop constraints were investigated by Christensen and Rousøe [20], who proposed a heuristic algorithm based on a tree search framework.

Moura and Oliveira [73] addressed the issue of integrating the CVRP with time windows and three-dimensional container loading, proposing a number of constructive heuristics. Moura [72] modeled the same problem as a multi-objective problem in which the multi-objective function space has three dimensions (number of vehicles, total traveled distance and volume utilization), and presented a genetic algorithm for its solution.

A special class of CVRP with pickup&delivery, delivery due dates, three-dimensional loading and other constraints, arising in an industrial context, was treated by Tadei, Perboli and Della Croce [84]. The considered problem is that of delivering cars and commercial vehicles using ad hoc trucks (the so-called *auto-carrier transportation problem*). The problem was solved through a three-step heuristic procedure strongly based on an integer programming formulation.

Xu, Chen, Rajagopal and Arunapuram [92] considered a real-world logistics problem involving multiple vehicle types, pickup&delivery, time windows, loading and unloading operations and drivers work rules. They proposed a column generation approach based on a set partitioning formulation.

Battarra, Erdogan, Laporte and Vigo [8] recently considered a problem in which a single, rear loaded vehicle has to visit a set of clients, each of which may require a certain quantity of commodity from the depot (*delivery commodity)* and supply a certain quantity of commodity to the depot (*pick-up commodity*). Differently from the CVRP with backhauls (see Section 2), each client must be visited only once, implying that the pickup commodities may obstruct the unloading of the delivery commodities. In such a case, the currently transported goods may have to be rearranged, thus increasing the overall transportation cost. The authors called this problem the *Traveling Salesman Problem with Pickup&Delivery and Handling Costs*, and proposed heuristics and branch-and-cut algorithms in which the handling decisions are restricted to three simplified policies.

9 Conclusions and open perspectives

We have presented a review on the integrated (exact or heuristic) solution of vehicle routing and vehicle loading. This recent and promising research areas opens new possibilities of solving real world problems in transportation. As it also arose from the comments from the discussants, further developments concern

- evaluation of classical vehicle routing situations (e.g., allowing split deliveries);
- heterogeneous vehicle fleets;
- addition of special loading requirements (e.g., issues related to the center of gravity of the load);
- use of column generation techniques for effectively determining exact solutions;
- study of different objective functions reflecting special practical needs, such as routes with similar lengths or loads;
- integration of these models and algorithms with location issues.

Acknowledgements

This work was supported by Ministero dell'Istruzione, dell'Università e della Ricerca (MIUR), Italy.

References

[1] D. Applegate, R.E. Bixby, V. Chvátal, and W. Cook. *The Traveling Salesman: a Computational Study*. Princeton University Press, Princeton, 2007.

[2] P. Avella, M. Boccia, and A. Sforza. Solving a fuel delivery problem by heuristic and exact approaches. *European Journal of Operational Research*, 151:170–179, 2004.

[3] B. S. Baker, E. G. Coffman, Jr., and R. L. Rivest. Orthogonal packing in two dimensions. *SIAM Journal on Computing*, 9:846–855, 1980.

[4] R. Baldacci, N. Christofides, and A. Mingozzi. An exact algorithm for the vehicle routing problem based on the set partitioning formulation with additional cuts. *Mathematical Programming*, 115:351–385, 2008.

[5] R. Baldacci and A. Mingozzi. A unified exact method for solving different classes of vehicle routing problems. *Mathematical Programming*, 120:347–380, 2009.

[6] R. Baldacci, P. Toth, and D. Vigo. Recent advances in vehicle routing exact algorithms. *4OR: A Quarterly Journal of Operations Research*, 5:269–298, 2007.

[7] R. Baldacci, P. Toth, and D. Vigo. Exact algorithms for routing problems under vehicle capacity constraints. In D. Bouyssou, S. Martello, and F. Plastria, editors, *Operations Research II, Invited Surveys from 4OR, 2006-2008*, volume 175 of *Annals of Operations Research*, pages 213–245. Springer, Berlin, 2010.

[8] M. Battarra, G. Erdogan, G. Laporte, and D. Vigo. The travelling salesman problem with pickups, deliveries and handling costs. *Transportation Science*, 2010. To appear.

[9] G. Berbeglia, J.-F. Cordeau, I. Gribkovskaia, and G. Laporte. Static pickup and delivery problems: A classification scheme and survey. *TOP*, 15:1–31, 2007.

[10] A. Bortfeldt and D. Mack. A heuristic for the three-dimensional strip packing problem. *European Journal of Operational Research*, 183:1267–1279, 2007.

[11] M.A. Boschetti. New lower bounds for the three-dimensional finite bin packing problem. *Discrete Applied Mathematics*, 140(1-3):241–258, 2004.

[12] M.A. Boschetti and A. Mingozzi. The two-dimensional finite bin packing problem. Part I: New lower bounds for the oriented case. *4OR*, 1:27–42, 2003.

[13] M.A. Boschetti and A. Mingozzi. The two-dimensional finite bin packing problem. Part II: New lower and upper bounds. *4OR*, 1:135–147, 2003.

[14] G.G. Brown, C.J. Ellis, G.W. Graves, and D. Ronen. Realtime, wide area dispatch of Mobil tank trucks. *Interfaces*, 17:107–120, 1987.

[15] G.G. Brown, C.E. Goodman, and R.K. Wood. Annual scheduling of atlantic fleet naval combatants. *Operations Research*, 38:249–259, 1990.

[16] G.G. Brown and G.W. Graves. Real-time dispatching of petroleum tank trucks. *Management Science*, 27:19–32, 1981.

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

[18] F. Carrabs, R. Cerulli, and J.-F. Cordeau. An additive branch-and-bound algorithm for the pickup and delivery traveling salesman problem with LIFO or FIFO loading. *INFOR*, 45:223–238, 2007.

[19] F. Carrabs, J.-F. Cordeau, and G. Laporte. Variable neighbourhood search for the pickup and delivery traveling salesman problem with LIFO loading. *INFORMS Journal on Computing*, 19:618–623, 2007.

[20] S.G. Christensen and D.M. Rousøe. Container loading with multi-drop constraints. *International Transactions in Operations Research*, 16:727–743, 2010.

[21] N. Christofides, A. Mingozzi, and P. Toth. The vehicle routing problem. In N. Christofides, A. Mingozzi, P. Toth, and C. Sandi, editors, *Combinatorial optimization*, pages 315–338. Wiley, Chichester, 1979.

[22] G. Clarke and J.V. Wright. Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research*, 12:568–581, 1964.

[23] F. Clautiaux, A. Jouglet, J. Carlier, and A. Moukrim. A new constraint programming approach for the orthogonal packing problem. *Computers & Operations Research*, 35:944–959, 2008.

[24] E.G.Jr. Coffman, G. Galambos, S. Martello, and D. Vigo. Bin packing approximation algorithms: Combinatorial analysis. In D.-Z. Du and P.M. Pardalos, editors, *Handbook of Combinatorial Optimization*, pages 151–208. Kluwer Academic Publishers, 1999.

[25] E.G.Jr. Coffman, M.R. Garey, and D.S. Johnson. Approximation algorithms for bin packing: a survey. pages 46–93. PWS Publishing Co., Boston, MA, USA, 1997.

[26] J.-F. Cordeau, D. Dell'Amico, and M. Iori. Branch-and-cut for the pickup and delivery traveling salesman problem with FIFO loading. *Computers & Operations Research*, 37:970–980, 2010.

[27] J.-F. Cordeau, M. Gendreau, A. Hertz, G. Laporte, and J.-S. Sormany. New heuristics for the vehicle routing problem. In A. Langevin and D. Riopel, editors, *Logistics Systems: Design and Optimization*, pages 279–297. Springer, New York, 2005.

[28] J.-F. Cordeau, M. Iori, G. Laporte, and J.J. Salazar González. Branch-and-cut for the pickup and delivery traveling salesman problem with lifo loading. *Networks*, 55:46–59, 2010.

[29] J.-F. Cordeau and G. Laporte. Tabu search heuristics for the vehicle routing problem. In C. Rego and B. Alidaee, editors, *Metaheuristic Optimization via Memory and Evolution: Tabu Search and Scatter Search*, pages 145–163. Kluwer, Boston, 2004.

[30] J.-F. Cordeau, G. Laporte, M.W.P. Savelsbergh, and D. Vigo. Vehicle routing. In C. Barnhart and G. Laporte, editors, *Transportation*, volume 14 of *Handbooks in Operations Research and Management Science*, pages 367–428. Elsevier, Amsterdam.

[31] F. Cornillier, F.F. Boctor, G. Laporte, and J. Renaud. An exact algorithm for the petrol station replenishment problem. *Journal of the Operational Research Society*, 59:607–615, 2008.

[32] F. Cornillier, F.F. Boctor, G. Laporte, and J. Renaud. A heuristic for the multi-period petrol station replenishment problem. *European Journal of Operational Research*, 191:295–305, 2008.

[33] F. Cornillier, F.F. Boctor, G. Laporte, and J. Renaud. The petrol station replenishment problem with time windows. *Computers & Operations Research*, 36:919–935, 2009.

[34] T.G. Crainic, G. Perboli, and R. Tadei. Extreme point-based heuristics for three-dimensional bin packing. *INFORMS Journal on Computing*, 20:368–384, 2008.

[35] C. D'Ambrosio, A. Lodi, and S. Martello. Combinatorial traveling salesman problem algorithms. In J.J. Cochran, editor, *Wiley Encyclopedia of Operations Research and Management Science*. Wiley, Chichester, 2010. To appear.

[36] G.B. Dantzig and J.H. Ramser. The truck dispatching problem. *Management Science*, 6:80, 1959.

[37] M. Dell'Amico and S. Martello. Optimal scheduling of tasks on identical parallel processors. *ORSA Journal on Computing*, 7:191–200, 1995.

[38] E. den Boef, J. Korst, S. Martello, D. Pisinger, and D. Vigo. Erratum to "the three-dimensional bin packing problem": Robot-packable and orthogonal variants of packing problems. *Operations Research*, 53:735–736, 2005.

[39] K. Doerner, G. Fuellerer, M. Gronalt, R. Hartl, and M. Iori. Metaheuristics for vehicle routing problems with loading constraints. *Networks*, 49:294–307, 2007.

[40] I. Dumitrescu, S. Ropke, J.-F. Cordeau, and G. Laporte. The traveling salesman problem with pickup and delivery: polyhedral results and a branch-and-cut algorithm. *Mathematical Programming, Series A*, 121:269–305, 2010.

[41] H. Dyckhoff and U. Finke. *Cutting and Packing in Production and Distribution*. Physica-Verlag, Heidelberg, 1992.

[42] G. Erdogan, J.-F. Cordeau, and G. Laporte. The pickup and delivery traveling salesman problem with first-in-first-out loading. *Computers & Operations Research*, 36:1800–1808, 2009.

[43] K. Fagerholt and M. Christiansen. A combined ship scheduling and allocation problem. *Journal of the Operational Research Society*, 51:834–842, 2000.

[44] O. Faroe, D. Pisinger, and M. Zachariasen. Guided local search for the three-dimensional bin packing problem. *INFORMS Journal on Computing*, 15:267–283, 2003.

[45] S.P. Fekete, J. Schepers, and J.C. van der Veen. An exact algorithm for higher-dimensional orthogonal packing. *Operations Research*, 55:569–587, 2007.

[46] A. Felipe, M.T. Ortuño, and G. Tirado. New neighborhood structures for the double traveling salesman problem with multiple stacks. *TOP*, 17:190–213, 2009.

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

[48] G. Fuellerer, K. Doerner, R. Hartl, and M. Iori. Metaheuristics for vehicle routing problems with three-dimensional loading constraints. *European Journal of Operational Research*, 201:751, 2010.

[49] R. Fukasawa, H. Longo, J. Lysgaard, M. Poggi de Aragão, M. Reis, Uchoa. E., and R.F. Werneck. Robust branch-and-cut-and-price for the capacitated vehicle routing problem. *Mathematical Programming*, 106:491–511, 2006.

[50] M. Gendreau, A. Hertz, and Laporte G. A tabu search heuristic for the vehicle routing problem. *Management Science*, 40:1276–1290, 1994.

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

[52] M. Gendreau, M. Iori, G. Laporte, and S. Martello. Erratum: A tabu search heuristic for the vehicle routing problem with two-dimensional loading constraints. *Networks*, 51:153–153, 2007.

[53] M. Gendreau, M. Iori, G. Laporte, and S. Martello. A tabu search heuristic for the vehicle routing problem with two-dimensional loading constraints. *Networks*, 51:4–18, 2007.

[54] B. Golden, S. Raghavan, and E. Wasil (eds.). *The Vehicle Routing Problem: Latest Advances And New Challenges*, volume 43 of *Operations Research/computer Science Interfaces Series*. Springer, Berlin, 2008.

[55] G. Gutin and A.P. Punnen (eds.). *The Traveling Salesman and its Variations*. Kluwer, Dordrecht, 2002.

[56] D. S. Hochbaum and D. B. Shmoys. Using dual approximation algorithms for scheduling problems: practical and theoretical results. *Journal of ACM*, 34:144–162, 1987.

[57] M. Iori. Metaheuristic algorithms for combinatorial optimization problems. *4OR*, 3:163–166, 2005.

[58] M. Iori, S. Martello, and M. Monaci. Metaheuristic algorithms for the strip packing problem. In P. Pardalos and V. Korotkich, editors, *Optimization and Industry: New Frontiers*, pages 159–179. Kluwer, Boston, 2003.

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

[60] M. Kenmochi, T. Imamichi, K. Nonobe, M. Yagiura, and H. Nagamochi. Exact algorithms for the two-dimensional strip packing problem with and without rotations. *European Journal of Operational Research*, 198:73–83, 2009.

[61] S. P. Ladany and A. Mehrez. Optimal routing of a single vehicle with loading constraints. *Transportation Planning and Technology*, 8:301–306, 1984.

[62] E.L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan, and D. Shmoys (eds.). *The Traveling Salesman Problem*. Wiley, Chichester, 1985.

[63] A.N. Letchford and A. Lodi. Mathematical programming approaches to the traveling salesman problem. In J.J. Cochran, editor, *Wiley Encyclopedia of Operations Research and Management Science*. Wiley, Chichester, 2010. To appear.

[64] A. Lodi, S. Martello, and D. Vigo. Heuristic and metaheuristic approaches for a class of two-dimensional bin packing problems. *INFORMS Journal on Computing*, 11:345–357, 1999.

[65] R. Lusby, J. Larsen, M. Ehrgott, and D. Ryan. An exact method for the double TSP with multiple stacks. *International Transactions in Operations Research*, 2010. To appear.

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

[67] A. Malapert, C. Guerét, N. Jussien, A. Langevin, and L.-M. Rousseau. Two-dimensional pickup and delivery routing problem with loading constraints. In *Proceedings of the First CPAIOR Workshop on Bin Packing and Placement Constraints (BPPC'08)*, Paris, France, 2008.

[68] S. Martello, D. Pisinger, and D. Vigo. The three-dimensional bin packing problem. *Operations Research*, 48:256–267, 2000.

[69] S. Martello, D. Pisinger, D. Vigo, E. Den Boef, and J. Korst. Algorithm 864: General and robot-packable variants of the three-dimensional bin packing problem. *ACM Transactions on Mathematical Software*, 33:7, 2007.

[70] S. Martello and P. Toth. *Knapsack Problems: Algorithms and Computer Implementations*. John Wiley & Sons, Chichester, 1990, (available on line at `http://www.or.deis.unibo.it/knapsack.html`).

[71] S. Martello and D. Vigo. Exact solution of the two-dimensional finite bin packing problem. *Management Science*, 44:388–399, 1998.

[72] A. Moura. A multi-objective genetic algorithm for the vehicle routing with time windows and loading. In A. Bortfeldt, J. Homberger, H. Kopfer, G. Pankratz, and R. Strangmeier, editors, *Intelligent Decision Support*, pages 187–201. Gabler, germany, 2008.

[73] A. Moura and J.F. Oliveira. An integrated approach to vehicle routing and container loading problems. *OR Spectrum*, 31:775–800, 2009.

[74] J. Pacheco. Heuristico para los problemas de ruta con carga y descarga en sistemas LIFO. *SORT, Statistics and Operations Research Transactions*, 21:153–175, 1997.

[75] S.N. Parragh, K.F. Doerner, and R.F. Hartl. A survey on pickup and delivery models. Part I: transportation between customers and depot. *Journal für Betriebswirtschaft*, 58:21–51, 2008.

[76] S.N. Parragh, K.F. Doerner, and R.F. Hartl. A survey on pickup and delivery models. Part II: transportation between pickup and delivery locations. *Journal für Betriebswirtschaft*, 58:81–117, 2008.

[77] H. Petersen, C. Archetti, and M.G. Speranza. Exact solution approaches to the double travelling salesman problem with multiple stacks. *Networks*, 2010. To appear.

[78] H.L. Petersen and O.B.G. Madsen. The double travelling salesman problem with multiple stacks formulation and heuristic solution approaches. *European Journal of Operational Research*, 198:139–147, 2009.

[79] D. Pisinger and M. Sigurd. Using decomposition techniques and constraint programming for solving the two-dimensional bin-packing problem. *INFORMS Journal on Computing*, 19:36–51, 2007.

[80] M. Reimann, K. Doerner, and R. F. Hartl. D-ants: Savings based ants divide and conquer the vehicle routing problem. *Computers & Operations Research*, 31:563–591, 2004.

[81] G. Reinelt. *The Traveling Salesman: Computational Solutions for TSP Applications*, volume 840 of *Lecture Notes in Computer Science*. Springer, Berlin, 1994.

[82] M.C. Riff, X. Bonnaire, and B. Neveu. A revision of recent approaches for two-dimensional strip-packing problems. *Engineering Applications of Artificial Intelligence*, 198:823–827, 2009.

[83] D. Ronen. Dispatching petroleum products. *Operations Research*, 43:379–387, 1995.

[84] R. Tadei, G. Perboli, and F. Della Croce. A heuristic algorithm for the auto-carrier transportation problem. *Transportation Science*, 36:55–62, 2002.

[85] E. Taillard, P. Badeau, M. Gendreau, F. Guertin, and J.-Y. Potvin. A tabu search heuristic for the vehicle routing problem with soft time windows. *Transportation Science*, 31:170–186, 1997.

[86] C.D. Tarantilis, E.E. Zachariadis, and C.T. Kiranoudis. A hybrid metaheuristic algorithm for the integrated vehicle routing and three-dimensional container-loading problem. *IEEE Transactions on Intelligent Transportation Systems*, 10:255–271, 2009.

[87] P. Toth and D. Vigo. *The Vehicle Routing Problem*. SIAM Monographs on Discrete Mathematics and Applications, Philadelphia, 2002.

[88] F. Tricoire, K. Doerner, R. Hartl, and M. Iori. Heuristic and exact algorithms for the multi-pile vehicle routing problem. *OR Spectrum*, 2010. To appear.

[89] L. van der Bruggen, R. Gruson, and M. Salomon. Reconsidering the distribution structure of gasoline products for a large oil company. *European Journal of Operational Research*, 81:460–473, 1995.

[90] F. Wang, Y. Tao, and N. Shi. A survey on vehicle routing problem with loading constraints. *International Joint Conference on Computational Sciences and Optimization*, 2:602–606, 2009.

[91] G. Wäscher, H. Haußner, and H. Schumann. An improved typology of cutting and packing problems. *European Journal of Operational Research*, 183:1109–1130, 2007.

[92] H. Xu, Z.-L. Chen, S. Rajagopal, and S. Arunapuram. Solving a practical pickup and delivery problem. *Transportation Science*, 37:347–364, 2003.

[93] 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*, 195:729–743, 2009.