Computers & Operations Research 40 (2013) 873–885

Contents lists available at SciVerse ScienceDirect

Computers & Operations Research

journal homepage: www.elsevier.com/locate/caor

Packing first, routing second—a heuristic for the vehicle routing and loading problem

Andreas Bortfeldt [a,1], Jörg Homberger [b,*]

[a] University in Hagen, Department of Information Systems, Profilstr. 8, 58093 Hagen, Germany
[b] University of Applied Sciences Stuttgart, Faculty Geomatics, Computer Science and Mathematics, Schellingstr. 24, 70174 Stuttgart, Germany

ARTICLE INFO

Available online 28 September 2012

Keywords:
Vehicle routing
3D packing
Vehicle routing and loading problem
Heuristic

ABSTRACT

The Vehicle Routing and Loading Problem (VRLP) results by combining vehicle routing, possibly with time windows, and three-dimensional loading. Some packing constraints of high practical relevance, among them an unloading sequence constraint and a support constraint, are also part of the VRLP. Different formulations of the VRLP are considered and the issue is discussed under which circumstances routing and packing should be tackled as a combined task. A two-stage heuristic is presented following a "packing first, routing second" approach, i.e. the packing of goods and the routing of vehicles is done in two strictly separated stages. High quality results are achieved in short computation times for the 46 VRLP instances recently introduced by Moura and Oliveira. Moreover 120 new large benchmark instances including up to 1000 customers and 50,000 boxes are introduced and results for these instances are also reported.

1. Introduction

Packing goods and delivering them to customers at different locations are two distribution logistics operations that are of vital importance for manufacturing and trading companies. On the one hand, their effective planning and implementation influence the costs that occur in a company to a considerable extent. On the other hand, the quality of the packing and transport processes is crucial in determining whether or not the company is sufficiently oriented to the needs of its customers. Some fundamental requirements are that goods reach the customers on time, arrive undamaged and in the quantities ordered, and that unloading the goods can be accomplished easily and in a time-saving manner (Moura and Oliveira [46]). Successful entrepreneurs today operate in logistics networks and criteria such as due date and quantity reliability are elementary qualifications for "network capability" (Dangelmaier et al. [18]).

Packing and transport processes in a company can display a high degree of interdependence. In this case it is important from the viewpoint of the company that both operations are carried out together efficiently and in high quality. For example, there is little advantage in having a well-filled truck loading space if packed goods are for customers who are located far away from each other, so that it is uneconomical, or even impossible, to deliver the goods in a single route.

In order to take account of the interdependence mentioned before, this paper deals with the Vehicle Routing and Loading Problem (VRLP) that results from a combination of the Vehicle Routing Problem (VRP) with a three-dimensional packing problem for rectangular pieces. Different formulations of the VRLP are compared and the question of a suitable problem formulation is examined. In the main part a heuristic for solving the VRLP is presented, which is capable, based on existing test results, of generating high-quality solutions in relatively short computation times.

The next section is dedicated to the formulation of the VRLP and provides an overview of the relevant literature. Section 3 describes the heuristic method. Section 4 reports on numerical experiments while the paper is summarised in Section 5.

2. Problem formulation and overview of the literature

In the following, a problem formulation of the VRLP is suggested and the literature on the VRLP and related problems is reviewed. Moreover, there is an examination of the circumstances under which routing and packing of goods are to be carried out as a combined planning operation.

2.1. Suggested problem formulation

The Vehicle Routing and Loading Problem (VRLP) is defined as follows:

– Let a vehicle fleet be given that consists of an unlimited number of vehicles with an identical rectangular loading space with length L, width W and height H.

* Corresponding author. Tel.: +49 711 8926 2511; fax: +49 711 8926 2553.
E-mail addresses: andreas.bortfeldt@fernuni-hagen.de (A. Bortfeldt), joerg.homberger@hft-stuttgart.de (J. Homberger).
[1] Tel.: +49 2331 987 4433; fax: +49 2331 987 4447.

0305-0548/$ - see front matter
http://dx.doi.org/10.1016/j.cor.2012.09.005

– Let $V=\{0,1,\ldots,n\}$ be a set of $n+1$ nodes that correspond to a depot (node 0) and n customers (nodes $1,\ldots,n$). Let E be a set of undirected edges (i,j) that connect all node pairs and let $G=(V,E)$ be the resulting graph. Let a distance c_{ij} ($c_{ij}>0$) be assigned to each edge (i,j) ($0 \le i < j \le$ n).
– Each customer i ($i=1,\ldots,n$) is to be supplied with a set of m_i rectangular packing pieces (boxes) I_{ik} ($k=1,\ldots,m_i$) which are initially located at the depot. Let the box I_{ik} have the length l_{ik}, the width w_{ik}, the height h_{ik} and the volume s_{ik} ($i=1,\ldots,n$, $k=1,\ldots,m_i$).
– Let the loading space of each vehicle be embedded in the first octant of a Cartesian coordinates system in such a way that the length, width and height of the loading space lie parallel to the x, y and z axes. The placement of a box I_{ik} in a loading space is given by the coordinates x_{ik}, y_{ik} and z_{ik} of the corner of the box that is closest to the origin of the coordinates system; in addition, an orientation index o_{ik} indicates which of the six possible spatial orientations is selected ($i=1,\ldots,n$, $k=1,\ldots,m_i$). A spatial orientation of a box is given by a one-to-one mapping of the three box dimensions and the three coordinate directions. A packing plan P for a loading space comprises one or more placements and is regarded as feasible if the following three conditions hold: (FP1) each placed box lies completely within the loading space; (FP2) any two boxes that are placed in the same truck loading space do not overlap; (FP3) each placed box lies parallel to the surface areas of the loading space. Fig. 1 (see [11]) shows a loading space with placed boxes.
– A feasible route R is a sequence of three or more nodes that starts and ends at the depot. A customer must not occur more than once in the sequence. A solution of the VRLP is a set of v ordered pairs (R_l, P_l), where R_l is a route and P_l is a packing plan ($l=1,\ldots,v$). In order to be feasible, a VRLP solution must fulfil the following three conditions: (F1) all routes R_l and packing plans P_l are feasible ($l=1,\ldots,v$); (F2) each customer belongs to exactly one route R_l ($i=1,\ldots,n$); (F3) the packing plan P_l for a route R_l contains exactly one placement ($l=1,\ldots,v$) for each customer i of R_l and each of its boxes I_{ik} ($k=1,\ldots,m_i$).
– Given a VRLP instance, a feasible solution is to be determined that minimises the number of routes with higher and the total travel distance with lower priority. Both criteria can be combined in a lexicographic objective function f=(number of routes, total travel distance). Alternatively, minimising the distance can have the higher priority. It is also possible that just one of the objective criteria has to be minimised.

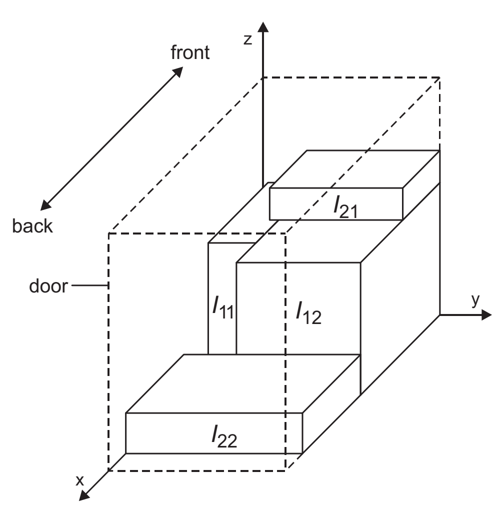

Fig. 1. Loading space of a truck with placed boxes.

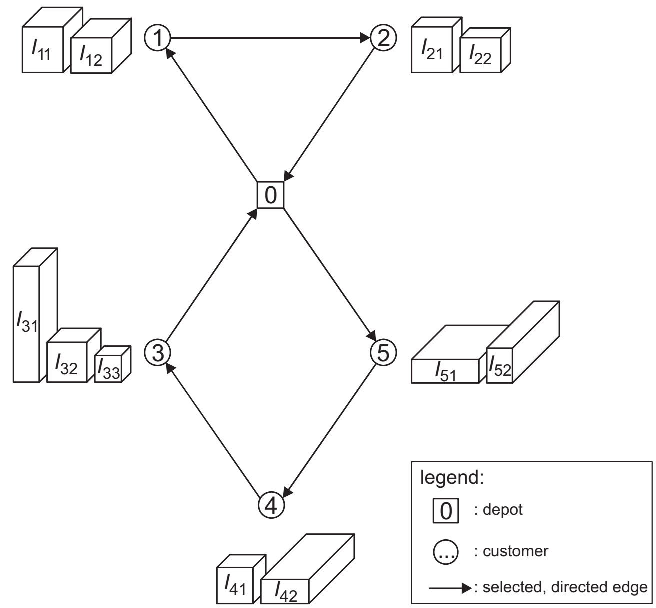

Fig. 2. VRLP instance with roβutes of a solution.

The previously formulated task will be termed below as the basic version of the VRLP. Fig. 2 (see [11]) illustrates a simple VRLP instance and the routes of a possible solution (while edges that do not belong to these routes are not shown). In addition to the basic version of the VRLP, one or more of the following constraints can be included in the problem.

2.1.1. (C1) Time window constraint

It is assumed that each vehicle moves at a constant speed (1 length unit/ 1 time unit). A delivery time window $[a_i, b_i]$ ($b_i > a_i$) and a service time s_i ($s_i \ge 0$) for unloading the freight ordered by the customer are assigned to each customer i ($i=1,\ldots,n$). In order to comply with the time window of customer i, a vehicle must arrive at this customer's location at time b_i at the latest. If the vehicle arrives at customer i before the time a_i, it must wait until time a_i with the delivery; however, the time window is not violated. There is an additional time window $[a_0,b_0]$ for the depot that limits the total tour duration and tour distance.

2.1.2. (C2) Unloading sequence constraint

When customer i ($i=1,\ldots,n$) is visited, it must be possible to unload all his boxes I_{ik} ($k=1,\ldots,m_i$) exclusively by using movements parallel to the longitudinal axis of the loading space. Consequently, no box demanded by another customer that is visited later than customer i must be placed over a box I_{ik} (of customer i) or between I_{ik} and the rear of the vehicle (cf. Fig. 1). This constraint is also referred to as the LIFO (last in, first out) constraint.

2.1.3. (C3) Weight constraint

Each box I_{ik} has a positive weight d_{ik} ($i=1,..,n$, $k=1,\ldots,m_i$) and the total weight of all the boxes placed in a truck must not exceed an upper limit of the load weight D which is identical for all vehicles.

2.1.4. (C4) Orientation constraint

In the general case, up to five spatial orientations may be inadmissible for certain boxes. Often one or two box dimensions are ruled out as the height dimension, while horizontal 90° rotations

Table 1
Variants of the 3D VRLP in the comparison.

3D VRLP components (cf. Section 2.1)	Gendreau et al. [29]	Moura and Oliveira [43]
Basic version	As defined in Section 2.1	As defined in Section 2.1
Objective function	Total travel distance	1. Number of routes 2. Total travel distance
(C1) Time window constraint	No	Yes
(C2) Unloading sequence constraint	Yes	Yes
(C3) Weight constraint	Yes	No
(C4) Orientation constraint	This-way-up constraint	Height constraint
(C5) Support constraint	Yes	Yes ($a=100\%$)
(C6) Stacking constraint	Yes	No
Other constraints	No	Lateral boxes support

are permitted (height constraint). If the height dimension is definitely given for all boxes, this constraint is also referred to as the "this-way-up" constraint.

2.1.5. *(C5) Support constraint*

If a box is not placed on the truck floor a certain percentage a of its base area is to be supported by other boxes. If b and c are the horizontal dimensions of a box above the floor, a portion of the base area of size $a*b*c$ is to be placed on other boxes.

2.1.6. *(C6) Stacking constraint*

A fragility flag f_{ik} is assigned to each box I_{ik} ($i=1,..,n$, $k=1,\ldots,m_i$). If a box is fragile ($f_{ik}=1$) only other fragile boxes may be placed on its top surface, whereas both fragile and non-fragile boxes may be stacked on a non-fragile box ($f_{ik}=0$).

Of course, a two-dimensional (2D) VRLP variant may also be considered, i.e. the VRLP with two-dimensional pieces and 2D loading spaces. Certain constraints then no longer apply and others have to be adapted to the 2D case.

2.2. *Related problems*

The VRLP clearly derives from the VRP (Vehicle Routing Problem), or, with the presence of the time window constraint (C1), from the Vehicle Routing Problem with Time Windows (VRPTW). If the fact that boxes and loading spaces are three-dimensional is ignored and instead only a one-dimensional volume constraint or the weight constraint (C3) is considered, then the VRLP is reduced to the VRP or to the VRPTW (if time windows do exist).

We refer to Toth and Vigo [58] for a comprehensive survey of the research area of route planning. The VRP and the VRPTW are known to be NP-hard. Although important progress has been made recently in the exact calculation of the VRP and the VRPTW (cf. Fukasawa et al. [26], Baldacci et al. [3,4]), meta-heuristic methods for solving large problem instances with hundreds of customers remain indispensable. More recent surveys of meta-heuristic methods for the VRP have been developed by Cordeau and Laporte [16], Cordeau et al. [17] and Laporte [37]. Successful meta-heuristic VRP methods have been suggested by Taillard [55], Rochat and Taillard [53], Toth and Vigo [59], Prins [52], Mester and Bräysy [44], Marinakis et al. [40,41], Derigs and Kaiser [19], Kytöjoki et al. [36] and Nagata [47]. Particularly effective meta-heuristic methods for the VRPTW have been introduced by Homberger and Gehring [33] (see also [27]), Bräysy et al. [15], Bent and Van Hentenryck [5], Bouthillier and Crainic [13], Mester and Bräysy [43] and by Pisinger and Ropke [51]. Recent solution methods were designed by Alvarenga et al. [2] and Brandão de Oliveira and Vasconcelos [14].

Various 3D packing problems are relevant in the context of the development of solution methods for the VRLP. Together with the Bin Packing Problem these include the CLP (or 3D Knapsack Problem) and the Strip Packing Problem (SPP), which are all NP-hard (cf., e.g., Pisinger [50] and Martello et al. [42]). In the 3D-SPP a set of boxes is to be packed completely into a container with a fixed cross-section so that the variable container length is minimised. In recent years, meta-heuristic solution methods have mainly been suggested for solving the particularly "cumbersome" 3D packing problems. An extensive bibliography of more recent research work on cutting and packing (C&P) problems is presented in Wäscher et al. [60] whereas an older C&P bibliography originates to Dyckhoff et al. [20]. Procedures for the 3D –BPP have been suggested among others by Terno et al. [57], Martello et al. [42] and by Faroe et al. [23]. Efficient meta-heuristic algorithms for the CLP, that consider some of the packing constraints introduced above, were developed by Bortfeldt and Gehring [9] (see also [28]), Eley [21], Pisinger [50], Mack et al. [39], Bischoff [7], Moura and Oliveira [45] and more recently by Parreño et al. [48,49], Fanslau and Bortfeldt [22], Gonçalves and Resende [31], and Weng et al. [61]. Meta-heuristic solution methods for the 3D-SPP have been established among others by Bortfeldt and Gehring [8], Bortfeldt and Mack [10] and Allen et al. [1].

2.3. *Literature on the VRLP*

Gendreau et al. [30] and Moura and Oliveira [46] were the first to formulate the three-dimensional VRLP, to suggest solution methods and to introduce benchmark instances. The VRLP variants dealt with in the two papers are compared to each other in Table 1 and confronted with the formulation given here.

A comparison of the two problem variants reveals differences with regard to the objective function and several constraints: Gendreau et al. [30] only minimise the total travel distance, and do not take account of time windows, but take all the packing constraints defined above into consideration. In contrast, Moura and Oliveira [46] consider a VRLP with an objective function derived from the VRPTW, but do not take either the weight or the stacking constraint into consideration. However, they observe the lateral box support as an additional stability criterion. This is defined as the percentage of boxes that have contact on at least three sides with other boxes or the lateral surfaces of the loading space. The lateral box support should be as high as possible to prevent horizontal movements of the freight.

The problem variants examined by Gendreau et al. [30] and Moura and Oliveira [46] can be subsumed under the more general formulation of the VRLP suggested here. By this new formulation the problem core of the VRLP (also called basic version) is more strongly emphasised as it is clearly separated from additional constraints which may occur in practical use cases. For a mathematical model of the VRLP the reader is referred to Moura and Oliveira [46].

Gendreau et al. [30] suggest a two-stage tabu search algorithm (in short TSA). The external TSA is used directly for planning the routes while the internal TSA solves a 3D strip packing problem for loading a vehicle in accordance with a given customer sequence. A move of the external TSA is defined through the displacement of a customer to another route. For each move tested by the external TSA the internal TSA is to be invoked again for all changed routes.

The two heuristics suggested by Moura and Oliveira [46] adopt different approaches. In the first method of the GRASP type, each solution is constructed in a single coherent operation, i.e. routes are planned and simultaneously vehicles are loaded. With the second method, for each solution all routes are initially determined with an integrated VRPTW method and only then the vehicles are loaded. This is done by solving a CLP instance by means of a GRASP packing method.

There are some other articles that deal with the 3D–VRLP variant introduced by Gendreau et al. [30]. Tarantilis et al. [56] designed a hybrid procedure combining the strategies tabu search and guided local search while Fuellerer et al. [25] developed an ant colony algorithm (ACO). Bortfeldt [12] proposed a hybrid algorithm including a tabu search method for routing and a tree search procedure for packing boxes into a loading space. Besides the 3D case the two-dimensional VRLP is studied as well. Iori et al. [34] have presented an exact branch-and-cut algorithm for the two-dimensional VRLP without a time window constraint, while Gendreau et al. [29] have devised a meta-heuristic solution method for the same problem. Fuellerer et al. [24] have proposed an ACO procedure for the 2D-VRLP.

An up-to-date survey on routing problems with loading constraints was penned by Iori and Martello [35]. The paper by Bortfeldt and Homberger [11] is a preliminary (German written) version of the paper at hand.

2.4. When should a VRLP modelling be selected?

In routing problems vehicles have generally a load capacity that can be interpreted as the maximum freight volume. The one-dimensionality of the capacity results in the fact that a routing plan can be feasible with regard to the capacity constraint while it cannot be implemented at the same time. To take a very simple example, two cubes with length of 2 m cannot be packed in a cube with length of 3 m, although the volume of both cubes together makes up less than 60% of the larger cube volume. It can thus be concluded that with shipments of packaged goods the practicability of a planned route can only be guaranteed *unerringly* if the route is accompanied by a packing plan. In other words, modelling a practical use case by means of the VRLP has the advantage that calculated routing plans can definitely be implemented. Hence, if the requirements for a generation of packing plans (e.g. availability of data and software) are fulfilled, then VRLP modelling generally represents a useful option.

Nevertheless, in various situations modelling of a practical transport problem as a VRP or VRPTW without providing a packing plan is completely sufficient. Two examples described by Haessler [32] are included as evidence of this. One case concerns the transport of extremely heavy goods, namely industrial trucks, in containers. The weight constraint dominates as against the volume constraint: if several goods can be packed in a container in terms of their weight then as a result of their high density they take up relatively little space and can thus be packed together inside the container. The packing problem is therefore essentially one-dimensional and the VRP (or VRPTW) would be the suitable model for route planning. The second example deals with the transport of lighter vehicle parts in trucks. The vehicle parts are packed on special carriers that take up both the full width and height of the vehicle and different parts of the vehicle length. Although in this case the volume constraint dominates, the resulting practical problem here is obviously once again one-dimensional and can be solved as a VRP. As the examples show, the "right" model problem is to be selected on the basis of the concrete circumstances and, of course, both the VRP and the VRPTW retain their significance as practical optimisation problems.

3. Solution method

In the following, a new heuristic for the 3D-VRLP is described. First, a generic approach is introduced that is based on an (abstract) VRP method and on an (abstract) method for the 3D strip packing problem (see Section 2.2). In a second step two suitable algorithms for the VRP and the 3D-SPP are selected from literature to devise an applicable method. Finally, the scope of the heuristic is extended by an additional component.

3.1. Method approach

The main idea of the heuristic is to separate packing boxes in loading spaces and constructing routes from each other in order to structurally reduce the packing effort and to thus obtain an efficient heuristic method. Consider the following two-stage method approach (see Algorithm 1 below):

1. In the *first* method stage ("packing"), the boxes of each customer are packed in a separate segment of a truck loading space. For each customer a 3D strip packing problem is solved separately. The 3D-SPP instance for customer i is well-defined by the width W, the height H of the loading space and the boxes I_{ik} $(k=1,\ldots,m_i)$ ordered by customer i. The calculated minimal loading lengths $ll(i)$ and the customer-related packing plans $P(i)$, which are referred to as individual arrangements, are stored $(i=1,\ldots,n)$.
2. In the *second* method stage ("routing"), the VRP or VRPTW instance derived from the VRLP instance is solved, whereby v routes are calculated. The so-called loading length constraint states that only those routes R are feasible for which the sum of loading lengths $ll(i)$ of the customers $i \in R$ visited in the route does not exceed the loading space length L:
$$\sum_{i \in R} ll(i) \le L \tag{1}$$
Clearly, the loading length constraint plays the role of the capacity constraint of the VRP(TW).
3. After both stages, a packing plan P_l is finally determined for each of the v previously calculated routes R_l. In the packing plan P_l the individual arrangements $P(i)$ of all customers on the route R_l are placed behind each other starting at the front and going to the back (cf. Fig. 1) of the loading space $(l=1,\ldots,v)$. The sequence of the individual arrangements $P(i)$ is selected inversely to the sequence of the customers in the route R_l. Then the complete VRLP solution is presented.

When solving the 3D-SPP instances in stage 1, it must be ensured that the required packing constraints (C4) to (C6) are fulfilled where applicable. In stage 2, the time window constraint (C1) and the weight constraint (C3) are to be observed if applicable, whilst the LIFO constraint (C2) is finally ensured by the implemented sequence of the individual arrangements for all vehicles. If a weight constraint (C3) is required, this represents – besides the loading length constraint (1) (see above) – another one-dimensional capacity constraint of the integrated VRP(TW). Compliance with the one-dimensional loading length constraint in the solution of the route planning problem ensures that the 3D packing plans for the vehicles do not exceed the length of the loading spaces.

In brief, the method approach bases the solution of a VRLP instance on the independent solution of *n* 3D-SPP instances and one VRP(TW) instance. It can therefore be classified as *generic*: to solve the two partial problems 3D-SPP and VRP(TW) any methods for these tasks that takes all the given constraints into account can be integrated in the VRLP heuristic. The two-stage heuristic is called P1R2 (“packing first, routing second”).

Algorithm 1. Frame procedure of heuristic P1R2.

```
P1R2_frame (in: instance data, parameters, out: best solution s_best)
// packing stage: provide separate packing plans for all customers
for i:= 1 to n do
   solve 3D-SPP instance given by width W, height H and box set I_ik,k = 1,…,m_i,
   of customer i by chosen 3D-SPP method;
   store resulting 3D packing plan P(i) and appropriate loading length ll(i);
endfor;
// routing stage: determine one routing plan for all customers
solve VRP(TW) instance given by
   - graph G (including distances c_ij,i = 0,1,…,n,j = 0,1,…, n, i ≠ j)
   - vehicle length L (as capacity)
   - customer loading lengths ll(i), i = 1,…,n (as demands)
   - weight data D, d_ik,i = 0,1,…,n,k = 1,…,m_i, and
   - time windows [a_i,b_i],i = 0,1,…,n, where necessary
by chosen VRP(TW) method;
store resulting routing plan sr_best, including vector R(l) of routes, l = 1,…,v;
// provide complete solution
for l := 1 to v do
   combine packing plans P(i) of all customers i visited at route R(l)
   in loading space of one vehicle such that LIFO constraint is met;
   store resulting packing plan P(l);
endfor;
provide best solution s_best including sr_best and P(l), l = 1,…,v;
end.
```

3.2. Integrated partial methods

To solve the 3D-SPP the tabu search algorithm proposed by Bortfeldt and Gehring [8] is integrated, whilst the hybrid heuristic presented by Homberger and Gehring [33] is used to solve the VRP(TW).

The 3D-SPP method can be characterised as follows:

– If an instance is handled by the 3D-SPP method a sequence of 3D-CLP instances with decreasing container lengths is independently solved. This sequence is finished only if for a 3D-CLP instance (with fixed container length) no solution has been found that includes all given boxes.
– Generating one feasible solution of a 3D-CLP instance is based on an elementary packing operation by which one residual packing space, i.e. an empty rectangular space inside the container, is filled by a local arrangement made up by one or two neighbouring blocks. A block consists of multiple boxes of the same box type that are arranged in same spatial orientation (cf. Fig. 3, and [11]). After a residual packing space was filled further residual packing spaces (inside the filled one) result which are processed later on. A complete feasible solution is then produced by a sequence of elementary packing operations. The first residual packing space to be filled is always the (empty) container. Afterwards the rule is observed that the smallest available residual packing space is processed first.
– In general a residual packing space can be alternatively filled by different local arrangements. All feasible local arrangements for a given residual space are sorted by some evaluation criteria. Based on this sorting a special encoding is introduced that stipulates which arrangements are to be applied in the residual spaces of a feasible solution. The neighbourhood search of the TSA is carried out then within the space of the encoded solutions.
– A box set is called strongly heterogeneous, if there are only few boxes per box type (a type is given by the box dimensions). It is called weakly heterogeneous if there are only few box types and many boxes per type. Since the block concept (as sketched above) is applied, the TSA is tailored to weakly heterogeneous box sets. As recent experimental results show (cf. Bortfeldt and Mack [10]), the TSA still belongs to the most effective methods for the 3D-SPP.
– The integrated VRPTW method can be described as follows:
– The algorithm by Homberger and Gehring [33] is designed as a two-phase meta-heuristic which minimises the number of vehicles in the first phase by means of an evolutionary strategy and minimises the total travel distance in the second phase by means of a TSA.
– The (μ,λ) evolutionary strategy of the first phase generates offspring using three familiar operators for route modification (Or-opt, 2-opt* and Single-Interchange). Of particular importance for a successful reduction of routes is the lexicographic evaluation function, which orients the search towards cancelling the smallest route (with a minimum number of customers). Another important component is a special heuristic for removing customers from the smallest route.
– The random-based TSA of the second phase also uses the three mentioned operators to generate neighbouring solutions. The tabu list stores links between nodes and a solution is set tabu if at least one of its links can be found in the tabu list. The number of vehicles achieved in the first phase is not worsened again when minimising the distance.

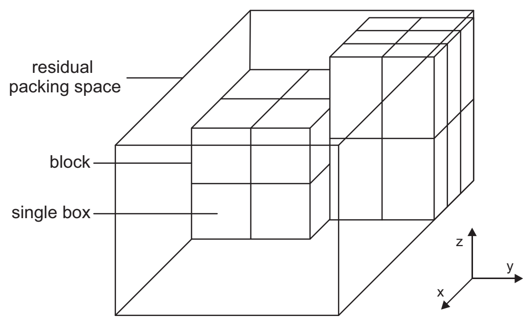

Fig. 3. Residual packing space with two blocks.

– The algorithm proposed by Homberger and Gehring [33] was at that time one of the most successful VRPTW methods. It still represents a high-quality method — at least regarding the minimisation of the number of vehicles.

3.3. Scope of application and customer combinations

The heuristic P1R2 is not equally suitable for all VRLP instances. A precondition for successful application is that the customers are at least mainly so-called segment customers. A segment customer is characterised by a set of boxes that can be arranged in a segment ($l \times W \times H$, $l < L$) with little loss. It can be agreed, for example, that a customer is deemed to be a segment customer if the integrated 3D-SPP method achieves a volume utilisation of at least 80%.

There may be several reasons why a customer is not a segment customer. The volume of the customer's boxes might be too low to achieve a sufficient volume utilisation. Moreover, even with a relatively large volume of the boxes of a single customer the proportions between the box dimensions and the loading space dimensions W and H could be so unfavourable that a high loss near to the surfaces of the loading space cannot be prevented. This typically happens when the ordered boxes of the customer correspond to only one or two box types. Finally, large volume loss can also occur if a customer does in fact have boxes of several types, but the boxes are relatively large in comparison with the loading space.

A significant reduction in volume utilisation caused by some of the individual arrangements will have a negative effect on the minimisation of the number of routes and the total travel distance. In order to counteract this, the principle of packing the boxes belonging to different customers separately is to be "softened" to some degree.

If joint packing of boxes in the same arrangement is permitted for smaller groups of customers with weak volume utilisation of their individual arrangements, it may be possible to achieve a much higher volume utilisation since the reasons referred to above no longer apply in some cases. This approach is implemented by customer combinations (in short CC).

A customer combination is defined by a partial route $(i_1,\ldots,i_p)$ of a given route R including $p \geq 2$ customers (without depot) and a corresponding box arrangement that is called collective arrangement. The collective arrangement is generated by the integrated 3D-SPP method as solution of the 3D-SPP instance given by the loading space dimensions W, H and the set of the boxes belonging to *all* customers in the partial route. Hence, the boxes of multiple customers are packed together in one segment of the loading space of minimum length. Let l be this loading length, then the saving of the customer combination is defined by the difference between the sum of the loading lengths $ll(i)$ of all relevant customers $i \in \{i_1,\ldots,i_p\}$ and l. While a customer combination without a positive saving is useless, a positive saving is not guaranteed. In order to enable higher savings the LIFO constraint (C2) is neglected *inside* a collective arrangement of a customer combination.

Since the VRPTW method only accepts feasible routes any new route R is rejected that does not meet the load length constraint (1) (see Section 3.2), i.e. the length defect of R – given as difference between the sum of the loading lengths $ll(i)$ $(i \in R)$ and the loading space length L – is positive. However, if customer combinations are admissible and a route R turns out to be infeasible because of a positive length defect a special procedure (cf. Algorithm 2 below) tries to generate one or more customer combinations with positive savings for *disjoint* partial routes $(i_1,\ldots,i_p)$ of R with $p \geq 2$ customers. Each collective arrangement replaces the individual arrangements of the customers who belong to the customer combination. If the total saving achieved by all constructed customer combinations is greater than or equal to the length defect, the route R is then feasible and it will be accepted for a further routing plan. Otherwise, the route is still infeasible and will be rejected. The generation of customer combinations is stopped immediately after a feasible route was achieved.

The approach can be illustrated by a simple example. Let the route $R=(0,2,3,6,1,5,4,7,0)$ be given with a supposed length defect of 5 units. Now two customer combinations are constructed: let cc_1 include the customers 3 and 6 while cc_2 includes the customers 5, 4 and 7, i.e. the boxes of customers 3 and 6 are packed together into one segment and the same applies to the boxes of customers 5, 4 and 7. Suppose that the loading lengths of the individual arrangements of the involved customers are $ll(3)=10$, $ll(6)=12$, $ll(5)=9$, $ll(4)=11$ and $ll(7)=8$ while the loading lengths of the collective arrangements are $l(cc_1)=18$ and $l(cc_2)=26$. Thus, by replacing the five individual arrangements by the two collective arrangements the length defect of 5 units is compensated and the route R is now feasible. Algorithm 2 specifies the procedure in detail and is commented below.

– A customer i is permitted to enter a customer combination only if the volume utilisation $VolUtil(i)$ of his individual arrangement does not exceed a specified limit $VolUtilLimit$. Note that with the combination of customers with poor individual arrangements there should be a greater chance of achieving appreciable savings. The (input) parameter $qUbad$ indicates a specific percentage of the individual arrangements with the poorest volume utilisations; the value $VolUtilLimit$ is then the maximum volume utilisation of the $qUbad*n$ individual arrangements with the poorest volume utilisations. E.g., for a number of customers $n=50$ and $qUbad=10\%$, $VolUtilLimit$ is obtained as the volume utilisation of the fifth poorest individual arrangement.
– Starting from a customer i, a customer combination contains at least the successor customer in the route, denoted by $i+$. If i or $i+$ does not fulfil the above said volume utilisation constraint, then there is no customer combination starting with i. Otherwise a collective arrangement is generated for i and $i+$. The customer combination may be supplemented by other successors in the route, and for each extended partial route a collective arrangement is established. The generation of a customer combination starting with customer i is halted if one of the following conditions is fulfilled: (i) there is no next customer in the route; (ii) the successor customer does not comply with the volume utilisation constraint; (iii) the collective arrangement has a volume utilisation less than $VolUtilLimit$; (iv) the partial route reaches the maximum length given by the parameter $maxlCC$. The volume utilisation constraint for the collective arrangement takes into account that a low volume utilisation hardly leads to a significant saving.

– A customer combination is only accepted for the route, if the saving exceeds a limit *minSav*; where *minSav* is the product of a minimal percentage savings parameter *qminSav* and the sum of the lengths of the individual arrangements of all the customers in the customer combination.
– An initial attempt is made to generate a customer combination starting with the first customer in the route. If no customer combination starting with the customer *i* is generated or accepted, the next attempt is made to generate a customer combination for the route starting with the successor $i+$ of *i*. If a customer combination *cc* is generated and accepted, the next customer combination starts where applicable with the successor of the last customer in *cc*. The generation of customer combinations ends if the route became feasible, or if the end of the route is reached.
– The accepted customer combinations for the route are saved together with the route once the route proves to be feasible, so that the combinations can be later used if necessary to prepare a 3D packing plan. Furthermore, all customer combinations that have ever been generated are filed in a separate memory and are made accessible when required. Therefore each collective arrangement is generated once only.
– In addition, there will be no attempt to obtain a feasible route through customer combinations, if the length defect exceeds a suitable percentage of the loading space length *L*, i.e. if the length defect is too large. In this way superfluous search effort is avoided. The percentage of *L* is given by the parameter *qlExcess* (see Section 4.3).

The whole search in the two stages of P1R2 is now divided into four phases *A* to *D*; phase *A* belongs to the first stage and phases *B* to *D* belong to the second stage. In phase *A* the packing plans for individual customers are generated. Phase *B* corresponds to the first phase of the VRPTW method and serves the minimisation of the number of routes while in phase *C*, corresponding to the second phase of the VRPTW method, the total travel distance is minimised (without using customer combinations). Starting from the best solution after phase *C* another distance-minimisation is then performed if necessary in phase *D*, but this time using customer combinations.

Finally, the concept of customer combinations should be discussed under different aspects:

– Customer combinations are introduced to maintain an acceptable level of solution quality even if not all customers are segment customers (see above). However, by their use the original strict "packing first, routing second" pattern is abandoned. Hence, the whole algorithm becomes more difficult and a considerable additional computational effort will result.
– The concept was designed as a feature to be added to an effective routing procedure and is not meant as a "second routing procedure". Therefore, the order of a given route is not changed by customer combinations. Hereby, the computational effort can be kept at least in certain limits and the implementation can be kept simple.
– As mentioned above, the LIFO constraint (C2) is neglected within collective arrangements to improve the effect of customer combinations. Moreover, by this stipulation it becomes possible to use packing algorithms that are not tailored to the LIFO constraint as the one by Bortfeldt and Gehring [8]. On the other hand, there are some 3D–SPP methods observing the LIFO constraint, e.g., the tree search method introduced in Bortfeldt [12]. By means of such a 3D–SPP method it could be ensured that the LIFO constraint is also strictly satisfied within collective arrangements of customer combinations. This option remains a topic of further research.
– If the LIFO constraint (C2) is not observed for customer combinations (as in this paper) some additional effort for unloading will result. However, this additional effort obviously depends on parameter *maxlCC* and it can be kept low by choosing small values (e.g., *maxlCC*=3).

The numerical results presented below will allow for a further evaluation of customer combinations.

Algorithm 2. Procedure testloadlengths.

```
testloadlengths (in: route, lengthDefect, out: lengthDefect, CCused)
CCused := ∅; // no customer combinations (CC) used
next customer to be processed i := first customer on route;
// form CC as long as length defect is positive and end of route not reached
while lengthDefect> 0 and i != depot do
   if a CC can be started with customer i then
      build customer combination cc of maximum length starting with i;
   endif;
   if savings(cc)is sufficiently great then
      CCused := CCused ∪ {cc}
      lengthDefect := lengthDefect - savings(cc);
      next customer i:= successor of last customer of cc;
   else
      next customer i:= i+;
   endif;
endwhile;
end.
```

4. Numerical experiments

The heuristic P1R2 was implemented in C++. In the sequel benchmark instances for the VRLP suggested in the relevant literature are described and selected for numerical experiments. Moreover, new larger problem instances are introduced. The test results are presented and analysed following information on the parameterisation of the heuristic.

4.1. Benchmark instances from literature

Moura and Oliveira [46] have introduced 46 benchmark instances for the VRLP that correspond to the VRLP variant that they examined (cf. Table 1). The instances are derived from the VRPTW instances from Solomon [54] and from some CLP instances from Bischoff and Ratcliff [6]. In all instances the number of customers is 25, each instance has a weakly heterogeneous box set. The instances are divided into the classes I1 and I2 and at the same time into the two groups GI and GII:

- The instances in class I1 are derived from the 12 Solomon instances in group R1 (cf. Solomon [54]. The customers are equally distributed in the plane. A distinctive characteristic of this class are narrow depot time windows. The generated solutions therefore have a larger number of routes, with each having a few customers only.
- The instances in class I2 are derived from the 11 Solomon instances in group R2. The customers are again distributed equally in the plane. This class is characterised by wide depot time windows. The generated solutions therefore have a lower number of routes, with each having a large number of customers.
- The instances in group GI have customers with 42 boxes on average and the total number of boxes per instance is 1050.
- The instances in group GII have customers with 62 boxes on average and the total number of boxes per instance is 1550.

A systematic combination results in the subgroups GI–I1, GI–I2, GII–I1 and GII–I2 with a total of 46 instances, which are available on the ESICUP website www.fe.up.pt/∼esicup.

Gendreau et al. [30] suggest 32 VRLP instances that correspond to the VRLP variant without time windows (cf. Table 1). Twenty-seven instances are derived from well-known VRP instances in the relevant literature and are available under www.or.deis.unibo.it/research.html. Note that for each of the 27 instances a maximum number of vehicles is stipulated. The other five instances were provided by an Italian furniture manufacturer. The customer numbers in all instances range between 15 and 100. Averaged over the 27 instances there are only 2 boxes assigned to a customer, whereas in the five practical instances the average number of boxes per customer is about 4 boxes. In addition, boxes are found that are quite large in comparison with the loading space of a vehicle. One can expect that the heuristic P1R2 is not very suitable for the 27 instances with 2 boxes per customer since it packs the boxes of different customers mainly in separated loading space segments (see Section 4.6).

4.2. New large benchmark instances

For the CVRP or VRPTW (without more-dimensional loading constraints) large benchmark instances with up to 1200 customers are available for several years (Gehring and Homberger [27], and Li et al. [38]). The evaluation of new VRPTW approaches by means of large instances with up to 1000 customers became usual in the last decade reflecting the fact that such large instances have also to be solved in practice, e.g. by U.S. American parcel service providers. However, the benchmark instances that were proposed so far for routing problems with loading constraints are rather small, never exceeding 125 customers with some hundred boxes. Therefore, a total of 120 new VRLP instances are introduced here, divided into 6 test cases, each with 20 instances (cf. Table 2). All the instances of a test case correspond with regard to the number of customers and the total number of boxes for all customers.

Table 2
New problem instances for the VRLP.

Test case (per 20 instances)	Customers per instance	Boxes per instance
1	100	5000
2	200	10000
3	400	20000
4	600	30000
5	800	40000
6	1000	50000

Table 3
Parameter sets for the test.

Parameter	No. of customers		
	$n < 100$	$100 \leq n < 600$	$600 \leq n$
Time limit phase B,C	each 30 s	each 240 s	each 240 s
Time limit phase D	60 s	480 s	480 s
qUbad	65%	15%	5%
qminSav	10%	15%	20%
qlExcess	10%	5%	15%
maxlCC	3	3	3

The 120 VRLP instances are derived from the VRPTW instances by Solomon [54] or by Gehring and Homberger [27] and from the CLP instances by Bischoff and Ratcliff [6]. To generate the 20 instances of a VRLP test case with the given number of customers, the first five VRPTW instances with the same number of customers are taken from groups R1 and R2 respectively (cf. Solomon [54], Gehring and Homberger [27]). In addition, the first five CLP instances of the CLP test sets BR2 (with 5 box types per instance) and BR4 (with 10 box types per instance) are always to be used for each VRLP test case. For each VRLP instance the set of box types is provided by one of the selected CLP instances. For each customer several box types are then selected randomly from this set. Finally, the corresponding numbers of boxes (per type and customer) are determined at random so that the given total number of boxes (cf. Table 2) per instance is observed.

In this way some of the VRLP instances are generated with narrow depot time windows and others with wide depot time windows (cf. Section 4.1). At the same time the instances show a wide variety of box sets and the box numbers achieve extreme sizes. The same constraints as with the VRLP instances from Moura and Oliveira are required, i.e. not only the time window constraint (C1) needs to be fulfilled but the packing constraints (C2, C4, C5) as well. The support constraint (C5) is required with the percentage $a = 100\%$.

4.3. Parameterisation of the heuristic P1R2

An AMD Athlon PC (FX-60 Dual Core, 2.61 GHz, 2 GB RAM) was used for the numerical experiments. For the integrated methods for solving the 3D-SPP and VRPTW the original parameterisation (see Bortfeldt and Gehring [8] and Homberger and Gehring [33]) was maintained in the test of the VRLP heuristic, with the exception of the time limits. The generation of the packing plans for the customer segments in phase A has *no* time limit. The time limits for the phases *B* to *D* and the parameters used for customer combinations in phase D were selected based on the size of the instances. Table 3 shows different sets of parameters that are tailored to instances of different size and were found by preliminary experiments (see Section 4.6). Each instance is calculated

Table 4
Results for the instance group GI–I1.

Instance	Moura and Oliveira		P1R2, 3 phases		P1R2, 4 phases	
	nv	*ttd*	*nv*	*ttd*	*nv*	*ttd*
GI–I1-01	9	762.59	8	625.03	8	625.03
GI–I1-02	8	675.24	7	575.09	7	575.09
GI–I1-03	6	1250.86	5	501.73	5	501.73
GI–I1-04	6	605.72	4	557.50	4	557.50
GI–I1-05	9	1398.47	6	550.83	6	550.83
GI–I1-06	7	757.08	5	549.13	5	549.13
GI–I1-07	6	1108.67	5	474.80	5	474.80
GI–I1-08	5	397.19	4	602.70	4	576.52
GI–I1-09	6	1050.70	5	517.29	5	517.29
GI–I1-10	6	578.36	5	505.46	5	505.46
GI–I1-11	6	1128.55	5	500.33	5	500.33
GI–I1-12	5	980.97	4	557.90	4	557.90

Table 5
Results for the instance group GI–I2.

Instance	Moura and Oliveira		P1R2, 3 phases		P1R2, 4 phases	
	n	*ttd*	*n*	*ttd*	*nv*	*ttd*
GI–I2-01	5	2668.55	4	781.34	4	765.83
GI–I2-02	5	2555.26	4	663.85	4	561.11
GI–I2-03	5	2526.11	4	508.65	4	508.65
GI–I2-04	5	1953.67	4	477.45	4	471.17
GI–I2-05	5	635.96	4	558.47	4	529.95
GI–I2-06	5	2394.25	4	565.28	4	542.60
GI–I2-07	5	2187.27	4	475.30	4	470.82
GI–I2-08	4	472.35	4	475.49	4	465.13
GI–I2-09	5	674.01	4	479.32	4	442.30
GI–I2-10	5	753.04	4	578.13	4	578.13
GI–I2-11	5	2049.39	4	499.67	4	477.16

Table 6
Results for the instance group GII-I1.

Instance	Moura and Oliveira		P1R2, 3 phases		P1R2, 4 phases	
	nv	*ttd*	*nv*	*ttd*	*nv*	*ttd*
GII–I1-01	9	823.04	8	654.62	8	654.62
GII–I1-02	9	1622.59	7	592.14	7	592.14
GII–I1-03	7	1451.39	6	548.49	6	548.49
GII–I1-04	7	1221.44	6	546.24	6	540.43
GII–I1-05	10	1532.44	6	693.22	6	693.22
GII–I1-06	8	1576.10	6	627.53	6	627.53
GII–I1-07	7	1378.36	6	622.50	6	622.50
GII–I1-08	7	1187.52	6	553.05	6	549.18
GII–I1-09	6	625.91	6	709.61	6	709.61
GII–I1-10	7	1235.62	6	624.24	6	570.38
GII–I1-11	7	1293.95	6	540.03	6	540.03
GII–I1-12	7	1069.11	6	540.82	6	539.84

once only with a uniform seed value for the random number generation.

4.4. Numerical results for the instances from Moura and Oliveira

The following Tables (4–7) show on the one hand the best results for the four instance groups GI–I1, GI–I2, GII–I1 and GII–I2 calculated by the Moura and Oliveira [46] methods, and on the other hand the results of using P1R2 for the three-phase search (without customer combinations) and the results using the four-phase search (with customer combinations). The number of vehicles (*nv*) (i.e. the number of routes) and the total travel distance (*ttd*) are given per method and instance.

Table 7
Results for the instance group GII–I2.

Instance	Moura and Oliveira		P1R2, 3 phases		P1R2, 4 phases	
	nv	*ttd*	*nv*	*ttd*	*nv*	*ttd*
GII–I2-01	7	3740.55	6	591.35	6	591.35
GII–I2-02	7	3496.39	6	560.88	6	560.88
GII–I2-03	7	3134.62	6	536.05	6	525.99
GII–I2-04	6	3814.29	6	516.35	6	516.35
GII–I2-05	7	627.66	6	657.73	6	649.80
GII–I2-06	7	3115.18	6	562.60	6	560.64
GII–I2-07	7	2740.03	6	518.11	6	506.16
GII–I2-08	7	2212.02	6	537.62	6	534.97
GII–I2-09	7	2962.35	6	546.62	6	544.63
GII–I2-10	7	3512.25	6	599.93	6	599.93
GII–I2-11	6	2631.39	6	502.14	6	501.26

Table 8
Overview of the results for the 46 instances from Moura and Oliveira.

Instance group	Moura and Oliveira		P1R2, 3 phases		P1R2, 4 phases	
	nv	*ttd*	*nv*	*ttd*	*nv*	*ttd*
GI–I1	79	891.2	63	543.1	63	541.0
GI–I2	54	1715.4	44	551.2	44	528.4
GII–I1	91	1251.5	75	604.4	75	599.0
GII–I2	75	2907.9	66	557.2	66	553.8
Total	299	1664.5	248	564.4	248	556.2
Mean CPU time (s)	<60		38.2		136.1	

Table 8 summarises the results for all 46 instances. In each case, the *sum* of the vehicle numbers and the *averaged* total travel distance are given per instance group and for all 46 instances respectively. The mean computation times of P1R2 in 2.6 GHz seconds are notified for P1R2 over all 46 instances; in the case of the methods from Moura and Oliveira [46] the average computation times are less than one minute on a Pentium PC with a clock rate of only 440 MHz.

An analysis of the test results leads to the following statements:

- Even without using customer combinations the heuristic P1R2 yields considerably better results than the methods by Moura and Oliveira. The number of vehicles saved over all the 46 instances amounts to 51 vehicles; this corresponds to a saving of 17%. The mean total travel distance of the route plans generated by P1R2 amounts to approximately 34% of the mean total travel distance of the compared methods. The improvements are spread more or less evenly over all four instance groups.
- Even without explicit comparison it can be established that high-quality solutions are achieved with regard to the number of vehicles. If the simple lower bound $lb0 = \lceil$Volume of all boxes/ loading space volume$\rceil$ is used as the basis, the result is 230 as the minimum number of vehicles for all 46 instances; at least 4 vehicles are necessary for each of the instances in group GI and at least 6 for each of the instances in group GII. The actual number of vehicles (248) exceeds this value by 7.8% only and the optimum number of vehicles is achieved in 35 of 46 instances.
- The mean computation time is less than 40 s in the three-phase search. The time effort can be viewed as rather low since there are over 1000 boxes in the GI instances and over 1500 boxes in the GII instances that need to be packed. Of course, it is beneficial for the computation time that the lower bound *lb0*

of the number of vehicles is frequently reached, since in this case the first search phase is aborted prematurely.

- The use of customer combinations in the four-phase search with P1R2 leads only to a moderate increase in the solution quality. There is no increase in vehicle savings. The mean percentage of the total distance savings is 1.5% in comparison with the three-phase search. Distance savings are achieved in 21 instances and savings of over 5% are achieved for 4 instances. The mean computation time per instance amounts to 2.3 minutes if customer combinations are used.
- As stated in Section 3.3 the LIFO constraint (C2) is neglected within a packing plan that belongs to a customer combination. Hence, the LIFO constraint is generally violated (to a certain extent) by a solution of the four-phase search while it is accurately observed by a solution of the three-phase search. However, the additional loading effort remains moderate on average. If the 21 instances are considered for which customer combinations were found, the mean percentage of boxes that are to be reloaded amounts to 5.4% of all boxes. Therefore, the concept of customer combinations can be viewed as a useful supplement: with acceptable computational effort the human dispatcher receives an additional problem solution since the solution of the three-phase search is also provided. Then he or she can decide whether a possible distance saving justifies a certain additional loading effort. This effort depends in the first instance on the number of boxes in all customer combinations of a solution that are badly placed with regard to the LIFO constraint. Note that this information is provided by the heuristic. Moreover, the additional loading effort and the distance saving could be automatically compared using some suitable coefficients (weights) to ease the decision of the dispatcher.

4.5. Numerical results for the 120 new VRLP instances

Table 9 contains the results calculated by P1R2 for the 120 new instances averaged over the test cases and over all instances respectively (for test case data as instance size see Table 2). Again, the results are presented for a three-phase search (without customer combinations) and for a four-phase search (with customer combinations). The following variables are listed for both variants:

(1) *sum-nv* — the sum of all vehicles used,
(2) *dev-lb0* — the mean deviation from the lower bound *lb0* of the number of vehicles (see above),
(3) *fill-rate* — the mean percentage filling rate that results per instance as the quotient of the total box volume and the volume of the loading spaces of all vehicles used,
(4) *mean-ttd* — the mean total travel distance and
(5) *time* — the mean computation time in 2.6 GHz seconds. In addition, for the four-phase search *ttd-impr* is shown, i.e. the percentage of distance savings in comparison with the three-phase search.

Because neither the optimal values nor reference values from other methods are available, the evaluation of the results should be carried out very carefully. In contrast to the relatively small instances from Moura and Oliveira [46], the lower bound of the number of vehicles *lb0* is in this case no longer reached and missed on average over all instances by approximately 20%. However, the mean volume utilisation of the vehicles per instance almost always exceeds 80% and usually achieves values greater than 83% for the larger instances (starting from 200 customers or 10,000 boxes). The customer combinations lead to a small reduction of the distance. This is about 1.4% on average over all instances whereas a reduction of vehicles is generally not successful. The percentage of badly placed boxes that are to be reloaded amounts here to only 0.1% of all boxes on average. Again, only those instances are considered to calculate the mean reloading effort for which customer combinations were found. If customer combinations are used, the mean computation time is ca. 15 minutes and it remains under 20 minutes on average for the instances with 1000 customers. If customer combinations are not used, the mean computation time over all instances is less than 7 minutes.

4.6. Further numerical results and overall evaluation

In the following further numerical results are reported concerning the impact of the time limits and other parameters, the lateral box support and the 27 VRLP instances by Gendreau et al. [30].

To study the impact of the time limits of the phases these parameters were varied in different calculations of the 46 instances by Moura and Oliveira [46]. The results are shown in Table 10 that is built as follows. Column 1 indicates whether a calculation was done using customer combinations (CC) or not. Columns 2 and 3 include the time limits of a calculation. In columns 4 and 5 the global outcome for the 46 instances is shown, given by the total number of used vehicles and the mean value of the total travel distance. The mean computing time per instance (in seconds) is presented in column 6. The average distance saving by CC (as percentage) and the mean reloading effort (given by the percentage of badly placed boxes) are indicated in columns 7 and 8 while the mean lateral boxes support (as defined in Section 2.3) is given in column 9. The first two lines contain once again the results from Table 8 for the time limits defined in Table 3.

The main observation is here that additional time effort compared to the first calculations (with lowest time limits) does not pay off. No improvements were made in terms of the number of vehicles and at best small distance savings are obtained. Note that the anomaly between the results for the second and third

Table 9
Results for the 120 new VRLP instances.

Test case	Three-phase search					Four-phase search					
	sum-nv	*dev-lb0* (%)	*fill-rate* (%)	*mean-ttd*	*time* (s)	*sum-nv*	*dev-lb0* (%)	*fill-rate* (%)	*mean-ttd*	*ttd-impr* (%)	*time* (s)
1	303	23.2	77.6	1363.9	347	301	22.4	78.0	1364.1	0.0	830
2	564	17.5	82.9	5445.4	292	564	17.5	82.9	5368.8	1.4	787
3	1120	17.9	83.5	13647.0	350	1120	17.9	83.5	13318.8	2.4	897
4	1680	18.3	83.6	32587.1	416	1680	18.3	83.6	31991.5	1.8	902
5	2238	18.4	83.7	56664.3	494	2238	18.4	83.7	56036.7	1.1	987
6	2801	18.8	83.6	87396.5	603	2801	18.8	83.6	86286.3	1.3	1112
Total	8706	18.5	82.5	32850.7	417	8704	18.5	82.5	32394.4	1.4	919.1

Table 10
Results of different calculations of the 46 instances by Moura and Oliveira.

1 CC used	2 Time limit phase *B*, C (s)	3 Time limit phase *D* (s)	4 Total no of vehicles	5 Mean total travel distance	6 Mean computing time (s)	7 Mean distance saving by CC (%)	8 Mean reloading effort by CC (%)	9 Mean lateral boxes support (%)
no	30	–	248	564.4	38.2	–	–	86.3
yes	30	60	248	556.2	136.1	1.5	5.4	86.5
no	60	–	248	560.3	75.4	–	–	86.4
yes	60	120	248	549.2	236.5	2.0	5.4	86.6
no	90	–	248	560.5	112.6	–	–	86.4
yes	90	180	248	551.8	332.6	1.6	5.6	86.5

variant of time limits is in each case caused by a single instance. The feature of customer combinations yields a small distance improvement independently of the time limits used.

At this point we add some remarks on the parameters of the customer combination module (see Section 3.3 and Table 3). Their values were specified by means of numerical experiments of limited scope. The parameters affect the flexibility in the construction of CC and they have an effect on the computing time, too. Therefore, for smaller instances such parameter values are chosen that enable an extensive search for suitable CC while more restrictive values have to be selected for larger instances. For example: if *qUbad* were set to 65% also for large instances the whole time in phase D would be spent constructing CC and not enough time would be left for searching better routes. If, conversely, the parameter *qUbad* is set to 15% for the small instances by Moura and Oliveira the mean distance saving by CC falls to only 0.2% in a run with time limits of 30, 30 and 60 s for the phases B to D (see Table 10). Moreover, the maximum length of customer combinations *maxlCC* was set to 3 to obtain a good compromise between the needed flexibility in the construction of CC and the extra reloading effort caused by their use.

The methods by Moura and Oliveira [46] consider the lateral boxes support as an additional stability condition (cf. Section 2.3) that is not explicitly taken into account by P1R2. As Table 10 shows about 86% of the boxes have a sufficient lateral support. High quality values for the lateral boxes support of packing plans depend on the degree of heterogeneity of the given box set and should be (roughly spoken) greater than 80% for weakly heterogeneous box sets (cf., e.g., Eley [21], p. 400f). Hence, the achieved lateral boxes support values seem to be satisfactorily. Note that a direct comparison is not possible since Moura and Oliveira [46] did not publish lateral boxes support results.

As expected, method P1R2 is not an adequate procedure for the 27 instances by Gendreau et al. [30] and this was proved by a numerical experiment. First, a simple lower bound (*LBnv*) on the number of vehicles that is needed by heuristic P1R2 to serve all customers of a given instance is defined: For each box b a feasible spatial orientation is chosen so that the box dimension $mx(b)$ in x-direction becomes minimum (cf. Fig. 1). Now, for each customer c a lower bound $LBll(c)$ of his loading length is given by the maximum of all lengths $mx(b)$ of his boxes. Finally, the lower bound on the number of vehicles *LBnv* is determined according to $LBnv = \lceil \Sigma\ LBll(c)/L \rceil$.

The lower bound *LBnv* was calculated for each of the 27 instances and in each case *LBnv* did exceed the stipulated number (v) of vehicles. Hence, the heuristic P1R2 is not able to generate feasible solutions for any of the 27 instances. Moreover, the difference between the lower bound *LBnv* and v amounts to 43.5% of the stipulated number of vehicles on average. Of course, this result reveals a limitation of the proposed solution method.

Taking into account all calculated results the heuristic P1R2 proves to be an effective solution method for the three-dimensional VRLP with time windows if the set of rectangular goods to be transported is structured in a suitable manner. Most of the customers should have ordered sufficiently many goods with not too large dimensions so that the goods of one customer can be packed in a segment of the loading space avoiding greater losses. This situation applies often in practice if, e.g., unfinished goods are delivered from a vendor to different customers or if a parcel service has to serve multiple major customers. On the other hand, P1R2 will often not be successful if the shipment per customer is always small, containing only a handful of boxes.

Moreover, it can be stated that P1R2 achieves a good solution quality consuming only a relatively small amount of time even for large instances. This statement results in particular from a comparison with other solution procedures for the three-dimensional VRLP (see [25,30,56]). The quoted procedures need 30 minutes or more on average to calculate the instances by Gendreau et al. [30] with 100 customers and 200 boxes at maximum. P1R2 takes 7 minutes on average to calculate instances with up to 1000 customers and 50,000 boxes. (Note that processors of not too different strengths were used in all cases).

5. Summary

The paper continues the research into the three-dimensional Vehicle Routing and Loading Problem (VRLP) by Gendreau et al. [30], Moura and Oliveira [46] and other authors. A heuristic is proposed that carries out packing and routing in separate stages and is therefore referred to as P1R2, i.e. "packing first, routing second". A 3D-SPP method by Bortfeldt and Gehring [8], and the VRPTW method by Homberger and Gehring [33], are used to implement this planning. However, the P1R2 approach is generic and can also integrate other methods for the partial problems. The supplementary concept of customer combinations is intended to counter quality losses that can result from the weak individual arrangements for some customers.

In the numerical test using the 46 benchmark instances from Moura and Oliveira the heuristic P1R2 achieves considerably better results than the methods from Moura and Oliveira, even without customer combinations. With regard to vehicle numbers and total travel distance, savings of 17% and 66% are achieved respectively, with mean computation times of less than 40 (2.6 GHz) seconds. An extra distance saving is achieved by means of customer combinations, but at the cost of a longer computation time. In addition, 120 new large VRLP instances are introduced with up to 1000 customers to be visited and up to 50,000 boxes to be transported. In testing these instances an average of 82.5% volume utilisation of the vehicles used was achieved with a mean computation time of just 7 minutes. The new instances and detailed results for all calculated instances will be provided at www.fernuni-hagen.de/evis (see "Forschung", "Downloads").

All in all, the method P1R2 presents an efficient heuristic for the three-dimensional VRLP with time windows that is suitable for situations in which the boxes (i.e. rectangular packing pieces) of different customers can be packed separately in different loading space segments with a sufficiently high volume utilisation.

Acknowledgement

The authors would like to thank Michael Prange and Danny Candra for their valuable support in implementing and testing the VRLP solution method proposed in this paper.

References

[1] Allen SD, Burke EK, Kendall G. A hybrid placement strategy for the three-dimensional strip packing problem. European Journal of Operational Research 2010http://dx.doi.org/10.1016/j.ejor.2010.09.023 Available from.
[2] Alvarenga GB, Mateus GR, de Tomi G. A genetic and set partitioning two-phase approach for the vehicle routing problem with time windows. Computers and Operations Research 2007;34:1561–84.
[3] Baldacci R, Toth P, Vigo D. Recent Advances in Vehicle Routing Exact Algorithms. 4OR 2007;5:269–298.
[4] Baldacci R, Toth P, Vigo D. Exact algorithms for routing problems under vehicle capacity constraints. In: Bouyssou D, Martello S, Plastria F, editors. Operations research II, invited surveys from 4OR, 2006–2008. Annals of operations research, 175. Berlin: Springer; 2010. p. 213–45.
[5] Bent R, Van Hentenryck P. A two-stage hybrid local search for the vehicle routing problem with time windows. Transportation Science 2004;38: 515–30.
[6] Bischoff EE, Ratcliff MSW. Issues in the development of approaches to container loading. Omega 1995;23:377–90.
[7] Bischoff EE. Three dimensional packing of items with limited load bearing strength. European Journal of Operational Research 2004;168:952–66.
[8] Bortfeldt A, Gehring H. Two meta heuristics for strip packing problems. In: Despotis DK, Zopounidis C, editors. Proceedings of the Fifth International Conference of the Decision Sciences Institute. Athens; 1999:(2) p. 1153–1156.
[9] Bortfeldt A, Gehring H. A hybrid genetic algorithm for the container loading problem. European Journal of Operational Research 2001;131:143–61.
[10] Bortfeldt A, Mack D. A heuristic for the three-dimensional strip packing problem. European Journal of Operational Research 2007;187:1267–79.
[11] Bortfeldt A, Homberger J. Packing first, routing second—eine Heuristik für das vehicle routing and loading problem. In: Bortfeldt A, Homberger J, Kopfer H, Pankratz G, Strangmeier R, editors. Intelligent decision support—current challenges and approaches, Intelligente Entscheidungsunterstützung—Aktuelle Herausforderungen und Lösungsansätze. Wiesbaden (Germany): Festschrift for/für Hermann Gehring. Gabler Edition Wissenschaft; 2008. p. 91–113.
[12] Bortfeldt A. A hybrid algorithm for the capacitated vehicle routing problem with three-dimensional loading constraints. Computers and Operations Research 2011http://dx.doi.org/10.1016/j.cor.2011.11.008.
[13] Bouthillier A, Le Crainic TG. A cooperative parallel meta-heuristic for the vehicle routing problem with time windows. Computers and Operations Research 2005;32:1685–708.
[14] Brandão de Oliveira H, Vasconcelos G. A hybrid search method for the vehicle routing problem with time windows. Annals of Operations Research 2010;180:125–44.
[15] Bräysy O, Hasle G, Dullaert W. A multi-start local search algorithm for the vehicle routing problem with time windows. European Journal of Operational Research 2004;159:586–605.
[16] Cordeau J-F, Laporte G. Tabu search heuristics for the vehicle routing problem. In: Rego C, Alidaee B, editors. Metaheuristic optimization via memory and evolution: tabu search and scatter search. Boston: Kluwer; 2004. p. 145–63.
[17] Cordeau J-F, Gendreau M, Hertz A, Laporte G, Sormany J-S. New heuristics for the vehicle routing problem. In: Langevin A, Riopel D, editors. Logistics systems: design and optimization. New York: Springer; 2005. p. 279–97.
[18] Dangelmaier W, Brockmann K, Hamady MO, Holtkamp R, Langemann T. OOPUS-PSCM—Ein Werkzeug zum Produktions- und Supply-Chain-Management. In: Kopfer H, Bierwirth Chr, editors. Logistik Management, Intelligente I+K Technologien. Berlin: Springer; 1999. p. 245–56.
[19] Derigs U, Kaiser R. Applying the attribute based hill climber heuristic to the vehicle routing problem. European Journal of Operational Research 2007;177:719–32.
[20] Dyckhoff H, Scheithauer G, Terno J. Cutting and packing. In: Dell'Amico M, Maffioli F, Martello S, editors. Annotated bibliographies in combinatorial optimization. Chichester: John Wiley & Sons Ltd.; 1997. p. 393–412.
[21] Eley M. Solving container loading problems by block arrangements. European Journal of Operational Research 2002;141:393–409.
[22] Fanslau T, Bortfeldt A, Tree Search A. Algorithm for solving the container loading problem. INFORMS Journal on Computing 2010;22:222–35.
[23] Faroe O, Pisinger D, Zachariasen M. Guided local search for the three-dimensional bin packing problem. INFORMS Journal on Computing 2003;15: 272–81.
[24] Fuellerer G, Doerner KF, Hartl R, Iori M. Ant colony optimization for the two-dimensional loading vehicle routing problem. Computers and Operations Research 2009;36:655–73.
[25] Fuellerer G, Doerner KF, Hartl R, Iori M. Metaheuristics for vehicle routing problems with three-dimensional loading constraints. European Journal of Operational Research 2010;201:751–9.
[26] Fukasawa R, Longo H, Lysgaard J, Poggi de Aragão M, Reis M, Uchoa E, Werneck RF. Robust branch-and-cut-and-price for the capacitated vehicle routing problem. Mathematical Programming Series A 2006;106:491–511.
[27] Gehring H, Homberger J. A parallel hybrid evolutionary metaheuristic for the vehicle routing problem with time windows. In: Miettinen K, Mäkelä M, Toivanen J. editors. Proceedings of EUROGEN99—short course on evolutionary algorithms in engineering and computer science, Reports of the department of mathematical information technology, Series A. Collections, No. A 2/1999, University of Jyväskylä, 1999. p. 57–64.
[28] Gehring H, Bortfeldt A. A parallel genetic algorithm for solving the container loading problem. International Transactions in Operational Research 2002;9:497–511.
[29] Gendreau M, Iori M, Laporte G, Martello S. A tabu search approach to vehicle routing problems with two-dimensional loading constraints. Networks 2008;51:4–18.
[30] Gendreau M, Iori M, Laporte G, Martello S. A tabu search algorithm for a routing and container loading problem. Transportation Science 2006;40: 342–50.
[31] Gonçalves JF, Resende MGC. A parallel multi-population biased random-key genetic algorithm for a container loading problem. Computers and Operations Research 2012;39:179–90.
[32] Haessler RW. Cost minimisation of multiple-vehicle shipments. International Journal of Physical Distribution and Logistics Management 1991;21: 37–41.
[33] Homberger J, Gehring H. A two-phase hybrid metaheuristic for the vehicle routing problem with time windows. European Journal of Operational Research 2005;162:220–38.
[34] Iori M, Salazar Gonzalez JJ, Vigo D. An exact approach for the symmetric capacitated vehicle routing problem with two dimensional loading constraints. Transportation Science 2007;41:253–64.
[35] Iori M, Martello S. Routing problems with loading constraints. Top 2010;18: 4–27.
[36] Kytöjoki J, Nuortio T, Bräysi O, Gendreau M. An efficient varibale neighborhood search heuristic for very large scale vehicle routing problems. Computers and Operations Research 2007;34:2743–57.
[37] Laporte G. What you should know about the vehicle routing problem. Naval Research Logistics 2007;54:811–9.
[38] Li F, Golden B, Wasil E. Very large-scale vehicle routing: new test problems,algorithms, and results. Computers and Operations Research 2005;32: 1165–79.
[39] Mack D, Bortfeldt A, Gehring H. A parallel hybrid local search algorithm for the container loading problem. International Transactions in Operational Research 2004;11:511–33.
[40] Marinakis Y, Migdalas A, Pardalos P. A new bilevel formulation for the vehicle routing problem and a solution method using a genetic algorithm. Journal of Global Optimization 2007;38:555–80.
[41] Marinakis Y, Marinaki M, Dounias G. Honey Bee Mating Optimization algorithm for large scale vehicle routing problems. Natural Computing 2010;9:5–27.
[42] Martello S, Pisinger D, Vigo D. The three-dimensional bin packing problem. Operations Research 2000;48:256–67.
[43] Mester D, Bräysy O. Active guided evolution strategies for the large scale vehicle routing problem with time windows. Computers and Operations Research 2005;32:1593–614.
[44] Mester D, Bräysy O. Active-guided evolution strategies for large-scale capacitated vehicle routing problems. Computers and Operations Research 2007;34:2964–75.
[45] Moura A, Oliveira JF. A grasp approach to the container loading problem. IEEE Intelligent Systems—Special issue on Metaheuristic Methods in Transportation and Logistics 2005;20:50–7.
[46] Moura A, Oliveira JF. An integrated approach to vehicle routing and container loading problems. Operations Research Spectrum 2009;31:775–800.
[47] Nagata Y. Edge assembly crossover for the capacitated vehicle routing problem. In: Cotta C, van Hemert J, editors. Lecture notes in computer science, vol. 4446. Berlin Heidelberg: Springer; 2007. p. 142–53.
[48] Parreño F, Alvarez-Valdes R, Oliveira JF, Tamarit JM. A maximal-space algorithm for the container loading problem. INFORMS Journal on Computing 2007;20:412–22.
[49] Parreño F, Alvarez-Valdes R, Oliveira JF, Tamarit JM. Neighborhood structures for the container loading problem: a VNS implementation. Journal of Heuristics 2010;16:1–22.
[50] Pisinger D. Heuristics for the container loading problem. European Journal of Operational Research 2002;141:143–53.
[51] Pisinger D, Ropke S. A general heuristic for vehicle routing problems. Computers and Operations Research 2007;34:2403–35.
[52] Prins Chr. A simple and effective evolutionary algorithm for the vehicle routing problem. Computers and Operations Research 2004;31: 1985–2002.
[53] Rochat Y, Taillard ÉD. Probabilistic diversification and intensification in local search for vehicle routing. Journal of Heuristics 1995;1:147–67.

[54] Solomon MM. Algorithms for the vehicle routing and scheduling problem with time windows constraints. Operations Research 1987;35:254–65.
[55] Taillard ÉD. Parallel iterative search methods for vehicle routing problems. Networks 1993;23:661673.
[56] Tarantilis CD, Zachariadis EE, Kiranoudis CT. A hybrid metaheuristic algorithm for the integrated vehicle routing and three-dimensional container-loading problem. IEEE Transactions on Intelligent Transportation Systems 2008;10:255–71.
[57] Terno J, Scheithauer G, Sommerweiß U, Rieme J. An efficient approach for the multi-pallet loading problem. European Journal of Operational Research 2000;123:372–81.
[58] Toth P,Vigo D. The vehicle routing problem. SIAM monographs on discrete mathematics and applications. Philadelphia, PA; 2002.
[59] Toth P, Vigo D. The granular tabu search and its application to the Vehicle Routing Problem. INFORMS Journal on Computing 2003;15:333–46.
[60] Wäscher G, Haussner H, Schumann H. An improved typology of cutting and packing problems. European Journal of Operational Research 2007;183: 1109–30.
[61] Weng Y, Guo S, Zhu W, Lim A, Oon W-C. The 6 key elements to SCLP block building approaches. Proceedings of the International Conference on Educational and Information Technology (ICEIT) 2010;1:402–7.