An integrated approach to the Vehicle Routing and Container Loading Problems

Ana Moura[1,2], José Fernando Oliveira[3,4]

[1]ESTG - Escola Superior de Tecnologia e Gestão de Leiria, IPL, Portugal
[2]INESC-Coimbra – Instituto de Engenharia de Sistemas e Computadores, Coimbra, Portugal
[3]FEUP - Faculdade de Engenharia da Universidade do Porto, Portugal
[4]INESC-Porto – Instituto de Engenharia de Sistemas e Computadores, Porto, Portugal
Morro de Lena – Alto do Vieiro, 2411-901 Leiria, PORTUGAL
Tel: +351244820365 / Email: ana.moura@estg.ipleiria.pt, jfo@fe.up.pt

Abstract: Real-world distribution problems raise some practical considerations usually not considered in a realistic way in more theoretical studies. One of these considerations is related to the vehicle capacity, not only in terms of cubic meters or weight capacity but also in terms of the cargo physical arrangements. In a distribution scene, two combinatorial optimization problems, the Vehicle Routing Problem with Time Windows and the Container Loading Problem, are inherently related to each other. This work presents a framework to integrate these two problems using two different resolutions methods. The first one treats the problem in a sequential approach while the second uses a hierarchical approach. In order to test the quality and efficiency of the proposed approaches some test problems were created based on the well-known Solomon, Bischoff and Ratcliff test problems. The results of the integrated approaches are presented and compared with results of the Vehicle Routing Problem with Time Windows and the Container Loading Problem applied separately.

Keywords: Vehicle Routing Problem with Time Windows, Container Loading Problem, GRASP.

1. Introduction

Managing the distribution of goods is a vital operation for many companies. Companies realize that distribution has a major economic impact. However, clients' satisfaction depends mainly on meeting their demand as effective as possible. This is commonly described as providing a service to a client. Usually a service is a mixture of different distribution characteristics, for example: product availability, delivery time, delivery programming and good condition after delivery. One of the most important areas in serving clients is said to be the transportation of good. With a good transportation, items arrive on time, not damaged and in the desired quantities. Indeed, those are the three main client's demands to be achieved and for that, the integration of route planning and vehicle packing is essential.

When solving a Vehicle Routing Problem with Time Windows (VRPTW), the solutions ensure that items arrive at the client within the time window. To ensure they do not suffer any damage during transportation, a stable load is necessary. This demand is achieved by solving the Container Loading Problem (CLP). To assure that all the items of each demand are delivered to a client, one must solve the VRPTW and CLP in an integrated way. Indeed, the classical model of vehicle routing ensures that the totality of the clients' demand, assigned to that vehicle, does not exceed the vehicle capacity restrictions in terms of weight (or other scalar measure). However it is not certain that the cargo can be physically loaded and arranged inside the vehicle/container. So, a cargo, which in terms of weight can be packed in a vehicle, can exceed its volume capacity, or vice versa. To deal with this drawback we propose the resolution of an integrated problem, named Vehicle Routing with Time Windows and Loading Problem (VRTWLP).

The client items must be packed in the vehicle considering a LIFO (Last-in-First-Out) strategy. [10] and [19] present a variation of the traveling salesman problem with pickup and delivery in which loading and unloading operations are executed in a LIFO order. [10] presents three local search operators embedded within a variable neighborhood search metaheuristic. [19] introduces the double traveling salesman problem with multiple stacks and presents three metaheuristics approaches where repacking is not allowed. The items are packed in several rows in the container and each row is considered a LIFO stack. The container loading problem is a three-dimensional problem that establishes arrangements of items in a container. Usually, the CLP aims to maximize loading efficiency – that is, the container space usage. For instance, [15], [5], [6], [7], [13], [3] and [17] deal with the container loading problem considering specifically the efficiency of the loading arrangements.

The other problem discussed in this paper is the Vehicle Routing Problem with Time Windows. In VRPTW, clients have to be served within a period of time. In literature there are four goals that are usually considered: (i) to minimize the number of vehicles; (ii) to minimize the total travel distance; (ii) to minimize the total time; and (iv) to minimize the vehicles total waiting time at clients. Some approaches use one of these goals and others combine two (or more) of them. [12], [11], [8], [9], [18] and [1] are some examples of recently published work where original algorithms for the VRPTW are presented. Very few and quite recent papers approach the vehicle routing problem and the two-dimensional bin packing problem integration. [16] presents a special case of the symmetric capacitated vehicle routing problem and proposes an exact approach based on branching algorithms. [14] presents a tabu search heuristic to solve the routing problem with two-dimensional loading constraints.

In the second section of this paper, a framework for the integration of the vehicle routing problem with time windows and the three-dimensional loading problem is proposed. In the two following sections (Section 3 and Section 4) three sequential approaches and one hierarchical approach to the VRTWLP resolution are presented. A problem generator was developed in order to create test problems for these approaches. The test problems combine the characteristics of the CLP and VRPTW problems available in the literature. All the results are compared and presented in Section 5. In Section 6, some conclusions are drawn.

2. Vehicle Routing with Time Windows and Loading Problem Description

The capacity constraint of the vehicles in the vehicle routing problem are often improperly used when real-world applications are considered. The capacity constraint is not only related to admissible weight but also to the vehicle's volume dimensions. The routes designed for a given vehicle capacity, in terms of weight limits, can lose its admissibility due to incompatibility of cargo dimensions, and vice versa. To address loading issues in more detail in routing problems, one needs a richer model. Loading constraints may seriously affect the nature of the problem. The integration of routing and loading problem calls for tailored resolution procedure. This integration results in the VRTWLP – Vehicle Routing with Time Windows and Loading Problem.

Let us consider in this problem a set of clients defined by their geographical coordinates and a fleet of homogeneous vehicles. There is only one depot from where the vehicles start to visit the clients and return at the end of the delivery. Each client has a demand to be satisfied by a single vehicle and a time window that must be respected. All the clients' demand must be satisfied even if another vehicle has to be used. The vehicle loading order is in the inverse relation to the clients visit order (LIFO strategy). The demand of each client must be packed together inside of the vehicle in order to increase the efficiency of the unloading operations. Vehicle capacity, defined in terms of weight and width, height and length of the loading volume must not be exceeded. A demand is composed by a set of different box types. A type of box is characterized by his physical dimensions, weight and orientation constraint. This means that each type of box could have one, two or three possible orientation (the "This Side Up" constraint). Each client is defined by the geographical coordinates, time window, demand (type of boxes and related quantities per type), total weight of the demand and the service time.

In this paper, this particular problem is solved using two approaches. In the first approach (Section 3), two constraints previously mentioned could be relaxed. The first one is the VRPTW constraint that states that each customer must be served exactly by one vehicle. The second one is the LIFO rule for the cargo's position in the vehicle. In this case, the problem can be faced like a Split Delivery VRP and the demand of a client is not necessarily packed together inside the vehicle. The second approach (Section 4) the problem is solved regarding all problem constraints. The interdependency between the VRPTW and CLP is greater when the number of clients visited by each vehicle is small. This means that each client's demand fill-in an important portion of the container. Therefore, the inclusion or not of a client in a route has a major impact on the CLP, and may become unfeasible by this decision. On the other hand, good volume utilization may lead to long and unfeasible routes. When we have many clients per vehicle, the routing aspects dominate the loading aspects, as the choice of clients to visit influences much more the routes than the loading efficiency. Conversely, when one client completely fills a vehicle, the only problem is how to load the cargo and the CLP dominates the VRPTW (that may end up just in a one client per container). However, with a relatively small number of clients per vehicle, and a weakly heterogeneous cargo, the integrated resolution of the two problems becomes rather important for the final solution quality. The relevance of the integrated resolution of the VRPTW and CLP problems is also dependent on the density of the goods to transport. If the goods are very heavy and of small size, the usual weight constraint will be the active constraint and there is no need to consider the CLP. Consequently, the study of the VRTWLP is especially relevant when the number of clients is small and their demand represents a weakly heterogeneous cargo. Considering these assumptions, we consider the following constraints and goals to the problem:

- Clients and depot time windows. All clients must be visited within a certain period of time and the vehicle has a maximum travel time (time to visit all clients and return to the depot).

- Homogeneous fleet. Vehicle capacity in terms of weight, volume and dimensions (length, width and height). The vehicle's capacity must always be respected.

- Cargo's orientation. The client's demand is parallelepiped boxes that may have to satisfy orientation constraints: for example, the "This side up" sign.

- Client demands are heterogeneous and the total demand of each client in terms of volume and weight does not fill a vehicle. Every demand must be satisfied by a single vehicle and the time windows must be respected.

- The density of the cargo[1] is such that the maximum weight of the container is not a constraint for the problem.

- Cargo's positioning inside the vehicle. Each client's demand must be packed together in order to make unloading easier. Although not strictly necessary to ensure compatibility between the routes and the loading pattern. A LIFO policy will be used so that when a client is visited, it must be possible to unload all items of his demand through a sequence of movements (see [14]). The loading order is the reverse of the client's visits order.

- Cargo stability. To ensure that the load cannot move significantly during transport, the cargo must be packed in such a way that it remains stable. Also an unstable load can have important safety implications for loading and unloading operations. From a stability viewpoint, two different measurements are considered. The first one is the full support of each item from below. This measurement does not indicate the potential for lateral movement of a box, though. The second measurement is the average percentage of boxes not surrounded by at least three sides ([4] and [17]).

The VRTWLP main goal, as for the VRPTW, is to minimize the number of vehicles. From the CLP point of view, the objective function is to maximize the container's volume utilization. Considering multiple vehicles for the CLP, this objective function can be seen as packing all the available cargo in the vehicles. Thus, the minimization of the number of needed vehicles is also implicit. The minimization of the total travel time for each route (a typical VRPTW objective) is also considered as a subordinated objective for the VRTWLP. So, in short, the VRTWLP's basic idea is trying to serve with each vehicle the greatest possible number of clients and pack their demand in a feasible way while considering also the minimization of the total travel distance for each route.

Mathematical formulation

To make a definition with graphical notations let $G(N, A)$ be a direct graph, where $N\{0,1,...,n,n+1\}$ are the set of nodes and $A = \{(i, j) : i, j \in N, i \neq j\}$ the set of edges. The 0 and n+1 node represent the depot and

$N' = \{1,...,n\}$ is relative to the set of clients that must be visited. Every route must start and end in node 0 and n+1 respectively. Each arc (i, j) as an associated cost $c_{ij} > 0$. Each node has a demand $e(i)$ defined by his weight p_i and C is the set of all clients' demands. Each demand consists in a set of different boxes with specific dimensions. $l_\alpha, w_\alpha, h_\alpha$ are the length, width and height (respectively) of box α. Every client has a time window $[a_i, b_i]$ where a_i corresponds to the beginning and b_i to the end of the time window. It is assumed that $a_i \geq 0$ and $b_i > 0$. The depot has a time window $[a_0, b_0]$ and each vehicle must leave the depot at instant $a_0 = 0$ that corresponds to the beginning of the depot time window and arrived before instant $b_0 > 0$ that corresponds to the end of depot time window. V represents the set of vehicles belonging to the fleet. Each vehicle k is defined by his maximum capacity q_k and his physical dimensions length, width and height (L_k, W_k, H_k respectively). t_{ik} is the service time of vehicle k in client i and h_k is a cost associated to the vehicle.

Four types of decision variables were defined for the model:

1. x_{ijk} defined by $\forall i, j \in N', \forall k \in V$, with $i \neq j, i \neq n+1, j \neq 0$, that takes value 1 if the vehicle k travels from i to j, and 0 if not.

2. s_{ik} defined by $\forall i \in N', \forall k \in V$, which denotes the moment in which vehicle k begins serving client i. It is assumed that $s_{0k} = 0$ (when vehicle k leaves the depot) and s_{n+1k} denotes the moment in which the vehicle arrives to the depot.

3. $\gamma_{\alpha k}$ defined by $\forall \alpha \in C, \forall k \in V$, that takes value 1 if a box α is transported by the vehicle k and 0 if not.

4. $z'_{\alpha k}, z''_{\alpha k}, z'''_{\alpha k}$ defined by $\forall \alpha \in C, \forall k \in V$, which denotes box α placement coordinates in vehicle k.

The model of VRTWLP can be obtained has follows:

$$\min\left[f_1 \sum_{k \in V} \sum_{j \in N'} h_k x_{0jk} + f_2 \sum_{k \in V} \sum_{i,j \in N'} c_{ij} x_{ijk} + f_3 \sum_{j \in N'} \sum_{k \in V} \left(L_k \times W_k \times H_k - \sum_{\alpha \in C} l_\alpha \times w_\alpha \times h_\alpha \times \gamma_{\alpha k} \right) \times x_{0jk} \right] \quad \text{(Eq.1)}$$

$$\sum_{k \in V} \sum_{j \in N'} x_{ijk} = 1 \qquad \forall i \in N' \quad \text{(Eq.2)}$$

[1] Density of a cargo is a measure of mass per unit volume. For example, an object made from a comparatively dense material (such as iron) will have more mass than an equal-sized object made from some less dense substance (such as aluminum).

$$\sum_{i.j\in S} x_{ijk} \leq |S|-1 \qquad S\subset N', 2\leq |S| \leq n-2 \qquad \text{(Eq.3)}$$

$$\sum_{j\in N'} x_{0jk} = 1 \qquad \forall k\in V \qquad \text{(Eq.4)}$$

$$\sum_{i\in N'} x_{ilk} - \sum_{j\in N'} x_{ljk} = 0 \qquad \forall l\in N', \forall k\in V \qquad \text{(Eq.5)}$$

$$\sum_{i\in N'} x_{in+1k} = 1 \qquad \forall k\in V \qquad \text{(Eq.6)}$$

$$\sum_{i\in N'}\left[p_i \times \sum_{j\in N} x_{ijk}\right] \leq q_k \qquad \forall k\in V \qquad \text{(Eq.7)}$$

$$s_{ik} + t_{ik} - M(1-x_{ijk}) \leq s_{ik} \qquad \forall i,j\in N', \forall k\in V, M\to +\infty \qquad \text{(Eq.8)}$$

$$a_i \leq s_{ik} \leq b_i \qquad \forall i\in N', \forall k\in V \qquad \text{(Eq.9)}$$

$$\sum_{k\in V} \gamma_{\alpha k} = 1 \qquad \forall \alpha\in C \qquad \text{(Eq.10)}$$

$$\gamma_{\alpha k} \geq \sum_{j\in N'} x_{ijk} \qquad \forall i\in N', \forall k\in V, \forall \alpha\in e(i) \qquad \text{(Eq.11)}$$

$$z'_{\alpha k} \geq (\gamma_{\alpha k}-1)M \qquad \forall \alpha\in C, \forall k\in V \qquad \text{(Eq.12)}$$

$$z'_{\alpha k} - L_k + l_\alpha \leq (1-\gamma_{\alpha k})M \qquad \forall \alpha\in C, \forall k\in V \qquad \text{(Eq.13)}$$

$$z''_{\alpha k} \geq (\gamma_{\alpha k}-1)M \qquad \forall \alpha\in C, \forall k\in V \qquad \text{(Eq.14)}$$

$$z''_{\alpha k} - W_k + w_\alpha \leq (1-\gamma_{\alpha k})M \qquad \forall \alpha\in C, \forall k\in V \qquad \text{(Eq.15)}$$

$$z'''_{\alpha k} \geq (\gamma_{\alpha k}-1)M \qquad \forall \alpha\in C, \forall k\in V \qquad \text{(Eq.16)}$$

$$z'''_{\alpha k} - H_k + h_\alpha \leq (1-\gamma_{\alpha k})M \qquad \forall \alpha\in C, \forall k\in V \qquad \text{(Eq. 17)}$$

$$-z'_{\alpha' k} + z'_{\alpha'' k} + l_{\alpha''} \leq \delta_{1\alpha'\alpha'' k}M \qquad \forall \alpha', \alpha''\in C, \forall k\in V \qquad \text{(Eq.18)}$$

$$z'_{\alpha' k} - z'_{\alpha'' k} + l_{\alpha'} \leq \delta_{2\alpha'\alpha'' k}M \qquad \forall \alpha', \alpha''\in C, \forall k\in V \qquad \text{(Eq.19)}$$

$$-z''_{\alpha'k} + z''_{\alpha''k} + w_{\alpha''} \leq \delta_{3\alpha'\alpha''k} M \qquad \forall \alpha', \alpha'' \in C, \forall k \in V \qquad \text{(Eq.20)}$$

$$z''_{\alpha'k} - z''_{\alpha''k} + w_{\alpha'} \leq \delta_{4\alpha'\alpha''k} M \qquad \forall \alpha', \alpha'' \in C, \forall k \in V \qquad \text{(Eq.21)}$$

$$-z'''_{\alpha'k} + z'''_{\alpha''k} + h_{\alpha''} \leq \delta_{5\alpha'\alpha''k} M \qquad \forall \alpha', \alpha'' \in C, \forall k \in V \qquad \text{(Eq.22)}$$

$$z'''_{\alpha'k} - z'''_{\alpha''k} + h_{\alpha'} \leq \delta_{6\alpha'\alpha''k} M \qquad \forall \alpha', \alpha'' \in C, \forall k \in V \qquad \text{(Eq.23)}$$

$$\delta_{1\alpha'\alpha''k} + \delta_{2\alpha'\alpha''k} + \delta_{3\alpha'\alpha''k} + \delta_{4\alpha'\alpha''k} + \delta_{5\alpha'\alpha''k} + \delta_{6\alpha'\alpha''k} \leq 5 + (1 - \gamma_{\alpha'k}) + (1 - \gamma_{\alpha''k}) \ \forall \alpha', \alpha'' \in C, \forall k \in V \qquad \text{(Eq.24)}$$

$$\delta_{n\alpha'\alpha''k} \in \{0,1\} \qquad \forall n \in \{1,..,6\}, \forall \alpha'\alpha'' \in C, \forall k \in V \qquad \text{(Eq.25)}$$

$$\gamma_{\alpha k} \in \{0,1\} \qquad \forall \alpha \in C, \forall k \in V \qquad \text{(Eq.26)}$$

$$x_{ijk} \in \{0,1\} \qquad \forall i, j \in N', \forall k \in V \qquad \text{(Eq.27)}$$

$$s_{ik} \geq 0 \qquad \forall i \in N', \forall k \in V \qquad \text{(Eq.28)}$$

$$z'_{\alpha k} \geq 0, z''_{\alpha k} \geq 0, z'''_{\alpha k} \geq 0 \qquad \forall \alpha \in C, \forall k \in V \qquad \text{(Eq.29)}$$

The objective function (Eq.1) considers the visit of all clients with a minimum number of vehicles and for each vehicle the wasted space and the total time of routes is minimized. Three arbitrary weights for each objective function component are assigned: f_1, f_2 and f_3, number of vehicles, time and space, respectively. Each one is established according to the practical application and the importance given by the decision-maker for each objective function components. Besides the vehicles and the packed cargo, must respect all the problem constraints.

Like in the VRPTW formulations, constraint (Eq.2) ensures that a client is assigned to only one vehicle and from each client the vehicle goes to another client or to the depot. Constraint (Eq.3) prevents cycles creation and (Eq.4) to (Eq.6) are the flow equations on the network. With (Eq.7) to (Eq.9) equations, the vehicles capacity and clients time windows are fulfilled. Constraint (Eq.8) ensures that the service does not begin until the vehicle arrives to the client. The remaining equations are related to the cargo loading inside the vehicle. Constraint (Eq.10) ensures that each box is transported by one and only one vehicle. Constraint (Eq.11) ensures that if a box belongs to a client, that box must be transported by the vehicle that serves that client. The mutual positioning of the loads in a vehicle is determined by equations (Eq.12) to (Eq.17). These equations ensure that each box is inside the vehicle and none of vehicle dimensions are exceeded. Related to depth width and height dimensions, the (Eq.12), (Eq.14) and (Eq.16) ensure that the box is inside the vehicle and (Eq.13), (Eq.15) and (Eq.17) that the box does not exceed the vehicle dimension, respectively. The "no intersection" between boxes inside the vehicle is determinate by equations (Eq.18) to (Eq.24). These conditions are valid only if the loads are in the same vehicle and each box could have three different positions related to all the other boxes (Eq.24). For the vehicle depth dimension (Eq.18) and (Eq.19), proves that a box α' has only two possible positions related to another box α'', beyond or in front of it. For width dimension (Eq.20) and (Eq.21), prove that a box α' could only be on the right side or left side of box α''. For height dimension (Eq.22) and (Eq.23), prove that a box α' could be under or on top of box α''. The remaining constraints of the model, (Eq. 25) to (Eq. 29) are related to variables domains.

3. The sequential method

In this approach the vehicle routes and the container loading are planned at the same time. The problem data is represented as a list named Sequential Candidate List (SCL). Each client defined by the geometric coordinates, time window, demand (type of boxes, related quantities and orientation per type), total weight of demand and service time. A candidate in SCL is composed by a client and a single box type of his demand. The number of candidates in the SCL depends on the different box types per client's demand (Fig. 1).

Candidate 1: Client 1; Box Type 1
Candidate 2: Client 1; Box Type 3
Candidate 3: Client 2; Box Type 2
Candidate 4: Client 2; Box Type 5

Candidate n: Client 25; Box Type 2

Fig. 1 - Sequential Candidate List

In this approach two of the problem constraints previously presented could be relaxed. The first one is related to the cargo's position in the vehicle and the second specifies that every client must be visited only by one vehicle. With the SCL, the routes are built using a simple constructive algorithm. A candidate of SCL is chosen, the related client is inserted in a route and the box type is loaded in a free space of the vehicle. If all the restrictions of VRPTW and CLP are satisfied, the solution is accepted and the algorithm selects another candidate from the SCL. When the vehicle is completed loaded, another vehicle must me chosen in order to serve all the SCL candidates. For example, the client 2 (Fig. 1) could be placed in 3^{rd}, 4^{th} and n-2 position in SCL if his demand is composed by three different box types. The candidate 3 and candidate 4 could be assigned to vehicle k and candidate n-2 to a different one. So, the same client is visited by two different vehicles. This means that constraint of Eq 2 (each client must be visited by a single vehicle) must be relaxed.

For this same reason, in this approach we will allow that each client's demand might be spread (although grouped by box type) in the vehicle. When this happens in a built solution, a client could have its demand not directly accessible and in this case one of the two following alternatives must be chosen:

1. the cargo that blocks the access to those boxes must be unloaded and reloaded at the client so that his/her complete demand is unloaded;

2. or, the client could be later revisited by the same vehicle.

The choice between these two alternatives is made by evaluating them in terms of total route time. If the first alternative is chosen the route must be rebuilt because the vehicle does not need to revisit the client that has the demand split up and an additional time for unloading and reloading the blocking cargo is considered. In each case the algorithm computes the cost and the alternative with smaller cost is chosen. Using this sequential method, three search approaches are presented and tested:

- Monte Carlo procedure;
- Local Search heuristic;
- GRASP heuristic.

These three approaches use the previously presented SCL (Sequential Candidate List) and the same constructive heuristic to generate routes. The well known Monte Carlo approach is basically an iterated generation of random solutions built from SCLs. The local search approach generates an initial solution with the same constructive heuristic, starting with a random solution built from the SCL. Then a local search phase, based on a 2-OPT neighbourhood to the SCL is applied. The GRASP heuristic builds upon the restricted candidate list (RCL) paradigm to build an initial solution. Following this paradigm, the next candidate to insert in the solution is randomly chosen with a certain probability from an elite list defined by ranking all the candidates not yet inserted in the solution. Five different ranking schemes are combined:

1. Smaller time windows: clients whose time window is smaller and starts earlier are inserted first;

2. Smaller distance: closer clients are inserted first;

3. Larger distance: the first client to insert is the farthest from the depot. The following ones are inserted according to the “smaller distance” strategy;

4. Smaller distance and smaller time window: firstly clients are ordered by distance. Then, following the list, the first client that has a time window opened is selected. If no time window is opened, the client with a closer time window start is selected;

5. Larger distance smaller time windows: similar to the previous ranking strategy with the exception of the first client being the farthest from the depot.

These five ranking schemes originate five different variants of the GRASP heuristic. To improve the initial solution, a local search procedure is applied ([20]) using a 2-OPT strategy.

The constructive heuristic, on which these three approaches build upon, is presented in Fig. 2. While the insertion of a given client in a route is a trivial operation, to pack its boxes in the vehicle, a 3D packing algorithm must be used. The approach described in [17] was followed. It is an improvement of the George and Robinson [15] heuristic for solving the Container Loading Problem. This wall-building constructive heuristic packs boxes in a container ensuring the cargo stability. All boxes are fully supported and all the columns of boxes have at least three sides supported. For each type of boxes the free space in the container is filled with the best possible arrangement. The arrangements are found by simulating all choices of possible orientations of the box types, and by computing the corresponding volume utilization. Then the best arrangement is chosen and packed in the vehicle's free space.

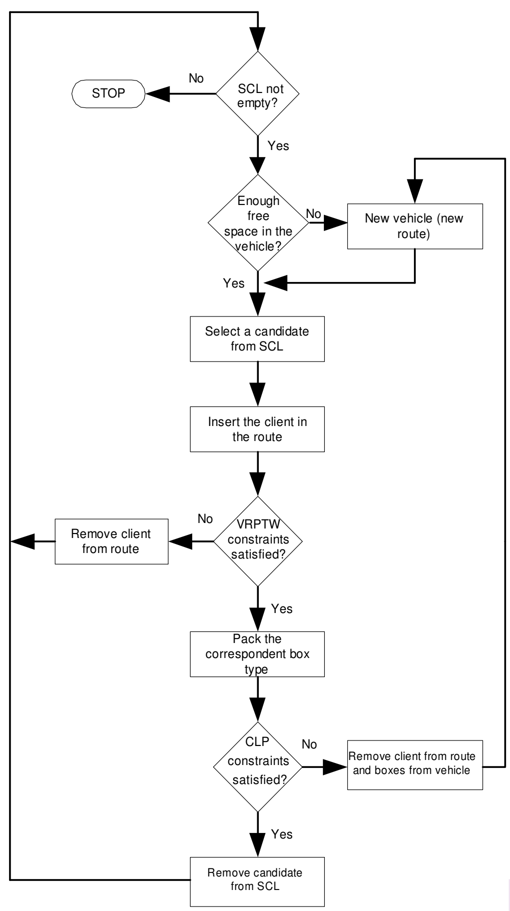

Fig. 2 - Constructive heuristic of the sequential approaches

In what concerns implementation issues, the Monte Carlo approach stops after 5000 iterations and the best solution is chosen. The Local Search and the GRASP heuristics use as stop criterion ten consecutive neighbourhoods without improvement. Each one of them runs 500 times with different initial solutions.

4. The Hierarchical method

The main difference of this approach to the previous one is that the CLP is now considered as a sub-problem of VRPTW. Therefore, in a first phase, the routes are built and only after that, for each route, the items are packed in the vehicles/containers. Moreover, in this method all the constraints (mentioned in Section 2) are considered. It is assumed that the two previous constraints, (i) the LIFO rule for the cargo's position in the vehicle and (ii) each client must be visited by a single vehicle, are not relaxed. In this hierarchical method, the problem's data is represented in a list named Client List (Fig. 3). The difference to the previously SCL list is that, every client of the Client List (CL) has assigned the totality of his demand (composed of different box types).

Client 1; Total Demand
Client 2; Total Demand
Client 3; Total Demand
Client 4; Total Demand
— — — — — — —
Client 25; Total Demand

Fig. 3 - Client List

The hierarchical method results from a composed approach divided in three different phases (Fig. 4): constructive; post-constructive; and local search.

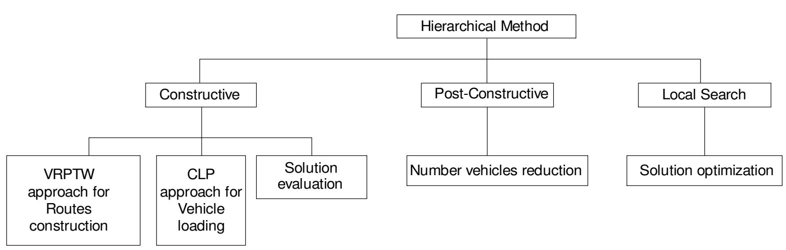

Fig. 4 - Block diagram of the hierarchical method

In the constructive phase, the routes are built using a GRASP algorithm [18] and ignoring the packing of the client's demand. The VRPTW objective function is the minimization of the number of vehicles and the minimization of total travel time. Considering this objective function in the routes construction phase of the algorithm, some feasible routes are built. Then, the algorithm tries to reduce the number of built routes in order to minimize the total number of routes. Finally, a local search is performed. This local search aims to improve the initial solution and tries to minimize the total travel time. Then, for every route, the boxes are packed using a greedy constructive heuristic (previously described in Section 3) developed for the CLP ([17]). The goal is to minimize the vehicle's wasted space. A LIFO policy is used to pack the client's demands (set of different boxes) in the vehicles. The algorithm never starts to pack a new client's demand until the previous one is totally packed. A packing is feasible if all the CLP restrictions are respected, like vehicle's volume capacity, boxes orientation, cargo stability and demand delivery order. At this point, a solution is composed of pairs of routes and the respective loaded vehicles (called pair route/pack). The solution (set of pairs) must be evaluated. For every pair it must be verified if all the cargo assigned to the route has been packed in the vehicle. A pair may have clients who can not be satisfied because the packing algorithm has stopped because the total volume of the demand exceeds the free volume of the vehicle. Those clients do not have the entire demand or some boxes of the demand in the vehicle. A set of procedures, presented in figures 5, 6, and 7, are used to check if this situation happens and to repair routes/packs binding feasibility.

In Fig. 5 the solution evaluation algorithm tests if all the clients of the constructed solution (A) are satisfied. Every pair is checked and kept if all demands of all the clients are packed in the vehicle's container. If not, there is an unsatisfied client (or clients) belonging to the pair. A client is unsatisfied if one of the two conditions holds:

- None of the requested items are packed;
- Some requested items are not packed.

In these situations, the algorithm checks if the total unpacked demand volume is less than, or equal to, the container free space volume. Two situations may occur:

1. If false, the algorithm removes the unsatisfied client from the pair; and tries to satisfy all clients using the "Unsatisfied client algorithm" (Fig. 6).

This algorithm tries to insert the removed client in any other solution pair. In case the insertion in a different route occurs, the related container loading is performed with the constructive CLP algorithm, creating a different pair. If this insertion can not be performed (because the route restrictions are not fulfilled), a new route is created in order to serve the unsatisfied client and the related container loading (with the GRASP VRPTW and CLP approaches, respectively). In each case and for all new pairs a solution evaluation is performed.

2. If true, the algorithm tries to reorder the demand packing in the container. This is done using the "Pack reordering algorithm" (Figure 7).

This algorithm deals only with the packed demand and stops when no more contiguous loading demand in the container can be exchanged, or when all the clients of the route are satisfied. First it tries to exchange two contiguously packed demands and checks if the container volume utilization is maximized, which corresponds to an improvement in the CLP objective function. If an improvement is achieved and if all clients in the route are satisfied, the algorithm tries to readjust the visiting order of the route. Doing that, if the one of the WRPTW constraints is violated, the route stays the same and a cost must be added to the VRPTW objective function. Because the loading order is changed and the cargo positioning in the vehicle constraint is relaxed. This cost corresponds to the unloading of the next client's demand. We assume that, if it is possible to pack all the demand in the same vehicle, but to do so the order of two consecutive demands must be exchanged, it will be better to use another vehicle in order to serve the unsatisfied client. If an improvement of the volume utilization is not achieved or if a client in the route is not satisfied, the previous loading order is restored and, if possible, more exchanges are performed. When no more exchanges between packed demands are possible, the unsatisfied client algorithm is applied.

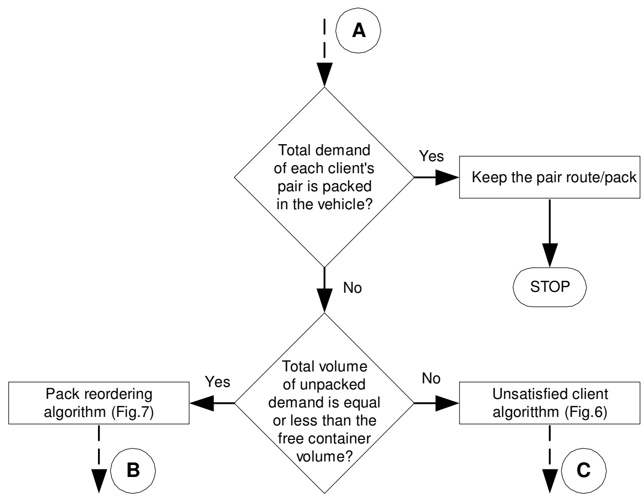

Fig. 5 – Solution evaluation algorithm

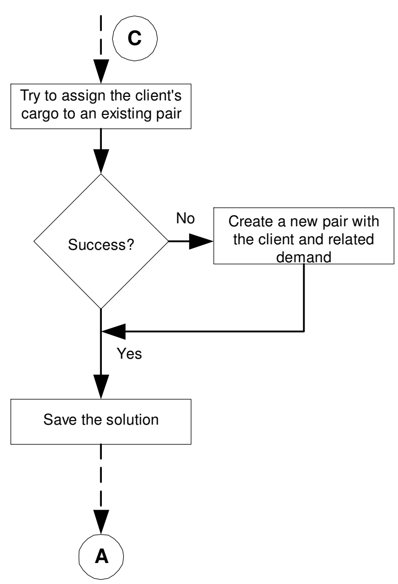

Fig. 6 - Unsatisfied client algorithm

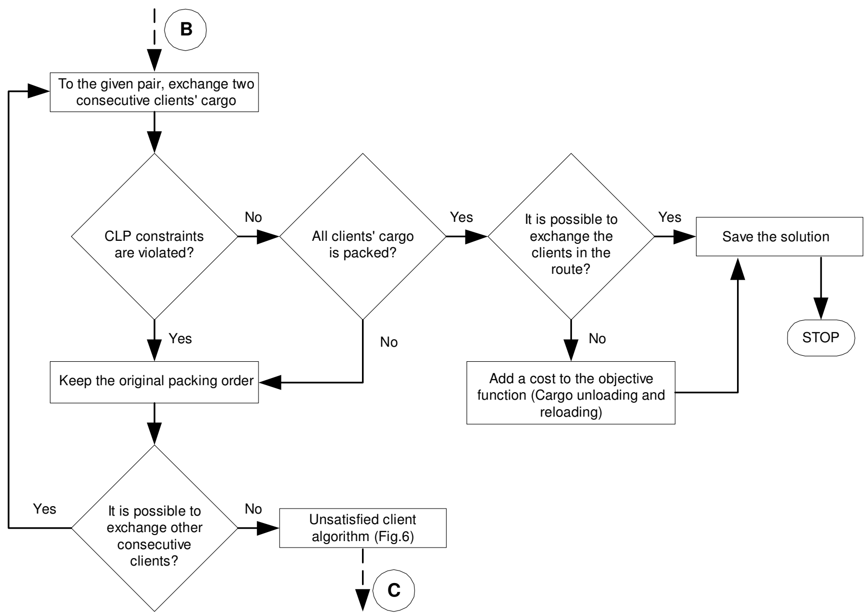

Fig. 7 - Pack reordering algorithm

In the end of the constructive phase, a solution with feasible pairs of routes and packing has been generated. The post-constructive phase is also described in detail in [18]. The algorithm looks for the pair (route/pack) that has the smallest container volume utilization and tries to insert all the clients of that pair in the other pairs whose vehicle still has free space. If all clients are inserted in the other routes without violating any restrictions, an improvement on the solution is achieved due to the decreasing number of vehicles.

The last phase is a local search procedure. The main objective is to perform a search in the neighborhood of the current solution in order to find a better one in terms of total travel time. If a better solution is found, it becomes the new current solution. A neighborhood is built in two different ways, each of them using the 2-Opt improvement procedure. These procedures are applied sequentially, first to each pair (intra-route exchanges), then among all pairs (inter-route exchanges). A better solution is accepted using a “first-improve” acceptance strategy, i.e., when a better solution is found it becomes the new current solution.

An improvement in the intra-route exchanges means that the new route total travel time has to be smaller than the original route total travel time. A new packing must be made, because the order delivery has changed. To improve the inter-route exchange procedure, the sum of the total travel time of the two new routes has to be smaller than the sum of the total travel time of the two original routes. New loadings are defined for each route by solving two new packing problems and two new pairs are created.

5. Computational Tests

Because the integrated problem (VRTWLP) is new, this is a pioneering study. For that reason, there are no problem's instances available in literature to try out the quality and efficiency of the integrated approaches. Therefore, some problem tests were developed by the authors of this paper using and joining the standard test problems characteristics available in the literature. The VRPTW test problems used are the R1 and R2 (with 25 clients) from [21] and the BR2 from [4] for the CLP test problems. Additionally, a problem test generator that uses the characteristics of these two kinds of problems was developed. Two new problem classes were created: I1 (R1 and BR2) and I2 (R2 and BR2). For each problem class, two different groups were defined:

- Group I - Clients' demand varies from 30 to 80 boxes, with 1 to 5 box types per client and an average of 42 boxes per demand. The total number of boxes is 1050;

- Group II - Clients' demand varies from 50 to 100 boxes, with 1 to 5 box types per client and an average of 62 boxes per demand. The total number of boxes is 1550.

As a result, we have generated forty six test problem instances covering the four combinations of classes and groups (around 12 instances per combination). These problem instances are available at the ESICUP web site (http://www.apdio.pt/esicup). The next tables presents the results of the four approaches of the integrated problem and the results of the CLP and VRPTW approaches when the sub-problem tests (VRPTW and CLP) were solved separately. The first table (Table 1) presents the results achieved with the GRASP approach applied to CLP. The results show the number of vehicles and volume utilization average needed to pack all the boxes of Group I and Group II problems. The volume utilization is defined by percentage of the total volume of loaded boxes related to the actual container volume. In Group I the number of boxes associated to each demand is relatively small (when compared to Group II problems). Solving the Group I packing problem with the GRASP CLP approach leads to a solution with four vehicles (Table 1) needed to pack all the cargo. Solving Group II problems with the GRASP CLP approach leads to a number of six vehicles needed to pack all the cargo (Table 1).

Table 1 – GRASP-CLP results for Group I and Group II

	GRASP – CLP **Number of Vehicles / Volume Utilization**
Group I	4 / 93.75
Group II	6 / 84.00

The problem instances of Class I1 have a short planning horizon, i.e., depot has a small time window, which implies many vehicles per problem and routes with a small number of clients. The problem instances of Class I2 have a long planning horizon, i.e., large depot time window, which implies few vehicles per problem and routes with a big number of clients. Table 2 and Table 3, presents the five different GRASP heuristic variants related to the ranking schemes applied to VRPTW (Section 3). RS1 corresponds to Smaller Time Windows, RS2 to Larger Distance and Smaller Time Windows, RS3 to Smaller Distance and Smaller Time Windows, RS4 Larger Distance and RS5 Smaller Distance ranking schemes. The results are in terms of number of vehicles and total travel distance. The GRASP VRPTW approach applied to Class I1 problem instances, originates a number of vehicles needed to visit all clients that varies from 5 to 8 (Table 2), depending on the instance. Using this approach to solve the problem instances of Class I2, a number of 2 vehicles is obtained (Table 3).

Table 2 -VRPTW results for Class I1

Class I1	**GRASP – VRPTW approach Number of Vehicles/Total Distance**				
Instance	**RS 1**	**RS 2**	**RS 3**	**RS 4**	**RS 5**
1	8/752.61	8/725.30	**7/735.40**	11/781.20	13/907.40
2	7/734.58	**7/734.50**	8/777.70	8/666.38	14/965.20
3	7/748.97	6/728.79	8/810.45	**5/510.91**	10/803.88
4	6/733.82	6/690.25	7/787.42	**6/558.70**	8/694.48
5	7/758.49	7/762.10	7/758.49	**6/593.89**	11/772.20
6	**6/705.28**	**6/705.28**	6/794.04	9/681.13	11/860.02
7	6/796.36	6/743.06	7/821.47	**6/588.67**	8/677.07
8	6/799.33	5/695.28	7/735.45	**5/521.92**	7/646.03
9	6/753.57	7/878.59	6/753.85	**5/568.32**	8/703.02
10	6/704.13	6/757.23	6/704.13	**5/534.40**	7/653.07
11	6/722.20	6/722.20	6/686.38	**6/564.02**	9/758.48
12	5/634.30	6/784.30	5/646.66	**5/539.75**	5/553.49

Table 3 - VRPTW results for Class I2

Class I2	**GRASP – VRPTW approach Number of Vehicles/Total Distance**				
Instance	**RS 1**	**RS 2**	**RS 3**	**RS 4**	**RS 5**
1	2/878.40	2/747.90	**2/733.90**	5/618.20	7/737.50
2	**2/653.30**	2/798.10	3/766.90	6/597.10	7/711.90
3	**2/876.74**	3/784.34	3/831.74	4/512.28	4/560.07
4	**2/913.01**	3/722.13	3/718.26	4/544.75	5/576.40
5	2/787.16	**2/749,74**	2/753.98	3/529.05	4/511.31
6	2/857.37	**2/717.35**	2/795.58	4/542.89	4/560.27
7	2/858.82	**2/730.16**	2/796.63	3/554.47	4/867.35
8	2/830.25	**2/666.10**	3/697.96	4/544.51	3/508.00
9	2/858.64	**2/741.18**	2/781.92	3/564.33	3/518.97
10	2/836.40	**2/737.46**	2/746.56	4/605.39	4/599.48
11	2/776.10	2/754.94	2/759.78	**2/463.86**	2/475.32

Legend: **Bold** – Best result

Combining the VRPTW and CLP problems, two classes and four groups of problem instances are obtained. The results of each Group/Class problem instances are presented in the next tables. The Table 4 and Table 5 include the results achieved with the Sequential and Hierarchical methods for Group I-Class I1 integrated problem (VRPL) respectively. When using the integrated approaches, that address both problems simultaneously, the number of vehicles for the best solutions varies from 5 to 9. Tables 6 and 7 present the results for Group I-Class I2 problem instances. By combining the two problems, a result of 4 to 5 vehicles for the best solutions is obtained. Tables 8 and 9 present the results for problem instances Group II-Class I1. The results vary from 6 to 10 vehicles for the best solutions. Table 10 and Table 11 present the results for the Group II-Class I2 integrated problem. The number of routes (vehicles) obtained with the integrated approaches varies from 6 to 7 for the best solutions.

Table 4 - VRTWLP Sequential methods for Group I Class I1

GI/I1	Monte Carlo Procedure	Local Search Heuristic	GRASP – VRTWLP Heuristic Number of Vehicles/Total Distance				
Instance	**Number of Vehicles/Total Distance (Iterations Number)**	**Number of Vehicles/Total Distance**	**RS 1**	**RS 2**	**RS 3**	**RS 4**	**RS 5**
1	11/1753.19(3727)	10/1672.92	10/1711.34	10/1626.27	10/1721.18	10/1654.67	10/1643.11
2	9/1611.22(1119)	9/1612.07	9/1617.70	9/1539.91	9/1563.51	8/1431.97	9/1589.75
3	8/1510.29(390)	7/1417.82	**6/1250.86**	7/1340.16	7/1302.05	7/1304.58	7/1287.11
4	7/1283.18(248)	6/1234.09	6/1212.51	6/1153.22	6/1184.20	6/1064.55	6/1230.86
5	9/1466.37(2832)	9/1557.28	9/1562.02	9/1432.31	9/1448.57	9/1527.15	**9/1398.47**
6	8/1484.77(3717)	7/1383.99	7/1273.26	7/1337.99	7/1301.29	7/1233.27	7/1262.35
7	7/1348.90(3137)	7/1299.04	6/1227.40	6/1177.89	7/1195.26	**6/1108.67**	6/1145.11
8	6/1121.81(2927)	6/1182.09	6/1158.76	6/1043.36	6/1138.08	6/1089.29	6/1104.82
9	7/1243.00(3918)	8/1288.45	7/1270.07	**6/1050.70**	7/1306.71	7/1118.01	7/1262.49
10	7/1231.02(1820)	7/1176.75	6/1098.71	6/1042.40	6/1069.83	6/1129.22	7/1128.64
11	7/1272.50(506)	6/1217.32	7/1198.02	7/1237.03	7/1269.96	**6/1128.55**	6/1170.30
12	6/1049.39(35)	6/1044.79	6/1070.44	6/1006.57	**5/980.97**	6/950.13	6/ 966.18

Table 5 - VRTWLP Hierarchical method for Group I Class I1

GI/I1	Hierarchical Method Number of Vehicles/Total Distance				
Instance	**RS 1**	**RS 2**	**RS 3**	**RS 4**	**RS 5**
1	**9/762.59**	11/743.32	12/799.74	11/790.32	13/852.71
2	11/944.44	**8/675.24**	13/892.88	11/895.69	14/965.25
3	8/754.27	8/673.38	10/803.88	9/720.48	11/789.20
4	6/804.14	8/671.68	8/667.21	**6/605.72**	8/694.48
5	10/815.36	10/742.56	11/786.02	11/795.61	11/775.39
6	**7/757.08**	8/679.49	11/860.02	9/705.27	10/799.67
7	7/901.80	8/679.91	8/677.07	9/705.05	8/677.07
8	6/785.95	7/696.04	8/649.04	**5/397.19**	7/646.03
9	7/820.17	7/673.77	10/827.83	7/700.96	8/621.48
10	7/753.01	8/697.67	9/711.18	**6/578.36**	8/706.20
11	7/851.45	8/691.08	9/758.48	7/627.67	10/747.09
12	6/803.62	7/698.98	12/786.74	6/698.39	7/631.49

Table 6 - VRTWLP Sequential methods for Group I Class I2

GI/I2	Monte Carlo Procedure	Local Search Heuristic	GRASP – VRTWLP Heuristic Number of Vehicles/Total Distance				
Instance	**Number of Vehicles/Total Distance (Iterations Number)**	**Number of Vehicles/Total Distance**	**RS 1**	**RS 2**	**RS 3**	**RS 4**	**RS 5**
1	7/3350.03(34)	6/3460.96	**5/2668.55**	6/3458.88	6/3180.87	6/3242.23	6/3052.05
2	6/3732.38(13)	6/3745.91	**5/2555.26**	6/3558.75	5/3126.48	6/3388.30	6/3248.94
3	5/3498.78(382)	6/3802.06	**5/2526.11**	5/3324.22	5/3288.81	5/2928.99	5/2875.46
4	5/2375.94(2036)	5/2525.06	**5/1953.67**	5/2729.68	5/2074.53	5/2364.58	5/2220.89
5	6/2868.47(2)	6/2874.70	5/2647.03	6/2823.35	6/2615.71	5/2443.93	5/2568.01
6	6/3156.67(30)	5/3286.00	**5/2394.25**	5/2831.04	6/3134.16	5/2907.20	5/2925.39
7	5/2902.58(186)	5/2734.85	**5/2187.27**	5/2773.78	5/3111.19	5/2563.57	5/2887.24
8	5/2072.25(92)	5/2242.90	5/1804.70	5/1991.56	5/2013.08	5/2113.14	5/1868.44
9	5/2316.86(32)	5/2364.72	5/2351.13	5/2486.56	5/2454.71	5/2350.55	5/2440.93
10	5/3056.37(39)	5/2840.82	5/3063.39	5/2948.08	5/2680.64	6/3334.28	5/2787.85
11	5/2232.80(2568)	5/2240.92	5/2076.89	5/2180.60	5/2385.62	5/2120.36	**5/2049.39**

Table 7 - VRTWLP Hierarchical method for Group I Class I2

GI/I2	Hierarchical Method Number of Vehicles/Total Distance				
Instance	RS 1	RS 2	RS 3	RS 4	RS 5
1	14/1105.19	7/692.16	8/842.73	7/935.75	8/842.03
2	12/987.27	9/813.95	9/835.58	7/644.90	8/844.09
3	11/1038.27	10/859.35	9/742.73	6/646.22	8/726.39
4	10/1091.51	15/958.82	14/853.43	6/660.74	15/1022.59
5	17/1129.44	8/864.70	**5/635.96**	5/640.96	**5/635.96**
6	14/1128.91	10/779.28	9/677.32	7/662.85	9/864.10
7	11/997.21	11/1144.90	10/935.50	7/778.00	8/817.26
8	9/903.80	10/739.33	9/741.28	**4/472.35**	9/682.86
9	13/1073.35	10/804.56	9/764.54	**5/674.01**	7/640.55
10	13/1058.32	8/666.28	7/713.85	**5/753.04**	6/749.10
11	13/1005.83	17/1260.46	9/792.93	8/714.51	6/536.68

Table 8 - VRTWLP Sequential methods for Group II Class I1

GII/I1	Monte Carlo Procedure	Local Search Heuristic	GRASP – VRTWLP Heuristic Number of Vehicles/Total Distance				
Instance	Number of Vehicles/Total Distance (Iterations Number)	Number of Vehicles/Total Distance	RS 1	RS 2	RS 3	RS 4	RS 5
1	11/1803.41(3429)	11/1826.68	13/2089.69	10/1675.65	11/1822.68	12/1800.76	12/1853.53
2	11/1903.93(1)	10/1860.83	**9/1622.59**	9/1624.37	9/1668.11	10/1740.39	10/1700.47
3	8/1556.54(2020)	8/1626.45	8/1380.53	8/1481.73	8/1478.13	8/1463.66	**7/1451.39**
4	8/1427.30(841)	8/1433.91	8/1405.80	8/1385.51	8/1264.79	**7/1221.44**	7/1252.98
5	10/1617.84(1179)	10/1604.76	10/1675.86	**10/1532.44**	11/1619.15	11/1587.78	10/1648.78
6	9/1605.89(2524)	**8/1576.10**	8/1476.54	9/1525.28	9/1512.84	9/1489.78	8/1470.08
7	8/1586.18(84)	8/1379.38	8/1381.41	8/1400.20	7/1398.36	8/1348.86	**7/1378.36**
8	8/1320.58(349)	7/1331.92	7/1303.89	8/1283.48	7/1243.19	**7/1187.52**	7/1195.51
9	8/1271.75(589)	9/1382.43	8/1359.87	8/1251.53	8/1366.81	8/1228.80	8/1246.38
10	8/1319.04(4958)	8/1282.37	8/1298.79	8/1214.00	8/1126.86	8/1200.74	**7/1235.62**
11	8/1312.92(2495)	8/1470.89	8/1502.61	8/1388.52	8/1358.64	8/1298.00	**7/1293.95**
12	7/1379.83(521)	7/1205.18	8/1377.93	7/1202.61	7/1086.82	**7/1069.11**	7/1123.28

Table 9 - VRTWLP Hierarchical method for Group II Class I1

GII/I1	Hierarchical Method Number of Vehicles/Total Distance				
Instance	RS 1	RS 2	RS 3	RS 4	RS 5
1	**9/823.04**	12/871.39	14/972.74	14/920.31	15/949.24
2	11/927.17	11/897.44	14/905.00	11/771.40	14/1017.89
3	10/970.26	9/706.74	10/856.51	9/702.87	10/856.51
4	8/844.66	10/741.33	10/761.16	8/524.96	10/675.25
5	12/864.48	12/838.32	13/889.43	13/861.41	13/829.06
6	14/1109.74	10/697.66	12/889.18	11/805.78	12/889.18
7	9/944.26	10/754.90	10/838.35	12/847.45	10/838.35
8	9/103537	9/699.52	10/717.10	8/593.30	10/717.10
9	15/1202.91	**6/625.91**	10/855.61	8/664.33	7/666.07
10	8/673.16	9/751.07	9/700.00	10/719.90	9/700.00
11	10/1023.30	8/605.31	10/885.02	8/550.00	10/885.02
12	8/844.40	9/763.45	13/879.69	9/691.47	9/660.59

Table 10 - VRTWLP Sequential methods for Group II Class I2

GII/I2	Monte Carlo Procedure	Local Search Heuristic	GRASP – VRTWLP Heuristic Number of Vehicles/Total Distance				
Instance	Number of Vehicles/Total Distance (Iterations Number)	Number of Vehicles/Total Distance	RS 1	RS 2	RS 3	RS 4	RS 5
1	8/4087.29(53)	8/3915.36	**7/3740.55**	8/4290.39	8/4181.74	8/4174.41	8/3895.85
2	7/4302.54(5)	7/4033.24	**7/3496.39**	7/4092.10	7/4211.86	7/4088.33	7/4381.37
3	8/4125.71(2139)	7/4038.13	**7/3134.62**	7/4015.78	7/4093.92	8/4307.52	7/4466.30
4	7/3326.76(216)	7/3260.77	**6/3814.29**	7/3019.28	7/2556.46	7/3048.27	7/2536.08
5	7/3400.67(1201)	7/3400.67	7/3180.35	8/3850.73	7/3511.13	7/3468.94	7/3390.09
6	7/3621.70(605)	7/3479.52	**7/3115.18**	7/3539.48	7/3820.80	7/3598.88	7/3733.76
7	7/3666.25(434)	7/3378.71	**7/2740.03**	7/3211.93	7/3087.93	7/3206.84	7/3788.90
8	7/3113.36(34)	7/2822.71	7/2330.75	7/2455.03	7/2238.34	7/2606.30	**7/2212.02**
9	7/3094.06(3088)	7/3108.10	7/3076.78	7/3055.81	**7/2962.35**	7/3083.32	7/3158.39
10	7/4309.37(2275)	7/4149.98	7/4081.19	7/3964.58	7/3663.32	7/3772.83	**7/3512.25**
11	7/2858.05(90)	7/2951.03	**6/2631.39**	7/2860.46	7/2900.74	7/3062.82	7/2837.53

Table 11 - VRTWLP Hierarchical method for Group II Class I2

GII/I2	Hierarchical Method Number of Vehicles/Total Distance				
Instance	RS 1	RS 2	RS 3	RS 4	RS 5
1	17/1168.68	11/1025.60	12/896.02	8/710.38	11/924.79
2	15/1069.66	13/1008.91	14/987.68	10/813.03	17/1091.37
3	10/923.09	20/1297.23	14/992.93	10/853.62	14/992.93
4	12/1051.02	13/912.34	18/1150.27	11/764.30	16/1135.64
5	19/1204.52	18/1315.51	11/884.94	**7/627.66**	9/720.54
6	16/1133.04	17/1198.08	19/1045.56	12/831.65	18/1023.20
7	14/1070.33	12/918.35	11/859.87	13/953.88	11/876.48
8	11/953.54	14/862.22	20/1198.62	11/703.48	20/1202.02
9	18/1189.13	18/1247.33	19/1148.51	8/780.93	15/989.20
10	16/1108.39	13/982.52	15/981.32	10/704.87	12/960.55
11	15/1055.11	20/1383.01	19/1327.72	8/661.24	12/782.98

Comparing the results of Class I and Class II for integrated approaches and VRPTW approach, in terms of number of vehicles, the solutions are worsened when we solve the problems in an integrated way. Those results are expected because the volume utilization must be considered in the routes building. In some instances the number of vehicles obtained with the integrated approaches is the same than with the VRPTW approach. This means that the optimum solution could be achieved with the integrated approaches.

In the Sequential methods the GRASP-VRTWLP heuristic clearly outperforms de Monte Carlo and Local search heuristics. Depending of the Class and Group of problem instances, the performances of Sequential heuristic and Hierarchical method have different behaviours. For Class I1 and Class I2 (short and long planning horizon problems respectively) and Group I smaller number of boxes per demand, the performance of the two integrated approaches are similar (Table 4, Table 5, Table 6 and Table 7). Increasing the number of boxes (Group II) and for the two different Classes (Table 8, Table 9, Table 10 and Table 11) the GRASP-VRTWLP approach outperforms the Hierarchical method.

Independently of the problem class or group, the Hierarchical approach always achieved better solutions when total travel time is under consideration.

6. Conclusions

The problem considered in this paper deals with the classical vehicle routing problem with time windows and the container loading problem. Some approaches for each independent problem was developed and tested with problem instances available in the literature. The problem features of the two classical problems were considered in an integrated way and two different approaches were developed, implemented and tested. Some integrated problems based in the well known problems instances for VRPTW and CLP were developed. The sequential approaches solve the two most important operational problems in distribution processes simultaneously, i.e., the VRPTW and the CLP. In the hierarchical approach, the VRPTW is the main problem and the CLP is the subsidiary problem, while in the sequential approaches the two problems are dealt at the same level. The obtained results with the integrated approaches show that the impact of the container loading problem constraints increases the solution costs. There are two important remarks related to the performance of the integrated approaches. The first concerns the routes' total time. In this case, and for all the problem instances, the hierarchical approach has always presented much better results. The second remark is concerned with the weak performance of the hierarchical approach when the number of boxes per demand increased. These two effects are consequence of the dominance of the VRPTW in this approach. The sequential approaches are better suited for problems in which the packing problem has a big impact on the overall quality of the distribution problem solutions.

Acknowledgments: Partially supported by FCT, POSI and FEDER (CPackMO – Cutting and Packing Problems with Multiple Objectives POSI/SRI/40908/2001).

References

1. Bent, R., and Van Hentenryck, P.: A two-stage hybrid local search for the vehicle routing problem with time windows. Transportation Science (2004) 38(4): 515-530.

2. Berger, J., Barkaoui, M., and Bräysy, O.: A route-directed hybrid genetic approach for the vehicle routing problem with time windows. Information Systems and Operations Research (2003) 41: 179-194.

3. Bischoff, E. E.: Three-dimensional packing of items with limited load bearing strength. European Journal of Operational Research (2006) 168 (3): 952-966.

4. Bischoff, E. E. and Ratcliff, M. S. W.: Issues in the development of approaches to container loading. Omega, International Journal of Management Science (1995) 23: 377-390.

5. Bortfeldt, A. and Gehring, H.: A tabu search algorithm for weakly heterogeneous container loading problems. OR Spektrum (1998) 20: 237-250.

6. Bortfeldt, A. and Gehring, H.: A hybrid genetic algorithm for the container loading problem. European Journal of Operational Research (2001) 131: 143-161.

7. Bortfeldt, A. Gehring, H. and Mack, D.: A parallel tabu search algorithm for solving the container loading problem. Working Paper, Diskussionsbeitrage des Fachbereichs Wirtschaftswissenschaft der FerUniversitat Hagen, 2002.

8. Bräysy, O.: A reactive variable neighborhood search algorithm for the vehicle routing problem with time windows. Informs Journal on Computing (2003) 15 nº4: 347-368.

9. Bräysy, O., G. Hasle and W. Dullaert: A multi-start local search algorithm for the vehicle routing problem with time windows. European Journal of Operational Research (2004) 159(3): 586-605.

10. Carrabs, F., Cordeau, J.F., Laporte, G.: Variable Neigborhood Search for the Pickup and Delivery Travelling Salesman Problem with LIFO Loading. Technical report. University of Salerno, Italy, (2005).

11. Czech, Z. J. and Czarnas, P.: A parallel simulated annealing for the vehicle routing problem with time windows. Proc. 10th Euromicro Workshop on Parallel Distributed and Network-based Processing (2002) 376-383.

12. Gehring, H. and Homberger, J.: Parallelization of a two phase metaheuristic for routing problems with time windows. Asia-Pacific Journal of Operational Research (2001) 18: 35-47.

13. Gehring, H. and Bortfeldt, A.: A parallel genetic algorithm for solving the container loading problem. International Transactions in Operational Research (2002) 9: 497-511.

14. Gendreau, M., Iori, M., Laporte, G., Martello, S.:A tabu search algorithm for a routing and container loading problem. Transportation Science (2006) 9 nº3: 342-350.

15. George, J. A. and Robinson, D. F.: A heuristic for packing boxes into a container. Computers and Operations Research (1980) 7: 147-156.

16. Iori, M. I., Salazar Gonzalez, J.J., Vigo, D.: An exact approach for the symmetric capacitated vehicle routing problem with two dimensional loading constraints. Research report, DEIS, University of Bologna, Italy, (2003).

17. Moura, A. and Oliveira, J. F.: A GRASP approach to the Container Loading Problem. IEEE Intelligent Systems - Special issue on Metaheuristic Methods in Transportation and Logistics July/August (2005), 20-Nº4: 50-57.

18. Moura, A. and Oliveira, J. F.: Uma heurística composta para a determinação de rotas para veículos em problemas com janelas temporais e entregas e recolhas. Investigação Operacional (2004) 24-Nº1: 45-62.

19. Petersen, H.L.: Heuristic Solution Approaches to the Double TSP with Multiple Stacks. Technical report, Technical University of Denmark (2006).

20. Resende, M. G. C. and Pitsoulis, L. S.: Greedy randomized adaptive search procedures. Handbook of Applied Optimization, Oxford University Press, (2002).

21. Solomon, M. M.: Algorithms for the vehicle routing and scheduling problem with time windows constraints", Operations Research (1987) 35: 254-265.