Pesquisa Operacional (2012) 32(2): 443-463
© 2012 Brazilian Operations Research Society
Printed version ISSN 0101-7438 / Online version ISSN 1678-5142
www.scielo.br/pope

SOME EXPERIMENTS WITH A SAVINGS HEURISTIC AND A TABU SEARCH APPROACH FOR THE VEHICLE ROUTING PROBLEM WITH MULTIPLE DELIVERYMEN

Vanessa de Oliveira Ferreira* and Vitória Pureza

Received February 22, 2011 / Accepted April 3, 2012

ABSTRACT. In this work we consider a variant of the vehicle routing problem that allows the assignment of multiple deliverymen to one or more routes. A practical motivation for this variant arises, for example, in the distribution of beverages in highly dense urban areas, characterized by the difficulty in serving daily requests within regular working day hours with a single deliveryman per vehicle. We present a mathematical model and a savings algorithm in order to generate low cost routes that maximize the number of requests served in compliance with the maximum route time. The impact of the extra deliverymen on the solutions provided by the proposed heuristic is assessed by means of sets of generated examples based on classical instances of literature. It is also presented the results obtained by an adaptation of a tabu search approach from the literature.

Keywords: Vehicle routing, multiple deliverymen, heuristics, commercial routing systems.

1 INTRODUCTION

The management of goods and services distribution presents a variety of decisions at strategic, tactical and operational levels. Decisions regarding facilities location, such as plants, depots and warehouses, can be defined as strategic, while the definition of the fleet size and mix is seen as tactic. At the operational level, decisions on routing and scheduling of vehicles and distribution personnel are taken day by day.

In vehicle routing problems the objective is to generate routes for serving demand sites, that is, to provide for each vehicle, the sequence of visits (routing) and the times the activities should take place (scheduling). In some situations, the size of the fleet is also defined. For those problems, a key challenge for practitioners and researchers is to apply or develop methodologies to elaborate the routes so that one or more objectives are optimized. Most vehicle routing problems has combinatorial nature, and it is considered difficult to solve (Lenstra & Rinnooy-Kan, 1981) which explains the continuous research.

*Corresponding author
Universidade Federal de São Carlos – UFSCar, Rodovia Washington Luís, km 235, SP-310 13565-905 São Carlos, SP, Brasil. E-mails: vanoferr@gmail.com / vpureza@dep.ufscar.br

Routing problems appear in several fields, including the delivery of goods to customers, pickup and transportation of people with mobility difficulties, the supply of gas stations, gathering and transportation of garbage to appropriate yards, installation of equipments in different houses, among many others. Surveys of problem classification, solution methods and applications of vehicle routing can be found, for example, in Bodin *et al.* (1983), Golden & Assad (1988), Assad (1988), Ronen (1988), Laporte (1992), Osman (1993), Fisher (1995), Desroisiers *et al.* (1995), Laporte *et al.* (2000), Cunha (2000), Van Breedam (2001), Cordeau *et al.* (2002), Golden *et al.* (2002), Toth & Vigo (2002), Bräysy & Gendreau (2005a, 2005b), Zhong & Cole (2005), Cordeau *et al.* (2007), Yeun *et al.* (2008), Parragh *et al.* (2008), Laporte (2009) and Baldacci *et al.* (2010). Other studies in vehicle routing appear in Fukasawa *et al.* (2006), Montané & Galvão (2006), Reimann & Ulrich (2006), Berbeglia *et al.* (2007) and Subramanian *et al.* (2010).

With the advance of computer technology and increasing awareness of the impact of distribution costs on product prices, several commercial routing systems have been developed. In addition to data and location of customers, these systems store road network information, average speed and traffic restrictions. From these data, exact and heuristic methods elaborate efficient routes under different operational conditions (such as soft or hard time windows, multiple depots, heterogeneous fleet, node or arc routing, among others), allowing companies logistic departments to carry out the distribution with satisfactory results. A recent survey on the performance and characteristics of commercial routing software can be found in Hall & Partyka (2008).

Despite these advances, new practical situations arise and define variations of the problem for which existing methods may not be the most suitable. For instance, large service times at each demand site may prevent service to all customers within the established working hours. This situation is faced, for example, by Brazilian companies when delivering beverages to small retailers in urban areas. Due to parking difficulties, demand sites close to each other (cluster) have a parking site associated with the vehicle that will serve them. The delivery of the cargo to each demand site of the cluster is then performed by a deliveryman that visits customers by foot, very often by making more than one trip between the vehicle and each customer. As this market segment is highly competitive, the usual policy is to serve all daily requests on the same day, despite the resulting overtime costs and fines from regulatory agencies, and the inconvenience caused to the customers.

Note, however, that if the number of deliverymen assigned to each route is greater than one, service times at each customer will most likely be reduced, therefore affecting the total service time in the cluster, and increasing the number of customers that can be served within regular working hours. This new reality encourages the study of a problem little or not yet explored in the literature, and here called the vehicle routing problem with multiple deliverymen (VRPMD).

This paper aims to analyze the impact of the assignment of multiple deliverymen in route generation so as to minimize the number of unserved customers on a given work day. To this end, we propose an extension of the classic savings algorithm (Clarke & Wright, 1964). We chose to extend the savings algorithm since its effectiveness and the simple manipulations of the data set involved in its application allow large problems to be satisfactorily solved, making it the basis of most commercial VRP systems (Laporte & Semet, 2002). The extension is applied to sets of examples based on data provided by classic instances of Solomon (1987), and the results are compared to those provided by the original savings algorithm according to a set of performance criteria. In order to assess possible gains from more sophisticated methods we also provide the results obtained with an adaptation of a tabu search approach from the literature.

The remainder of this paper is organized as follows. Section 2 describes the problem of vehicle routing with multiple deliverymen tackled in this research, and Section 3 presents the associated formulation. Sections 4 and 5 respectively discuss the savings heuristic and the tabu search approach. Section 6 describes the computational experiments and results for the sets of generated instances, followed by conclusions and perspectives for future research in Section 7.

2 THE VEHICLE ROUTING PROBLEM WITH MULTIPLE DELIVERYMEN

The capacitated time-constrained VRPMD can be defined as follows: given a set of nodes, each of which requiring a given demand of goods to be delivered, a homogeneous vehicle fleet, and a set of deliverymen, define a set of routes to be carried out by the vehicles in order to serve the nodes demand. The routes should minimize operation costs and satisfy the following restrictions: (a) each route must start and finish at a central depot; (b) all visited nodes must be served by a single vehicle; (c) the demand of each served node must be completely met; (d) the vehicle load, at any time, must not exceed the vehicle capacity (without loss of generality, it is admitted that the demand of each node does not surpass the vehicle capacity); (e) the crew in each vehicle must not exceed the capacity of the vehicle's cabin; and (f) the total time spent on each route (consisting of the vehicle travel times and service times at each node) should not exceed a predetermined limit associated to end of the work day. The service time at each demand node i includes the deliverymen walking time and the cargo unloading time in the node.

Note that these restrictions have a reasonable degree of generality so as to allow the inclusion of additional constraints describing different variations of the problem. It may be required, for instance, that the fleet and/or set of deliverymen has (have) unlimited size. In this case, and assuming that the instance is feasible (the problem data allow any node i to be served by the direct route $\Delta - i - \Delta$, where Δ represents the depot), we can impose that all demand nodes must be served. Alternatively, one can consider limitations on the size of one or both resources so that service cannot be guaranteed to all nodes. For the application dealt in this paper, we assume that the fleet size is limited to F vehicles and the number of deliverymen is large enough to allow each vehicle to have the maximum crew size. We also assume that the direct route $\Delta - i - \Delta$ does not require more than one deliveryman to ensure that the maximum route time constraint is satisfied.

The costs to be minimized in the VRPMD decisions also depend on the application. If no limitations are imposed on the fleet size and the available number of deliverymen, the main objective is to minimize the number of vehicles (fixed cost), followed by the number of deliverymen, and the distance/total time of the operation (variable cost). For the application considered in this paper, these objectives are preceded by the minimization of unserved nodes. Priority is given, thus, to optimizing the level of customer service.

Note that for the beverage delivery example described in Section 1, the nodes to be routed correspond to the vehicles parking sites, and the demand of each node is the sum of the cluster's total demand. Service times at each node depend on the size of the vehicle crew, the delivery strategy (more than one deliveryman serves one customer at a time/customers are served simultaneously, each by a single deliveryman), and cluster characteristics, such as demand and customers geographic dispersion.

Despite its potential practical applications in urban areas (other examples include the tobacco and snacks distribution), as far as we know, the use of extra deliverymen was only recently formalized for the vehicle routing problem with time windows (Pureza *et al.*, 2012), hence resulting in the variant VRPTWMD. In that paper, the authors propose two heuristic procedures: a tabu search approach and an ant colony optimization algorithm. Their procedures are tailored to address situations for which there are limitations on the number of available deliverymen (rather than on the size of the fleet), which means that all demand nodes are required to be visited.

3 FORMULATION

The VRPMD described in Section 2 can be modeled based on the two-index variable model presented in Pureza *et al.* (2012) for the VRPTWMD. Since fleet and crew size in each vehicle may prevent a subset of nodes to be served, the model also include some features presented in the three-index variable model of Tang & Wang (2006) for the prize-collecting vehicle routing problem (PCVRP). The network comprises n nodes indexed by $i = 1, \dots, n$; $i = 1$ represents the depot and $i = 2, \dots, n$ refer to the parking sites (clusters). For the sake of simplicity, it is assumed that service times in each node are provided by the user.

The following notation is used for the model's description:

Input Data

n Number of nodes or clusters ($i = 1, \dots, n$); the depot is represented by node $i = 1$, and the parking sites by nodes $i = 2, \dots, n$.

L Maximum crew size (the driver plus the extra deliverymen) that can be assigned to a single vehicle ($l = 1, \dots, L$). If the crew size assigned to a vehicle is l, we say that this vehicle travels in mode l.

F Fleet size

p Prize of one served node

c_1 Cost of a vehicle

c_2 Cost of one deliveryman

c_3 Cost of an unitary distance traveled by a vehicle

Q Capacity of each vehicle

T Maximum route time

v Average speed of the vehicles

d_{ij} Distance between nodes i and $j\,(i, j = 1, \ldots, n, i \neq j)$; we consider that d_{ij} may be different from d_{ji} due to, *e.g.*, road restrictions and one-way directions (asymmetric)

$tv_i j$ Average direct travel time between nodes i and $j\,(i, j = 1, \ldots, n, i \neq j)$ (*e.g.*, $tv_{ij} = d_{ij}/v$)

st_{il} Service time in node $i = 1, \ldots, n$ with $l = 1, \ldots, L$ deliverymen; it is assumed that $st_{1l} = 0$

q_i Non-negative demand (in the same units of capacity Q) of node $i = 1, \ldots, n$, with $q_i = Q$; if $i = 2, \ldots, n$, q_i is equal to the quantities to be delivered to the cluster associated to parking site i; it is assumed that $q_1 = 0$

a Vehicles departure time at node $i = 1$; it is assumed that $a = 0$

b Vehicles latest arrival time at node $i = 1$; it is assumed that $b = T$

Variables

$$x_{ijl} = \begin{cases} 1, & \text{if the vehicle travels directly from node } i \text{ to node } j \text{ in mode } l \\ 0, & \text{otherwise} \quad (i, j = 1, \ldots, n;\ i \neq j;\ l = 1, \ldots, L) \end{cases}$$

$$w_{il} = \begin{cases} 1, & \text{if node } i \text{ is served by a vehicle in mode } l \\ 0, & \text{otherwise} \quad (i = 1, \ldots, n;\ l = 1, \ldots, L) \end{cases}$$

t_{il} Service start time of the vehicle in mode $l = 1, \ldots, L$ at node $i = 1, \ldots, n$; t_{1l} corresponds to the time the vehicle in mode $l = 1, \ldots, L$ returns to the depot

y_{il} Vehicle load in mode $l = 1, \ldots, L$ right after serving node $i = 1, \ldots, n$.

The VRPMD is formulated according to the following mixed 0-1 linear model:

$$\min z = -p \sum_{i=2}^{n} \sum_{k=1}^{K} \sum_{l=1}^{L} w_{ikl} + c_1 \sum_{i=2}^{n} \sum_{l=1}^{L} x_{1il} + c_2 \sum_{j=2}^{n} \sum_{l=1}^{L} l x_{1jl} + c_3 \sum_{i=1}^{n} \sum_{\substack{j=1 \\ j \neq i}}^{n} \sum_{l=1}^{L} d_{ij} x_{ijl} \tag{1}$$

subject to

$$\sum_{\substack{i=1 \\ i \neq j}}^{n} \sum_{l=1}^{L} x_{ijl} \leq 1, \quad j = 2, \ldots, n \tag{2}$$

$$\sum_{\substack{j=1 \\ j \neq i}}^{n} \sum_{l=1}^{L} x_{ijl} \leq 1, \quad j = 2, \ldots, n \tag{3}$$

$$\sum_{\substack{j=1 \\ j \neq i}}^{n} x_{ijl} = w_{il}, \quad i = 2, \ldots, n;\ l = 1, \ldots, L \tag{4}$$

$$\sum_{\substack{j=1\\j\neq i}}^{n} x_{jil} = \sum_{\substack{j=1\\j\neq i}}^{n} x_{ijl}, \quad i = 1, \ldots, n;\ l = 1, \ldots, L \tag{5}$$

$$t_{jl} \geq t_{il} + (st_{il} + tv_{ij})x_{ijl} - M_{ijl}(1 - x_{ijl}), \quad i = 2, \ldots, n;\ j = 1, \ldots, n; \quad i \neq j;\ l = 1, \ldots, L \tag{6}$$

$$y_{jl} \geq y_{il} + q_j x_{ijl} - Q(1 - x_{ijl}), \quad i = 1, \ldots, n;\ j = 2, \ldots, n; \quad i \neq j;\ l = 1, \ldots, L \tag{7}$$

$$\sum_{j=2}^{n}\sum_{l=1}^{L} x_{1jl} \leq F \tag{8}$$

$$w_{il}, x_{ijl} \in \{0, 1\}, \quad i, j = 1, \ldots, n;\ i \neq j;\ l = 1, \ldots, L \tag{9}$$

$$a \leq t_{1l} \leq b, \quad l = 1, \ldots, L \tag{10}$$

$$q_i \leq y_{il} \leq Q; \quad i, j = 1, \ldots, n;\ i \neq j;\ l = 1, \ldots, L \tag{11}$$

where M_{ijl} is a sufficiently large number. The objective function (1) minimizes the number of unserved nodes, the fixed and variable costs of the required fleet as well as the cost of assigned deliverymen. Constraints (2) ensure that at most one vehicle in one mode enters in each node $j(j \neq 1)$, whereas constraints (3) ensure that one vehicle in one mode at most leaves each node $i(i \neq 1)$. Constraints (4) state that the traversal of an arc i-j by a vehicle in mode $l(i = 2, \ldots, n; j = 1, \ldots, n)$, that is, $x_{ijl} = 1$, implies that node i is served in mode $l(w_{il} = 1)$. If $x_{ijl} = 0(j = 1, \ldots, n; l = 1, \ldots, L)$ then node i is not served. Constraints (5) are the flow conservation equations that ensure that the same vehicle that enters in a node i in mode l should leave this node i in the same mode l. Note that these constraints are also valid if node i is not visited (along with constraints (4), they imply that $w_{il} = x_{ijl} = 0, j = 1, \ldots, n; l = 1, \ldots, L$).

Constraints (6) define the relationship between the flow variables x_{ijl} and the variables of service start time t_{il}; since increasing service start times are required along the tour, these constraints also prevent against the formation of subtours that do not contain the depot. Note that these constraints cannot be defined for $i = 1$ (*i.e.*, departure of the depot, given that $st_{1l} = 0$ and $t_{jl} \geq tv_{1j}, j = 2, \ldots, n$), but they are defined for $j = 1$ (return to the depot), in order to guarantee that the time window of the depot is met (given that $a \leq t_{1l} \leq b$). The formation of subtours that do not contain the depot is also similarly ensured by constraints (7) which define the relationship between the flow variables x_{ijl} and the load variables y_{il} (note that the vehicle load increases along a tour). These constraints are defined for $i = 1$ (*i.e.*, departure from the depot), but need not be defined for $j = 1$ (return to the depot), given that $q_1 = 0$. Constraints (8) ensure that the size of the available fleet F is not exceeded. Constraints (9) define the domain of the variables, while constraints (10) and (11) respectively ensure that the time window of the depot is satisfied, and the maximum capacity Q of each vehicle is not exceeded. Without loss of generality, the model size can be reduced based on the input data; for instance, by fixing the variables $x_{ijl} = 0$ if $q_i + q_j > Q$ (capacity constraint violated).

The problem described by formulation (1)-(11) is NP-hard since the VRP can be seen as its particular case for which there is only one mode (*i.e.*, $L = 1$), $p \gg c_1 \gg c_3, c_2 = 0$, and F is sufficiently large. As pointed out in Bräysy and Gendreau (2005a, 2005b), the VRP is NP-hard.

4 A CONSTRUCTIVE HEURISTIC APPROACH

In the following we show how the savings algorithm proposed by Clarke & Wright (1964) can be extended for solving the VRPMD described in the previous sections. The original savings algorithm is a simple and popular constructive heuristic that applies to problems for which the number of vehicles is a decision variable (unlimited fleet). It computes the savings that arise from merging two existent routes, and iteratively selects the option with the largest savings that do not violate capacity and temporal restrictions. Note that in addition to the reduction of the total distance, each merge decrements the current number of vehicles in one unity.

Figure 1 briefly describes the sequential version of the savings algorithm (henceforth called SAV). Some of the model's notation applies in this description.

```
1. Let $S$ be a solution comprising $r = n$ routes, each of which serving a demand node (with one deliveryman).
2. (Savings computation) Compute the savings in distance resulting from serving the pair of nodes $i$ and $j (i \neq j)$ in the same route (that is, $s_{ij} = d_{i1} + d_{1j} - d_{ij}$), and store the resulting values (ordered in a non increasing fashion) in a list $\mathcal{L}$.
3. While $\mathcal{L} \neq \emptyset$:
    3.1. Starting from the first element of $\mathcal{L}$, select the pair of nodes $i$ and $j$ that satisfies one of the following conditions:
        a) Neither $i$'s nor $j$'s routes have already been merged. In this case, merging would include both $i$ and $j$ in the same route.
        b) Exactly one of the two nodes' routes ($i$'s or $j$'s) has already been merged and the corresponding node is not interior to that route. In this case, the link $i - j$ would be added to that same route.
        c) Both $i$'s and $j$'s routes are distinct from each other, have already been merged, and neither node is interior to its route. In this case, the two routes would be merged.
    3.2. Obtain a new solution $S$ by implementing the selected merge, rescheduling the resulting route, and making $r = r - 1$ if the vehicle capacity and maximum route time constraints are not violated.
    3.3. Eliminate the element associated to the selected merge from $\mathcal{L}$.
4. Return solution $S$.
```

Figure 1 – Sequential savings algorithm (SAV).

Note that if the fleet size is limited to F vehicles, the $r - F$ routes resulting from the application of algorithm SAV actually comprise unrouted nodes. On the other hand, if multiple deliverymen could be assigned to one or more routes, service times would be reduced, favoring the increase of the number of feasible merges. As a result, the number of required vehicles is reduced, and the number of served nodes increases.

The main feature of algorithm SAV with multiple deliverymen (henceforth called algorithm SAV_{md} consists in the generation of candidate solutions in which the number of deliverymen in the current solution S is iteratively incremented until the maximum crew size in each vehicle is tested or reached, or all nodes are routed in the solution. For a given iteration, the candidate solution that produces the largest number of served nodes is selected to replace S. The description of SAV_{md} is presented in Figure 2 and discussed in more detail in the following paragraphs.

```
1. Let $F$ be the fleet size ($F < n$), and $L$ be the maximum crew size that can be assigned to a vehicle. Apply algorithm SAV (described in Fig. 1), obtaining a solution with one deliveryman per route. Furthermore, for each route $k (k = 1, \dots, r)$, let $server_k$ and $tserver_k$ be respectively the number of deliverymen assigned to route $k$ (in the current candidate solution $S$), and the last tested number of deliverymen in route $k$ (that is, $server_k = tserver_k = 1$).
2. While ($r > F$) and ($server_k < L$ for some $k = 1, \dots, r$) and ($tserver_k < L$ for some $k = 1, \dots, r$):
    2.1. Let $server\_s_k = server_k$ for $k = 1, \dots, r$ (store the number of deliverymen of each route in the current solution $S$).
    2.2. For each route $k (k = 1, \dots, r)$ such that ($server_k < L$) and ($tserver_k < L$):
        2.2.1. Make $server_k = server_k + 1$ (assign an extra deliveryman to route $k$).
        2.2.2. Reschedule route $k$.
        2.2.3. Apply algorithm SAV (described in Fig. 1) from step 2 on, obtaining a candidate solution $St_k$ with $r_k$ routes.
        2.2.4. Make $tserver_k = server_k$ and $server_k = server\_s_k$ (restore $k$'s original crew size).
    2.3. If at least one feasible merge was produced in step 2.2, select $St_k$ (with incremented number of deliverymen) that serves the largest number of nodes and make $S = St_k$.
3. If $r > F$, select the $F$ routes with the largest number of nodes to comprise solution $S$.
4. Return $S$.
```

Figure 2 – Sequential savings algorithm with multiple deliverymen (SAV_{md}).

In step 1 of SAV_{md} a starting solution S with r routes is produced, each of which served by a single deliveryman. At this point, if all nodes are served ($r \leq F$) the route construction ends. Otherwise, an iterative procedure that computes the impact of an additional deliveryman in the solution (steps 2.1-2.2) is employed.

In the latter case, an extra deliveryman is temporarily assigned to each route k, one route at a time, as long as the maximum crew size (L) is not exceeded in k and the resulting provisional number of deliveryman ($tserver_k$) has not yet been tested in k. Each assignment requires the associated route to be rescheduled (step 2.2.2), hopefully increasing its temporal slack. The application of algorithm SAV is then resumed in step 2.2.3, resulting in the candidate solution St_k. The candidate solution with the largest number of served nodes is selected to replace solution S, and the additional deliveryman is permanently assigned to the associated route (step 2.3). No improvement is obtained if all candidate solutions have the same number of routes as solution S (no feasible merge is produced), hence no assignment is performed.

Steps 2.1-2.3 are repeated as long as $r > F$ and there is at least one route for which both the actual number of deliverymen and the last tested crew size is less than the maximum crew in a vehicle. If the resulting solution has $r > F$, the Froutes with the largest number of nodes are chosen to comprise S while the nodes of the $r - F$ remaining routes are not served (step 3). It should be noted, however, that many real time applications allow some percentage of violation of the maximum route time. In these cases, algorithm SAV$_{\text{md}}$ can be modified by increasing the maximum route time and penalizing merges according to the amount of the violation.

5 A TABU SEARCH APPROACH

This research mainly aims at proposing a basic routing method for environments that require a rapid response upon processing a substantial amount of data. This is the reality of many distribution companies with the difficult task of planning routes for hundreds of trucks in order to serve thousands of clients in a daily basis. Nevertheless, when the application allows longer computational time, further improvement in the solutions provided by a constructive heuristic can be attained by more sophisticated methods such as local search or metaheuristics. In order to observe the impact of such methods in relation to SAV$_{\text{md}}$, the tabu search approach proposed in Pureza *et al.* (2012) for the VRPTWMD was modified to tackle the variation of the VRPMD addressed in this work and simplified to some extent. A high-level description of the adapted approach (henceforth called TS$_{\text{vrpmd}}$) is described in Figure 3.

```
1. Apply algorithm SAV$_{md}$ (described in Section 4), followed by the tabu search algorithm for number
   of routes reduction (described in Section 5.2).
2. If all nodes are routed in the best found solution $S^*$, go to step 4.
3. Repeat
     3.1. Apply the insertion heuristic (described in Section 5.1) to include unrouted nodes in $S^*$.
     3.2. Apply the tabu search algorithm for number of routes reduction.
   Until there is no available vehicle or all nodes are routed in $S^*$.
4. If at least one route in $S^*$ is served by more than one deliveryman, apply the tabu search algorithm
   for crew size reduction.
5. Apply the tabu search algorithm for distance reduction.
6. Return $S^*$.
```

Figure 3 – The tabu search approach (TS$_{\text{vrpmd}}$).

In step 1 of the tabu search approach, reducing the number of routes of the starting solution addresses not only this objective itself, but also the minimization of the number of unrouted nodes and the total crew size. The deliverymen assigned to the eliminated routes become unnecessary, and in case of unrouted nodes (step 3) an insertion-type heuristic (step 3.1) can use the released vehicles to serve them. In order to strengthen the approach, the insertion heuristic and the number of routes reduction algorithm (step 3.2) are alternated until all vehicles are utilized or all nodes are routed. The application of the tabu search algorithm for crew size reduction (step 4) and distance reduction (step 5) provide additional improvement.

As one can see, the structure of the tabu search algorithm is general enough to be used either for reducing the number of routes, the number of deliverymen or the total distance. It should be noted that in Pureza *et al.* (2012), the fleet size can be incremented so as to guarantee that all nodes are routed.

The following two subsections describe the insertion heuristic and the tabu search algorithm employed in TS_{vrpmd}.

5.1 The insertion heuristic

We use a variant of the cheapest insertion heuristic (Lawler *et al.*, 1985) to include unrouted nodes in the solution. The heuristic is a simple but effective method that sequentially chooses the unrouted node and its insertion position in a current route that minimize the resulting extra distance. In case of no feasible insertion positions for all nodes but one or more available vehicles, a new route is created with the farthest node from the depot. A more formal description of the insertion heuristic is presented in Figure 4. Note that in Pureza *et al.* (2012), unrouted nodes are included in the solution by a procedure based on Solomon (1987)'s I1 insertion heuristic.

```
1. Let $\mathbf{U}$ be the set of unserved nodes in the best found solution $S^*$ comprising $r$ routes.
2. Repeat
    2.1. Repeat
        2.1.1. For each node $i \in \mathbf{U}$ obtain the feasible insertion position $p$ in a current route
               $v(v = 1, \dots, r)$ that minimizes the resulting extra distance.
        2.1.2. If there is at least one feasible insertion position for some node in the current routes,
               select the node $i^*$ and its insertion position $p^*$ in route $v^*$ with the smaller extra distance.
               Insert $i^*$ accordingly and reschedule $v^*$. Make $\mathbf{U} = \mathbf{U} - \{i^*\}$.
    Until there is no feasible insertion position in the current routes.
    2.2. If $\mathbf{U} \neq \emptyset$ and $r < F$, let $i^*$ be the farthest node from the depot ($i^* \in \mathbf{U}$). Start a new
         route with $i^*$, and make $\mathbf{U} = \mathbf{U} - \{i^*\}$ and $r = r + 1$.
Until there are no available vehicles ($r = F$) or all nodes are routed ($\mathbf{U} = \emptyset$).
3. Return $S^*$.
```

Figure 4 – The insertion heuristic.

5.2 The tabu search algorithm

As in the work of Pureza *et al.* (2012) we use an integrated intensification/diversification mechanism that modifies selected tabu search parameters (Glover & Laguna, 1997) based on the analysis of search trajectory patterns (the curve of solution cost *vs.* iteration). The core of the mechanism is the assumption that trajectory patterns reflect the restrictiveness level imposed by the tabu framework design. Under this assumption, setting parameter values that strategically respond to the observed patterns can be a means for improving search effectiveness.

In order to identify the current trajectory pattern, the search process is divided into stages, and for each two consecutive stages $\theta - 1$ and θ, the current average solution cost in stage $g$is compared to the previous average solution cost in stage $\theta - 1$.The search describes a stagnated trajectory, an ascent trajectory or a descent trajectory depending on whether the two averages are approximately the same, the current average is larger than the previous average, or the current average is smaller than the previous average, respectively. Once the trajectory pattern is identified, the approach reacts to it by imposing a new operational setting for some specified number of iterations. In Pureza *et al.* (2012), the number of nodes of the smaller route is used as the solution cost, and parameter changes are applied for $\sigma * hoz$ iterations, where hoz is an integer randomly generated within a pre-specified range and σ is a tuning factor whose value depends on the prescribed change. Prior to the trajectory evaluation, it is verified if the last stage corresponds to an improvement phase; in this case, the tabu parameters and the tuning factor are also set accordingly. Note that the period of application of the new operational setting corresponds to a new stage $(\theta + 1)$.

A high-level description of the algorithm is presented in Figure 5. For further details, the reader should refer to Pureza *et al.* (2012).

```
1. From the best found solution $S^*$, proceed with the search for two stages using standard parameter values.
2. Repeat until no feasible moves are available or some stopping criterion is met:
    2.1. If any solution improvement is verified in the last stage, update $S^*$. Otherwise, identify the current pattern described by the last two stages.
    2.2. Set parameters according to the trajectory pattern or to an improvement phase.
         Apply the setting for the prescribed period of application, obtaining a new stage.
3. Return $S^*$.
```

Figure 5 – The tabu search algorithm.

In our adapted approach the tabu search algorithm is first used to reduce the required fleet size (steps 1 and step 3.2 in Fig. 3), then to reduce the crew sizes on the vehicles (step 4 in Fig. 3), and finally to decrease the total traveled distance (step 5 in Fig. 3). As in the previous work, four inter-route move operators are used in these steps to generate the neighborhood structure: relocation of a single node to a different route (single insertion), relocation of two nodes in the same route to a different route (double insertion), exchange of two nodes in different routes (exchange), and an exchange type move followed by an insertion type move (exchange/insertion). These operators are instantiations of the λ-interchange operators proposed by Osman (1993) with $\lambda = 2$. Only moves that maintain the solution feasibility are allowed.

Added (removed) arcs in recent moves are classified as tabu-active and restrictiveness levels are controlled by the tabu activation rule. The latter prescribes the tolerable number of tabu-active arcs for each move type in a given search stage. Standard tolerance values are applied along the first two search stages (step 1 of Fig. 5); the first stage comprises all solutions up to the first local optimum, while the second stage ends with the solution found *hoz* iterations ahead. Minimum tolerance values are applied in the occurrence of a stagnated trajectory as means to induce diversification. Stagnated trajectories are identified when the absolute value of the percent deviation of the solution's current average from the previous average (*apd*) is less than or equal to a given threshold *ST*. In the occurrence of ascent trajectories, the prescribed parameter values depend on the range of *apd* values. The general idea is that the larger *apd* is, the more ascending is the trajectory, and more vigorous should be the relaxation imposed. Finally, if an improvement phase or a descent trajectory is identified, intensification is induced by setting large tolerance values.

Since we also use the number of nodes of the smaller route as solution cost, and the same datasets to generate test instances (see Section 6) as in the previous work, identical tabu parameter settings are employed in our experiments. These settings are summarized in Tables 1 and 2.

Table 1 – Stagnation threshold, move type tolerances and tuning factors for some trajectory types or search phase.

Trajectory type or search phase	Tolerances for move type				Tuning factor (σ)
	Single insertion	Double insertion	Exchange	Exchange/ insertion	
First two stages	3	4	6	7	does not apply
Stagnation ($ST = 2\%$)	0	0	0	0	0.5
Descent	4	5	4	5	1
Improvement phase	6	7	6	7	2

Table 2 – Move type tolerances and tuning factors for ascent trajectories.

Apd range (%)	Tolerances for move type				Tuning factor (σ)
	Single insertion	Double insertion	Exchange	Exchange/ insertion	
$[12, \infty)$	6	7	6	7	1
[10, 12)	5	6	5	5	1
[8, 10)	4	5	4	5	1
[6, 8)	3	4	3	4	1
[2, 6)	3	3	3	3	1

For the reduction of the required fleet, the lexicographic ordering used for move selection consists of the following two criteria: $\langle$maximum reduction in the number of vehicles, maximum difference of route sizes in the resulting solution$\rangle$. The basis of the second criterion is that the relocation of nodes from small routes to larger routes may eventually empty and eliminate some of the smaller routes. Note that in Pureza *et al.* (2012), two additional criteria are used to include node urgency considerations due to the existence of time windows.

For the reduction of the crew size (step 4 in Fig. 3), the four types of moves have an additional step in which the number of deliverymen of each affected route is iteratively decremented until a further reduction results in an infeasible route. The lexicographic ordering described above thus includes the resulting crew sizes as a third criterion. For distance reduction (step 5 in Fig. 3), moves are selected according to the lexicographic order: $\langle$number of vehicles, number of deliverymen, total distance$\rangle$.

Each individual application of the algorithm of reduction of the number of vehicles, crew or distance is performed for 15,000 iterations in case there are feasible moves. After each application, the best solution found so far is further subjected to a simple local search provided by intra-route 2-opt moves (Croes, 1958).

Differently from the work of Pureza *et al.* (2012), the period in which changes in parameter setting are applied (*hoz*) and the tabu tenure are not randomly drawn from some specified ranges, but set to $\left[3/2\sqrt{n}\right]$ and $\left[\sqrt{n}\right]$, respectively. These correspond to the mean values of the ranges used in the former work, making TS_{vrpmd} a deterministic approach.

6 COMPUTATIONAL EXPERIMENTS

Algorithms SAV, SAV_{md} and TS_{vrpmd} were implemented in Borland Delphi 7, and the experiments were performed in a microcomputer Intel Core2 2.4 GHz with 2 GB RAM. The purpose of our tests is to assess the relative gains of SAV_{md} to SAV and of TS_{vrpmd} to SAV_{md} in terms of number of served nodes, and the impact of multiple deliverymen solutions on the number of vehicles and total distance. For this end, 108 examples with 100 demand nodes were generated from benchmark instances of sets C1, C2, R1, R2, RC1 and RC2, proposed in Solomon (1987) for the VRP with time windows. Since the time windows are the only factors that differentiate instances in the same set, only one instance from each set can be used as seed. For all generated examples, the original x and y coordinates of each node in the seed instance are maintained and the maximum crew size in each vehicle is set to 3.

Six scenarios prescribe how service times, demand, fleet size and vehicles' capacity are modified in order to generate the examples. In scenario 1, the original data of the seed instance is maintained, with the exception of the fleet size which is set to 18 vehicles for all cases. This value corresponds to the largest trivial lower bound on the number of vehicles (the trivial bound of a particular instance is given by $\sum_{i=2}^{n} q_i/Q$) computed for all seed instances.

The following four scenarios are similar to scenario 1 except for one factor. In scenario 2 the demand in each node is doubled. In scenario 3 service times in each node $i(i = 2, \dots, n)$ are computed by the function

$$st_i = \min \left\{ \varphi \times q_i, T - 2 \times tv_{1i} \right\} \tag{12}$$

where q_i, as before, is the product demand of node i and ϕ is the number of products delivered per time unit in each node (in the experiments we used $\varphi = 2$). As aforementioned, T is the maximum return time for all vehicles to the depot, and tv_{1i} is the travel time between the depot and node i (*i.e.*, $tv_{1i} = d_{1i}/v$; in the experiments we used $v = 1$). Note that st_i in (12) is proportional to the demand q_i of node i when the first term in the right-hand-side of (12) is the minimum. The second term in (12) is to ensure that this service time does not turn the instance infeasible regarding the maximum route time T. In the experiments, the value of st_i in (12) corresponds to the service time in node i with one deliveryman. For the sake of simplicity, for the cases with two and three deliverymen we simply divided st_i by 2 and 3, respectively, so that $st_{il} = st_i/l$ for $l = 1, 2, 3$. As pointed out in Pureza *et al.* (2012), in practice service times may not be linearly dependent on the crew size.

In scenario 4 the fleet size is reduced to a third of its original value (that is, to 6 vehicles) while in scenario 5, the vehicle's capacity is reduced by 15%. Finally, the sixth scenario combines the prescribed modification of the previous four scenarios. Table 3 sumarizes the above discussion.

Table 3 – Characteristics of the test scenarios.

Scenario	st_i	q_i	F	Q
1	○	○	●	○
2	○	○ × 2	●	○
3	(12)	○	●	○
4	○	○	● × 1/3	○
5	○	○	●	○ × 0,85
6	(12)	○ × 2	● × 1/3	○ × 0,85

○ As prescribed by the seed instance. ● 18 vehicles.

The last necessary piece of data for generating our test examples consists of the maximum route time T. We use 3 different values (210, 270 and 330 t.u.) for each scenario, totalizing 108 (6 seed instances × 6 scenarios × 3 maximum route time values) examples. It should be noted that for all instances, each direct route with one deliveryman is feasible for $T = 210$.

6.1 Results for SAV$_{\text{md}}$

Table 4 presents the average results obtained by SAV$_{\text{md}}$ for the six instances in each scenario and maximum route time T. Columns SERV, VEH, DLVM and DIST and TIME show, respectively, the number of served nodes, the required number of vehicles, the required number of deliverymen, the average total distance traveled by the vehicles, and the average route time expressed as the percent deviation from SAV corresponding results. Furthermore, columns SERV and VEH depict, in parenthesis, the percentage of served nodes and used vehicles obtained by the proposed algorithms regarding the *total* number of nodes and fleet size (small percentages of served nodes and large percentages of used vehicles characterize difficult instances). Runtimes are not presented since they are very short, typically below 1 second. As expected, SAV$_{\text{md}}$ produce better solutions than SAV for all examples, considering the lexicographic order ⟨number of served nodes, number of vehicles, number of deliverymen, total distance⟩ employed in model (1-11).

Table 4 – Computational results: relative performance of SAV$_{\text{md}}$ to SAV.

Scenario	T (t.u.)	SERV (%)	VEH (%)	DLVM (%)	DIST (%)	TIME (%)
1	210	62.1 (87.2)	0.0 (72.2)	57.4	21.5	7.7
	270	44.0 (94.5)	–2.1 (63.0)	49.1	18.7	1.8
	330	28.8 (98.3)	0.0 (60.2)	40.8	29.8	0.6
2	210	62.1 (87.2)	0.0 (83.3)	57.4	21.5	7.7
	270	44.0 (94.5)	–2.1 (77.8)	48.2	17.9	2.1
	330	26.9 (97.3)	0.0 (75.0)	38.0	30.6	0.7
3	210	8.0 (83.0)	0.0 (100.0)	14.8	5.9	1.0
	270	4.8 (98.5)	–5.9 (92.6)	8.3	3.8	–2.7
	330	0.0 (100.0)	–8.9 (75.9)	0.0	–3.5	–6.4
4	210	56.5 (50.2)	0.0 (100.0)	66.7	26.3	2.4
	270	69.1 (65.2)	0.0 (100.0)	69.5	4.6	0.2
	330	47.2 (73.2)	0.0 (100.0)	63.9	23.8	–1.0
5	210	62.1 (87.2)	–1.5 (73.2)	57.4	21.1	7.1
	270	44.0 (94.5)	–2.1 (65.7)	49.1	18.7	1.8
	330	28.8 (98.3)	0.0 (63.0)	40.8	29.8	0.6
6	210	69.7 (33.8)	0.0 (100.0)	133.4	20.5	2.3
	270	41.1 (34.3)	0.0 (100.0)	88.9	25.9	0.4
	330	29.8 (37.0)	0.0 (100.0)	58.4	9.7	1.8

Scenario 1 is the test setting for which T is the main limiting factor for route construction. This may be verified by the increase in the percentage of served nodes (reaching 98.3%) along with the decrease of the number of required vehicles (from 72.2% to 60.2%) as T gets larger. The average deviation in the number of served nodes reaches its peak (62.1%) with $T = 210$. Even though SAV$_{\text{md}}$ uses the same average number of vehicles as SAV, the total distance is considerably larger (21.5%), and the number of deliverymen represents an increase of 57.4% (1.8 deliverymen per vehicle). When $T = 330$, gains in served nodes go down to 28.8%. This can be explained by the fact that although SAV$_{\text{md}}$ significantly improves the number of served nodes, the increase in T also allows a large improvement with the application of SAV (73.8% to 83.6% of served nodes).

When the demand of each node is doubled (scenario 2), serving all requests become just a little more difficult because most of the increase in the demand is satisfied by using vehicles that are idle in scenario 1. The relative performance of SAV$_{\mathrm{md}}$ is almost identical to what is observed in scenario 1, reaching its peak with 62.1% of additional served nodes for $T = 210$.

In scenario 3, service times are a function of the node demand and this seems to have a different effect on the algorithm performance. The increase in the percentage of served nodes is smaller than for the other two scenarios when $T = 210$ (83.0% for SAV$_{\mathrm{md}}$) but reaches 100% with $T = 330$ for both algorithms. These results along with the fact that all vehicles are used when $T = 210$ and a quarter of them become idle when T = 330 indicate that service times consume more of the route time available in this scenario than in the previous two. The average deviation in number of served nodes reaches its peak (8.0%) with $T = 210$, requiring an increase of 14.8% in the number of deliverymen. It should be noted that when $T = 330$, the same amount of deliverymen is used in both algorithms even though SAV$_{\mathrm{md}}$ requires 8.9% less vehicles. This means that the number of extra deliverymen assigned is equal to the number of drivers that became unnecessary with the reduction of the number of vehicles.

The reduction of the fleet size F in $1/3$ (scenario 4) makes this parameter a strong limiting factor for route construction. The percentage of served nodes is smaller than in the previous scenarios for all T, reaching a maximum of 73.2% for SAV$_{\mathrm{md}}$ with $T = 330$. All vehicles are used in each T. The average gain in number of served nodes of SAV$_{\mathrm{md}}$ relative to SAV reaches its peak (69.1%) with $T = 270$. Note that even though only 65.2% of nodes are being served, SAV$_{\mathrm{md}}$ uses 69.5% more deliverymen (1.7 deliverymen per vehicle) than SAV, and these numbers remain high for $T = 210$ and 330.

Results for scenario 5 (reduction of the vehicle's capacity Q in 15%) are very similar to what is found for scenario 1. The increase in the percentage of served nodes (reaching up 98.3%) along with the decrease of the number of required vehicles (from 73.2% to 63.0%) shows that T (rather than Q) is the main limiting factor. Note that the decrease in capacity is compensated by using vehicles that are idle in scenario 1.

Scenario 6 is the most restrictive of all test environments (as mentioned before, it combines all modifications of the four previous scenarios). This is reflected on the smaller percentage of served nodes and larger percentage of used vehicles for all T. As one can see, the combination of the modifications do not contribute additively to the gains of SAV$_{\mathrm{md}}$ relative to SAV (compare, for instance, the average deviation in number of served nodes with $T = 270$ with the corresponding route time in scenario 4) and to the extra costs in number of deliverymen and distance. On the other hand, the use of extra deliverymen substantially improves the service level, providing the largest gain in number of served nodes found in the experiments (69.7% when $T = 210$).

6.2 Results for TS$_{\mathrm{vrpmd}}$

Table 5 shows the average results obtained by TS$_{\mathrm{vrpmd}}$ for the six instances in each scenario and maximum route time T. Columns SERV, VEH, DLVM and DIST and TIME refer to the same performance criteria as in the previous table and are given as the percent deviation from SAV$_{\mathrm{md}}$ corresponding average results. Average times (in seconds) required to find the best solutions are also presented (column CPU).

Table 5 – Computational results: relative performance of $\mathrm{SAV_{md}}$ to SAV.

Scenario	T	SERV	VEH	DLVM	DIST	TIME	CPU
	(t.u.)	(%)	(%)	(%)	(%)	(%)	(%)
1	210	5.2 (91.7)	0.0 (72.2)	1.7	–12.2	–2.1	359
	270	5.5 (99.7)	–1.2 (62.2)	4.1	–10.8	–3.8	648
	330	1.7 (100)	–5.9 (56.7)	–0.4	–12.0	–5.4	253
2	210	4.9 (91.5)	0.0 (83.3)	1.1	–11.8	–1.8	275
	270	4.0 (98.3)	0.0 (77.8)	3.0	–7.9	–2.4	443
	330	2.1 (99.3)	–3.7 (72.2)	1.0	–11.3	–5.0	192
3	210	20.5 (100)	–10.0 (90.0)	42.9	–5.6	–11.4	298
	270	1.5 (100)	0.0 (92.6)	4.2	–7.9	–3.1	343
	330	0.0 (100)	0.0 (75.9)	0.0	–4.6	–1.7	164
4	210	27.5 (64.0)	0.0 (100.0)	68.5	6.5	–5.5	100
	270	8.9 (71.0)	–3.3 (96.7)	30.3	0.0	–5.4	430
	330	2.7 (75.2)	0.0 (100.0)	1.0	–4.0	0.3	108
5	210	5.2 (91.7)	0.0 (73.2)	0.2	–12.4	–1.8	337
	270	5.5 (99.7)	–1.3 (64.8)	4.1	–10.9	–3.8	589
	330	1.7 (100)	–5.9 (59.3)	–0.4	–12.0	–5.5	223
6	210	48.9 (50.3)	0.0 (100.0)	7.8	22.9	–5.3	74
	270	64.2 (56.3)	0.0 (100.0)	6.2	21.1	–5.3	127
	330	71.6 (63.5)	0.0 (100.0)	18.4	38.2	–12.7	101

As expected, $\mathrm{TS_{vrpmd}}$ improves $\mathrm{SAV_{md}}$ solutions for all examples considering the lexicographic order ⟨number of served nodes, number of vehicles, number of deliverymen, total distance⟩ employed in model (1-11). Largest gains in scenarios 1, 2 and 5 are mostly concentrated in traveled distance which may suggest that the other performance criteria in $\mathrm{SAV_{md}}$ solutions have reasonable quality in these settings.

Results for scenario 3 reinforce the observation presented in subsection 6.1 that service times as a function of the node demand consume more of the route time than service times provided by the seed instances. $\mathrm{TS_{vrpmd}}$ considerably improves the service levels (20.5% of served nodes) when $T = 210$. It should be emphasized the role of the insertion heuristic (Fig. 4) when short maximum route times limits the effectiveness of route merging type algorithms. The route reduction algorithm is particularly beneficial when F is small (scenario 4) and T is short; the average gain in number of served nodes reaches 27.5%. Note however that such improvements require an increase greater than 40% in number of deliverymen for both scenarios. Finally, the largest gains in service levels obtained in scenario 6 highlight the positive impact of more sophisticated algorithms when treating difficult settings.

The analysis of the preceding sections allows us to list the studied scenarios in decreasing order of difficulty (regarding the model's objective function) as: the combined modifications (scenario 6), fleet reduction in $1/3$ (scenario 4), doubled demand (scenario 2), reduction of the vehicles' capacity in 15% (scenario 5), standard setting (scenario 1) and service times as a function of the demand (scenario 3). The largest gains obtained by $\mathrm{SAV_{md}}$ relative to SAV (and by $\mathrm{TS_{vrpmd}}$

relative to SAV_{md} as well) with these sets of instances were usually obtained for the most difficult environments, that is, when T is short and the available fleet is small.

7 CONCLUSIONS AND PERSPECTIVES OF FUTURE RESEARCH

This paper addressed a variant of the vehicle routing problem, which despite representing several real distribution systems, has not yet received enough attention from the literature and commercial routing systems. The variant allows the use of multiple deliverymen in each route as means to reduce service times and facilitate service to a larger number of requests within regular working hours. For this end, we developed an extended version of the classic savings algorithm which iteratively tests and selects increments in the number of deliverymen in each route. We also presented the results with an adaptation of a tabu search approach for a related problem.

Results obtained with six sets of generated 100-node examples based on Solomon (1987)'s instances has shown that the proposed extension (SAV_{md}) can provide relevant gains relative to the solutions with one deliveryman. Considering the sets all together, SAV_{md} produced average increments in the number of served nodes of 40% with 1.2% less vehicles. In contrast, such gains often imply in larger route times and total traveled distances (18%) and a substantial increase (52%) in number of deliverymen. The application of the tabu search approach (TS_{vprmd}) improves the solutions of SAVmd in number of served nodes, required fleet, total distance and route time in 15.6%, 1.7%, 1.93% and 4.5%, respectively, with an increase of 10.7% in the required crew.

We conclude that the assignment of multiple deliverymen can provide significant benefits to companies for which customer service is a priority issue. An important development for this research thus consists in applying the heuristics to real world instances and performing a qualitative analysis of the solutions, considering what is actually adopted in practice. Solving the model with an exact method can also give more insight on the performance of SAV_{md} and TS_{vprmd}, even though the difficulties in deriving tight lower bounds on the objective value of vehicle routing problems will most likely imply in a slow convergence rate if partial enumeration methods are employed (Cordeau *et al.*, 2002). An interesting perspective for future research is to modify the model and proposed methodologies in order to include features such as heterogeneous fleet, multiple depots, and mixed pickup and delivery of goods. The last feature in particular represents more realistically many distribution systems in the beverage industry (among others), since empty bottles are very often picked up in each demand point and returned to the depot upon the last visit completion. In addition, a study of how service times vary with the cluster demand and configuration, the crew size, and the delivery strategy is essential for supporting VRPMD decisions in practice. Finally, due to its simplicity and short runtimes, SAV_{md} could be used as the basis for an algorithm suitable for commercial routing systems.

ACKNOWLEDGMENTS

The authors thank the two anonymous referees for their valuable comments and sugestions, and Bebidas Ipiranga for their collaboration. This study was partially supported by CAPES (DS) and CNPq (grant 303001/2009-7).

REFERENCES

[1] ASSAD AA. 1988. Modeling and implementation issues in vehicle routing. *Vehicle Routing: Methods and Studies*, 7–46.

[2] BALDACCI R, TOTH P & VIGO D. 2010. Exact algorithms for routing problems under vehicle capacity constraints. *Annals of Operations Research*, **175**: 213–245.

[3] BERBEGLIA G, CORDEAU J, GRIBKOVSKAIA I & LAPORTE G. 2007. Static pickup and delivery problems: A classification scheme and survey. *TOP: business and economics*, **15**(1): 1–31.

[4] BODIN LD, GOLDEN BL, ASSAD AA & BALL M. 1983. Routing and scheduling of vehicle and crews, the state of the art. *Computers and Operational Research*, **10**: 69–211.

[5] BRÄYSY O & GENDREAU M. 2005a. Vehicle routing problem with time windows, part I: Route construction and local search algorithms, *Transportation Science*, **39**(1): 104–118.

[6] BRÄYSY O & GENDREAU M. 2005b. Vehicle routing problem with time windows, part II: Metaheuristics, *Transportation Science*, **39**(1): 119–139.

[7] CLARKE G & WRIGHT WJ. 1964. Scheduling of vehicle from a central depot to a number of delivery points. *Operations Research*, **12**: 568–581.

[8] CORDEAU JF, DESAULNIERS G, DESROSIERS J, SOLOMON MM & SOUMIS F. 2002. The VRP with time windows. In: *The vehicle routing problem. SIAM. Monographs on Discrete Mathematics and applications* [edited by TOTH P & VIGO D], Philadelphia, PA, 157–193.

[9] CORDEAU JF, LAPORTE G, SAVELSBERGH M & VIGO D. 2007. Vehicle routing. In: *Transportation, Handbooks on Operations Research and Management Science* [edited by BARNHART C & LAPORTE G], North Holland, Amsterdam, 367–428.

[10] CROES A. 1958. A Method for Solving Traveling-Salesman Problems. *Operations Research*, **5**: 791–812.

[11] CUNHA CB. 2000. Aspectos práticos da aplicação de modelos de roteirização de veículos a problemas reais. *Transportes*, **8**(2): 51–74.

[12] DESROISIERS J, DUMAS Y, SOLOMON M & SOUMIS F. 1995. Time constrained routing and scheduling. In: *NetworK Routing, Handbooks in Operations Research and Management Science* [edited by BALL MT, MAGNANTI L, MONMA CL & NEMHAUSER GL], North-Holland, Amsterdam, 35–139.

[13] FISHER M. 1995. Vehicle routing. In: *Network Routing, Volume 8 of Handbooks in Operations Research and Management Science* [edited by BALL MT, MAGNANTI L, MONMA CL & NEMHAUSER GL], Elsevier Science, Amsterdam, 1–33.

[14] FUKASAWA R, LONGO H, LYSGAARD J, POGGI DM, REIS M, UCHOA E & WERNECK RF. 2006. Robust branch-and-cut-and-price for the capacitated vehicle routing problem. *Mathematical Programming*, **106**(3): 491–511.

[15] GLOVER F & LAGUNA M. 1997. *Tabu Search*. Kluwer Academic Publishers, Massachusetts.

[16] GOLDEN B & ASSAD A. 1988. *Vehicle routing: Methods and studies*, Elsevier Science Publishers, North-Holland, Amsterdam.

[17] GOLDEN BL, ASSAD AA & WASIL EA. 2002. Routing vehicles in the real world: Appplications in the solid waste, beverage, food, dairy, and newspaper industries. In: *The Vehicle Routing Problem. SIAM. Monographs on Discrete Mathematics and Applications* [edited by TOTH P & VIGO D], Philadelphia, 245–286.

[18] HALL RW & PARTYKA J. 2008. Software Survey: On The Road to Mobility. *OR/MS Today*, **35**(1). Available in: <http://www.lionhrtpub.com/orms/orms-2-08/frvrss.html>.

[19] LAPORTE G. 2009. Fifty years of vehicle routing. *Transportation Science*, **43**(4): 408–416.

[20] LAPORTE G & SEMET F. 2002. Classical heuristics for the capacitated VRP. In: *The Vehicle Routing Problem. SIAM. Monographs on Discrete Mathematics and Applications* [edited by TOTH P & VIGO D], Philadelphia, 109–128.

[21] LAPORTE G. 1992. The vehicle routing problem: An overview of exact and approximate algorithms. *European Journal of Operational Research*, **59**: 345–358.

[22] LAPORTE G, GENDREAU M, POTVIN JY & SEMET F. 2000. Classical and modern heuristics for the vehicle routing problem. *Operational Research*, **7**(4-5): 285–300.

[23] LAWLER EL, LENSTRA JK, RINNOOY KAN AHG & SHMOYS DB. 1985. *The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization*, John Wiley & Sons.

[24] LENSTRA J & RINNOOY-KAN A. 1981. Complexity of vehicle routing and scheduling problems. *Networks*, **11**: 221–227.

[25] MONTANÉ FAT & GALVÃO RD. 2006. A tabu search algorithm for the vehicle routing problem with simultaneous pick-up and delivery service. *Computers & Operations Research*, **33**: 595–619.

[26] OSMAN IH. 1993. Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem. *Annals of Operations Research*, **41**: 421–451.

[27] PARRAGH S, DOERNER K & HARTL R. 2008. A survey on pickup and delivery problems. *Journal für Betriebswirtschaft*, **58**(1): 21–51.

[28] PUREZA V, MORABITO R & REIMANN M. 2012. Vehicle routing with multiple deliverymen: modeling and heuristic approaches for the VRPTW. *European Journal of Operational Research*, **218**(3): 636–647.

[29] REIMANN M & ULRICH H. 2006. Comparing backhauling strategies in vehicle routing using Ant Colony Optimization. *Central European Journal of Operations Research*, **14**(2): 105–123.

[30] RONEN D. 1988. Perspectives on practical aspects of truck routing and scheduling. *European Journal of Operational Research*, **35**(2): 137–145.

[31] SOLOMON MM. 1987. Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research*, **35**(2): 254–265.

[32] SUBRAMANIAN A, DRUMMOND LMA, BENTES C, OCHI LS & FARIAS R. 2010. A parallel heuristic for the vehicle routing problem with simultaneous pickup and delivery. *Computers & Operations Research*, **37**(11): 1899–1911.

[33] TANG L & WANG X. 2006. Iterated local search algorithm based on very large-scale neighborhood for prize-collecting vehicle routing problem. *The International Journal of Advanced Manufacturing Technology*, **29**(11-12): 1246–1258.

[34] TOTH P & VIGO D. 2002. An overview of vehicle routing problems. In: *The Vehicle Routing Problem. SIAM. Monographs on Discrete Mathematics and Applications* [edited by TOTH P & VIGO D], Philadelphia, 1–26.

[35] VAN BREEDAM A. 2001. Comparing descent heuristics and metaheuristics for the vehicle routing problem. *Computers & Operations Research*, **28**(4): 289–315.

[36] YEUN LC, ISMAIL WR, OMAR K & ZIROUR M. 2008. Vehicle routing problem: Models and solutions. *Journal of Quality Measurement and Analysis – JQMA*, **4**(1): 205–218.

[37] ZHONG Y & COLE MH. 2005. A vehicle routing problem with backhauls and time windows: a guided local search solution. *Transportation Research Part E*, **41**: 131–144.