Int. J. Production Economics 144 (2013) 68–75

Contents lists available at SciVerse ScienceDirect

Int. J. Production Economics

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

The heterogeneous fleet vehicle routing problem with overloads and time windows

Manolis N. Kritikos *, George Ioannou

Management Science Laboratory, Department of Management Science and Technology, Athens University of Economics and Business, Evelpidon 47A & Lefkados 33, 113-62 Athens, Greece

ARTICLE INFO

Article history:
Received 30 September 2011
Accepted 18 January 2013
Available online 29 January 2013

Keywords:
Vehicle routing
Heuristics
Time windows

ABSTRACT

This paper addresses a new variant of the vehicle routing problem (VRP) with time windows, in which the vehicle fleet comprises units of different capacities and some overloads (i.e., loading vehicles above nominal capacity) are allowed. Although often encountered in practice, the problem, which we call heterogeneous fleet vehicle routing with overloads and time windows (HFVROTW), has not been previously tackled in the literature. We model it by integrating the constraint of the total trip load into the objective function, and solve it via a sequential insertion heuristic that employs a penalty function allowing capacity violations but limiting them to a variable predefined upper bound. Computational results on benchmark problems show the effectiveness of the proposed approach in reducing vehicle costs with minimal capacity violations, thus offering evidence of the significance of this VRP variant.

1. Introduction

Real-life vehicle routing emanates in a large variety of cases where goods or people have to be moved between locations in specific time intervals by different transportation means (Golden et al., 2008). The academic community has studied several versions of vehicle routing and scheduling problems, proposing an immense list of solution approaches, ranging from simple heuristics to complex meta-heuristics and exact methods—see Gendreau and Tarantilis (2010) for a thorough review of past and recent developments in the field. Most researchers though, have concentrated on simplified instances of the problem, i.e., instances that do not accommodate constraints or objectives often encountered in practice, which guide the actual solutions in real vehicle routing and scheduling problems—see Kritikos and Ioannou (2010) for such a discussion.

In this work, we study the heterogeneous fleet vehicle routing with overloads and time windows (HFVROTW), motivated by the fact that real-life scheduling of vehicles involves, to some extent and in addition to time restrictions, some small capacity violations in most distribution scenarios or public transportation in urban or rural areas. Indeed, bus overloads in peak hours, truck overloads enforced by large product demand, or overloads in power and telecommunication networks, are often encountered in real-life but when approaching routing from a research perspective, these facts are always ignored.

The HFVROTW can be described as follows: Consider a heterogeneous fleet of vehicles, i.e., comprising vehicles with different capacities, located at a central depot (distribution center or transportation hub). The vehicles are required to serve a set of customers, which are geographically dispersed in the area covered by the depot. Each customer has a known demand and a time window for service. Also, there is a service time associated with each customer, and the distance between each pair of customers is known, as is the distance between all customers and the depot. In the solution of the HFVROTW, vehicles are allowed to carry load over their capacity at a penalty incurred in the total solution cost. Thus, in contrast to the classical vehicle routing problem, the goal of the HFVROTW is to minimize a combined objective of the total distance traveled by vehicles, the fixed costs of vehicles performing service, and the capacity violations of all vehicles included in the final schedule. An implicit decision embedded in the problem is the selection of the fleet's composition, i.e., how many vehicles of each available type (capacity) are selected for service.

To our knowledge, research related to the HFVROTW is non-existent. Most approaches deal either with: (a) the heterogeneous vehicle routing problem (HVRP—i.e., the problem with vehicles of different capacities) or (b) the heterogeneous vehicle routing problem with time windows (HVRPTW—i.e., the previous problem with time windows). Heuristic methods proposed for the HVRP include, as described in Golden et al. (1984), adaptations of the Clarke and Wright savings algorithms, the giant tour partitioning approach, the

* Corresponding author. Tel.: +30 210 8203449; fax: +30 210 8828078.
E-mail addresses: kmn@aueb.gr (M.N. Kritikos), ioannou@aueb.gr (G. Ioannou).

0925-5273/$ - see front matter
http://dx.doi.org/10.1016/j.ijpe.2013.01.020

matching based savings heuristics, the generalized assignment based heuristic, the sophisticated improvement based heuristic, composite heuristics, and a multi level composite heuristic for the multi-depot HVRP.

Liu and Shen (1999) were the first to tackle the HVRPTW and developed a number of parallel insertions heuristics based on the insertion scheme of Solomon, and embedding in the calculations of the relevant criteria the acquisition costs of Golden et al. (1984). Dullaert et al. (2002) proposed a sequential construction algorithm, extending Solomon's I1 heuristic with vehicle insertion savings calculations based again on the criteria of Golden et al. (1984). Dondo and Cerda (2007) proposed a 3-phase algorithm for the multi-depot HVRPTW motivated by cluster-based optimization, while Paraskevopoulos et al. (2008) presented a two-phase solution framework relying on a hybridized tabu search integrated within a new reactive variable neighborhood search meta-heuristic algorithm, with very good results. Braysy et al. (2008) presented a deterministic annealing metaheuristic for the HVRPTW, outperforming the results of Liu and Shen (1999), and Braysy et al. (2009) developed a linearly scalable hybrid threshold-accepting and guided local search meta-heuristic for solving large scale HVRPTW instances. Finally, Repoussis and Tarantilis (2010) proposed an Adaptive Memory Programming solution approach for the HVRPTW that provides very good results in the majority of the benchmark instances examined.

In a relevant research thread, Rochat and Semet (1994) developed a tabu search approach for the HVRPTW, which takes into account the drivers' breaks and possible accessibility restrictions. Brandao and Mercer (1997) developed also a tabu search for the multi-trip vehicle routing and scheduling problem, in which each vehicle can make several trips per day, while access can be restricted for some vehicles to some customers; the algorithm the authors proposed allows for both weight and volume capacity restrictions on the vehicles.

In this work, we address for the first time the vehicle routing problem with time windows when the fleet is heterogeneous, i.e., comprises vehicles of different capacities and associated costs, and overloads are allowed up to a pre-specified bound, at a penalty though embedded in the problem's objective function. The penalty, which is a measure of the deviation of the actual load from the nominal vehicle capacity, is similar to the one presented by Gheysens et al. (1984), while the capacity bound varies in order to examine a large area of the potential solution space. For the solution of the HFVROTW we propose a simple solution method, i.e., a sequential insertion heuristic, extending the traditional insertion criteria of Solomon (1987), and adapting Golden et al.'s (1984), Dullaert et al.'s (2002), and Liu and Shen's (1999) ones. The computational results on benchmark problems reinforce our intuition for practical applicability of the proposed approach, with minimal adverse effects on vehicle loads and positive impact on total costs.

The remainder of the paper is organized as follows. In Section 2 the HFVROTW is formulated. Section 3 offers the basic contribution of our work, i.e., the way we devise and employ the overload penalty function, the criteria we use within the solution schemes, and the overall solution approach we propose. Section 4 presents an illustrative example based on sample literature instances, and Section 5 includes computational results on benchmark problems. Finally, Section 6 provides our concluding remarks and suggestions for future research.

2. Mathematical model

The HFVROTW can be stated as follows: Find a set of closed routes, for a fleet of T vehicles with known capacities $C_1, C_2, \ldots, C_T$, servicing a set of $|L|-1=n-1$ customers, from a central depot at minimum cost. L is the set of customers including the depot, which is a distinct node of the underlying connected graph. Indices i, j and u refer to customers and take values between 2 and n, while index $i=1$ refers to the depot; an additional index k counts the vehicles. Vehicles are initially located at the central depot. Each customer i poses a demand q_i, requires a service time, s_i, has a time window $[e_i, l_i]$, and is serviced by exactly one vehicle. There is a cost c_{ij}^k, (related to the travel time t_{ij}^k and distance d_{ij}) associated with the path from customer i to customer j, using vehicle k. Furthermore, a fixed acquisition cost f_k is incurred for each of vehicle k in the routes. Each route originates and terminates at the central depot and must satisfy the time window constraints, i.e., a vehicle cannot start servicing customer i before e_i and after l_i; however, the vehicle can arrive before e_i and wait for service. Note that capacity constraints are relaxed in the HFVROTW.

Gheysens et al. (1984) presented a mathematical programming formulation for the HVRP. We extend their model to formulate the HFVROTW, the variables of which include: (a) the arrival–departure time to/from customer i, respectively, denoted by a_i and p_i for each customer I; (b) the vehicle load Q_k; (c) he sequence in which vehicles visit customers, x_{ij}^k, and (d) the activation of a vehicle k, z_k. Variables (c) and (d) are defined as follows:

$$x_{ij}^k = \begin{cases} 1, \text{ if vehicle } k \text{ travels from } i \text{ to } j \\ \text{and} \\ 0, \text{ otherwise} \end{cases} \tag{1}$$

$$z_k = \begin{cases} 1, \text{ if vehicle } k \text{ is active} \\ 0, \text{ otherwise} \end{cases} \tag{2}$$

Given the above-defined variables the HFVRPOTW can be formulated as follows:

$$\text{Minimize} \quad \sum_{k=1}^{T}\sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}x_{ij}^k + \sum_{k=1}^{T} f_k z_k + \sum_{k \in T*} \lambda_k (Q_k - C_k) \tag{3}$$

Subject to:

$$\sum_{i=1}^{n}\sum_{k=1}^{T} x_{ij}^k = 1, \quad \forall j = 2,3,\ldots,n \tag{4}$$

$$\sum_{j=1}^{n}\sum_{k=1}^{T} x_{ij}^k = 1, \quad \forall i = 2,3,\ldots,n \tag{5}$$

$$x_{ij}^k \le z_k, \quad \forall i,j = 1,2,\ldots,n, \quad \forall k = 1,2,\ldots,T \tag{6}$$

\

$$\sum_{j=2}^{n} x_{1j}^k \le 1 \quad \forall k = 1,2,\ldots,T \tag{7}$$

$$\sum_{i=2}^{n} x_{i1}^k \le 1 \quad \forall k = 1,2,\ldots,T \tag{8}$$

$$\sum_{i=1}^{n} x_{iu}^k - \sum_{j=1}^{n} x_{uj}^k = 0, \quad \forall k = 1,\ldots,T, \quad \forall u = 2,\ldots,n \tag{9}$$

$$a_j \ge (p_i + t_{ij}) - (1 - x_{ij}^k)M, \quad \forall i,j = 1,2,\ldots,n, \quad \forall k = 1,2,\ldots,T \tag{10}$$

$$a_j \le (p_i + t_{ij}) - (1 - x_{ij}^k)M, \quad \forall i,j = 1,2,\ldots,n, \quad \forall k = 1,2,\ldots,T \tag{11}$$

$$a_i \le p_i - s_i, \quad \forall i = 1,\ldots,n \tag{12}$$

$$e_i \le p_i - s_i \le l_i, \quad \forall i = 1,\ldots,n \tag{13}$$

$$a_1 = 0 \tag{14}$$

$$\sum_{i=1}^{n} q_i \Big(\sum_{j=1}^{n} x_{ij}^k\Big) = Q_k, \quad \forall k = 1,2,\ldots,T \tag{15}$$

$$Q_k \ge C_k(1+\alpha), \quad \forall k = 1,2,\ldots,T \quad \text{and} \quad 0 \le \alpha \le 1 \tag{16}$$

$$x_{ij}^k \in \{0,1\}, \quad \forall i,j = 1,\ldots,n, \quad \forall k = 1,2,\ldots,T \tag{17}$$

$$z_k \in \{0,1\}, \quad \forall k = 1,2,\ldots,T \tag{18}$$

In this problem $T^* = \{k / \sum_{i=1}^{n} q_i (\sum_{j=1}^{n} x_{ij}^k) > C_k\}$, which is the set of vehicles that exhibit capacity violations; for these vehicles, we employ parameters $\lambda_k \ge 0$, that determine a penalty included in the objective function. The values of $\lambda_k's$ are fixed and they are mathematically defined in Gheysens et al. (1984); for a thorough discussion on parameters $\lambda_k \ge 0$, the reader is referred to Section 3.2.

The objective function (3) models the trade-off between route and vehicle costs, and comprises the penalty for overloads. The first term of (3) reflects the cost of the routes followed by all vehicles after they depart from and before they return to the depot, as well as the cost of the first and last segment of each route. The second term of (3) reflects the total vehicle acquisition or set-up cost. The third term is the penalty associated with capacity violations, following the approach of Gheysens et al. (1984).

Constraints (4) and (5) ensure that exactly one vehicle enters and departs from every customer and from the depot. Constraint (6) guarantees that no customers are serviced by inactive vehicles, i.e., by vehicles with $z_k = 0$. Constraints (7) and (8) account for the availability of vehicles by bounding the number of arcs, related to each vehicle k, directly leaving from and returning to the depot, below one, respectively. Constraint set (9) is the typical flow conservation equation that ensures the continuity of each vehicle route. Constraints (10)–(14) are related to time windows and guarantee the feasibility of the schedule for each vehicle. In particular, constraints (10) and (11) ensure that, if customers i and j are consecutive in the schedule of vehicle k, then the arrival time at customer j equals the departure time from customer i, plus the travel time between these two customers; note that M is a large number. In case customers i and j are not serviced by the same vehicle or are not consecutive, constraints (10) and (11) are inactive. Constraints (12) and (13) guarantee that the relationships between arrival time, departure time, and service time with respect to customer i are compatible to the customer's time window. Constraint (14) sets the departure time from the depot equal to zero, since all routes originate at the depot. Constraint (15) calculates the total load of each vehicle and sets it equal to the relevant variable Q_k. Constraint (16) allows the vehicle capacity violation up to a percentage of the total capacity, which is determined by the scheduler through parameter α. Finally, constraints (17) and (18) enforce integrality for the x_{ij}^k and z_k variables, respectively.

3. Solution approach

To solve the problem modeled in Section 2, we propose a new heuristic method that is based on the insertion framework (Solomon, 1987), and includes two main components in the selection criteria of the non-routed customers: The first component is a variant of the criteria proposed by Golden et al. (1984), and also used by Dullaert et al. (2002), Paraskevopoulos et al. (2008), and Liu and Shen (1999). The second component is a penalty function brought into the problem's objective according to Gheysens et al. (1984) for the Heterogeneous VRP, appropriately modified to address the Heterogeneous VRPTW. The heuristic terminates with a complete HFVROTW solution that satisfies all problem constraints. The elements of the proposed solution method are examined in detail below.

3.1. Customer selection criteria

Let i and j be two consecutive customers in a partially constructed route r, and let u denote an unassigned customer. The function that measures the cost of inserting u between i and j is denoted by $c_1(i,u,j)$; it is a composite criterion merging via appropriate weights several sub-metrics and incorporates the overload penalties that will be discussed later in this section.

The first such metric is the coverage of the time window for the selected customer u (see Ioannou et al., 2001) denoted by $c_{ij}^0 = a_u - e_u$. Furthermore, $c_1(i,u,j)$ includes the time gap c_{ij}^1 between the latest service time l_u of customer u and the vehicle's arrival time at customer u, caused by the insertion of customer u between customers i and j; this metric expresses the compatibility of the time window of the selected customer with the specific insertion place in the current route (see also Ioannou et al., 2001 for clarifications) and is given by: $c_{ij}^1 = l_u - (a_u + s_u + d_{iu})$. In addition, the insertion of u between i and j results in a distance increases $c_{ij}^2 = d_{iu} + d_{uj} - d_{ij}$, a metric also included in $c_1(i,u,j)$.

Metric $c_{ij}^3 = |C_k - Q_k - q_u|$ models the deviation from the vehicle's capacity when customer u is inserted into the route; note that we employ the absolute value for this difference since overloads may cause negative expression of $c_{ij}{}^3$. Finally $c_{ij}{}^4$, measures the difference of the average fixed cost per unit transferred by vehicle k prior and after the insertion of customer u: $c_{ij}^4 = |(f_k/Q_k) - [f_k/(Q_k + q_u)]|$.

The vehicle saving insertion metric c_{ij}^5 is equal to one of the adapted savings concepts defined in Dullaert et al. (2002), i.e., c_{ij}^5=ACS, AOOS, AROS. The adapted combined savings (ACS) concept is defined as the difference between the fixed costs of the vehicle capable of transporting the load of the route after and before inserting customer u, $f(Q^{new}) - f(Q)$. To reflect the original notion of Golden et al.'s (1984) optimistic opportunity savings, the adapted optimistic opportunity savings (AOOS) concept extends the ACS by subtracting $f(Q^{-\,new} - Q^{new})$. This is the fixed cost of the smallest vehicle that can service the unused capacity $Q^{-\,new} - Q^{new}$. The adapted realistic opportunity savings (AROS) concept takes the fixed cost of the largest vehicle smaller than or equal to the unused capacity, $F'(Q^{-\,new} - Q^{new})$, into account as opportunity saving. It only does so if a larger vehicle is required to service the current tour after a new customer has been inserted. For a detailed discussion of the aforementioned metrics, refer to Dullaert et al. (2002).

Finally, the penalty component c_{ij}^6 enables the selection of vehicles for which the sum of their fixed cost and capacity penalty is minimized for each route; c_{ij}^6 is formally defined in the next subsection. Thus, $c_1(i,u,j)$ is given by the following relationship:

$$c_1(i,u,j) = \gamma_1(c_{ij}^0 + c_{ij}^1) + \gamma_2 c_{ij}^2 + \gamma_3 c_{ij}^3 + \gamma_4 c_{ij}^4 + \gamma_5 c_{ij}^5 + \gamma_6 c_{ij}^6 \tag{19}$$

In (19), the γ weights define the relative contribution of each individual metric to the overall selection criterion and $\gamma_1,\gamma_2,\gamma_3,\gamma_4,\gamma_5,\gamma_6 \ge 0$. To identify the best combination of these weights for the implementation of our algorithm, the following evaluation procedure was used:

a. For each γ, we have performed an iterative search with a step-increase of 0.1 (i.e., setting γ_1=0.1, 0.2, 0.3, …, 0.9, 1.0, then γ_2=0.1, 0.2, 0.3, …, 0.9, 1.0, and so on) while keeping all the other γ values constant and equal to 0.5 (to simulate some form of equal contribution of all these other γ's).

b. For a each of these iterations, we have calculated the value of the objective function and determined the best (minimum) one.
c. We examined the sensitivity of the problem solution to the changes in the γ-values. Our observations led to the conclusions below:

i. The value of the objective function is very sensitive to the changes in the value of parameters γ_2 and γ_6; thus we keep this level of step (0.1) in the actual implementation, leading to 10×10 total iterations.
ii. The value of the objective function is insensitive to the changes in the value of parameters γ_3 and γ_4; thus, we kept just two levels for these parameters (0.5 and 1.0), leading to 2×2 iterations.
iii. The value of the objective function is moderately sensitive to the changes in the value of parameters γ_1 and γ_5; thus we keep a level of step equal to 0.2 in the actual implementation, leading to 5×5 total iterations.

The above iterative scheme was performed in order to approximate some form of local search, as proposed by Dullaert et al. (2002), and increased the computational time of the proposed approach (since a total of 10,000 iterations were embedded within the algorithmic steps) in order to reach better quality solutions.

3.2. The penalty component

We solve the overloading heterogeneous vehicle routing problem with time windows using a penalty function. The idea is to allow violations in the capacity constraints of some vehicles but to penalize the resulting overloads. Fig. 1 illustrates how the penalty function works.

In Fig. 1, parameters y_{k-1} and y_k are the bounds associated with the allowable capacity violations. This means that if a vehicle of nominal capacity of C_k is loaded over this capacity and up to y_k, ($C_k \le Q_k \le y_k$), then this is allowed at a penalty, determined together with the vehicle's fixed cost by the relationship: $P(Q_k)=f_k+\lambda_k(Q_k-C_k)$; a similar definition holds for y_{k-1}. If the vehicle is loaded up to C_k, while exceeding the bound imposed by the vehicle of smaller capacity (i.e., $y_{k-1}\le Q_k \le C_k$), then the penalty is zero and the following holds: $P(Q_k)=f_k$. Note that the bounds y_k are determined by the equation: $f_k+\lambda_k(y_k-C_k)=f_{k+1}$ (see Gheysens et al., 1984, for further discussion). The values of λ_k are defined via the a simple parameter θ which ensures that $y_k < C_{k+1}$ for $k=1, 2, \ldots, T-1, T$ (the number of vehicle types). Setting $\theta=(y_k-C_k)/(C_{k+1}-C_k)$, then $\lambda_k=(1/\theta)(f_{k+1}-f_k)/(C_{k+1}-C_k)=(f_{k+1}-f_k)/(y_k-C_k)$ for $k=1, \ldots, T-1$.

The penalty component is, thus, defined as $c_{ij}^6=P(Q_k)$ for the active vehicle k for which the current route is constructed. Note that in the application of our algorithm, we further constraint the capacity violation area. This is achieved by employing a factoring metric α that scales the capacity of each vehicle, with α being from 0.1, 0.2 and up to satisfy the y_k relationship below.

$$y'_k = C_k+\alpha\times C_k, \quad \text{where} \quad y'_k \le y_k$$

In this way we are capable of examining multiple scenarios of potential capacity violations, while restricting to the bare minimum the vehicle overloads.

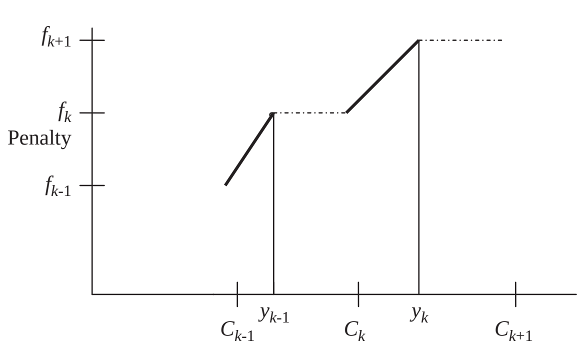

Fig. 1. The shape of the penalty function $P(Q)$.

3.3. Criterion for customer insertion into a route

In the second phase we can define the customer selection criterion, i.e., the customer that is best according to the selection criterion

$$c_2(i,u,j)=\max_u[c_2(i,u,j)],$$

where $c_2(i,u,j)=(\mu(d_{ou}+d_{uo})+s_u+F(q_u)-c_1(i,u,j)),\mu\ge 0$, with s_u the service time of customer u, μ a positive real number, and $F(q_u)$ the fixed cost of the smallest vehicle capable of moving a load q_u.

The above selection criterion ensures that a customer u selected for insertion into a route attempts to maximize the benefit derived from inserting this customer on the partial route being constructed, rather than commencing a new route.

3.4. Overall solution heuristic

To generate a feasible route for the HFVROTW problem, we use the steps of the algorithm presented below:

Algorithm OVER-L

Step 0: Initialization. Read n, T, C_k, f_k, $k=1,2,3\ldots.T$, c_{ij}, and e_i, l_i, $\forall i$, $j=2,\ldots, n$. Also $J=L$.
Step 1: Select a "seed" customer to start a route r, finding the farthest customer from the depot. If there does not exist a non-routed feasible customer to start a route, go to Step 6.
Step 2: Find the feasible non-routed customer u that minimizes the composite criterion $c_1(i,u,j)$, i.e.

> *Step* 2*a*: Examine all possible feasible insertions of customer u into the current route. For each feasible insertion, calculate $c_1(i,u,j)$. Select insertion location that results in minimum $c_1(i,u,j)$ for this customer.
> *Step* 2*b*: Repeat Step 2a for all feasible non-routed customers.
> *Step* 2*c*: Select customer u with maximum $c_2(i,u,j)$.

Step 3: Insert the selected customer u, to the best insertion location on the current route r (see Steps 2a and 2c). Update the route and set u as a routed customer, decreasing set J.
Step 4: If there are non-routed customers that are feasible for insertion into the current route r, return to Step 2; otherwise proceed to Step 5.
Step 5: If all customers have been scheduled, go to Step 6. Otherwise, go to Step 1—initiate new route.
Step 6: Terminate; output number of routes (active vehicles), sequence of customers visited by each vehicle, total distance (time) and total cost.

The above algorithm terminates with a complete solution to the HFVROTW.

4. Illustrative example

To illustrate the mechanics of OVER-*L* and decipher the ideas behind capacity violations, we provide an example of its application on benchmark data set R101 of Solomon (1987). This data set includes 100 customers, and the best solution reached by heuristics for the heterogeneous vehicle fleet version comprises 24 vehicles (one of type A, eleven of type B, eleven of type C, one of type D, and none of type E, denoted by A^1,B^{11},C^{11},D^1,E^0) and has a total cost of 5061 units (see Paraskevopoulos et al., 2008). The vehicle types A, B, C, D and E have capacities 30, 50, 80, 120 and 200 units, and fixed costs 50, 80, 140, 250, 500 units, respectively. Setting $\theta=0.5$, the penalty function of Section 3 gives the following values of λ:

$$\lambda_A = \frac{80-50}{0.5\times(50-30)} = \frac{30}{0.5\times 20} = \frac{3}{1} = 3$$
$$\lambda_B = \frac{140-80}{0.5\times(80-50)} = \frac{60}{0.5\times 30} = 4$$
$$\lambda_C = \frac{250-140}{0.5\times(120-80)} = \frac{110}{0.5\times(40)} = 5.5$$
$$\lambda_D = \frac{500-250}{0.5\times(200-120)} = \frac{250}{0.5\times 80} = \frac{25}{4} = 6.25$$
$$\lambda_E = \lambda_D$$

Applying OVER-*L* to the data, we reach the solution shown in Table 1, with 20 active vehicles of types A^3, B^4, C^{10},D^3,E^0. Table 1 shows the vehicle fixed cost in the solution where capacity violations are allowed, and also provides the respective fixed costs if violations are eliminated by activating larger capacity vehicles. It also provides the total load carried by each vehicle serving the route, as well as the fixed cost incorporating the penalty component. Note that the results of Table 1 have been obtained without using the α parameter of the customer insertion rule; we rather rely directly on the data of Gheysens et al., (1984)

Table 1
OVER-*L* results for R101.

	Total cargo	Fixed cost plus penalty	Fixed cost with violation	Fixed cost without violation
1	78	140.00	140	140
2	86	140+5.5(86−80)=173.00	140	250
3	112	250.00	250	250
4	79	140.00	140	140
5	83	140+5.5(83−80)=156.50	140	250
6	75	140.00	140	140
7	115	250.00	250	250
8	75	140.00	140	140
9	59	80+4(59−50)=116.00	80	140
10	75	140.00	140	140
11	81	140+5.5(81−80)=145.50	140	250
12	82	140+5.5(82−80)=151.00	140	250
13	123	250+6.25(123−120)=268.75	250	500
14	88	140+5.5(88−80)=184.00	140	250
15	39	50+3(39−30)=77.00	50	80
16	57	80+4(57−50)=108.00	80	140
17	40	80.00	80	140
18	50	80.00	80	140
19	34	50+3(34−30)=62.00	50	80
20	27	50.00	50	50
		Total cost	2620	3720

Table 3
Results for C101.

	Customers	Distance	Waiting time	Vehicle cost	Route cost
1	82,79,77,72,71	120.96	0	300	420.96
2	91,88,87,84,83,74,78,80,81	127.71	0	300	427.71
3	56,64,54,57,59,61,60,51,53,50,48	163.26	0	800	963.26
4	99,97,96,95,93,85	103.62	0	300	403.62
5	6,4,8,9,11,94	103.09	0	300	403.09
6	21,34,36,38,39	89.06	71.17	300	460.23
7	19,20,16,17	90.67	15	300	405.67
8	33,32,30,40,37,35	115.65	224.73	300	640.38
9	14,18,63,75,12,98,101,100,2,76	201.62	9.20	800	1010.82
10	58,55,41,45,46	88.94	114.43	300	503.37
11	44,25,26,28,31,15,13	112.94	47.20	300	460.14
12	68,66,73,86,89,90,92	89.72	245.61	300	635.33
13	43,42,47,62,49,52,23	105.78	193	300	598.78
14	29,65,69,67,70	79.27	0	300	379.27
15	10,7,5,3,22	61.81	12.92	300	374.73
16	27,24	31.81	17	300	348.81
	Total	**1685.89**	**950.26**	**5800**	**8436.15**

Table 2
Additional OVER-*L* results for R101.

	Customers	Overload	Distance	Waiting time	Vehicle cost	Route cost
1	66,72,10,35,36,71	0 (78)	138.46	13.57	140	292.03
2	64,65,50,33,78	6 (86)	142.42	25.92	140	308.34
3	40,24,68,56,5,26	0 (112)	125.10	22.96	250	398.06
4	15,45,39,44,92,101	0 (79)	111.69	47.35	140	299.04
5	37,48,20,9,47,18,94	3 (83)	136.64	11.61	140	288.25
6	63,12,91,67,2,81	0 (75)	129.31	0.79	140	270.10
7	93,43,99,17,87,38,61,90,59	0 (115)	116.27	17.30	250	383.57
8	34,30,52,21,49	0 (75)	131.96	14.14	140	286.10
9	28,70,79,4,55,25	9 (59)	102.14	7.99	80	190.13
10	60,96,16,42,57,75	0 (75)	97.37	26.03	140	263.40
11	46,84,62,85,97,14	1 (81)	88.57	28.76	140	257.33
12	73,76,74,23,98	2 (82)	82.77	18.60	140	241.37
13	32,31,82,51,69	3 (123)	89.71	5.17	250	344.88
14	29,13,77,80,27	8 (88)	65.43	5.07	140	210.50
15	53,89,8,11	9 (39)	66.33	12.47	50	128.80
16	6,83,19,7	7 (57)	68.58	7.40	80	155.98
17	3,88,58	0 (40)	55.48	25.93	80	161.41
18	100,86	0 (50)	46.17	0.00	80	126.17
19	22,41,54	4 (34)	36.28	5.93	50	92.21
20	95	0 (27)	24.08	0.00	50	74.08
		Total	**1854.76**	**296.99**	**2620**	**4771.75**

Table 2 provides additional data concerning the output of OVER-*L*, i.e., the subset of customers and the level of overload of each route (total load in parenthesis), the total route distance, the overall waiting time of each route, the vehicle fixed cost and the total route cost; the latter includes the vehicle fixed cost, the route distance that is equivalent to route time, and the waiting time at each customer except the first. From Table 2, it is evident that small violations of the vehicle capacities result in significantly lower fixed vehicle costs, a fact that supports our claim that the HFVROTW is a worth studying problem. The total cost of the solution determined by OVER-*L* is 4771.75, as shown in Table 2. This value is lower than the best solution found in the literature for the case where violations are not allowed, even by complex and intelligent meta-heuristics (see Paraskevopoulos et al., 2008). Thus, the controlled violations of the capacity of vehicles results in reduced total costs for routing and scheduling problems with minimal capacity violations.

The results of Table 3, referring to data set C101 of Solomon (1987), further reinforce and confirm that even for very restrictive problems such as C101, where time windows in combination with service time play the most critical role in determining feasible solutions, the proposed approach is superior to other methods. Specifically, for data set C101, OVER-*L* derives a solution with 16 vehicles and a total cost of 8436.15; this solution is almost 10% better (lower) than the one derived by Paraskevopoulos et al. (2008), which has a cost of 9272. This is directly attributed to the small capacity violations that can reduce significantly the fixed vehicle costs.

Note that the results of Table 3 are obtained using a α parameter equal to 0.1 (α=0.1), while one could expect much larger cost reductions if the HFVROTW was solved via complex meta-heuristics.

5. Computational results

We have executed numerical experiment of the proposed algorithm on the benchmark data sets of Solomon (1987)—100 customer instances R, C, and RC proposed for the original VRP with time windows. All experiments were performed on a PC equipped with a Pentium Dual Core T4200 running at 2.0 GHz. We compare our heuristic's performance to Liu and Shen (1999). As we have already mentioned the objective is to achieve cost reductions with minimal and controlled violations in the capacity of vehicles. Liu and Shen (1999) use the distance travelled, the waiting time (excluding the waiting time for the first customer) and the vehicle costs to measure the solution quality. Our approach reduces the total delivery cost as defined by Liu and Shen (1999), allowing a small percentage of capacity overloads.

Table 4 illustrates the performance of OVER-*L* vis-à-vis the Liu and Shen (1999) approach for the HFVROTW problem, in terms of the total cost. The results show that this metric is substantially improved by OVER-*L*. Note that for all the data sets reported in

Table 4
OVER-*L* vis-à-vis Liu and Shen's heuristic.

Instance	Liu and Shen	OVER-*L*	(%) DEV
R101	5061.00	4687.00	7.38
R102	5013.00	4655.00	7.14
R103	4772.00	4484.00	6.03
R104	4455.00	4165.00	6.50
C101	9272.00	7286.00	21.41
C102	8433.00	7553.00	10.43
C103	8033.00	7722.00	3.87
RC101	5687.00	5176.00	8.98
RC102	5649.00	5175.00	8.39
RC103	5419.00	5009.00	7.56

Table 5
Results for R1 data set.

Instance	Vehicles	Distance	Cost w/o violation	Cost with violation	FC w/o violation	FC with violation	% improvement on FC cost	% capacity violation	CPU times (sec)
R101	$A^3B^4C^{13}$	1920.84	5667.30	4687.30	3470	2490	28.24	5.8 (86)	67.39
R102	$A^4B^3C^5D^5E^1$	1793.33	6545.89	4655.89	4529	2640	41.70	7.8 (114)	108.52
R103	$A^1B^1C^4D^8$	1613.73	6204.38	4484.37	4410	2690	39.00	7.9 (116)	160.14
R104	$B^1C^2D^9$	1372.01	6695.56	4165.56	5140	2610	49.22	11.5 (168)	253.63
R105	$B^2C^8D^5$	1659.43	5917.94	4247.94	200	2530	39.76	10.4 (153)	93.02
R106	$A^2B^1C^3D^8$	1561.11	6590.48	4200.48	4990	2600	47.89	10.6 (156)	140.65
R107	$A^2C^2D^9$	1426.07	6541.17	4151.17	5020	2630	47.60	11.3 (166)	193.47
R108	$A^1C^1D^{10}$	1343.42	6442.20	4082.19	5050	2690	46.73	10.6 (155)	252.48
R109	DC^4D^8	1471.74	6741.99	4141.99	4970	2640	46.88	9.1 (133)	136.07
R110	$A^1C^4D^8$	1498.86	6320.64	4130.64	4800	2610	45.62	10.4 (153)	183.24
R111	$B^1C^1D^{10}$	1483.42	6577.69	4107.69	5080	2610	48.62	12.2 (176)	173.60
R112	B^1 C^2 D^9	1336.47	6493.47	3963.47	5140	2610	49.22	11.5 (168)	277.17

Table 6
Results for C1 data set.

Instance	Vehicles	Distance	Cost w/o violation	Cost with violation	FC w/o violation	FC with violation	% improvement on FC cost	% capacity violation	CPU times (sec)
C101	A^8B^4	1338.81	12986.65	7286.64	11300	5600	50.44	13.8 (250)	114.65
C102	$A^{14}B^1$	1902.34	14603.65	7553.65	12050	5000	58.50	15.4 (280)	158.52
C103	A^8B^4	1461.12	12872.72	7722.71	10750	5600	47.90	12.1 (220)	219.15
C104	A^6B^5	1313.58	13067.57	7315.52	11550	5800	50.21	11.6 (210)	278.56
C105	$A^{10}B^3$	1349.31	12743.56	7093.56	11050	5400	51.13	14.9 (270)	121.05
C106	$A^{12}B^2$	1557.71	13513.56	7413.55	11300	5200	53.98	14.9 (270)	136.43
C107	A^8B^4	1411.74	12767.81	7067.80	11300	5600	50.44	16.0 (290)	133.70
C108	A^8B^4	1348.95	12743.15	7043.14	11300	5600	50.44	16.0 (290)	169.65
C109	A^6B^5	1232.74	12282.74	7032.74	11050	5800	47.51	14.9 (270)	234.83

Table 7
Results for RC1 data set.

Instance	Vehicles	Distance	Cost w/o violation	Cost with violation	FC w/o violation	FC with violation	% improvement on FC cost	% capacity violation	CPU times (sec)
RC101	$A^3B^{10}C^3D^1$	2047.47	7156.64	5176.63	5010	3030	65.34	9.8 (170)	77.42
RC102	$A^3B^5C^6D^1$	1890.71	6465.98	5175.98	4470	3180	40.56	7.6 (132)	109.87
RC103	$A^4B^3C^7D^1$	1696.47	6449.03	5009.03	4680	3240	44.44	6.3 (110)	144.52
RC104	$A^1B^8C^2D^2$	1786.13	5730.82	4590.82	3900	2760	41.30	4.8 (84)	193.91
RC105	$A^3B^9C^4D^1$	1904.42	7077.65	5187.65	5070	3180	59.43	9.5 (164)	133.33
RC106	$A^1B^8C^6$	1808.04	6831.00	4941.00	4949	3060	61.76	10.2 (177)	108.03
RC107	$A^1B^7C^5D^1$	1821.07	6231.43	4881.43	4410	3060	44.11	5.6 (98)	141.35
RC108	$A^1B^6C^5D^1$	1678.23	5646.16	4596.15	3960	2910	36.08	4.0 (70)	179.19

Table 4, the maximum violation is less than 10% of the nominal capacity in the majority of the cases, while the parameters of our algorithm were set to $\theta=0.5$ (to follow Gheysens et al., 1984) and $\alpha=0.2$. The results for all R1, C1 and RC1 data set are summarized in Tables 5–7. Specifically, we present the route distance, the cost without violations and the cost with violations, the fixed cost (FC) without violations, the fixed cost (FC) with violation, the percentage improvement in the fixed and total cost due to the capacity violations, and the CPU times.

From the results of Tables 5–7, we can securely infer that there is indeed cost reduction in all routing instances with capacity violations that are quite insignificant. This is exactly the goal of the proposed approach and the fact that it can be demonstrated in most Solomon's data sets, which were not developed for testing such problems, is very encouraging. Furthermore, the CPU times are much shorter from those of the metaheuristic of Braysy et al. (2008). The latter is demonstrated in the aggregate data concerning CPU times that Table 8 provides. Note that we report only aggregate values since Liu and Shen (1999) report only such CPU-time results; in addition, for fair comparison, we include in parentheses the CPU's used to run the experiments for each method. Thus, when comparing our CPU times to Liu and Shen (1999), one must consider the fact that the computer used by Liu and Shen (1999) is slower by more than one order of magnitude to the one used for our tests.

Finally, Table 9 extends the results of Table 4 for all Solomon's data sets included in our computational experiments. The values for *a*–*c* are adopted from Liu and Shen (1999) and are given in Table 10; these values refer to cost and capacity equivalences. From the results of Table 9, one can see that our approach improves the total cost from 2% (for data set RC1c) to almost 10% (for data set C1b) by small capacity violations. This is very important since it has been demonstrated for a very large set of data (36 data sets for R1, 27 data sets for C1, and 24 data sets for RC1).

Finally, to provide a vis-à-vis list of results from metaheuristics and OVER-*L*, we include in Table 11 the total cost of OVER-*L* and the total cost of the best metaheuristic solution as reported by Repoussis and Tarantilis (2010) for the data set R1 of Solomon (1987). The results show that this metric is improved by OVER-*L*, as was the case for the Liu and Shen heuristic. Note that for all the data sets reported in Table 11, the maximum violation is less than 15% of the nominal capacity (a fact that justifies the differences with Table 4).

Table 8
CPU time comparisons (aggregate)—times in sec.

Data set	Liu and Shen (Pentium II, 233 MHz)	Braysy et al. (AMD Athlon 2600+, 2.13 GHz)	OVER-*L* (Pentium Dual Core T4200, 2.0 GHz)
R1 (36)	175.83	787.99	152.85
C1 (27)	144.00	641.28	148.80
RC1 (24)	155.67	536.66	128.76

Table 9
OVER-*L* vis-à-vis Liu and Sen's heuristic.

Data set	**Liu and Sen**	**OVER-*L***	**(%) DEV**
R1a(12)	4562.00	4251.00	6.81
R1b(12)	2149.00	2080.00	3.32
R1c(12)	1788.00	1741.00	2.62
R1(36)	2833.00	2690.00	5.04
C1a(9)	8042.00	7281.00	9.50
C1b(9)	2626.00	2544.00	3.12
C1c(9)	1870.00	1828.00	2.21
C1(27)	4179.33	3884.33	7.05
RC1a(8)	5184.00	4944.00	4.62
RC1b(8)	2235.00	2187.00	2.15
RC1c(8)	1849.00	1813.00	1.95
RC1(24)	3089.33	2981.33	3.49

Table 10
Liu and Shen (1999) heterogeneous vehicle fleet characteristics for the HFVROTW (subclasses a, b, c for R1, C1, RC1 data set).

Vehicles	Capacity	R1a	R1b	R1c
A	30	50	10	5
B	50	80	16	8
C	80	140	28	14
D	120	250	50	25
E	200	500	100	50
Vehicles	Capacity	C1a	C1b	C1c
A	100	300	60	30
B	200	800	160	80
C	3009	1350	270	135
Vehicles	Capacity	RC1a	RC1b	RC1c
A	40	60	12	6
B	80	150	30	15
C	150	300	60	30
D	200	450	90	45

6. Conclusions

In this paper, we addressed for the first time the vehicle routing problem with time windows when the fleet is heterogeneous, i.e., comprises vehicles of different capacities and associated costs, and overloads are allowed up to a pre-specified bound, at a penalty embedded in the problem's objective function. For the solution of the HFVROTW we developed a sequential insertion heuristic, extending the traditional insertion criteria of Solomon (1987), and adapting

Table 11
OVER-*L* vis-à-vis best metaheuristic for HFVRTW.

Data set	Best metaheuristic solution	OVER-*L*
R101	4536.40	4242.51
R102	4348.92	4294.81
R103	4119.04	4018.14
R104	3981.28	3831.11
R105	4229.67	3815.21
R106	4118.43	3793.39
R107	4031.16	3831.93
R108	3962.20	3695.75
R109	4052.21	3793.19
R110	3995.18	3762.34
R111	4016.19	3722.37
R112	3947.30	3689.69

Golden et al.'s (1984), Dullaert et al.'s (2002), and Liu and Shen's (1999) ones. The computational results on benchmark problems showed that competitive results are obtained for problems examined and this reinforced our intuition for practical applicability of the proposed approach, with minimal adverse effects on vehicle loads and positive impact on total costs. Future research directions include the development of metaheuristics for the relaxed HVRPTW, hybridization of clever heuristics with complex search methods, examination of non-linear penalty functions, and the study of the effect of very small capacity violations (white noise) to the overall solution quality of the problem.

Acknowledgements

The authors are indebted to the two anonymous reviewers for their useful comments and suggestions that have helped us improve the original manuscript.

References

Brandao, J., Mercer, A., 1997. A tabu-search algorithm for the multi-trip vehicle routing and scheduling problem. European Journal of Operational Research 100, 180–191.
Braysy, O., Dullaert, W., Hasle, G., Mester, D., Gendreau, M., 2008. An effective multi-restart deterministic annealing metaheuristic for the fleet size and mix vehicle routing problem with time windows. Transportation Science 42 (3), 371–386.
Braysy, O., Porkka, P.P., Dullaert, W., Repoussis, P.P., Tarantilis, C.D., 2009. A well scalable metaheuristic for the fleet size and mix vehicle routing problem with time windows. Expert System and Applications 36 (4), 8460–8475.
Dondo, R., Cerda, J., 2007. A cluster-based optimization approach for the multi-depot heterogeneous fleet vehicle routing problem with time windows. European Journal of Operational Research 176, 1478–1507.
Dullaert, W., Janssens, G.K., Sorensen, K., Vernimmen, B., 2002. New heuristics for the fleet size and mix vehicle routing problem with time windows. Journal of the Operational Research Society 53, 1232–1238.
Gendreau, M., Tarantilis, C.D., 2010. Solving large-Scale Vehicle Routing Problems with Time Windows: The State-of-the-Art. CIRRELT-2010-04. Université de Montreal, Canada.
Gheysens, F., Golden, B., Assad, A., 1984. A comparison of techniques for solving the fleet size and mix vehicle routing problem. OR Spektrum 6, 207–216.
Golden, B., Assad, A., Levy, L., Gheysens, F., 1984. The fleet size and mix vehicle routing problem. Computers and Operation Research 11 (1), 49–66.
Golden, B., Raghavan, S., Wasil, E., 2008. The Vehicle Routing Problem: Latest Advances and New Challenges. Springer, NY.
Ioannou, G., Kritikos, M., Prastacos, G., 2001. A greedy look-ahead heuristic for the vehicle routing problem with time windows. Journal of the Operational Research Society 52, 523–537.
Kritikos, M., Ioannou, G., 2010. The balanced cargo vehicle routing problem with time windows. The International Journal of Production Economics 123 (1), 42–51.
Liu, F.-H., Shen, S.-Y., 1999. The fleet size and mix vehicle routing problem with time windows. Journal of the Operational Research Society 50, 721–732.
Paraskevopoulos, D.C., Repoussis, P.P., Tarantilis, C.D., Ioannou, G., Prastacos, G.P., 2008. A reactive variable neighbourhood tabu search for the heterogeneous fleet vehicle routing problem with time windows. Journal of Heuristics 14 (5) 425–255.
Repoussis, P.P., Tarantilis, C.D., 2010. Solving the fleet size and mix vehicle routing problem with time windows via adaptive memory programming. Transportation Research Part C 18 (5), 695–712.
Rochat, Y., Semet, F., 1994. A tabu-search approach for delivering pet food and flour in Switzerland. Journal of the Operational Research Society 45, 1233–1246.
Solomon, M.M., 1987. Algorithms for the vehicle routing and scheduling problems with time windows constrains. Operations Research 35, 254–265.