Math. Program., Ser. A
DOI 10.1007/s10107-008-0218-9

FULL LENGTH PAPER

A unified exact method for solving different classes of vehicle routing problems

Roberto Baldacci · Aristide Mingozzi

Received: 29 September 2007 / Accepted: 12 March 2008

Abstract This paper presents a unified exact method for solving an extended model of the well-known Capacitated Vehicle Routing Problem (CVRP), called the Heterogenous Vehicle Routing Problem (HVRP), where a mixed fleet of vehicles having different capacities, routing and fixed costs is used to supply a set of customers. The HVRP model considered in this paper contains as special cases: the Single Depot CVRP, all variants of the HVRP presented in the literature, the Site-Dependent Vehicle Routing Problem (SDVRP) and the Multi-Depot Vehicle Routing Problem (MDVRP). This paper presents an exact algorithm for the HVRP based on the set partitioning formulation. The exact algorithm uses three types of bounding procedures based on the LP-relaxation and on the Lagrangean relaxation of the mathematical formulation. The bounding procedures allow to reduce the number of variables of the formulation so that the resulting problem can be solved by an integer linear programming solver. Extensive computational results over the main instances from the literature of the different variants of HVRPs, SDVRP and MDVRP show that the proposed lower bound is superior to the ones presented in the literature and that the exact algorithm can solve, for the first time ever, several test instances of all problem types considered.

Keywords Vehicle routing · Set partitioning · Dual ascent · Dynamic programming

Mathematics Subject Classification (2000) 90C27 · 49M29 · 90C39

R. Baldacci (✉)
DEIS, University of Bologna, Via Venezia 52, 47023 Cesena, Italy
e-mail: r.baldacci@unibo.it

A. Mingozzi
Department of Mathematics, University of Bologna, Via Sacchi 3, 47023 Cesena, Italy
e-mail: mingozzi@csr.unibo.it

1 Introduction

The Heterogenous Vehicle Routing Problem (HVRP) considered in this paper has been introduced by Li et al. [19] and is a generalization of the Capacitated Vehicle Routing Problem (CVRP). The HVRP can be described as follows.

An undirected graph $G = (V', E)$ is given, where $V' = \{0, 1, \dots, n\}$ is the set of $n + 1$ nodes and E is the set of edges. Node 0 represents the depot, while the remaining node set $V = V' \backslash \{0\}$ corresponds to n customers. Each customer $i \in V$ requires a supply of q_i units from the depot (we assume $q_0 = 0$). A heterogeneous fleet of vehicles is stationed at the depot and is used to supply the customers. The vehicle fleet is composed of a set $M = \{1, \dots, m\}$ of m different vehicle types. For each type $k \in M$, U_k vehicles are available at the depot, each having a *capacity* equal to Q_k. With each vehicle type is also associated a *fixed cost* F_k modelling, e.g., rental or capital amortization costs. In addition, for each edge $\{i, j\} \in E$ and for each vehicle type $k \in M$, a *routing cost* d^k_{ij} is given.

A route $R = (0, i_1, \dots, i_r, 0)$ performed by a vehicle of type k, is a simple cycle in G passing through the depot and customers $\{i_1, \dots, i_r\} \subseteq V$, with $r \geq 1$, such that the total demand of the customers visited does not exceed the vehicle capacity Q_k (i.e., $\sum_{h=1}^{r} i_h \leq Q_k$). Note that if $r = 1$ then route R represents the single-customer route $R = (0, i_1, 0)$. The cost of a route is equal to the sum of the routing costs and the fixed cost of the associated vehicle. The HVRP consists of designing a set of feasible routes of minimum total cost such that each customer is visited by exactly one route and the number of routes performed by the vehicles of type k is not greater than U_k, $k \in M$.

This model subsumes the following different classes of vehicle routing problems.

(1) The well-known CVRP, where a given fleet of p vehicles of identical capacity Q must service customers with known demands from a central depot at minimum routing cost. The CVRP corresponds to the HVRP where $m = 1$, $Q_1 = Q$, $F_1 = 0$ and $U_1 = p$.

(2) The fleet size and mix CVRP with fixed vehicle costs, unlimited number of vehicles (i.e., $U_k = n, \forall k \in M$) and independent routing costs (i.e., $d^r_{ij} = d^s_{ij}, \forall r, s \in M, r \neq s$), called FSMF.

(3) The fleet size and mix CVRP with fixed vehicle costs, unlimited number of vehicles (i.e., $U_k = n, \forall k \in M$) and vehicle dependent routing costs, called FSMFD.

(4) The heterogeneous CVRP with no fixed vehicle costs (i.e., $F_k = 0, \forall k \in M$) and vehicle dependent routing costs, called HD.

(5) The fleet size and mix CVRP with no fixed vehicle costs (i.e., $F_k = 0, \forall k \in M$), unlimited number of vehicles (i.e., $U_k = n, \forall k \in M$) and vehicle dependent routing costs, called FSMD.

(6) The Site-Dependent CVRP, called SDVRP. In the SDVRP, introduced by Nag et al. [22], a customer $i \in V$ can only be served by a subset of vehicle types $M_i \subseteq M$. The routing costs are vehicle independent and represented by a symmetric edge cost matrix $[d_{ij}]$ and no fixed costs are associated to the vehicles. Any SDVRP instance can be converted into an equivalent HD instance by setting for

Table 1 Characteristics of the different problems

Problem	Vehicle fixed costs	Vehicle dependent routing costs	Heterogenous vehicle fleet	Limited fleet
HVRP	Yes	Yes	Yes	Yes
CVRP	No	No	No	Yes
FSMF	Yes	No	Yes	No
FSMFD	Yes	Yes	Yes	No
HD / SDVRP	No	Yes	Yes	Yes
FSMD	No	Yes	Yes	No
MDVRP	No	Yes	No	No

each vehicle type $k \in M$:

$$F_k = 0 \text{ and } d_{ij}^k = \begin{cases} d_{ij}, & \text{if } k \in M_i \cap M_j \\ \infty, & \text{otherwise} \end{cases}, \ \forall \{i, j\} \in E,$$

where $M_0 = M$.

(7) The Multi-Depot CVRP, called MDVRP. This problem is an extension of the CVRP where a customer can be served by an unlimited fleet of identical vehicles of capacity Q, located at p depots, and inter-depot routes are not allowed (see Cordeau et al. [8]). Let $[\hat{d}_{ij}]$ be a $(n + p) \times (n + p)$ symmetric cost matrix, where $\hat{d}_{n+k\ i}$ is the travel cost for going from depot $k = 1, \ldots, p$ to customer $i \in V$. Any MDVRP can be converted into an equivalent HVRP instance generating $m = p$ different vehicle types and setting for each vehicle type $k \in M$:

$$Q_k = Q, \ U_k = n, \ F_k = 0 \quad \text{and} \quad d_{ij}^k = \begin{cases} \hat{d}_{n+k\ j}, & \text{if } i = 0 \\ \hat{d}_{ij}, & \text{otherwise} \end{cases}, \ \forall \{i, j\} \in E.$$

The main characteristics of the presented problems are summarized in Table 1. All problems described above are $\mathcal{NP}$-hard as they are natural generalizations of the CVRP.

1.1 Literature review

Many different heuristics are proposed in the literature for the HVRP and its variants. Among the various surveys on heuristic algorithms for the CVRP, we mention the surveys of Laporte and Semet [18] and of Gendreau et al. [14] in the book edited by Toth and Vigo [27] and the more recent update by Cordeau et al. [10], whereas a specific survey on heterogeneous vehicle routing problems can be found in Baldacci et al. [1].

The book edited by Toth and Vigo [27] surveys the most effective exact methods for the CVRP proposed in the literature up to 2002. A recent survey of the CVRP covering exact algorithms can also be found in the chapter of Cordeau et al. [10]. The most promising exact algorithms for the CVRP are due to Baldacci et al. [4], Lysgaard et al. [21], Fukasawa et al. [13] and Baldacci et al. [3]. Concerning the other variants, exact algorithms have been proposed in the literature only for the FSMF and the MDVRP.

The survey of Baldacci et al. [1] also covers lower bounds for the HVRP and its variants. Lower bounds for the FSMF were proposed by Golden et al. [15], Yaman [28] and Choi and Tcha [6]. The lower bound of Golden et al. [15] is a combinatorial lower bound based on the computation of shortest paths on particular layered networks. Yaman [28] proposed several lower bounds based on cutting-plane techniques used to strengthen the LP-relaxation of different mathematical formulations. Choi and Tcha [6] proposed lower bounds for the FSMF based on the Set Partitioning (SP) formulation, that were computed by using q-route relaxation and column generation techniques. These latter authors also described lower bounds for the FSMFD and the FSMD.

The only exact method for the FSMF was recently proposed by Pessoa et al. [24]. These authors extend to the FSMF the branch-and-cut-and-price method proposed for the CVRP by Fukasawa et al. [13]. The computational results show that their lower bounds dominate the lower bounds proposed by Golden et al. [15], Yaman [28] and Choi and Tcha [6], and that instances involving up to 75 vertices can be solved to optimality.

To our knowledge, only two exact algorithms have been proposed for the MDVRP. Laporte et al. [16,17] have developed exact branch-and-bound algorithms, but these only work well on relatively small instances (see Crevier et al. [12]).

1.2 Contributions of the paper

In this paper, we present an exact algorithm for the HVRP that generalizes the bounding procedures and the exact method recently proposed by Baldacci et al. [2] for the asymmetric VRP on a multi-graph and by Baldacci et al. [3] for the CVRP. Moreover, it introduces new bounding methods that are particularly effective when the vehicle fixed cost contribution to the total cost is relevant. Also in this paper, we describe reduction rules to resize the vehicle fleet.

The main contribution of this paper is a unified exact algorithm for the HVRP that can solve all different classes of vehicle routing problems shown in Table 1. For the different problem types considered, we report extensive computational results on test instances from the literature and on newly generated test instances. The computational results show that the proposed lower bound is superior to the lower bounds presented in the literature. Moreover, the exact algorithm outperforms the exact method of Pessoa et al. [24] and can solve, for the first time ever, several test instances of all problem types considered.

The paper is organized as follows. In Sect. 2 we describe a mathematical formulation of the HVRP and three relaxations that are used to derive valid lower bounds. Section 3 presents the exact algorithm and in Sects. 4, 5 and 6 we describe the bounding procedures based on the three relaxations given in Sect. 2. The overall bounding method that combines the different bounding procedures is also described in Sect. 2.

In Sect. 7 we report the computational results on test problems taken from the literature. Finally, Sect. 8 contains concluding remarks.

2 Mathematical formulation and its relaxations

Let $\mathscr{R}^k$ be the index set of all feasible routes of vehicle type $k \in M$ and let $\mathscr{R} = \bigcup_{k \in M} \mathscr{R}^k$. With each route $\ell \in \mathscr{R}^k$ is associated a routing cost c^k_ℓ. Let $\mathscr{R}^k_i \subset \mathscr{R}^k$ be the index subset of the routes of a vehicle of type k covering customer $i \in V$. In the following we use R^k_ℓ to indicate the subset of customers visited by route $\ell \in \mathscr{R}^k$.

Let x^k_ℓ be a binary variable that is equal to 1 if and only if route $\ell \in \mathscr{R}^k$ is chosen in the solution. The mathematical formulation of the HVRP, called F, is as follows:

$$(F) \quad z(F) = \min \sum_{k \in M} \sum_{\ell \in \mathscr{R}^k} (F_k + c^k_\ell) x^k_\ell \tag{1}$$

$$\text{s.t.} \quad \sum_{k \in M} \sum_{\ell \in \mathscr{R}^k_i} x^k_\ell = 1, \quad \forall i \in V, \tag{2}$$

$$\sum_{\ell \in \mathscr{R}^k} x^k_\ell \le U_k, \quad \forall k \in M, \tag{3}$$

$$x^k_\ell \in \{0, 1\}, \quad \forall \ell \in \mathscr{R}^k, \forall k \in M. \tag{4}$$

Constraints (2) specify that each customer $i \in V$ must be covered exactly by one route. Constraints (3) impose the upper bound on the number of vehicles of each type that can be used.

In the following, we describe three relaxations of problem F that are used to derive different lower bounds as well as different methods for reducing the size of sets $\mathscr{R}^k$, $k \in M$, by eliminating those routes that cannot belong to any optimal HVRP solution.

2.1 Relaxation LF

Let LF be the LP-relaxation of problem F and let $z(LF)$ be its optimal solution cost. We denote by DLF the dual of problem LF. Let $\mathbf{u} = (u_1, \ldots, u_n)$ and $\mathbf{v} = (v_1, \ldots, v_m)$ be the dual variable vectors associated with constraints (2) and (3), respectively.

$$(DLF) \quad z(DLF) = \max \sum_{i \in V} u_i + \sum_{k \in M} U_k v_k \tag{5}$$

$$\text{s.t.} \quad \sum_{i \in R^k_\ell} u_i + v_k \le c^k_\ell + F_k, \quad \forall \ell \in \mathscr{R}^k, \forall k \in M, \tag{6}$$

$$u_i \in \mathbb{R}, \quad \forall i \in V, \tag{7}$$

$$v_k \le 0, \quad \forall k \in M. \tag{8}$$

Both problem LF and DLF are impractical to solve, even for moderate size HVRP, because the number of variables and constraints are typically exponential. However, it is quite easy to find near optimal solution of DLF without generating constraints (6) and any feasible DLF solution provides a valid lower bound on the HVRP.

In Sect. 4, we describe two dual ascent heuristics for solving DLF, called H^1 and H^2. H^1 is based on the q-route relaxation of the HVRP and is an extension of the method proposed for the CVRP by Christofides et al. [7]. H^1 produces a DLF solution $(\mathbf{u}^1, \mathbf{v}^1)$ of cost $LH1$. H^2 is a column generation method that is executed after H^1 and uses the dual solution achieved by H^1 to generate the initial master problem. It differs from standard column generation methods as it uses Lagrangean relaxation and subgradient optimization to solve the master problem. H^2 produces a DLF solution $(\mathbf{u}^2, \mathbf{v}^2)$ of cost $LH2$.

It is easy to show that $LH1 \leq LH2 \leq z(LF)$.

2.2 Relaxation RP

One relaxation of the HVRP was presented in Sect. 2.1. A second relaxation of the HVRP corresponds to an integer problem, called RP, that can provide a better lower bound than $z(LF)$ for those HVRPs where the vehicle fixed cost contribution to the optimal cost is relevant or dominates the routing cost contribution.

RP involves two types of integer variables: $\xi_{ik} \in \{0, 1\}$, $i \in V$, $k \in M$ and $y_k \in \mathbb{Z}^+$, $k \in M$. Variable ξ_{ik} is equal to 1 if and only if customer $i \in V$ is served by a vehicle of type $k \in M$. Variable y_k represents the number of vehicles of type k used in the solution.

Let β_{ik} be the *marginal routing cost* for servicing customer $i \in V$ with a vehicle of type $k \in M$. We assume that the values β_{ik}, $i \in V$, $k \in M$, satisfy the following inequalities:

$$\sum_{i \in R^k_\ell} \beta_{ik} \leq c^k_\ell, \quad \forall \ell \in \mathscr{R}^k, \quad \forall k \in M. \tag{9}$$

Lemma 1, reported in Appendix A, shows that the following integer problem RP provides a valid lower bound on the HVRP for any solution β_{ik} of inequalities (9).

$$(RP) \quad z(RP) = \min \sum_{k \in M} \sum_{i \in V} \beta_{ik}\xi_{ik} + \sum_{k \in M} F_k y_k \tag{10}$$

$$\text{s.t.} \quad \sum_{k \in M} \sum_{i \in V} q_i \xi_{ik} = q(V), \tag{11}$$

$$\sum_{i \in V} q_i \xi_{ik} \leq Q_k y_k, \quad \forall k \in M, \tag{12}$$

$$y_k \leq U_k, \quad \forall k \in M, \tag{13}$$

$$\xi_{ik} \in \{0, 1\}, \quad \forall i \in V, \forall k \in M, \tag{14}$$

$$y_k \in \mathbb{Z}^+, \quad \forall k \in M. \tag{15}$$

The effectiveness of the lower bound obtained by solving RP is strongly dependent on the values β_{ik} used. In Sect. 5 we describe two bounding procedures, called DP^1 and DP^2, that correspond to two different methods for computing β_{ik} satisfying inequalities (9). DP^1 uses q-route relaxation while DP^2 uses column generation. Both procedures use the same dynamic programming algorithm to solve problem RP. The initial master problem for DP^2 is generated by using the cost contributions β^1_{ik} obtained by DP^1. Lower bounds $LD1$ and $LD2$ correspond to the cost $z(RP)$ of the RP solution achieved by DP^1 and DP^2, respectively.

It is quite easy to show that $LD1 \leq LD2 \leq z(RP)$. The computational results (see Sect. 7) show that no dominance relation exists between $z(RP)$ and $z(LF)$ but, in practice, both $LD1$ and $LD2$ can be greater than $LH2$ when $F_k > 0$, for some $k \in M$.

2.3 Relaxation $\overline{LF}$

A better relaxation than LF, called $\overline{LF}$, is obtained by adding the following valid inequalities to LF.

(a) *Capacity Constraints*. These constraints are added to LF only if all vehicle types have the same capacity Q (i.e., $Q_k = Q, \forall k \in M$). Let $\mathscr{S} = \{S : S \subseteq V, |S| \geq 2\}$, let $q(S) = \sum_{i \in S} q_i$ be the total demand of customers in S, and let $k(S) = \lceil q(S)/Q \rceil$. The following inequalities are added to LF:

$$\sum_{k \in M} \sum_{\ell \in \mathscr{R}^k(S)} x^k_\ell \geq k(S), \quad \forall S \in \mathscr{S}, \tag{16}$$

where $\mathscr{R}^k(S) = \{\ell \in \mathscr{R}^k : R^k_\ell \cap S \neq \emptyset\}, k \in M$.

(b) *Clique Inequalities*. Let $H = (N, \overline{E})$ be the *conflict graph* where each node $i \in N$ is associated to a route $\ell \in \mathscr{R}$. For each route $\ell \in \mathscr{R}^k$ and for a given k, the corresponding node in graph H has index $i = \sum_{h=1}^{k-1} |\mathscr{R}^h| + \ell$. We denote by $k(i)$ and by $\ell(i)$ the vehicle type and the index of the route represented by node $i \in N$, respectively.
The edge set $\overline{E}$ contains every pair $\{i, j\}, i < j$ such that $R^{k(i)}_{\ell(i)} \cap R^{k(j)}_{\ell(j)} \neq \emptyset$. Let $\mathscr{C}$ be the set of all maximal cliques of H. The following inequalities are added to LF:

$$\sum_{i \in C} x^{k(i)}_{\ell(i)} \leq 1, \quad \forall C \in \mathscr{C}. \tag{17}$$

Let $z(\overline{LF})$ be the optimal solution cost of $\overline{LF}$. Relaxation $\overline{LF}$ is solved by means of a standard column and cut generation method, called CG, that is described in Sect. 6. The initial master problem is generated by using either the dual solution $(\mathbf{u}^2, \mathbf{v}^2)$ given by H^2 or the marginal routing cost β^2_{ik} obtained by DP^2 as described in Sect. 2.2. The master problem is then solved by using a simplex algorithm where, at each iteration, a limited subset of inequalities (16) and (17), violated by the current fractional solution, are added to the master. Lower bound LCG corresponds to the cost $z(\overline{LF})$ of the final $\overline{LF}$ solution achieved by CG.

It is easy to observe that $z(\overline{LF}) \geq z(LF)$. No dominance relation exists between $z(\overline{LF})$ and $z(RP)$ and, in practice, $LD1$ and $LD2$ can be greater than LCG (see Sect. 7).

3 An exact method for solving the HVRP

In this section we describe an exact algorithm for solving the HVRP. This algorithm generalizes the method proposed by Baldacci et al. [3] for the CVRP.

The method consists of finding, by means of an integer linear programming solver (such as CPLEX [11]), an optimal integer solution of a reduced problem $\hat{F}$ obtained from F by replacing each set $\mathscr{R}^k$, $k \in M$, with a subset $\hat{\mathscr{R}}^k$, and adding two subsets $\hat{\mathscr{S}} \subset \mathscr{S}$ and $\hat{\mathscr{C}} \subset \mathscr{C}$ of inequalities (16) and (17), respectively. The subsets $\hat{\mathscr{R}}^k$, $\forall k \in M$, are generated in such a way that any optimal $\hat{F}$ solution is also optimal for the HVRP.

The core of the algorithm is the bounding method that combines different bounding procedures based on the three relaxations described in Sect. 2.

3.1 Bounding method

The bounding method combines the different lower bounds $LH1$, $LH2$, $LD1$, $LD2$ and LCG as follows.

1. Execute in sequence H^1 and H^2. Let $LH2$ be the lower bound corresponding to the cost of the DLF solution $(\mathbf{u}^2, \mathbf{v}^2)$ obtained by bounding procedure H^2. If $F_k = 0$, $\forall k \in M$, set $LD1 = 0$, $LD2 = 0$ and go to Step 3, otherwise perform Step 2.
2. Execute DP^1. If $LD1 \geq LH2$, execute DP^2 producing lower bound $LD2$ corresponding to the marginal routing cost β^2_{ik}. If $LD1 < LH2$ set $LD2 = 0$.
3. Execute bounding procedure CG, as described in Sect. 6. Compute the final lower bound $LB = \max\{LD2, LCG\}$.

3.2 Generating the reduced problem $\hat{F}$

In generating the route subsets $\hat{\mathscr{R}}^k$, $k \in M$, we use either the dual solution of $\overline{LF}$ obtained by procedure CG or the marginal routing costs β^2_{ik} obtained by DP^2. We have the following two cases:

(a) $LB = LCG$. We replace the route sets $\mathscr{R}^k$, $k \in M$, with the route subsets $\hat{\mathscr{R}}^k$, $k \in M$, containing all routes whose reduced costs, with respect to the dual solution of $\overline{LF}$ achieved by CG is smaller than the gap $z_{UB} - LCG$, where z_{UB} is a valid upper bound on the HVRP. We consider only those constraints (16) and (17) generated by CG that have zero slack in the final $\overline{LF}$ solution.

(b) $LB = LD2$. We generate for each $k \in M$ the subset $\hat{\mathscr{R}}^k$ containing all routes satisfying the following inequalities:

$$c_\ell^k - \sum_{i \in R_\ell^k} \beta_{ik}^2 < z_{UB} - z(LD2), \quad \forall \ell \in \hat{\mathscr{R}}^k. \tag{18}$$

> It is quite easy to show (see Corollary 1 in Appendix A) that any route $\ell \in \mathscr{R}^k \backslash \hat{\mathscr{R}}^k$, for a given vehicle type k, cannot belong to an optimal HVRP solution of cost smaller than z_{UB}.

In both cases, the route sets $\hat{\mathscr{R}}^k, k \in M$, are generated by using procedure GENROUTE described in Appendix B.

The effectiveness of the proposed exact method strongly depends on the quality of the lower bounds $LD2$ and LCG achieved. As the lower bounds $LD2$ and LCG get better, the reduced costs of the routes of an optimal HVRP solution get smaller and, hopefully, the size of subsets $\hat{\mathscr{R}}^k$, $k \in M$, that must be generated to find an optimal HVRP solution gets smaller, and the resulting problem $\hat{F}$ becomes easier to solve.

4 Bounding procedures based on relaxation LF

In this section we describe the two bounding procedure H^1 and H^2 that find two near-optimal feasible solutions of DLF without requiring the generation of the entire route sets $\mathscr{R}^k$, $k \in M$. Moreover, we present a method, based on procedure H^1, for reducing the upper bound U_k on the number of vehicles of type $k \in M$ that can be in any optimal solution. Both H^1 and H^2 are based on the following theorem.

Theorem 1 *Associate penalties $\lambda_i \in \mathbb{R}$, $\forall i \in V$, with constraints (2) and penalties $\mu_k \leq 0$, $\forall k \in M$, with constraints (3). Define*

$$b_{ik} = q_i \min_{\ell \in \mathscr{R}_i^k} \left\{ \frac{c_\ell^k + F_k - \lambda(R_\ell^k) - \mu_k}{q(R_\ell^k)} \right\}, \qquad \forall i \in V, \quad \forall k \in M, \tag{19}$$

where $\lambda(R_\ell^k) = \sum_{i \in R_\ell^k} \lambda_i$ and $q(R_\ell^k) = \sum_{i \in R_\ell^k} q_i$.

A feasible DLF solution $(\mathbf{u}, \mathbf{v})$ of cost $z(DLF(\boldsymbol{\lambda}, \boldsymbol{\mu}))$ is given by setting:

$$\left.\begin{aligned} u_i &= \min_{k \in M}\{b_{ik}\} + \lambda_i, \ \forall i \in V, && (a) \\ v_k &= \mu_k, \ \forall k \in M. && (b) \end{aligned}\right\} \tag{20}$$

Proof See Appendix A. □

Both H^1 and H^2 use subgradient optimization to solve $\max_{\boldsymbol{\lambda},\boldsymbol{\mu}}\{z(DLF(\boldsymbol{\lambda}, \boldsymbol{\mu}))\}$.

4.1 Bounding procedure H^1

We relax the requirement that a route be a simple cycle of graph G. This relaxation allows us to compute in pseudo-polynomial time a lower bound $\overline{b}_{ik} \leq b_{ik}$, $\forall i \in V$, $\forall k \in M$, as follows.

For a given penalty vector $\boldsymbol{\lambda}$, define the *modified edge costs* $\overline{d}_{ij}^k$ as

$$\overline{d}_{ij}^k = d_{ij}^k - \frac{1}{2}\lambda_i - \frac{1}{2}\lambda_j, \qquad \forall \{i, j\} \in E, \quad \forall k \in M. \tag{21}$$

By using the modified edge costs $\overline{d}_{ij}^k$, let $\phi(k, q, i)$ be the cost of a least cost (not necessarily simple) cycle $C(k, q, i)$, called q-route, passing through the depot and vertex $i \in V$, for a vehicle of type k, such that the total demand of the customers visited is equal to q ($q_i \le q \le Q_k$). $\phi(k, q, i)$ provides a lower bound on the cost $c_\ell^k - \lambda(R_\ell)$ of any feasible route $\ell \in \mathscr{R}_i^k$ of load $q(R_\ell^k) = q$. The values $\phi(k, q, i)$, $\forall k \in M$, $\forall i \in V$, and for all integer values of q such that $q_i \le q \le Q_k$, can be computed by a straightforward extension of the method proposed in Christofides et al. [7]. Thus, values $\overline{b}_{ik}$ can be computed as follows:

$$\overline{b}_{ik} = q_i \min_{q_i \le q \le Q_k} \left\{ \frac{F_k + \phi(k, q, i) - \mu_k}{q} \right\} \le b_{ik}, \quad \forall i \in V, \ \forall k \in M. \tag{22}$$

Let $\overline{q}_{ik}$ be the value of q giving the minimum in expression (22).

It can be shown that Theorem 1 remains valid if in expression (20) each b_{ik} is replaced with $\overline{b}_{ik}$, $i \in V$, $k \in M$.

4.1.1 Computing lower bound $LH1$

H^1 uses subgradient optimization to solve $LH1 = \max_{\boldsymbol{\lambda},\boldsymbol{\mu}}\{z(DLF(\boldsymbol{\lambda}, \boldsymbol{\mu}))\}$. At the end of H^1 we have a DLF solution $(\mathbf{u}^1, \mathbf{v}^1)$ of cost $LH1$.

H^1 starts by initializing $LH1 = 0$, $\boldsymbol{\lambda} = \mathbf{0}$, $\boldsymbol{\mu} = \mathbf{0}$. At each iteration, for the current values of $\boldsymbol{\lambda}$ and $\boldsymbol{\mu}$, H^1 performs the following two steps:

1. Compute the modified edge costs $\overline{d}_{ij}^k$ according to expression (21) and functions $\phi(k, q, i)$, $\forall k \in M$, $\forall i \in V$ and for all integer values of q such that $q_i \le q \le Q_k$. Then, compute the values $\overline{b}_{ik}$ by using expressions (22) and the corresponding DLF solution $(\mathbf{u}, \mathbf{v})$ of cost $z(DLF(\boldsymbol{\lambda}, \boldsymbol{\mu}))$ by using expressions (20), where b_{ik} is replaced with $\overline{b}_{ik}$.
   Let k_i, $i \in V$, be the vehicle type giving the minimum in expression (20a), that is $u_i = \overline{b}_{ik_i} + \lambda_i = q_i(F_{k_i} + \phi(k_i, \overline{q}_{ik_i}, i) - \mu_{k_i})/\overline{q}_{ik_i} + \lambda_i$.
   If $z(DLF(\boldsymbol{\lambda}, \boldsymbol{\mu}))$ is greater than $LH1$, then update $LH1 = z(DLF(\boldsymbol{\lambda}, \boldsymbol{\mu}))$, $\mathbf{u}^1 = \mathbf{u}$, $\mathbf{v}^1 = \mathbf{v}$, $\boldsymbol{\lambda}^1 = \boldsymbol{\lambda}$ and $\boldsymbol{\mu}^1 = \boldsymbol{\mu}$.
2. Let θ_j be the number of times that customer $j \in V$ is visited and ρ_k be the number of vehicles of type k used by the q-routes $C(k_i, \overline{q}_{ik_i}, i)$, $\forall i \in V$, respectively. Let $\delta_j(k_i, \overline{q}_{ik_i}, i)$, $i \in V$, be the number of times that customer j is visited by the q-route $C(k_i, \overline{q}_{ik_i}, i)$. The values of $\boldsymbol{\theta}$ and $\boldsymbol{\rho}$ are computed as follows:
   - Initialize $\theta_j = 0$, $\forall j \in V$, and $\rho_k = 0$, $\forall k \in M$.
   - For each $i \in V$, update $\theta_j = \theta_j + \frac{q_i}{\overline{q}_{ik_i}}\delta_j(k_i, \overline{q}_{ik_i}, i)$, $\forall j \in V$, and $\rho_k = \rho_k + \frac{q_i}{\overline{q}_{ik_i}}$, $\forall k \in M$.

The values of $\boldsymbol{\lambda}$ and $\boldsymbol{\mu}$ are modified as follows: $\lambda_j = \lambda_j - \varepsilon\gamma(\theta_j - 1), \forall j \in V$ and $\mu_k = \min\{0, \mu_k - \varepsilon\gamma(\rho_k - U_k)\}, \forall k \in M$, where ε is a positive constant and $\gamma = (z_{UB} - z(DLF(\boldsymbol{\lambda}, \boldsymbol{\mu})))/(\sum_{j\in V}(\theta_j - 1)^2 + \sum_{k\in M}(\rho_k - U_k)^2)$.

Procedure H^1 terminates in $Maxt1$ iterations, where $Maxt1$ is a parameter defined a priori.

4.1.2 Reduction tests

At the end of H^1, for HVRP instances having $F_k > 0, \forall k \in M$, we apply the following method to reduce the upper bounds $U_k, \forall k \in M$.

Let LB_R be the lower bound obtained by performing $Maxt1r$ iterations of bounding procedure H^1 after having set $F_k = 0, \forall k \in M$. LB_R is a valid lower bound on the total routing cost of the solution. Then, the upper bounds U_k can be updated by setting:

$$U_k = \min\{U_k, \lfloor (z_{UB} - LB_R)/F_k \rfloor\}, \quad \forall k \in M. \tag{23}$$

4.2 Bounding procedure H^2

Procedure H^2 is a column generation method that computes $LH2$ as the cost of a near optimal solution $(\mathbf{u}^2, \mathbf{v}^2)$ of the dual problem DLF. H^2 differs from standard column generation methods as the master problem is solved heuristically by using expressions (19) and (20) and subgradient optimization.

H^2 is initialized by setting $\boldsymbol{\lambda} = \boldsymbol{\lambda}^1$, $\boldsymbol{\mu} = \boldsymbol{\mu}^1$ and $LH2 = 0$ and by generating, for each $k \in M$, the subset $\overline{\mathscr{R}}^k \subset \mathscr{R}^k$ of the $\Delta^{\min}$ routes of minimum reduced cost with respect to the DLF solution $(\mathbf{u}^1, \mathbf{v}^1)$ obtained by H^1 ($\Delta^{\min}$ is a parameter defined a priori). H^2 executes an a priori defined number $Maxt2$ of macro iterations. On each macro iteration it performs the following two steps:

1. *Solve the master problem.* The master problem is obtained from LF by replacing each $\mathscr{R}^k$ with $\overline{\mathscr{R}}^k, \forall k \in M$. The dual solution of the master is computed by means of an iterative procedure based on subgradient optimization. At each iteration, for the current $\boldsymbol{\lambda}$ and $\boldsymbol{\mu}$, the dual solution $(\mathbf{u}, \mathbf{v})$ is computed by using expressions (19) and (20) where $\mathscr{R}^k$ is replaced with $\overline{\mathscr{R}}^k, \forall k \in M$.
   The initial values of $\boldsymbol{\lambda}$ and $\boldsymbol{\mu}$ are set equal to the best values achieved at the previous macro iteration. Penalties $\boldsymbol{\lambda}$ and $\boldsymbol{\mu}$ are modified as described for procedure H^1.
   Let $(\mathbf{u}^*, \mathbf{v}^*)$ be the best dual solution of the master, of cost z^*, achieved after $Maxt3$ iterations.
2. *Check if the master solution* $(\mathbf{u}^*, \mathbf{v}^*)$ *is a feasible* DLF *solution.* For each $k \in M$, generate the largest subset $\mathscr{N}^k \subset \mathscr{R}^k \backslash \overline{\mathscr{R}}^k$ containing the routes having the largest negative reduced cost with respect to the dual master solution $(\mathbf{u}^*, \mathbf{v}^*)$ and such that $|\mathscr{N}^k| \leq \Delta^a$, where Δ^a is a parameter defined a priori. We have two cases:
   (a) if $\mathscr{N}^k = \emptyset, \forall k \in M$, and z^* is greater than $LH2$, update $LH2 = z^*, \mathbf{u}^2 = \mathbf{u}^*$ and $\mathbf{v}^2 = \mathbf{v}^*$.
   (b) if $\mathscr{N}^k \neq \emptyset$, for some $k \in M$, update $\overline{\mathscr{R}}^k = \overline{\mathscr{R}}^k \cup \mathscr{N}^k, \forall k \in M$.

The initial route subsets $\overline{\mathscr{R}}^k$, $\forall k \in M$, and the subsets $\mathscr{N}^k$, $\forall k \in M$, are generated by procedure GENROUTE, described in Appendix B, using different input parameters.

5 Bounding procedures based on relaxation RP

In this section we describe the two methods used by the two bounding procedures DP^1 and DP^2 to find a feasible solution of inequalities (9) as well as the dynamic programming algorithm used by both DP^1 and DP^2 to solve the integer problem RP. Also in this section, we present two methods, based on procedure DP^1 and DP^2, for computing a lower bound L_k and for reducing the upper bound U_k on the number of vehicles of type $k \in M$ that can be in any optimal solution.

The method used by DP^1 and DP^2 to find a feasible solution of (9) is based on the following theorem.

Theorem 2 *Let $\lambda_i \in \mathbb{R}$, $\forall i \in V$, be a set of penalties associated to the customers. A feasible solution β_{ik} of inequalities* (9) *is given by setting*:

$$\beta_{ik} = q_i \min_{\ell \in \mathscr{R}_i^k} \left\{ \frac{c_\ell^k - \lambda(R_\ell^k)}{q(R_\ell^k)} \right\} + \lambda_i, \quad \forall i \in V, \ \forall k \in M. \tag{24}$$

Proof See Appendix A. □

The value of the lower bound on the HVRP achieved from relaxation RP is a function of the penalty vector $\boldsymbol{\lambda}$ used in computing the values of β_{ik} according to expression (24). Thus, the value of the lower bound can be maximized by using subgradient optimization.

5.1 Bounding procedure DP^1

This procedure is based on the q-route relaxation of the HVRP for computing the values β_{ik} and uses subgradient optimization to maximize lower bound $LD1$.

Let functions $\phi(k, q, i)$, introduced in Sect. 4.1, be computed by using the modified edge costs $\overline{d}_{ij}^k$ that are defined with respect to a given penalty vector $\boldsymbol{\lambda}$, according to expressions (21). It is easy to show that a solution of inequalities (9) is given by setting:

$$\beta_{ik} = q_i \min_{q_i \le q \le Q_k} \left\{ \frac{\phi(k, q, i)}{q} \right\} + \lambda_i, \quad \forall i \in V, \ \forall k \in M. \tag{25}$$

DP^1 is an iterative procedure that performs at most $Maxt1$ subgradient iterations to compute lower bound $LD1$. DP^1 is initialized by setting $\lambda_i = 0$, $\forall i \in V$, and $LD1 = 0$. At each iteration, DP^1 performs the following steps:

1. For the current penalty vector $\boldsymbol{\lambda}$, compute functions $\phi(k, q, i)$ and values β_{ik} by using expressions (25).

2. Solve problem RP, as described in Sect. 5.3. Let $(\boldsymbol{\xi}^*, \mathbf{y}^*)$ be the optimal RP solution of cost $z(RP)$. If $z(RP)$ is greater than $LD1$, update $LD1 = z(RP)$ and $\beta^1_{ik} = \beta_{ik}, \forall i \in V, \forall k \in M$.
3. Update penalties $\lambda_i, \forall i \in V$. Because any feasible HVRP solution must satisfy $\sum_{k \in M} \xi_{ik} = 1, \forall i \in V$, a valid subgradient to the Lagrangean function $z(RP)$ is given by the vector $\boldsymbol{\theta} = (\theta_i, \ldots, \theta_n)$, where $\theta_i = \sum_{k \in M} \xi^*_{ik} - 1, i \in V$. Penalty vector $\boldsymbol{\lambda}$ is updated as $\lambda_i = \lambda_i + \varepsilon\gamma\theta_i, \forall i \in V$, where ε is a positive constant and $\gamma = (z_{UB} - z(RP))/\sum_{i \in V} \theta_i^2$.

5.2 Bounding procedure DP^2

DP^2 is an iterative procedure that uses column generation for computing the values β_{ik} by means of expressions (24) and subgradient optimization to maximize the value of the lower bound $LD2$.

The core of DP^2 consists in solving problem $\overline{RP}$ that is derived from RP by replacing cost β_{ik} with the solution $\overline{\beta}_{ik}$ of the reduced set of inequalities (9) obtained after replacing each set $\mathscr{R}^k$ with a subset $\overline{\mathscr{R}}^k \subset \mathscr{R}^k$ of limited size. The values $\overline{\beta}_{ik}$ are obtained, for a given $\boldsymbol{\lambda}$, by means of expressions (24) by using the subsets $\overline{\mathscr{R}}^k$ instead of sets $\mathscr{R}^k, \forall k \in M$.

Let $z(\overline{RP})$ be the optimal cost of $\overline{RP}$. It is simple to notice that $z(\overline{RP})$ represents a valid lower bound on the HVRP if and only if the values $\overline{\beta}_{ik}$ given by expressions (24) satisfy all inequalities (9) for all routes $\mathscr{R}^k, \forall k \in M$. At each iteration of DP^2, a limited number of routes, whose inequalities (9) are violated by the values $\overline{\beta}_{ik}$ are added to $\overline{\mathscr{R}}^k, k \in M$.

DP^2 is initialized by generating for each $k \in M$ the subset $\overline{\mathscr{R}}^k \subset \mathscr{R}^k$ containing the $\Delta^{\min}$ routes of minimum reduced cost with respect to the values β^1_{ik} achieved by DP^1 and by setting $\lambda_i = 0, \forall i \in V$, and $LD2 = 0$.

DP^2 executes an a priori defined number, $Maxt2$, of macro iterations. On each iteration it performs the following steps:

1. *Maximize* $z(\overline{RP})$ *with respect to* $\boldsymbol{\lambda}$. Initialize $z^* = 0$ and perform $Maxt3$ iterations of the following procedure:
   (i) By using the current penalty vector $\boldsymbol{\lambda}$, compute $\overline{\beta}_{ik}$ by means of expressions (24), where each $\mathscr{R}^k$ is replaced with $\overline{\mathscr{R}}^k, \forall k \in M$.
   (ii) Solve $\overline{RP}$ by using the algorithm described in Sect. 5.3. If $z(\overline{RP})$ is greater than z^*, then update $z^* = z(\overline{RP})$ and set $\beta^*_{ik} = \overline{\beta}_{ik}, \forall i \in V, \forall k \in M$.
   (iii) Update penalties $\lambda_i, \forall i \in V$, as described for algorithm DP^1 in Sect. 5.1.
2. Use procedure GENROUTE to generate the subset of routes $\mathscr{N}^k \subseteq \mathscr{R}^k \backslash \overline{\mathscr{R}}^k, \forall k \in M$, such that $|\mathscr{N}^k| \le \Delta^a$ and whose inequalities (9) are maximal violated by the values β^*_{ik}. We have two cases:
   (i) $\mathscr{N}^k = \emptyset, \forall k \in M$. Then, the values β^*_{ik} satisfy all inequalities (9) and z^* is a valid lower bound on the HVRP. If z^* is greater than $LD2$, then update $LD2 = z^*$ and $\beta^2_{ik} = \beta^*_{ik}, \forall i \in V, \forall k \in M$.
   (ii) $\mathscr{N}^k \neq \emptyset$, for some $k \in M$. Update $\overline{\mathscr{R}}^k = \overline{\mathscr{R}}^k \cup \mathscr{N}^k, \forall k \in M$.

5.3 Solving problem RP

In this section we describe a dynamic programming algorithm for solving problem RP.

Let h_w^k be the contribution to the objective function of RP if a load $w \in W_k$ is delivered by the vehicles of type $k \in M$, where $W_k = \{w : w = \min_{i\in V}\{q_i\}, \dots, Q_k U_k\}$. Each value h_w^k, $w \in W_k$, $k \in M$, is the optimal solution cost of the following subproblem:

$$(KP(k,w)) \quad h_w^k = \min \sum_{i\in V} \beta_{ik}\xi_{ik} \tag{26}$$

$$\text{s.t.} \quad \sum_{i\in V} q_i \xi_{ik} = w, \tag{27}$$

$$\xi_{ik} \in \{0,1\}, \quad \forall i \in V,\ \forall k \in M. \tag{28}$$

We denote by $V(k,w) = \{i \in V : \xi_{ik} = 1\}$ the subset of customers belonging to the optimal solution of problem $KP(k,w)$. We assume $h_w^k = \infty$ if $KP(k,w)$ has no feasible solution for some pair (k,w).

An alternative formulation of problem RP is as follows. Let ζ_{kw} be a 0-1 integer variable that is equal to 1 if and only if the vehicles of type k deliver to customers a load w. For any optimal RP solution $(\boldsymbol{\xi}, \mathbf{y})$ we have:

$$\sum_{w\in W_k} w\zeta_{kw} = \sum_{i\in V} q_i\xi_{ik}, \quad \forall k \in M, \tag{29}$$

$$\sum_{w\in W_k} \zeta_{kw} \le 1, \quad \forall k \in M, \tag{30}$$

$$\sum_{w\in W_k} \left\lceil \frac{w}{Q_k} \right\rceil \zeta_{kw} = y_k, \quad \forall k \in M. \tag{31}$$

Therefore, problem RP can be reformulated as the following problem $RP2$:

$$(RP2) \quad z(RP) = \min \sum_{k\in M}\sum_{w\in W_k} \left(h_w^k + F_k \left\lceil \frac{w}{Q_k} \right\rceil \right) \zeta_{kw} \tag{32}$$

$$\text{s.t.} \quad \sum_{k\in M}\sum_{w\in W_k} w\zeta_{kw} = q(V), \tag{33}$$

$$\sum_{w\in W_k} \zeta_{kw} \le 1, \quad \forall k \in M, \tag{34}$$

$$\zeta_{kw} \in \{0,1\}, \quad \forall w \in W_k,\ \forall k \in M. \tag{35}$$

The method used to solve problem $RP2$ consists of the following two-phase procedure.

1. *Computing* h_w^k, $\forall w \in W_k$, $\forall k \in M$. It is easy to note that, for a given k, all values h_w^k, $w \in W_k$, can be obtained by solving problem $KP(k, Q_k U_k)$ by using a dynamic programming recursion.

2. *Solving problem $RP2$.* $RP2$ can be solved by dynamic programming as follows. Let $g_k(w)$ be the optimal solution cost of $RP2$ by using vehicle types $1, 2, \dots, k$ and by replacing the term $q(V)$ in Eq. (33) with $w \le q(V)$. The recursion for computing function $g_k(w)$ is:

$$g_k(w) = \min\left\{g_{k-1}(w), \min_{w' \le w}\left\{g_{k-1}(w - w') + h^k_w + F_k\left\lceil\frac{w}{Q_k}\right\rceil\right\}\right\} \qquad (36)$$

   for $k = 2, \dots, m$ and $\forall w \in W_k$.
   The recursion is initialized by setting $g_1(w) = h^1_w + F_1\lceil w/Q_k\rceil, \forall w \in W_1$ and $g_k(0) = 0,\ k = 0, \dots, m,\ g_0(q) = \infty,\ q = 1, \dots, q(V)$. Then, the optimal solution cost of $RP2$ is $z(RP) = g_m(q(V))$.
   Usual backtracking can be used to derive the optimal $RP2$ solution $\boldsymbol{\zeta}$. Moreover, given $\boldsymbol{\zeta}$ and the sets $V(k, w)$ associated to $h^k_w, \forall w \in W_k, \forall k \in M$, we can derive the optimal RP solution $(\boldsymbol{\xi}^*, \mathbf{y}^*)$ of cost $z(RP)$ by setting:
   - for each $k \in M$ such that $\sum_{w \in W_k} \zeta^*_{kw} = 1$, define $\xi^*_{ik} = 1, \forall i \in V(k, w)$, $\xi^*_{ik} = 0, \forall i \notin V(k, w)$ and $y^*_k = \lceil w/Q_k\rceil$, where w satisfies $\zeta^*_{kw} = 1$;
   - for each $k \in M$ such that $\sum_{w \in W_k} \zeta^*_{kw} = 0$, define $\xi^*_{ik} = 0, \forall i \in V$, and $y^*_k = 0$.

5.4 Reduction tests

In this section, we describe two reduction methods to define both lower and upper bounds on the number of vehicles of each type to be used in the solution. To this end, in addition to the upper bound U_k, $k \in M$, we introduce a lower bound L_k that represents the minimum number of vehicles of type k that must be used in any optimal solution. We assume that lower bounds $L_k, \forall k \in M$, are all initialized equal to 0. The two methods are applied at the end of both procedures DP^1 and DP^2.

Let h^k_w be the optimal cost of problem $KP(k, w)$ obtained by replacing the values β_{ik} in expression (26) either with β^1_{ik} or with β^2_{ik} if the reductions are applied after DP^1 or DP^2, respectively. Moreover, let $g^{-1}_k(w)$ be the optimal solution cost of $RP2$ by using vehicle types $k, k+1, \dots, m$ and by replacing in Eq. (33) the term $q(V)$ with $w \le q(V)$. Function $g^{-1}_k(w)$ can be computed by using a dynamic programming recursion similar to the one used for computing function $g_k(w)$ [see expression (36)].

5.4.1 Reduction R1: computing L_k, $\forall k \in M$

By means of function $g_k(w)$, $g^{-1}_k(w)$ and h^k_w we can compute a valid lower bound $LB_k(w)$ on the HVRP with the restriction that vehicles of type k deliver a total load w as follows:

$$LB_k(w) = \min_{\substack{q, q' \\ q + q' = q(V) - w}}\left\{g_{k-1}(q) + h^k_w + F_k\left\lceil\frac{w}{Q_k}\right\rceil + g^{-1}_{k+1}(q')\right\} \qquad (37)$$

for $w = 0, \ldots, Q_k U_k$. We assume $g_{m+1}^{-1}(0) = 0$ and $g_{m+1}^{-1}(q) = \infty, q = 1, \ldots, q(V)$. Let $w_k^{\min}$ be the minimum value of $w = 1, \ldots, Q_k U_k$ such that

$$LB_k(w_k^{\min} - 1) \geq z_{UB} \quad \text{and} \quad LB_k(w_k^{\min}) < z_{UB}, \tag{38}$$

then $L_k = \max\{L_k, \lceil w_k^{\min}/Q_k \rceil\}$ is a valid lower bound on the minimum number of vehicles of type k in any optimal solution.

5.4.2 Reduction R2: updating U_k, $\forall k \in M$

Let $LBU_k(r)$ be a lower bound on the HVRP with the additional requirement that exactly r vehicles of type k are in the solution. We have:

$$LBU_k(r) = F_k r + \min_{\substack{q, q', w \\ q + q' = q(V) - w \\ w \leq \min\{q(V), Q_k r\}}} \{g_{k-1}(q) + h_w^k + g_{k+1}^{-1}(q')\} \tag{39}$$

for $r = U_k, \ldots, \max\{L_k, 1\}$.

For any $k \in M$ such that $LBU_k(U_k) \geq z_{UB}$, the value U_k can be reduced as follows. Let $r_k^{\max}$ be the maximum value of $r < U_k$ such that

$$LBU_k(r_k^{\max} + 1) \geq z_{UB} \text{ and } LBU_k(r_k^{\max}) < z_{UB}. \tag{40}$$

Then, update $U_k = \min\{U_k, r_k^{\max}\}$.

6 Bounding procedure based on relaxation $\overline{LF}$

Procedure CG computes lower bound LCG as the cost of an optimal solution of problem $\overline{LF}$. Whenever DP^1 and DP^2 are executed before CG, we add to $\overline{LF}$ the following constraints:

$$\sum_{\ell \in \mathscr{R}^k} x_\ell^k \geq L_k, \quad \forall k \in M. \tag{41}$$

CG solves problem $\overline{LF}$ by using cut and column generation. The initial master problem is obtained from $\overline{LF}$ by replacing the sets $\mathscr{R}^k$ with the route subsets $\overline{\mathscr{R}}^k$, $\forall k \in M$, generated by H^2 and by setting $\overline{\mathscr{S}} = \emptyset$ and $\overline{\mathscr{C}} = \emptyset$.

At each iteration (say t), CG performs the following steps:

1. *Solve problem $\overline{LF}$*. Let $\overline{\mathbf{x}}$ and $(\overline{\mathbf{u}}, \overline{\mathbf{v}}, \overline{\mathbf{w}}, \overline{\mathbf{g}})$ be the optimal primal and dual solutions of $\overline{LF}$, respectively. Vector $\overline{\mathbf{v}}$ is the sum of the dual vectors associated to constraints (3) and (41). Vectors $\overline{\mathbf{w}}$ and $\overline{\mathbf{g}}$ represent the dual variables associated to inequalities (16) and (17), respectively.
2. *Generate the route subsets $\mathscr{N}^k$, $\forall k \in M$, having negative reduced cost*. Generate the largest route subsets $\mathscr{N}^k \subset \mathscr{R}^k \backslash \overline{\mathscr{R}}^k$, $\forall k \in M$, of routes having the largest negative reduced cost with respect to $(\overline{\mathbf{u}}, \overline{\mathbf{v}}, \overline{\mathbf{w}}, \overline{\mathbf{g}})$, such that $|\mathscr{N}^k| \leq \Delta^a$, $\forall k \in M$.

If $\mathcal{N}^k = \emptyset, \forall k \in M$, then CG terminates; otherwise a new iteration is made. At iteration $t+1$, CG solves a new master problem $\overline{LF}$ by adding to each set $\overline{\mathscr{R}}^k$ the subset $\mathcal{N}^k$, $\forall k \in M$, and the constraints (16) and (17) violated by the $\overline{LF}$ solution $\bar{\mathbf{x}}$ achieved at iteration t. Procedure CG terminates whenever $\mathcal{N}^k = \emptyset$, $\forall k \in M$.

3. *Separate violated inequalities*. Both capacity constraints and clique inequalities are separated as described by Baldacci et al. [3] for the CVRP.
   - *Capacity Constraints*. For a given $\overline{LF}$ solution $\bar{\mathbf{x}}$ of the master, define the weight ω_{ij} for each edge $\{i, j\} \in E$ as follows:

$$\omega_{ij} = \sum_{k \in M} \sum_{\ell \in \mathscr{R}^k} \eta_{ij}^{k\ell} \bar{x}_\ell^k, \tag{42}$$

     where the coefficients $\eta_{ij}^{k\ell}$ are defined, for each $\ell \in \mathscr{R}^k$ and $k \in M$, as follows.
     - If ℓ is a single customer route covering customer h, then $\eta_{0h}^{k\ell} = 2$ and $\eta_{ij}^{k\ell} = 0, \forall \{i, j\} \in E \setminus \{0, h\}$.
     - If ℓ is not a single customer route, then $\eta_{ij}^{k\ell} = 1$ for each edge $\{i, j\} \in E(R_\ell^k)$ and $\eta_{ij}^{k\ell} = 0, \forall \{i, j\} \in E \setminus E(R_\ell^k)$, where $E(R_\ell^k)$ represents the subset of edges covered by route R_ℓ^k.

     It is quite simple to note that any feasible HVRP solution, where all vehicle types have the same capacity Q, satisfies the constraints

$$\sum_{i \in S} \sum_{\substack{j \in V' \setminus S \\ i<j}} \omega_{ij} + \sum_{i \in V' \setminus S} \sum_{\substack{j \in S \\ i<j}} \omega_{ij} \geq 2k(S), \quad \forall S \in \mathscr{S}. \tag{43}$$

     It can be shown that if the $\overline{LF}$ solution $\bar{\mathbf{x}}$ violates inequalities (43) for some S, then it also violates inequalities (16) for the same S. Therefore, we add to $\overline{\mathscr{S}}$ those sets S violating inequalities (43). These inequalities are separated by using the package CVRPSEP (see Lysgaard [20]) but limiting the separation routine in order to find a maximum of 70 violated inequalities.
   - *Clique Inequalities*. Let $H(\bar{\mathbf{x}}) = (\mathscr{R}(\bar{\mathbf{x}}), \overline{E}(\bar{\mathbf{x}}))$ be the subgraph of graph H induced by the $\overline{LF}$ solution $\bar{\mathbf{x}}$ where $\mathscr{R}(\bar{\mathbf{x}}) = \{\ell \in \overline{\mathscr{R}}^k : \bar{x}_\ell^k > 0, k \in M\}$, and let $\bar{x}_\ell^k$ be the weight of vertex $\ell \in \mathscr{R}(\bar{\mathbf{x}})$.

     The inequalities (17) violated by solution $\bar{\mathbf{x}}$ can be separated by finding all the maximal cliques in $H(\bar{\mathbf{x}})$ of weight greater than one. This problem is solved, as described in Baldacci et al. [3], by using the CLIQUER 1.1 package (see Niskanen and Östergård [23]). Every new clique is expanded by adding the $\hat{h}$ least reduced cost routes of the current sets $\overline{\mathscr{R}}^k$, $\forall k \in M$, and the resulting lifted clique inequality is added to the set $\overline{\mathscr{C}}$. In our computational results we used $\hat{h} = 1,000$.

Let LCG and $(\mathbf{u}^3, \mathbf{v}^3, \mathbf{w}^3, \mathbf{g}^3)$ be the optimal solution cost and the optimal dual solution of $\overline{LF}$ achieved by CG, respectively.

7 Computational results

This section presents computational results of the exact method described in Sect. 3 and of the bounding procedures described in Sects. 4 and 5.

The algorithms described in this paper were coded in Fortran 77, compiled with the Compaq Digital Fortran 6.6 compiler and linked with the C source codes of the packages CVRPSEP (see [20]) and CLIQUER (see [23]). CPLEX 10.1 was used as the LP solver in procedure CG and as the integer linear programming solver in the exact method. The experiments were performed on a personal computer equipped with an AMD Athlon 64 X2 Dual Core 4200+ processor at 2.6 GHz and with 3 GB of RAM.

We considered the following three main sets of instances from the literature corresponding to the different variants of HVRPs, SDVRP and MDVRP.

(a) *HVRP instances*

All the instances considered are based on the set of twelve symmetric instances proposed by Golden et al. [15] and Taillard [26]. The data of all instances including the best known upper bounds can be found at the Internet address http://or.ingce.unibo.it/hvrp/.

The relevant data for the twelve HVRP instances considered are summarized in Table 2, which reports the following for each vehicle type k: (i) the capacity Q_k, (ii) the fixed cost F_k, (iii) the routing cost coefficient r_k and (iv) the upper bound U_k. The coefficient r_k is used to compute the edge cost d_{ij}^k as $d_{ij}^k = r_k e_{ij}$, where e_{ij} is the Euclidean distance between vertices i and j. To the best of our knowledge, neither exact nor heuristic algorithms have been proposed in the literature for solving these instances.

The instances for the different variants of the HVRP were obtained from the above instances as follows. For each variant of the HVRP we considered twelve instances that are obtained by changing some of the data of the HVRP instances as follows:

- FSMF: for each $k \in M$ set $U_k = n$ and $r_k = 1.0$;
- FSMFD: for each $k \in M$ set $U_k = n$;
- HD: for each $k \in M$ set $F_k = 0$;
- FSMD: for each $k \in M$ set $F_k = 0$ and $U_k = n$.

(b) *SDVRP instances*

We considered thirteen test instances involving up to 108 customers proposed by Nag et al. [22], Chao et al. [5] and Cordeau and Laporte [9]. The data of all instances can be found at the Internet address http://neumann.hec.ca/chairedistributique/data/sdvrp, while the best upper bounds were taken from Pisinger and Ropke [25].

(c) *MDVRP instances in the literature*

For the MDVRP, we considered a set of nine instances used by Cordeau et al. [8] involving up to 160 customers and 5 depots. The data of all instances can be found at the Internet address http://neumann.hec.ca/chairedistributique/data/mdvrp, while the best upper bounds were taken from Pisinger and Ropke [25].

(d) *New MDVRP instances*

Table 2 Instance data for the HVRP

No.	n	m	Veh. type 1				Veh. type 2				Veh. type 3				Veh. type 4				Veh. type 5				Veh. type 6			
			Q_1	F_1	r_1	U_1	Q_2	F_2	r_2	U_2	Q_3	F_3	r_3	U_3	Q_4	F_4	r_4	U_4	Q_5	F_5	r_5	U_5	Q_6	F_6	r_6	U_6
3	20	5	20	20	1.0	20	30	35	1.1	20	40	50	1.2	20	70	120	1.7	20	120	225	2.5	20				
4	20	3	60	1000	1.0	20	80	1500	1.1	20	150	3000	1.4	20												
5	20	5	20	20	1.0	20	30	35	1.1	20	40	50	1.2	20	70	120	1.7	20	120	225	2.5	20				
6	20	3	60	1000	1.0	20	80	1500	1.1	20	150	3000	1.4	20												
13	50	6	20	20	1.0	4	30	35	1.1	2	40	50	1.2	4	70	120	1.7	4	120	225	2.5	2	200	400	3.2	1
14	50	3	120	1000	1.0	4	160	1500	1.1	2	300	3500	1.4	1												
15	50	3	50	100	1.0	4	100	250	1.6	3	160	450	2.0	2												
16	50	3	40	100	1.0	2	80	200	1.6	4	140	400	2.1	3												
17	75	4	50	25	1.0	4	120	80	1.2	4	200	150	1.5	2	350	320	1.8	1								
18	75	6	20	10	1.0	4	50	35	1.3	4	100	100	1.9	2	150	180	2.4	2	250	400	2.9	1	400	800	3.2	1
19	100	3	100	500	1.0	4	200	1200	1.4	3	300	2100	1.7	3												
20	100	3	60	100	1.0	6	140	300	1.7	4	200	500	2.0	3												

In order to further evaluate the performance of our exact algorithm, we generated a new set of eight MDVRP instances. The instances were generated by considering the two CVRP instances M-n151-k12 and M-n200-k16 (available at the internet address http://branchandcut.org/VRP/data). More precisely, for each of the two CVRP instances, we generated four MDVRP instances by considering as the set of customers the original set of customers and by varying the vehicle capacity Q and the number of depots p. In particular, we considered instances with vehicle capacity $Q \in \{80, 100\}$ and number of depots $p \in \{3, 4\}$. The depot coordinates for the instances with $p = 3$ depots are $(20, 40)$, $(45, 25)$ and $(45, 55)$, while the depot coordinates for the instances with $p = 4$ depots are $(20, 20)$, $(45, 20)$, $(20, 50)$ and $(45, 50)$.

For the SDVRP and MDVRP instances, all the computations were performed by using real-valued Euclidean distances.

Based on the results of several preliminary experiments to identify good parameter settings for our method, we decided to use the following parameter settings:

- in procedures H^1 and DP^1: $Maxt1 = 300$, $\varepsilon = 2.0$;
- for the reduction tests based on H^1: $Maxt1r = 100$;
- in procedures H^2 and DP^2: $Maxt2 = 25$, $\varepsilon = 1.0$, $Maxt3 = 400$, $\Delta^{\min} = 5,000$, $\Delta^a = 200$;
- in procedure CG: $\Delta^a = 200$;
- in order to avoid *out-of-memory* errors, we impose that the size of the final set $\hat{\mathscr{R}}$ does not exceed the limit of 500, 000 routes (i.e., $\Delta = 500,000$ in procedure GENROUTE, see Appendix B);
- in the exact method described in Sect. 3 we disabled the separation of all the cuts embedded into CPLEX, because their use increases the overall computing time.

Tables 4, 5, 6, 7, 8, 9, 10, 11, 12, 13 of this section report the following columns:

z^* : cost of the optimal solution or cost of the best solution found by our exact method; values printed in bold indicate that the solution cost found is less than the cost of the best upper bound known;
z_{UB} : best upper bound value available in the literature;
$\%LHx$: percentage ratio of lower bound LHx computed as $100.0\ LHx/z^*$;
$\%LDx$: percentage ratio of lower bound LDx computed as $100.0\ LDx/z^*$;
$\%LCG$: percentage ratio of lower bound LCG computed as $100.0\ LCG/z^*$;
$\%LB$: percentage ratio of the best lower bound obtained computed as $\%LB = \max\{\%LCG, \%LD2\}$; the best of the values $\%LCG$ and $\%LD2$ are printed in bold;
t_{BM} : time in seconds spent by the bounding procedure for computing the final lower bound;
t_{CPX} : total computing time in seconds spent by CPLEX to solve the final $\hat{F}$ problem;
t_{EM} : total computing time in seconds of the exact method; t_{EM} includes also the time spent by procedure GENROUTE for generating the final route subsets $\hat{\mathscr{R}}_k, \forall k \in M$;

$\%LB_P$: percentage ratio of the lower bound LB_P produced by Pessoa et al. [24] computed as 100.0 LB_P/z^*;

T_{LB_P},T_P : time in seconds for computing LB_P and total computing time of the exact method of Pessoa et al. [24], respectively. The computing times are in seconds of a Pentium Core 2 Duo 2.13 GHz;

$\%LB_C$: percentage ratio of the lower bound LB_C produced by Choi and Tcha [6] computed as 100.0 LB_C/z^*.

$\#Inst$: total number of instances considered for the corresponding problem variant;

$\#Opt$: number of instances solved to optimality by the corresponding exact method;

The last line of each table reports:

- the average percentage ratios and the average running times in seconds of the lower bounds computed over all instances;
- the average running times in seconds computed over all instances solved to optimality. For the FSMF variant, the average is computed over all instances solved to optimality by both the method of Pessoa et al. [24] and our exact method.

For HVRP instances and for the new MDVRP instances, for which upper bound values were not available, the upper bounds were set equal to the solution obtained with our implementation of the heuristic algorithms proposed by Choi and Tcha [6] and by Cordeau et al. [8], respectively.

For all the instances of variants HVRP, FSMF, FSMFD, HD, FSMD and SDVRP except FSMF instance 17, a time limit of 7,200 s was imposed to CPLEX. For MDVRP instances a time limit of 20,000 s was imposed to CPLEX. Due to the final cardinality of the subsets of routes $\hat{\mathscr{R}}$, $\forall k \in M$, instance 17 of the FSMF variant was solved to optimality by running CPLEX using the depth-first search strategy as the node selection strategy. This strategy let CPLEX to avoid memory overflows but at a higher computing time.

Table 3 reports the results obtained on FSMF by the reduction tests described in Sects. 4.1.2 and 5.4. More precisely, for each instance, the table reports the lower and upper bounds on the number of vehicles of each type together with the number of vehicles used in the corresponding optimal solution.

Table 4 shows the effectiveness of bounding procedures DP^1 and DP^2 on FSMF instances. The table reports under heading "Bounding procedure" the results obtained by using the procedure described in Sect. 3, and under the heading "H^1,H^2,CG" the results obtained without using bounding procedure DP^1 and DP^2 and the associated reduction tests.

Table 3 shows that the reduction tests were able to define both tight lower and upper bounds on the number of vehicles of each vehicle type used in the optimal solutions. In particular, for four out of twelve instances, the reductions tests were able to exactly define the composition of the vehicle fleet of the optimal solution. Moreover, as shown by Table 4, the final procedure CG takes advantage of the reductions made by DP^1 and DP^2, as testified by the average values $\%LCG$ computed with and without using DP^1 and DP^2, which are equal to 99.7 and 99.1, respectively. Moreover, the results obtained for instance 15 of Table 4, show that $LD1$ and $LD2$ are not dominated by lower bound LCG.

Table 3 Effectiveness of reduction procedures on FSMF

No.	n	m	Lower bounds						Upper bounds						Optimal solution					
			L_1	L_2	L_3	L_4	L_5	L_6	U_1	U_2	U_3	U_4	U_5	U_6	1	2	3	4	5	6
3	20	5	0	1	0	0	1		2	4	6	3	2		1	2	1	0	2	
4	20	3	6	0	0				6	0	0				6	0	0			
5	20	5	0	0	0	0	1		2	4	3	3	3		1	1	0	1	2	
6	20	3	6	0	0				6	0	0				6	0	0			
13	50	6	0	0	0	0	0	1	6	9	12	6	6	4	2	2	2	0	0	4
14	50	3	7	1	0				7	1	0				7	1	0			
15	50	3	1	1	0				14	7	3				10	3	0			
16	50	3	0	4	1				2	8	3				0	8	1			
17	75	4	1	0	2	0			5	6	6	1			1	1	6	0		
18	75	6	0	0	0	0	0	0	8	15	13	9	0	0	2	5	5	4	0	0
19	100	3	15	0	0				15	0	0				15	0	0			
20	100	3	1	1	0				22	10	1				–	–	–	–	–	–

Table 4 Effectiveness of bounding procedures DP^1 and DP^2 on FSMF

No.	n	m	z^*	H^1,H^2,CG		Bounding procedure			
				$\%LCG$	T_{BM}	$\%LD1$	$\%LD2$	$\%LCG$	T_{BM}
3	20	5	961.03	100.0	3.1	98.8	0.0	100.0	5.3
4	20	3	6,437.33	99.2	3.9	99.5	99.5	100.0	5.4
5	20	5	1,007.05	99.6	3.9	98.0	0.0	99.6	6.6
6	20	3	6,516.47	99.2	3.3	99.6	99.7	100.0	7.8
13	50	6	2,406.36	100.0	29.5	98.9	0.0	100.0	54.0
14	50	3	9,119.03	96.3	83.0	99.9	99.9	100.0	67.9
15	50	3	2,586.37	99.1	92.0	99.4	99.5	99.1	52.9
16	50	3	2,720.43	99.5	21.9	98.8	0.0	99.7	31.9
17	75	4	1,734.53	99.3	292.3	98.7	0.0	99.3	296.7
18	75	6	2,369.65	99.7	277.5	98.4	0.0	99.7	335.3
19	100	3	8,661.81	97.6	390.8	99.7	99.7	99.9	421.1
20	100	3	4,039.49	99.4	462.7	98.5	0.0	99.4	479.3
				99.1	138.7	99.0	99.7	99.7	147.0

Tables 5, 6, 7, 8, 9, 10, 11, 12 report the lower bounds obtained by the bounding procedure and the results of the exact method for the different variants considered.

On the FSMF variant, a comparison between the lower bound LB_P of Pessoa et al. and our lower bound LB shows that LB is superior to lower bound LB_P in all instances. Taking the computer we used and the computer used by Pessoa et al. into account in examining the computational results, Table 6 indicates that our bounding procedure is on average about ten times faster than the one of Pessoa et al. Our exact method solved to optimality eleven out of twelve instances. Two more instances were solved to optimality with respect to the instances solved by the method of Pessoa et al. On the instances solved to optimality by both methods, the new exact method is on average much faster than the algorithm of Pessoa et al.

On FSMFD and FSMD variants, Tables 7 and 9 show that our lower bound is superior to lower bound LB_C computed by Choi and Tcha [6] in all instances.

For the other variants, for which neither lower bounds nor exact algorithms have been presented in the literature, our exact method was able to solve to optimality several test instances.

Table 13 summarizes the results obtained over all variants considered. The unified solution method proposed in this paper was able to solve to optimality 74 out of 86 instances considered. The table shows that the average percentage ratio of lower bound LB computed over all 86 instances is equal to 99.5 and that the corresponding average computing time is 176.2 s. These results show the effectiveness of our bounding procedure.

Tables 5, 6, 7, 8, 9, 10, 11, 12 show that the new algorithm can solve to optimality all instances of all problem types involving up to 75 customers and several instances involving 100 and 199 customers.

Table 5 Computational results for HVRP instances

No.	n	m	z_{UB}	z^*	Bounding procedure							Exact method		
					$\%LH1$	$\%LH2$	$\%LD1$	$\%LD2$	$\%LCG$	$\%LB$	T_{BM}	$\|\hat{\mathscr{R}}\|$	T_{CPX}	T_{EM}
3	20	5	1,144.22	1,144.22	99.5	99.6	97.5	0.0	**100.0**	100.0	5.9	39	0.0	5.9
4	20	3	6,437.33	6,437.33	98.9	99.0	99.5	99.5	**100.0**	100.0	6.9	1,022	0.0	6.9
5	20	5	1,322.26	1,322.26	98.9	99.0	97.4	0.0	**99.1**	99.1	6.5	92	0.0	6.5
6	20	3	6,516.47	6,516.47	99.0	99.0	99.6	99.7	**100.0**	100.0	9.2	524	0.0	9.2
13	50	6	3,185.09	3,185.09	98.2	98.9	98.2	0.0	**99.8**	99.8	51.4	4,459	0.3	51.8
14	50	3	10,107.53	10,107.53	95.2	95.2	99.8	99.9	**99.9**	99.9	105.5	12,746	87.3	195.4
15	50	3	3,065.29	3,065.29	95.5	95.8	98.9	99.2	**99.6**	99.6	61.6	13,508	93.3	163.0
16	50	3	3,265.41	3,265.41	98.4	98.7	98.7	98.9	**99.6**	99.6	55.5	12,437	17.4	79.6
17	75	4	2,076.96	2,076.96	98.4	98.6	98.7	98.9	**99.4**	99.4	277.6	85,416	1,372.0	1,821.3
18	75	6	3,743.58	3,743.58	97.4	97.7	98.9	99.1	**99.5**	99.5	544.9	23,446	298.3	963.6
19	100	3	10,425.48	**10,423.32** [a]	94.9	95.1	99.7	99.8	**99.9**	99.9	1,090.7	>500,000	4,500.1	6,234.0
20	100	3	4,806.69	4,806.69 [a]	97.0	97.3	97.9	98.2	**98.5**	98.5	482.1	>500,000	5,349.9	6,446.8
					97.6	97.8	98.7	99.3	99.6	99.6	224.8			259.9

The values printed in bold under column z^* indicate that the solution cost found is less than the cost of the best upper bound known

The best of the values $\%LD2$ and $\%LCG$ are printed in bold

[a] Optimality not proved

Table 6 Computational results for FSMF instances

No.	n	m	z_{UB}	z^*	Pessoa et al. (2007)			Bounding procedure							Exact method		
					$\%LB_P$	T_{LB_P}	T_P	$\%LH1$	$\%LH2$	$\%LD1$	$\%LD2$	$\%LCG$	$\%LB$	T_{BM}	$\lvert\hat{\mathscr{R}}\rvert$	T_{CPX}	T_{EM}
3	20	5	961.03	961.03	100.0	3.2	3.0	99.0	99.4	98.8	0.0	**100.0**	100.0	5.3	74	0.0	5.3
4	20	3	6,437.33	6,437.33	100.0	6.6	6.6	98.9	99.0	99.5	99.5	**100.0**	100.0	5.4	998	0.1	5.5
5	20	5	1,007.05	1,007.05	99.4	14.3	14.3	98.1	98.5	98.0	0.0	**99.6**	99.6	6.6	818	0.5	7.1
6	20	3	6,516.47	6,516.47	100.0	8.4	8.4	99.0	99.0	99.6	99.7	**100.0**	100.0	7.8	524	0.0	7.8
13	50	6	2,406.36	2,406.36	99.8	152.8	152.8	99.4	99.8	98.9	0.0	**100.0**	100.0	54.0	691	0.1	54.1
14	50	3	9,119.03	9,119.03	99.9	335.9	335.9	96.0	96.1	99.9	99.9	**100.0**	100.0	67.9	4,787	2.2	70.8
15	50	3	2,586.37	2,586.37	99.5	352.2	352.2	98.4	98.7	99.4	**99.5**	99.1	99.5	52.9	13,760	347.7	401.1
16	50	3	2,720.43	2,720.43	99.5	358.3	358.3	98.7	99.1	98.8	0.0	**99.7**	99.7	31.9	7,395	6.8	50.0
17	75	4	1,744.83	**1,734.53**	99.0	4,729.2	–	98.6	98.8	98.7	0.0	**99.3**	99.3	296.7	462,551	86,516.4	90,624.1
18	75	6	2,371.49	**2,369.65**	99.2	5,154.3	84,286.0	98.9	99.1	98.4	0.0	**99.7**	99.7	335.3	13,295	86.0	527.3
19	100	3	8,661.81	8,661.81	99.9	2,938.1	–	97.3	97.4	99.7	99.7	**99.9**	99.9	421.1	110,823	6,825.1	7,657.9
20	100	3	4,039.49	4,039.49 [a]	99.2	3,066.7	–	98.9	99.1	98.5	0.0	**99.4**	99.4	479.3	>500,000	3,779.0	5,472.1
					99.6	1,426.7	9,501.9	98.4	98.7	99.0	99.7	99.7	99.8	147.0			125.4

The values printed in bold under column z^* indicate that the solution cost found is less than the cost of the best upper bound known

The best of the values $\%LD2$ and $\%LCG$ are printed in bold

[a] Optimality not proved

Table 7 Computational results for FSMFD instances

No.	n	m	z_{UB}	z^*	Choi and Tcha (2006)	Bounding procedure							Exact method		
					$\%LB_C$	$\%LH1$	$\%LH2$	$\%LD1$	$\%LD2$	$\%LCG$	$\%LB$	T_{BM}	$\lvert\hat{\mathscr{R}}\rvert$	T_{CPX}	T_{EM}
3	20	5	1,144.22	1,144.22	99.5	99.5	99.6	97.5	0.0	**100.0**	100.0	6.3	37	0.0	6.3
4	20	3	6,437.33	6,437.33	98.9	98.9	99.0	99.5	99.5	**100.0**	100.0	6.9	998	0.1	7.0
5	20	5	1,322.26	1,322.26	98.9	98.9	98.9	97.4	0.0	**99.1**	99.1	6.5	93	0.1	6.5
6	20	3	6,516.47	6,516.47	99.0	99.0	99.0	99.6	99.7	**100.0**	100.0	9.5	524	0.0	9.5
13	50	6	2,964.65	2,964.65	99.8	99.8	99.8	97.6	0.0	**100.0**	100.0	49.4	91	0.0	49.4
14	50	3	9,126.90	9,126.90	95.9	95.9	96.0	99.9	99.9	**100.0**	100.0	84.5	4,734	1.4	86.2
15	50	3	2,634.96	2,634.96	98.6	98.6	98.9	99.1	99.3	**99.8**	99.8	53.7	4,544	0.5	54.2
16	50	3	3,168.92	3,168.92	98.3	98.3	98.5	98.6	98.8	**99.0**	99.0	54.1	21,733	846.0	910.4
17	75	4	2,023.61	**2,004.48**	98.8	98.8	99.0	98.2	0.0	**99.8**	99.8	249.6	5,856	10.4	272.7
18	75	6	3,147.99	3,147.99	99.4	99.4	99.6	98.3	0.0	**99.9**	99.9	326.8	1,093	0.2	327.0
19	100	3	8,664.29	**8,661.81**	97.3	97.3	97.4	99.7	99.7	**99.9**	99.9	625.1	96,515	6,292.7	7,355.7
20	100	3	4,154.49	4,154.49 [a]	98.3	98.3	98.5	98.9	**99.0**	98.8	99.0	249.2	>500,000	5,091.2	5,460.0
					98.6	98.6	98.7	98.7	99.4	99.7	99.7	143.5			825.9

The values printed in bold under column z^* indicate that the solution cost found is less than the cost of the best upper bound known

The best of the values $\%LD2$ and $\%LCG$ are printed in bold

[a] Optimality not proved

Table 8 Computational results for HD instances

No.	n	m	z_{UB}	z^*	Bounding procedure				Exact method		
					$\%LH1$	$\%LH2$	$\%LCG$	T_{BM}	$\|\hat{\mathscr{R}}\|$	T_{CPX}	T_{EM}
13	50	6	1,517.84	1,517.84	98.5	99.3	100.0	28.3	409	0.1	28.4
14	50	3	607.53	607.53	97.3	98.0	98.6	50.4	16,543	420.8	481.6
15	50	3	1,015.29	1,015.29	96.7	97.7	98.6	34.9	16,779	413.1	474.7
16	50	3	1,144.94	1,144.94	97.5	98.1	99.4	38.9	8,081	34.1	80.9
17	75	4	1,061.96	1,061.96	97.4	98.1	98.8	127.1	92,013	1,612.4	1,876.8
18	75	6	1,823.58	1,823.58	98.3	98.8	99.5	255.9	10,136	134.1	413.7
19	100	3	1,117.51	1,117.51 [a]	97.1	98.3	99.1	376.8	>500,000	7,200.0	9,081.4
20	100	3	1,534.17	1,534.17	98.2	99.2	99.5	118.6	24,851	400.3	597.6
					97.6	98.4	99.2	128.9			564.8

[a] Optimality not proved

Table 9 Computational results for FSMD instances

No.	n	m	z_{UB}	z^*	Choi and Tcha (2006)	Bounding procedure				Exact method		
					$\%LB_C$	$\%LH1$	$\%LH2$	$\%LCG$	T_{BM}	$\|\hat{\mathscr{R}}\|$	T_{CPX}	T_{EM}
3	20	5	623.22	623.22	98.9	98.9	99.0	99.5	3.0	48	0.1	3.0
4	20	3	387.18	387.18	97.4	97.3	97.9	99.7	3.6	214	0.1	3.6
5	20	5	742.87	742.87	99.6	99.6	99.6	100.0	2.0	26	0.0	2.0
6	20	3	415.03	415.03	98.0	98.0	98.2	100.0	3.1	276	0.0	3.1
13	50	6	1,491.86	1,491.86	98.5	98.5	99.5	100.0	28.6	322	0.0	28.7
14	50	3	603.21	603.21	96.5	96.5	97.2	98.0	56.9	50,862	949.9	1,070.4
15	50	3	999.82	999.82	97.8	97.8	98.6	99.6	17.3	5,737	2.2	19.5
16	50	3	1,131.00	1,131.00	97.8	97.8	98.1	99.4	17.8	7,792	26.0	45.2
17	75	4	1,038.60	1,038.60	98.4	98.4	98.7	99.3	144.9	28,935	433.6	679.2
18	75	6	1,801.40	**1,800.80**	98.8	98.8	99.2	99.3	228.9	22,469	675.2	955.9
19	100	3	1,105.44	1,105.44	97.8	97.7	98.7	99.6	366.5	25,366	348.6	949.3
20	100	3	1,530.43	1,530.43	98.1	98.1	99.1	99.5	110.0	34,758	785.3	1,006.7
					98.1	98.1	98.6	99.5	81.9			397.2

The values printed in bold under column z^* indicate that the solution cost found is less than the cost of the best upper bound known

Finally, it is also worth mentioning that not only is this algorithm a generalization of the method described by Baldacci et al. [3] for the CVRP but, when applied to CVRP instances, it produces the same computational results as those reported by Baldacci et al. [3].

Table 10 Computational results for SDVRP instances

No.	n	m	z_{UB}	z^*	Bounding procedure				Exact method		
					$\%LH1$	$\%LH2$	$\%LCG$	T_{BM}	$\lvert\hat{\mathscr{R}}\rvert$	T_{CPX}	T_{EM}
p07	27	3	391.30	391.30	100.0	100.0	100.0	0.0	0	0.0	0.0
p01	50	3	640.32	640.32	97.7	99.0	99.8	15.4	1,516	0.2	15.6
p02	50	2	598.10	598.10	97.2	97.7	98.7	20.3	15,122	334.9	360.6
p08	54	3	664.46	664.46	100.0	100.0	100.0	14.3	0	0.0	14.3
p13	54	3	1,194.18	1,194.18	91.2	96.9	98.8	109.0	12,875	26.5	164.5
p03	75	3	957.04	**954.32**	97.4	98.4	99.0	36.5	80,130	3,048.5	3,169.0
p04	75	2	854.43	854.43	97.4	98.1	99.4	84.5	17,276	1,075.7	1,194.3
p09	81	3	948.23	948.23	97.1	100.0	100.0	40.6	0	0.0	40.6
p05	100	3	1,003.57	1,003.57	94.7	98.8	99.2	453.2	149,765	2,185.4	2,966.3
p06	100	2	1,028.52	1,028.52[a]	94.0	97.5	97.9	438.9	>500,000	7,166.5	8,275.5
p23	100	3	803.29	803.29[a]	91.5	97.8	98.3	580.1	>500,000	3,688.9	5,970.5
p10	108	3	1,218.75	1,218.75[a]	94.9	99.0	99.0	138.2	>500,000	3,338.5	3,866.4
p14	108	3	1,960.62	1,960.62[a]	94.5	98.0	98.1	459.1	>500,000	3,673.6	4,600.3
					96.0	98.6	99.1	183.9			880.6

The values printed in bold under column z^* indicate that the solution cost found is less than the cost of the best upper bound known

[a] Optimality not proved

Table 11 Computational results for MDVRP instances

No.	n	m	z_{UB}	z^*	Bounding procedure				Exact method		
					$\%LH1$	$\%LH2$	$\%LCG$	T_{BM}	$\lvert\hat{\mathscr{R}}\rvert$	T_{CPX}	T_{EM}
p01	50	4	576.87	576.87	98.3	99.0	100.0	10.9	403	0.0	10.9
p02	50	4	473.53	473.53	96.3	98.9	99.7	54.2	1,788	0.2	54.4
p03	75	5	641.19	**640.65**	98.3	98.9	99.9	92.1	7,067	0.3	92.4
p12	80	2	1,318.95	1,318.95	94.4	98.5	98.6	345.7	37,494	80.7	463.7
p04	100	2	1,001.04	999.21	97.4	98.3	99.3	95.5	167,774	4,579.3	5,106.9
p05	100	2	751.26	751.26[a]	96.2	97.0	97.5	296.7	>500,000	1,021.0	1,647.6
p06	100	3	876.50	876.50	97.8	98.8	99.6	59.7	22,538	25.0	104.3
p07	100	4	881.97	881.97	96.5	98.3	99.5	187.3	23,536	53.0	294.4
p15	160	4	2,505.42	2,505.42[a]	95.1	99.2	99.2	1,652.9	56,926	25.2	2,147.9
					96.7	98.5	99.2	310.5			875.3

The values printed in bold under column z^* indicate that the solution cost found is less than the cost of the best upper bound known

[a] Optimality not proved

Table 12 Computational results for new MDVRP instances

Problem	n	m	Q	z_{UB}	z^*	Bounding procedure				Exact method		
						$\%LH1$	$\%LH2$	$\%LCG$	T_{BM}	$\lvert\hat{\mathscr{R}}\rvert$	T_{CPX}	T_{EM}
M-n151-k12	150	3	80	1,374.03	1,374.03	97.3	98.6	99.7	117.4	24,754	229.0	377.8
M-n151-k12	150	4	80	1,200.54	1,200.54	97.8	98.8	99.9	90.1	11,542	5.6	102.1
M-n151-k12	150	3	100	1,197.52	1,197.52	97.1	98.2	99.5	156.1	97,944	9,315.6	9,689.2
M-n151-k12	150	4	100	1,058.38	1,058.38	98.0	98.8	99.7	125.8	17,618	113.8	263.3
M-n200-k16	199	3	80	1,757.86	1,757.86[a]	97.3	98.5	99.5	266.6	270,938	20,000.0	21,213.3
M-n200-k16	199	4	80	1,535.04	**1,534.56**	97.8	99.0	99.8	126.3	15,641	73.8	208.1
M-n200-k16	199	3	100	1,511.35	1,511.35	96.9	98.4	99.6	294.4	224,302	17,124.6	18,091.2
M-n200-k16	199	4	100	1,347.19	1,347.19	96.7	98.6	99.7	339.3	41,430	449.9	946.6
						97.4	98.6	99.7	189.5			4,788.6

The values printed in bold under column z^* indicate that the solution cost found is less than the cost of the best upper bound known

[a] Optimality not proved

Table 13 Summary results

Variant	#Inst	Lower bounds					Exact methods			
		Pessoa et al. (2007)		Choi and Tcha (2006)	Our		Pessoa et al. (2007)		Our	
		$\%LB_P$	T_{LB_P}	$\%LB_C$	$\%LB$	T_{BM}	#Opt	T_P	#Opt	T_{EM}
HVRP	12	–	–	–	99.6	224.8	–	–	10	259.9
FSMF	12	99.6	1,426.7	98.4	99.8	147.0	9	9,501.9	11	125.4
FSMFD	12	–	–	98.6	99.7	143.5	–	–	11	825.9
HD	8	–	–	–	99.2	128.9	–	–	7	564.8
FSMD	12	–	–	98.1	99.5	81.9	–	–	12	397.2
SDVRP	13	–	–	–	99.1	183.9	–	–	9	880.6
MDVRP	9	–	–	–	99.2	310.5	–	–	7	875.3
MDVRP	8	–	–	–	99.7	189.5	–	–	7	4,788.6
	86				99.5	176.2			74	1,089.7

8 Conclusions

In this paper we described a unified exact method for solving an extended model of the well-known Capacitated Vehicle Routing Problem (CVRP), called the Heterogenous Vehicle Routing Problem (HVRP).

We designed an exact algorithm for the HVRP based on the set partitioning formulation and we performed extensive computational results over the main instances from the literature of the different variants of HVRP. Further, we produced new tight lower bounds for some variants of the HVRP for which neither lower bounds nor exact algorithms have been proposed in the literature. For the variants already studied in the literature, our computational results reveal that the new lower bounds are tighter than all other lower bounds proposed in the literature so far. The exact algorithm solved several test instances of all problem types considered, and can solve problems involving up to 199 customers.

Future research will investigate the generalization of the HVRP model described in this paper to deal with the complexity of real-world applications such as time-window constraints and multiple vehicle trips.

Acknowledgments The authors would like to express their special thanks to Prof. L. D. Bodin for his invaluable help and useful discussion. Thanks are also due to anonymous referees for detailed and helpful comments.

Appendix A: Proofs of lemmas and theorems

Lemma 1 (*see Sect.* 2.2) *The optimal solution cost $z(RP)$ of problem RP provides a valid lower bound on the HVRP for any solution β_{ik} of inequalities* (9).

Proof Consider a F solution $\overline{\mathbf{x}}$ of cost $\overline{z}(F)$ and let $\overline{\mathscr{J}}_k = \{\ell \in \mathscr{R}^k : \overline{x}^k_\ell = 1\}$ and $\overline{M} = \{k \in M : \sum_{\ell \in \mathscr{R}^k} \overline{x}^k_\ell \geq 1\}$.

Let $\overline{c}^k_\ell = c^k_\ell - \sum_{i \in R^k_\ell} \beta_{ik}$ be the reduced cost of route $\ell \in \mathscr{R}^k$, $k \in M$, with respect to the solution β_{ik} of inequalities (9) used in RP. In the following we show that

$$z(RP) + \sum_{k \in \overline{M}} \sum_{\ell \in \overline{\mathscr{J}}_k} \overline{c}^k_\ell \leq \overline{z}(F) \tag{44}$$

that implies $z(RP) \leq \overline{z}(F)$ as $\overline{c}^k_\ell \geq 0, \forall \ell \in \mathscr{R}^k, \forall k \in M$.

By using the definition of the reduced cost $\overline{c}^k_\ell$, we derive:

$$\sum_{k \in \overline{M}} \sum_{\ell \in \overline{\mathscr{J}}_k} \overline{c}^k_\ell = \sum_{k \in \overline{M}} \sum_{\ell \in \overline{\mathscr{J}}_k} c^k_\ell - \sum_{k \in \overline{M}} \sum_{\ell \in \overline{\mathscr{J}}_k} \sum_{i \in R^k_\ell} \beta_{ik}. \tag{45}$$

Let $\overline{V}_k = \{i \in R^k_\ell : \ell \in \overline{\mathscr{J}}_k\}$, $k \in \overline{M}$. As $\overline{\mathbf{x}}$ is a F solution, then each customer i belongs to one of the sets $\overline{V}_k$, $k \in \overline{M}$, therefore we have

$$\sum_{k \in \overline{M}} \sum_{\ell \in \overline{\mathscr{J}}_k} \sum_{i \in R^k_\ell} \beta_{ik} = \sum_{k \in \overline{M}} \sum_{i \in \overline{V}_k} \beta_{ik}. \tag{46}$$

From expressions (45) and (46) we obtain:

$$\sum_{k \in \overline{M}} \sum_{\ell \in \overline{\mathscr{J}}_k} \overline{c}^k_\ell = \sum_{k \in \overline{M}} \sum_{\ell \in \overline{\mathscr{J}}_k} c^k_\ell - \sum_{k \in \overline{M}} \sum_{i \in \overline{V}_k} \beta_{ik}. \tag{47}$$

Adding and subtracting the term $\sum_{k\in\overline{M}}|\overline{\mathscr{J}}_k|F_k$ to the right-hand-side of expression (47) we have:

$$\sum_{k\in\overline{M}}\sum_{\ell\in\overline{\mathscr{J}}_k}\overline{c}^k_\ell=\overline{z}(F)-\left(\sum_{k\in\overline{M}}\sum_{i\in\overline{V}_k}\beta_{ik}+\sum_{k\in\overline{M}}|\overline{\mathscr{J}}_k|F_k\right). \tag{48}$$

Notice that the second term of the right-hand-side of expression (48) corresponds to the cost (say $\tilde{z}(RP)$) of a feasible, but not necessarily optimal, RP solution $(\tilde{\boldsymbol{\xi}},\tilde{\mathbf{y}})$ that is derived from the F solution $\overline{\mathbf{x}}$ by setting:

$$\tilde{\xi}_{ik}=1,\ \forall i\in\overline{V}_k,\ \tilde{\xi}_{ik}=0,\ \forall i\in V\backslash\overline{V}_k,\ \tilde{y}_k=|\overline{\mathscr{J}}_k|,\ k\in\overline{M} \tag{49}$$

and

$$\tilde{\xi}_{ik}=0,\ \forall i\in V,\quad \text{and}\quad \tilde{y}_k=0,\ k\in M\backslash\overline{M}. \tag{50}$$

As $\tilde{z}(RP)\geq z(RP)$, from expression (48) we obtain

$$\sum_{k\in\overline{M}}\sum_{\ell\in\overline{\mathscr{J}}_k}\overline{c}^k_\ell\leq\overline{z}(F)-z(RP) \tag{51}$$

that corresponds to inequality (44). □

Corollary 1 (*see Sect.* 3) *Let $z(RP)$ be the cost of an optimal RP solution obtained for a given solution β_{ik} of inequalities* (9). *Any HVRP solution of cost smaller than z_{UB} cannot contain any route $\ell\in\mathscr{R}^k$, $k\in M$, such that*

$$c^k_\ell-\sum_{i\in R^k_\ell}\beta_{ik}\geq z_{UB}-z(RP). \tag{52}$$

Proof It follows directly from Lemma 1. □

Proof of Theorem 1 (*see Sect.* 4). For a given route $\ell\in\mathscr{R}^k$ of a given vehicle type $k\in M$, we have that $\ell\in\mathscr{R}^k_i$, $\forall i\in R^k_\ell$. Thus, the following inequalities hold:

$$b_{ik}\leq q_i\frac{c^k_\ell+F_k-\lambda(R^k_\ell)-\mu_k}{q(R^k_\ell)},\quad \forall i\in R^k_\ell. \tag{53}$$

From expression (20a) and inequalities (53) we derive:

$$\begin{aligned}\sum_{i\in R^k_\ell}u_i&\leq\sum_{i\in R^k_\ell}q_i\frac{c^k_\ell+F_k-\lambda(R^k_\ell)-\mu_k}{q(R^k_\ell)}+\sum_{i\in R^k_\ell}\lambda_i\\&=c^k_\ell+F_k-\mu_k,\end{aligned} \tag{54}$$

that corresponds to the dual constraint (6), as $v_k=\mu_k$ according to expression (20b). □

Proof of Theorem 2 (*see Sect.* 5). Consider a route $\ell \in \mathscr{R}^k$ of a given vehicle type $k \in M$. Since $\ell \in \mathscr{R}_i^k$, $\forall i \in R_\ell^k$, from expressions (24) we derive:

$$\beta_{ik} \leq q_i \frac{c_\ell^k - \lambda(R_\ell^k)}{q(R_\ell^k)} + \lambda_i, \quad \forall i \in R_\ell^k, \tag{55}$$

and, adding inequalities (55) for all $i \in R_\ell^k$, we obtain:

$$\sum_{i \in R_\ell^k} \beta_{ik} \leq \sum_{i \in R_\ell^k} q_i \frac{c_\ell^k - \lambda(R_\ell^k)}{q(R_\ell^k)} + \lambda(R_\ell^k) = c_\ell^k. \tag{56}$$

□

Appendix B: Procedure GENROUTE

GENROUTE is a procedure for generating routes that has been described in Baldacci et al. [3].

This procedure is used for generating the initial sets $\overline{\mathscr{R}}^k$, $\forall k \in M$, required by H^2 and DP^2, the sets $\mathscr{N}^k$, $\forall k \in M$, used by H^2, CG and DP^2 and the sets $\hat{\mathscr{R}}^k$, $\forall k \in M$, required by the exact method described in Sect. 3.

Let $\hat{\mathbf{u}}$, $\hat{\mathbf{w}}$ and $\hat{\mathbf{g}}$ be three vectors of marginal costs associated to customer V, with subset $\hat{\mathscr{S}}$ of capacity constraints (16) and with subset $\hat{\mathscr{C}}$ of clique inequalities (17). Moreover, let $\hat{v}_k$ be a marginal cost associated to vehicle type $k \in M$. For a given vehicle type $k \in M$, define the reduced cost $\hat{c}_\ell^k$ of route $\ell \in \mathscr{R}^k$ with respect to $\hat{\mathbf{u}}$, $\hat{\mathbf{w}}$, $\hat{\mathbf{g}}$ and $\hat{v}_k$ as:

$$\hat{c}_\ell^k = c_\ell^k - \sum_{i \in R_\ell^k} \hat{u}_i - \hat{v}_k - \sum_{S \in \hat{\mathscr{S}}} b_\ell^k(S)\hat{w}_S - \sum_{C \in \hat{\mathscr{C}}_\ell^k} \hat{g}_C, \tag{57}$$

where $\hat{\mathscr{C}}_\ell^k = \{C \in \hat{\mathscr{C}} : \exists i \in C$ such that $k(i) = k$ and $\ell(i) = \ell\}$ and $b_\ell^k(S) = 1$, $\forall \ell \in \mathscr{R}^k$, such that $R_\ell^k \cap S \neq \emptyset$ and $b_\ell^k(S) = 0$, otherwise.

Given the vectors $\hat{\mathbf{u}}$, $\hat{\mathbf{w}}$, $\hat{\mathbf{g}}$, the cost $\hat{v}_k$ for a selected vehicle type k and two user defined parameters γ and Δ, GENROUTE produces for a vehicle type k the largest subset $\mathscr{B}^k \subseteq \mathscr{R}^k$ satisfying the following conditions:

$$\left.\begin{array}{l} (a)\ \max\limits_{\ell \in \mathscr{B}^k} \{\hat{c}_\ell^k\} \leq \min\limits_{\ell \in \mathscr{R}^k \setminus \mathscr{B}^k} \{\hat{c}_\ell^k\} \\ (b)\ |\mathscr{B}^k| \leq \Delta \\ (c)\ \max\limits_{\ell \in \mathscr{B}^k} \{\hat{c}_\ell^k\} \leq \gamma. \end{array}\right\} \tag{58}$$

In order to generate the sets $\mathscr{B}^k$, $\forall k \in M$, GENROUTE must be executed m times, one time for each $k \in M$. Parameters γ and Δ permit GENROUTE to generate the different route sets required by H^2, CG, DP^2 and the exact method as follows:

- *The initial core problem $\overline{\mathscr{R}}^k$, $\forall k \in M$, of H^2.*
  Define $\hat{\mathbf{u}} = \mathbf{u}^1$, $\hat{\mathbf{w}} = \mathbf{0}$, $\hat{\mathbf{g}} = \mathbf{0}$, $\gamma = \infty$ and $\Delta = \Delta^{\min}$. For a given $k \in M$, set $\hat{v}^k = v_k^1$.
- *$\mathscr{N}^k$, $\forall k \in M$, required by H^2.*
  Define $\hat{\mathbf{u}} = \mathbf{u}^*$, $\hat{\mathbf{w}} = \mathbf{0}$, $\hat{\mathbf{g}} = \mathbf{0}$, $\gamma = 0$ and $\Delta = \Delta^a$. For a given $k \in M$, set $\hat{v}^k = v_k^*$.
- *The initial core problem $\overline{\mathscr{R}}^k$, $\forall k \in M$, of DP^2.*
  Define $\hat{\mathbf{w}} = \mathbf{0}$, $\hat{\mathbf{g}} = \mathbf{0}$, $\gamma = \infty$ and $\Delta = \Delta^{\min}$. For a given $k \in M$, set $\hat{u}_i = \beta_{ik}^1$, $\forall i \in V$, and $\hat{v}_k = 0$.
- *$\mathscr{N}^k$, $\forall k \in M$, required by DP^2.*
  Define $\hat{\mathbf{w}} = \mathbf{0}$, $\hat{\mathbf{g}} = \mathbf{0}$, $\gamma = 0$ and $\Delta = \Delta^a$. For a given $k \in M$, set $\hat{u}_i = \overline{\beta}_{ik}$, $\forall i \in V$, and $\hat{v}_k = 0$.
- *$\mathscr{N}^k$, $\forall k \in M$, required by CG.*
  Define $\hat{\mathbf{u}} = \overline{\mathbf{u}}$, $\hat{\mathbf{w}} = \overline{\mathbf{w}}$, $\hat{\mathbf{g}} = \overline{\mathbf{g}}$, $\gamma = 0$ and $\Delta = \Delta^a$. For a given $k \in M$, set $\hat{v}^k = \overline{v}_k$.
- *$\hat{\mathscr{R}}_k$, $k \in M$, required by the exact method.*
  We have two cases:
  (i) $LD2 > LCG$. Define $\hat{\mathbf{w}} = \mathbf{0}$, $\hat{\mathbf{g}} = \mathbf{0}$, $\gamma = z_{UB} - LD2$ and $\Delta = \infty$. For a given $k \in M$, set $\hat{u}_i = \beta_{ik}^2$, $\forall i \in V$, and $\hat{v}_k = 0$.
  (ii) $LD2 \leq LCG$. Define $\hat{\mathbf{u}} = \mathbf{u}^3$, $\hat{\mathbf{w}} = \mathbf{w}^3$, $\hat{\mathbf{g}} = \mathbf{0}$, $\gamma = z_{UB} - LCG$ and $\Delta = \infty$. For a given $k \in M$, set $\hat{v}_k = v_k^3$.

References

1. Baldacci, R., Battarra, M., Vigo, D.: Routing a heterogeneous fleet of vehicles. In: Golden, B., Raghavan, S., Wasil, E. (eds.) The Vehicle Routing Problem: Latest Advances and New Challenges, vol. 43. Springer, Heidelberg (2008)
2. Baldacci, R., Bodin, L.D., Mingozzi, A.: The multiple disposal facilities and multiple inventory locations rollon-rolloff vehicle routing problem. Comput. Oper. Res. **33**, 2667–2702 (2006)
3. Baldacci, R., Christofides, N., Mingozzi, A.: An exact algorithm for the vehicle routing problem based on the set partitioning formulation with additional cuts. Mathematical Programming Ser. A (2007). http://dx.doi.org/10.1007/s10107-007-0178-5
4. Baldacci, R., Hadjiconstantinou, E.A., Mingozzi, A.: An exact algorithm for the capacitated vehicle routing problem based on a two-commodity network flow formulation. Oper. Res. **52**, 723–738 (2004)
5. Chao, I.M., Golden, B.L., Wasil, E.A.: A computational study of a new heuristic for the site-dependent vehicle routing problem. INFOR **37**(3), 319–336 (1999)
6. Choi, E., Tcha, D.W.: A column generation approach to the heterogeneous fleet vehicle routing problem. Comput. Oper. Res. **34**, 2080–2095 (2007)
7. Christofides, N., Mingozzi, A., Toth, P.: Exact algorithms for the vehicle routing problem based on spanning tree and shortest path relaxations. Math. Program. **10**, 255–280 (1981)
8. Cordeau, J.F., Gendreau, M., Laporte, G.: A tabu search heuristic for periodic and multi-depot vehicle routing problems. Networks **30**, 105–119 (1997)
9. Cordeau, J.F., Laporte, G.: A tabu search algorithm for the site dependent vehicle routing problem with time windows. INFOR **39**, 292–298 (2001)
10. Cordeau, J.F., Laporte, G., Savelsbergh, M.W.P., Vigo, D.: Vehicle routing. In: Barnhart, C., Laporte, G. Transportation, Handbooks in Operations Research and Management Science, vol. 14, pp. 367–428. Elsevier, Amsterdam (2007)
11. CPLEX: ILOG CPLEX 10.1 callable library. ILOG (2007)
12. Crevier, B., Cordeau, J.F., Laporte, G.: The multi-depot vehicle routing problem with inter-depot routes. Eur. J. Oper. Res. **176**, 756–773 (2007)

13. Fukasawa, R., Longo, H., Lysgaard, J., Poggi de Aragão, M., Reis, M., Uchoa, E., Werneck, R.: Robust branch-and-cut-and-price for the capacitated vehicle routing problem. Math. Program. Ser. A **106**, 491–511 (2006)
14. Gendreau, M., Laporte, G., Potvin, J.Y. : Metaheuristics for the capacitated VRP. In: Toth, P., Vigo, D. (eds.) The Vehicle Routing Problem, vol. 9, pp. 129–154. SIAM Monogr. Discrete Math. Appl., Philadelphia (2002)
15. Golden, B.L., Assad, A.A., Levy, L., Gheysens, F.G.: The fleet size and mix vehicle routing problem. Comput. Oper. Res. **11**, 49–66 (1984)
16. Laporte, G., Nobert, Y., Arpin, D.: Optimal solutions to capacitated multi depot vehicle routing problems. Congressus Numerantium **44**, 283–292 (1984)
17. Laporte, G., Nobert, Y., Taillefer, S.: Solving a family of multi-depot vehicle routing and location-routing problems. Transp. Sci **22**, 161–172 (1988)
18. Laporte, G., Semet, F. : Classical heuristics for the capacitated VRP. In: Toth, P., Vigo, D. (eds.) The Vehicle Routing Problem, vol. 9, pp. 109–128. SIAM Monogr. Discrete Math. Appl., Philadelphia (2002)
19. Li, F., Golden, B.L., Wasil, E.A.: A record-to-record travel algorithm for solving the heterogeneous fleet vehicle routing problem. Comput. Oper. Res. **34**, 2734–2742 (2007)
20. Lysgaard, J.: CVRPSEP: a package of separation routines for the capacitated vehicle routing problem. Technical report, Department of Mgt. Science and Logistics, Aarhus School of Business (2003)
21. Lysgaard, J., Letchford, A.N., Eglese, R.W.: A new branch-and-cut algorithm for the capacitated vehicle routing problem. Math. Program. Ser. A **100**, 423–445 (2004)
22. Nag, B., Golden, B.L., Assad, A.A.: Vehicle routing with site dependencies. In: Golden, B., Assad, A. (eds.) Vehicle Routing: Methods and Studies, pp. 149–159. Elsevier, Amsterdam (1988)
23. Niskanen, S., Östergård, P.R.J.: Cliquer user's guide. Technical Report 48, Helsinki University of Technology Communications Laboratory (2003)
24. Pessoa, A., Poggi de Aragão, M., Uchoa, E.: A Robust Branch-Cut-and-Price Algorithm for the Heterogeneous Fleet Vehicle Routing Problem, Lecture Notes in Computer Science, vol. 4525, pp. 150–160. Springer, Berlin (2007)
25. Pisinger, D., Ropke, S.: A general heuristic for vehicle routing problems. Comput. Oper. Res. **34**, 2403–2435 (2007)
26. Taillard, E.D.: A heuristic column generation method for the heterogeneous fleet VRP. RAIRO Rech. Opérationnelle **33**, 1–14 (1999)
27. Toth, P., Vigo , D. (eds.): The Vehicle Routing Problem. Monogr. Discrete Math. Appl. SIAM, Philadelphia (2002)
28. Yaman, H.D.: Formulations and valid inequalities for the heterogeneous vehicle routing problem. Math. Program. Ser. A **106**, 365–390 (2006)