Published online ahead of print February 13, 2012

INFORMS Journal on Computing
Articles in Advance, pp. 1–15
ISSN 1091-9856 (print) | ISSN 1526-5528 (online)

http://dx.doi.org/10.1287/ijoc.1110.0495
© 2012 INFORMS

An Exact Algorithm for the Multi-Trip Vehicle Routing Problem

Aristide Mingozzi
Department of Mathematics, University of Bologna, 40126 Bologna, Italy,
mingozzi@csr.unibo.it

Roberto Roberti, Paolo Toth
Department of Electronics, Computer Science and Systems, University of Bologna, 40126 Bologna, Italy
{roberto.roberti6@unibo.it, paolo.toth@unibo.it}

The *multi-trip vehicle routing problem* (MTVRP) is a variant of the *capacitated vehicle routing problem* where each vehicle can perform a subset of routes, called a vehicle schedule, subject to maximum driving time constraints. Despite its practical importance, the MTVRP has received little attention in the literature. Few heuristics have been proposed, and only an exact algorithm has been presented for a variant of the MTVRP with customer time window constraints and unlimited driving time for each vehicle. We describe two set-partitioning-like formulations of the MTVRP. The first formulation requires the generation of all feasible routes, whereas the second formulation is based on the generation of all feasible schedules. We study valid lower bounds, based on the linear relaxations of both formulations enforced with valid inequalities, that are embedded into an exact solution method. The computational results show that the proposed exact algorithm can solve MTVRP instances taken from the literature, with up to 120 customers.

Key words: vehicle routing; multiple trips; dual ascent heuristics; column-and-cut generation
History: Accepted by Karen Aardal, Area Editor for Design and Analysis of Algorithms; received February 2011; revised August 2011; accepted November 2011. Published online in *Articles in Advance*.

1. Introduction

The *capacitated vehicle routing problem* (CVRP) and its many variations play an important role in the management of many distribution systems. In most of the studied models, the vehicles are identical and each vehicle is allowed to perform, at most, a single route.

In many contexts, a distribution company uses leased vehicles to service the customers and incurs a significant cost for each vehicle used. Whenever the planning period is large with respect to the route duration, and some vehicles can perform several routes in the period, the primary concern of the company is to minimize the number of vehicles used. The *multi-trip vehicle routing problem* (MTVRP) is an extension of the CVRP where each vehicle can perform multiple routes during its working period. The MTVRP can be described as follows.

A complete undirected graph $G = (V', E)$ is given, where V' is the set of vertices and E is the set of edges. We have $V' = \{0\} \cup V$, where vertex 0 represents the *depot* and set $V = \{1, \ldots, n\}$ represents n *customers*, each one requiring q_i units of product from the depot. A fleet $M = \{1, \ldots, m\}$ of m identical vehicles is located at the depot. Each vehicle has capacity Q and maximum driving time T. A *travel cost* α_{ij} and a *travel time* τ_{ij} are associated with each edge $\{i, j\} \in E$. All input data are assumed to be nonnegative.

A *route* of a vehicle is a least-cost elementary cycle in G passing through the depot and a subset of the customers such that the total demand of the customers visited does not exceed the vehicle capacity Q. The cost (duration) of a route is equal to the sum of the travel costs (travel times) of the edges traversed.

A *schedule* of a vehicle is a subset of routes whose total duration is less than or equal to the maximum driving time T. The cost of a schedule is equal to the sum of the costs of its routes.

The MTVRP calls for the design of a set of m schedules of minimum total cost such that each customer is visited exactly once by the routes of the schedules.

1.1. Literature Review

To the best of our knowledge, no exact algorithm for the MTVRP has been presented in the literature so far. The main heuristic methods proposed are surveyed in the following. Fleischmann (1990) addressed the problem first. He proposed a modification of the well-known saving algorithm and used a bin packing heuristic to assign the routes to the vehicles. Taillard et al. (1996) proposed a three-phase algorithm. In the first phase, a set of routes satisfying the capacity constraints are designed. Next, the routes are combined to derive different CVRP solutions. Finally, the routes of each CVRP solution are assigned to the vehicles by solving a bin packing problem. Petch and Salhi (2004) described a multi-phase heuristic that constructs many feasible CVRP solutions and, for each one, assigns routes to vehicles with a bin packing heuristic. Salhi and Petch (2007) investigated a genetic algorithm approach. Olivera and Viera (2007) described an adaptive memory heuristic algorithm. Different tabu search algorithms were proposed by Brandão and Mercer (1997, 1998) and Alonso et al. (2008). A real-life application of the MTVRP is described in Gribkovskaia et al. (2006).

Azi et al. (2010) addressed a variant of the MTVRP where (a) a time window and a revenue are associated with each customer, (b) the duration of each route is limited, (c) the maximum driving time of the vehicles is unlimited, and (d) it is not required to service all customers. Thus, the customers must be chosen based on their associated revenue minus the traveling cost to reach them. Azi et al. (2010) described a branch-and-price algorithm based on a set-packing formulation that can routinely solve instances with 25 customers and a few instances with up to 50 customers. When the time windows are removed, the problem becomes the well-known distance-constrained CVRP. In this case, the algorithm was able to solve two instances with 25 customers.

1.2. Contribution of This Paper

In this paper, we present an exact method to solve the MTVRP based on two set-partitioning-like formulations of the problem, called (F1) and (F2). Formulation (F1) has a binary variable for each route and each vehicle indicating whether a given route is assigned to the schedule of a given vehicle. The second formulation, (F2), has a binary variable for each schedule indicating whether a schedule is performed or not. We study the linear relaxations of both formulations and the relation between the relative lower bounds. We propose four column-and-cut generation (CCG) bounding procedures to solve the linear relaxations of formulations (F1) and (F2), strengthened by valid inequalities derived from the CVRP and by a new class of inequalities called *working time inequalities*. We present a method to improve the lower bounds based on the use of multiple dual solutions achieved by the different bounding procedures. We describe an exact method that uses the lower bounds achieved through the proposed bounding procedures to generate either a reduced set of all routes of (F1) or a reduced set of all schedules of (F2) containing any optimal MTVRP solution. The resulting reduced problems are directly solved by using an integer programming (IP) solver. The computational results show that the proposed method can optimally solve MTVRP instances involving up to 120 customers.

The rest of this paper is organized as follows. In §2, we introduce formulations (F1) and (F2), show the relation between the corresponding linear programming (LP)-relaxations, and describe a relaxation based on (F1). In §3, we describe valid inequalities and a method based on the use of multiple dual solutions to improve the previous relaxations. Section 4 presents three bounding procedures based on (F1). Section 5 describes a bounding procedure based on (F2). In §6, we describe a method for generating feasible schedules. A detailed description of the proposed exact method is given in §7. Finally, computational results are reported in §8.

2. Mathematical Formulations and Relaxations

In this section, we describe two set-partitioning-like formulations, called (F1) and (F2), and a valid integer relaxation, called (RF1), of the MTVRP. Formulation (F1) requires the a priori generation of all feasible routes, whereas formulation (F2) is based on the generation of all feasible schedules. Relaxation (RF1) is an IP problem similar to the set partitioning formulation of the CVRP.

2.1. Formulation (F1)

Let $\mathcal{R}$ be the index set of all feasible routes on graph G, and let $\mathcal{R}_i \subseteq \mathcal{R}$ be the index subset of the routes visiting customer $i \in V$. A cost d_l and a duration τ_l are associated with each route $l \in \mathcal{R}$. In the following, we use R_l and $E(R_l)$ to indicate the set of customers visited and the edges traversed by route $l \in \mathcal{R}$, respectively.

Let ξ_l^j be a binary variable equal to 1 if and only if route $l \in \mathcal{R}$ is assigned to vehicle $j \in M$. The mathematical formulation (F1) is as follows:

$$(\text{F1}) \quad z(\text{F1}) = \min \sum_{l \in \mathcal{R}} d_l \sum_{j \in M} \xi_l^j \tag{1}$$

$$\text{s.t.} \sum_{l \in \mathcal{R}_i} \sum_{j \in M} \xi_l^j = 1 \quad \forall\, i \in V, \tag{2}$$

$$\sum_{l \in \mathcal{R}} \tau_l \xi_l^j \le T \quad \forall\, j \in M, \tag{3}$$

$$\xi_l^j \in \{0, 1\} \quad \forall\, j \in M\ \forall\, l \in \mathcal{R}. \tag{4}$$

Constraints (2) impose that each customer is visited exactly once, and constraints (3) define a feasible schedule for each vehicle used.

We denote by (LF1) the LP-relaxation of (F1) and by $z(\text{LF1})$ its optimal solution cost.

2.2. Formulation (F2)

Let $\mathscr{H}$ be the index set of all schedules. For each schedule $k \in \mathscr{H}$, we denote by $\Omega_k \subseteq \mathscr{R}$ the index subset of the routes in the schedule, by $c_k = \sum_{l\in\Omega_k} d_l$ its cost, and by $\tau(\Omega_k) = \sum_{l\in\Omega_k} \tau_l$ its total duration. Moreover, we define $V(\Omega_k) = \bigcup_{l\in\Omega_k} R_l$ and $E(\Omega_k) = \bigcup_{l\in\Omega_k} E(R_l)$.

We assume that $\mathscr{H}$ contains *undominated* schedules only (i.e., given $k \in \mathscr{H}$, there exists no $k' \in \mathscr{H}\backslash\{k\}$ such that $V(\Omega_k) = V(\Omega_{k'})$ and $c_k > c_{k'}$). Notice that whenever $m = 1$, $\mathscr{H}$ contains only schedules corresponding to optimal integer solutions of the CVRP, where the total duration of the routes is less than or equal to T.

Let y_k be a binary variable equal to 1 if and only if schedule $k \in \mathscr{H}$ is assigned to a vehicle. Formulation (F2) of the MTVRP is as follows:

$$\text{(F2)} \qquad z(\text{F2}) = \min \sum_{k\in\mathscr{H}} c_k y_k \tag{5}$$

$$\text{s.t.} \quad \sum_{k\in\mathscr{H}:\, i\in V(\Omega_k)} y_k = 1 \quad \forall\, i \in V, \tag{6}$$

$$\sum_{k\in\mathscr{H}} y_k \le m, \tag{7}$$

$$y_k \in \{0, 1\} \quad \forall\, k \in \mathscr{H}. \tag{8}$$

Constraints (6) specify that each customer $i \in V$ must be visited exactly once. Constraint (7) imposes the upper bound on the number of vehicles used.

We denote by (LF2) the LP-relaxation of (F2) and by $z(\text{LF2})$ its optimal solution cost.

2.3. Comparing Relaxations (LF1) and (LF2)

The following proposition shows the relation between relaxations (LF1) and (LF2).

Proposition 1. *The following inequality holds*:

$$z(\text{LF1}) \le z(\text{LF2}),$$

and such inequality can be strict.

Proof. We show that any optimal dual solution of (LF1) of cost $z(\text{LF1})$ is a feasible dual solution of (LF2) of cost $z(\text{LF1})$, as well. Let $\mathbf{u} = (u_1, \ldots, u_n)$ and $\mathbf{w} = (w_1, \ldots, w_m)$ be the vectors of the dual variables associated with constraints (2) and (3), respectively. The dual of (LF1) is

$$\text{(DF1)} \qquad z(\text{DF1}) = \max \sum_{i\in V} u_i + T \sum_{j\in M} w_j \tag{9}$$

$$\text{s.t.} \quad \sum_{i\in R_l} u_i + \tau_l w_j \le d_l$$

$$\forall\, j \in M, \forall\, l \in \mathscr{R}, \tag{10}$$

$$u_i \in \mathbb{R} \quad \forall\, i \in V, \tag{11}$$

$$w_j \le 0 \quad \forall\, j \in M. \tag{12}$$

Consider a vehicle schedule Ω_k. Adding inequalities (10) over the routes of Ω_k and all vehicles of M and dividing by m, we obtain

$$\sum_{l\in\Omega_k} \sum_{i\in R_l} u_i + \sum_{l\in\Omega_k} \frac{\tau_l}{m} \sum_{j\in M} w_j \le \sum_{l\in\Omega_k} d_l. \tag{13}$$

Define $\bar{u}_0 = (T/m) \sum_{j\in M} w_j$. Because $\sum_{l\in\Omega_k} \tau_l \le T$ and $w_j \le 0$, we have

$$\bar{u}_0 \le \sum_{l\in\Omega_k} \frac{\tau_l}{m} \sum_{j\in M} w_j \le 0. \tag{14}$$

From inequalities (13) and (14), we derive

$$\sum_{l\in\Omega_k} \sum_{i\in R_l} u_i + \bar{u}_0 \le c_k,$$

which corresponds to the dual constraint associated with variable y_k of problem (LF2), thus showing that $\bar{\mathbf{u}} = (\bar{u}_0, u_1, \ldots, u_n)$ is a dual solution of (LF2), where $u_i, \forall\, i \in V$, are the dual variables of constraints (6) and $\bar{u}_0$ is the dual variable of constraint (7). The cost of the dual solution $\bar{\mathbf{u}}$ of (LF2) defined above is

$$\sum_{i\in V} u_i + m\bar{u}_0 = \sum_{i\in V} u_i + \frac{mT}{m} \sum_{j\in M} w_j = z(\text{LF1}). \quad \square$$

The following example shows that $z(\text{LF1})$ can be strictly smaller than $z(\text{LF2})$.

Example 1. Consider an MTVRP instance with $n = 4$, $m = 2$, $T = 100$, $Q = 2$, $q_1 = 2$, $q_2 = 1$, $q_3 = 1$, and $q_4 = 2$. Travel costs and times coincide; in particular, we have $\alpha_{01} = \tau_{01} = 30$, $\alpha_{02} = \tau_{02} = 15$, $\alpha_{03} = \tau_{03} = 15$, $\alpha_{04} = \tau_{04} = 30$, and $\alpha_{23} = \tau_{23} = 20$.

Formulation (F1) has the following five feasible routes (i.e., $|\mathscr{R}| = 5$) for each vehicle: $R_1 = \{1\}$, $d_1 = 60$; $R_2 = \{2\}$, $d_2 = 30$; $R_3 = \{3\}$, $d_3 = 30$; $R_4 = \{4\}$, $d_4 = 60$; and $R_5 = \{2, 3\}$, $d_5 = 50$. Solving (LF1), we obtain $z(\text{LF1}) = 170$ corresponding to $\xi_1^1 = 1$, $\xi_5^1 = \frac{1}{2}$, $\xi_4^2 = 1$, and $\xi_5^2 = \frac{1}{2}$ (the other ξ_l^j variables having a null value).

Formulation (F2) has the following nine schedules (i.e., $|\mathscr{H}| = 9$): $\Omega_1 = \{1\}$, $c_1 = 60$; $\Omega_2 = \{2\}$, $c_2 = 30$; $\Omega_3 = \{3\}$, $c_3 = 30$; $\Omega_4 = \{4\}$, $c_4 = 60$; $\Omega_5 = \{5\}$, $c_5 = 50$; $\Omega_6 = \{1, 2\}$, $c_6 = 90$; $\Omega_7 = \{1, 3\}$, $c_7 = 90$; $\Omega_8 = \{2, 4\}$, $c_8 = 90$; and $\Omega_9 = \{3, 4\}$, $c_9 = 90$. Solving (LF2), we obtain $z(\text{LF2}) = 180 > z(\text{LF1})$ corresponding to solution $y_6 = 1$, $y_9 = 1$, and $y_k = 0\ \forall\, k \ne 6, 9$.

2.4. Relaxation (RF1)

Problem (RF1) is an integer problem derived from (F1) and is used, in the bounding procedures described in the next sections, to compute three lower bounds to the MTVRP. (RF1) is obtained from (F1) as follows.

1. Replace constraints (3) with the following surrogate constraint:

$$\sum_{l\in\mathscr{R}}\tau_l\sum_{j\in M}\xi_l^j\le mT. \tag{15}$$

2. Set $x_l=\sum_{j\in M}\xi_l^j\ \forall\, l\in\mathscr{R}$. Notice that, because of constraints (2), we have $x_l\in\{0,1\}$.

3. Replace the term $\sum_{j\in M}\xi_l^j$ with x_l into expressions (1), (2), and (15).

The resulting relaxed problem (RF1) involves only the binary variables x_l, $l\in\mathscr{R}$, where x_l is equal to 1 if and only if route l is in the solution:

$$\text{(RF1)}\qquad z(\text{RF1})=\min\ \sum_{l\in\mathscr{R}}d_lx_l \tag{16}$$

$$\text{s.t.}\ \sum_{l\in\mathscr{R}_i}x_l=1\quad\forall\, i\in V, \tag{17}$$

$$\sum_{l\in\mathscr{R}}\tau_lx_l\le mT, \tag{18}$$

$$x_l\in\{0,1\}\quad\forall\, l\in\mathscr{R}. \tag{19}$$

In the following, we refer to problem (RF1) as the CVRP relaxation of the MTVRP (or as the CVRP associated to the MTVRP) and denote by (LRF1) the LP-relaxation of (RF1) and by z(LRF1) its optimal solution cost.

Proposition 2. *The following equality holds*:

$$z(\text{LF1})=z(\text{LRF1}). \tag{20}$$

Proof. Equality (20) follows from the observation that any (LF1) solution $\boldsymbol{\xi}$ can be transformed into an (LRF1) solution $\mathbf{x}$ of the same cost, and vice versa.

The following algorithm transforms a solution $\mathbf{x}$ of (LRF1) into a feasible (LF1) solution $\boldsymbol{\xi}$ of cost z(LRF1).

1. Initialize $j=0$.
2. Set $j=j+1$ and $wt=0$.
3. Let $l^*=\min\{l\in\mathscr{R}\colon x_{l^*}>0\}$. If $\tau_{l^*}x_{l^*}\le T-wt$, set $wt=wt+\tau_{l^*}x_{l^*}$, $\xi_{l^*}^j=x_{l^*}$, $x_{l^*}=0$, and repeat step 3. If $\tau_{l^*}x_{l^*}>T-wt$, define $p=((T-wt)/(\tau_{l^*}x_{l^*}))$, and set $\xi_{l^*}^j=p$, $x_{l^*}=1-p$, and return to step 2.

Any (LF1) solution $\boldsymbol{\xi}$ can be transformed into a feasible solution $\mathbf{x}$ of (LRF1) of cost z(LF1) by simply setting $x_l=\sum_{j\in M}\xi_l^j$. □

3. Improving Relaxations (LRF1) and (LF2) and Outline of the Exact Method

In this section, we describe two relaxations, called $(\overline{\text{LRF1}})$ and $(\overline{\text{LF2}})$, derived respectively from (LRF1) and (LF2) by adding some families of valid inequalities. These relaxations are used to derive valid lower bounds on the MTVRP that are embedded in the exact method proposed in §7.

3.1. Relaxation $(\overline{\text{LRF1}})$.

Relaxation $(\overline{\text{LRF1}})$ derives from (LRF1) by adding two types of valid inequalities for the set-partitioning formulation of the CVRP (see Baldacci et al. 2011), called *strengthened capacity inequalities* and *subset row inequalities*, and replacing inequality (18) with a new type of inequality, called *working time inequalities*.

Strengthened Capacity Inequalities. Let $\mathscr{S}$ be the set of all the subsets of customers of cardinality greater than or equal to 2 (i.e., $\mathscr{S}=\{S\subseteq V\colon |S|\ge 2\}$), and let $\delta(S)$ be the cutset of $S\in\mathscr{S}$. Let $\rho_l(S)$ be the number of edges of $\delta(S)$ traversed by route $l\in\mathscr{R}$ (i.e., $\rho_l(S)=|E(R_l)\cap\delta(S)|$). The strengthened capacity (SC) inequalities are

$$\sum_{l\in\mathscr{R}}\rho_l(S)x_l\ge 2k(S)\quad\forall\, S\in\mathscr{S},$$

where $k(S)$ is a lower bound on the number of vehicles required to service the customer set S and can be computed as $\lceil\sum_{i\in S}q_i/Q\rceil$.

Subset Row Inequalities. Subset row (SR) inequalities were introduced by Jepsen et al. (2008) for the *vehicle routing problem with time windows*. They combine clique inequalities with odd-hole inequalities. Let $\mathscr{C}=\{C\subseteq V\colon |C|\ge 3\}$, and let η be an integer such that $\eta>1$. Any feasible (RF1) solution must satisfy the following SR inequality:

$$\sum_{l\in\mathscr{R}}\varphi_l(C)x_l\le\varphi_0(C)\quad\forall\, C\in\mathscr{C},$$

where

$$\varphi_l(C)=\left\lfloor\frac{|R_l\cap C|}{\eta}\right\rfloor,\quad\text{and}\quad\varphi_0(C)=\left\lfloor\frac{|C|}{\eta}\right\rfloor.$$

We consider the following three special cases of the SR inequalities where $\eta=2$.

1. **SR3 inequalities**. SR3 inequalities correspond to the subset $\mathscr{C}^3\subseteq\mathscr{C}$ defined as $\mathscr{C}^3=\{C\subseteq V\colon |C|=3\}$. SR3 inequalities correspond to a subset of clique inequalities and impose that, at most, one of the routes visiting at least two of the customers in $C\in\mathscr{C}^3$ can be in the solution.
2. **SR5 inequalities**. SR5 inequalities correspond to the subset $\mathscr{C}^5\subseteq\mathscr{C}$ defined as $\mathscr{C}^5=\{C\subseteq V\colon |C|=5\}$. SR5 inequalities correspond to odd-hole inequalities and impose that, at most, two of the routes visiting at least two of the customers in $C\in\mathscr{C}^5$ can be in the solution.
3. **WSR3 inequalities**. Weak subset row (WSR3) inequalities, introduced by Baldacci et al. (2011), are downlifted SR3 inequalities. They are used as an alternative to the SR3 inequalities in the bounding procedures described in §4. Let $E(C)$, $\forall\, C\in\mathscr{C}^3$, be the

edges whose terminal vertices are both in C. The coefficients of the WSR3 inequalities, $\varphi_l(C)$, are computed as $\varphi_l(C)=1$ if $|E(R_l)\cap E(C)|\ge 1$, and $\varphi_l(C)=0$ if $|E(R_l)\cap E(C)|=0$.

Hereafter, we use $\mathscr{C}=\mathscr{C}^3\cup\mathscr{C}^5$ and $C\in\mathscr{C}$ to refer to both the index and the customer subset of an SR inequality. The SR3, SR5, and WSR3 inequalities are separated through complete enumeration.

Working Time Inequalities. These new inequalities strengthen inequality (18). They are given by the following proposition.

Proposition 3. *Let $t_{\min}$ be a lower bound on the duration of any route $l\in\mathscr{R}$. If travel costs α_{ij} satisfy the triangle inequality, we can define $t_{\min}=\min_{i\in V}\{2\tau_{0i}\}$. For a given time $t=t_{\min},\dots,T$, let $\hat{t}=T$ mod $(t)+1$, and let $\mathscr{L}=\{L\subseteq\mathscr{R}\colon \hat{t}\le\tau_l<t, \forall l\in L$, and $R_{l'}\cap R_{l''}\ne\varnothing, \forall l',l''\in L\}$. The following inequalities, called* working time (*WT*) inequalities, *are valid for* (LRF1):

$$\sum_{l\in\mathscr{R}\setminus L}\left\lfloor\frac{\tau_l}{t}\right\rfloor x_l+\sum_{l\in L}x_l\le\left\lfloor\frac{T}{t}\right\rfloor m \qquad \forall t=t_{\min},\dots,T,\ \forall L\in\mathscr{L}. \tag{21}$$

Proof. For a given $L\in\mathscr{L}$, inequalities (3) can be written as

$$\sum_{l\in\mathscr{R}\setminus L}\tau_l\xi_l^j\le T-\sum_{l\in L}\tau_l\xi_l^j \quad \forall j\in M. \tag{22}$$

Inequalities (22) can be relaxed by replacing τ_l with $\hat{t},\forall l\in L$:

$$\sum_{l\in\mathscr{R}\setminus L}\tau_l\xi_l^j\le T-\hat{t}\sum_{l\in L}\xi_l^j \quad \forall j\in M. \tag{23}$$

By dividing inequalities (23) by t and rounding down the coefficients, we have

$$\sum_{l\in\mathscr{R}\setminus L}\left\lfloor\frac{\tau_l}{t}\right\rfloor\xi_l^j\le\left\lfloor\frac{T-\hat{t}\sum_{l\in L}\xi_l^j}{t}\right\rfloor \quad \forall j\in M. \tag{24}$$

Notice that because of the definition of L, $\sum_{l\in L}\xi_l^j$ is equal to either 0 or 1 for any feasible MTVRP solution. If $\sum_{l\in L}\xi_l^j=0$, then $\lfloor(T-\hat{t}\sum_{l\in L}\xi_l^j)/t\rfloor=\lfloor T/t\rfloor$. If $\sum_{l\in L}\xi_l^j=1$, because of the definition of $\hat{t}$, we have $\lfloor(T-\hat{t}\sum_{l\in L}\xi_l^j)/t\rfloor=\lfloor(T-\hat{t})/t\rfloor=\lfloor T/t\rfloor-1$. Thus, we have

$$\left\lfloor\frac{T-\hat{t}\sum_{l\in L}\xi_l^j}{t}\right\rfloor=\left\lfloor\frac{T}{t}\right\rfloor-\sum_{l\in L}\xi_l^j \quad \forall j\in M. \tag{25}$$

From inequalities (24) and Equation (25), we derive

$$\sum_{l\in\mathscr{R}\setminus L}\left\lfloor\frac{\tau_l}{t}\right\rfloor\xi_l^j+\sum_{l\in L}\xi_l^j\le\left\lfloor\frac{T}{t}\right\rfloor \quad \forall j\in M. \tag{26}$$

By summing up inequalities (26) and replacing $\sum_{j\in M}\xi_l^j$ with $x_l\in\{0,1\}$, we derive inequalities (21). □

The separation of inequalities (21) is $\mathcal{NP}$-hard because it requires the computation of the set $\mathscr{L}$. Therefore, we use a subset of the WT inequalities (21) that are obtained by defining $L_{it}=\{l\in\mathscr{R}_i\colon \hat{t}\le\tau_l<t\}$, $\forall i\in V$, and by setting $\mathscr{L}=\{L_{it}\colon i\in V, t_{\min}\le t\le T\}$. Let $\mathscr{W}=\{(t,i)\colon t_{\min}\le t\le T, i\in V\}$. These latter inequalities can be conveniently written as follows:

$$\sum_{l\in\mathscr{R}}\vartheta_l(t,i)x_l\le\vartheta_0(t) \quad \forall(t,i)\in\mathscr{W}, \tag{27}$$

where, for a given $(t,i)\in\mathscr{W}$, $\vartheta_l(t,i)=\lfloor\tau_l/t\rfloor$ $\forall l\in\mathscr{R}\setminus L_{it}$, $\vartheta_l(t,i)=1$ $\forall l\in L_{it}$, and $\vartheta_0(t)=\lfloor T/t\rfloor m$. Inequalities (27) can be separated by complete enumeration.

Example 2. Consider the MTVRP instance defined in Example 1 of §2.3. Problem (RF1) involves the same set $\mathscr{R}$ of routes of formulation (F1). An optimal solution of (RF1) of cost $z(\text{RF1})=z(\text{LRF1})=170$ is given by $x_1=1, x_4=1$, and $x_5=1$. Consider the WT inequality (27) for $t=60$ and $i=3$. As $\hat{t}=41$, we have $L_{60,3}=\{5\}$. Thus, the coefficients of (27) are $\vartheta_1(60,30)=1$, $\vartheta_2(60,30)=0$, $\vartheta_3(60,30)=0$, $\vartheta_4(60,30)=1$, $\vartheta_5(60,30)=1$, and $\vartheta_0(60)=2$. Inequality (27) corresponds to $x_1+x_4+x_5\le 2$, which is violated by the aforementioned solution of (RF1).

Relaxation $(\overline{\text{LRF1}})$ is the following problem:

$$(\overline{\text{LRF1}}) \qquad z(\overline{\text{LRF1}})=\min\sum_{l\in\mathscr{R}}d_lx_l \tag{28}$$

$$\text{s.t.}\ \sum_{l\in\mathscr{R}_i}x_l=1 \quad \forall i\in V, \tag{29}$$

$$\sum_{l\in\mathscr{R}}\rho_l(S)x_l\ge 2k(S) \quad \forall S\in\mathscr{S}, \tag{30}$$

$$\sum_{l\in\mathscr{R}}\varphi_l(C)x_l\le\varphi_0(C) \quad \forall C\in\mathscr{C}, \tag{31}$$

$$\sum_{l\in\mathscr{R}}\vartheta_l(t,i)x_l\le\vartheta_0(t) \quad \forall(t,i)\in\mathscr{W}, \tag{32}$$

$$x_l\ge 0 \quad \forall l\in\mathscr{R}. \tag{33}$$

Let $(\mathbf{u},\mathbf{v},\mathbf{g},\mathbf{h})$ be the dual variables of $(\overline{\text{LRF1}})$, where $\mathbf{u}\in\mathbb{R}^n$ is associated with (29), $\mathbf{v}\in\mathbb{R}_+^{|\mathscr{S}|}$ with (30), $\mathbf{g}\in\mathbb{R}_-^{|\mathscr{C}|}$ with (31), and $\mathbf{h}\in\mathbb{R}_-^{|\mathscr{W}|}$ with (32).

In §4, we describe three bounding procedures to solve $(\overline{\text{LRF1}})$, called H^1, H^2, and H^3, that are extensions of similar procedures proposed by Baldacci et al. (2011) for the CVRP. The three procedures are executed in sequence, and the dual solution of procedure H^k is used to hot-start procedure H^{k+1}, $k=1,2$.

H^1 and H^2 are dual ascent heuristics based on a CCG method that differs from standard CCG methods based on the simplex for the use of a dual ascent heuristic to find a near-optimal dual solution of problem ($\overline{\text{LRF1}}$). H^1 finds a dual solution $(\mathbf{u}^1, \mathbf{v}^1, \mathbf{g}^1)$ of cost LB_1 of problem ($\overline{\text{LRF1}}$) by adding SC and WSR3 inequalities, ignoring WT inequalities, and replacing the route set $\mathcal{R}$ with the set of all q-routes (see, e.g., Christofides et al. 1981). H^2 differs from H^1 as it uses elementary routes instead of q-routes. H^2 achieves a dual solution $(\mathbf{u}^2, \mathbf{v}^2, \mathbf{g}^2)$ of cost LB_2, such that $\text{LB}_1 \le \text{LB}_2$. H^3 is a CCG method based on the simplex to solve ($\overline{\text{LRF1}}$). H^3 inherits from H^2 the master problem and achieves an optimal dual solution $(\mathbf{u}^3, \mathbf{v}^3, \mathbf{g}^3, \mathbf{h}^3)$ of cost LB_3.

3.2. Relaxation ($\overline{\text{LF2}}$)

Relaxation ($\overline{\text{LF2}}$) is obtained by adding SC, SR3, and SR5 inequalities to (LF2):

$$(\overline{\text{LF2}}) \qquad z(\overline{\text{LF2}}) = \min \sum_{k\in\mathcal{H}} c_k y_k \tag{34}$$

$$\text{s.t.} \quad \sum_{k\in\mathcal{H}:\ i\in V(\Omega_k)} y_k = 1 \quad \forall\, i \in V, \tag{35}$$

$$\sum_{k\in\mathcal{H}} y_k \le m, \tag{36}$$

$$\sum_{k\in\mathcal{H}} \rho_k(S) y_k \ge 2k(S) \quad \forall\, S \in \mathcal{S}, \tag{37}$$

$$\sum_{k\in\mathcal{H}} \varphi_k(C) y_k \le \varphi_0(C) \quad \forall\, C \in \mathcal{C}, \tag{38}$$

$$y_k \ge 0 \quad \forall\, k \in \mathcal{H}, \tag{39}$$

where $\rho_k(S)$ and $\varphi_k(C)$ of constraints (37) and (38) are computed as $\rho_k(S) = |E(\Omega_k) \cap \delta(S)|$ and $\varphi_k(C) = \lfloor |V(\Omega_k) \cap C|/\eta \rfloor$.

Let $(\mathbf{w}, \mathbf{v}, \mathbf{g})$ be the dual variables of ($\overline{\text{LF2}}$), where $\mathbf{w} = (w_0, w_1, \ldots, w_n)$ and $w_0 \in \mathbb{R}_-$ is associated with (36), $(w_1, \ldots, w_n) \in \mathbb{R}^n$ with (35), $\mathbf{v} \in \mathbb{R}_+^{|\mathcal{S}|}$ with (37), and $\mathbf{g} \in \mathbb{R}_-^{|\mathcal{C}|}$ with (38).

In §5, we describe a CCG procedure, called H^4, to solve ($\overline{\text{LF2}}$). We denote by $\mathbf{x}^4$ and $(\mathbf{w}^4, \mathbf{v}^4, \mathbf{g}^4)$ the optimal primal and dual solutions, respectively, of ($\overline{\text{LF2}}$) of cost LB_4 achieved by H^4.

3.3. Using Multiple Dual Solutions to Improve Procedures H^2, H^3, and H^4

In this section, we describe two simple methods to improve the lower bounds and the computing times of procedures H^2, H^3, and H^4.

The method used in H^2 and H^3 is based on the following simple observation, which applies to any problem with binary variables. Consider the following problem with n variables and m constraints:

$$\begin{aligned} \text{F} \quad z\text{F} = \min \quad & \mathbf{cx} \\ \text{s.t.} \quad & \mathbf{Ax} = \mathbf{b}, \\ & \mathbf{x} \in \{0, 1\}^n. \end{aligned}$$

Let us denote by (LF) the LP-relaxation of F and by (D) the dual of (LF), and let zF, z(LF), and z(D) be the optimal solution costs of F, (LF), and (D), respectively. Moreover, we assume to know a valid upper bound $z(ub)$ on z(F).

Observation 1. Let $\mathbf{w}'$ be a feasible (nonnecessarily optimal) (D) solution of cost z'(D). Because any optimal (F) solution $\mathbf{x}^*$ satisfies $z(\text{F}) = z'(\text{D}) + \sum_{j\in J} c'_j$, where $c'_j = c_j - \mathbf{w}'\mathbf{a}_j$, $J = \{j\colon x^*_j = 1, 1 \le j \le n\}$ is the reduced cost of variable x_j, and a_j is the jth column vector of matrix $\mathbf{A}$, then any variable x_j such that $z'(\text{D}) + c'_j > z(ub)$ cannot be in any optimal solution of cost less than or equal to $z(ub)$ and can be removed from (F). The resulting problem $J(\text{F}')$ has the same optimal solutions of (F). Let (LF)$'$ be the LP-relaxation of (F$'$), and let $z(\text{LF}')$ be its optimal solution cost. It follows that $z(\text{LF}') \ge z\,(\text{LF})$, and such inequality can be strict if $c'_j > z(ub) - z\,(\text{LF})$ for some variable x_j of the optimal basis of (LF). In practice, it might happen that $z\,(\text{LF}) < z(\text{LF}') = z(\text{F})$, as shown in the following example.

Example 3. Consider the following set partitioning problem (F):

$$\begin{array}{llllllll} \min = & x_1 & +x_2 & +x_3 & +4x_4 & +3x_5 & +4.5x_6 & \\ \text{s.t.} & x_1 & & & +x_4 & +x_5 & +x_6 & =1, \\ & & x_2 & & +x_4 & +x_5 & +x_6 & =1, \\ & & & x_3 & +x_4 & & +x_6 & =1, \\ & x_1 & +x_2 & & & +x_5 & & =1, \\ & x_1 & & +x_3 & & & +x_6 & =1, \\ & & x_2 & +x_3 & & & & =1, \\ & \multicolumn{7}{l}{x_j \in \{0,1\}, \quad j = 1, \ldots, 6.} \end{array}$$

A valid upper bound is $z(ub) = 4.5$, whereas the optimal integer solution cost of (F) is $z(\text{F}) = 4$ corresponding to $x_3 = 1, x_5 = 1, x_j = 0, j \ne 3, 5$. The optimal (LF) solution cost is $z\,(\text{LF}) = 2$ corresponding to $\mathbf{x} = (\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, 0, 0)$. Consider the dual solution $\mathbf{w} = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3}, \frac{1}{3}, \frac{1}{3}, \frac{1}{3})$ of cost $z(\text{D}) = 2$; the vector $\mathbf{c}'$ of the reduced costs with respect to $\mathbf{w}$ is $\mathbf{c}' = (0, 0, 0, 3, 2, \frac{19}{6})$.

According to Observation 1, we can remove variables x_4 and x_6 from (F) as $z(\text{D}) + c'_4 > z(ub)$ and $z(\text{D}) + c'_6 > z(ub)$. Thus, we derive problem (F$'$) that involves variables x_1, x_2, x_3 and x_5 only.

An optimal solution of (LF') is $\mathbf{x} = (0, 0, 1, 0, 1, 0)$ of cost $z(\mathrm{LF}') = 4$, which is also an optimal integer solution of (F); that is $z\,\mathrm{LF} < z(\mathrm{LF}') = z(\mathrm{F})$.

Observation 1 is used by procedures H^2 and H^3 for solving the pricing problem. Procedure H^k, $k = 2, 3$, uses the dual solution $(\mathbf{u}^{k-1}, \mathbf{v}^{k-1}, \mathbf{g}^{k-1})$ produced by H^{k-1} to reject any route of negative reduced cost with respect to the incumbent dual solution of problem $(\overline{\mathrm{LRF1}})$ but having reduced cost with respect to $(\mathbf{u}^{k-1}, \mathbf{v}^{k-1}, \mathbf{g}^{k-1})$ greater than $z(ub) - \mathrm{LB}_{k-1}$, where $z(ub)$ is a valid upper bound to the MTVRP that can be computed by running any of the heuristic algorithms available from the literature (hereafter, we assume to know such an upper bound on the MTVRP).

In practice, this method improves both the final lower bound LB_k achieved by H^k and the computing time of H^k, avoiding the generation of routes of negative reduced cost that cannot be in any optimal integer solution.

To improve the lower bound value as well as the computing time of H^4, we use the following observation.

Observation 2. Let d_l^3 be the reduced cost of route $l \in \mathcal{R}$ with respect to the dual solution $(\mathbf{u}^3, \mathbf{v}^3, \mathbf{g}^3, \mathbf{h}^3)$ of $(\overline{\mathrm{LRF1}})$ of cost LB_3 achieved by H^3. Let $z(\mathrm{F2})$ be the cost of an optimal (F2) solution $\mathbf{y}^*$, and let $Y = \{k \in \mathcal{H}\colon y_k^* = 1\}$. The following inequality holds: $z(\mathrm{F2}) \geq \mathrm{LB}_3 + \sum_{k \in Y} d^3(\Omega_k)$, where $d^3(\Omega_k) = \sum_{l \in \Omega_k} d_l^3$. Thus, any schedule $k \in \mathcal{H}$ such that $d^3(\Omega_k) > z(ub) - \mathrm{LB}_3$ cannot be in any optimal solution and can be removed from (F2).

This observation is used in procedure H^4 to reject any schedule Ω_k of negative reduced cost with respect to the dual solution of the master such that $d^3(\Omega_k) > z(ub) - \mathrm{LB}_3$.

4. Bounding Procedures H^1, H^2, and H^3 Based on Relaxation $(\overline{\mathrm{LRF1}})$

In this section, we describe three bounding procedures H^1, H^2, and H^3 to derive lower bounds LB_1, LB_2, and LB_3, respectively. H^1 and H^2 use a method called CCG, proposed by Baldacci et al. (2008) for the CVRP. H^3 is a CCG method, based on the simplex, that uses a new strategy to solve the pricing problem.

4.1. Algorithm CCG

This algorithm differs from standard CCG methods based on the simplex as it uses a dual ascent heuristic to find a near-optimal dual solution of the master problem. The main advantages of this method are

(i) it is faster than simplex based methods,

(ii) it is not affected by the typical degeneration of the simplex, and

(iii) the dual solution produced does not correspond to a dual basic solution of the master, thus making the pricing problem easier to solve.

4.1.1. The Dual Heuristic to Solve the Master Problem. The master problem of $(\overline{\mathrm{LRF1}})$ with SC and WSR3 inequalities is defined by a subset $\bar{\mathcal{R}} \subseteq \mathcal{R}$ and the sets $\mathcal{S}$ and $\mathcal{C}$ of SC and WSR3 inequalities. The initializations of $\bar{\mathcal{R}}$, $\mathcal{S}$, and $\mathcal{C}$ are described in §§4.2 and 4.4. The dual ascent heuristic to compute a feasible dual solution $(\bar{\mathbf{u}}, \bar{\mathbf{v}}, \bar{\mathbf{g}})$ of the master problem corresponds to the method proposed by Baldacci et al. (2011) and is based on the following theorem.

Theorem 1. *Let us associate penalties $\lambda_i \in \mathbb{R}$ $\forall\, i \in V$ with constraints* (29), *$\mu_S \in \mathbb{R}_+$ $\forall\, S \in \mathcal{S}$ with SC constraints* (30), *and $\omega_C \in \mathbb{R}_-$ $\forall\, C \in \mathcal{C}$ with constraints* (31) *in the form of WSR3 inequalities. For each route R_l ($l \in \mathcal{R}$), we indicate with $\bar{c}_l$ its reduced cost with respect to $(\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\omega})$ (i.e., $\bar{c}_l = c_l - \sum_{i \in R_l} \lambda_i - \sum_{S \in \mathcal{S}} \rho_l(S)\mu_S - \sum_{C \in \mathcal{C}} \varphi_l(C)\omega_C$). For each $i \in V$, let us compute*

$$\beta_i = q_i \min_{l \in \bar{\mathcal{R}}\,:\, i \in R_l} \left\{ \frac{\bar{c}_l}{\sum_{i \in R_l} q_i} \right\} \quad \forall\, i \in V.$$

A feasible dual solution $(\mathbf{u}, \mathbf{v}, \mathbf{g})$ of the master of cost $z((\overline{\mathrm{LRF1}})(\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\omega}))$ is obtained by setting

$$u_i = \beta_i + \lambda_i \quad \forall\, i \in V, \qquad v_S = \mu_S \quad \forall\, S \in \mathcal{S}, \quad \textit{and}$$

$$g_C = \omega_C \quad \forall\, C \in \mathcal{C}.$$

Proof. See Baldacci et al. (2011). □

A near-optimal dual solution $(\bar{\mathbf{u}}, \bar{\mathbf{v}}, \bar{\mathbf{g}})$ of the current master problem is obtained by an iterative procedure that performs a predefined number *Maxit*2 of subgradient iterations to solve the problem

$$\mathrm{LCCG} = \max_{(\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\omega})} \left\{ z\big((\overline{\mathrm{LRF1}})(\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\omega})\big) \right\}.$$

We denote by $(\bar{\mathbf{u}}, \bar{\mathbf{v}}, \bar{\mathbf{g}})$ the near-optimal dual solution of $(\overline{\mathrm{LRF1}})$ achieved after *Maxit*2 iterations and by $\bar{\mathbf{x}}$ the corresponding nonnecessarily feasible solution of $(\overline{\mathrm{LRF1}})$.

4.1.2. Adding Violated WSR3 Inequalities and Solving the Pricing Problem. After computing the master dual solution $(\bar{\mathbf{u}}, \bar{\mathbf{v}}, \bar{\mathbf{g}})$, CCG adds, to the master, the largest subset $\mathcal{C}'$ of at most $\Delta(\mathcal{C})$ WSR3 inequalities most violated by $\bar{\mathbf{x}}$, where $\Delta(\mathcal{C})$ is a given parameter. In our computational results, we set $\Delta(\mathcal{C}) = 100$. The set $\mathcal{S}$ of SC inequalities is generated at the beginning of CCG, as described in Baldacci et al. (2008), and is not changed.

Moreover, CCG generates a subset $\mathcal{N} \subseteq \mathcal{R}$ of routes of negative reduced cost with respect to $(\bar{\mathbf{u}}, \bar{\mathbf{v}}, \bar{\mathbf{g}})$. If $\mathcal{N} \neq \varnothing$, then CCG sets $\bar{\mathcal{R}} = \bar{\mathcal{R}} \cup \mathcal{N}$; otherwise, $(\bar{\mathbf{u}}, \bar{\mathbf{v}}, \bar{\mathbf{g}})$ is a feasible dual solution.

CCG terminates after *Maxit*1 macro iterations (*Maxit*1 defined a priori) and provides an $(\overline{\mathrm{LRF1}})$ dual solution, $(\mathbf{u}^*, \mathbf{v}^*, \mathbf{g}^*)$, of cost LCCG corresponding to the penalty vector $(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*, \boldsymbol{\omega}^*)$.

4.2. Bounding Procedure H^1

Procedure H^1 enlarges the route set $\mathcal{R}$ with an extension of the q-routes introduced by Christofides et al. (1981). The initial route set $\bar{\mathcal{R}}$ of the master problem contains all single customer routes $(0, i, 0)$, $i \in V$. We initialize $\mathcal{C} = \varnothing$ and $(\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\omega}) = (\mathbf{0}, \mathbf{0}, \mathbf{0})$.

At a given macro iteration, the set $\mathcal{N}$ of routes of negative reduced cost with respect to the dual solution $(\bar{\mathbf{u}}, \bar{\mathbf{v}}, \bar{\mathbf{g}})$ is computed as follows. Define the modified arc cost

$$\bar{\alpha}_{ij} = \alpha_{ij} - \tfrac{1}{2}(\bar{u}_i + \bar{u}_j) - \sum_{S \in \mathcal{S}:\{i,j\} \in \delta(S)} \bar{v}_S, \quad \forall \{i, j\} \in E,$$

with respect to $(\bar{\mathbf{u}}, \bar{\mathbf{v}}, \bar{\mathbf{g}})$.

A (q, j, i)-path is a nonnecessarily elementary path that starts from the depot and visits a subset of customers (once or more) such that their total demand is q and the last two customers visited are j and i in this order. A (q, j, i)-path is defined for any q, $(q_j + q_i \le q \le Q)$, and for any pair of vertices $j, i \in V'$, $j \ne i$. A $(q, j, 0)$-route (or simply a q-route) is a $(q, j, 0)$-path. Let $f(q, j, i)$ be the cost of a least-cost (q, j, i)-path using the modified arc costs $\bar{\alpha}_{ij}$, and let $\mathcal{C}_{ij} = \{C \in \mathcal{C}\colon \{i, j\} \in E(C)\}$. Functions $f(q, j, i)$ can be computed with the following dynamic programming (DP) recursion:

$$f(q, j, i) = \min_{k \in V' \setminus \{i\}} \Big\{ f(q - q_i, k, j) + \bar{\alpha}_{ji} - \sum_{C \in \mathcal{C}_{ji} \setminus \mathcal{C}_{kj}} \bar{g}_C \Big\}$$
$$\forall q = q_j + q_i, \ldots, Q, \ \forall j, i \in V', \ j \ne i, \ j \ne 0.$$

Functions $f(q, j, i)$ are initialized as $f(q_i, 0, i) = \bar{\alpha}_{0i}$ $\forall i \in V$.

The set $\mathcal{N}$ contains any q-route of negative reduced cost corresponding to the least-cost q-route that visits $j \in V$ as the last customer before arriving at the depot.

At the end, H^1 sets $(\mathbf{u}^1, \mathbf{v}^1, \mathbf{g}^1) = (\mathbf{u}^*, \mathbf{v}^*, \mathbf{g}^*)$, $(\boldsymbol{\lambda}^1, \boldsymbol{\mu}^1, \boldsymbol{\omega}^1) = (\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*, \boldsymbol{\omega}^*)$, and $\mathrm{LB}_1 = \mathrm{LCCG}$.

4.3. Route Generation Algorithm GENR

In this section, we describe the two-phase algorithm GENR used by bounding procedure H^2 to initialize the master problem and to solve the pricing problem. Moreover, the second phase of GENR is used by H^3 to solve the pricing problem and by the exact method to generate all routes of reduced cost, with respect to $(\mathbf{u}^3, \mathbf{v}^3, \mathbf{g}^3, \mathbf{h}^3)$, less than $z(ub) - \mathrm{LB}_3$. GENR generates elementary routes and extends the method proposed by Baldacci et al. (2008) for the CVRP by introducing new bounding functions to reduce the state space graph. GENR is based on the following observations.

Let $V(P) \subseteq V$ be the subset of customers visited by path P, and let $E(P)$ be the edges traversed by path P. Let $\mathcal{P}$ be the set of all elementary paths of minimum cost from the depot such that $q(P) \le Q/2 + q_{\sigma(P)}$, $\forall P \in \mathcal{P}$, where $q(P) = \sum_{i \in V(P)} q_i$, $\tau(P) = \sum_{\{i,j\} \in E(P)} \tau_{ij}$, and $\sigma(P)$ represent the load, the duration, and the terminal customer of path P, respectively. We denote by $\pi(P)$ the immediate predecessor of $\sigma(P)$ in P.

Every route passing through customer i can be obtained by combining a pair of paths $P, \bar{P} \in \mathcal{P}$ such that

$$\sigma(P) = \sigma(\bar{P}) = i, \quad V(P) \cap V(\bar{P}) = \{i\},$$
$$\tau(P) + \tau(\bar{P}) \le T, \quad \text{and} \quad q(P) + q(\bar{P}) \le Q + q_i. \tag{40}$$

Given a nonnecessarily feasible dual solution $(\tilde{\mathbf{u}}, \tilde{\mathbf{v}}, \tilde{\mathbf{g}})$ of $(\overline{\mathrm{LRF1}})$, a feasible dual solution $(\mathbf{u}', \mathbf{v}', \mathbf{g}')$ of $(\overline{\mathrm{LRF1}})$, and four parameters $\Delta(\mathcal{P})$, $\Delta(\mathcal{R})$, γ, and γ', GENR generates the largest subset $\mathcal{B}$ of routes such that $|\mathcal{B}| \le \Delta(\mathcal{R})$, $\tilde{d}_l \le \gamma$, and $d'_l \le \gamma'$, $\forall l \in \mathcal{B}$, where $\tilde{d}_l$ and d'_l are the reduced costs of route l with respect to $(\tilde{\mathbf{u}}, \tilde{\mathbf{v}}, \tilde{\mathbf{g}})$ and $(\mathbf{u}', \mathbf{v}', \mathbf{g}')$, respectively. In the first phase, GENR generates the path set $\mathcal{P}$ such that $|\mathcal{P}| \le \Delta(\mathcal{P})$, where $\Delta(\mathcal{P})$ is a parameter that limits the cardinality of the set $\mathcal{P}$ and is imposed for memory limits. In the second phase, GENR combines the paths of $\mathcal{P}$ to provide the subset $\mathcal{B} \subseteq \mathcal{R}$. Parameters $\Delta(\mathcal{R})$, γ, and γ' are defined according to the type of subset $\mathcal{B}$ that must be generated.

Define the modified path cost $\tilde{d}(P)$ with respect to $(\tilde{\mathbf{u}}, \tilde{\mathbf{v}})$ as follows:

$$\tilde{d}(P) = \sum_{\{i,j\} \in E(P)} \tilde{\alpha}_{ij}, \tag{41}$$

where $\tilde{\alpha}_{ij} = \alpha_{ij} - \frac{1}{2}(\tilde{u}_i + \tilde{u}_j) - \sum_{S \in \mathcal{S}:\{i,j\} \in \delta(S)} \tilde{v}_S$. Let $lb(P)$ be a lower bound on the reduced cost with respect to $(\tilde{\mathbf{u}}, \tilde{\mathbf{v}}, \tilde{\mathbf{g}})$ of any route containing path P. For a given path $P \in \mathcal{P}$, let $\sigma(P) = i$. The lower bound $lb(P)$ is computed as

$$lb(P) = \tilde{d}(P) - \sum_{C \in \mathcal{C}} \left\lceil \frac{|E(P) \cap E(C)|}{2} \right\rceil \tilde{g}_C$$
$$+ \min_{j \in V \setminus V(P),\, q_i \le q' \le Q - q(P) + q_i} \{ f(q', j, i) + \tilde{g}_{\pi(P)ij} \},$$

where $\tilde{g}_{\pi(P)ij}$ is the dual of the WSR3 inequality $\{\pi(P), i, j\} \in \mathcal{C}$ (we assume $\tilde{g}_{\pi(P)ij} = 0$ if $\{\pi(P), i, j\} \notin \mathcal{C}$).

The two phases of GENR are described in brief below. For a more detailed description of GENR, see Baldacci et al. (2008).

4.3.1. Phase 1 of GENR.

Phase 1 generates a sequence $\mathcal{P}$ of paths $(P^1, \ldots, P^k)$, with $k \le \Delta(\mathcal{P})$, such that $lb(P^1) \le \cdots \le lb(P^k) \le \gamma$. If Phase 1 terminates with $|\mathcal{P}| = \Delta(\mathcal{P})$, the entire algorithm terminates prematurely as there is no guarantee that all routes of reduced cost less than or equal to γ are generated by combining the pairs $P, \bar{P} \in \mathcal{P}$.

4.3.2. Phase 2 of GENR. Phase 2 is executed only if $|\mathscr{P}| < \Delta(\mathscr{P})$. Consider a pair of paths $(P, \bar{P}) \in \mathscr{P}$ satisfying conditions (40). Let $l \in \mathscr{R}$ be the index of the route resulting from $(P, \bar{P})$ of reduced cost $\tilde{d}_l$ with respect to $(\tilde{\mathbf{u}}, \tilde{\mathbf{v}}, \tilde{\mathbf{g}})$. From (41), we have $\tilde{d}_l = \tilde{d}(P) + \tilde{d}(\bar{P}) - \sum_{C \in \mathscr{C}} \varphi_l(C)\tilde{g}_C$. As $\tilde{\mathbf{g}} \le 0$, it follows that $\tilde{d}(P) + \tilde{d}(\bar{P}) \le \tilde{d}_l$, so only pairs of paths $(P, \bar{P})$ such that $\tilde{d}(P) + \tilde{d}(\bar{P}) \le \gamma$ can generate routes of reduced cost less than or equal to γ.

Phase 2 dynamically generates a sequence of path pairs $(P^{r_1}, \bar{P}^{s_1}), \ldots, (P^{r_k}, \bar{P}^{s_k}), \ldots, (P^{r_h}, \bar{P}^{s_h})$, where each path pair satisfies conditions (40) and $\tilde{d}(P^{r_1}) + \tilde{d}(\bar{P}^{s_1}) \le \cdots \le \tilde{d}(P^{r_k}) + \tilde{d}(\bar{P}^{s_k}) \le \cdots \le \tilde{d}(P^{r_h}) + \tilde{d}(\bar{P}^{s_h}) \le \gamma$. The pool $\mathscr{B}$ of routes generated contains any route $l \in \mathscr{R}$ resulting from the pairs of paths in the sequence such that $\tau_l \le T$, $\tilde{d}_l \le \gamma$, $d'_l \le \gamma'$, and route l is not dominated by any other route l' previously generated (i.e., route l' dominates route l if $R_{l'} = R_l$ and $\tilde{d}_{l'} \le \tilde{d}_l$).

4.4. Bounding Procedure H^2

Procedure H^2 is a straightforward implementation of CCG, where $\mathscr{R}$ is the set of elementary routes. The initial route set $\bar{\mathscr{R}}$ is obtained by executing GENR by setting $(\tilde{\mathbf{u}}, \tilde{\mathbf{v}}, \tilde{\mathbf{g}}) = (\mathbf{u}', \mathbf{v}', \mathbf{g}') = (\mathbf{u}^1, \mathbf{v}^1, \mathbf{g}^1)$, $\Delta(\mathscr{P}) = 10^7$, $\Delta(\mathscr{R}) = 10^4$, and $\gamma = \gamma' = z(ub) - \mathrm{LB}_1$, and adding all single customer routes to $\bar{\mathscr{R}} = \mathscr{B}$. H^2 starts with the same sets $\mathscr{S}$ and $\mathscr{C}$ of SC and WSR3 inequalities resulting from H^1. We initialize $(\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\omega}) = (\boldsymbol{\lambda}^1, \boldsymbol{\mu}^1, \boldsymbol{\omega}^1)$.

At each macro iteration, a set $\mathscr{N}$ of the routes of negative reduced cost with respect to $(\bar{\mathbf{u}}, \bar{\mathbf{v}}, \bar{\mathbf{g}})$ is given by $\mathscr{N} = \mathscr{B}$, where the set $\mathscr{B}$ is generated by GENR setting $(\tilde{\mathbf{u}}, \tilde{\mathbf{v}}, \tilde{\mathbf{g}}) = (\bar{\mathbf{u}}, \bar{\mathbf{v}}, \bar{\mathbf{g}})$, $(\mathbf{u}', \mathbf{v}', \mathbf{g}') = (\mathbf{u}^1, \mathbf{v}^1, \mathbf{g}^1)$, $\Delta(\mathscr{P}) = 10^7$, $\Delta(\mathscr{R}) = 150$, $\gamma = 0$, and $\gamma' = z(ub) - \mathrm{LB}_1$.

At the end, H^2 sets $(\mathbf{u}^2, \mathbf{v}^2, \mathbf{g}^2) = (\mathbf{u}^*, \mathbf{v}^*, \mathbf{g}^*)$, $(\boldsymbol{\lambda}^2, \boldsymbol{\mu}^2, \boldsymbol{\omega}^2) = (\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*, \boldsymbol{\omega}^*)$, and $\mathrm{LB}_2 = \mathrm{LCCG}$.

4.5. Generating the Path Set $\mathscr{P}^2$ and the Route Set $\mathscr{R}^2$

After having executed H^2, we try to generate the route set $\mathscr{R}^2$ of all routes of reduced cost with respect to $(\mathbf{u}^2, \mathbf{v}^2, \mathbf{g}^2)$ less than or equal to $z(ub) - \mathrm{LB}_2$. We call GENR by setting $(\tilde{\mathbf{u}}, \tilde{\mathbf{v}}, \tilde{\mathbf{g}}) = (\mathbf{u}^2, \mathbf{v}^2, \mathbf{g}^2)$, $(\mathbf{u}', \mathbf{v}', \mathbf{g}') = (\mathbf{u}^1, \mathbf{v}^1, \mathbf{g}^1)$, $\Delta(\mathscr{P}) = 10^7$, $\Delta(\mathscr{R}) = 5 \cdot 10^6$, $\gamma = z(ub) - \mathrm{LB}_2$, and $\gamma' = z(ub) - \mathrm{LB}_1$. We denote by $\mathscr{P}^2$ and $\mathscr{R}^2$ the path set $\mathscr{P}$ and the route set $\mathscr{B}$ generated by GENR, respectively. Moreover, we denote by $d^2(P)$ the modified cost of path $P \in \mathscr{P}^2$ with respect to $(\mathbf{u}^2, \mathbf{v}^2)$ computed according to expression (41) by replacing $(\tilde{\mathbf{u}}, \tilde{\mathbf{v}})$ with $(\mathbf{u}^2, \mathbf{v}^2)$.

We define $\mathscr{P}^2$ to be *optimal* if $|\mathscr{P}^2| < \Delta(\mathscr{P})$, because it contains any path generating any route of any optimal MTVRP solution. We define $\mathscr{R}^2$ to be *optimal* if $\mathscr{P}^2$ is optimal and $|\mathscr{R}^2| < \Delta(\mathscr{R})$.

Because H^3 and the exact algorithm require that $\mathscr{P}^2$ be optimal, whenever $\mathscr{P}^2$ is not optimal (i.e., $|\mathscr{P}^2| = \Delta(\mathscr{P})$), H^3 is not executed, and the entire procedure terminates prematurely.

4.6. Bounding Procedure H^3

H^3 is a CCG procedure, based on the simplex algorithm, to solve $(\overline{\text{LRF1}})$ with SC, SR3, SR5, and WT inequalities. H^3 differs from classical CCG algorithms because of the method used to solve the pricing problem.

H^3 uses sets $\mathscr{P}^2$ and $\mathscr{R}^2$, generated at the end of H^2, to generate routes of negative reduced cost. The route set $\bar{\mathscr{R}}$ of the master problem of H^3 is generated by extracting, from $\mathscr{R}^2$, the largest subset $\bar{\mathscr{R}}$ of routes of minimum reduced cost with respect to $(\mathbf{u}^2, \mathbf{v}^2, \mathbf{g}^2)$ such that $|\bar{\mathscr{R}}| \le \Delta(\mathscr{B})$ and by adding all single customer routes to $\bar{\mathscr{R}}$. The set $\mathscr{S}$ is set equal to the set of SC inequalities used in H^1 and H^2, and $\mathscr{W}$ and $\mathscr{C}$ are defined as $\mathscr{W} = \varnothing$, $\mathscr{C} = \varnothing$, where the set $\mathscr{C} = \mathscr{C}^3 \cup \mathscr{C}^5$ represents SR3 and SR5 inequalities.

Let $\bar{\mathbf{x}}$ and $(\bar{\mathbf{u}}, \bar{\mathbf{v}}, \bar{\mathbf{g}}, \bar{\mathbf{h}})$ be, respectively, the primal and dual solutions of the master problem achieved at a given iteration. A set $\mathscr{N}$ of at most $\Delta(\mathscr{N})$ negative reduced cost routes with respect to $(\bar{\mathbf{u}}, \bar{\mathbf{v}}, \bar{\mathbf{g}}, \bar{\mathbf{h}})$ is generated with the following algorithm.

1. Extract, from $\mathscr{R}^2$, the largest subset $\mathscr{N}$ of at most $\Delta(\mathscr{N})$ routes having the largest negative reduced cost, and add them to $\bar{\mathscr{R}}$. If $\mathscr{N} = \varnothing$ and $\mathscr{R}^2$ is not optimal, continue with Step 2; otherwise, Stop.
2. Compute the modified cost $\bar{d}(P)$, $\forall\, P \in \mathscr{P}^2$, with respect to $(\bar{\mathbf{u}}, \bar{\mathbf{v}})$ according to (41) by replacing $(\tilde{\mathbf{u}}, \tilde{\mathbf{v}})$ with $(\bar{\mathbf{u}}, \bar{\mathbf{v}})$. Using the method described in §4.3.2, combine any pair of paths $P, P' \in \mathscr{P}^2$ such that $d^2(P) + d^2(P') \le z(ub) - \mathrm{LB}_2$ and $\bar{d}(P) + \bar{d}(P') \le 0$ to derive the set $\mathscr{N}$.

At each iteration, H^3 adds, to $\mathscr{C}$, a set of at most $\Delta(\mathscr{C})$ most violated SR3 and SR5 inequalities and, to $\mathscr{W}$, a set of at most $\Delta(\mathscr{W})$ WT inequalities most violated by $\bar{\mathbf{x}}$. H^3 ends whenever $\mathscr{N} = \varnothing$ and no cuts are added, and it achieves an $(\overline{\text{LRF1}})$ dual solution $(\mathbf{u}^3, \mathbf{v}^3, \mathbf{g}^3, \mathbf{h}^3)$ of cost LB_3.

4.7. Generating the Route Set $\mathscr{R}^3$ and Solving (F1)

After having executed H^3, we try to generate the largest set $\mathscr{R}^3$ of routes such that $d^3_l \le z(ub) - \mathrm{LB}_3$, $\forall\, l \in \mathscr{R}^3$, and $|\mathscr{R}^3| \le \Delta(\mathscr{R})$, where d^3_l is the reduced cost of route l with respect to $(\mathbf{u}^3, \mathbf{v}^3, \mathbf{g}^3, \mathbf{h}^3)$. In generating $\mathscr{R}^3$ we have two cases:

a. $\mathscr{R}^2$ is optimal. We set $\mathscr{R}^3 = \{l \in \mathscr{R}^2\colon d^3_l \le z(ub) - \mathrm{LB}_3\}$ and define $\mathscr{R}^3$ as *optimal*.

b. $\mathscr{R}^2$ is not optimal. Recall that in this case $\mathscr{P}^2$ is optimal, so $\mathscr{P}^2$ allows us to generate the routes of any optimal MTVRP solution. We compute the reduced cost $d^3(P)$, $\forall\, P \in \mathscr{P}^2$, with respect to $(\mathbf{u}^3, \mathbf{v}^3)$ according to expression (41) by replacing $(\tilde{\mathbf{u}}, \tilde{\mathbf{v}})$ with $(\mathbf{u}^3, \mathbf{v}^3)$. Then, we combine the path pairs of $\mathscr{P}^2$, by using the method of §4.3.2, to derive the set $\mathscr{R}^3$ of all routes such that $d_l^2 \le z(ub) - \mathrm{LB}_2$ and $d_l^3 \le z(ub) - \mathrm{LB}_3, \forall\, l \in \mathcal{R}^3$, and $|\mathcal{R}^3| \le \Delta(\mathcal{R})$. If $|\mathcal{R}^3| < \Delta(\mathcal{R})$, then $\mathcal{R}^3$ is defined as *optimal*.

If $|\mathcal{R}^3|$ is small (say, $|\mathcal{R}^3| \le 5{,}000$), we find an optimal MTVRP solution by solving the reduced problem (F1), obtained by replacing route set $\mathcal{R}$ with $\mathcal{R}^3$, through an IP solver. Otherwise (if $|\mathcal{R}^3| > 5{,}000$), we execute bounding procedure H^4 (described in §5).

5. Bounding Procedure H^4 Based on Relaxation ($\overline{\mathrm{LF2}}$)

H^4 is a CCG method, based on the simplex, to solve relaxation ($\overline{\mathrm{LF2}}$).

The initial sets $\mathcal{S}$ and $\mathcal{C}$ of SC, SR3, and SR5 inequalities are those resulting at the end of H^3. The initial master problem, $\bar{\mathcal{H}} \subseteq \mathcal{H}$, is defined as follows. Execute procedure GENSCHED (described in §6) to derive the schedule set $\mathcal{D}$ containing at most $\Delta(\mathcal{D})$ (where $\Delta(\mathcal{D})$ is a given parameter) schedules such that $c_k^3 \le z(ub) - \mathrm{LB}_3, \forall\, k \in \mathcal{D}$, where c_k^3 is the reduced cost of schedule k with respect to $(\mathbf{u}^3, \mathbf{v}^3, \mathbf{g}^3, \mathbf{h}^3)$ computed as $c_k^3 = \sum_{l \in \Omega_k} d_l^3$. Set $\bar{\mathcal{H}} = \mathcal{D}$, and add to $\bar{\mathcal{H}}$ a schedule composed of a single customer route for every customer i not covered by any schedule of $\mathcal{D}$.

At each iteration, GENSCHED generates a set $\mathcal{N}$ of at most $\Delta(\mathcal{D})$ schedules of negative reduced cost with respect to the dual solution $(\bar{\mathbf{w}}, \bar{\mathbf{v}}, \bar{\mathbf{g}})$ of the master and such that $c_k^3 \le z(ub) - \mathrm{LB}_3\ \forall\, k \in \mathcal{N}$. Moreover, the subset $\mathcal{C}'$ of the $\Delta(\mathcal{C})$ SR3 and SR5 inequalities most violated by the current master solution is added to $\mathcal{C}$. In our computational experiments, we used $\Delta(\mathcal{D}) = 10^4$ in generating $\mathcal{D}$, $\Delta(\mathcal{D}) = 150$ in generating $\mathcal{N}$, and $\Delta(\mathcal{C}) = 100$ in generating $\mathcal{C}'$.

H^4 terminates if $\mathcal{N} = \varnothing$ and $\mathcal{C}' = \varnothing$ and provides a dual solution $(\mathbf{w}^4, \mathbf{v}^4, \mathbf{g}^4)$ of cost LB_4.

6. Algorithm GENSCHED to Generate Vehicle Schedules

In this section, we describe procedure GENSCHED to generate the schedule sets $\bar{\mathcal{H}}$ and $\mathcal{N}$, which are required by bounding procedure H^4, and the schedule subset $\mathcal{H}^4 \subseteq \mathcal{H}$, which is required by the exact method described in §7.

Given a dual solution $(\hat{\mathbf{w}}, \hat{\mathbf{v}}, \hat{\mathbf{g}})$ of ($\overline{\mathrm{LF2}}$) and two parameters $\Delta(\mathcal{D})$ and γ, GENSCHED combines the routes of set $\mathcal{R}^3$ to produce the largest subset $\mathcal{D}$ of schedules of minimum reduced costs with respect to $(\hat{\mathbf{w}}, \hat{\mathbf{v}}, \hat{\mathbf{g}})$ such that $|\mathcal{D}| \le \Delta(\mathcal{D})$, $c_k^3 \le z(ub) - \mathrm{LB}_3$, and $\hat{c}_k \le \gamma\ \forall\, k \in \mathcal{D}$, where $\hat{c}_k$ is the reduced cost of schedule k with respect to $(\hat{\mathbf{w}}, \hat{\mathbf{v}}, \hat{\mathbf{g}})$.

To generate $\bar{\mathcal{H}}$, $\mathcal{N}$, and $\mathcal{H}^4$, GENSCHED requires the following parameter settings:

- *The schedule set $\bar{\mathcal{H}}$ defining the initial master problem of H^4.* Define $(\hat{\mathbf{w}}, \hat{\mathbf{v}}, \hat{\mathbf{g}}) = (\mathbf{0}, \mathbf{0}, \mathbf{0})$ and $\gamma = z(ub)$.
- *The set $\mathcal{N}$ of schedules of negative reduced costs in H^4.* Define $(\hat{\mathbf{w}}, \hat{\mathbf{v}}, \hat{\mathbf{g}})$ as the dual of the current master problem of H^4 and $\gamma = 0$.
- *The schedule subset $\mathcal{H}^4 \subseteq \mathcal{H}$ of all the schedules of any optimal MTVRP solution.* Define $(\hat{\mathbf{w}}, \hat{\mathbf{v}}, \hat{\mathbf{g}}) = (\mathbf{w}^4, \mathbf{v}^4, \mathbf{g}^4)$ and $\gamma = z(ub) - \mathrm{LB}_4$.

Procedure GENSCHED is based on the following observation.

Proposition 4. *Let* $\hat{d}_l = \sum_{\{i,j\} \in E(R_l)} (\alpha_{ij} - \frac{1}{2}(\hat{w}_i + \hat{w}_j) - \sum_{S \in \mathcal{S} : \{i,j\} \in \delta(S)} \hat{v}_S) - \sum_{C \in \mathcal{C}} \varphi_l(C) \hat{g}_C, \forall\, l \in \mathcal{R}^3$, *and let* $\hat{d}(\Omega_k) = \sum_{l \in \Omega_k} \hat{d}_l\ \forall\, k \in \mathcal{H}$. *The reduced cost* $\hat{c}_k$ *of schedule* $k \in \mathcal{H}$ *with respect to* $(\hat{\mathbf{w}}, \hat{\mathbf{v}}, \hat{\mathbf{g}})$ *satisfies the inequality* $\hat{c}_k \ge \hat{d}(\Omega_k)$.

Proof. This happens as $w_0 \le 0$, $\hat{\mathbf{g}} \le \mathbf{0}$, and $\varphi_k(C) \ge \sum_{l \in \Omega_k} \varphi_l(C)\ \forall\, C \in \mathcal{C}\ \forall\, k \in \mathcal{H}$. □

GENSCHED is a two-phase procedure. In the first phase, it computes $\hat{d}_l, \forall\, l \in \mathcal{R}^3$, and orders the routes of the set $\mathcal{R}^3$ so that $\hat{d}_{l_1} \le \hat{d}_{l_2} \le \cdots \le \hat{d}_{|\mathcal{R}^3|}$. The second phase is an iterative procedure that combines the routes $\mathcal{R}^3$ in schedules. At iteration j, an attempt is made to add the jth route of the ordered set $\mathcal{R}^3$ to the schedules of a temporary schedule set POOL containing all the schedules composed of the first $j-1$ routes $l_1, l_2, \ldots, l_{j-1}$ of $\mathcal{R}^3$.

We assume that POOL is composed of *undominated* schedules, where schedule k' is *dominated* by schedule k'' if $V(\Omega_{k''}) = V(\Omega_{k'})$, $\hat{d}(\Omega_{k''}) \le \hat{d}(\Omega_{k'})$, and $\tau(\Omega_{k''}) \le \tau(\Omega_{k'})$. A schedule k' that is obtained by adding route l_j to a schedule $k \in$ POOL becomes a new member of POOL only if $\hat{d}(\Omega_{k'}) \le \gamma$, k' is not dominated by any other schedule $k'' \in$ POOL, and $c^3(\Omega_{k'}) = \sum_{l \in \Omega_{k'}} d_l^3 \le z(ub) - \mathrm{LB}_3$.

We denote by p the number of schedules in POOL. The second phase of GENSCHED can be described as follows.

1. Initialize POOL $= \varnothing$, $\mathcal{D} = \varnothing$, and $p = 0$.
2. For $l = 1, \ldots, |\mathcal{R}^3|$, repeat step 3.
3. *Expansion of the temporary schedule set* POOL *with route* $l \in \mathcal{R}^3$. For each $k \in$ POOL such that $V(\Omega_k) \cap R_l = \varnothing$, $\tau(\Omega_k) + \tau_l \le T$, $c^3(\Omega_k) + d_l^3 \le z(ub) - \mathrm{LB}_3$, and $\hat{d}(\Omega_k) + \hat{d}_l \le \gamma$, repeat the following steps:
   - Define the schedule $\Omega' = \Omega_k \cup \{l\}$ of duration $\tau(\Omega') = \tau(\Omega_k) + \tau_l$, and set $\hat{d}(\Omega') = \hat{d}(\Omega) + \hat{d}_l$.
   - If Ω' is not dominated by any other schedule $\Omega'' \in$ POOL, insert $\Omega' \in$ POOL and set $p = p + 1$. Remove any schedule Ω'' dominated by Ω' from POOL, and update p accordingly.
   - Compute $\hat{c}(\Omega')$. If $\hat{c}(\Omega') \le \gamma$, add Ω' to $\mathcal{D}$. If $|\mathcal{D}| = p$, Stop.

7. Description of the Exact Method to Solve the MTVRP

The exact method we propose consists of seven main steps, described as follows.

1. **Solving relaxation ($\overline{\text{LRF1}}$) with procedures H^1 and H^2.** Solve relaxation ($\overline{\text{LRF1}}$) with SC and WSR3 inequalities, executing, in sequence, procedures H^1 and H^2. Let LB_2 be the cost of the dual solution $(\mathbf{u}^2, \mathbf{v}^2, \mathbf{g}^2)$ achieved by H^2 (see §§4.2 and 4.4).

2. **Generating path set $\mathcal{P}^2$ and route set $\mathcal{R}^2$.** Generate path set $\mathcal{P}^2$ and set $\mathcal{R}^2$ of the routes of reduced cost with respect to $(\mathbf{u}^2, \mathbf{v}^2, \mathbf{g}^2)$ less than or equal to $z(ub) - \text{LB}_2$ (see §4.5). If $\mathcal{P}^2$ is not optimal, Stop (the algorithm terminates without providing any optimal solution).

3. **Solving relaxation ($\overline{\text{LRF1}}$) with procedure H^3.** Solve relaxation ($\overline{\text{LRF1}}$) with SC, SR3, SR5, and WT inequalities with procedure H^3 (see §4.6). Let LB_3 be the lower bound corresponding to the dual solution $(\mathbf{u}^3, \mathbf{v}^3, \mathbf{g}^3, \mathbf{h}^3)$ of ($\overline{\text{LRF1}}$) achieved by H^3.

4. **Generating the route set $\mathcal{R}^3$.** Generate the largest set $\mathcal{R}^3$ of the routes of reduced cost with respect to $(\mathbf{u}^3, \mathbf{v}^3, \mathbf{g}^3, \mathbf{h}^3)$ less than or equal to $z(ub) - \text{LB}_3$ and such that $|\mathcal{R}^3| \le \Delta(\mathcal{R})$ (see §4.7). If $|\mathcal{R}^3| < \Delta(\mathcal{R})$, $\mathcal{R}^3$ contains the routes of any optimal solution of cost less than or equal to $z(ub)$ and $\mathcal{R}^3$ is defined as *optimal*. Otherwise, $\mathcal{R}^3$ is defined as *not optimal*.

5. **Solving the reduced problem** (F1)**.** If $m = 1$, or if $m > 1$, $\mathcal{R}^3$ is optimal, and $|\mathcal{R}^3| \le 5{,}000$, then find an optimal MTVRP solution by solving, with an IP solver, problem (F1) by replacing the route set $\mathcal{R}$ with the route set $\mathcal{R}^3$, and Stop.

6. **Solving relaxation ($\overline{\text{LF2}}$) with procedure H^4.** If $m > 1$, and either $\mathcal{R}^3$ is not optimal or $|\mathcal{R}^3| > 5{,}000$, then solve ($\overline{\text{LF2}}$) with SC, SR3, and SR5 inequalities with H^4 (see §5). Let LB_4 be the lower bound corresponding to an optimal dual solution $(\mathbf{w}^4, \mathbf{v}^4, \mathbf{g}^4)$ of ($\overline{\text{LF2}}$) obtained by H^4. If the optimal primal solution $\mathbf{y}$ of ($\overline{\text{LF2}}$) is integer, Stop.

7. **Solving the reduced problem** (F2)**.** Call GENSCHED to generate the largest set of schedules $\mathcal{H}^4 \subseteq \mathcal{H}$ by setting $(\tilde{\mathbf{w}}, \tilde{\mathbf{v}}, \tilde{\mathbf{g}}) = (\mathbf{w}^4, \mathbf{v}^4, \mathbf{g}^4)$, $\gamma = z(ub) - \text{LB}_4$, and $\Delta(\mathcal{D}) = 10^6$.

If $|\mathcal{H}^4| \ge \Delta(\mathcal{D})$, as there is no guarantee that $\mathcal{H}^4$ contains all schedules of any optimal MTVRP solution, the algorithm terminates prematurely without providing any optimal solution. Otherwise (i.e., if $|\mathcal{H}^4| < \Delta(\mathcal{D})$), compute an optimal MTVRP solution by solving, with an IP solver, the integer problem (F2) obtained by replacing the schedule set $\mathcal{H}$ with the subset $\mathcal{H}^4$ and by adding all the valid inequalities saturated at the end of H^4.

Table 1 **Upper Bounds on the Test Instances**

Inst	*m*	*T*	$z(ub)$	CPU_{UB}
CMT-1	1	551	524.61	27
	2	275	533.00	27
	1	577	524.61	27
	2	289	529.85	28
	3	192	552.68	30
	4	144	546.29	28
CMT-2	1	877	835.26	49
	2	439	835.26	49
	3	292	835.26	49
	4	219	835.26	50
	5	175	835.80	50
	6	146	857.00	111
	1	919	835.26	49
	2	459	835.26	49
	3	306	835.26	50
	4	230	835.26	50
	5	184	835.26	51
	6	153	839.22	51
	7	131	872.64	51
CMT-3	1	867	826.14	106
	2	434	826.14	105
	3	289	826.14	107
	4	217	828.73	105
	5	173	851.47	107
	6	145	839.90	172
	1	909	826.14	106
	2	454	826.14	106
	3	303	826.14	105
	4	227	826.14	107
	5	182	833.15	109
	6	151	842.21	109
CMT-11	1	1,094	1, 042.11	156
	2	547	1,042.11	161
	3	365	1,042.11	160
	4	274	1,085.92	210
	5	219	1,042.11	161
	1	1,146	1,042.11	160
	2	573	1,042.11	159
	3	382	1,042.11	160
	4	287	1,042.11	161
	5	229	1,042.11	164
CMT-12	1	861	819.56	93
	2	430	819.56	95
	3	287	819.56	95
	4	215	819.56	95
	5	172	845.37	95
	1	902	819.56	95
	2	451	819.56	93
	3	301	819.56	93
	4	225	819.56	96
	5	180	824.78	96
	6	150	825.36	96

8. Computational Results

This section reports on the computational results of the exact method described in §7. All algorithms were coded in C and compiled with Visual Studio 2008. CPLEX 12.1 was used as the LP solver in H^3 and H^4 and as the IP solver in steps 5 and 7 of the exact method. All tests were performed on a Sony Vaio P8400 laptop (Intel Core 2 Duo@2.26 GHz with 4 GB of RAM).

The exact algorithm was tested on a subset of the benchmark instances proposed in Taillard et al. (1996), which were used to test all heuristic algorithms in the literature. These instances were generated by starting

Table 2 Computational Results on CMT-1, CMT-2, and CMT-3 Instances

					Formulation $F1$								Solving $F1$		Formulation $F2$				Solving $F2$	
Inst	m	T	$z(ub)$	z^*	$\%LB_1$	$\%LB_2$	$\mathscr{P}^2$	$\mathscr{R}^2$	LB_3	$\%LB_3$	CPU_3	$\mathscr{R}^3$	CPU_{IP}	CPU_T	LB_4	$\%LB_4$	CPU_4	$\mathscr{H}^4$	CPU_{IP}	CPU_T
CMT-1	1	551	524.61	524.61	99.6	99.8	2,549	329	524.61	100.0	9.4	47	0.1	9.5						
($n = 50$)	2	275	533.00	533.00	98.1	98.3	142,645	95,763	532.22	99.9	36.3	509	0.9	37.2						
	1	577	524.61	524.61	99.6	99.8	1,922	329	524.61	100.0	10.3	62	0.5	10.8						
	2	289	529.85	529.85	98.7	98.9	35,676	16,669	527.68	99.6	15.0	806	1.9	16.9						
	3	192	552.68		94.6	94.8	m.o.	—	—	—	34.8									
	4	144	546.29	546.29	95.7	96.0	2,365,662	3,941,493	543.50	99.5	588.8	1,268	15.5	603.3						
CMT-2	1	877	835.26	835.26	98.5	98.9	138,112	114,078	832.78	99.7	205.9	3,117	10.6	216.5						
($n = 75$)	2	439	835.26	835.26	98.5	98.9	138,112	114,078	832.78	99.7	205.9	3,117	16.5	222.4						
	3	292	835.26	835.26	98.5	98.9	138,112	114,078	832.78	99.7	205.9	3,117	21.0	226.9						
	4	219	835.26	835.26	98.5	98.9	138,112	114,078	832.78	99.7	205.9	3,117	27.7	233.6						
	5	175	835.80	835.80	98.5	98.8	170,128	148,826	832.77	99.6	188.8	4,746	336.0	524.8						
	6	146	857.00		96.0	96.4	m.o.	—	—	—	48.3									
	1	919	835.26	835.26	98.5	98.9	138,112	114,078	832.78	99.7	205.9	3,117	10.6	216.5						
	2	459	835.26	835.26	98.5	98.9	138,112	114,078	832.78	99.7	205.9	3,117	16.5	222.4						
	3	306	835.26	835.26	98.5	98.9	138,112	114,078	832.78	99.7	205.9	3,117	21.4	227.3						
	4	230	835.26	835.26	98.5	98.9	138,112	114,078	832.78	99.7	205.9	3,117	28.6	234.5						
	5	184	835.26	835.26	98.5	98.9	138,112	114,078	832.78	99.7	205.9	3,117	54.2	260.1						
	6	153	839.22	839.22	98.1	98.4	472,756	551,114	832.79	99.2	221.4	49,511	⇒		834.12	99.4	374.2	343,626	4,209.1	4,583.3
	7	131	872.64		94.3	94.7	m.o.	—	—	—	47.6									
CMT-3	1	867	826.14	826.14	98.7	99.0	5,248,271	$> 5 \cdot 10^6$	826.14	100.0	2,766.7	1,471	0.1	2,766.8						
($n = 100$)	2	434	826.14	826.14	98.7	99.0	5,248,271	$> 5 \cdot 10^6$	826.14	100.0	2,766.7	1,471	15.0	2,781.7						
	3	289	826.14	826.14	98.7	99.0	5,248,271	$> 5 \cdot 10^6$	826.14	100.0	2,766.7	1,471	27.2	2,793.9						
	4	217	828.73		98.4	98.7	m.o.	—	—	—	96.2									
	5	173	851.47[a]		97.6	97.9	m.o.	—	—	—	95.7									
	6	145	839.90[b]		97.5	97.9	m.o.	—	—	—	93.4									
	1	909	826.14	826.14	98.7	99.0	5,248,271	$> 5 \cdot 10^6$	826.14	100.0	2,766.7	1,471	0.1	2,766.8						
	2	454	826.14	826.14	98.7	99.0	5,248,271	$> 5 \cdot 10^6$	826.14	100.0	2,766.7	1,471	15.3	2,782.0						
	3	303	826.14	826.14	98.7	99.0	5,248,271	$> 5 \cdot 10^6$	826.14	100.0	2,766.7	1,471	28.5	2,795.2						
	4	227	826.14	826.14	98.7	99.0	5,248,271	$> 5 \cdot 10^6$	826.14	100.0	2,766.7	1,471	46.6	2,813.3						
	5	182	833.15		97.9	98.2	m.o.	—	—	—	94.1									
	6	151	842.21[b]		97.5	97.8	m.o.	—	—	—	88.0									

[a] Best-known upper bound 836.01 from Alonso et al. (2008).

[b] Best-known upper bound 836.21 from Olivera (2005).

Table 3 Computational Results on CMT-11 and CMT-12 Instances

					Formulation $F1$								Solving $F1$		Formulation $F2$				Solving $F2$	
Inst	m	T	$z(ub)$	z^*	%LB$_1$	%LB$_2$	$\mathcal{P}^2$	$\mathcal{R}^2$	LB$_3$	%LB$_3$	CPU$_3$	$\mathcal{R}^3$	CPU$_{IP}$	CPU$_T$	LB$_4$	%LB$_4$	CPU$_4$	$\mathcal{H}^4$	CPU$_{IP}$	CPU$_T$
CMT-11	1	1,094	1,042.11	1,042.11	99.6	99.9	185,303	4,705	1,042.11	100.0	632.1	83	0.2	632.3						
($n = 120$)	2	547	1,042.11	1,042.11	99.6	99.9	185,303	4,705	1,042.11	100.0	632.1	83	0.2	632.3						
	3	365	1,042.11	1,042.11	99.6	99.9	185,303	4,705	1,042.11	100.0	632.1	83	0.2	632.3						
	4	274	1,085.92[a]		96.3	96.6	m.o.	—	—	—	776.1									
	5	219	1,042.11	1,042.11	99.6	99.9	182,530	5,102	1,042.11	100.0	620.0	105	0.2	620.2						
	1	1,146	1,042.11	1,042.11	99.6	99.9	185,303	4,705	1,042.11	100.0	632.1	83	0.2	632.3						
	2	573	1,042.11	1,042.11	99.6	99.9	185,303	4,705	1,042.11	100.0	632.1	83	0.2	632.3						
	3	382	1,042.11	1,042.11	99.6	99.9	185,303	4,705	1,042.11	100.0	632.1	83	0.2	632.3						
	4	287	1,042.11	1,042.11	99.6	99.9	185,303	4,705	1,042.11	100.0	631.0	83	0.2	631.2						
	5	229	1,042.11	1,042.11	99.6	99.9	185,303	4,705	1,042.11	100.0	619.8	83	0.2	620.0						
CMT-12	1	861	819.56	819.56	99.7	100.0	49,368	71	819.56	100.0	39.9	24	0.3	40.2						
($n = 100$)	2	430	819.56	819.56	99.7	100.0	46,578	60	819.56	100.0	44.7	21	0.3	45.0						
	3	287	819.56	819.56	99.7	100.0	43,219	95	819.56	100.0	43.0	23	0.2	43.2						
	4	215	819.56	819.56	99.7	100.0	46,578	60	819.56	100.0	45.0	21	0.2	45.2						
	5	172	845.37		96.6	96.9	m.o.	—	—	—	16.6									
	1	902	819.56	819.56	99.7	100.0	49,368	71	819.56	100.0	39.9	24	0.3	40.2						
	2	451	819.56	819.56	99.7	100.0	46,578	60	819.56	100.0	44.7	21	0.3	45.0						
	3	301	819.56	819.56	99.7	100.0	43,219	95	819.56	100.0	43.0	23	0.3	43.3						
	4	225	819.56	819.56	99.7	100.0	46,578	60	819.56	100.0	45.0	21	0.5	45.5						
	5	180	824.78	824.78	99.0	99.4	1,857,419	818,631	819.56	99.4	314.6	746,542	⇒		823.69	99.9	2,290.3	274,857	77.6	2,367.9
	6	150	825.36	823.14	99.2	99.6	2,724,943	1,717,379	819.56	99.6	402.5	1,520,016	⇒		823.14	100.0	1,058.2	3,819	0.2	1,058.4

[a] Best-known upper bound 1, 078.64 from Olivera (2005).

from the graphs, demands, and vehicle capacities of five CVRP problems (namely, CMT-1, CMT-2, CMT-3, CMT-11, and CMT-12) proposed in Christofides et al. (1979) and by varying the number of vehicles m. For each problem and each value of m, two instances with different values of T were generated: in the first instance, $T = [1.05 z_{RT}/m]$, and in the second one, $T = [1.10 z_{RT}/m]$, where $[x]$ is the integer value nearest to x and z_{RT} is the CVRP solution cost obtained by Rochat and Taillard (1995). All instances feature x-y coordinates. The travel costs coincide with the travel times and are real values computed as Euclidean distances between the vertices. For an accurate description of the test instances, see Taillard et al. (1996).

The exact method described in §7 requires an upper bound $z(ub)$ (see steps 2, 4, and 7). In our experiments, such an upper bound was set equal to the best of three upper bounds: (a) the upper bound computed by the heuristic of Taillard et al. (1996) (sent to us by private communication; see Taillard 2009), (b) the upper bound reported in Olivera and Viera (2007), and (c) the cost of the CVRP solution found by the heuristic of Mester and Bräysy (2005) if the routes of such solution can be arranged so as to obtain a feasible MTVRP solution.

For each instance, Table 1 reports the number of vehicles (m), the maximum driving time (T), the upper bound ($z(ub)$), and the estimated CPU time for computing $z(ub)$ ($\mathrm{CPU_{UB}}$) on our machine. The estimated CPU time $\mathrm{CPU_{UB}}$ is the sum of the CPU times of the algorithms of Taillard et al. (1996), Olivera and Viera (2007), and Mester and Bräysy (2005) normalized to our machine by considering that, according to http://www.cpubenchmark.net/cpu_list.php, our machine is ≈100 times faster than the 100 MHz machine used by Taillard et al. (1996), 4 times faster than the 1.8 GHz AMD Athlon XP 2200+ used by Olivera and Viera (2007), and 6.3 times faster than the Pentium 4.2 GHz used by Mester and Bräysy (2005).

We did not test our algorithm on the four instances for which no feasible solution is known (namely, instances CMT-1 with $m = 3$ and $T = 184$, CMT-1 with $m = 4$ and $T = 138$, CMT-2 with $m = 7$ and $T = 125$, and CMT-12 with $m = 6$ and $T = 143$) because the exact algorithm described in §7 would run out of memory at step 2 while generating the path set $\mathscr{P}^2$.

Tables 2 and 3 report detailed computational results. For each instance, the columns report the following information: instance name, *Inst*; number of vehicles, m; maximum driving time, T; initial upper bound, $z(ub)$; optimal solution cost, z^*; percentage lower bound, $\%\mathrm{LB}_k$ ($k = 1, \ldots, 4$), achieved by bounding procedure H^k; lower bounds, LB_3 and LB_4, achieved by bounding procedures H^3 and H^4, respectively; cardinality of the sets $\mathscr{P}^2$, $\mathscr{R}^2$, $\mathscr{R}^3$, and $\mathscr{H}^4$; global time spent, $\mathrm{CPU_3}$, to execute H^1, H^2, and H^3; time spent, $\mathrm{CPU_4}$, by H^4 (including $\mathrm{CPU_3}$); time spent, $\mathrm{CPU_{IP}}$, by CPLEX to solve (F1) or (F2); total computing time, CPU_T. In column $\mathscr{P}^2$, "m.o." means that the exact method terminated prematurely at step 2 because $\mathscr{P}^2$ was not optimal (i.e., $|\mathscr{P}^2| > \Delta(\mathscr{P}) = 10^7$). In this case, $\mathrm{CPU_3}$ indicates the time spent by H^1 and H^2 plus the computing time for generating $\mathscr{P}^2$.

Tables 2 and 3 show that the proposed method could solve 42 out of 52 test instances. All but one of the instances were solved within one hour of CPU time. Thirty-five of the 42 instances solved to optimality have the optimal solution cost equal to the cost of the corresponding CVRP instance. The other seven instances have optimal solution cost slightly greater than the cost of the corresponding CVRP instance.

The lower bounds proposed seem to be effective. The lower bounds achieved by H^1 and H^2 are, on average, 98.6% and 98.9% with respect to the best-known solution cost, respectively. Bounding procedure H^3 closed the gap of 26 instances. Only three instances were solved by using formulation (F2). In all such cases, lower bound LB_4 is strictly greater than LB_3, requiring the generation of few schedules (set $\mathscr{H}^4$) in step 7 with respect to the routes generated at step 4 (set $\mathscr{R}^3$).

For the sake of fairness, we have to say that the performance of the proposed algorithm strongly depends on the initial upper bound, $z(ub)$, used at steps 2, 4, and 7. For example, by increasing such an upper bound by 1%, the exact method, as it is, could not solve any of the instances of classes CMT-2, CMT-3, and CMT-11, and it could solve only 12 out of 17 instances of classes CMT-1 and CMT-12. Anyway, as shown in Table 1, high-quality upper bounds can be computed in a few minutes by running the heuristic algorithms of Taillard et al. (1996), Olivera and Viera (2007), and Mester and Bräysy (2005).

9. Conclusions

In this paper, we describe an exact method to solve the MTVRP based on two set-partitioning-like formulations. We describe four different bounding procedures, based on the linear relaxations of both formulations, enforced by valid inequalities, that are embedded into an exact solution method. The computational results show that the proposed exact algorithm can solve, to optimality, 42 out of 52 benchmark instances involving up to 120 customers used in the literature by the heuristic algorithms.

Acknowledgments

The authors thank two anonymous referees for helpful comments. This work has been partially supported by the Italian MIUR (Ministero dell'Istruzione, dell'Universitá e della Ricerca) under PRIN 2008 Research Project 20088SBBH9.

References

Alonso, F., M. J. Alvarez, J. E. Beasley. 2008. A tabu search algorithm for the periodic vehicle routing problem with multiple vehicle trips and accessibility restrictions. *J. Oper. Res. Soc.* **59**(7) 963–976.

Azi, N., M. Gendreau, J.-Y. Potvin. 2010. An exact algorithm for a vehicle routing problem with time windows and multiple use of vehicles. *Eur. J. Oper. Res.* **202**(3) 756–763.

Baldacci, R., N. Christofides, A. Mingozzi. 2008. An exact algorithm for the vehicle routing problem based on the set partitioning formulation with additional cuts. *Math. Programming Ser. A* **115**(2) 351–385.

Baldacci, R., A. Mingozzi, R. Roberti. 2011. New route relaxation and pricing strategies for the vehicle routing problem. *Oper. Res.* **59**(5) 1269–1283.

Brandão, J. C. S., A. Mercer. 1997. A tabu search algorithm for the multi-trip vehicle routing and scheduling problem. *Eur. J. Oper. Res.* **100**(1) 180–191.

Brandão, J. C. S., A. Mercer. 1998. The multi-trip vehicle routing problem. *J. Oper. Res. Soc.* **49**(8) 799–805.

Christofides, N., A. Mingozzi, P. Toth. 1979. The vehicle routing problem. N. Christofides, A. Mingozzi, P. Toth, C. Sandi, eds. *Combinatorial Optimization*. John Wiley & Sons, Chichester, UK, 315–338.

Christofides, N., A. Mingozzi, P. Toth. 1981. Exact algorithms for the vehicle routing problem, based on spanning tree and shortest path relaxations. *Math. Programming Ser. A* **20**(1) 255–280.

Fleischmann, B. 1990. The vehicle routing problem with multiple use of vehicles. Working paper, Facbereich Wirtschaftswissenschafte, Universitat Hamburg, Hamburg, Germany.

Gribkovskaia, I., B. O. Gullberg, K. J. Hovden, S. W. Wallace. 2006. Optimization model for a livestock collection problem. *Internat. J. Physical Distribution Logist. Management* **36**(2) 136–152.

Jepsen, M., B. Petersen, S. Spoorendonk, D. Pisinger. 2008. Subset-row inequalities applied to the vehicle-routing problem with time windows. *Oper. Res.* **56**(2) 497–511.

Mester, D., O. Bräysy. 2005. Active guided evolution strategies for large-scale vehicle routing problems with time windows. *Comput. Oper. Res.* **32**(6) 1593–1614.

Olivera, A. 2005. Memorias adaptativas para el problema de ruteo de vehículos con múltiples viajes. Ph.D. thesis, Instituto de Computación, Facultad de Ingeniería, Universidad de la República, Montevideo, Uruguay.

Olivera, A., O. Viera. 2007. Adaptive memory programming for the vehicle routing problem with multiple trips. *Comput. Oper. Res.* **34**(1) 28–47.

Petch, R. J., S. Salhi. 2004. A multi-phase constructive heuristic for the vehicle routing problem with multiple trips. *Discrete Appl. Math.* **133**(1–3) 69–92.

Rochat, Y., É. D. Taillard. 1995. Probabilistic intensification and diversification in local search for vehicle routing. *J. Heuristics* **1**(1) 147–167.

Salhi, S., R. J. Petch. 2007. A GA based heuristic for the vehicle routing problem with multiple trips. *J. Math. Model. Algorithms* **6**(4) 591–613.

Taillard, É. D. 2009. Private communication. (June 16).

Taillard, É. D., G. Laporte, M. Gendreau. 1996. Vehicle routeing with multiple use of vehicles. *J. Oper. Res. Soc.* **47**(8) 1065–1070.