# Dissimilar Arc Routing Problems 

Miguel Constantino<br>Universidade de Lisboa, Faculdade de Ciências, DEIO, Lisboa, Portugal<br>CMAF-CIO - Centro de Matemática, Aplicações Fundamentais e Investigação Operacional, Lisboa, Portugal

M. Cândida Mourão (©)<br>Universidade de Lisboa, Instituto Superior de Economia e Gestão, Lisboa, Portugal<br>CMAF-CIO - Centro de Matemática, Aplicações Fundamentais e Investigação Operacional, Lisboa, Portugal

Leonor S. Pinto<br>Universidade de Lisboa, Instituto Superior de Economia e Gestão, Lisboa, Portugal CEMAPRE - Centro de Matemática Aplicada à Previsão e Decisão Económica, Lisboa, Portugal


#### Abstract

Money collection presents particular problems in terms of effective vehicle routing. Planning the collection or distribution of money for ATMs or parking meters gives rise to two problems: while the total collecting time should be minimized, tours on successive days should be different to prevent robberies. The combination of these two problems is named as the Dissimilar Routing Problem. When the safes to be collected are located along the streets, it corresponds to an arc routing problem, which we call DARP, and when the money is from ATMs, it corresponds to a vehicle routing problem, usually referred to as the peripatetic routing problem. The former problem arises in a Portuguese company in charge of street parking in Lisbon. The firm needs to define tours to collect safes from parking meters, minimizing the total collecting time. To avoid robberies these tours cannot be repeated or somehow anticipated. For this new problem, we present a mixed integer linear programming (MILP) model and develop a matheuristic. Preliminary experiments are provided with data that mimic the real confidential data. Results point to a good performance of the matheuristic, while the smaller instances can be solved to optimality with the MILP model and a commercial solver. © 2017 Wiley Periodicals, Inc. NETWORKS, Vol. 70(3), 233-245 2017


[^0]Keywords: arc routing; dissimilar arc routing; mixed integer linear programming formulation; flow models; matheuristics; risk constrained cash-in-transit

## 1. INTRODUCTION

This article considers arc routing problems (ARP) were the tours for a specific vehicle must be defined in a weekly based time horizon and, for safety purposes, need to be dissimilar. As in an ARP it starts and finishes its servicing tours in the same point, the depot. The services are associated to the links of the network. The main differences regarding a generic ARP are:

- A weekly time horizon is considered, and the services, as required by the real case in study, are to be performed on a daily basis, which mean that all demand links must be serviced once a day, thus identifying an ARP tour per day;
- Dissimilar tours must be defined so as to prevent robberies, and thus we aim to identify one ARP tour per day which, in turn, is somehow dissimilar from remaining tours.

We consider that tours similarity is related with the position each task is served in its tour, and thus, to measure similarity between tours we divide each day into a fixed number of periods. Each period corresponds to a subsequence of the sequence of tasks in the tour, that is, the first tasks subsequence corresponds to period 1 , the second to period 2 , and so on. A minimum number of tasks is imposed for each period. Two tours are fully dissimilar if no task is served during the same period. Of course, if the number of tasks served during the same period in two tours increases (decreases) the dissimilarity between them decreases (increases).


FIG. 1. Dissimilar tours. \#.P\%-\# represents the order in the tour at period \%; shaded node marks the first period ends. [Color figure can be viewed at wileyonlinelibrary.com]

Next example illustrates this concept of dissimilar tours.
Example 1. Consider a network with six edge tasks, $(1,2)$, $(2,3),(2,4),(2,5),(3,4)$, and $(4,5)$, and two deadhead links $(0,1)$ and $(0,4)$ connecting the depot, node 0 . Two feasible tours for two days, starting and ending at the depot are depicted in Figure 1. Each tour has two periods with three edges each, and numbers in links represent the order of the edge in the tour and the period it belongs to.

Tour $1: \underbrace{\{(0,1,2,3,4)}_{\text {period } 1}, \underbrace{(4,2,5,4,0)\}}_{\text {period } 2}$
Tour $2: \underbrace{\{(0,4,5,2,4)}_{\text {period } 1}, \underbrace{(4,3,2,1,0)\}}_{\text {period } 2}$.
With no links repeated during the same period these tours are then considered dissimilar.

Imposing the identification of a group of tours, one per day, with no similarities between each pair would be too restrictive. Thus, we opt to relax this imposition, and in accordance to our application, we considered less restrictive types of constraints to avoid similarities. Hence, in a MILP, defined on Section 3, we impose that the service of each link in two consecutive days is performed in different periods. In the developed matheuristic, as tours are considered as a whole, we bound the similarity between two tours: (i) in consecutive days; or (ii) in the planning horizon.

This work was motivated by a case study in a Portuguese company in charge of street parking in Lisbon. The firm needs to define tours to daily collect the safes from parking meters that minimize the total collecting time. To avoid robberies these tours should not be repeated or somehow anticipated.

The contribution of this article is fourfold. First, we define and model a new problem, the Dissimilar Arc Routing (DARP), in Section 3. Second, the definition of a dissimilarity measure is proposed for the first time for arc routing cases. Third, in Section 4, we develop a matheuristic, based on two different models here proposed, allowing the identification of good quality feasible solutions. Fourth, we analyze the application of this methodology to random generated instances, inspired by a real case study involving collection and money transportation, in Section 5.

A literature review, next presented, illustrates the few studies on dissimilarity and routing problems, always focusing on the node routing case.

## 2. LITERATURE REVIEW

Dissimilar tours often appear related to cash-in-transit, security patrol tours, evacuation or even the transportation of hazardous materials. These problems require different approaches, justifying different studies in the literature to tackle them. Additionally, and as far as we know, the works considering dissimilar tours only deal with node routing cases. This paper is thus the first application embedding the dissimilarity of the tours within an arc routing environment. Note that in a node routing case the clients must be visited and reached through dissimilar links. In a parallel arc routing case, the links are the ones to be visited, and the dissimilarity must then be defined on the sequence of repeated links, which, in turn, results in a different and harder problem to solve.

Despite being a critical point with regard to safety, only a few papers address the dissimilarity of the tours, and the VRP version is, as usual, the starting point. Talarico et al. [15] defined and studied the $k$-dissimilar VRP ( $k$ d-VRP), where the similarity between two VRP solutions is defined based on the edges shared between them. The aim is to identify $k$ dissimilar VRP solutions, that is, tours, starting and ending at a depot node, visiting all clients once, and within the vehicles' capacity. Talarico et al. [16] defined the Risk-constrained Cash-in-Transit Vehicle Routing Problem (RCTVRP), taking special attention to the risk of being robbed, which they assumed to be proportional both to the amount of cash being carried and to the distance covered by the vehicle carrying the cash. The total risk incurred by a vehicle is, in turn, limited by a risk threshold that can be computed. They also presented metaheuristics and benchmark instances, and further developments on this work gave rise to the recent publication [17].

An identical problem, also endorsing the node routing case, is referred to as the "Peripatetic" Vehicle Routing Problem (PVRP) introduced by Krarup [10] for the multiple salesman case, and named as the m-Peripatetic Salesman Problem (m-PSP). In this problem, no repeated arcs are allowed to visit the clients. Thus, while in the $k \mathrm{~d}-\mathrm{VRP}$, the repetition of arcs is upper limited, the PVRP explicitly forbids repetitions along the planning horizon. These problems also differ on the defined objective. While $k$ d-VRP aims to minimize the worst case travelling cost, the PVRP minimizes the total cost over all periods.

The PVRP may also be applied to network design, as proposed by De Kort [3] whom identifies several edges-disjoint cycles to prevent link failure in a network. De Kort [3]and De Kort [4] proposed lower bounds and exact procedures to solve the problem. De Kort and Volgenant [5] generalized the previous studies to tackle a 2-PSP in which each cycle contains each vertex at most once and a penalty is payed for vertices not included in any cycle.

Later on, Duchenne et al. [6]and Duchenne et al. [7] proposed new exact algorithms for the m-PSP to identify disjoint Hamiltonian cycles of minimum total cost. In [7] vehicles with limited capacity are also considered.

Ngueveu et al. [14] and Ngueveu et al. [13] apply the PVRP to identify patrol tours to security agents, knowing that customers are visited several times within a planning horizon, and no repeated arcs are allowed to reach each client.

Wolfler-Calvo and Cordone [18] studied a security problem where every night, guards must visit all the assigned clients through different tours, amongst other impositions. Alarms can occur, requiring for immediate reaction, that is, for the redesign of tours in a just-in-time way. An ideal time is identified for each client node in such a way that the times are uniformly distributed through the night. To avoid the tours repetition along the time horizon, time windows are defined around these ideal times, imposing minimum and maximum times between two consecutive inspections.

Yan et al. [19] aim to reduce operating costs and ensure safety in a cash-in-transit problem. Authors argue that different tours and schedules to enforce safety make it difficult to formulate. The developed model, a multi-commodity network flow, incorporates a similarity defined from both time and space measures for routing and scheduling purposes. Thus, different visit times of the same customer during the planning horizon and different sequences of visited points (space measures) are imposed. In their application, pickup and delivery services are needed, and thus the amount of money carried by a vehicle along a road is not usually correlated with the number of demand points.

Michallet et al. [12] also deal with cash in transit problems, with the scope to design tours that look "random," and spread over the time horizon. As the probability of being robbed increases at the vehicles stop (e.g., needed to load or unload an ATM) authors forbid the vehicles arriving out of the time windows of the clients to avoid waiting times. The problem is named as the periodic VRP with time spread constraints on services (PVRPTS).

The transportation of hazardous materials also demands for dissimilar tours to prevent disasters as well as to not expose always to the same population. Usually, these transports also consider population densities, to try to avoid the use of paths through highly density population areas, making an acceptable trade-off between geographic diversity and performance. Dadkar et al. [2] and Erkut et al. [8] are examples of this type of studies, focusing on the paths diversification as well as some stochastic characteristics. These problems significantly differ from the one here tackled. In fact, different objectives are defined, as minimizing the risk of accidents
(e.g., avoiding the use of tunnels) being the solutions' characteristics also distinct (e.g., the dissimilarity imposed may depend on the population densities and on some pollution aspects in case of accidents). The tours designed are often used repeatedly during some time, and then new and dissimilar tours are found to repeat again for some time, making the dissimilarity issue simpler to handle.

Emergency situations represent other application for dissimilar paths pursing. However, in such cases the dissimilarity is defined to avoid the use of damage paths, resulting in problems significantly distinct from the one in study. A major concern within an emergency case is related to the uncertainty of road conditions after the disaster (earthquakes, hurricanes, chemical explosions, etc.). Lim and Rhee [11] for instance, developed an algorithm to provide alternative paths with overlaps among them.

To sum up, although being more similar to the $k$ d-VRP, the RCTVRP or even the PVRP the problem in study is significantly different, and, as far as authors' knowledge, it is also a new problem. Its challenge comes from the fact that the similarity is here related with the sequences of links traversed, which in turn are harder to identify if compared with a dissimilarity based on links to reach nodes, as in VRP cases. For the first time, we present a new valid model as well as new models to deal within a matheuristic, which is also a novelty.

## 3. MODEL FOR DISSIMILAR MIXED ARC ROUTING PROBLEMS

The model here developed is a generalization of the flow based model for the mixed capacitated ARP (MCARP) from Gouveia et al. [9]. The problem under study is defined on a mixed graph $\left(N, A^{\prime} \cup E\right)$. Edges in $E$, characterize narrow two-way streets that may be served by only one traversal (zigzag services). Arcs in $A^{\prime}$ represent either one way or large two-way streets that must be served in both directions, in which case the street is modeled with two reverse arcs. The vehicle is parked at a depot node, $0 \in N$, from where it starts/ends and its service is performed by only one crew. The depot is far from the service area, and thus no demand arcs are incident into it, and it cannot be used as an intermediate node as well. Node set $N$ represents the depot, the street crossings or the dead-end streets.

Two types of links in $A^{\prime} \cup E$ are distinguished: demand links or tasks, and deadheading links (i.e., links that can be traversed without need of service). All tasks may also be deadheaded for connectivity purposes.

The time horizon is here defined as the set of days $H=\{1,2,3,4,5\}$. This time horizon is divided into several periods per day.

A vehicle tour is a closed walk starting and ending at the depot and representing the vehicle service in a given day, $h \in H$. In the application, each day all the tasks need to be serviced once. A vehicle service is a combination of its tours, one for each $h \in H$, that are considered dissimilar. We name



FIG. 2. Tours with no fixed periods' size. \#.P\% - \# represents the order in the tour at period \%; shaded node marks the first period ends. [Color figure can be viewed at wileyonlinelibrary.com]

Dissimilar Arc Routing Problem (DARP), the problem that determines a vehicle service of minimum total time.

The similarity of two vehicle tours, related to the routine of the services, is defined as the percentage of tasks that is served in the same period. Remember that periods are artificial and correspond to sub-sequences of the sequence of tasks in the tour, and thus vehicle tours with services in different periods point to dissimilar vehicle tours. The similarity of a vehicle service, also named as the total similarity, is the sum of the similarity of all pairs of vehicle tours it includes.

The following example illustrates the concept of similarity between two tours.

Example 2. Consider a network with two arc tasks, $(2,3)$ and $(3,2)$, six edge tasks, $(1,2),(2,5),(3,4),(3,5),(4,5)$ and $(5,6)$, and two deadhead links $(0,1)$ and $(0,6)$ connecting the depot, node 0 . Two periods were fixed per tour. The two feasible tours for two days, starting and ending at the depot, depicted in Figure 2, have no minimum number of services per period imposed.

$$
\text { Tour } 1: \underbrace{\{(0,1,2,3,4,5,3,2,5)}_{\text {period } 1}, \underbrace{(5,6,0)\}}_{\text {period } 2}
$$

Tour $2: \underbrace{\{(0,6,5)}_{\text {period } 1}, \underbrace{(5,3,4,5,2,3,2,1,0)\}}_{\text {period } 2}$.
Although no tasks are in the same period in both tours, we may see that tasks $(3,4)$ and $(4,5)$ are served in exactly the same position in the sequence (fourth and fifth) which does not agree with the idea of two dissimilar tours. To avoid situations like this we define periods' size, in terms of number of tasks, by imposing a minimum number of $\left\lfloor\frac{n . \text { tasks }}{n \text {. periods }}\right\rfloor=$ $\left\lfloor\frac{8}{2}\right\rfloor=4$ services per period. The same feasible tours are now decomposed into periods in a different way, and thus a similarity between the tours is found. In fact, by forcing the size of periods to a minimum of 4 , being

Tour 1: $\underbrace{\{(0,1,2,3,4,5))}_{\text {period } 1}, \underbrace{(5,3,2,5,6,0)\}}_{\text {period } 2}$
Tour $2: \underbrace{\{(0,6,5,3,4,5)}_{\text {period } 1}, \underbrace{(5,2,3,2,1,0)\}}_{\text {period } 2}$,
tasks $(3,4)$ and $(4,5)$ are served in the same period, as well as tasks $(3,2)$ and $(2,5)$, and a similarity is detected.

We decided to deal with the similarity issue regarding the periods tasks are served. Another possibility would have been measuring the number of common sequences of tasks. However, this approach seems to be much harder to model and solve. In Example 2 we saw that some repetitions of sequences of tasks can be detected by imposing a bound on the size of periods. For instance, the sequence $(3,4),(4,5)$ appears in period 1 in both tour 1 and tour 2. Nevertheless, the same sequence can exist in two dissimilar tours as long as it is in different periods. For instance, Tour3: $\{(0,1,2,3,2,5),(5,3,4,5,6,0)\}$, and Tour2 defined before, are fully dissimilar (i.e., with no services during the same period) but they share the same subsequence of tasks $(3,4),(4,5)$. Observe, however, that they are in a different position in the sequence 3rd and 4th for Tour2 and 6th and 7th for Tour3. Thus, although not looking into the sequences of arcs served, by imposing lower bounds for the number of tasks per period, its size, we relate the similarity of the routes with the order tasks are serviced.

Each link in a tour can be just traversed (deadheaded) or, in the case of a demand link, it can also be served. Each time a link is deadheaded, task or not, a time is taken into account.

In what follows, we present the notation used in order to define and model the mixed DARP here tackled.

- $\mathrm{G}=(N, A)$ is a directed graph, derived from $\left(N, A^{\prime} \cup E\right)$, by replacing each edge in $E$ by two arcs with opposite directions, that is, $A=A^{\prime} \cup\{(i, j),(j, i):(i, j) \in E\}$ with no repetitions.
- $R \subseteq A$ is the set of arcs in $G$ associated with the tasks, and its cardinality is $|R|=\left|A_{R}\right|+2\left|E_{R}\right|$, being $A_{R}$ and $E_{R}$ the set of arc-tasks and edge-tasks, respectively.
- $c_{a}$ is the time needed to serve each task $a=(i, j) \in R$.
- $d_{a}$ is the deadheading time, that is, the time needed to traverse $\operatorname{arc} a=(i, j) \in A$ without serving it.
- $H=\{1,2,3,4,5\}$ is the time horizon that may represent the days in a working week and $h \in H$ is a specific day.
- $L$ is the set of periods and $\ell \in L$ is a specific period.
- $W$ is a large number.

The problem we are modeling basically consists of finding a group of minimum length vehicle tours that are dissimilar in consecutive days. The major differences here included, when compared with the MCARP model of Gouveia et al. [9] are the following:

1. Variables are defined with an extra index to identify the periods $(\ell)$, as the days $(h)$ represent different tours and thus are related with the multiple tours in the MCARP model;
2. New variables and constraints are needed to define different start and ending points per tour to identify the periods;
3. Usual balance and flow constraints at each node must be carefully written as they may be related to a node that will be selected as the starting or the ending node of a period;
4. A minimum number of services per period is imposed to control the similarity measure;
5. New constraints are added to prevent the similarity of the tours.

### 3.1. Flow Based Model

For each day $h \in H$ and each period $\ell \in L$, we define:

- $x_{i j}^{\ell h}=\left\{\begin{array}{l}1 \text { if }(i, j) \in R \text { is served during period } \ell \text { in day } \mathrm{h} \\ 0 \text { otherwise }\end{array}\right.$
- $u_{i}^{\ell h}=\left\{\begin{array}{c}1 \text { if } i \text { is the ending point of the tour in } \\ \text { period } \ell \text { in day } \mathrm{h} \\ 0 \text { otherwise }\end{array}\right.$
- $v_{i}^{\ell h}=\left\{\begin{array}{c}1 \text { if } i \text { is the starting point of the tour in } \\ \text { period } \ell \text { in day } \mathrm{h} \\ 0 \text { otherwise }\end{array}\right.$
- $y_{i j}^{\ell h}$ is the number of times that arc $(i, j) \in A$ is deadheaded during period $\ell$ in day $h$.
- $f_{i j}^{\ell h}$ is the flow traversing arc $(i, j) \in A$ during period $\ell$ in day $h$. It is related to the number of remaining services in the tour, or in a subtour of it.

The problem to identify a vehicle service in $H$, minimizing the total routing time, is next detailed.
(M1DAR)

$$
\begin{equation*}
\min Z=\sum_{h \in H} \sum_{\ell \in L}\left(\sum_{a \in A} d_{a} y_{a}^{\ell h}+\sum_{a \in R} c_{a} x_{a}^{\ell h}\right) \tag{1}
\end{equation*}
$$

$\sum_{j:(i, j) \in A} y_{i j}^{\ell h}+\sum_{j:(i, j) \in R} x_{i j}^{\ell h}-\sum_{j:(j, i) \in A} y_{j i}^{\ell h}-\sum_{j:(j, i) \in R} x_{j i}^{\ell h}$

$$
=v_{i}^{\ell h}-u_{i}^{\ell h} \quad i \in N \backslash\{0\} ; h \in H ; \ell \in L
$$

$$
\sum_{\ell \in L}\left(\sum_{j:(i, j) \in A} y_{i j}^{\ell h}+\sum_{j:(i, j) \in R} x_{i j}^{\ell h}-\sum_{j:(j, i) \in A} y_{j i}^{\ell h}-\sum_{j:(j, i) \in R} x_{j i}^{\ell h}\right)
$$

$$
\begin{equation*}
=0 \tag{3}
\end{equation*}
$$

$$
i \in N \backslash\{0\} ; h \in H
$$

$\sum_{h \in H}\left(\sum_{\ell \in L \backslash\{1\}} \sum_{j:(0, j) \in A} y_{0 j}^{\ell h}\right)=0$
$\sum_{j:(0, j) \in A} y_{0 j}^{1 h}=1$
$h \in H$

$$
\begin{equation*}
\sum_{h \in H}\left(\sum_{\ell \in L \backslash\{|L|\}} \sum_{j:(j, 0) \in A} y_{j 0}^{\ell h}\right)=0 \tag{6}
\end{equation*}
$$

$$
\sum_{j:(j, 0) \in A} y_{j 0}^{|L| h}=1
$$

$$
h \in H
$$

$$
\sum_{\ell \in L} x_{i j}^{\ell h}=1
$$

$$
a=(i, j) \in A_{R} ; h \in H
$$

$$
\sum_{\ell \in L}\left(x_{i j}^{\ell h}+x_{j i}^{\ell h}\right)=1 \quad a=(i, j) \in E_{R} ; h \in H
$$

$$
\sum_{a \in R} x_{a}^{\ell h} \geq\left\lfloor\frac{\left|A_{R}\right|+\left|E_{R}\right|}{|L|}\right\rfloor \quad \ell \in L ; h \in H
$$

$$
\sum_{j:(j, i) \in A} f_{j i}^{1 h}-\sum_{j:(i, j) \in A} f_{i j}^{1 h}=\sum_{j:(j, i) \in R} x_{j i}^{1 h}
$$

$$
i \in N \backslash\{0\} ; h \in H
$$

$$
\begin{aligned}
& \quad \sum_{j:(j, i) \in A} f_{j i}^{\ell h}-\sum_{j:(i, j) \in A} f_{i j}^{\ell h} \leq \sum_{j:(j, i) \in R} x_{j i}^{\ell h}+W v_{i}^{\ell h} \\
& \quad-\sum_{j:(j, i) \in A} f_{j i}^{\ell h}+\sum_{j:(i, j) \in A} f_{i j}^{\ell h} \leq-\sum_{j:(j, i) \in R} x_{j i}^{\ell h}+W v_{i}^{\ell h} \\
& \quad i \in N \backslash\{0\} ; \ell \in L \backslash\{1\} ; h \in H \\
& \sum_{j:(0, j) \in A} f_{0 j}^{1 h}=\sum_{a \in R} x_{a}^{1 h}
\end{aligned}
$$

$$
\sum_{j:(i, j) \in A} f_{i j}^{\ell h} \leq \sum_{a \in R} x_{a}^{\ell h}+W\left(1-v_{i}^{\ell h}\right)
$$

$$
i \in N \backslash\{0\} ; h \in H ; \ell \in L \backslash\{1\}
$$

$$
-\sum_{j:(i, j) \in A} f_{i j}^{\ell h} \leq-\sum_{a \in R} x_{a}^{\ell h}+W\left(1-v_{i}^{\ell h}\right)
$$

$$
i \in N \backslash\{0\} ; h \in H ; \ell \in L \backslash\{1\}
$$

$$
x_{a}^{\ell h} \leq f_{a}^{\ell h} \leq W\left(y_{a}^{\ell h}+x_{a}^{\ell h}\right)
$$

$$
\begin{equation*}
a \in R ; h \in H ; \quad \ell \in L \tag{17}
\end{equation*}
$$

$$
\begin{equation*}
f_{a}^{\ell h} \leq W \sum_{\ell \in L} y_{a}^{\ell h} \quad a \in A \backslash R ; h \in H \tag{18}
\end{equation*}
$$

$$
u_{i}^{\ell h} \leq \sum_{j:(j, i) \in A} y_{j i}^{\ell h}+\sum_{j:(j, i) \in R} x_{j i}^{\ell h}
$$

$$
\begin{equation*}
i \in N \backslash\{0\} ; h \in H ; \ell \in L \tag{19}
\end{equation*}
$$

$$
v_{i}^{\ell h} \leq \sum_{j:(i, j) \in A} y_{i j}^{\ell h}+\sum_{j:(i, j) \in R} x_{i j}^{\ell h}
$$

$$
\begin{equation*}
i \in N \backslash\{0\} ; h \in H ; \ell \in L \tag{20}
\end{equation*}
$$

$$
\begin{equation*}
u_{i}^{\ell h}=v_{i}^{\ell+1 h} \quad i \in N \backslash\{0\} ; h \in H ; \ell \in L \backslash\{|L|\} \tag{21}
\end{equation*}
$$

$$
\begin{align*}
& \sum_{i \in N \backslash\{0\}} u_{i}^{\ell h}=1 \quad h \in H ; \ell \in L \backslash\{|L|\}  \tag{22}\\
& \sum_{i \in N \backslash\{0\}} v_{i}^{\ell h}=1 \quad h \in H ; \ell \in L \backslash\{1\}  \tag{23}\\
& x_{i j}^{\ell h}+x_{i j}^{\ell h+1} \leq 1 \\
& \quad(i, j) \in A_{R} ; h \in H \backslash\{|H|\} ; \ell \in L \\
& x_{i j}^{\ell h}+x_{j i}^{\ell h}+x_{i j}^{\ell h+1}+x_{j i}^{\ell h+1} \leq 1  \tag{24}\\
& \quad(i, j) \in E_{R} ; h \in H \backslash\{|H|\} ; \ell \in L \\
& x_{i j}^{\ell h} \in\{0,1\}  \tag{25}\\
& f_{i j}^{\ell h} \geq 0 \quad(i, j) \in R ; h \in H ; \ell \in L  \tag{26}\\
& y_{i j}^{\ell h} \geq 0, \text { integer } \quad(i, j) \in A ; h \in H ; \ell \in L \\
& u_{i j}^{\ell h} ; v_{i j}^{\ell h} \in\{0,1\} \quad(i, j) \in A ; h \in H ; \ell \in L \tag{27}
\end{align*}
$$

Conditions (2) and (3) imply the continuity of the vehicle tours at each node, considering three different types of nodes: starting, ending or intermediate; (4)-(7) fix the depot as the starting point of the first period $\ell=1$ ((4) and (5)), and ending point of the last period $\ell=|L|((6)$ and (7)), each day; (8) and (9) guarantee that each task is serviced once a day and in only one period; (10) imposes a minimum number of services per period, used to define the size of the periods to impose the similarities; (11) and (14) are flow conservation constraints for the first period, while (12), (13), (15), and (16) represent these constraints for the remaining periods; these, together with the linking constraints (17) and (18) force the connectivity of the vehicle tours. Note that (11) are typical generalized flow conservation constraints at each node $i$, guaranteeing that if an $\operatorname{arc}(j, i)$ is served in the first period, then a unit of flow is absorbed at node $i$. Constraints (12) and (13) are similar for the remaining periods $(2, \ldots,|L|)$ and for all nodes that do not represent its starting node. Constraints (14) compute the flow leaving the depot as the number of tasks served during the first period, while (15) and (16) compute similarly the flow leaving the starting point of each remaining period. Constraints (19) and (20) ensure that the vehicle may use a node as an ending or starting point of a period only if it is traversed by the vehicle during the same period; (21) relates the ending of a period with the beginning of the next period, each day, while (22) and (23) guarantee that only one node may be used as a starting/ending point, per period and per day. Constraints (24) and (25) are used to impose vehicle services dissimilarity. Variable domains are settled in (26)-(29). There is no need to define flow variables as integer as their purpose is only to ensure tours connectivity. Left hand side of constraints (17), although not needed are imposed to speed up the solver (CPLEX), as preliminary results show their effectiveness.

Although giving rise to an undesirable increase in the number of constraints, we note that (24) and (25) may be generalized to consider more than two consecutive days. Without this generalization, we may get solutions where day $h+2$ is
a replica of day $h$, and so on, which may represent a model handicap.

Instead, to bound repetitions in the same period all over the time horizon, we may consider the alternative set of constraints:

$$
\begin{cases}\sum_{h \in H} x_{i j}^{\ell h} \leq M & (i, j) \in A_{R}, \ell \in L  \tag{30}\\ \sum_{h \in H}\left(x_{i j}^{\ell h}+x_{j i}^{\ell h}\right) \leq M & (i, j) \in E_{R}, \ell \in L\end{cases}
$$

where $M \geq 1$, and $M=1$ if no repetitions are allowed.
As referred to above, this would be too restrictive. We thus opt to consider the simpler version, that is, including only the constraints that avoid similar tours on two consecutive days. More general situations are elaborated through a matheuristic we developed and next detail.

## 4. MATHEURISTIC

Matheuristic approaches, linking modeling with heuristics have been increasingly suggested to complex problems. In fact, new hardware technology allows the resolution of more complicated problems and thus integer programming models may be used to help to find solutions for bigger instances. Archetti and Speranza [1] recently published a survey on matheuristics for routing problems.

Leaving aside, for now, the similarity issue, the proposed matheuristic starts by generating a pool of feasible tours. Model (M1DAR) is thus applied considering only one day ( $h=1$ ) as well as different objective functions. With the pool of feasible solutions two models were developed to generate different feasible solutions, regarding the similarity issue. These models aim to select $|H|$ tours (the number of days) that may be considered dissimilar so they can be used in a real case. The matheuristic is next detailed.

### 4.1. Matheuristic

1. Use model (M1DAR) with $h=1$, thus without constraints (24) and (25), to identify several feasible tours. Solve (M1DAR) with $h=1$ and a time limit of $3 h$; Add to the pool of feasible tours, $F T$, all feasible solutions provided by CPLEX;

## Repeat

i. In $F T$, find a task that was never serviced in a period and add a constraint to fix the service of that task on that period,
ii. run once more model (M1DAR) with $h=1$, within a 3 h cpu time limit;
iii. Add to the pool of feasible tours, $F T$, all the feasible solutions provided by CPLEX, if any;
Until (all tasks are tried to be serviced in every periods); iv. Use different objective functions, as for example, the minimization of the number deadheading traversals and repeat the above procedure;
2. Apply one of the models next defined to identify $|H|$ tours, one per day, using:



FIG. 3. Similarity of two tours. P\#.D \% represents the period \# in day \%, and the shaded node the first period ends. [Color figure can be viewed at wileyonlinelibrary.com]

MRH $\mu$-to minimize the total time to collect safes not repeating a fixed percentage of tasks in each pair of tours in two successive days
or
MR $\mu$-to minimize the total routing time to collect the safes, within a given maximum similarity between any pair of tours.

The models referred to in step 2 are next defined.
Let: $F T$ be a group of vehicle tours; $C_{r}$ the total routing time of tour $r \in F T ; S_{r t}$ the similarity between tours $r \in F T$ and $t \in F T$; and $\mu$ the maximum similarity allowed.

The similarity index is computed as :

$$
S_{r t}=\frac{\begin{array}{c}
\text { number of tasks served during the } \\
\text { same period in tours } r \text { and } t \tag{31}
\end{array}}{\text { total number of tasks }}
$$

Example 3. Let us consider the network with nine edge tasks, $(1,2),(2,3),(2,4),(2,5),(3,4),(4,5),(4,6),(5,6)$, and $(5,7)$, and two deadhead links connecting the depot, node 0 , depicted in Figure 3, and two feasible tours for two consecutive days, starting and ending at the depot:

$$
\text { Day } 1: \underbrace{\{(0,1, \overline{2,3,4}, \overline{2,5})}_{\text {period } 1}, \underbrace{(\overline{5,4}, \overline{6,5}, 7,0)\}}_{\text {period } 2}
$$

Day $2: \underbrace{\{(0,7, \overline{5,2,3,4}, 6)}_{\text {period } 1}, \underbrace{(\overline{6,5,4}, 2,1,0)\}}_{\text {period } 2}$.
These tours repeat the periods for servicing tasks $(2,3)$, $(3,4),(2,5)$, in period 1 , and tasks $(5,4)$ and $(6,5)$ in period 2. So, the similarity index is $\frac{5}{9}$.

### 4.2. Model MRH $\mu$

The variables are:

- $g_{r}^{h}=\left\{\begin{array}{l}1 \text { if } r \in F T \text { is selected for day } h \in H \\ 0 \text { otherwise }\end{array}\right.$
and the model to identify the best vehicle service, that is, the best tours per time horizon minimizing the total routing
time is:

$$
\begin{align*}
& \mathrm{MRH} \mu \\
& \quad \min \sum_{h \in H} \sum_{r \in F T} C_{r} g_{r}^{h} \tag{32}
\end{align*}
$$

$$
\left\{\begin{array}{lr}
S_{r t}\left(g_{r}^{h}+g_{t}^{h+1}-1\right) \leq \mu r, t \in F T, h \in H \backslash\{|H|\}  \tag{33}\\
\sum_{r \in F T} g_{r}^{h}=1 & h \in H \\
g_{r}^{h} \in\{0,1\} & r \in F T, h \in H
\end{array}\right.
$$

Within the minimization of the total routing time objective (32), the aim is to choose one tour per day (34), not allowing tours on two successive days with a similarity index greater than $\mu$ (33).

Note that if $\mu=0$ the solutions provided with this model can fairly be compared with the ones generated by (M1DAR). In fact, both models avoid similar tours not allowing tasks services during the same period in two consecutive days. Thus, if (M1DAR) optimal tours are in the pool $F T$, the optimal values for both models, (MRH0) and (M1DAR), coincide.

Computational tests are also performed for $\mu=0.1$ and for $\mu=0.3$, being so less restrictive regarding the dissimilarity.

The total similarity, TS, of the tours generated with a model can be computed if (31) is applied to all the pairs of the chosen tours. Thus, and considering a feasible solution of (MRH $\mu$ ), the total similarity is:

$$
\begin{equation*}
T S=\sum_{h=1}^{|H|-1} \sum_{k=h+1}^{|H|} \sum_{r, t \in F T} S_{r t} g_{r}^{h} g_{t}^{k} \tag{36}
\end{equation*}
$$

Next proposition is used to establish the upper bound limits for $T S$ in the above model.

Proposition 1. Any feasible solution of (MRH $\mu$ ) has a total similarity bounded by:

$$
\begin{equation*}
T S \leq(|H|-1)\left(\frac{|H|}{2}+\mu-1\right) \tag{37}
\end{equation*}
$$

Proof. If all tours selected are not constrained on its similarity then the maximum value of $T S$ would be $\binom{|H|}{2}=$ $\frac{|H|!}{2!(|H|-2)!}=\frac{|H|(|H|-1)}{2}$. However, as the similarity for each pair of the $|H|-1$ consecutive tours is bounded by $\mu$, the total similarity is not greater than $\frac{|H|(|H|-1)}{2}-(|H|-1)(1-\mu)$, and (37) follows.

Corollary. Any feasible solution of (M1DAR) has a maximum total similarity given by:

$$
\begin{equation*}
T S \leq(|H|-1)\left(\frac{|H|}{2}-1\right) \tag{38}
\end{equation*}
$$

Proof. Observe that (38) is (37) with $\mu=0$.

### 4.3. Model $M R \mu$

Let now simplify the model, and define the variables without the identification of the days, as:

- $g_{r}=\left\{\begin{array}{l}1 \text { if } r \in F T \text { is selected } \\ 0 \text { otherwise }\end{array}\right.$
and the model to generate the best vehicle service, that is, the best tours minimizing the total routing time is:

$$
\begin{gather*}
(\mathrm{MR} \mu) \\
\min \sum_{r \in F R} C_{r} g_{r}  \tag{39}\\
\begin{cases}S_{r t}\left(g_{r}+g_{t}-1\right) \leq \mu & \forall r, t \in F T \\
\sum_{r \in F T} g_{r}=|H| & \\
g_{r} \in\{0,1\} & \forall r \in F T\end{cases} \tag{40}
\end{gather*}
$$

Within the minimization of the total routing time objective (39), the aim is to choose as many tours as the number of days (41), with a predefined upper bound $(\mu)$ on the similarity between any pair of chosen tours (40).

Note that, in this model, $\mu$ indicates the maximum percentage of tasks that can be served in the same period for any pair of tours in a vehicle service, and not only for two consecutive days, as in model (MRH $\mu$ ).

Observe that tours in Example 3 are incompatible for this problem if $\mu=0.3$, as in nine tasks, no more than three services can repeat the period, and we have five repetitions.

The total similarity, $T S$, of a feasible solution of ( $\mathrm{MR} \mu$ ) is:

$$
\begin{equation*}
T S=\sum_{r, t \in F T} S_{r t} g_{r} g_{t} \tag{43}
\end{equation*}
$$

Proposition 2. The total similarity of any feasible solution of $(M R \mu)$ is bounded by: $T S \leq \mu\binom{|H|}{2}$, where $\binom{|H|}{2}$ stands for combinations of 2 drawn from $|H|$.

TABLE 1. Characteristics of the instances.

| Name | $\|\boldsymbol{V}\|$ | $\|\boldsymbol{A}\|$ | $\left\|\boldsymbol{A}_{\boldsymbol{R}}\right\|$ | $\|\boldsymbol{E}\|$ | $\left\|\boldsymbol{E}_{\boldsymbol{R}}\right\|$ |
| :--- | :---: | :---: | :---: | :---: | :---: |
| ex1 | 11 | 33 | 11 | 1 | 1 |
| ex2 | 16 | 34 | 14 | 2 | 2 |
| ex3 | 19 | 40 | 18 | 2 | 2 |
| ex4 | 21 | 47 | 19 | 3 | 3 |
| ex5 | 21 | 63 | 21 | 2 | 2 |
| ex6 | 24 | 63 | 25 | 2 | 2 |
| ex7 | 25 | 54 | 24 | 10 | 10 |
| ex8 | 35 | 80 | 35 | 16 | 16 |
| ex9 | 40 | 65 | 45 | 12 | 12 |
| ex10 | 34 | 77 | 29 | 18 | 18 |
| ex11 | 45 | 98 | 42 | 32 | 32 |
| ex12 | 50 | 111 | 66 | 18 | 18 |

TABLE 2. Feasible tours generation.

|  | Matheuristic - step 1 |  |  | Matheuristic - step 1 |  |
| :--- | :---: | :---: | :---: | :---: | :---: |
| Instance | $\|\mathbf{F T}\|$ | tcpu (s) | Instance |  | $\|\mathbf{F T}\|$ |
| tcpu (h) |  |  |  |  |  |
| ex1 | 125 | 70.7 |  |  |  |
| ex2 | 148 | 115.7 |  |  |  |
| ex3 | 278 | 192.8 |  |  |  |
| ex4 | 238 | 265.3 |  |  |  |
| ex5 | 293 | 340.8 |  |  |  |
| ex6 | 844 | 2388.3 |  |  |  |
| ex7 | 779 | 1989.3 |  |  |  |
| ex8 | 523 | 1560.4 |  | ex8_6 | 1000 |
| ex9 | 1178 | 6527.3 |  | ex9_6 | 1000 |
| ex10 | 1234 | 9394.1 |  | ex10_6 | 1000 |
| ex11 | 1658 | 19980.1 |  | ex11_6 | 1000 |
| ex12 | 1277 | 13212.2 |  | ex12_6 | 1000 |

Proof. This result follows by (43), by noting that $S_{r t} \leq$ 1 , the number of pairs $r, t \in F T(r \neq t)$ such that $g_{r}=g_{t}=$ 1 is $\binom{|H|}{2}$, and each pair similarity is bounded by $\mu$.

To give us an idea about the values for $\mu$ parameter in (MR $\mu$ ) that allow the identification of feasible solutions we solved a model aiming to minimize the maximum similarity, defined as variable SMax. Such model is next detailed.

\[

\]

The maximum similarity is defined in (45) through the tours assigned and as a positive variable (48). Constraints (46) ensure that are assigned as many tours as the number of days.

TABLE 3. Computational results-(M1DAR) versus matheuristic with (MRH0).

| Instance | LB | (M1DAR) |  |  | Gap0 (\%) | gapUB0 (\%) | Matheuristic \& (MRH0) |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  | UB | Gap (\%) | tcpu (s) |  |  | Step 2 tcpu (s) | tcpu (s) |
| $\|\boldsymbol{H}\|=3$ |  |  |  |  |  |  |  |  |
| ex1 | 519 | 519 | 0.00 | 24.6 | 0.96 | 0.96 | 1.2 | 72.0 |
| ex2 | 2200 | 2200 | 0.00 | 45.7 | 0.00 | 0.00 | 2.1 | 117.8 |
| ex3 | 755 | 755 | 0.00 | 98.1 | 0.00 | 0.00 | 9.8 | 202.6 |
| ex4 | 855 | 855 | 0.00 | 8.8 | 0.00 | 0.00 | 5.1 | 270.3 |
| ex5 | 1165 | 1165 | 0.00 | 235.7 | 0.00 | 0.00 | 8.6 | 349.4 |
| ex6 | 1110 | 1110 | 0.00 | 1055.0 | 0.00 | 0.00 | 82.8 | 2471.1 |
| ex7 | 13700 | 13700 | 0.00 | 632.5 | 0.00 | 0.00 | 58.0 | 2047.3 |
| ex8 | 23148 | 23148 | 0.00 | 4005.1 | 0.16 | 0.16 | 3.6 | 1564.0 |
| ex9 | 24935 | 24935 | 0.00 | 1855.3 | 0.00 | 0.00 | 125.9 | 6653.1 |
| ex10 | 2094 | 2109 | 0.72 | 10800.0 | 2.96 | 2.23 | 177.8 | 9571.9 |
| ex11 | 31395 | 31590 | 0.62 | 10800.0 | 0.78 | 0.16 | 275.5 | 20255.6 |
| ex12 | 38052 | 38602 | 1.45 | 10800.0 | - | - | 435.7 | 13647.9 |
| ex8_6 | 23148 | 25015 | 8.07 | 36000.0 | 0.00 | -7.46 | 82.0 | 24007.5 |
| ex9_6 | 24935 | 24935 | 0.00 | 13182.2 | 0.00 | 0.00 | 200.0 | 27546.2 |
| ex10_6 | 2094 | 2239 | 6.92 | 36000.0 | 0.72 | -5.81 | 87.4 | 36439.8 |
| ex11_6 | 31395 | 40778 | 29.89 | 36000.0 | 0.00 | -23.01 | 88.3 | 123067.3 |
| ex12_6 | 38052 | 43127 | 13.34 | 36000.0 | 0.00 | -11.77 | 87.1 | 63974.0 |


| ex1 | 704 | 704 | 0.00 | 158.4 | 0.00 | 0.00 | 2.0 | 72.8 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| ex2 | 2940 | 2940 | 0.00 | 189.0 | 0.00 | 0.00 | 3.5 | 119.2 |
| ex3 | 1010 | 1010 | 0.00 | 285.0 | 0.00 | 0.00 | 14.1 | 206.9 |
| ex4 | 1140 | 1140 | 0.00 | 10.9 | 0.00 | 0.00 | 8.6 | 273.8 |
| ex5 | 1556 | 1556 | 0.00 | 841.8 | 0.00 | 0.00 | 11.7 | 352.5 |
| ex6 | 1475 | 1490 | 1.02 | 10800.0 | 1.02 | 0.00 | 105.5 | 2493.7 |
| ex7 | 18310 | 18310 | 0.00 | 1256.2 | 0.00 | 0.00 | 108.2 | 2097.5 |
| ex8 | 30864 | 30864 | 0.00 | 6124.9 | 0.24 | 0.24 | 6.0 | 1566.4 |
| ex9 | 33290 | 33290 | 0.00 | 7091.3 | 0.00 | 0.00 | 196.2 | 6723.5 |
| ex10 | 2792 | 2822 | 1.07 | 10800.0 | 4.44 | 3.33 | 354.8 | 9748.9 |
| ex11 | 41860 | 51357 | 22.69 | 10800.0 | 0.81 | -17.83 | 630.0 | 20610.1 |
| ex12 | 50736 | 51526 | 1.56 | 10800.0 | - | - | 535.1 | 13747.3 |
| ex8_6 | 30864 | 31982 | 3.62 | 36000.0 | 0.00 | -3.50 | 123.9 | 24049.4 |
| ex9_6 | 33160 | 34145 | 2.97 | 36000.0 | 0.39 | -2.50 | 483.7 | 27829.9 |
| ex10_6 | 2792 | 3086 | 10.53 | 36000.0 | 1.07 | -8.55 | 130.6 | 36483.0 |
| ex11_6 | 41860 | 58273 | 39.21 | 36000.0 | 0.00 | -28.17 | 120.4 | 123099.4 |
| ex12_6 | 50702.2 | 70672 | 39.39 | 36000.0 | 0.07 | -28.21 | 173.4 | 64060.3 |


| $\|H\|=5$ |  |  |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| ex1 | 871 | 871 | 0.00 | 445.8 | 0,57 | 0.57 | 3.0 | 73.7 |
| ex2 | 3670 | 3670 | 0.00 | 406.4 | 0,00 | 0.00 | 5.6 | 121.3 |
| ex3 | 1260 | 1260 | 0.00 | 538.5 | 0,00 | 0.00 | 20.8 | 213.6 |
| ex4 | 1425 | 1425 | 0.00 | 258.2 | 0,00 | 0.00 | 13.7 | 279.0 |
| ex5 | 1943 | 1943 | 0.00 | 6337.1 | 0,00 | 0.00 | 16.1 | 357.0 |
| ex6 | 1855 | 1855 | 0.00 | 4639.7 | 0,00 | 0.00 | 155.2 | 2543.5 |
| ex7 | 22855 | 22855 | 0.00 | 2820.6 | 0,00 | 0.00 | 165.9 | 2155.3 |
| ex8 | 38580 | 38580 | 0.00 | 10800.0 | 0,19 | 0.19 | 7.5 | 1567.8 |
| ex9 | 41481.5 | 41695 | 0.51 | 10800.0 | 0,24 | -0.28 | 288.1 | 6815.4 |
| ex10 | 3490 | 3889 | 11.43 | 10800.0 | 3,55 | -7.07 | 620.6 | 10014.7 |
| ex11 | 52325 | 64052 | 22.41 | 10800.0 | 0,79 | -17.66 | 867.1 | 20847.2 |
| ex12 | 63418.3 | 75215 | 18.60 | 10800.0 | - | - | 526.1 | 13738.3 |
| ex8_6 | 38580 | 45517 | 17.98 | 36000.0 | 0.00 | -15.24 | 172.0 | 24097.5 |
| ex9_6 | 41515 | 42390 | 2.11 | 36000.0 | 0.16 | -1.91 | 551.4 | 27897.6 |
| ex10_6 | 3490 | 5365 | 53.72 | 36000.0 | 0.86 | -34.39 | 186.8 | 36539.2 |
| ex11_6 | 52325 | 71747 | 37.12 | 36000.0 | 0.00 | -27.07 | 179.0 | 123157.9 |
| ex12_6 | 63420 | 77810 | 22.69 | 36000.0 | 0.00 | -18.49 | 243.6 | 64130.5 |

TABLE 4. Upper bounds and average values on TS values.

| $\mid \boldsymbol{H \|}$ | TS upper bound |  |  |  | Average TS values |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | (MRH $\mu$ ) |  |  | $\begin{aligned} & (\mathrm{MR} \mu) \\ & \mu \_0.3 \end{aligned}$ | (MRH $\mu$ ) |  |  | $\begin{gathered} (\mathrm{MR} \mu) \\ \mu \_0.3 \end{gathered}$ |
|  | $\mu \_0$ | $\mu \_0.1$ | $\mu \_0.3$ |  | $\mu \_0$ | $\mu \_0.1$ | $\mu \_0.3$ |  |
| $\|\boldsymbol{H}\|=3$ | 1 | 1.2 | 1.6 | 0.9 | 0.9 | 1.0 | 1.3 | 0.6 |
| $\|\boldsymbol{H}\|=4$ | 3 | 3.3 | 3.9 | 1.8 | 1.9 | 2.0 | 2.7 | 1.3 |
| $\|\boldsymbol{H}\|=5$ | 6 | 6.4 | 7.2 | 3.0 | 3.8 | 4.1 | 4.7 | 2.2 |

## 5. COMPUTATIONAL RESULTS

The proposed models are evaluated over some newly generated instances, as the problems are also new. The computational results were obtained using CPLEX 12.6.0.0, with default settings, in a computer with 2 AMD Opteron 6172 processors ( 24 cores) at 2.1 GHz and with 64 GB RAM. A time limit was established, each time the CPLEX was used. When an integer program is being solved and the time limit is reached before an optimal solution is proved to be found, CPLEX provides the best bounds that are computed taking into account all the live nodes of the branch-and-cut tree. Such bounds are used to evaluate the procedures.

### 5.1. Data Instances

Twelve instances (ex1 to ex12) were generated to assess the performance of the models, with dimensions varying between 11 to 50 nodes and 34 to 129 links. The graphs are based on real street networks, while deadheading and service times were randomly generated. We emphasize that bigger instances, ex8-ex12, present dimensions that can be considered similar to real data, including one collecting vehicle. The relevant characteristics of these instances are depicted in Table 1. Different number of days are also considered, namely, $|H|=3,4,5$. The number of periods was fixed to 3 , and the larger 5 instances were also tried with 6 periods, and named by ex\#_6, thus a total of 51 instances were tested.

### 5.2. Results

In step 1 of the matheuristic, and for the 3 period instances the number of feasible tours $(|F T|)$ varies between 125 and about 1,660 , with computational times (tcpu) varying from 71 s to less than 5.5 h (see Table 2). For the 6 period instances $|F T|$ was limited to 1,000 , and the cpu time varies between 6.5 and 34 h .

Table 3 allows the comparison between the valid model (M1DAR) and the matheuristic using model (MRH $\mu$ ) with $\mu=0$, named as (MRH0). As referred to, the similarity is treated in the same way in both models. So, this is a fair comparison.

Second to fifth columns in Table 3 display the results for the valid model (M1DAR) namely, the lower (LB) and upper (UB) total routing times bounds provided by CPLEX, percentage gap values computed by gap $=\frac{\mathrm{UB}-\mathrm{LB}}{\mathrm{LB}} \times 100 \%$,
and the computational time (tcpu) in seconds. Values obtained with the matheuristic using model ( $\mathrm{MRH} \mu$ ), with $\mu=0$, are in columns six to nine. Column six presents the percentage gap values between the matheuristic upper bound (UB0) and LB, that is gap $0=\frac{\mathrm{UB} 0-\mathrm{LB}}{\mathrm{LB}} \times 100 \%$. Colum seven, headed as gapUB0, depicts percentage gap values comparing upper bounds obtained by both models, that is, gapUB0 $=\frac{\mathrm{UB} 0-\mathrm{UB}}{\mathrm{UB}} \times 100 \%$. Therefore, positive values represent instances for which (M1DAR) provides better bounds, while negative values point to a better performance of (MRH0). Note that the optimal value of (M1DAR) can never be greater than the optimum of (MRH0), and if the optimal tours selected by (M1DAR) are in the $F T$ pool the two values must coincide. However, when (M1DAR) does not reach the optimal solution in its time limit (10h for instances with $|L|=6$, and 3 h for the remaining), (MRH0) may then get a smaller value. Columns eight and nine present, respectively, cpu times (in seconds) regarding the matheuristic (second step only) and the total (both steps). Lower and upper bound bold values mark optimums; gaps less or equal than zero are also signed in bold.

Most of the time (M1DAR) succeeds in generating an optimal solution for the smaller instances. Note that, for higher time horizon values its performance tends to decrease ( $|H|=3: 10$ optimums; $|H|=4: 8$ optimums; $|H|=5$ : 8 optimums, out of 17). From column six, we may conclude that in 29 out of 51 instances (MRH0) ends up with an optimal solution. The biggest gap found was $4.4 \%$, and (MRH0) got eighteen better solutions than (M1DAR). Moreover, this occurs for the bigger instances. However, (MRH0) fails to find feasible solutions for three instances (ex12 with $|H|=3,4,5)$ suggesting that the pool of feasible solutions needs further improvement. Matheuristic cpu times, including step 1 (column 9), tend to overcome (M1DAR) for bigger time horizons (column 5). Note that these include step 1 cpu times which is also used to achieve other solutions through step 2 in small cpu times (see Tables 5 and 6).

Henceforward, and for simplicity, model (MRH $\mu$ ) for $\mu=0 ; 0.1 ; 0.3$ is referred to as (MRH0), (MRH0.1) and (MRH0.3), respectively. Correspondent columns in the tables are headed by $\mu \_0, \mu \_0.1$ and $\mu \_0.3$. In relation to model (MR $\mu$ ), results are presented for only $\mu=0.3$, named as (MR0.3), with columns headed by $\mu_{-} 0.3$. This value was provided by (MRS) model. In fact, for smaller values no vehicle services can be found from the pool for too many instances.

TABLE 5. Total routing times comparing (MRH $\mu$ ) and (MR $\mu$ ).

|  |  |  |  |  |  |  |
| :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

TABLE 6. Execution times comparing (MRH $\mu$ ), (MR $\mu$ ) and (MRS), tcpu (s).

|  | $\|L\|=3$ | (MRH $\mu$ ) |  |  | $\frac{(\mathrm{MR} \mu)}{\mu_{-} 0.3}$ | $\|L\|=6$ | (MRH $\mu$ ) |  |  | $\frac{(\operatorname{MR} \mu)}{\mu_{-} 0.3}$ |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  | $\mu \_0$ | $\mu \_0.1$ | $\mu \_0.3$ |  |  | $\mu \_0$ | $\mu \_0.1$ | $\mu \_0.3$ |  |
|  |  | $\|H\|=3$ |  |  |  |  | $\|H\|=3$ |  |  |  |
| Min |  | 1.2 | 0.9 | 1.5 | 0.1 |  | 82.0 | 294.8 | 10.1 | 0.6 |
| Average |  | 98.9 | 146.8 | 178.9 | 26.1 |  | 109.0 | 320.3 | 19.8 | 2.5 |
| Max |  | 435.7 | 524.9 | 689.3 | 108.5 |  | 200.0 | 393.0 | 34.6 | 7.2 |
|  |  | $\|H\|=4$ |  |  |  |  | $\|H\|=4$ |  |  |  |
| Min |  | 2.0 | 1.8 | 1.4 | 0.2 |  | 120.4 | 434.1 | 17.6 | 0.6 |
| Average |  | 164.6 | 277.6 | 288.9 | 30.8 |  | 206.4 | 560.2 | 32.3 | 2.6 |
| Max |  | 630.0 | 868.5 | 1163.1 | 124.7 |  | 483.7 | 880.7 | 51.2 | 7.3 |
|  |  | $\|H\|=5$ |  |  |  |  | $\|H\|=5$ |  |  |  |
| Min |  | 3.0 | 2.5 | 1.6 | 0.7 |  | 172.0 | 618.5 | 21.1 | 0.6 |
| Average |  | 224.1 | 370.9 | 410.2 | 42.0 |  | 266.5 | 762.0 | 41.5 | 3.7 |
| Max |  | 867.1 | 1317.5 | 1619.3 | 134.9 |  | 551.4 | 966.1 | 68.1 | 12.3 |

Although this was the first value from which we could find solutions, for some small sized instances the routing times displayed are too big (as ex1 in Table 5).

Next, we compare the performance of the models used within the matheuristic, to select a vehicle service from the pool generated, $F T$. Although it is assumed that an adequate way to deal with this real application is to limit the total similarity and then to minimise the routing time, it would be interesting to observe the upper bounds and average values on the total similarities

The bound on the total maximum similarity computed through propositions 1-2 is given in Table 4 (columns two to five). Columns six to nine depict the average total similarity of the feasible solutions found. We may observe that their percentage over their maximum values regarding ( $\mathrm{MRH} \mu$ ), varies between $60 \%$ (MRH0.1 with $|H|=3$ ) with and $90 \%$ (MRH0 with $|H|=3$ ). Finally, the matheuristic with model ( $\operatorname{MR} \mu$ ) found the smallest average total similarity values, as expected.

We also noticed that the most frequent value regarding the maximum total similarities of the feasible solutions for models (MRH $\mu$ ) over the smaller instances with $|H|=3$ is one (the maximum). This was not achieved by remaining values of $H$ for any of the instances, $(\mathrm{MR} \mu)$ is always close to its limit.

Regarding other perspectives about identical tours we look deeply to some of the instances results and we noticed, on one hand, that models ( $\mathrm{MRH} \mu$ ) tend to select only a few tours and repeat it every other day, which is a drawback. On the other hand, even not imposed, the sequences of tasks served in different tours do not usually coincide. The tours selected within (MR $\mu$ ), which imposes a maximum similarity between any pair of tours, do not share this disadvantage.

Table 5 depicts total routing time gap values for the matheuristic using models (MRH0), (MRH0.1), (MRH0.3) and (MR0.3), in columns three to six for $|L|=3$ and in columns eight to eleven for $|L|=6$. Each instance bounds are computed against the better value. Thus, if $\mathrm{RT}^{*}$ is the lowest routing time value for an instance (i.e., the better upper bound) the percentage gap for model (MR\#) that generates a feasible solution with a total routing time equal to RT\# is gap $(M R \#)=\frac{R T \#-\mathrm{RT}^{*}}{\mathrm{RT}^{*}} \times 100 \%$. Lines ending each group (for $|H|=3,4,5$ ) display minimum, average and maximum gap values to summarize the information.

As can be observed, more restrictive models regarding similarity do not deteriorate to much the total routing time. In fact, models (MRH0) and (MRH0.1) routing time gap values vary between $0 \%$ and $4.4 \%$, being, as expected, (MRH0.3) the better one. Model (MR0.3) sometimes generate feasible vehicle services with very high total routing time. This results from the fact that $\mu=0.3$ is the lowest value of the parameter that (MR $\mu$ ) can handle, and probably FT includes only one group of $|H|$ tours that meet the similarity requirement, for smaller instances. However, for the bigger instances, its performance is quite good.

Average, minimum and maximum values for the computational times (in seconds) referring to models (MRH $\mu$ ) and ( $\mathrm{MR} \mu$ ) may be consulted in Table 6. These values are considered small as they vary between 0 and $1,620 \mathrm{~s}$ ( 30 min ), no matter they do not include the feasible tours generation from Table 2.

To sum up, for smaller instances the valid model (M1DAR) seems to be the best option, in case it is acceptable to repeat tours every other day. On the other hand, (MR0.3) seems to be a good option for larger instances, as having a maximum similarity for all pairs of tours controlled, its routing times are close to the better ones.

## 6. FINAL REMARKS

In this work, we present a new problem named DARP, Dissimilar Arc Routing Problem. It arises in one application where service is to be performed on arcs, every day of a time horizon, and similar tours should be avoided to prevent robberies.

We propose a definition of similarity between two tours based on the number of tasks that are visited by both tours in the same time periods of the day. Constraints can be used to prevent the selection of similar tours. A measure is also proposed to evaluate the total similarity of a group of tours.

A mixed integer linear programming formulation is presented for DARP and the computational results show that CPLEX is able to solve small sized instances. To deal with larger instances a matheuristic is developed. Framed on the matheuristic, different constraints to avoid similarity of tours were essayed and evaluated. One of the alternatives tested (MR0.3) displayed a better balance for total routing time and total similarity, and it is not very demanding in terms of cpu time.

Topics for future research include several vehicles with a fixed capacity, different tasks demand for service in different days with distinct periodicities, as well as a further study on constraints to avoid similarities.

## ACKNOWLEDGMENTS

The authors thank anonymous referees which contribute to an improved version of the written work and to Fundação para a Ciência e Tecnologia (FCT).

## REFERENCES

[1] C. Archetti and M. Speranza, A survey on matheuristics for routing problems, Eur J Comput Optim 2 (2014), 223-246.
[2] Y. Dadkar, D. Jones, and L. Nozick, Identifying geographically diverse routes for the transportation of hazardous materials, Transp Res Part E Logist Transp Rev 44 (2008), 333-349.
[3] J. De Kort, Lower bounds for symmetric $K$-peripatetic salesman problems, Optimization 22 (1991), 113-122.
[4] J. De Kort, A branch and bound algorithm for symmetric 2peripatetic salesman problems, Eur J Oper Res 70 (1993), 229-243.
[5] J. De Kort, and A. Volgenant, On the generalized 2-peripatetic salesman problem, Eur J Oper Res 73 (1994), 175-180.
[6] E. Duchenne, G. Laporte, and F. Semet, Branch-and-cut algorithms for the undirected $m$-Peripatetic Salesman Problem, Eur J Oper Res 162 (2005), 700-712.
[7] E. Duchenne, G. Laporte, and F. Semet, The undirected $m$ capacitated peripatetic salesman problem, Eur J Oper Res 223 (2012), 637-643.
[8] E. Erkut, S. Tjandra, and V. Verter, "Hazardous materials transportation," Handbooks in operations research and management science, transportation, Chapter 9, C. Barnhart and G. Laporte (Editors), North Holland, Amsterdam, 2007, pp. 539-621.
[9] L. Gouveia, M.C. Mourão, and L.S. Pinto, Lower bounds for the mixed capacitated arc routing problem, Comput Oper Res 37 (2010), 692-699.
[10] J. Krarup, "The peripatetic salesman and some related unsolved problems," Combinatorial programming methods and applications, B. Roy, (Editor), Reidel, Dordrecht, 1975, pp. 173-178.
[11] Y. Lim and S. Rhee, An efficient dissimilar path searching method for evacuation routing, Journal of Civil Engineering 14 (2010), 61-67.
[12] J. Michallet, C. Prins, L. Amodeo, F. Yalaoui, and G. Vitry, Multi-start iterated local search for the periodic vehicle routing problem with time windows and time spread constraints on services, Comput Oper Res 41 (2014), 196-207.
[13] S. Ngueveu, C. Prins, and R. Wolfer-Calvo, Lower and upper bounds for the $m$-peripatetic vehicle routing problem, 4OR Q J Oper Res 8 (2010), 387-406.
[14] S. Ngueveu, C. Prins, and R. Wolfler-Calvo, A hybrid tabu search for the $m$-peripatetic vehicle routing problem, Matheuristics 10 (2010), 253-266.
[15] L. Talarico, K. Sörensen, and J. Springael, The $k$-dissimilar vehicle routing problem, Eur J Oper Res 244 (2015), 129-140.
[16] L. Talarico, K. Sörensen, and J. Springael, Metaheuristics for the risk-constrained cash-in-transit vehicle routing problem, Eur J Oper Res 244 (2015), 457-470.
[17] L. Talarico, J. Springael, K. Sörensen, and F. Talarico, A large neighbourhood metaheuristic for the risk-constrained cash-in-transit vehicle routing problem, Comput Oper Res 78 (2017), 547-556.
[18] R. Wolfler-Calvo and R. Cordone, A heuristic approach to the overnight security service problem, Comput Oper Res 30 (2003), 1269-1287.
[19] S. Yan, S. Wang, and M. Wu, A model with a solution algorithm for the cash transportation vehicle routing and scheduling problem, Comput Ind Eng 63 (2012), 464-473.


[^0]:    Received July 2017; accepted July 2017
    Correspondence to: M. Cândida Mourão; e-mail: cmourao@iseg.ulisboa.pt Contract grant sponsor: FCT/MCTES and FEDER; Contract grant number: Partnership Agreement PT2020 and projects UID/MULTI/00491/2013; UID/MAT/04561/2013; PTDC/MAT-NAN/2196/2014
    DOI 10.1002/net. 21763
    Published online 23 August 2017 in Wiley Online Library (wileyonlinelibrary.com).
    (C) 2017 Wiley Periodicals, Inc.

