Routing of delivery vehicles in a city with access time windows

J. Muñuzuri, J. Larrañeta & J. N. Ibáñez
School of Engineering, University of Sevilla, Spain

Abstract

The introduction of access time windows, or periods of the day during which delivery vehicles are not allowed to enter certain areas of a city, is a policy commonly applied by local authorities in order to better reorganise urban traffic flows. However, there are no estimations of the additional costs inflicted on carriers, even though the need for more complicated routes, and often more vehicles, is evident. This paper describes a model for estimating the effect of introducing access time windows in a given city by minimising the number of required vehicles and the route lengths and comparing them with the required vehicles and routes without time windows. The linear model used for the optimisation is described, as well as the solutions obtained for a test network, consisting of a number of nodes and links, part of which cannot be entered during a certain period of the day.

1 Introduction

The widespread use of the Vehicle Routing Problem with Time Windows (VRPTW) (Bramel and Simchi-Levi [3]) in scientific literature and commercial software applications enables the planning of routes for a vehicle fleet when certain time restrictions are imposed by the recipients, i.e. the obligation of making each delivery during a fixed time slot. However, in the case of urban deliveries, it is found that in many cases these time window restrictions are not imposed by the recipients, but by the local administration. The impossibility of accessing an area of the city, normally the city centre, during a relatively large period of the day clearly imposes restrictions on the routing of delivery vehicles. And the VRPTW is not sufficient for taking into account these access time windows, since it is not only the destination nodes the ones that are not to be
reached during the time windows, but all the network area around them, which affects the calculation of available routes.

Access time windows represent one of the most commonplace policies used by local authorities to control the accessibility to congested urban areas (OECD [7]). And the high density of commercial premises in these central areas implies a high number of deliveries. In certain sectors, transport companies often have to make deliveries at several points on the same day in the restricted area, and therefore a routing procedure which contemplates access time windows is applicable.

Besides, a relevant aspect of city logistics policies is that the implicit cost increments they induce in freight carriers, as well as the effects on pollution and congestion due to route alterations or the need for a higher number of vehicles, is often unknown. The use of the model presented here and its application to deliveries in the congested areas of a city may improve the understanding on the effect of policies on carriers’ practices.

2 Problem description

The problem considered is based on a standard vehicle routing problem (VRP) (Ball et al [1]): a vehicle fleet, departing from a depot and returning to it at the end of their routes, must make deliveries at certain points located in a network. The main assumptions made for the model were:

- The model has a double objective: the most important aim is to determine the minimum number of vehicles required, but it is also sought to calculate the optimal routes that must be followed by those vehicles so that the total time is as small as possible.
- All vehicles leave the depot at time = 0.
- No capacity restrictions are considered. This is realistic in the case of less-than-truckload urban deliveries, where the load factors of vehicles is often below 50% (GART [5]; Larrañeta et al [6]).
- Each destination point has a fixed delivery time during which the delivery vehicle must remained parked at the node.
- The restricted time window period is assumed to be located in the middle of the day, that is, there is a certain fraction of the working day before the restricted period starts, and another fraction after the restricted period finishes. This is also realistic in case of urban access policies.

3 The model

The modelling of the VRPATW (Vehicle Routing Problem with Access Time Windows) is described in this section. The objective of the model is to determine the minimum possible amount of delivery vehicles required and the routes to be followed by them in order to make deliveries in a certain number of points in a city. This city has a restricted access area, which also contains several delivery destinations, and which can only be entered outside of a fixed period of the day.
3.1 Data required

The known data to be used in the model formulation is:

\[ G(N,A) \]: graph corresponding to the urban area, with the links \((i,j)\) being directional.
\[ N \]: depot node (there is only one depot)
\[ D \]: destination nodes
\[ Q \]: final arrival node (the same node as the depot)
\[ AV \]: length of the period of the day before the restricted period starts (in time units)
\[ V \]: length of the restricted period (in time units)
\[ DV \]: length of the period of the day after the restricted period ends (in time units)
\[ c_{ij} \]: time required to travel from node \(i\) to node \(j\) via the link \((i,j)\) outside of the restricted period (in time units).
\[ d_{ij} \]: time required to travel from node \(i\) to node \(j\) via the link \((i,j)\) within the restricted period (in time units). They are equal to the \(c_{ij}\) for the nodes outside the restricted area, and infinite (very high) for those within the restricted area.
\[ T_D \]: delivery time at destination node \(D\).
\[ K \]: fixed cost of using a vehicle.
\[ K' \]: very high constant

3.2 Variables notation

The notation used in the model is the following:
\[ x_{ij}^t \]: for each link \((i,j)\) in the graph, it takes the value 1 if vehicle \(t\) crosses the link before the restricted period, and 0 if it does not.
\[ y_{ij}^t \]: for each link \((i,j)\) in the graph, it takes the value 1 if vehicle \(t\) crosses the link during the restricted period, and 0 if it does not.
\[ z_{ij}^t \]: for each link \((i,j)\) in the graph, it takes the value 1 if vehicle \(t\) crosses the link after the restricted period, and 0 if it does not.
\[ s_i^t \]: represents the time at which vehicle \(t\) passes each node \(j\) in the graph.
\[ \alpha_D^t \]: takes the value 1 if vehicle \(t\) makes a delivery at destination node \(D\) before the restricted period, and 0 otherwise.
\[ \beta_D^t \]: takes the value 1 if vehicle \(t\) makes a delivery at destination node \(D\) during the restricted period, and 0 otherwise.
\[ \gamma_D^t \]: takes the value 1 if vehicle \(t\) makes a delivery at destination node \(D\) after the restricted period, and 0 otherwise.
3.3 Model formulation

Minimise: $\sum_{i,j,t} c_{ij} \cdot x_{ij}^t + \sum_{i,j,t} d_{ij} \cdot y_{ij}^t + \sum_{i,j,t} c_{ij} \cdot z_{ij}^t + K \cdot \sum_{j,t} (x_{nj}^t + y_{nj}^t + z_{nj}^t)$

Subject to: 

\[ x_{ij}^t + y_{ij}^t + z_{ij}^t \leq 1 \quad \forall i, j, t \]  

\[ \sum_i x_{ij}^t = \sum_k x_{jk}^t + \sum_k y_{jk}^t \quad \forall j, t \]  

\[ \sum_i y_{ij}^t = \sum_k y_{jk}^t + \sum_k z_{jk}^t \quad \forall j, t \]  

\[ \sum_i z_{ij}^t = \sum_k z_{jk}^t \quad \forall j, t \]  

\[ \sum_{j,t} (x_{nj}^t + y_{nj}^t + z_{nj}^t) = \sum_{i,t} (x_{iQ}^t + y_{iQ}^t + z_{iQ}^t) \]  

\[ \sum_{t} (\alpha_D^t + \beta_D^t + \gamma_D^t) = 1 \quad \forall D \]  

\[ \sum_{i,j} c_{ij} \cdot x_{ij}^t + \sum_{D} \alpha_D^t \cdot T_D \leq AV \quad \forall t \]  

\[ \sum_{i,j} d_{ij} \cdot y_{ij}^t + \sum_{D} \beta_D^t \cdot T_D \leq DV \quad \forall t \]  

\[ \sum_{i,j} d_{ij} \cdot y_{ij}^t + \sum_{D} \gamma_D^t \cdot T_D \leq V \quad \forall t \]  

\[ \sum_i x_{id}^t \geq \alpha_D^t \quad \forall D, t \]  

\[ \sum_i y_{id}^t \geq \beta_D^t \quad \forall D, t \]  

\[ \sum_i z_{id}^t \geq \gamma_D^t \quad \forall D, t \]  

\[ s_i^t + c_{ij} - K' (1 - x_{ij}^t) \leq s_j^t \quad \forall i, j, t \]  

\[ s_i^t \geq 0 \quad \forall i, t \]  

\[ s_N^t = 0 \quad \forall t \]  

\[ x_{ij}^t, y_{ij}^t, z_{ij}^t = 0, 1 \]

This is therefore a linear model with binary variables. The objective function minimises the total sum of route costs, including those incurred before, during and after the restricted period, plus the cost associated to the use of vehicles. The restrictions have the following implications:

1. Vehicle $t$ can only cross link $(i,j)$ before, during or after the restricted period.
(2) If vehicle $t$ enters a node $j$ before the restricted period, it must leave the node before or during the restricted period.

(3) If vehicle $t$ enters a node $j$ during the restricted period, it must leave the node during or after the restricted period.

(4) If vehicle $t$ enters a node $j$ after the restricted period, it must leave the node after the restricted period.

(5) All vehicles $t$ exiting the depot $N$ must return to the final node $Q$ (again the same depot node).

(6) All deliveries at destination nodes $D$ must be made.

(7) The time during which a vehicle $t$ is circulating before the restricted period must be smaller or equal to the time length $AV$ before the restricted period.

(8) The time during which a vehicle $t$ is circulating after the restricted period must be smaller or equal to the time length $DV$ after the restricted period.

(9) The time during which a vehicle $t$ is circulating during the restricted period must be smaller or equal to the time length $V$ of the restricted period.

(10) If a vehicle is to deliver at a destination node before the restricted period starts, it must access the node before the restricted period.

(11) If a vehicle is to deliver at a destination node during the restricted period starts, it must access the node during the restricted period.

(12) If a vehicle is to deliver at a destination node after the restricted period ends, it must access the node after the restricted period.

(13) This restriction avoids loops in the vehicles’ routes. It implies that, if vehicle $t$ crosses the link $(i,j)$, the time at which it reaches $i$ must be smaller than the time at which it reaches $j$.

(14) Vehicles are available at the depot $N$ from time origin, and reach all other nodes at positive times.

(15) Restrictions imposing 0 and 1 values for the binary variables.

### 4 Solution procedure using a genetic algorithm

The above problem being NP-hard, a heuristic solution method was implemented making use of a permutation genetic algorithm (Blanton, 1993). Given $n$ destination nodes in the network, each individual of the population was defined as a permutation of those $n$ nodes, and cross and mutation procedures were implemented. Also, in order to determine the best way to travel between each pair of destination nodes, a shortest path algorithm Dantzig [4] is applied to the network before running the genetic algorithm.
For each individual or chromosome, the assigning of destinations to vehicles is done as follows. The first vehicle is assumed to visit the first node of the chromosome permutation. Then the next nodes are added in the order expressed by the chromosome to the route until the vehicle is no more able to make all the deliveries before the end of the working day. Then the next node is assigned to a new vehicle, and so are the following nodes, and the procedure is repeated until all destination nodes have been assigned to a vehicle.

Figure 1: Cases in which the shortest paths need to be disaggregated in order to find out whether the route needs to be penalised.

When calculating the length of a route, in case the vehicle is inside the restricted area during the restricted time window period, that length is strongly penalised in the algorithm. This is the main difference between the solution method used here and that of a normal VRP.

This specific formulation of the problem also introduces an additional difficulty in the solution process. The difficulty appears in the two cases represented in Figure 1, which may be described as:

- Case 1: a vehicle makes a stop outside the restricted area during the restricted period, and in the next stop it is inside the restricted area and outside of the restricted period.
- Case 2: a vehicle makes a stop inside the restricted area outside of the restricted period, and in the next stop it is outside the restricted area and inside the restricted period.

The difficulty of these cases lies in the fact that it is not sufficient to know the times at which the vehicle reaches and leaves the stops. It is also necessary to
know at exactly what time it enters or leaves the restricted area, in order to know whether the route length should be penalised or not. To do so, the shortest path between the two stops \((i\text{ and } i+1 \text{ in case 1 and } j\text{ and } j+1 \text{ in case 2})\) needs to be disaggregated, and the exact time at which the vehicle passes each node of the path needs to be computed.

The final result, given by the best chromosome of the last iteration of the algorithm, is then the number of vehicles required and the routes to be followed by them, together with the length of each route.

The solutions obtained by the genetic algorithm for each scenario are shown in Tables 1 and 2. It can be seen that, whereas the scenario without a time window restriction requires only two vehicles to make all the deliveries, three vehicles are needed in the scenario with a time window.

The route for each vehicle is described showing the nodes contained in it, as well as the time at which the vehicle leaves each node (except for the returning to the depot, where the arrival time is shown). Besides the higher number of vehicles required, the total sum of route lengths in the second scenario adds up to 130 time units, while only 110 are needed in the first one.

It is interesting to note, in the scenario with time windows, one of the cases in which the shortest paths need to be disaggregated. In the route for vehicle 3, node 23, located inside the restricted area, is left at \(t=18\) and the next destination node, in this case the depot, is reached at \(t=26\). Since the time window restriction...
starts at t=20, the algorithm had to contemplate the times at which the vehicle
passes each individual node along that route. The result is that the vehicle leaves
the restricted area through node 11 at exactly t=20, and therefore the route length
does not need to be penalised.

Table 1: Solution to the VRPATW given by the genetic algorithm for the
test network without access time window.

<table>
<thead>
<tr>
<th>Route for vehicle 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
</tr>
<tr>
<td>t = 0</td>
</tr>
<tr>
<td>27</td>
</tr>
<tr>
<td>t = 15</td>
</tr>
<tr>
<td>30</td>
</tr>
<tr>
<td>t = 28</td>
</tr>
<tr>
<td>31</td>
</tr>
<tr>
<td>t = 44</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>t = 50</td>
</tr>
</tbody>
</table>

Table 2: Solution to the VRPATW given by the genetic algorithm for the
test network with access time window.

<table>
<thead>
<tr>
<th>Route for vehicle 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
</tr>
<tr>
<td>t = 0</td>
</tr>
<tr>
<td>16</td>
</tr>
<tr>
<td>t = 14</td>
</tr>
<tr>
<td>5</td>
</tr>
<tr>
<td>t = 27</td>
</tr>
<tr>
<td>30</td>
</tr>
<tr>
<td>t = 42</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>t = 54</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Route for vehicle 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
</tr>
<tr>
<td>t = 0</td>
</tr>
<tr>
<td>27</td>
</tr>
<tr>
<td>t = 15</td>
</tr>
<tr>
<td>14</td>
</tr>
<tr>
<td>t = 30</td>
</tr>
<tr>
<td>31</td>
</tr>
<tr>
<td>t = 44</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>t = 50</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Route for vehicle 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
</tr>
<tr>
<td>t = 0</td>
</tr>
<tr>
<td>23</td>
</tr>
<tr>
<td>t = 18</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>t = 26</td>
</tr>
</tbody>
</table>

5 Conclusions

As a variation of the general vehicle routing problem, the VRPATW (Vehicle
Routing Problem with Access Time Windows) covers the case, rather common
in urban freight policies, of having time restrictions for access to a certain area of
the city. The presented model attempts to formulate mathematically the
analytical problem, introducing several realistic assumptions.

Also, the genetic algorithm implemented, based on the VRPTW case but with
certain modifications, reaches the optimal solution for the test network in a very
short computational time. The need to re-evaluate in detail some route segments
because of the specific cases mentioned causes the algorithm to be slightly
slower than it would be in the case of a regular VRPTW. In any case, at this
stage the authors were concerned with the estimation of good solutions to the
problem, rather than with the speed of the algorithm.
The illustration of the model with the test network shows the need for more vehicles and more route time in the case of a time window scenario. The implications of this fact in terms of congestion and pollution may thus be evaluated. Future research along this line contemplates the application of this model to actual cities where a time window policy is implemented, in order to estimate the additional emissions, congestion and costs induced by it.

References


