Jan Fabry *

A DYNAMIC MESSENGER PROBLEM

Messenger services provide customers to deliver their package from a specific origin to a specific destination. Real situations require the fast messenger's reaction to the on-line customer's request. While it is not possible to change the planned route in static distribution problems, a dynamic version enables a dispatcher the integration of a new request to existing route for optimization. The mathematical models proposed in the paper are based on the Miller-Tucker-Zemlin's formulation of Traveling Salesman Problem. Because of NP hardness of the problem it is impossible for most real problems to find the optimal solution in acceptable time. Heuristic algorithms represent the important alternative to solving real dynamic problems. Basic formulation of the messenger problem can be extended to problems with time windows. Limited capacity of the vehicle and multiple vehicles can be also considered.

1. Introduction

In a static messenger problem (Cordeau, 2006), requirements of all customers are known before a dispatcher starts to plan the route. Each customer specifies an origin where the driver has to pick up the package and a destination for the package delivery. Distances between all the points in the distribution network are given. A single vehicle is available for distribution of all the packages and a capacity of the vehicle is large enough to load all of them. First, no time windows are considered in the problem. The objective is to minimize the total length of the route for pick-up and delivery of all the packages.

Example 1. Figure 1 shows an example of a static messenger problem with four customers. Each arc in the network corresponds to the requirement of a customer for delivery. Customers are purposely situated in the even nodes, while destinations in the odd nodes. If i is a number of the customer, $i+1$ is a number of its package destination. A depot of the vehicle is located in node 1.
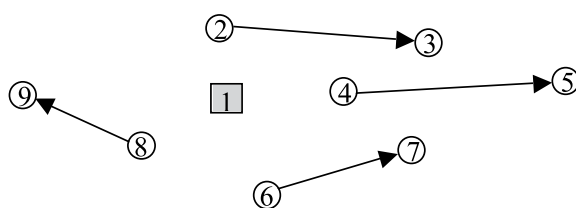

Fig. 1 Example of static messenger problem

2. Optimization model of a static messenger problem

Let us have n customers; each customer requires delivery of one package from his office to a specific destination. Considering one depot in the problem, there are $(2n + 1)$ locations in the dis-

tribution network. Let c_{ij} denote the shortest distance between locations i and j. The mathematical model of a static messenger problem is defined as follows (Fábry, 2006):

$$\text{Minimize } z = \sum_{i=1}^{2n+1} \sum_{j=1}^{2n+1} c_{ij} x_{ij}, \tag{1}$$

subject to

$$\sum_{j=1}^{2n+1} x_{ij} = 1, \qquad i = 1, 2, ..., 2n+1 \tag{2}$$

$$\sum_{i=1}^{2n+1} x_{ij} = 1, \qquad j = 1, 2, ..., 2n+1 \tag{3}$$

$$u_i - u_j + (2n+1)x_{ij} \leq 2n, \tag{4}$$

$$i = 1, 2, ..., 2n+1, \ j = 2, 3, ..., 2n+1, i \neq j$$

$$u_{2i} \leq u_{2i+1}, \qquad i = 1, 2, ..., n \tag{5}$$

$$u_1 = 0, \tag{6}$$

$$x_{ij} \in [0,1], \qquad i,j = 1, 2, ..., 2n+1 \tag{7}$$

A binary variable x_{ij} equals 1, if the vehicle goes from location i to location j, 0 otherwise. The objective (1) corresponds to the total length of the vehicle's route. The sets of equations (2) and (3) assure that each location is visited exactly once. Constraints (4) including variables u_i are Miller-Tucker-Zemlin's inequalities to avoid partial cycles in the solution. As each package has to be picked up before its delivery inequalities (5) must be respected.

Figure 2 illustrates the feasible route in the example given above. The package is delivered to location 9 immediately after its pick-up in location 8, while the package being picked up in location 6 is delivered to location 7 after several vists in other locations.

* **Jan Fabry**
  Department of Econometrics, University of Economics Prague, Czech Republic, E-mail: fabry@vse.cz
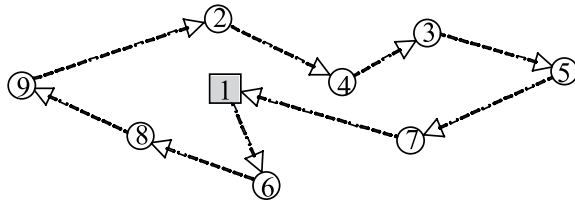
Fig. 2 Feasible route

3. Static messenger problem with time windows

In a real messenger problem each customer specifies time window for both pick-up and delivery of the package. For this purpose it is necessary to know travelling times t_{ij} between all pairs of locations. The optimization model follows (Fábry, 2006):

$$\text{Minimize } z = \sum_{i=1}^{2n+1} \sum_{j=1}^{2n+1} c_{ij} x_{ij}, \tag{8}$$

subject to

$$\sum_{j=1}^{2n+1} x_{ij} = 1, \quad i = 1, 2, ..., 2n+1, \tag{9}$$

$$\sum_{i=1}^{2n+1} x_{ij} = 1, \quad j = 1, 2, ..., 2n+1, \tag{10}$$

$$e_i \leq \tau_i \leq l_i, \quad i = 1, 2, ..., 2n+1, \tag{11}$$

$$\tau_i + t_{ij} - M(1 - x_{ij}) \leq \tau_j, \tag{12}$$

$$i = 1, 2, ..., 2n+1, \; j = 2, 3, ..., 2n+1, i \neq j,$$

$$\tau_{2i} \leq \tau_{2i+1}, \quad i = 1, 2, ..., n, \tag{13}$$

$$\tau_1 = 0, \tag{14}$$

$$x_{ij} \in [0,1], \quad i,j = 1, 2, ..., 2n+1. \tag{15}$$

The value of a variable τ_i determines when the location i is visited by the vehicle. The objective (8) corresponds to the total length of the vehicle's route. The set of constraints (9) and (10) are taken from the previous model. Parameters e_i and l_i in the inequalities (11) are the earliest poss⊄ible and latest acceptable times of pick-up (in case of even nodes) and delivery (in case of odd nodes). Inequalities (12) avoiding partial cycles include large constant M. The constraints assure for each package that its delivery will follow its pick-up.

In real applications, time windows can be defined in a different way. The firm specifies the earliest possible pick-up time and latest acceptable delivery time that have to be respected:

$$e_{2i} \leq \tau_{2i}, \quad i = 1, 2, ..., n, \tag{16}$$

$$\tau_{2i+1} \leq l_{2i+1}, \quad i = 1, 2, ..., n. \tag{17}$$

In case the soft time windows are given in the problem it is possible to consider penalties for time windows violations (Fábry, 2006).

4. Dynamic messenger problem

After finding the optimal solution of the static messenger problem the vehicle starts to pick up and deliver packages of all advanced customers. A new requirement for pick-up and delivery can occur during the travel. These two actions will be integrated into the current route. In the paper two algorithms are described for this purpose: insertion method and re-optimization (Fábry, 2006).
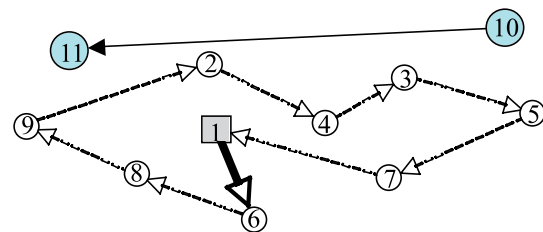

Fig. 3 A new on-line requirement

Example 2. Suppose the route in Figure 2 is the optimal solution of example 1. When the vehicle is travelling from the depot to location 6, a new customer calls for pick-up of a package in location 10 and its delivery to location 11 (see Figure 3).

Insertion algorithm.

Let $U_N = [i_1, i_2, ..., i_m]$ be a sequence of m locations ($i_m = 1$) that have to be visited by the vehicle according to the plan. Denote by r location where a new package is to be picked up and $r + 1$ destination to which it must be delivered. If location r is inserted between the locations i_k and i_{k+1}, the destination $r + 1$ has to be visited after the pick-up of the package. Two possibilities are considered:

(1) Destination $r + 1$ is inserted immediately behind the location r (see Figure 4).
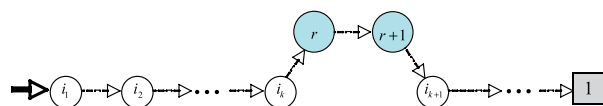

Fig. 4 Immediate delivery after pick-up of a new package

the extension of the current route is calculated as follows:

$$\Delta z_k = c_{i_k,r} + c_{r,r+1} + c_{r+1,i_{k+1}} - c_{i_k,i_{k+1}}, \tag{18}$$
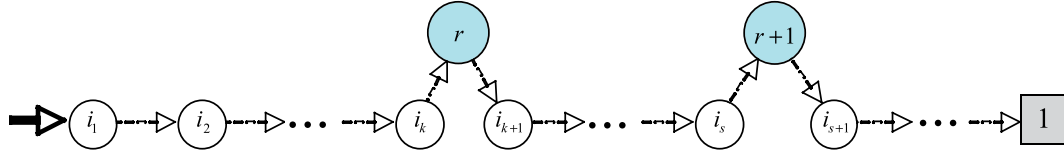
$$k = 1, 2, ..., m-1$$

Fig. 5 Visiting other locations between pick-up and delivery of a new package

The objective is to determine the index t minimizing function (18):

$$\Delta z_t^1 = \min_{k=1,2,\ldots m-1} \Delta z_k .$$ (19)

(2) While the location r is inserted between the locations i_k and i_{k+1}, the destination is inserted between the locations i_s and i_{s+1}. The destination has to be certainly inserted behind the location i_{k+1} (see Figure 5).

The extension of the route is

$$\Delta z_{ks} = c_{i_k,r} + c_{r,i_{k+1}} - c_{i_k,i_{k+1}} + c_{i_s,r+1} + c_{r+1,i_{s+1}} - c_{i_s,i_{s+1}},$$ (20)

$$k = 1, 2, \ldots, m-2, \ s = k+1, k+2, \ldots, m-1.$$

The objective is to find the following indices t and p:

$$\Delta z_{tp} = \min_{\substack{k=1,2,\ldots m-2 \\ s=k+1, k+2, \ldots, m-1}} \Delta z_{ks} .$$ (21)
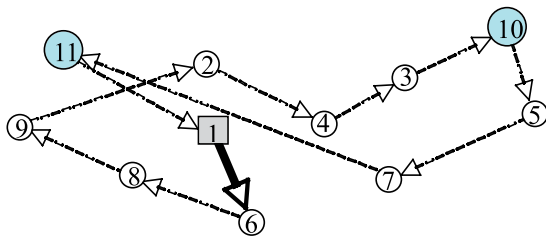

Fig. 6 Application of the insertion method

Comparing values (20) and (21), the lower of them determines the modification of the current route. The application of this strategy is illustrated in Figure 6.

Re-optimization.

Re-optimization algorithm consists in finding the optimal route of the vehicle after accepting a new request. The current route including all non-visited locations is re-optimized for the best incorporation of pick-up and delivery of the new package. For this purpose, it is possible to define the following mathematical model:

$$\text{Minimize } z = \sum_{i \in U_N} \sum_{j \in U_N} c_{ij} x_{ij} ,$$ (22)

subject to

$$\sum_{j \in U_N} x_{ij} = 1, \quad i \in U_N ,$$ (23)

$$\sum_{i \in U_N} x_{ij} = 1, \quad j \in U_N ,$$ (24)

$$u_i - u_j + |U_N| \cdot x_{ij} \le |U_N| - 1,$$ (25)

$$i \in U_N, j \in U_N - \{1\}, i \ne j ,$$

$$u_{i_{2k-1}} \le u_{i_{2k}}, \quad i_{2k-1}, i_{2k} \in \omega, k = 1, 2, \ldots, \frac{|\omega|}{2} - 1,$$ (26)

$$u_1 = 0,$$ (27)

$$x_{1j_{next}} = 1,$$ (28)

$$x_{ij} \in [0,1], \quad i,j \in U_N .$$ (29)

First, the original matrix of distances must be changed. Let j_{next} be an index of the location the vehicle is approaching when a new customer calls. Then, distance $c_{1j_{next}}$ between depot 1 and this location will be adjusted to the value corresponding to the length of the vehicle's route that will have been completed at the location j_{next}. Set U_N contains indices of locations that have not been visited yet, including depot 1 and two locations for pick-up and delivery of a new package. A number of all indices is denoted by $|U_N|$. The set includes two following subsets:

(1) the set of pick-up and delivery locations of packages that have not been picked up yet,
(2) delivery locations of the packages having been picked up before reaching the location j_{next}.

Let denote an increasing sequence of indices from the first of these sets. If j_{next} is the index of a pick-up location it will not be included (together with its delivery point) in the sequence ω. A number of all indices in the sequence denoted by $|\omega|$ is always an even value. The set of inequalities (26) is defined just for members of the sequence to assure each package will be picked up before its delivery. These constraints have the same relevancy as restrictions (5) in the static messenger problem.

In example 2 the set U_N contains all the indices, i.e. $U_N = \{1, 2, \ldots, 11\}$. As a new requirement occurs when the vehicle is approaching location 6, we set $j_{next} = 6$. The sequence of indices of pick-up and delivery locations which package will not be on the

vehicle after visit location 6 is ω = [2, 3, 4, 5, 8, 9, 10, 11]. Using the optimization model (22) – (29), we obtain the route shown in Figure 7.
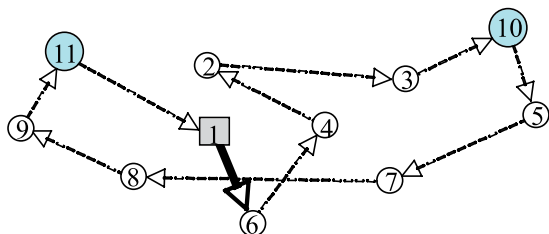

Fig. 7 Re-optimization of the route

5. Conclusion

In the paper, two approaches for a dynamic messenger problem are presented. While re-optimization algorithm can be used for problems with small numbers of locations that have to be visited, huge problems need heuristics. The insertion method is offered as a very simple and effective method. Considering time windows, the model is much closer to real messenger problems.

The analysis can be extended if a capacity of the vehicle is a significant attribute for tour designing (Cordeau, 2006). Several routes can be offered instead of the only one. In reality, the introduction of multiple vehicles is necessary for completing all requests in time. Vehicles can be located in one common depot or several separated depots. Besides the total distance travelled by vehicles, other criteria can be considered, e.g. total routing cost (Cordeau, 2006), time necessary for delivery of all packages, total profit, etc.

References

[1] CORDEAU, J.-F.: *A Branch-and-Cut Algorithm for the Dial-a-Ride Problem,* Operations Research 54, 2006, pp. 573–586.
[2] FABRY, J.: *Dynamic Vehicle Routing Problems (in Czech),* Dissertation thesis, VSE-FIS, Praha 2006, p. 141.