Multiple messenger problem

Jan Fábry[1], Maria Kobzareva[2]

Abstract. Multiple messenger problem is a generalization of the problem, where more messengers can be used in the solution. This problem is frequently solved problem in logistic management and it is more appropriate for real-life applications, because companies often use several messengers to transport shipments. The paper describes static problems with multiple messengers and suggests a possibility of solving such problems with modified heuristic methods. To solve multiple messenger problem, modified nearest neighbor heuristics and modified insertion heuristics are used. To reduce transportation costs, the generated solution is improved using modified exchange heuristic method. The main contribution of the paper are applications, developed in VBA in MS Excel that can solve static problems with multiple messengers and that can be beneficial for real companies. Results of the computation experiments are presented in the paper.

Keywords: travelling salesman problem, distribution problem, static multiple messenger problem, modified nearest neighbor heuristics, modified insertion heuristics, modified exchange heuristics

JEL Classification: C44
AMS Classification: 90C15

1 Introduction

Messenger problem is a special instance of the travelling salesman problem (TSP), where request consists of a pick-up location and a destination location. It is analogous to dial-a-ride problem described in literature. Psaraftis [8] created exact algorithm for the static problem with one vehicle. Cordeau [2] offers branch-and-cut algorithm for solving this problem. Heuristic algorithms for the transportation of handicapped persons are presented by Toth and Vigo [9]. Jorgensen et al. [7] use genetic algorithms for solving the problem.

Multiple messenger problem, where more than one messenger can be used to transport shipments, consists of generating a set of routes for *m* messengers. The problem is based on multiple travelling salesman problem [1]. Gavish and Srikanth [5] offer the optimal solution method for large-scale multiple TSP. In multiple messenger problem, all messengers start at and turn back to a depot respecting the essential constraint that a messenger cannot visit the delivery location before the pick-up location. Solution of multiple vehicle dial-a-ride problem using tabu search heuristics is described by Cordeau and Laporte [3]. The purpose of this paper is to present modified heuristic methods, being developed specially for multiple messenger problem. The text consists of the following parts: section 2 describes nearest neighbor heuristic algorithms, section 3 presents insertion heuristic algorithm and section 4 offers exchange heuristic algorithm for solving static multiple messenger problem. In section 5, results of computational experiments are presented and in section 6 some concluding statements are given.

2 Modified nearest neighbor algorithm

Nearest neighbor heuristics for TSP is a sequential construction process that starts by initializing the current route at the depot. Then, the customer nearest to the depot, in terms of a given cost or distance, is selected to extend the route. Subsequently, the customer nearest to the last customer will be added to the route, if its selection does not cause any constraint violation [6]. The process is repeated until the Hamiltonian cycle is generated. In [4], the messenger problem is defined for the network in which all even nodes correspond to

[1] Jan Fábry, University of Economics Prague, Faculty of Informatics and Statistics, Department of Econometrics, Prague 3, W. Churchill sq.4, fabry@ vse.cz

[2] Maria Kobzareva, University of Economics Prague, Faculty of Informatics and Statistics, Department of Econometrics, Prague 3, W. Churchill sq.4, Ph.D. student, mariakobzareva@gmail.com

pick-up locations and all odd nodes to delivery locations. In addition, if a shipment is picked-up in node i, it will be delivered to node $i+1$. The modification of the heuristic method offers in each step such nodes that can be selected for extension of the route. Thus, in the first step, only pick-up locations (even nodes) can be considered. The modified nearest neighbor heuristic algorithm for static multiple messenger problem is presented below. Let denote 1 as a depot and n as a number of all locations (respecting the previous assumption, n must be the odd number).

Algorithm 1

Let T be the maximum number of messengers that can be used in the solution, A_t is a set of locations that had not been visited yet and could be visited by a messenger t at a given step, i.e. at the beginning of the algorithm, all sets A_t contain only even numbers 2, 4, …, $n-1$. S_t is a sequence of locations being subsequently visited by messenger t and i_t is the last node, visited by messenger t, c_{ij} is the minimal distance between nodes i and j, z_t is a distance, which messenger t travels on the route, z is the total distance travelled by all messengers.

Step 1:

$A_t = \{2, 4, \ldots, n\text{-}1\}$, $S_t = \{1\}$, $i_t = 1$, $z_t = 0$, $t = 1, 2, \ldots, T$.

Step 2:

$c_{i_m k} = \min_{t=1,2,\ldots,T} \min_{j \in A_t} c_{i_t j}$,

$S_m = S_m + \{k\}$, $z_m = z_m + c_{i_m k}$,

$A_t = A_t - \{k\}$, $t = 1, 2, \ldots, T$,

if k is even, then $A_m = A_m + \{k+1\}$,

$i_m = k$.

Step 3:

If $A_t = \emptyset$ for all $t = 1, 2, \ldots, T$, go to step 4, else go to step 2.

Step 4:

$S_t = S_t + \{1\}$, $z_t = z_t + c_{i_t 1}$, $t = 1,2, \ldots, T$ (only for $S_t \neq \{1\}$),

$$z = \sum_{t=1}^{T} z_t \,.$$

End

3 Modified insertion algorithm

Insertion heuristics is based on extending the current route by inserting one location at each iteration. In the first step, we find the farthest location from the depot for each messenger, and then we insert other nodes trying to achieve minimum extension of current routes. The modification of the insertion algorithm, used for multiple TSP, has to respect the following conditions:

a) related pick-up and delivery locations (shipment) must not be included in two different routes,

b) delivery location for each shipment has to be visited after its picking-up.

The modified insertion heuristic algorithm for multiple messenger problem is presented below. Ordering of all nodes follows the assumptions of the nearest neighbor algorithm presented above.

Algorithm 2

Let T be the number of messengers that will be used to transport all shipments, A_t is a set of locations that had not been visited yet and could be visited by a messenger t at a given step, i.e. at the beginning of the algorithm, all sets A_t contain numbers 2,3, …, n. S_t is a sequence of locations being subsequently visited by messenger t, s_i^t is the element of sequence S_t on position i, h_t is the number of elements in sequence S_t, c_{ij} is the minimal distance between nodes i and j, z_t is a distance, which messenger t travels on the route, z is total distance travelled by all messengers, Δz is an increase in total distance.

Step 1:

$A_t = \{2,3,\dots, n\}$, $t = 1, 2, \dots, T$,

for $t = 1, 2, \dots, T$ repeat:

$c_{1k} = \max_{j \in A_t} c_{1j}$,

$S_t = \{1, k, 1\}$, $z_t = c_{1k} + c_{k1}$, $h_t = 3$; $A_t = A_t - \{k\}$,

for $i = 1,2, \dots, T$, $i \neq t$ do:

$A_i = A_i - \{k\}$,

if k is even, then $A_i = A_i - \{k+1\}$,

if k is odd, then $A_i = A_i - \{k-1\}$.

Step 2:

$\Delta z = \infty$,

for $t = 1, 2, \dots, T$ repeat:

for each $r \in A_t$ repeat

a) if r is even and sequence S_t contains node $r + 1$ on position f, then

$$\Delta z_{rj} = \min_{i=1,2,\dots,f-1} (c_{s_i^t r} + c_{r s_{i+1}^t} - c_{s_i^t s_{i+1}^t}),$$

b) if r is odd and sequence S_t contains node $r - 1$ on position f, then

$$\Delta z_{rj} = \min_{i=f,f+1,\dots,h_t-1} (c_{s_i^t r} + c_{r s_{i+1}^t} - c_{s_i^t s_{i+1}^t}),$$

c) else

$$\Delta z_{rj} = \min_{i=1,2,\dots,h_t-1} (c_{s_i^t r} + c_{r s_{i+1}^t} - c_{s_i^t s_{i+1}^t}).$$

If $\Delta z_{rj} < \Delta z$, then $\Delta z = \Delta z_{rj}$, $v = r$, $q = j$, $m = t$.

Insert node v to sequence S_m after element s_q^m,

$h_m = h_m + 1$,

$z_m = z_m + \Delta z$,

$A_m = A_m - \{v\}$,

for $i = 1,2, \dots, T$, $i \neq m$ do:

$A_i = A_i - \{v\}$,

if v is even, then $A_i = A_i - \{v+1\}$,

if v is odd, then $A_i = A_i - \{v-1\}$.

Step 3:

If $A_t = \emptyset$, for all $t = 1, 2, \dots, T$, then go to step 4, else go to step 2.

Step 4:

$$z = \sum_{t=1}^{T} z_t .$$

End

4 Modified exchange algorithm

Both nearest neighbor algorithm and insertion algorithm described above generate the solution of multiple messenger problem. It can be improved using modified exchange algorithm. The main idea is to exclude shipments from generated routes and include them in other routes if this exchange is advantageous.

Algorithm 3

Let T be the number of messengers with routes generated by Algorithm 1 or Algorithm 2, S_t is a sequence of locations to be subsequently visited by messenger t, s_i^t is the element of sequence S_t on position i.

Step 1:

For each messenger route, find a shipment, i.e. pair of nodes i (even) and $i + 1$ (odd), which will bring maximum reduction of the length of the route when excluded from the route. These shipments are candidates for possible exchange. Denote the reductions as Δzk_t ($t = 1, 2, \ldots, T$). Select route m with the maximal reduction:

$$\Delta zk_m = \max_{t=1,2,\ldots,T} \Delta zk_t.$$

Let this reduction correspond to the extraction of nodes r and $r + 1$ from route m.

Step 2:

For $t = 1, 2, \ldots, T$, $t \neq m$ determine the value of minimum extension of the appropriate route when inserting nodes r and $r + 1$. Denote these extensions as Δpr_t. Select route u with the minimal extension:

$$\Delta pr_u = \min_{\substack{t=1,2,\ldots,T \\ t\neq m}} \Delta pr_t.$$

Let the extension of route u correspond to visiting node r after node s_k^u, and node $r + 1$ after node s_l^u, eventually visiting both nodes after node s_k^u [4].

Step 3:

If $\Delta zk_m > \Delta pr_u$, then accomplish the exchange and continue with step 1, else go to the end.

End

The algorithm can be modified in several ways. For example, if at the step 3 the algorithm stopped because of no effective exchange was found, we could continue with step 1 and try to reduce the length of route with the second greatest value of Δzk_t, eventually the third one, etc. More time demanding modification is based on comparison of maximum reductions of all routes with minimum extensions of all other routes after the inclusion of corresponding pairs of nodes. Thus, in step 3, there is accomplished the exchange corresponding to the maximum positive difference $\Delta zk_i - \Delta pr_j$ ($i,j = 1,2, \ldots, T$; $i \neq j$). Another opportunity how to use effectively the modified exchange algorithm is to try to reduce the length of the longest route regardless this reduction is maximum within all routes, eventually to forbid the assignment of additional shipments to busy messengers with long routes.

5 Computational experiments

To perform computational experiments, we generated 25 multiple messenger problems with 21 nodes. Modified nearest neighbor heuristics and insertion heuristics were applied to solve all generated instances. Then we applied modified exchange heuristics to improve obtained results. The instances were divided into 6 groups; each group had different variation of minimal distances between nodes. In the first group, minimal distances between nodes varied up to 60 km, in the last group to 200 km. Each group had 5 generated cases. Figure1 contains the results of computational experiments with the application of the algorithms generating multiple routes. Figure 2 illustrates results of applying modified exchange heuristics on the generated routes. Figures 3 and 4 present the efficiency of the use of exchange algorithm, applied on results, obtained by modified nearest neighbor heuristics and insertion heuristics.

It is obvious that in given set of instances nearest neighbor heuristics offers better solution than insertion heuristics in 96 % of cases, which could be partly given by the structure of minimal distances matrix. Nearest neighbor heuristics turns to be very efficient, as in every step it works only with nodes that can be inserted after the "last" node, so we do not have to test, whether we can add the current node. For example, at the beginning only pick-up (even) nodes can be selected, because no shipments are processed currently, i.e. there is nothing to deliver.

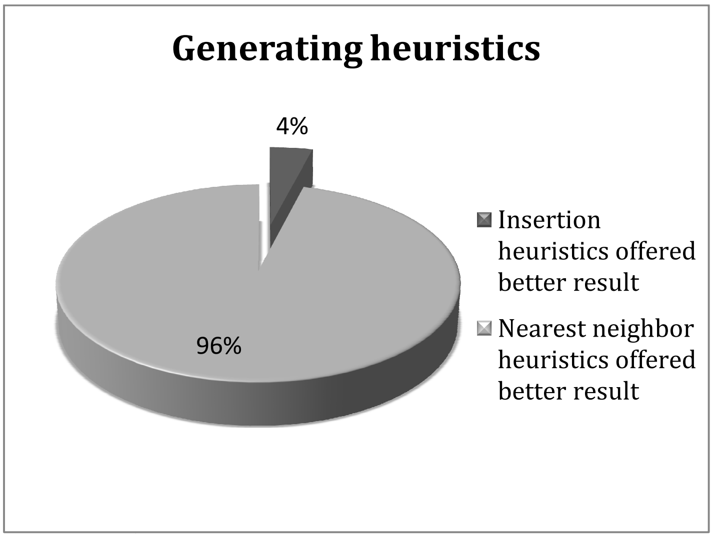

Figure 1 Comparison of modified heuristics for generating solution

When the modified exchange heuristics was applied on solutions generated by nearest neighbor heuristics and insertion heuristics, 63 % of results were better for nearest neighbor algorithm, while 37 % of them were better for insertion algorithm. It supports the idea that the modification of nearest neighbor heuristics for multiple messenger problem was designed successfully. We can also consider that insertion heuristics can achieve better solution in almost 40 % of cases after applying modified exchange heuristics on results, thus combination of insertion and exchange methods can be beneficial.

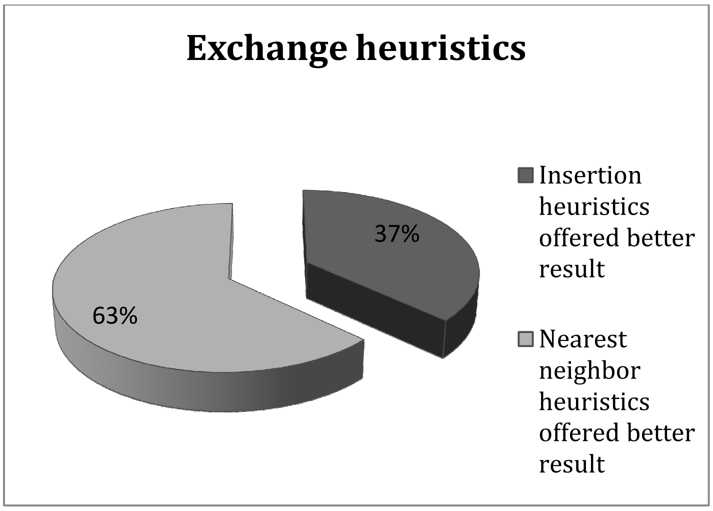

Figure 2 Modified exchange heuristics results

After applying exchange heuristics on solution, generated by modified nearest neighbor heuristics, the results (in terms of total travelled distance) were improved by 3 to 37 %. This fact suggests the idea that modified exchange heuristics is efficient and should be used to improve the solution, obtained by other heuristic methods. In 12 cases out of 25, the improvement after using exchange heuristics was greater than 15 %, only in 3 cases out of 25, the improvement was less than 5 %. In 2 cases, exchange heuristics was unable to find better solution. The average efficiency of solution improvement by modified exchange heuristics is 16 %.

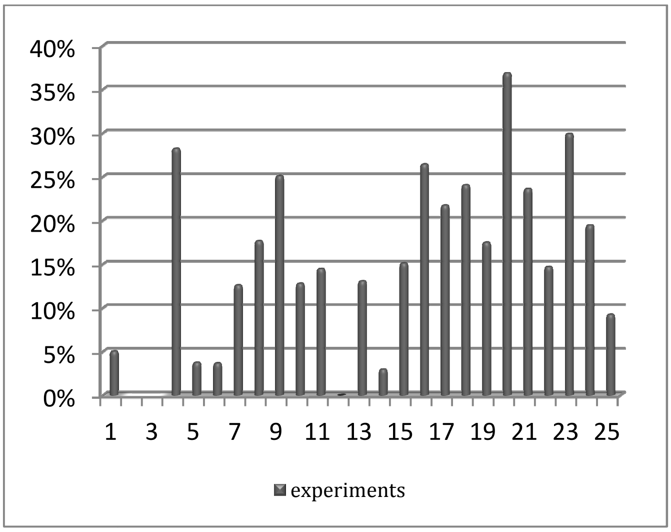

Figure 3 Efficiency of modified exchange heuristics applied on nearest neighbor heuristics

After applying exchange heuristics on solution generated by modified insertion heuristics, the results were improved by 9 to 54 %. We can see that modified exchange heuristics is even more effective when applied on the results, obtained by insertion heuristics. In 19 cases out of 25, the improvement after using exchange heuristics was greater than 20 %, only in 1 case out of 25, the improvement was less than 10 %. In 12 cases out of 25, the improvement was greater than 30 %; exchange heuristics was able to find better solution in every case. The average effectiveness of modified exchange heuristics is 29 %. This fact supports the idea that modified exchange heuristics is efficient and should be used to improve the solution, obtained by different heuristic methods, especially the insertion algorithm.

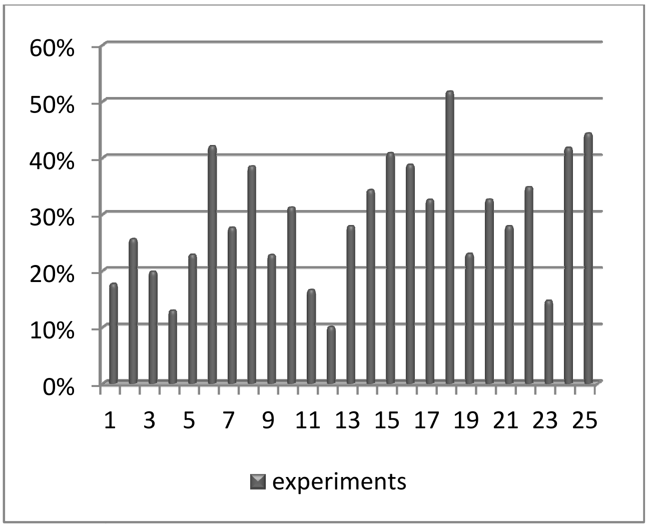

Figure 4 Efficiency of modified exchange heuristic applied on insertion heuristics

6 Conclusions

This paper offers modifications of nearest neighbor algorithm and insertion algorithm to solve multiple messenger problem. As the result, multiple routes are generated. For improvement of solutions, we modified the exchange algorithm, which enables to exclude shipments from some routes and include them in other routes to minimize total distance travelled by all messengers. Beside detailed description of all algorithms, we also present results of computational experiments to determine efficiency of proposed methods.

Acknowledgements

This publication was supported by the project F4/18/2011 founded by the Internal Grant Agency of the University of Economics Prague.

References

[1] Bektas, T. The Multiple Traveling Salesman Problem: an Overview of Formulations and Solution Procedures. *Omega*.Vol. 34, 2006, pp. 209-219.

[2] Cordeau, J.-F. A Branch-and-Cut Algorithm for the Dial-a-Ride Problem. *Operations Research*, Vol. 54, 2006, pp. 573-586.

[3] Cordeau, J.-F. and Laporte, G. A Tabu Search Heuristics for the Static Multi-Vehicle Dial-a-Ride Problem. *Transportation Research Part B*. Vol. 37, 2003, pp. 579-594.

[4] Fábry, J. Dynamic Messenger Problem. *Communications*, Vol. 9 (4), 2007, pp. 66-69.

[5] Gavish, B. and Srikanth, K. An Optimal Solution Method for Large-Scale Multiple Traveling Salesman Problems. *Operations Research*, Vol. 34, 1986, pp. 698-717.

[6] Hosny, M. Investigating heuristic and meta-heuristic algorithms for solving pickup and delivery problems, dissertation, *Cardiff University*, 2010.

[7] Jorgensen, R.M., Larsen, J., Bergvinsdottir, K.B. Solving the Dial-a-Ride Problem using Genetic Algorithms. *Journal of the Operational Research Society*, Vol. 58 (10), 2006, pp. 1321-1331.

[8] Psaraftis, H.N. An Exact Algorithm for the Single Vehicle Many-to-Many Dial-a-Ride Problem with Time Windows. Transportation Science., Vol. 17, 1983, pp. 351-357.

[9] Toth, P., Vigo, D. Heuristic Algorithms for the Handicapped Persons Transportation Problem. Transportation Science 31, 1997, pp. 60-71.