An Exact Solution Method for the MTSP

Author(s): Ahmad Husban

Source: *The Journal of the Operational Research Society*, May, 1989, Vol. 40, No. 5 (May, 1989), pp. 461-469

Published by: Palgrave Macmillan Journals on behalf of the Operational Research Society

Stable URL: https://www.jstor.org/stable/2583618

J. Opl Res. Soc. Vol. 40, No. 5, pp. 461–469, 1989
Printed in Great Britain.
0160-5682/89 $3.00 + 0.10

An Exact Solution Method for the MTSP

AHMAD HUSBAN
Mu'tah University, Mu'tah/Al-Karak, Jordan

A new mathematical model for the multi-travelling salesman problem (MTSP) is presented. The MTSP formulation is modified, and a branch-and-bound algorithm for solving this problem exactly is developed. The significance of this procedure is that it does not need to transform the problem into a single travelling salesman problem, which has been the case in the dominant algorithms for solving the above problem. Moreover, computational experience has shown that for a fixed number of cities to be visited, the time required to solve the problem decreases markedly as the number of salesmen increases.

Key words: branch-and-bound algorithm, modelling, multi-travelling salesman problem

INTRODUCTION

The travelling salesman problem (TSP) is one of the most intensely studied problems by operational researchers. The problem is to find the minimum-cost tour of a salesman residing in one city, the home city, and wishing to visit n cities before returning home without going through any city twice. A cost matrix, $C = (c_{ij})$, containing the cost of travelling between any pair of cities, is associated with this problem. Many exact and heuristic solution procedures for solving the problem have been suggested, and Bodin *et al.*[1] have provided a detailed survey of the literature of this problem.

One of the most interesting and applicable extensions of the TSP is the multi-travelling salesman problem (MTSP), where there is more than one salesman in the home city and every other city is to be visited by only one salesman with minimum total cost.

The TSP approaches have been the dominant methods for solving the MTSP since it has been proved that an MTSP with n cities and m salesmen can be transformed to a single TSP with $n + m - 1$ cities.[2]

Laporte and Nobert[3] suggested a cutting-plane algorithm for solving the MTSP without transforming it to a single TSP. Their motivation for this approach was that when an MTSP is transformed to a single TSP, the resulting problem is more arduous to solve than an ordinary TSP with the same number of cities, owing to the existence of $m - 1$ identical rows and columns in the cost matrix of the transformed problem. This fact has been confirmed to them by Christofides and Gavish through private communication.

Gavish *et al.*[4] developed a branch-and-bound algorithm by relaxing the degree constraints and computing a degree-constrained minimal spanning tree. A subgradient optimization procedure was used to update the Lagrangean multipliers.

In this paper a new formulation of the MTSP is introduced, after which the new solution method, which is based on the principle of savings, first introduced by Clarke and Wright,[5] is developed. Thereafter, the extensions of the formulation and the solution method for the MTSP with variable number of salesmen are described. Finally, the computational results and conclusion are presented.

THE NEW MTSP FORMULATION

An important observation about the MTSP with n cities and m salesmen is that, in any feasible solution, $2m$ cities have to be connected to the home city while the other cities are connected to each other. Hence, we can divide the links in any feasible solution into two sets: first, the set of links that connect cities other than the home city to each other; second, the set of links that connect cities to the home city. Therefore, if the home city is excluded, we can think of two types of connection between the cities to be visited: an X-type connection and a Y-type connection. Each of them determines to which of the two sets the connected pair of cities belongs. If the connection between two cities is of the X-type, then they are on some salesman's route; the connection between cities 2 and 3 in Figure 1 is of the X-type. However, if the connection between two cities is of the Y-type, then the first city will be the last city visited by a salesman before returning home, while the second city will be the first one to be visited by another salesman; in Figure 1, cities 3 and 4 are Y-type connected.

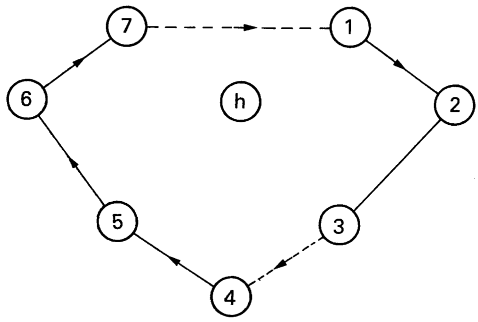

FIG. 1. *Feasible solution for n = 7 and m = 2.*

——— $X_{ij} = 1$; ········ $Y_{ij} = 1$.

Now we introduce some notations to prepare for the new formulation.

Let

$N = \{h, 1, 2, \ldots, n\}$, the set of cities to be visited by the salesmen, where city h denotes the home city;

$D = (d_{ij})$, the cost matrix, $d_{ii} = \infty$, for all i;

$$x_{ij} = \begin{cases} 1 & \text{if city } i \text{ is connected to city } j,\ i, j \neq h, \\ 0 & \text{otherwise;} \end{cases}$$

$$y_{ij} = \begin{cases} 1 & \text{if city } i \text{ is connected to city } h \text{ and city } h \text{ is connected to city } j,\ i, j \neq h, \\ 0 & \text{otherwise;} \end{cases}$$

m = number of salesmen.

Then the MTSP can be formulated as

$$\min f(X, Y) = \sum_{\substack{i, j \geqslant 1 \\ i \neq j}} d_{ij} x_{ij} + \sum_{\substack{i, j \geqslant 1 \\ i \neq j}} (d_{ih} + d_{hj}) y_{ij} \tag{1}$$

$$\text{s.t.} \sum_{\substack{j \geqslant 1 \\ j \neq i}} x_{ij} + y_{ij} = 1 \qquad i \geqslant 1 \tag{2}$$

$$\sum_{\substack{j \geqslant 1 \\ j \neq i}} x_{ji} + y_{ji} = 1 \qquad i \geqslant 1 \tag{3}$$

$$\sum_{\substack{i, j \geqslant 1 \\ i \neq j}} y_{ij} = m \tag{4}$$

$$(x_{ij} + y_{ij}) \in TB \tag{5}$$

$$x_{ij}, y_{ij} \in \{0, 1\} \qquad i, j \geqslant 1 \tag{6}$$

Constraints in (2), (3) and (6) define an assignment problem. Constraint (4) is a forcing constraint which implies that there are m links of the Y-type; that is, $2m$ cities are connected to the home city. Constraint (5) is a subtour breaking constraint. The set TB can be any restriction prohibiting subtours,[1] such as

$$TB = \{(x_{ij} + y_{ij}) : \sum_{i \in Q} \sum_{j \in Q} (x_{ij} + y_{ij}) \leqslant |Q| - 1$$

$$\text{for every non-empty subset } Q \text{ of } \{1, 2, 3, \ldots, n\}\}.$$

Any feasible X and Y form a travelling salesman route on $\{1, \ldots, n\}$, as shown in Figure 1. However, since the Y-type connections imply the end of one route and the start of another, X and Y determine a unique set of routes for all salesmen starting from the home city h, as shown in Figure 2.

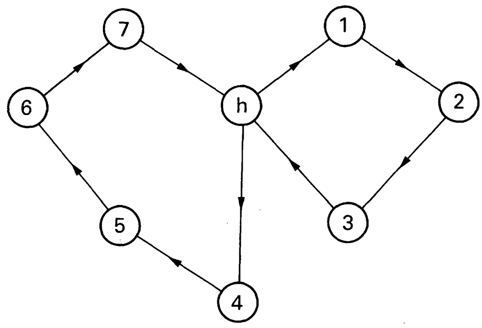

FIG. 2. *Actual solution for $n = 7$ and $m = 2$ corresponding to the feasible solution in Figure 1.*

THE SOLUTION METHOD

The above formulation represents a variation of an ordinary TSP on n nodes. A modification of the Little *et al.*[6] algorithm can be used to solve this problem. This approach will be addressed in a future work. In this paper a different procedure will be developed. The new algorithm is developed for a problem equivalent to the MTSP obtained from the above model through constraint modification. The new model will not include the Y-type links.

Constraints in (2) imply that

$$\sum_{j \geqslant 1,\, j \neq i} x_{ij} \leqslant 1 \qquad i \geqslant 1. \tag{2'}$$

Constraints in (3) imply that

$$\sum_{j \geqslant 1,\, j \neq i} x_{ji} \leqslant 1, \qquad i \geqslant 1. \tag{3'}$$

Moreover, (2) and (3) together imply that

$$\sum_{\substack{i,\, j \geqslant 1 \\ i \neq j}} x_{ij} + y_{ij} = n.$$

Therefore, constraint (4) can be replaced by

$$\sum_{\substack{i,\, j \geqslant 1 \\ i \neq j}} x_{ij} = n - m. \tag{4'}$$

Finally, it has been shown by the author[7] that

$$f(X, Y) = \sum_{i \geqslant 1} d_{ih} + \sum_{j \geqslant 1} d_{hj} - \sum_{\substack{i,\, j \geqslant 1 \\ i \neq j}} s_{ij} x_{ij}, \tag{1'}$$

where $s_{ij} = d_{ih} + d_{hj} - d_{ij}$.

Therefore, the MTSP formulation is equivalent to

$$\min - \sum_{\substack{i,\, j \geqslant 1 \\ i \neq j}} s_{ij} x_{ij} \tag{1'}$$

s.t.

$$(2'), (3'), (4')$$

$$(x_{ij}) \in TB \tag{5'}$$

$$x_{ij} \in \{0, 1\} \qquad i, j \geqslant 1, \tag{6'}$$

where TB is the same as the old one without y_{ij}.

The objective function can be rewritten as

$$\max \sum_{\substack{i, j \geqslant 1 \\ i \neq j}} s_{ij} x_{ij}.$$

Once the x_{ij} are determined, the set of salesmen's tours is uniquely determined. That is, once the X-type links are determined, the Y-type links will be any set of links that feasibly complete a salesman's tour on the set $\{1, \ldots, n\}$. Therefore, a solution of the above problem will automatically solve the MTSP.

Therefore, the MTSP with $n + 1$ cities and m salesmen has been transformed to a problem of finding a set of $(n - m)$ links with no circuits on the set $\{1, \ldots, n\}$ with maximum weight with respect to the cost matrix $S = (s_{ij})$.

It is worth noting that S is actually the matrix of saving which was first introduced by Clarke and Wright[5] as a list.

The new problem is surely an NP-complete problem because it is equivalent to the MTSP, which is NP-complete.

A greedy heuristic algorithm can be used to find an approximate solution for this problem. However, an exact branch-and-bound procedure for solving this problem is developed in this work.

Consider the problem of finding $n - m$ links with maximum weight and without circuits, $m < n$, in a network $G = (N, A)$ consisting of n nodes and described by the cost matrix $S = (s_{ij})$. The procedure for finding the best solution breaks down the set of all possible solutions into smaller and smaller subsets. This partitioning, which will eventually lead to the best solution, is guided by the bounds. The best solution is a subset that contains a single solution with cost greater than or equal to the upper bounds of all other subsets.

The construction of smaller subsets from the original set can be thought of as branching on a tree, where the subsets are represented by the nodes of the tree and the partitioning is represented by the branches. The branching and bounding rules will be explained in the algorithm. Subsets and the matrices describing them will be used interchangeably. The algorithm is described in the following steps.

Step 1

Let $V = S$, and set all diagonal elements of S to infinity. Let $Z_0 = -\infty$, where Z_0 is the cost of the best solution so far obtained. S describes the set of all possible solutions. To set an upper bound $U(S)$ on the cost of the best solution in the set described by S we define

$$t_i = \max_j \{s_{ij}\, ;\, s_{ij} \neq \infty\} \tag{7}$$

and let $t_{[i]}$s be a descending order arrangement of the t_is. Now

$$U(S) = \sum_{i=1}^{n-m} t_{[i]}. \tag{8}$$

Moreover, let

$$t = t_{[n-m+1]}.$$

Step 2

Find the link with the largest entry in the updated matrix to branch on, (p, q) say, according to the following rule:

$$s_{pq} = \max_{i, j} \{s_{ij},\, s_{ij} \neq \infty\}.$$

Step 3

Partition S into X, the set of all solutions including the link (p, q), and $\bar{X}$, the set of all solutions not including (p, q). Compute the upper bound on $\bar{X}$ using the following rule:

$$U(\bar{X}) = U(S) - s_{pq} + \max\{t, r\},$$

where

$$r = \max_{j \neq q} \{s_{pj};\ s_{pj} \neq \infty\},$$

and develop the matrix describing $\bar{X}$ from the matrix describing S by setting s_{pq} to infinity.

Step 4

Develop the matrix describing X by:

(a) setting $s_{pj} = -\infty$ for all $j, j \neq p$;
(b) setting $s_{iq} = \infty$ for all $i, i \neq p$;
(c) setting $s_{ij} = \infty$ if the link (i, j) will create a circuit.

Step 5

Compute $U(X)$ according to (8), noting that if t_i is negative infinity, then $t_i = v_{iq}$, where (i, q) is a link in the partial solution so far obtained. However, if m or fewer of the sets in (7) are empty, t_i is set less than the smallest finite element in the matrix for all such is. However, if more than m sets in (8) are empty, the subset under consideration does not have any feasible solution.

If there are fewer than $n - m$ links in the partial solution and $U(X)$ is greater than Z_0, let $S = X$ and go to step 2. If there are $n - m$ links, the matrix describing X has a unique solution; if $U(X) > Z_0$, set $Z_0 = U(X)$.

Step 6

If X contains a single solution or if $U(X) \leqslant Z_0$, then backtrack to the smallest subset with $U(X) > Z_0$. This subset is described by the matrix with the least number of finite rows. Call this subset S and go to step 2. If no such subset exists, the current best solution is optimal and the algorithm terminates.

It is worth noting that the above algorithm is biased toward complete solution; that is, the algorithm backtracks to the smallest subset whenever backtracking is needed. The new algorithm will be illustrated with the help of an example.

Example

Suppose we are interested in three links with maximum weight and with no circuits in a network described by a cost matrix S. Let

$$S = \begin{matrix} -\infty & 9 & 9 & 1 & 1 \\ 9 & -\infty & 8 & 0 & 1 \\ 9 & 8 & -\infty & 3 & 3 \\ 1 & 0 & 3 & -\infty & 3 \\ 1 & 1 & 3 & 3 & -\infty \end{matrix}$$

In this example $n = 5$ and $n - m = 3$; therefore, $m = 2$. Start the algorithm by letting $s_{ii} = \infty$ for all i and $Z_0 = -\infty$. Compute $U(S)$ as

$$U(S) = \sum_{i=1}^{3} t_{[i]} = 9 + 9 + 9 = 27.$$

Note that $t_{[1]} = 9$, $t_{[2]} = 9$, $t_{[3]} = 9$, $t_{[4]} = 3$, $t_{[5]} = 3$; therefore, $t = 3$. The link (p, q) in step 2 is found to be $(3, 1)$ since $s_{pq} = s_{31} = 9$. Therefore,

$$r = \max_{j \neq 1} \{s_{3j} : s_{3j} \neq \infty\} = 8.$$

The matrix describing $\bar{X}$ is obtained by setting s_{31} to infinity in the above matrix and

$$U(\bar{X}) = 27 - 9 + \max\{8, 3\} = 26,$$

which completes step 3.

The matrix describing X in step 4 is as follows:

$$\begin{array}{ccccc} + & 9 & + & 1 & 1 \\ + & + & 8 & 0 & 1 \\ - & - & + & - & - \\ + & 0 & 3 & + & 3 \\ + & 0 & 3 & + & + \end{array}$$

where $+$ is used instead of ∞ and $-$ instead of $-\infty$. Note that s_{13} is set to infinity to prevent circuits. The bound on X in step 5 is computed as

$$U(X) = 9 + 8 + 9 = 26.$$

Since there are fewer than three links in the partial solution and $U(X) > Z_0$, we let $S = X$, go back to step 2, and the algorithm repeats steps 2–5. The complete solution is shown on the tree in Figure 3, where the subset X is denoted by the link (i, j) and the subset $\bar{X}$ is denoted by the link $\overline{(i, j)}$. From Figure 3, the set of links with maximum weight and without loops is {(3, 1), (1, 2), (5, 4)} with cost equal to 21. Now we demonstrate by an example how to obtain the solution of the MTSP if the solution of the above problem is known.

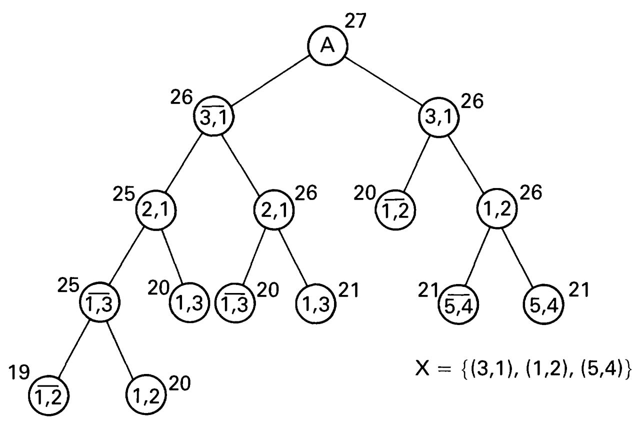

FIG. 3. *Complete solution of the example.*

Example

Suppose that the minimum cost routes for two salesmen stationed at city h is to be found on the network of cities described by the cost matrix

$$D = \begin{array}{c|cccccc} & h & 1 & 2 & 3 & 4 & 5 \\ \hline h & \infty & 6 & 7 & 6 & 2 & 4 \\ 1 & 6 & \infty & 4 & 3 & 7 & 9 \\ 2 & 7 & 4 & \infty & 5 & 9 & 10 \\ 3 & 6 & 3 & 5 & \infty & 5 & 7 \\ 4 & 2 & 7 & 9 & 5 & \infty & 3 \\ 5 & 4 & 9 & 10 & 7 & 3 & \infty \end{array}$$

It is easily seen that the matrix S obtained from this matrix is exactly the one in the previous example, where the set of links with maximum weight and no loops was found to be {(3, 1), (1, 2), (5, 4)}.

The links that complete a feasible salesman's tour on $\{1, 2, \ldots, 5\}$ are (2, 4) and (3, 5). Although the set of Y-type links are not unique, the routes that are determined by the X-type and Y-type links are unique. Therefore, the two required routes are

$$h - 3 - 1 - 2 - h$$

$$h - 5 - 4 - h,$$

and the cost of these routes is

$$C = \sum_{i=1}^{n} d_{ih} + \sum_{j=1}^{n} d_{hj} - \sum_{\substack{i,\, j \geqslant 1 \\ i \neq j}} s_{ij} x_{ij}$$

$$= 50 - 21 = 29.$$

EXTENSION TO OTHER VARIANTS

MTSP with variable number of salesmen

The above formulation can be modified to represent the problem when the number of salesman is not fixed beforehand and has to be determined by the solution. The only change needed is in constraint (4), which becomes

$$\sum_{i,\, j \geqslant 1} y_{ij} \geqslant 1.$$

This change will reflect the following change on constraint (4′) in the equivalent model:

$$\sum_{i,\, j \geqslant 1} x_{ij} \leqslant n - 1.$$

In order to handle the modified problem, the proposed algorithm is modified in the following manner.

In step 1, all negative entries of S, if there are any, are set to infinity, and the upper bound on S becomes

$$U(S) = \sum_{i=1}^{n} t_i - t,$$

where

$$t_i = \begin{cases} \max\limits_{j} \{s_{ij};\ s_{ij} \neq \infty\} & \text{if this is finite,} \\ 0 & \text{if } \{s_{ij};\ s_{ij} \neq \infty\} \text{ is empty,} \\ v_{iq} & \text{if } \max\limits_{j} \{s_{ij};\ s_{ij} \neq \infty\} \text{ is infinite, where } (i, q) \text{ is a link in the partial solution,} \end{cases}$$

and

$$t = \min \{t_1;\ t_i \neq 0\}.$$

Note that in step 5, any matrix will describe a feasible solution. However, a matrix will contain one solution if all its entries are either infinity or negative infinity.

TSP

The TSP is a special case of the MTSP with $m = 1$. Therefore the previous model and solution procedure can apply to the TSP without any modification if m is set equal to one.

COMPUTATIONAL RESULTS

The computational experience with the new algorithm was very limited owing to the fact that the only computing facility available for the author was a personal computer, and the process of trying more and larger problems is time-consuming. However, the conceptual implications of the new algorithm might compensate for this deficiency.

All computations were carried out on an IBM-compatible personal computer with 512 kb.

Two sets of randomly generated problems using the random-number generator program in Law[8] were used. The first set contains four symmetric problems of each of the sizes 10, 12, 14 and 16 cities, which were solved for one, two, three and four salesmen using the new algorithm. The first pass of the algorithm produces the Clarke and Wright heuristic solution if certain agreement on breaking ties in the savings list is adopted. Table 1 shows the average processing time in seconds for each group in this set.

TABLE 1

	No. of salesmen			
No. of cities	1	2	3	4
10	29.50	11.25	4.75	3.00
12	151.75	73.75	17.50	7.75
14	1903.00	646.00	149.50	65.00
16	5229.00	1579.25	826.25	436.25

The processing time presented in Table 1 compares favourably with the results presented in Waters and Brodie[9] for randomly generated problems solved using a VAX 8600 and a general mathematical package.

Table 2 shows the average number of iterations for the previous problems.

TABLE 2

	No. of salesmen			
No. of cities	1	2	3	4
10	88	33	11	7
12	367	92	42	18
14	3634	965	298	129
16	12699	4008	2110	1101

Table 3 shows the average processing time for four non-symmetric problems, generated randomly, of sizes 10, 12, 14 and 16 and solved for one, two, three and four salesmen.

TABLE 3

	No. of salesmen			
No. of cities	1	2	3	4
10	16.00	5.75	3.50	2.00
12	105.50	24.00	9.25	5.00
14	737.50	247.50	103.25	28.75
16	3245.75	969.25	265.25	77.75

In Table 4 the average number of iterations for the above problems is presented.

TABLE 4

	No. of salesmen			
No. of cities	1	2	3	4
10	48	17	8	4
12	253	55	20	10
14	1401	475	194	53
16	7839	2331	648	189

The first-pass processing time was between 1 and 4 seconds. The first-pass solution exceeded 1.5 of the optimal solution in 15 out of the 128 cases used, of which seven are in the symmetric cases. One important observation is that the first-pass solution for four salesmen exceeded 1.15 of the optimal solution in five cases out of the 32 cases solved, two of which are in the symmetric cases.

In most of the cases solved, the optimal solution was found in less then 1/4 of the time needed to solve the problem; the rest of the time was used to prove the optimality of the solution. Therefore, the new algorithm can be used as a good heuristic if one decides to stop it after a certain number of iterations.

The modification of the algorithm to the MTSP with variable number of salesmen was tried on the 16 problems of sizes 10, 12, 14 and 16 cities that were used in the non-symmetric case. The computational experience is not reported because it was almost the same as the results for one salesman in the non-symmetric case.

CONCLUSION

In conclusion, this paper presents a new mathematical formulation for the MTSP which can be easily extended to the version of the problem with a variable number of salesmen. This new formulation is transformed to an equivalent model, the solution of which will automatically provide a solution for the MTSP.

Two important advantages of the new procedure are to be emphasized. First, as the number of salesmen increases for a fixed number of cities, the time required to solve the problem decreases markedly. This is achieved without increasing the storage requirements, which are independent of the number of salesmen. This advantage is very clear in the computational results, which indicate that the algorithm becomes more useful as the number of salesmen increases. Second, the first pass of the new procedure produces the heuristic solution of Clarke and Wright, which has never been thought of as an optimization-based heuristic.

REFERENCES

1. L. Bodin, B. Golden, A. Assad and M. Ball (1983) Routing and scheduling of vehicles and crews. *Comput. Opns Res.* **10**, 82–96.
2. M. Bellmore and S. Hong (1974) Transformation of multi-salesmen problem to standard traveling salesman problem. *Journal of ACM* **21**, 500–504.
3. G. Laporte and Y. Nobert (1980) Cutting planes algorithm for the M-salesmen problem. *J. Opl Res. Soc.* **31**, 1017–1023.
4. B. Gavish and K. Srikanth (1986) An optimal solution method for large-scale multiple traveling salesmen problems. *Opns Res.* **34**, 698–717.
5. G. Clarke and J. Wright (1964) Scheduling of vehicles from a central depot to a number of delivery points. *Opns Res.* **12**, 568–581.
6. J. Little, K. Murty, D. Sweeny and C. Karel (1963) An algorithm for the traveling salesman problem. *Opns Res.* **11**, 972–989.
7. A. O. Husban (1985) Balancing routes in a class of vehicle routeing problems. Ph.D Dissertation, Rensselaer Polytechnic Institute.
8. A. M. Law and W. D. Kelton (1982) *Simulation Modeling and Analysis*. McGraw-Hill, New York.
9. C. D. J. Waters and G. P. Brodie (1987) Realistic sizes for routeing problems. *J. Opl Res. Soc.* **38**, 565–566.