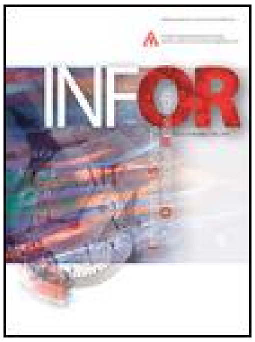


INFOR: Information Systems and Operational Research

ISSN: 0315-5986 (Print) 1916-0615 (Online) Journal homepage: http://www.tandfonline.com/loi/tinf20

Computational Evaluation Of A Transformation Procedure For The Symmetric Generalized Traveling Salesman Problem

Gilbert Laporte & Frédéric Semet

To cite this article: Gilbert Laporte & Frédéric Semet (1999) Computational Evaluation Of A Transformation Procedure For The Symmetric Generalized Traveling Salesman Problem, INFOR: Information Systems and Operational Research, 37:2, 114-120, DOI: 10.1080/03155986.1999.11732374

To link to this article: http://dx.doi.org/10.1080/03155986.1999.11732374

Published online: 25 May 2016.

COMPUTATIONAL EVALUATION OF A TRANSFORMATION PROCEDURE FOR THE SYMMETRIC GENERALIZED TRAVELING SALESMAN PROBLEM[1]

GILBERT LAPORTE

Centre de recherche sur les transports, Université de Montréal
C.P. 6128, succ. Centre-ville, Montreal PQ H3C 3J7, Canada

FRÉDÉRIC SEMET

LIMAV, Université de Valenciennes, 59304 Valenciennes Cedex, France

ABSTRACT

This note proposes a transformation of the Generalized Traveling Salesman Problem on an undirected graph into an equivalent Traveling Salesman Problem and provides an assessment of its computational efficiency.

Keywords: Generalized Traveling Salesman Problem.

RÉSUMÉ

Dans cette note on propose une transformation du problème du voyageur de commerce généralisé sur un graphe non dirigé en un problème du voyageur de commerce équivalent et on étudie sa performance numérique.

Mots clefs : Problème du voyageur de commerce généralisé.

1. INTRODUCTION

The *Generalized Traveling Salesman Problem* (GTSP) consists of determining a least cost tour passing through each of several clusters of vertices of a graph. Several versions of the problem exist. Here we consider the GTSP defined on an undirected graph $G = (V, E)$, where $V = \{v_1, \dots v_n\}$ is the vertex set and $E = \{(v_i, v_j) : v_i, v_j \in V, i < j\}$ is the edge set. In what follows, (v_i, v_j) must be interpreted as (v_j, v_i) whenever $j < i$. The set V is partitioned into m clusters $V_1, \dots, V_m$ and a non-negative cost c_{ij} is associated with each edge (v_i, v_j). The version of the GTSP treated here is to determine a least cost Hamiltonian cycle containing exactly one vertex from each cluster.

The GTSP was introduced in the late sixties and early seventies (Henry-Labordere [3], Srivastava *et al.* [13], Saksena [12]) in the context of computer record balancing and of visit sequencing through welfare agencies. Several other interesting applications exist in areas as diverse as location-routing, postal delivery and the layout of tracks for automated guided vehicles in factories (Laporte, Asef-Vaziri and Sriskandarajah [5]). Recently, Laporte [4] as well as Dror and Langevin [1] have shown how several classes of arc routing problems can be formulated and solved as GTSPs.

Early algorithms for the GTSP were based on dynamic programming [6, 13] but these can only handle modest size instances. Better algorithms are now available. Noon and Bean [9] have developed a branch and bound procedure combined with Lagrangean relaxation for the asymmetric case. For symmetric problems, branch and cut is probably the best method available (Laporte and Nobert [6], Fischetti, Salazar and Toth [2]).

In a recent paper, Noon and Bean [10] have shown how asymmetric GTSPs can be transformed into asymmetric *Traveling Salesman Problems* (TSPs) over the same number of vertices. Lien, Ma and Wah [7] have proposed a transformation of the

[1]Recd. May 96; Revd. June 97, Feb. 98.

symmetric GTSP into and equivalent asymmetric TSP with roughly three times as many vertices. Volgenant [14] also suggested a transformation of the GTSP into a TSP which appears to be similar to ours.

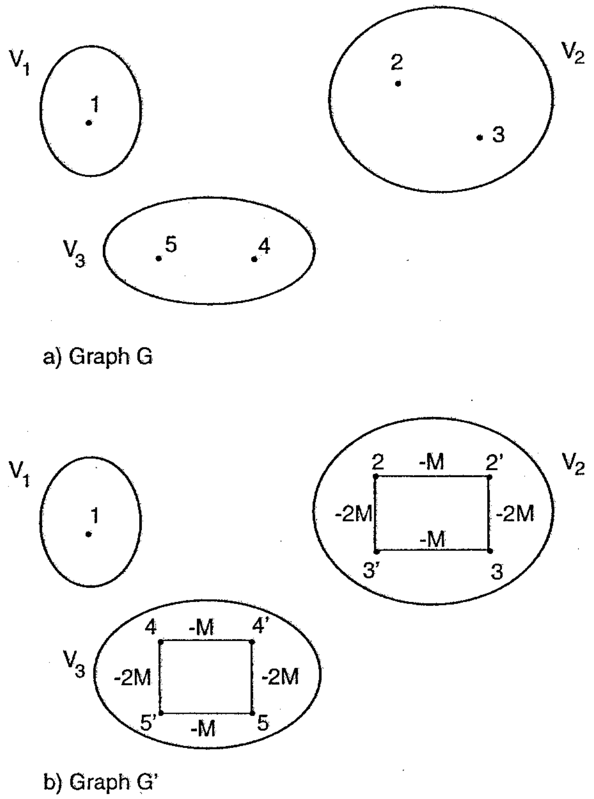


Figure 1 (a) and (b) Transformation of G into G'. Optimal TSP and GTSP Solutions.

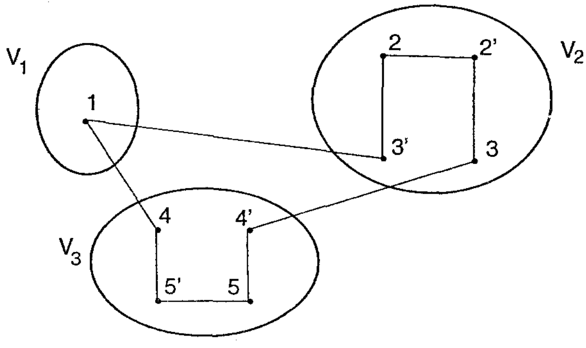
It is not obvious whether such transformations are beneficial. On the positive side, several good exact and approximate algorithms already exist for the TSP and could therefore be used as "black boxes" to solve transformed GTSP instances. This is the approach recently taken by Laporte [4] who transformed several classes of arc routing problems into GTSPs and then into TSPs. On the negative side, such transformations typically yield highly degenerate TSPs which may be much harder to solve than TSP instances generated on random graphs. To our knowledge, this computational question has never been investigated in the past. Volgenant [14] points to the "practical limitation of [his] transformation".

In this note we first present our transformation of the symmetric GTSP into a symmetric TSP doubling the number of vertices. We then report the results of some computational experiments relative to this transformation.

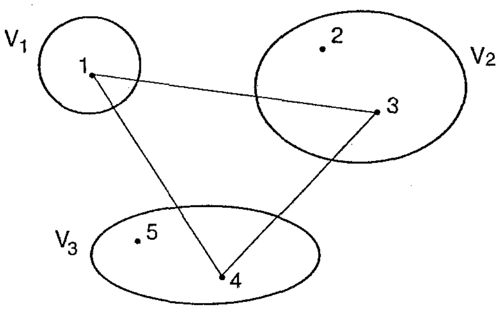
2. TRANSFORMATION

The first step of the transformation consists of creating a copy $v_{i'}$ of each vertex v_i. If $v_i \in V_k$ and $|V_k| > 1$, set $V_k := V_k \cup \{v_{i'}\}$. If $v_{i'}$ and $v_{j'}$ are two vertices belonging to different clusters, create an edge between $v_{i'}$ and $v_{j'}$ having a cost $c_{i'j'} = c_{ij}$. Consider in turn all clusters V_k with $|V_k| > 1$. Set $t(k) := |V_k|/2$ and sequence the vertices of V_k

in any order $v_{i_1}, v_{i'_1}, v_{i_2}, v_{i'_2}, \ldots, v_{i_{t(k)}}, v_{i'_{t(k)}}$ that places a vertex next to its copy. Then create the edges $(v_{i_1}, v_{i'_1})$, $(v_{i'_1}, v_{i_2}), \ldots, (v_{i_{t(k)}}, v_{i'_{t(k)}})$, $(v_{i'_{t(k)}}, v_{i_1})$. Let (v_i, v_j) be one of these edges. If v_i and v_j are copies of the same vertex, set $c_{ij} := -M$; otherwise, set $c_{ij} := -2M$, where M is a large constant exceeding the sum of the n largest edge costs in G. Let $G' = (V', E')$ be the graph defined through this transformation.



c) Optimal TSP solution on Graph G'



d) Optimal GTSP solution on G

Figure 1 (c) and (d) Transformation of G into G'. Optimal TSP and GTSP Solutions.

Proposition 1

The TSP defined on G' yields an optimal solution for the GTSP defined on G.

Proof

The thesis follows from the fact that the optimal solutions to the two problems have inter-cluster edges having the same total cost. Indeed, whenever $|V_k| > 1$, any optimal TSP solution will contain the $2t_k - 1$ cheapest edges in cluster V_k, *i.e.*, t_k edges of cost $-2M$ and $t_k - 1$ edges of cost $-M$, since M is larger than the sum of the costs of any n edges of E. It follows that in the TSP solution on G', there will be exactly two edges linking each cluster V_k to other clusters, and these two edges will be incident to some vertex v_i and its copy $v_{i'}$. The GTSP solution is immediately obtained from the TSP solution by conserving only the inter-cluster edges, and coalescing each vertex with its copy. •

The transformation is illustrated in Figure 1. Here $V = \{1, 2, 3, 4, 5\}$, $V_1 = \{1\}$, $V_2 = \{2, 3\}$, $V_3 = \{4, 5\}$. $E = \{(1, 2), (1, 3), (1, 4), (1, 5), (2, 4), (2, 5), (3, 4), (3, 5)\}$. For simplicity, these edges are not shown on the figure. Assume the GTSP solution is (1,3,4,1).

m'	p	q	SUCC	SUB-TOURS	COMBS	CL-TREES	LB/OPT (G)	UB/OPT (G)
0	0	100	5	53	82	9	–	–
2	25	0	5	1	0	0	1.0000	1.0002
2	20	10	5	841	5826	162	0.9950	1.0006
3	10	20	2	2845	11017	523	0.9957	1.0002
2	15	20	5	2404	13010	529	0.9953	1.0016
10	2	30	3	4200	10219	210	0.9937	1.0024
5	4	30	3	4249	10542	342	0.9962	1.0015
4	5	30	4	3712	12754	443	0.9961	1.0018
2	10	30	5	2040	6438	414	0.9950	1.0013

Table 1a: Summary of computational results for random instances

m'	p	q	LB/OPT (T)	UB/OPT (T)	NODES	SECONDS
0	0	100	0.9995	1.0042	1	8
2	25	0	1.0000	1.0001	1	22
2	20	10	0.9984	1.0002	154	1557
3	10	20	0.9989	1.0000	301	1492
2	15	20	0.9990	1.0003	314	1506
10	2	30	0.9993	1.0003	289	3482
5	4	30	0.9994	1.0002	449	2512
4	5	30	0.9994	1.0003	381	1726
2	10	30	0.9994	1.0001	234	807

Table 1b: Summary of computational results for random instances

3. COMPUTATIONAL RESULTS

To assess the computational efficiency of the proposed approach, we solved a number of 50-vertex instances of the GTSP. Tests were carried out on two types of instances: random and clustered. In random instances, $n = 50$ points were first generated in the $[0, 100]^2$ square according to a continous uniform distribution. Then m' clusters of cardinality p and q clusters of cardinality 1 (with $m'p + q = n$ and $q + m' = m$) were created by arbitrarily assigning each point to a cluster. In clustered instances, we first divided the $[0, 100]^2$ square into m' equal or approximately equal rectangles. We then associated a cluster with each rectangle and generated within each cluster p vertices according to a continuous uniform distribution over the rectangle. We then created $q = n - m'p = m - m'$ additional clusters of cardinality 1 by generating vertices randomly in the $[0, 100]^2$ square. For both instance types and for each combination of m', p and q considered, we generated five instances. These were first transformed into a TSP and then solved exactly by means of the Padberg and Rinaldi [11] code. This code also incorporates a heuristic based on the Lin-Kernighan [8] edge interchange scheme. Each instance was solved on a Sun Ultra Sparc 1 Station, with a time limit of two hours. For each case we computed the TSP objective using the transformed costs, and the corresponding GTSP objective using the original costs.

m'	p	q	SUCC	SUB-TOURS	COMBS	CL-TREES	LB/OPT (G)	UB/OPT (G)
0	0	100	5	53	82	9	-	-
10	5	0	4	1096	4261	80	0.7707	1.0305
2	25	0	5	1	0	0	1.0000	1.0000
20	2	10	3	3133	10708	133	0.9661	1.0161
10	4	10	2	7788	23798	488	0.8857	1.0087
4	10	10	4	2544	17429	424	0.8367	1.0295
2	20	10	5	664	5793	144	0.8588	1.0142
15	2	20	4	2352	6659	94	0.9617	1.0236
10	3	20	1	2256	5181	98	0.9280	1.0140
3	10	20	4	4369	20446	821	0.8923	1.0341
2	15	20	5	1611	7690	279	0.9308	1.0041
10	2	30	5	542	1339	17	0.9592	1.0049
5	4	30	4	1792	4891	145	0.9431	1.0055
4	5	30	5	1294	3131	100	0.9348	1.0074
2	10	30	5	2431	6354	334	0.9520	1.0049

Table 2a: Summary of computational results for clustered instances

m'	p	q	LB/OPT (T)	UB/OPT (T)	NODES	SECONDS
0	0	100	0.9995	1.0042	1	8
10	5	0	0.9985	1.0002	201	2858
2	25	0	1.0000	1.0000	0	6
20	2	10	0.9996	1.0002	211	4019
10	4	10	0.9990	1.0001	458	6902
4	10	10	0.9985	1.0003	364	2843
2	20	10	0.9987	1.0001	89	574
15	2	20	0.9996	1.0003	148	2341
10	3	20	0.9992	1.0002	196	1187
3	10	20	0.9989	1.0004	486	2531
2	15	20	0.9992	1.0000	200	973
10	2	30	0.9996	1.0001	60	269
5	4	30	0.9994	1.0001	170	739
4	5	30	0.9993	1.0001	111	440
2	10	30	0.9995	1.0001	263	896

Table 2b: Summary of computational results for clustered instances

Our computational results are presented in Tables 1 and 2. The table headings are as follows:

m' number of clusters of cardinality greater than 1;
p number of vertices in each of these clusters;
q number of clusters of cardinality 1;
SUCC: number of instances, out of five, thant could be solved to optimality within two hours;

SUBTOURS: number of subtour elemeination constraints generated in the Padberg-Rinaldi algorithm;
COMBS: number of comb inequalities;
CL-TREES: number of clique-tree inequalities;
LB/OPT: Objective value at the root of the search tree divided by the optimal solution value; the letter "G" refers to the GTSP and the letter "T" refers to the TSP;
UB/OPT: heuristic solution value divided by the optimal solution value;
NODES: number of nodes in the search tree;
SECONDS: CPU time in seconds.

All statistics represent average values over the number of successful instances.

The first line of Tables 1 and 2 corresponds to a 100-vertex TSP. It is reported for comparison purposes. The number of vertices in the TSPs resulting from the GTSP transformations is equal to $q+2m'p$ and this value is equal to $100-q$ in our experiments since $n=50$. Thus the number of vertices varies between 50 and 100 in the transformed test problems since $0 \leq q \leq n$. Still, the pure 100-vertex TSP is much easier to solve than the transformed GTSPs, sometimes by three orders of magnitude. As a rule, clustered instances are easier to solve than random instances. For both cases, problems become easier when q increases. Also, for a given q, problem difficulty increases with p, except perhaps for $q=30$. The two columns LB/OPT(T) and UB/OPT (T) indicate that the Padberg-Rinaldi method performs quite well on the transformed instances. However, since M is large with respect to the other costs, the branching process has difficulty distinguishing between two solutions that are close to each other with respect to the TSP objective, but quite different with respect to the GTSP objective. This is confirmed by looking at the LB/OPT (G) and UB/OPT (G) columns. Even if the corresponding ratios are good for the TSP, they can be quite poor for the GTSP. While the exact TSP algorithm does not always perform well, the heuristic is generally quite reliable. As a rule, it provides GTSP solutions close to the optimum. In this context, the type of transformation proposed would appear to be valuable.

ACKNOWLEDGEMENTS

This research was partly funded by the Natural Sciences and Engineering Research Council of Canada under grants OPG0039682 and OPG0005392. This support is gratefully acknowledged. Thanks are also due to Manfred Padberg and Giovanni Rinaldi for having made their TSP code available to the authors. Finally, we thank the anonymous reviewers for their valuable comments.

REFERENCES

1. M. Dror and A. Langevin (1997). "A Generalized Traveling Salesman Approach to the Directed Clustered Rural Postman Problem". *Transportation Science* 31, 187-192.
2. M. Fischetti, J.J. Salazar and P. Toth (1997). "A Branch-and-Cut Algorithm for the Symmetric Generalized Traveling Salesman Problem", *Operations Research* 45, 378-394.
3. A.L. Henry-Labordere (1969). "The Record Balancing Problem: A Dynamic Programming Solution of a Generalized Travelling Salesman Problem".*RIRO* B-2, 43-9.
4. G. Laporte (1997). "Modeling and Solving Several Classes of Arc Routing Problems as Traveling Salesman Problems". *Computers & Operations Research,* 24, 1057-1061.
5. G. Laporte, A. Asef-Vaziri and C. Sriskandarajah (1996). "Some Applications of the Generalized Traveling Salesman Problem". *Journal of the Operational Research Society* 47, 1461-1467.
6. G. Laporte and Y. Nobert (1983). "Generalized Travelling Salesman Problem through n Sets of Nodes: An Integer Programming Approach". *INFOR* 21, 61-75.

7. Y.N. Lien, E. Ma and B.W.-S. Wah (1993). "Transformation of the Generalized Traveling-Salesman Problem into the Standard Traveling-Salesman Problem". *Information Sciences* 74, 177-189.
8. S. Lin and B.W. Kernighan (1973). "An Effective Heuristic Algorithm for the Traveling Salesman Problem". *Operations Research* 21, 498-516.
9. C.E. Noon and J.C. Bean (1991). "A Lagrangian Based Approach for the Asymmetric Generalized Traveling Salesman Problem". *Operations Research* 39, 623-32.
10. C.E. Noon and J.C. Bean (1993). "An Efficient Transformation of the Generalized Traveling Salesman Problem". *INFOR* 31, 39-44.
11. M.W. Padberg and G. Rinaldi (1991). "A Branch-and-Cut Algorithm for the Resolution of Large-Scale Symmetric Traveling Salesman Problems". *SIAM Review* 33, 6-100.
12. J.P. Saksena (1970). "Mathematical Model for Scheduling Clients through Welfare Agencies". *CORS Journal* 8, 185-200.
13. S.S. Srivastava, S. Kumar, R.C. Garg and P. Sen (1969). "Generalized Travelling Salesman Problem through Sets of Nodes". *CORS Journal* 7, 97-101.
14. A. Volgenant (1991). "Transforming the Generalized Traveling Salesman Problem through n Sets of Nodes into a Standard Problem". Presented at the 14th International Symposium on Mathematical Programming, Amsterdam.

Gilbert Laporte obtained his Ph.D. in Operations Research at the London School of Economics in 1975. He is Professor of Operations Research at the École des Hautes Études Commerciales, de Montréal, and adjunct Professor at the University of Alberta. He is also a member of the Centre for Research on Transportation (serving as director from 1987 to 1991) and Fellow of the Center for Management of Operations and Logistics, University of Texas at Austin. He has authored or coauthored several books, as well as more than 150 scientific articles in combinatorial optimization, mostly in the area of vehicle routing, location and timetabling. He is the Editor of *Transportation Science*, and associate editor or editorial board member of several other operations research journals. He has received many scientific awards including the Pergamon Prize (United Kingdom) in 1987, the 1994 Merit Award of the Canadian Operational Research Society, and the CORS Practice Prize on two occasions. In 1998 he became a member of the Royal Society of Canada.

Frédéric Semet is asociate professor at the University of Valenciennes, France. He received his Ph.D. in the Department of Mathematics, École Polytechnique Fédérale de Lausanne. He is a member of the Groupe de recherche interdisciplinaire en santé and of the Centre for research on transportation. His research interests include vehicle routing, workforce scheduling and classification. He has been involved in a variety of projects for distribution companies.