Solving Arc Routing Problems Using the Lin-Kernighan-Helsgaun Algorithm

Keld Helsgaun
E-mail: keld@ruc.dk

Computer Science
Roskilde University
DK-4000 Roskilde, Denmark

Abstract

It is well known that many arc routing problems can be transformed into the Equality Generalized Traveling Salesman Problem (E-GTSP), which in turn can be transformed into a standard Asymmetric Traveling Salesman Problem (TSP). This opens up the possibility of solving arc routing problems using existing solvers for TSP. This paper evaluates the performance of the state-of-the art TSP solver Lin-Kernighan-Helsgaun (LKH) on a broad class of transformed arc routing instances. It is shown that LKH makes it possible to find solutions of good quality to large-scale undirected, mixed, and windy postman and general routing problem instances.

Keywords: Arc routing problems, Equality generalized traveling salesman problem, E-GTSP, Traveling salesman problem, TSP

Mathematics Subject Classification: 90C27, 90C35, 90C59

1. Introduction

The goal of arc routing problems (ARPs) is to determine a minimum cost closed walk passing through some arcs and edges of a graph. Formally, ARPs are defined on a graph $G = (V, A, E)$ where $V = \{v_1, ..., v_n\}$ is a set of vertices, A is a set of (directed) arcs $a_{ij}\, (i \neq j)$, and E is a set of (undirected) edges $e_{ij}\, (i < j)$. Non-negative costs c_{ij} and d_{ij} are associated with arcs a_{ij} and with edges e_{ij}, respectively. It is not necessary to traverse all arcs or edges. Denote by A_R and E_R the subsets of required arcs and edges, respectively. The aim is to determine a least cost closed walk on G including all required arcs and edges at least once.

If the walk also has to pass through a certain subset of required vertices, $V_R \subseteq V$, we have the general routing problem (GRP).

Depending on problem properties, some well-known classes of routing problems can be obtained from this definition. In this paper the following classes will be tackled:

- The Mixed Chinese Postman Problem (MCPP): $A_R = A \neq \emptyset$, $E_R = E \neq \emptyset$, $d_{ij} = d_{ji}$ for all i, j, $V_R = \emptyset$.
- The Windy Postman Problem (WPP): $A = \emptyset$, $E_R = E \neq \emptyset$, $d_{ij} \neq d_{ji}$ for at least one edge e_{ij}, $V_R = \emptyset$.

- The Undirected, Mixed and Windy Rural Postman Problems (URPP, MRPP, WRPP), which are defined similarly, except that now $A_R \subset A$ or $E_R \subset E$.
- The General Routing Problem (GRP): $A = \emptyset$, $E_R = E \neq \emptyset$, $d_{ij} = d_{ji}$ for all i, j, $V_R \neq \emptyset$.
- The Mixed General Routing Problem (MGRP): $A = \emptyset$, $E_R = E \neq \emptyset$, $d_{ij} = d_{ji}$ for all i, j, $V_R \neq \emptyset$.
- The Windy General Routing Problem (WGRP): $A = \emptyset$, $E_R = E \neq \emptyset$, $d_{ij} \neq d_{ji}$ for at least one edge e_{ij}, $V_R \neq \emptyset$.

All these problems can easily be transformed into E-GTSP [1]. The transformed problem is defined on a graph $H = (W, B)$. In this graph W consists of one vertex w_{ij} for each required arc v_{ij} of G, one vertex w_{ii} for each required vertex v_i, and two vertices w_{ij} and w_{ji} for each required edge e_{ij} (one for each of the corresponding opposite arcs, only one of which is required). B is the set of all arcs linking two vertices of W. Each vertex pair (w_{ki}, w_{lj}) in the transformed problem defines an arc of W with a cost equal to $s_{il} + c_{lj}$, where s_{il} denotes the cost of a shortest path from v_i to v_l on G.

We have thus transformed the original arc routing problem into the Equality Generalized Traveling Salesman Problem (E-GTSP), where each cluster consists of either one or two vertices. Clusters consisting of two vertices correspond to required edges in the original problem, whereas single vertex clusters correspond to required arcs and required vertices in the original problem.

A recent paper [2] has described GLKH, an effective solver for E-GTSP based on the Lin-Kernighan-Helsgaun algorithm, LKH [3]. GLKH will be used in the following computational study.

2. Computational Results

The program was coded in C and run under Linux on an iMac 3.4 GHz Intel Core i7 with 32 GB RAM. Version 1.0 of GLKH was used. The program uses only one of the computer's four CPU cores.

Coberán et al. have provided a large library of test instances for arc routing problems [4]. The library includes 1042 instances of URPP, GRP, MCPP, MRPP, MGRP, WPP, WRPP and WGRP. All these instances have be transformed into E-GTSP and then solved by GLKH using the following non-default parameter settings:

ASCENT_CANDIDATES = 500
INITIAL_PERIOD = 1000
MAX_CANDIDATES = 12
MAX_TRIALS = 1000
OPTIMUM = *<best known cost>*
RUNS = 1

Table 4 summarizes the computational results. The table contains average values over all considered instances of the respective type. The columns '*n*' and '*m*' report the average number of vertices and clusters in the E-GTSP instances.

The following observations can be made:

- The solution quality is good for all instances. Optima are found for about half of the instances and the average deviation from the optimal solution is less than 3%.

- The instances in MCPP and WPP are the most difficult for GLKH. Other parameter settings might lead to a better solution quality. However, this will probably be at the expense of unacceptable running times. For these large instances, GLKH cannot compete with the highly sophisticated exact algorithm of Corberán et al. [5].

- Tests similar to those reported in Table 1 have been conducted by Drexl [6, p. 10]. Using Gutin and Karapetyan's heuristic E-GTSP solver GK [7], he found that GK performed acceptable for instances with up to about 200 clusters. However, for instances with more than 500 clusters, the gap to the optimal solutions usually exceeded 10%. As seen, GLKH performs better than GK for instances with many clusters.

Currently, optima are known for 998 out of the 1042 instances. It may be mentioned, that until now GLKH has been able to find new best upper bounds for 22 of the remaining 44 instances:

MCPP MA3067	6,529,588	WPP WB3061:	178,684
MCCP MB2052:	125,566	WPP WB3062:	177,765
MCPP MB3052:	151,284	WRPP C422:	21,181
MCPP MB3065:	201,187	WRPP D322:	23,784
MGRP GD422:	32,057	WRPP D421:	24,539
MGRP GD425:	37,581	WRPP D422:	23,943
MGRP GD522:	34,482	WGRP GB321:	20,549
MGRP GD525:	40,077	WGRP GB322:	20,328
WPP WA3065:	4,500,431	WGRP GB421:	20,774
WPP WB3035:	83,596	WGRP GB422:	20,452
WPP WB3055:	133,501	WGRP GB622:	24,102

Instance classes	# of inst.	$\lvert V \rvert$	$\lvert A \rvert + \lvert E \rvert$	$\lvert A_R \rvert + \lvert E_R \rvert$	n	m	Error (%)	Opt. (%)	Time (s)
URPP: UR500	12	446	1129	616	1232	616	0.32	41.7	141.1
URPP: UR750	12	666	1698	907	1813	907	0.40	25.0	231.4
URPP: UR1000	12	886	2290	1215	2430	1215	0.50	25.0	358.4
GRP: Alba	15	116	174	86	196	110	0.00	100.0	0.2
GRP: Madr	15	196	316	158	347	189	0.00	100.0	1.5
GRP: GRP	10	116	174	75	178	102	0.00	100.0	0.2
MCPP: MA05	12	500	1158	1158	1773	1158	0.56	8.3	1028.9
MCPP: MB05	12	500	1210	1210	1836	1210	0.24	25.0	740.2
MCPP: MA10	12	1000	2319	2319	3555	2319	0.86	0.0	2958.7
MCPP: MB10	12	1000	2442	2442	3702	2442	0.61	16.7	2283.6
MCPP: MA15	12	1500	3479	3479	5330	3479	1.06	0.0	4686.8
MCPP: MB15	12	1500	3631	3631	5511	3631	0.57	8.3	3726.4
MCPP: MA20	12	2000	4645	4645	7108	4645	1.09	0.0	7031.0
MCPP: MB20	12	2000	4829	4829	7329	4829	0.58	0.0	5226.9
MCPP: MA30	12	3000	6959	6959	10664	6959	1.20	0.0	12627.1
MCPP: MB30	12	3000	7131	7131	10877	7131	0.75	0.0	8511.3
MRPP: RB	18	449	1134	610	1376	610	0.02	72.2	141.0
MRPP: RD	18	900	2315	1230	2759	1230	0.08	33.3	530.9
MGRP: Alba	25	116	174	88	177	118	0.00	100.0	0.2
MGRP: Alda	31	214	351	168	324	217	0.00	93.5	7.1
MGRP: Madr	25	196	316	158	301	205	0.00	100.0	1.7
MGRP: GB	18	500	1218	610	980	661	0.03	83.3	110.5
MGRP: GD	18	1000	2450	1230	1958	1330	0.08	44.4	558.3
WPP: WA05	12	500	1160	1160	2321	1160	2.00	0.0	489.8
WPP: WB05	12	500	1213	1213	2426	1213	1.28	0.0	386.3
WPP: WA10	12	1000	2317	2317	4634	2317	2.44	0.0	1230.2
WPP: WB10	12	1000	2434	2434	4868	2434	2.15	0.0	899.4
WPP: WA15	12	1500	3493	3493	6986	3493	2.68	0.0	2229.5
WPP: WB15	12	1500	3655	3655	7309	3655	2.17	0.0	1678.3
WPP: WA20	12	2000	4645	4645	9289	4645	2.86	0.0	3412.7
WPP: WB20	12	2000	4826	4826	9652	4826	2.36	0.0	2561.3
WPP: WA30	12	3000	6961	6961	13922	6961	2.97	0.0	7375.6
WPP: WB30	12	3000	7141	7141	14282	7141	2.35	0.0	5060.1
WRPP: A100	72	116	174	102	204	102	0.01	94.4	1.1
WRPP: A500	27	401	1268	481	963	481	1.13	3.7	134.8
WRPP: A1000	27	848	2522	1149	2297	1149	1.71	0.0	505.4
WRPP: B	24	446	1132	610	1220	610	0.28	12.5	125.2
WRPP: C	24	673	1706	918	1837	918	0.40	8.3	170.8
WRPP: D	24	895	2287	1222	2443	1222	0.58	12.5	251.1
WRPP: M	72	196	316	187	374	187	0.04	68.1	10.8
WRPP: HD	54	86	173	85	170	85	0.01	96.3	0.9
WRPP: HG	54	83	149	77	154	77	0.00	100.0	0.6
WRPP: P	144	25	59	28	56	28	0.00	100.0	0.0
WGRP: A	27	500	1135	575	1223	648	1.18	0.0	191.1
WGRP: G	24	500	1210	599	1255	656	0.30	12.5	124.8

Table 1 *Results for arc routing problems*.

3. Conclusion

The computational results show that LKH makes it possible to find solutions of good quality to large-scale undirected, mixed, and windy postman and general routing problem instances.

The developed software is free of charge for academic and non-commercial use and can be downloaded in source code together with test instances via
http://www.ruc.dk/~keld/research/GLKH/.

References

1. Blais, M., Laporte, G.: Exact Solution of the Generalized Routing Problem through Graph Transformations. J. Oper. Res. Soc., 54(8):906-910 (2003)

2. Helsgaun, K.: Solving the Equality Generalized Traveling Salesman Problem Using the Lin-Kernighan-Helsgaun Algorithm. Computer Science Report #141, Roskilde University (2014)

3. Helsgaun, K.: An Effective Implementation of the Lin-Kernighan Traveling Salesman Heuristic. Eur. J. Oper. Res., 126(1):106-130 (2000)

4. Corberán, Á., Plana I., Sanchis, J.M.: Arc Routing Problems: Data Instances. http://www.uv.es/corberan/instancias.htm

5. Corberán, A., Oswald, M., Plana I., Reinelt, G., Sanchis, J.M.: New results on the Windy Postman Problem, Math. Program., Ser. A 132:309–332 (2012)

6. Drexl, M.: On the generalized directed rural postman problem. J. Oper. Res. Soc., doi:10.1057/jors.2013.60 (2013)

7. Gutin, G., Karapetyan, D.: A memetic algorithm for the generalized traveling salesman problem. Nat. Comput., 9(1):47-60 (2010)