Undirected Capacitated Arc Routing Problems in Debris Collection Operation After Disaster

Andie PRAMUDITA[1*], Eiichi TANIGUCHI[2] and Ali G. QURESHI[3]

[1]Dept. of Urban Management, Kyoto University
(C1-2-334 Bldg., Kyoto daigaku-Katsura, Nishikyo-ku, Kyoto 615-8530, Japan)
E-mail: apramudita@kiban.kuciv.kyoto-u.ac.jp
[2]Professor, Dept. of Urban Management, Kyoto University
(C1-2-335 Bldg., Kyoto daigaku-Katsura, Nishikyo-ku, Kyoto 615-8530, Japan)
E-mail: taniguchi@kiban.kuciv.kyoto-u.ac.jp
[3]Lecturer, Dept. of Urban Management, Kyoto University
(C1-2-334 Bldg., Kyoto daigaku-Katsura, Nishikyo-ku, Kyoto 615-8530, Japan)
E-mail: aligul@ums.mbox.media.kyoto-u.ac.jp

We study a variant of the undirected Capacitated Arc Routing Problem (CARP). This problem is motivated from debris collection operation after disaster. In this paper, the sequences in visiting and servicing arcs become very important. It is because one section may block other sections. Only adjacent arcs can be connected with each other, while for distant arcs have no way to be connected before the blocked access between them being removed.

Key Words : *debris collection, disaster, blocked access, possibility access, CVRP, CARP, transformation, tabu search*

1. INTRODUCTION

We study a variant of the undirected Capacitated Arc Routing Problem (CARP)[1,2]. This problem is motivated from debris collection operation after disaster. In this paper, the sequences in visiting and servicing arcs become very important. It is because one section may block other sections. Only adjacent arcs can be connected with each other, while for distant arcs have no way to be connected before the blocked access between them being removed. The other problem that similar with our problem is winter gritting operation[3-8]. The fundamental difference between debris collection operation after disaster and winter gritting operation is the pointing of intervention. In debris collection operation after disaster, because some access are blocked by the debris, sequence in visiting and servicing arcs at the previous structure will influence aggregate accessibility at the next structure. Whereas in winter gritting operation, the timing of an intervention is of prime importance. That is, if the intervention is too early or too late, the cost in material and time sharply increases.

The CARP is NP-hard (non-deterministic polynomial-time hard). It was first addressed with relatively simple heuristics, like the path-scanning [9], construct strike[10], and greedy[11] heuristics. Then, they have been improved over time using metaheuristics like the tabu search[11-22]. Many surveys[1,2,23-26] can be found, explaining many CARP variants.

In this study, the underlying CARP to our debris collection operation will be transformed into the Capacitated Vehicle Routing Problem (CVRP). There exist some efficient methods to tranform CARP into CVRP[27-29]. CARP and CVRP are in fact closely related, the main difference being that in the CARP customers are set of arcs while in the CVRP customers are set of nodes.

The paper is organized as follows. The problem is first introduced in Section 2. Then, its transformation into an equivalent node routing problem is described and the resulting mathematical formulation is presented in Section 3. The tabu search approach for solving this problem and instance problem are reported in Section 4. Finally, the conclusion follows in Section 5.

2. PROBLEM DESCRIPTION

CARP can be defined on an undirected graph $G = (V, A)$, in which V is the set of nodes and A is the set of arcs. A is partitioned into a subset of required arcs A_1, which must be serviced, and another subset of arcs A_2, required to maintain connectivity. Each required arc $a \in A_1$ is associated a demand $z(a)$, a travel cost $tc(a)$, and a service cost $sc(a)$. The other arcs in subset A_2 have a travel cost tc only. Usually, the service cost is greater than the travel cost because it takes more effort to service an arc than to only simply travel along the arc.

A set of identical vehicles $K = \{1\ldots m\}$ are placed at a central depot node. These vehicles with capacity Q_k, are available to service the required arcs. Each vehicle services a single route that must start and end at the depot. The vehicles are only allowed to move from or to adjacent arcs. However for distant arcs, some of arcs blocked, so those are not allowed to be visited before removing the blockage first. The objective is to service all required arcs in the graph at least cost with feasible routes, where the cost is related to the number of vehicles used, the travel cost and the service cost. In our case, all of the service cost sc is assumed to be 0, and also fix vehicle cost fc is assumed to be 0. So the cost involved is only travel cost. But if not so, we can add the service cost once the vehicle visit and service the required arc and add the fix vehicle cost everytime the vehicle starts from depot.

3. PROBLEM FORMULATION

We make a transformation from ARP (Arc Routing Problem) in graph $G = (V, A)$ into an equivalent VRP (Vehicle Routing Problem) in a transformed graph $G` = (V`, A`)$. In a well known transformation[27], an arc $a \in A$ in CARP is represented by three nodes in the equivalent CVRP. Since we anticipate large instance of debris collection operation, we will use the type of transformation ARP into VRP with making two nodes for each required undirected arc[28,29].

The transformation proposed by Longo et al.[29], is described as follows. An arc (i, j) in A_1 is associated with two nodes sij and sji, thus the resulting CVRP instance is defined on a complete undirected graph $G` = (V`, A`)$, where:

$$V` = \bigcup_{(i,j)\in A} \{sij, sji\} \cup \{o\} \tag{1a}$$

Node 0 serves as the depot. The arc costs d and the demands z are defined Longo et al.[29] as follows:

$$d\,(sij, skl) = \begin{cases} 0 & \text{if } (i, j) = (k, l) \\ d\,(i, j) & \text{if } (i, j) = (l, k) \\ dist\,(i, k) & \text{if } (i, j) \neq (k, l),\ (i, j) \neq (l, k) \end{cases}$$

$$d\,(o, sij) = dist\,(o, i), \tag{1b}$$

Here $dist\,(i, j)$ is the value of the shortest path from node i to node j in G. Eventhough it depends on whether the access between node i and node j is blocked or not, but we assume that for going back to depot (j = depot) the vehicle always can traverse the shortest path. The new demands are:

$$z(sij) = z(sji) = \frac{1}{2} z(i, j) \tag{1c}$$

The transformation fixes variable on all undirected arcs $\{(sij, sji) \mid (i, j) \in A\}$ to 1. It means that CVRP solutions are only feasible where sij and sji are visited in sequence, either sij from or to sji. But in our case, a vehicle which load over capacity after visiting sij cannot move to sji in sequence and must go back to the depot, without even servicing sij.

After the transformation, we obtain a VRP with blocked access. This type of problem has never been addressed in the literature. Some variants of the Vehicle Routing Problem with Time Windows (VRPTW) can be reviewed in[13,17,18,30,31] and a good review of different time constraint VRP can also be found in[32]. Our problem is without using time windows, considering time constraint is not appropriate to be implemented in urgent situation such as after disaster.

We use the same notation as in Tagmouti et al.[3], where $N` \in V`$, is the set of nodes that must be serviced. The depot is a single node, but duplicated into an origin depot o and a destination depot d in $V`$. As a new idea for the debris collection operation, we introduce a possibility access constraint on the nodes $p_{ij}^k$, $(i, j) \in A`$, $k \in K$, which are equal to 1 if vehicle k from node i can possibly visit and service node j, 0 otherwise. The decision variables are: (1) the binary flow variables on the arcs x_{ij}^k $(i, j) \in A`$, $k \in K$, which are equal to 1 if vehicle $k$ travels on arc $(i, j)$ to service node $j$, 0 otherwise; and (2) the non-negative load variables $Q_i^k$, $i \in V`$ which specify the load of vehicle k just after servicing node i. Note that $Q_o^k = Q_k$, $k \in K$; $d = tc + sc$; and $z_o = z_d = 0$. The transformed CVRP can be formulated as follows:

Min

$$\sum_{k\in K} \sum_{(i,j)\in A`} d_{ij}\, x_{ij}^k \tag{2a}$$

Subject to

$$\sum_{k\in K} \sum_{i\in N`\cup\{o\}} x_{ij}^k = 1 \quad ; j \in N` \tag{2b}$$

$$\sum_{k\in K}\sum_{j\in N`} x_{oj}^{k} \leq m \quad ; K = \{1\ldots m\} \quad (2c)$$

$$\sum_{j\in N`\cup\{d\}} x_{oj}^{k} = 1 \quad ; k \in K \quad (2d)$$

$$\sum_{i\in N`\cup\{d\}} x_{ij}^{k} - \sum_{j\in N`\cup\{o\}} x_{ji}^{k} = 0 \quad ; k \in K \quad (2e)$$

$$\sum_{i\in N`\cup\{o\}} x_{id}^{k} = 1 \quad ; k \in K \quad (2f)$$

$$x_{ij}^{k}\left(Q_i^k - z_j - Q_j^k\right) \leq 0 \quad ; k \in K, (i,j)\in A` \quad (2g)$$

$$x_{ij}^{k} - p_{ij}^{k} \leq 0 \quad ; k \in K, (i,j)\in A` \quad (2h)$$

$$0 \leq Q_i^k \leq Qk \quad ; k \in K, i\in V` \quad (2i)$$

$$0 \leq x_{ij}^{k} \leq 1 \quad ; k \in K, (i,j)\in A` \quad (2j)$$

$$0 \leq p_{ij}^{k} \leq 1 \quad ; k \in K, (i,j)\in A` \quad (2k)$$

$$x_{ij}^{k} \in \{0,1\} \quad ; k \in K, (i,j)\in A` \quad (2l)$$

$$p_{ij}^{k} \in \{0,1\} \quad ; k \in K, (i,j)\in A` \quad (2m)$$

The objective function (2a) minimizes the sum of travel costs. A fixed vehicle cost *fc* can also be added to the travel costs d_{oj} , $j\in V`$ if one wants to penalize the use of an additional vehicle, but in our case we set it as 0. Constraints (2b) require that each node in *V`* must be serviced once. Constraints (2c) limit the number of vehicles used. Constraints (2d)-(2f) are the flow conservation constraints. Constraint (2g) are for the feasibility of the loads. Constraints (2h) impose that each node in *V`* travels from or to possible node. Constraints (2i) ensure load values that do not exceed vehicle capacity Q_k and are > 0. Constraints (2j)-(2m) are binary values for the flow variables and possibility access constraints.

4. TABU SEARCH

(1) Tabu Search Algorithm

In this study, we propose a tabu search heuristics to solve the CVRP problem, as tabu search or heuristics in general is more appropriate and faster to solve large problems practically. The tabu search scheme proposed here, is well documented in the literature [11-22]. Tabu search has quickly become one of the best and most widespread local search methods for combinatorial optimization. It is deserved to say that tabu search has been greatly successful in solving some difficult problems.

The method performs an exploration of the solution area in a subset of the neighborhood *N(s)* by moving from a solution *s* at iteration *k* to the best solution *s`* at iteration *k*+1. Since *s`* at iteration *k*+1 does not always have an improvement upon *s* at iteration *k*, a tabu mechanism is implemented to prevent the process from cycling over a sequence of solutions.

We put prohibited moves in the list called as tabu list *T(s,k)*. Aspiration criteria *A(s,k)* is set as an exception, which says eventhough some moves are tabu, but as long as making improvement for the solution, then the tabu list can be violated.

The other mechanisms used in our tabu search are diversification and intensification. The diversification keeps track of past solutions and imposes penalty for the frequently performed moves. The intensification performs search around solution features historically found good. In Fig.1, we can show the basic algorithm of tabu search that reviewed in details in[32].

```
k = 1.
Generate initial solution
WHILE the stopping condition is not met DO
   Identify N(s). (Neighborhood set)
   Identify T(s,k). (Tabu set)
   Identify A(s,k). (Aspirant set)
   Choose the best s` ∈ N(s,k) = {N(s) - T(s,k)}+A(s,k).
   Memorize s` if it improves the previous best solution
   s = s`.
   k = k+1.
END WHILE
```

Fig.1 Basic Tabu Search Algorithm

(2) Instance Problem

We test our model on a small problem instance, see Fig. 2. An instance with 6 nodes (node 1 = depot) and 9 arcs (all are required arcs), where:

- *V* = {1, 2, 3, 4, 5, 6};
- Q_k = 30 ton;
- Cost *d(i, j)* : *d*(1,2) = 8, *d*(1,4) = 3, *d*(2,3) = 7, *d*(2,4) = 6, *d*(3,4) = 4, *d*(3,5) = 5, *d*(3,6) = 5, *d*(4,5) = 4, *d*(5,6) = 6 ;
- Demand *z(i, j)* : *z*(1,2) = 8 ton, *z*(1,4) = 3 ton, *z*(2,3) = 7 ton, *z*(2,4) = 6 ton, *z*(3,4) = 4 ton, *z*(3,5) = 5 ton, *z*(3,6) = 5 ton, *z*(4,5) = 4 ton, *z*(5,6) = 6 ton.
- *sc* = 0 and *fc* = 0, thus *d* = *tc*.

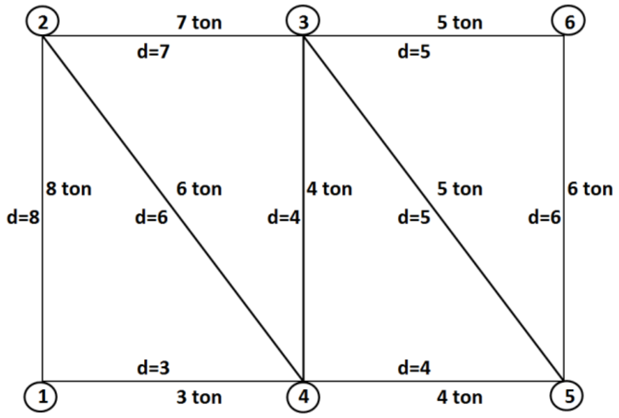

Fig.2 Capacitated Arc Routing Problem

The CARP of Fig.2 is transformed into a CVRP with number of nodes $V` = 2n+1$; $n$ = number of arc. After being transformed, graph tuns into VRP with number of nodes $V` = 19$, and node 1 still serves as depot. The transformation introduces nodes s_{ij} and s_{ji} for all required arcs $(i, j) \in A$, as shown in Fig. 2, where:

- $V`$= {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19};
- Demand z : $z2$ = 4 ton, $z3$ = 1.5 ton, $z4$ = 4 ton, $z5$ = 3.5 ton, $z6$ = 3 ton, $z7$ = 3.5 ton, $z8$ = 2 ton, $z9$ = 2.5 ton, $z10$ = 2.5 ton, $z11$ = 1.5 ton, $z12$ = 3 ton, $z13$ = 2 ton, $z14$ = 2 ton, $z15$ = 2.5 ton, $z16$ = 2 ton, $z17$ = 3 ton, $z18$ = 2.5 ton, $z19$ = 3 ton.

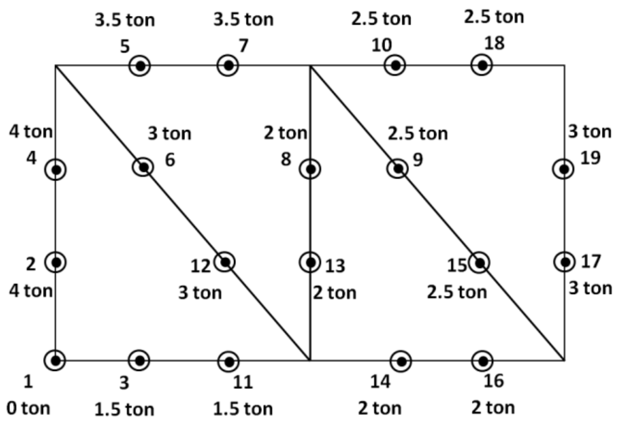

Fig.3 Capacitated Vehicle Routing Problem (transformed)

The final transformed instance, a constrained CVRP instance with 19 nodes, is defined over a complete graph with the costs between nodes presented in Table 1, calculated by the equation (1b).

Table 1 Cost Matrix of Capacitated Vehicle Routing Problem

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
1	-	0	0	8	8	8	7	7	7	7	3	3	3	3	7	7	7	12	12
2	0	-	0	8	8	8	7	7	7	7	3	3	3	3	7	7	7	12	12
3	0	0	-	8	8	8	7	7	7	7	3	3	3	3	7	7	7	12	12
4	8	8	8	-	0	0	7	7	7	7	6	6	6	6	10	10	10	12	12
5	8	8	8	0	-	0	7	7	7	7	6	6	6	6	10	10	10	12	12
6	8	8	8	0	0	-	7	7	7	7	6	6	6	6	10	10	10	12	12
7	7	7	7	7	7	7	-	0	0	0	4	4	4	4	5	5	5	5	5
8	7	7	7	7	7	7	0	-	0	0	4	4	4	4	5	5	5	5	5
9	7	7	7	7	7	7	0	0	-	0	4	4	4	4	5	5	5	5	5
10	7	7	7	7	7	7	0	0	0	-	4	4	4	4	5	5	5	5	5
11	3	3	3	6	6	6	4	4	4	4	-	0	0	0	4	4	4	9	9
12	3	3	3	6	6	6	4	4	4	4	0	-	0	0	4	4	4	9	9
13	3	3	3	6	6	6	4	4	4	4	0	0	-	0	4	4	4	9	9
14	3	3	3	6	6	6	4	4	4	4	0	0	0	-	4	4	4	9	9
15	7	7	7	10	10	10	5	5	5	5	4	4	4	4	-	0	0	6	6
16	7	7	7	10	10	10	5	5	5	5	4	4	4	4	0	-	0	6	6
17	7	7	7	10	10	10	5	5	5	5	4	4	4	4	0	0	-	6	6
18	12	12	12	12	12	12	5	5	5	5	9	9	9	9	6	6	6	-	0
19	12	12	12	12	12	12	5	5	5	5	9	9	9	9	6	6	6	0	-

The new idea, that we proposed in this study, is possibility access matrix i.e., a constraint, on whether it is possible or not, the vehicle can move from one node to another. If vehicle k want to move from i to j, so node i and j must be either adjacent or nodes other than i and j that must be visited by vehicle k, in order to move from i to j, have to have demand $z = 0$ (they must have been serviced before). The possibility access matrix would always change from original and previous positions, everytime a vehicle service demand in each required node. It is because the blocked access condition will change, everytime demands on a node have been serviced.

We start with finding an initial solution, using a greedy heuristic[11]. The greedy heuristic attempts to construct a feasible solution moving from the current point. The idea is fairly simple, starting at the depot, a vehicle simply choose the closest customer at each iteration until all customers are visited. The vehicle goes back to depot only if all customers are visited; or loads exceed the vehicle capacity; or no possible way to move without visiting depot. Cost obtained is **69** and the route is **1 – 2 – 4 – 5 – 7 – 8 – 13 – 11 – 3 – 12 – 6 – 1 – 14 – 16 – 15 – 9 – 10 – 18 – 19 – 17 – 1**.

Then, we continue using tabu search to solve VRP to find a better solution than the initial solution and finally found a better and the best solution for the debris collection operation after disaster VRP problem. Cost obtained is **63** and the route is **1 – 2 – 4 – 5 – 7 – 8 – 13 – 11 – 3 – 1 – 6 – 12 – 14 – 16 – 15 – 9 – 10 – 18 – 19 – 17 – 1**.

5. CONCLUSIONS

The debris collection operation after disaster is a new CVRP problem and not much research has been done in this topic. The uniqueness of this kind of CVRP problem is due to the limited access from one section to another, as a result of the blocked access by debris. Therefore a modification in classical CVRP is required to solve this kind of problem. It is committed by adding a new constraint, which is mentioned in this study as possibility access constraint. This constraint sets whether a vehicle possibly moves from one node to another in a particular structure.

This problem is solved by using a tabu search, considering that in practice of debris collection operation after disaster would involve large instances. Practically, metaheuristics such as tabu search are more appropriate and faster to solve such problems. Future research will now focus on giving scale of priority for the access to be serviced, because in this study, the entire access have same scale of priority. Furthermore, solving the problem with multi-depot and split delivery could be choosen as other directions of research to get a better solution.

ACKNOWLEDGMENT: This work was partially supported by Department of Urban Managament, Graduate School of Engineering, Kyoto University. This support is gratefully acknowledged.

REFERENCES

1) Golden, B.L., Wong, R., Capacitated Arc Routing Problems, *Networks 11*, 305–315, 1981.
2) Dror, M., Arc Routing: Theory, Solutions and Applications, *Kluwer*, 2000.
3) Tagmouti, M., Gendreau, M., Potvin, J., Arc Routing Problem with Time Dependent Service Costs, *European Journal of Operational Research* 181, 30-39, 2007.
4) Campbell, J.F., Langevin, A., Roadway Snow and Ice Control, in: Dror, M. (Ed.), Arc Routing: Theory, Solutions and Applications, *Kluwer*, pp. 389–418, 2000.
5) Eglese, R.W., Li, L.Y.O., Efficient Routing for Winter Gritting, *Journal of the Operational Research Society* 43, 1031–1034,1992.
6) Eglese, R.W., Routing Winter Gritting Vehicles, *Discrete Applied Mathematics 48*, 231–244, 1944.
7) Eglese, R.W., Li, L.Y.O., An Interactive Algorithm For Vehicle Routing for Winter Gritting, *Journal of the Operational Research Society 47*, 217–228, 1996.
8) Lotan, T., Cattrysse, D., Oudheusden, D.V., Winter Gritting in the Province of Antwerp: A Combined Location and Routing Problem, *Technical Report IS-MG 96/4, Institut de Statistique, Universite´ Libre de Bruxelles*, 1996.
9) Golden, B.L., DeArmmon, J.S., Baker, E.K., Computational Experiments with Algorithms for A Class of Routing Problems, *Computers & Operations Research 10*, 47–52, 1983.
10) Christofides, N., The Optimum Traversal of A Graph, *Omega 1*, 719–732, 1973.
11) Costa, M.C., Hertz A., Mittaz, M., Bounds and Heuristics for the Shortest Capacitated Paths Problem, *Journal of Heuristics, 8*, 449–465, 2002
12) Hertz, A., Laporte, G., Mittaz, M., A Tabu Search Heuristic for The Capacitated Arc Routing Problem, *Operations Research 48*, 129–135, 2000.
13) Taillard, E., Badeau, P., Guertin, F., Gendreau, M., Potvin, J., A Tabu Search Heuristic for the Vehicle Routing Problem with Soft Time Windows, *Transportation Science 31*, 170-186, 1997.
14) Joubert, J.W., A Tabu Search Solution Algorithm, University of Pretoria etd, chapter 5, 83-91, 2007.
15) Glover, F. and Laguna, M., Tabu Search, *Kluwer Boston*, 1997.
16) Cordeau, J.F., Gendreau, M., Laporte, G., Potvin J., Semet, F., "A Guide to Vehicle Routing Heuristics," *Journal of the Operational Research Society 53*, 512-522, 2002.
17) Cordeau, J.F., Laporte, G., "A Tabu Search Algorithm for the Site Dependent Vehicle Routing Problem with Time Windows," *INFOR 39*, 292-298, 2001.
18) Cordeau, J.F., Laporte, Mercier, A., "A Unified Tabu Search Heuristic for Vehicle Routing Problems with Time Windows," *Journal of the Operational Research Society 52*, 928-936, 2001.
19) DeBacker, B. et.al, Solving Vehicle Routing Problems Using Constraint Programming and Metaheuristics, *Journal of Heuristics*, Volume No., 1–16, 1997.
20) Hertz, A., DeWerra, D., "The Tabu Search Metaheuristic: How We Used It," *Annals of Mathematics and Artificial Intelligence 1*, 111-121, 1991.
21) Gendreau, M., "An Introduction to Tabu Search," in Handbook of Metaheuristics, Glover F., Kochenberger G.A., (eds), *Kluwer, Boston*, 37-54, 2003.
22) Mastrolilli, M., Tabu Search, IDSIA - *Instituto Dalle Molle Institute for Artificial Intelligence*, 2001.
23) Assad, A.A., Golden, B.L., Arc Routing Methods and Applications, in: Ball M.O., et al. (Eds.), *Network Routing, Elsevier*, pp. 375–483, 1995.
24) Cordeau, J.F., Laporte, G., Modeling and Optimization of Vehicle Routing and Arc Routing Problems, *Les Cahiers du Gerard G-2002-30*, Montreal, 2002.
25) Eiselt, H.A., Gendreau, M., Laporte, G., Arc Routing Problems, Part I: The Chinese Postman Problem, *Operations Research 43*, 231–242, 1995.
26) Eiselt, H.A., Gendreau, M., Laporte, G., Arc Routing Problems, Part II: The Rural Postman Problem, *Operations Research 43*, 399–414, 1995.
27) Pearn, W., Assad, A., Golden, B.L., Transforming Arc Routing into Node Routing Problems, *Computers & Operations Research 14*, 285–288, 1987.
28) Baldacci, R., Maniezzo, V., Exact Methods Based on Node Routing Formulations for Undirected Arc Routing Problems, *Networks 47*, 52–60, 2006.
29) Longo, H., DeAragao, M.P., Uchoa, E., Solving Capacitated Arc Routing Problems Using A Transformation to the CVRP, *Computers & Operations Research 33*, 1823–1837, 2006.
30) Ibaraki, T., Imahori, S., Kubo, M., Masuda, T., Uno, T., Yagiura, M., Effective Local Search Algorithms for Routing and Scheduling Problems with General Time Window Constraints, *Transportation Science 39*, 206–232, 2005.
31) Ioachim, I., Ge´linas, S., Desrosiers, J., Soumis, F., A Dynamic Programming Algorithm for the Shortest Path Problem with Time Windows and Linear Node Costs, *Networks 31*, 193–204, 1998.
32) Desrosiers, J., Dumas, Y., Solomon, M.M., Soumis, F., Time Constrained Routing and Scheduling, in: M.O. Ball et al. (Eds.), *Network Routing, Elsevier*, pp. 35–139, 1995.