Print ISSN: 1738-3110 / Online ISSN 2093-7717
http://dx.doi.org/10.15722/jds.14.2.201602.41

Optimization of Transportation Problem in Dynamic Logistics Network*

Ji-Bok Chung**, Byung-Cheon Choi***

Received: January 19, 2016. Revised: January 31, 2016. Accepted: February 15, 2016

Abstract

Purpose – Finding an optimal path is an essential component for the design and operation of smart transportation or logistics network. Many applications in navigation system assume that travel time of each link is fixed and same. However, in practice, the travel time of each link changes over time. In this paper, we introduce a new transportation problem to find a latest departing time and delivery path between the two nodes, while not violating the appointed time at the destination node.

Research design, data, and methodology – To solve the problem, we suggest a mathematical model based on network optimization theory and a backward search method to find an optimal solution.

Results – First, we introduce a dynamic transportation problem which is different with traditional shortest path or minimum cost path. Second, we propose an algorithm solution based on backward search to solve the problem in a large-sized network.

Conclusions – We proposed a new transportation problem which is different with traditional shortest path or minimum cost path. We analyzed the problem under the conditions that travel time is changing, and proposed an algorithm to solve them. Extending our models for visiting two or more destinations is one of the further research topics.

Keywords: Time-Dependent Network, Optimization Model, Search Algorithm.

JEL Classifications: C44, C61, C65.

1. Introduction

Delivery services are becoming more important and competitive among major online shopping companies such as amazon and eBay, and they have introduced various delivery options for customers. For example, eBay has introduced a shipping service called 'eBay now' which delivers a parcel to customers at the appointed place and time. However, many logistic companies have to solve the tradeoff problems between customer services and operational efficiency, since providing various options to customer is to increase the operational complexity of company. In order to increase the operational efficiency in various delivery services, we have to consider not only the delivery time but also the appointment with customers. Traditionally many researches have been worked to find a shortest path(SP) or fastest path. But sometimes a shortest path is not a best solution when there is an appointed time to meet with the customer at the destination place. In case the appointment time with customer is 7 p.m., it is not necessary to arrive and wait for 20 minutes along the fastest path. Logistic companies have to consider not only the travel time but also the appointment time with customers. Furthermore, it is more complicated in congested transportation networks where travel times of each road change over time due to time-of-day variations in traffic congestion(Yajun et al., 2014).

Therefore, computing an efficient delivery path has been an essential component for the design and operation of smart transportation or logistic network(Ahuja et al.,1993; Low & Gao, 2011; Amrapali et al., 2015; Shuai & Chao, 2012 Nejad et al., 2016).With the advancement of IT technology, recently South Korea started to provide not only current information about traffic congestion but also estimated future traffic congestion such that logistic company find a delayed departing time to arrive at destination node not to wait for the appointed time. By using this information, it is expected that logistic company or user of navigation system can make a smart or efficient decision about their delivery problems.

In this paper, we introduce a new transportation problem under dynamic travel time (TP_dynamic for short) that is to find a latest (delayed) departing time and a delivery path between the two nodes (from source node u to destination node v),while not violating the appointment time (due date) at the destination node v.

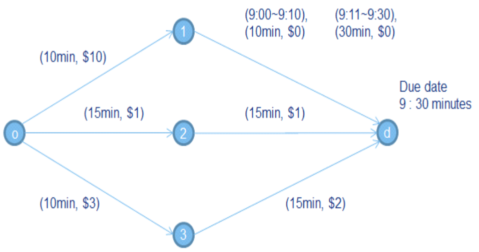
* A preliminary version of paper was published and revised after discussion with three reviewers at winter conference of ICBE2015.
This study was financially supported by the research fund of Kongju National University in 2015.
This study was financially supported by the research fund of Chungnam National University in 2014.

** First author, Associate Professor, Department of Retail Management, Kongju National University, Korea. Tel: +82-41-330-1428, E-mail: jbchung@kongju.ac.kr

*** Corresponding author, Associate Professor, Department of Business Management, Chungnam National University, Korea. Tel: +82-42-821-5577, E-mail: polytime@cnu.ac.kr

We want to introduce our problem (TP_dynamic) by comparing with traditional shortest path and minimum cost path problem in Figure 1. As you can see in Figure 1, we have a logistic network composed of 5 nodes and 6 links. We assume that each link has an information about travel cost such as toll fee and future travel time which can be changed depending on the time horizon. For a link (o1), it takes 10 minutes and costs $10 and for a link (1d), it takes 10 minutes between 9:00~9:10,while it takes 30 minutes between 9:11~9:30. Finally we have an appointment time with customer at 9:30.

In terms of travel time, delivery path P1(o->1->d) is a shortest (fastest) path because it takes 20 minutes when departing at 9:00 while delivery path P2(o->2->d) is a minimum cost path because it costs $2. Although P1 is a shortest path, we have to depart at 9:00 and arrive at 9:20 and wait 10 minutes to the appointment time with customer, since if we depart after 9:00 at node o, we will be late to the appointment time due to the traffic congestion. Although travel time of delivery path P3(o->3->d) is 25 minutes which is longer than P1, we may depart at 9:05 at node o to arrive at node d not later than the appointment time. We can save 5 minutes if we decide to follow the delivery path P3 rather than P1.Therefore, our problem is related with finding the delivery path P3 for general logistic network, and that is the originality of our paper.



Delivery Path	Travel time	Travel cost	Departing time(o)	Arriving time(v)	
P1(o->1->d)	20 minutes	$10	9:00	9:20	Shortest path
P2(o->2->d)	25 minutes	$2	9:00	9:30	Minimum cost path
P3(o->3->d)	30 minutes	$5	9:05	9:30	Our problem(TP_dynamic)

<Figure 1> Comparison of SP, MCP and Our problem(TP_dynamic)

The contribution of our paper is two-fold. First, we introduce a new transportation problem under dynamic travel time(TP_dynamic)which is different with traditional shortest path problem or minimum cost problem. Second, we propose a solution algorithm based on backward search to solve our problem.

2. Problem Description

2.1. Literature review

The single source shortest path problem of finding a shortest path between a source node u and a target node v can be always solved by applying a Dijkstra's algorithm (Dijkstra, 1959), who gave an algorithm using O(m+n) priority queue operations for a graph G = (V, E) with n nodes and m edges. Many application models have been followed after Dijkstra(1959) such as reverse shortest path problem, K-shortest path problem, one-to-all shortest path problem, and shortest pair of disjoint path problem(Ahuja et al., 1993).

Due to varying congestion on roads during the day, both the time to travel from a source node s to a destination d and the optimal path between them can change over time. Therefore, time-dependent networks are used to model situations in which traveling times are changing with time and time-dependent shortest paths are also a fundamental problem with non-trivial complexity.

The time-dependent shortest path problem has been first studied by Dreyfus(1969) who observed that Dijkstra's algorithm can be used to find a time-dependent shortest path, given a starting time at the source node. Orda and Rom(1990) showed that the choice of waiting policies and the type of the edge cost functions have non-trivial implications on time-dependent shortest paths. Among three kinds of waiting policies of Orda and Rom(1990), that is, unrestricted waiting, forbidden waiting and source waiting, our model assumes source waiting policy since waiting is allowed only at the source node. Sherali et al.(1998) studied about the problem to find a time-dependent shortest pair of disjoint paths and showed that many variants of problem belongs to NP-complete and suggest a binary linear programming formulation to solve their problem. Dean(2004) studied to minimize travel cost and travel time for all-to-one shortest path problem in time dependent network. Horst et al. (2006) proposed an efficient algorithm for time dependent shortest path problem with two conflicting objective functions. Abbasi and Ebrahimnejad(2011) studied the problem of finding shortest paths from one node to all the other nodes in dynamic network. Each link has a transit time and parking with a corresponding time whichis allowed at the nodes. They showed that this problem is equivalent to a classical shortest path problem in a time-expanded network, and suggest a label correcting algorithm as a solution algorithm. Low and Gao (2011) introduced the just-in-time delivery and scheduling problem in construction industry considering the characteristics of concrete. Shuai and Chao(2012) studied the heterogeneous fixed fleet vehicle routing problem with pick-up and delivery for vehicles with different capacities, fixed costs, and travel costs, and they proposed a practical mathematical model and simulated annealing algorithm to solve their problem. Song and Kim (2012) analyzed the structural relationships between customer loyalty and delivery service quality in Korean courier services. Dewen et al. (2014) proposed an application case of vehicle navigation system considering dynamic traffic congestion and providing users with multiple options for path selection. Yang et al.(2014) proposed a cost-optimal optimization problem in time dependent network which considers travel cost and travel time simultaneously. Jiang and Wub(2014) have proposed a dynamic navigation system that is considering the traffic congestion, bad weather condition or link disruptions. El-Sherbeny (2014)studied variants of the time-dependent shortest path problem such that in order to arrive at the destination node between time windows [a, b].Maurizioet al. (2014) introduced the case of project (IMPULSO) for the integration between ICT systems and logistic operators for the distribution in metropolitan areas. IMPULSO considered a multi-attribute road network such as travel times, travel costs and risks, and tried to solve multi-objective time dependent shortest path with forbidden turns problem. Recently, Amrapali et al. (2015)propose an implementation case of finding the shortest path from the user location to hospitals selected by using ArcGIS and the Dijkstra's algorithm where ArcGIS is software used to digitize the map into road network.

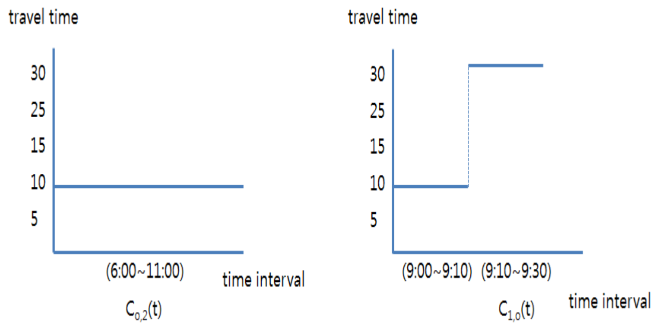
2.2. Problem Description

In this section, we will explain our problem based on the network optimization modeling.

Definition 2.1: (Time Dependent Network) A time-dependent network is defined as G(V,A,C) : V is a set of nodes, V = {1,2, ... ,n}, A is a set of links, A ⊆ V x V, C is a set of travel times, C = {$c_{ij}(t)$| (i,j) ∈A, t∈T}, $c_{ij}(t)$is an estimated travel time for link (i,j) if departing at time interval t = t[i].

$$c_{i,j}(t) = \left\{ \begin{matrix} c_1, t[1] = (t_1, t_2) \\ c_2, t[2] = (t_2, t_3) \\ c_3, t[3] = (t_3, t_4) \\ \cdots\cdots \\ \cdots\cdots \\ c_k, t[k] = (t_k, t_{k+1}) \end{matrix} \right\}$$

As you can see in Figure 2, the travel time at link (1o) is 10 minutes at time interval t[1]=[9:00~9:10] and 30 minutes at time interval t[2]=[9:10~9:30] while the travel time at link (o2) is 10 minutes.



<Figure 2> Example of time-dependent network

2.3. Mathematical Model

In this paper, we will use the following symbols and notations to explain our problems.

Decision Variables:

$x_{ij}(t)$ = 1, if path p traverse link (ij) at time interval t
= 0, otherwise

depart(i) : departing time at node i
wait(i) : waiting time at node i
arrive(i) : arriving time at node i
successor(i) : set of successor nodes of i

When arriving at node i, one may wait a time w(i) if the departing time can be delayed. For each node, we have a relationship that is

depart(i) = arrive(i) + wait(i)

Let p = <o, n_1, ...,nk, d> be a given path, then we have
arrive(n_1) = depart(o) + $c_{ij}(depart(o))$
arrive(n_2) = depart(n_1) + $c_{ij}(depart(n_1))$
arrive(d) = depart(n_k) + $c_{ij}(depart(n_k))$

By using the above symbols and notations, we can suggest an optimization model as follows:

Optimization Model:
maximize depart(o) (1)

subject to

$$\mathrm{depart(i)} + \Sigma_t c_{ij}(t) x_{ij}(t) \le depart(j), i \in pred(j), \forall j \ne d \quad (2)$$

$$\mathrm{depart(i)} + \Sigma_t c_{ij}(t) x_{ij}(t) \le d, i \in pred(j), j \ne d \quad (3)$$

$$\Sigma_t \Sigma_{ij} x_{ij}(t) - \Sigma_t \Sigma_{ji} x_{ji}(t) = 1, i = o \quad (4)$$

$$\Sigma_t \Sigma_{ij} x_{ij}(t) - \Sigma_t \Sigma_{ji} x_{ji}(t) = -1, i = d \quad (5)$$

$$\Sigma_t \Sigma_{ij} x_{ij}(t) - \Sigma_t \Sigma_{ji} x_{ji}(t) = 0, i = o, d \quad (6)$$

$$x_{ij}(t) \ge 0, \mathrm{depart(i)} \ge 0$$

The objective function (1) means that we want to delay the departing time at the node o. Constraints (2)-(3) mean the precedence relationship between two nodes (ij) such that departing time of node j cannot be larger than the arrival time at node j. Constraints (4)-(6) mean the flow conservation constraints in network flow theory.

3. Solution method for TP_dynamic

In the following, we propose a solution algorithm based on backward search composed of two functions, find_latest_time_interval() and update_depart_time() for the TP_dynamic and we will explain each functions in details.

3.1 find_latest_time_interval()

find_latest_time_interval() is a function to find a latest time interval at link (i, j) not later than depart(j) between two nodes i and j. From the end of time_interval lists, simple check whether the last time_interval is not delaying the departing time of successor node. For the example in Figure 3, since there are three time_intervals between nodes 3 and d and the last time_interval t[3] is not possible to arrive not later than the due date, the next time_interval t[2] becomes the latest time_interval. Likewise, among three time_intervals between nodes 2 and d, time_interval t[1] becomes the latest time_interval since the both time_interval t[3] and t[2] are not possible to arrive not later than the due date.

find_latest_time_interval()

Input : two nodes i, j and departure time of node j
Output :find a latest time interval at link (i,j) not later than depart(j)

```
from the largest time_interval t[i] at link (i,j)
if (ti + $c_{ij}$ (ti)< depart(j)) return t[i];
```

3.2 update_depart_time()

update_depart_time()is a function to update or assign new depart time based on the latest time_interval determined by the function find_latest_time_interval(). For the example in Figure 3, since the latest time_interval between node 3 and d is determined by t[2] and travel time at t[2] is 18 minutes, the departing time at node 3 can be 5:42 not to be late at node d. Likewise, since the latest time_interval between node 2 and d is determined by t[1] and travel time at t[1] is 25 minutes, the departing time at node 3 can be 5:35 not to be late at node d.

update_depart_time()

Input : two nodes k, i and time interval s
Output : depart time(k)

```
depart(k) = depart(i) - cij(s)
```

two-step heuristic method based on backward search

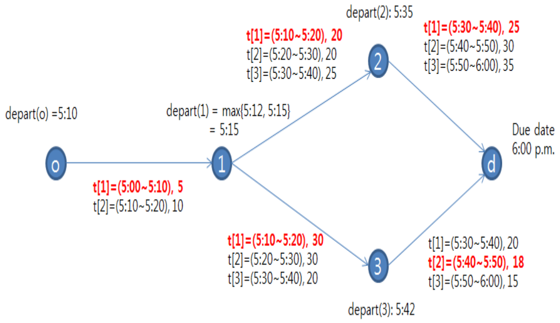
Input Input: a time-dependent network GT, a query <o, d, t>, some big number M
Output : optimal path p*=<o, n1, n2, ... , nk, d>
latest departure time, s*=$\max_p\{depart(o)\}$

```
k = d;
depart(k) = t;
do {
k = k-1;
for each node i in successor(k)
depart(k) = M;
for each link(k, i)
s* = find_latest_time_interval[k, i, depart(i)];
temp_depart(k) = update_depart_time(k, i, s*);
if (temp_depart(k) < depart(k)) then
depart(k) = temp_depart(k);} while( k ≠ o)
```

Detailed steps to explain our methods are provided in <Table 1> and <Figure 3>.

<Table 1> Running example of solution method for example in Figure3

at node d	depart(d) = due date = 6:00
at node 3	find_latest_time_interval[3,d,6:00] = t[2] update_depart_time(3, d, t[2]) = depart(3) = 5:42
at node 2	find_latest_time_interval[2,d,6:00] = t[1] update_depart_time(2, d, t[1]) = depart(2) = 5:35
at node 1	find_latest_time_interval[1,3,5:42] = t[3] update_depart_time(1, 3, t[3]) = depart(1) = 5:12
	find_latest_time_interval[1,2,5:35] = t[1] update_depart_time(1, 2, t[1]) = depart(1) = 5:15
at node o	find_latest_time_interval[o,1,5:15] = t[1] update_depart_time(o, 1, 5:15) = depart(o) = 5:10



<Figure 3> example to explain our heuristic algorithm

5. Conclusion

In this paper, we have proposed a new transportation problem, which is different with traditional shortest path problem or minimum cost path problem since our problem tries to find a just-in-time delivery path. The objective of our problem is to find a latest departing time and a delivery path from source node u to destination node v, while not violating the due date at the destination node v. We have proposed a mathematical model based on conditions that travel time is changing with time and suggested an efficient solution algorithm based on backward search to solve our problem. Extending our results for more than two or more destinations is one of the further research topics.

References

Abbasi, S., & Ebrahimnejad, S. (2011). Finding the Shortest Path in Dynamic Network using Labeling Algorithm. *International Journal of Business and Social Science*, 2(20), 239-243.

Ahuja, R. K., Magnanti, T. L.., & Orlin, J. B. (1993).*Network Flows: Theory, Algorithms, and Applications.* Englewood Cliffs, New Jersey: Prentice Hall.

Amrapali, D., Kale, K. V., & Yogesh, G. (2015). Network

Analysis for Finding Shortest Pathin Hospital Information System. *IJARCSSE,* 5(7), 618-623.

Brian, D. (2004). Introduction Algorithms for Minimum-Cost Paths in Time-Dependent Networks with Waiting Policies. *Networks*, 44(1), 41-46.

Dijkstra, E. W. (1959). A Note on Two Problems in Connection with Graphs. *Numerische Mathematic,* 1, 269-271.

Dewen, S., Meixia, T., Wu, H., Fang, X., & Xu, H. (2014). Multiple Constrained Dynamic Path Optimization based on Improved Ant Colony Algorithm. *IJUNESST*, 7(6), 117-130.

Dreyfus, S. E. (1969). AnAppraisal of Some Shortest Path Algorithms. *Operations Research,* 17(3), 395–412.

El-Sherbeny, N. A. (2014). The Algorithm of the Time-Dependent Shortest Path Problem with Time Windows. A*pplied Mathematics,* 5, 2764-2770.

Horst, W. H., Stefan, R., & Stevanus, A. T. (2006). Algorithms for Time-Dependent Bicriteria Shortest Path Problems. *Discrete Optimization,* 3, 238–254.

Jiang, J., & Wub, L. (2014). Dynamic Navigation Algorithm Considering Network Disruptions. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences,* XL-4, 111-115.

Low, S.P., & Gao, L. (2011). The Application of the Just-in-Time Philosophy in the Chinese Construction Industry. *Journal of Construction in Developing Country,* 16(1), 91-111.

Maurizio, B., Alberto C., Federico, L., & Alessandro, L. (2014). A Multi-Objective Time-Dependent Route Planner: a Real World Application to Milano city. *Transportation Research Procedia,* 3, 460- 469.

Nejad, M., Mashayekhy, L., Chinnam, R., & Phillips, A. (2016). Hierarchical Time-DependentShortestPathAlgorithms for Vehicle Routing under ITS. *IIE Transactions*, 48(2), 158-169.

Orda, A., & Rom, R. (1990). Shortest-path and Minimum Delay Algorithms in Networks with Time-dependent Edge-length. *Journal of the ACM,* 37(3), 607–625.

Sherali, H. D., Ozbay, K., & Subramanian, S. (1998). The Time-Dependent Shortest Pair of Disjoint Paths Problem: Complexity, models, and algorithms. *Networks*, 31(4), 259–272.

Shuai, Z., & Chao, M. (2012), Model and Heuristics for the Heterogeneous Fixed Fleet Vehicle Routing Problem with Pick-Up and Delivery. *Journal of Distribution Science*, 10(12), 19-24.

Song, J. G., & Kim, T. R. (2012).The Performance Formation Model of Service Quality Factors for Courier Service. *Journal of Distribution Science*, 10(4), 37-45.

Yajun, Y., Hong, G., Jeffrey, X., & Jianzhong, L. (2014). Finding the Cost Optimal Path with Time Constraint over Time Dependent Graphs. *The Proceedings of the VLDB Endowment (PVLDB),* 7(9), 673-684.