Information Sciences 255 (2014) 135–154

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Processing time-dependent shortest path queries without pre-computed speed information on road networks ☆

Jinha Kim, Wook-Shin Han, Jinoh Oh, Sungchul Kim, Hwanjo Yu *

Pohang University of Science and Technology (POSTECH), Pohang, South Korea

A R T I C L E I N F O

Article history:
Received 11 March 2013
Accepted 22 July 2013
Available online 27 July 2013

Keywords:
Shortest path
Road network
Routing

A B S T R A C T

Shortest path (or least travel time path) identification has been actively studied for direct application to road networks. In addition, the processing of time-dependent shortest-path queries, which use past traffic data to compute the speed variations of road segments, has been investigated in order to incorporate speed variations over time. However, speed information pre-computed from static past traffic data is often invalid because road traffic is inherently dynamic.

This paper addresses a new problem in processing a Dynamic Time-Dependent Shortest Path (DTDSP) query, which considers the current road status without assuming pre-determined speed patterns on roads. By dynamically adjusting the speed patterns of roads instead of fixing them based on past traffic data, the recommended paths, which reflect the current road status, are more effective in distributing the road traffic and thus reducing the travel time.

To process DTDSP queries, we first propose a Continuous Piece-wise Linear Speed Pattern (CPLSP) model to compute the vehicle speed patterns, which is more flexible and realistic than previously adopted piece-wise constant speed pattern models. Using dynamically computed CPLSPs, we process a DTDSP query in two phases: (1) the least travel time path is found for the query and (2) the speed patterns of the following vehicles, which are affected by the participation of the new vehicle on the road network, are updated. We propose efficient algorithms for finding the least travel time path of a new query (vehicle) and for updating the speed patterns of the existing vehicles. Experiments on real data sets show that our query processing algorithms effectively distribute road traffic, and thus, significantly reduce both global and individual travel times.

1. Introduction

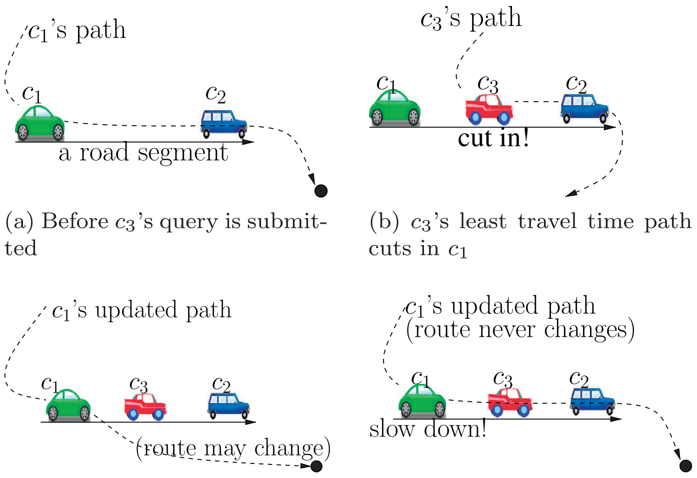
Identifying the shortest path is important for its direct application to road networks. When commuting or traveling, we want to minimize the time spent on roads. The shortest path query abstracts this natural demand and has been actively studied. The most basic approach is to find the path with the shortest length based on a map. This corresponds to the

☆ This research was partially supported by Next-Generation Information Computing Development Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (MEST) (No. 2012M3C4A7033344). This work was also supported by Mid-career Researcher Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (MEST) (No.KRF-2011-0016029).

* Corresponding author. Tel.: +82 542792388.

E-mail addresses: goldbar@postech.ac.kr (J. Kim), wshan.postech@gmail.com (W.-S. Han), kurin@postech.ac.kr (J. Oh), subright@postech.ac.kr (S. Kim), hwanjoyu@postech.ac.kr (H. Yu).

0020-0255/$ - see front matter © 2013 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.ins.2013.07.009



(a) Before c_3's query is submitted

(b) c_3's least travel time path cuts in c_1

(c) Arrival Time Priority algorithm

(d) Query Time Priority algorithm

Fig. 1. Algorithms for DTDSP.

classical shortest path query [8,11,3,10,25,1,2,26,12,15,14,20,21,13,19,22]. However, this approach does not recommend good paths, particularly during rush hour or holiday seasons, since it assumes that every road segment has a constant driving speed.

To reflect real road network conditions, variants of the shortest path query have been proposed. If travelers utilize the future speed pattern of each road segment over time along with a map, they can find a better path, i.e., the least travel time path. This setting corresponds to the time-dependent shortest path query [7,18,17,23]. Moreover, when the starting time can be chosen flexibly within a certain range, travelers may also want to know the best starting time that results in the least travel time. This is a time-dependent shortest path with starting time range query [18,5,6,16,9]. However, these queries assume that future speed patterns are pre-determined from the past traffic data. Such speed patterns are easily invalidated as more vehicles request the shortest paths. During the pre-computation phase, a high speed is likely assigned to uncrowded road segments. These road segments tend to be frequently included in the shortest path selection. The chosen road segments then become crowded, and their speeds will decrease.

This paper addresses a new problem (or new query) of finding the least travel time path dynamically by considering the *current* road status without assuming pre-determined speed patterns on the roads. We call this a *Dynamic Time-Dependent Shortest Path* (DTDSP) query. Its main difference from existing time-dependent shortest path queries is that, instead of fixing the speed patterns of roads based on past traffic data, the speed patterns are adjusted dynamically as new vehicles set up their paths. Thus, the recommended routes, reflecting the current road status, are better in distributing the road traffic and thus reducing the travel time.

A DTDSP query is applicable to a server–client path recommendation system. When a client such as an autonomous car requests the best path from the source to destination starting at a specific time, the server system monitors the traffic condition from a holistic view and recommends the best path. This kind of scenario is frequently illustrated in many science-fiction movies.

To process a DTDSP query, we first propose a Continuous Piece-wise Linear Speed Pattern (CPLSP) model to compute the vehicle speed patterns. The CPLSP model is more flexible and realistic than previously adopted piece-wise constant speed pattern models [23]. We dynamically compute a CPLSP based on the maximum speed limit and the safe distance between two successive vehicles. These assumptions are reasonable owing to the advances of a vehicle-to-vehicle (V2V) communication system [24] and an autonomous cruise control system.[1]

Based on dynamically computed CPLSPs, we process a DTDSP query in two phases. The first phase is finding the least travel time path for the query. We propose an efficient path finding algorithm that exploits a novel admissible heuristic estimator based on landmarks. The second phase is updating the speed patterns of the following vehicles that are affected by the participation of a new vehicle on the road network. After computing the least travel time path of the current query in the first phase, the speed patterns that constitute the path are registered in the road network. Newly registered speed patterns of the current query may cut into the existing speed patterns of the past queries. As illustrated in Example 1, these affected speed patterns of past queries should be updated, and their paths may also be re-calculated.

[1] http://en.wikipedia.org/wiki/Autonomous_cruise_control_system.