Design of an in-memory index for fast map matching in electric vehicle services

Junghoon Lee[1*], Jihyun Kang[1], Hyoung Sin Park[1], Hye-Jin Kim[1]

[*1] Dept. of Computer Science and Statistics,
Jeju National University, Republic of Korea
{jhlee,hjkim82}@jejunu.ac.kr, {gudtls1204,nativegod12}@naver.com

Abstract. This paper designs a graph-style complementary index for fast map matching, aiming at providing an intelligent service framework for electric vehicles. For a sequential shape file containing about 25,000 links, the proposed in-memory index consists of nodes and links, where each link contains two end points and offset in the shape file. By high speed retrieval of the neighbor links based on graph traversal, the map matching procedure can begin the search from the previously matched link in a spatio-temporal movement stream, significantly narrowing its search space. The offset of the link allows the procedure to randomly access the shape file and avoid unnecessary segment-by-segment distance calculation by investigating the bounding box of a link record. Using this index, we can build a real-time analyzer for battery dynamics and thus a battery discharge model for electric vehicles even on mobile devices.

Keywords: Smart transportation, electric vehicle, map matching, graph index, road network

1 Introduction

Electric vehicles, or EVs in short, are considered to be one of the most important elements in the future transport system, as they can reduce air pollution and improve energy efficiency [1]. Not only cheap nuclear energy but also eco-friendly renewable energy can be exploited for their power sources. However, due to the capacity limitation in EV batteries, the driving distance is practically at most 100 *km* and it takes 30 ~ 40 minutes to charge an EV even with a fast charger [2]. The battery discharge is affected by so many factors such as road shapes, altitude changes, drivers' behaviors, air conditioner operations, and the like. Hence, it is necessary to estimate current battery remaining, or interchangeably SoC (State of Charge), and the reachable distance on the way to the destination [3]. If the current SoC is not enough, the driver must have his or her EV charged.

This work (Grants No. C0026912) was supported by Business for Cooperative R&D between Industry, Academy, and Research Institute funded by Korea Small and Medium Business Administration in 2012.

In the mean time, such a service can be developed by an application running on in-vehicle computers such as telematics devices, just as many other smart grid components take advantage of computational intelligence and communication networks [4]. Moreover, GPS receivers make the location information available to the application. Basically, by the current coordinate of an EV, it is possible to know the road segment the EV is currently moving on. The process of binding the coordinate to a road segment is map matching. For a given coordinate, mainly specified by longitude and latitude in the world-wide coordinate system, the map matching procedure searches the set of road segments to find the match. Each road segment is represented by a series of coordinates following the shape of roads.

For a fast moving EV, map matching speed is important to give more room for the execution of other complex and time-consuming applications such as SoC analyzers and inference engines [5]. Basically, the efficiency of map matching processes is subject to how the road segments are organized and thus how many comparisons are needed to find the match [6]. For a moving object which creates a spatio-temporal trajectory, the current location is not far away from the past location. Hence, it is reasonable to start the map matching procedure from the road segment to which the past coordinate is matched. In this regard, this paper designs a complementary index for fast map matching, aiming at providing an efficient EV service framework capable of hosting many sophisticated applications.

2 Indexing scheme

To begin with, a road network is comprised of intersections and links connecting them. EVs can move only along the links, each of which has its own shape. While just a single point is enough to represent an intersection, a link needs a series of points to trace its shape. As the curve is approximated by a series of lines, plotting error is unavoidable. Map matching procedure checks whether a given point is on a line segment for each link by calculating the distance between them. So, if the road network has a lot of links and they have complex shapes, the map matching time will be overwhelming. Our system is targeting at Jeju City, Republic of Korea. As a middle-level city, it has about 18,000 intersections and 27,000 links. This city is ambitiously pursing the large deployment of EVs as a part of smart grid model city enterprise.

A road layout is stored in ESRI shape files. Sometimes, this shape record is converted to a spatial database records for better retrieval and management. However, for the city-level road networks, it is possible to build a complementary index for efficient map matching, relieving a burden of maintaining a complex and expensive database system. The size of its link shape file depends on the pattern of road layout. Figure 1 shows the actual road shape of Jeju City. In downtown area, the road segment is generally straight, possibly specified just by 2 points. On the contrary, in the mountain and seaside area, the roads are winding and needs a number of points. In shape file, each link record is stored sequentially. The shape file can have a general-purpose index, but it can support an efficient move-forward and backward.

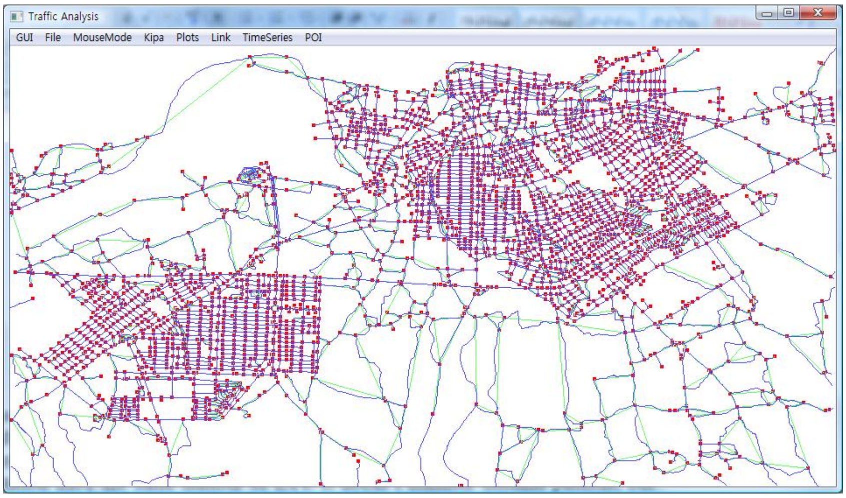

Fig.1. Road layout

Our design exploits an in-memory graph which includes intersections and links as shown in Figure 2. Each intersection, or node, has the set of links emanating from it. Each link has two end nodes and offset in the shape file. The size of a link record is different by the number of points in the link. According to the shape file format, a link record begins with the coordinates of its bounding box, namely, x-max, x-min, y-max, and y-min. Here, x and y means longitude and latitude for simplicity. If the match point is included in this boundary, it is necessary to check if the point is on a line segment of the link one by one. Otherwise, we can skip this link. For a line segment, the distance from the match point is calculated and if it is less than the small bound, the link match is found.

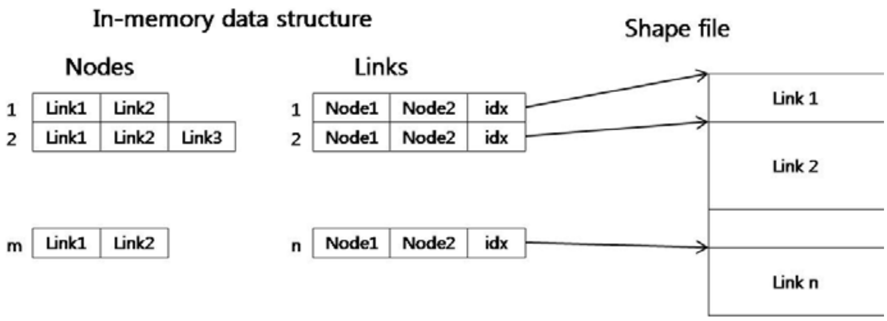

Fig. 2. Index architecture

The graph index allows the map matching procedure to traverse the shape file according to the connection architecture of the road network, taking advantage of existing graph search algorithms such as Dijkstra's and breadth-first schemes. That is, if the procedure investigates a link but does not find a match, it will proceed to the adjacent links. The graph index contains the two end points of a link, so the connected links can be easily found via them. As the in-memory index includes the offset in the shape file, the map matching procedure can randomly access the file. Practically, adjacent links are stored in the shape file not far away from each other. As operating systems read records from the disk system by the block, it is high likely that additional disk access is not necessary.

3 Conclusions

The smart grid pursues energy efficiency in many areas and also in transportation systems. Here, EVs are the key element of the smart transportation. To overcome the range anxiety, it is necessary to build a SoC dynamics model for a set of popular routes. It can inform drivers of how much battery power will be consumed on each road, making it possible to make a charging plan during the trip. This service binds battery remaining to the road network, so map matching is the most fundamental building block. This paper has designed a graph-based complementary index for the link shape file to speed-up the neighbor link searches, based on the observation that the current location is not far away from the previous location. Graph traversal schemes gradually extend the search area until it finds a link match from the starting point. In the mean time, the offset field allows us to avoid unnecessary segment-by-segment on-the-line tests, significantly enhancing the map matching speed.

References

1. Sortomme, E., Hindi, M., MacPherson, S., Venkata, S.: Coordinated Charging of Plug-in Hybrid Electric Vehicles to Minimize Distribution System Losses. IEEE Transactions on Smart Grid, 198-205 (2012)
2. Botsford, C., Szczepanek, A.: Fast Charging vs. Slow Charging: Pros and Cons for the New Age of Electric Vehicles. In: International Battery Hybrid Fuel Cell Electric Vehicle Symposium (2009)
3. Trembly, O., Dessaing, L.: Experimental Validation of a Battery Dynamic Model for EV Applications. World Electric Vehicle Journal, Vol. 3 (2009)
4. Ramchurn, S., Vytelingum, R., Rogers, A., Jennings, N.: Putting the 'Smarts' into the Smart Grid: A Grand Challenge for Artificial Intelligence", Communications of The ACM, Vol. 55, No. 4, 86-97 (2012)
5. Lee, J., Kim, H., Park, G.: Relocation Action Planning in Electric Vehicle Sharing Systems. In: C. Sombattheera et al. (eds.) MIWAI 2012, LNCS, vol. 7694, pp. 47-56. Springer, Heidelberg (2012)
6. Lee, J., Kim, H., Park, G.: Integration of Battery Charging to Tour Schedule Generation for an EV-Based Rent-a-Car Business. In: Y. Tan, Y. Shi, and Z. Ji (Eds.): ICSI 2012, Part II, LNCS, vol. 7332, pp. 399-406. Springer, Heidelberg (2012)