ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

A novel method for Online Shortest Path

M.Prathibha Bharathi
M.Tech,
Department of Software Engineering,
Vinuthna Institute of Technology and sciences,
Warangal.

T. Moulika
Assistant Professor,
Department of Computer Science & Engineering,
Vinuthna Institute of Technology and sciences,
Warangal.

Abstract:

The online shortest path problem aims at computing the shortest path based on live traffic circumstances. This is very important in modern car navigation systems as it helps drivers to make sensible decisions. To our best knowledge, there is no efficient system/solution that can offer affordable costs at both client and server sides for online shortest path computation. Unfortunately, the conventional client-server architecture scales poorly with the number of clients. A promising approach is to let the server collect live traffic information and then broadcast them over radio or wireless network.

This approach has excellent scalability with the number of clients. Thus, we develop a new framework called live traffic index (LTI) which enables drivers to quickly and effectively collect the live traffic information on the broadcasting channel. An impressive result is that the driver can compute/update their shortest path result by receiving only a small fraction of the index. The experimental study shows that LTI is robust to various parameters and it offers relatively short tune-in cost (at client side), fast query response time (at client side), small broadcast size (at server side), and light maintenance time (at server side) for online shortest path problem.

Index Terms:

Shortest path, air index, broadcasting.

1. INTRODUCTION:

Shortest path computation is an important function in modern car navigation systems .This function helps a driver to figure out the best route from his current position to destination.

Typically, the shortest path is computed by offline data pre-stored in the navigation systems and the weight (travel time) of the road edges is estimated by the road distance or historical data. Unfortunately, road traffic circumstances change over time. Without live traffic circumstances, the route returned by the navigation system is no longer guaranteed an accurate result. Those old navigation systems would suggest a route based on the pre-stored distance information .Note that this route passes through four road maintenance operations (indicated by maintenance icons) and one traffic congested road (indicated by a red line). Nowadays, several online services provide live traffic data (by analyzing collected data from road sensors, traffic cameras, and crowdsourcing techniques), These systems can calculate the snapshot shortest path queries based on current live traffic data; however, they do not report routes to drivers continuously due to high operating costs. Answering the shortest paths on the live traffic data can be viewed as a continuous monitoring problem in spatial databases, which is termed online shortest paths computation (OSP) in this work.

To the best knowledge, this problem has not received much attention and the costs of answering such continuous queries vary hugely in different system architectures. Typical client-server architecture can be used to answer shortest path queries on live traffic data. In this case, the navigation system typically sends the shortest path query to the service provider and waits the result back from the provider (called result transmission model). However, given the rapid growth of mobile devices and services, this model is facing scalability limitations in terms of network bandwidth and server loading. Based on a telecommunication expert theworld's cellular networks need to provide 100 times the capacity in 2015 when compared to the networks in 2011. Furthermore, live traffic are updated frequently as these data can be collected by using crowdsourcing techniques (e.g., anonymous traffic data from Google map users on certain mobile devices).

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

As such, huge communication cost will be spent on sending result paths on the this model. Obviously, the client-server architecture will soon become impractical in dealing with massive live traffic in near future. client-server architecture, it cannot scale well with a large number of users.In addition, the reported paths are approximate results and the system does not provide any accuracy guarantee. An alternative solution is to broadcast live traffic data over wireless network (e.g., 3G, LTE, Mobile WiMAX, etc.). The navigation system receives the live traffic data from the broadcast channel and executes the computation locally (called raw transmission model).

The traffic data are broadcasted by a sequence of packets for each broadcast cycle. To answer shortest path queries based on live traffic circumstances, the navigation system must fetch those updated packets for each broadcast cycle. The main challenge on answering live shortest paths is scalability, in terms of the number of clients and the amount of live traffic updates. A new and promising solution to the shortest path computation is to broadcast an air index over the wireless network (called index transmission model).

The main advantages of this model are that the network overhead is independent of the number of clients and every client only downloads a portion of the entire road map according to the index information. For instance, the proposed index constitutes a set of pairwise minimum and maximum traveling costs between every two subpartitions of the road map. However, these methods only solve the scalability issue for the number of clients but not for the amount of live traffic updates.

As reported the re computation time of the index takes 2 hours for the San Francisco (CA) road map. It is prohibitively expensive to update the index for OSP, in order to keep up with live traffic circumstances. Motivatedby the lack of off-the-shelf solution for OSP, Anew solution based on the index transmission model by introducing live traffic index (LTI) as the core technique. LTI is expected to provide relatively short tune-in cost (at client side), fast query response time (at client side), small broadcast size (at server side), and light maintenance time (at server side) for OSP.

LTI features as follows.

- The index structure of LTI is optimized by two novel techniques, graph partitioning and stochastic-based construction, after conducting a thorough analysis on the hierarchical index techniques. To the best of our knowledge, this is the first work to give a thorough cost analysis on the hierarchical index techniques and apply stochastic process to optimize the index hierarchical structure.

- LTI efficiently maintains the index for live traffic circumstances by incorporating Dynamic Shortest Path Tree (DSPT) into hierarchial index techniques.

- LTI reduces the tune-in cost up to an order of magnitude as compared to the state-of-the-art competitors; while it still provides competitive query response time, broadcast size, and maintenance time. To the best of our knowledge, we are the first work that attempts to minimize all these performance factors for OSP.

2.LITERATURE SURVEY :

2.1.Spectral Clustering Based on The Graph Laplacian

A connection between the Cheeger cut and the second eigenvectorof the graph p-Laplacian, a nonlinear generalization of the graph Laplacian.A p-Laplacian which is slightly from the one used. Has been used for semisupervised learning .The main motivation for the use of eigenvectors of the graph p-Laplacian was the generalized isoperimetric inequality. In which relates the second eigenvalue of the graph p-Laplacian to the optimal Cheeger cut. The isoperimetric inequality becomes tight as p , so that the second eigenvalue converges to the optimal Cheeger cut value.

2.2. SHARC: Fast and Robust Unidirectional Routing

Introduce SHARC-Routing, a fast and robust approach for unidi-rectional routing in large networks. The central idea of SHARC (Shortcuts + Arc-Flags) is the adaptation of techniques developed for Highway Hierarchies to Arc-Flags.In general, SHARC-Routing iteratively constructs a contraction-based hierarchy during preprocessing and automatically sets arc- ags for edges removed during contraction.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

More precisely, arc- ags are set in such a way that a unidirectional query considers these removed component-edges only at the beginning and the end of a query. As a result, able to route very efficiently in scenarios where other techniques fail due to their bidirectional nature. It turned out that SHARC was a promising candidate for routing in time-dependent networks.

2.3.Computing point to point shortest path from External Memory

The ALT algorithm for the point-to-point shortest path problem in the context of road networks. The suggest improvements to the algorithm itself and to its preprocessing stage. Also develop a memory-efficient implementation of the algorithm that runs on a Pocket PC(Personal Computer).It stores graph data in a ash memory card and uses RAM(Random Access Memory) to store information only for the part of the graph visited by the current shortest path computation. Theimplementation works even on very large graphs, including that of the North America road network, with almost 30 million vertices.

2.4 Time-Dependent SHARC-Routing

During the last years, many speed-up techniques for Dijkstra's algorithm have been developed. As a result, computing a shortest path in a static road network is a matter of microseconds. However, only few of those techniques work in time-dependent networks. Unfortunately, such networks appear frequently in reality.

2.5. Shortest Path Tree Computation in Dynamic Graphs

The Dynamic Shortest Path (DSP) problem is to compute S from D.This problem either focuses on a single edge weight change, or for multiple edge weight changes, some of them are incorrect or are not optimized. The correct and extend a few state-of-the-art dynamic SPT algorithms to handle multiple edge weight updates. Hence prove that these algorithms are correct. Dynamic algorithms may not out perform static algorithms all the time. To evaluate the proposed dynamic algorithms, compare them with the well-known static Dijkstra's algorithm.

3. CONCLUSION:

The online shortest path computation; the shortest path result is computed/updated based on the live traffic circumstances. Analyze the existing work and discuss their inapplicability to the problem (due to their prohibitive maintenance time and large transmission overhead). To address the problem, suggest a promising architecture that broadcasts the index on the air. First identifyan important feature of the hierarchical index structure which enables us to compute shortest path on a small portion of index. This important feature is thoroughly used in our solution, LTI. The experiments confirm that LTI is a Pareto optimal solution in terms of four performance factors for online shortest path computation. In the future, Extend this solution on time dependent networks. This is a very interesting topic since the decision of a shortest path depends not only on current traffic data but also based on the predicted traffic circumstances.

REFERENCES:

[1] T. Buhler and M. Hein, "Spectral Clustering Based on The Graph Laplacian," Proc. Int'l

Conf. Machine Learning (ICML), p. 11, 2009.

[2]R. Bauer and D. Delling, "SHARC: Fast and Robust Unidirectional Routing,"pp. 13-26, 2008.

[3]A.V. Goldberg and R.F.F. Werneck, "Computing Point to point Shortest Path from External Memory", 2005.

[4]Daniel Delling,"Time-Dependent SHARC-Routing", 2008

[5] E.P.F. Chan and Y. Yang, "Shortest Path Tree Computation in Dynamic Graphs", Apr, 2009.