

Efficient Shortest Distance Query Processing and Indexing on Large Road Network

by

DIAN OUYANG

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Centre for Artificial Intelligence (CAI)
Faculty of Engineering and Information Technology (FEIT)
University of Technology Sydney (UTS)

February, 2019

CERTIFICATE OF AUTHORSHIP / ORIGINALITY

I certify that the work in this thesis has not previously been submitted for a degree nor has it been submitted as part of requirements for a degree except as fully acknowledged within the text.

I also certify that the thesis has been written by me. Any help that I have received in my research work and the preparation of the thesis itself has been acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

This research is supported by the Australian Government Research Training Program

Signature of Candidate

Production Note:
Signature removed
prior to publication.

ACKNOWLEDGEMENTS

First and foremost, I would like to express my sincere gratitude to my supervisor Dr. Lu Qin for his continuous encouragement and guidance of my PhD career. He is professional, kind, patient and wise. He introduces knowledge of database area and research skill to me. His creative ideas extends thinking and horizon for me. Additionally, Lu is a good mentor and friend for me. He shares his experiment of research and life to me. Thanks to his help and encouragement, I am always confident and brave when experiencing challenge and trouble. This thesis could not reach its present form without his whole-heartedly instructions.

Secondly, I would like to deliver my great gratitude to my co-supervisor Prof. Ying Zhang for his constant encouragement and guidance, especially supporting for my academic career. His brilliant ideas and inspirations extent my research area and mode of thinking. He is also farsighted and provide efficient advise for career.

Thirdly, I would like to thank Prof. Xuemin Lin, Dr. Lijun Chang and Dr. Long Yuan for supporting the works in this thesis, as most of the works are conducted in collaboration with them. I thank Prof. Lin for letting me join his research group. I learn writing and researching skill from Prof Lin. I thank Dr. Chang for

his brilliant ideas and support for academic career to me. His wonderful research works inspires my research work in this thesis. I thank Dr. Long Yuan for his advise of writing in this thesis.

I would also like to thank the following people: Prof. Qin Zhu, Prof. Wenjie Zhang, Dr. Xin Cao, Dr. Zengfeng Huang, Dr. Xiaoyang Wang, Dr. Shiyu Yang, for sharing interesting ideas and knowledge. Thanks to Dr. Fan Zhang, Dr. Longbin Lai, Dr. Xing Feng, Dr. Xiang Wang, Dr. Jianye Yang, Dr. Fei Bi, Mr. Xubo Wang, Mr. Wei Li, Mr. Haida Zhang, Ms. Chen Zhang, Ms. Xiaoshuang Chen, Mr. Yang Yang, Mr. Kai Wang, Mr. You Peng, Mr. Hanchen Wang, Mr. Xuefeng Chen, Mr. Boge Liu, Mr. Dong Wen, Mr. Wentao Li, Mr. Bohua Yang, Mr. Mingjie Li, Ms. Conggai Li, Mr. Daokun Zhang, Mr. Fujin Zhu, Mr. Adi Lin for sharing the happiness with me in my PhD career.

Last but not least, I would like to thank my father Mr. Jianzhi Ouyang and my mother Mrs. Meihua Tan, for giving me great living and studying environment. Thanks for their love and supporting during the Ph.D studying.

ABSTRACT

Computing the shortest distance between two vertices is a fundamental problem on road networks. State-of-the-art indexing-based solutions can be categorized into hierarchy-based solutions and hop-based solutions. However, the hierarchy-based solutions require a large search space for long-distance queries while the hop-based solutions result in a high computational waste for short-distance queries. Moreover, in real life, the weight of edges changes frequently. For example, building a road need several months, but the travel time of road changes frequently such as traffic jam in the morning peak. We model this problem as the shortest path problem on a dynamic road network. The existing solutions are not efficient to update the index for the dynamic condition. Shortest path query on bicriteria road network is another important and practical problem in real life. To compute shortest path between any two vertices, we can get the shortest path set which is called path skyline. We propose an efficient exploring strategy to accelerate path skyline computing.

We propose a novel hierarchical 2-hop index (H2H-Index) which assigns a label for each vertex and at the same time preserves a hierarchy among all vertices. With the H2H-Index, we design an efficient query processing algorithm with performance guarantees by visiting part of the labels for the source and destination based on the vertex hierarchy. We also propose an algorithm to construct the H2H-Index based on distance preserved graphs. The algorithm is further optimized by computing the labels based on the partially computed

labels of other vertices.

We use dynamic road network to define the graph model whose topological structure is stable and weight of edges changes frequently. In this model, we have two processing, shortest path query and road update processing, to do on road network. We use Contraction Hierarchies which is one of art-of-the-state index algorithm for shortest path problem to answer queries. And propose an efficient index updating algorithm to update CH index for road updating processing. In contrast to vertex centric algorithm, our shortcut centric algorithm has better theoretical bound.

In the literature, PSQ is a fundamental algorithm for path skyline query and is also used as a building block for the afterwards proposed algorithms. In PSQ, a key operation is to record the skyline paths for each node v that is possible on the skyline paths from s to t . However, to obtain the skyline paths for v , PSQ has to maintain other paths that are not skyline paths for v , which makes PSQ inefficient. Motivated by this, in this chapter, we propose a new algorithm PSQ⁺ for the path skyline query. By adopting an ordered path exploring strategy, our algorithm can totally avoid the fruitless path maintenance problem in PSQ.

We conducted extensive performance studies using large real road networks including the whole USA road network. The experimental results demonstrate that our approach can make significant improvement to every problem.

PUBLICATIONS

- *Dian Ouyang, Lu Qin, Lijun Chang, Xuemin Lin, Ying Zhang, and Qing Zhu, When Hierarchy Meets 2-Hop-Labeling: Efficient Shortest Distance Queries on Road Networks, to appear in SIGMOD 2018.*
- *Dian Ouyang, Long Yuan, Fan Zhang, Lu Qin and Xuemin Lin, Towards Efficient Path Skyline Computation in Bicriteria Networks, to appear in DASFAA 2018.*
- *Dian Ouyang, Long Yuan, Lu Qin, Ying Zhang and Lijun Chang, Efficient Shortest Distance Query Processing on Dynamic Road Networks with Theoretical Guarantees, in submission*

TABLE OF CONTENT

CERTIFICATE OF AUTHORSHIP/ORGINALITY	ii
ACKNOWLEDGEMENTS	iii
ABSTRACT	v
PUBLICATIONS	vii
TABLE OF CONTENT	viii
LIST OF FIGURES	x
LIST OF TABLES	xi
Chapter 1 INTRODUCTION	1
1.1 H2H Index for Shortest Path Distance	2
1.2 Shortest Path Query on Dynamic Roadnetwork	5
1.3 Efficient Path Skyline Computation in Bicriteria Networks	7
1.4 Graph Model and Preliminaries	11
Chapter 2 LITERATURE REVIEW	13
2.1 Shortest distance on road networks.	13
2.2 Tree decomposition and treewidth.	14
2.3 Shortest Distance in Other Networks.	15
2.4 Bicriteria Shortest Path Problem	16
2.5 Shortest path in dynamic road networks	18
Chapter 3 H2H index for shortest path distance query	19
3.1 Overview	19
3.2 Problem Statement	20
3.3 Existing Solutions	21
3.3.1 Hierarchy-based Solution	21
3.3.2 Hop-based Solution	24

3.4	The Basic Idea	26
3.4.1	Problem Analysis.	26
3.4.2	Vertex Hierarchy by Tree Decomposition	29
3.4.3	A Naive Solution	33
3.5	Query Processing by H2H-Index	35
3.6	H2H-Index Construction	41
3.6.1	Index Construction using DP-Graph	41
3.6.2	Index Construction using Partial Labels	50
3.7	Performance Studies	54
3.8	Chapter Summary	61
Chapter 4 Efficient Shortest Distance Query on Dynamic Road Networks		62
4.1	Overview	62
4.2	Preliminaries	62
4.2.1	Contraction Hierarchies	63
4.2.2	Problem Statement	66
4.3	The State of the art	67
4.4	Streaming Updates Algorithm	71
4.4.1	A Shortcut-Centric Paradigm	71
4.4.2	Weight-Updated Shortcuts Bounded Algorithms	73
4.4.3	Algorithms without Materialized SS-Graph	87
4.5	Performance Studies	90
4.6	Chapter Summary	93
Chapter 5 Path Skyline on Bicriteria Network		96
5.1	Overview	96
5.2	Preliminaries	96
5.3	Existing Solution	97
5.4	Our New Approach	100
5.5	Performance Studies	105
5.6	Chapter Summary	110
Chapter 6 EPILOGUE		112
BIBLIOGRAPHY		114

LIST OF FIGURES

1.1	Path Skyline Query	8
3.1	An Example of a Road Network G	20
3.2	The index structure G_I of CH	23
3.3	The Search Space for CH ($q = (v_{19}, v_{12})$)	27
3.4	Tree Decomposition T_G of the Network in Fig. 3.1	30
3.5	The H2H-Index for the Road Network in Fig. 3.1	38
3.6	Query Processing Examples using the H2H-Index	41
3.7	Proof of Lemma 2	43
3.8	Distance Preserved Tree Decomposition T_G	46
3.9	DP-Graphs $G(v_{20})$ and $G(v_3)$	47
3.10	Process to Calculate $X(v_4).dis$	54
3.11	Query Processing Time (Varying Query Distance)	55
3.12	Query Processing Time (Varying Dataset Size)	56
3.13	Indexing Time and Index Size Testing	57
4.1	A Road Network	63
4.2	Shortcut Graph G_{sc}	65
4.3	new CH graph after (v_4, v_6) 's weight decreasing	68
4.4	new CH graph after (v_9, v_{10}) 's weight increasing	70
4.5	SS-Graph G^*	75
4.6	new CH graph after (v_7, v_{10}) 's weight increasing	86
4.7	SS-Graph with $c_\phi(v_s)$ G^*	86
4.8	Updating Processing Time (Varying extent of decline)	93
4.9	Updating Processing Time (Varying increasing rate)	94
5.1	Percentage of time used for the maintenance of $v.Skyline$ in PSQ	100
5.2	PSQ and PSQ ⁺	103
5.3	Query Processing time (Varying Query Location)	107
5.4	Query Processing Time (Varying Dataset Size)	109

LIST OF TABLES

1.1	Notations	12
3.1	2-hop Labels for the road network in Fig. 3.1	25
3.2	Datasets used in the Experiments	57
4.1	Space consumption of G' vs G^*	88
4.2	Datasets used in Experiments	91
5.1	Datasets used in Experiments	106

LIST OF TABLES
