International Journal of Research
Available at https://edupediapublications.org/journals
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 04 Issue 09
August 2017

A Review on Efficient Cache-Supported Path Planning on Roads (Ppr)

Chitturi Prasad & B Venkata Ramana
[1]M.Tech, Department of CSE Priyadarshini Institute of Technology and Science, Chintalapudi, India
[2]Assistant professor, Department of CSE Priyadarshini Institute of Technology and Science, Chintalapudi, India
[1]Chprasad4u@gmail.com, [2]bvr9989@gmail.com

ABSTRACT:

In present day's broad accessibility and easy availability of the global positioning system (GPS) and digital mapping of roads, road network navigation services have become a basic application on many mobile devices. Path planning (PP), a basic purpose of road network navigation services, discover a route amid the particular start location and end location. The competence of this path planning function is grave for mobile users on roads due to different dynamic scenarios, such as an unexpected change in driving direction, unpredicted traffic situation, lost or unstable GPS signals, and so on. In these scenarios, the path planning service needs to be delivered in a timely fashion. In this paper, we recommend a system, explicitly, Path Planning by Caching (PPC), to response a new path planning query in real time by proficiently caching and reusing historical queried-paths. Unlike the predictable cache-based path planning systems, where a queried-path in cache is used only when it matches perfectly with the new query, PPC leverages the partially matched queries to answer part(s) of the new query.

KEYWORDS: Spatial database, path planning, cache

RELATED WORK:

"Shared execution of path queries on road networks," Authors: H. Mahmud, A. M. Amin, M. E. Ali, and T. Hashem

The advancement of mobile technologies and the proliferation of map-based applications have enabled a user to access a wide variety of services that range from information queries to navigation systems. Due to the popularity of map-based applications among the users, the service provider often requires to answer a large number of simultaneous queries. Thus, processing queries efficiently on spatial networks (i.e., road networks) have become an important research area in recent years. In this paper, we focus on path queries that find the shortest path between a source and a destination of the user. In particular, we address the problem of finding the shortest paths for a large number of simultaneous path queries in road networks. Traditional systems that consider one query at a time are not suitable for many applications due to high computational and service costs. These systems cannot guarantee required response time in high load conditions. We propose an efficient group based approach that provides a practical solution with reduced cost. The key concept for our approach is to group queries that share a common travel path and then compute the shortest path for the group. Experimental results show that our approach is on an average ten times faster than the traditional approach in return of sacrificing the accuracy by 0.5% in the worst case, which is acceptable for most of the users.

"An efficient path computation model for hierarchically structured topographical road maps,"

In this paper, we have developed a HiTi (Hierarchical MulTi) graph model for structuring large topographical road maps to speed up the minimum cost route computation. The HiTi graph model provides a novel approach to abstracting and structuring a topographical road map in a hierarchical fashion. We propose a new shortest path algorithm named SPAH, which utilizes HiTi graph model of a topographical road map for its computation. We give the proof for the optimality of SPAH. Our performance analysis of SPAH on grid graphs showed that it significantly reduces the search space over existing methods. We also present an in-depth experimental analysis of HiTi graph method by comparing it with other similar works on grid graphs. Within the HiTi graph framework, we also propose a parallel shortest path algorithm named ISPAH. Experimental results show that inter query shortest path problem provides more opportunity for scalable parallelism than the intra query shortest path problem.

"Computing the shortest path: A search meets graph theory," Authors: A. V. Goldberg and C. Harrelson

We propose shortest path algorithms that use A search in combination with a new graph-theoretic lower-bounding technique based on landmarks and the triangle inequality. Our algorithms compute optimal shortest paths and work on any directed graph. We give experimental results showing that the most efficient of our new algorithms outperforms previous algorithms, in particular A* search with Euclidean bounds, by a wide margin on road networks and on some synthetic problem families.

"Reach-based routing: A new approach to shortest path algorithms optimized for road networks," Authors: R. Gutman

The problem of identifying the shortest path along a road network is a fundamental problem in network analysis, ranging from route guidance in a navigation system to solving spatial allocation problems. Since this type of problem is solved so frequently, it is important to craft an approach that is as efficient as possible. Based upon past research, it is generally accepted that several efficient implementations of the Dijkstra algorithm are the fastest at optimally solving the 'one-to-one' shortest path problem We show that the most efficient state-of-the-art implementations of Dijkstra can be improved by taking advantage of network properties associated with GIS-sourced data. The results of this paper, derived from tests of different algorithmic approaches on real road networks, will be extremely valuable for application developers and researchers in the GIS community.

DISADVANTAGES:

- A cached query is returned only when itmatches completely with a new query.
- The time complexity is high.
- The cache content may not be up to date to respond to recent trends in issued queries.
- The cost of constructing a cache is high, since the system must calculate the benefitvalues for all sub-paths in a full-path of query results.

PROPOSED SYSTEM:

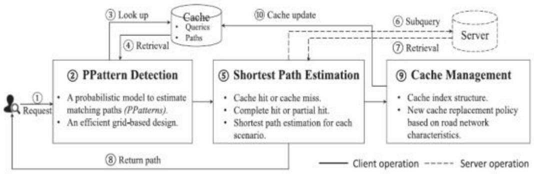
To meet existing need, we propose asystem, namely, Path Planning by Caching (PPC), that aimsto answer a new path planning query efficiently by cachingand reusing historically queried

International Journal of Research
Available at https://edupediapublications.org/journals
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 04 Issue 09
August 2017

paths (queried-paths in short). The proposed system consists of three main components: (i) PPattern Detection,(ii) Shortest Path Estimation, and (iii) Cache Management. Given a path planning query, which contains asource location and a destination location, PPC firstly determinesand retrieves a number of historical paths in cache,called PPatterns, that may match this new query with highprobability.

The idea of PPatterns isbased on an observation that similar starting and destinationnodes of two queries may result in similar shortestpaths (known as the path coherence property).

In the componentPPatern Detection, we propose a novel probabilisticmodel to estimate the likelihood for a cached queried-pathto be useful for answering the new query by exploring theirgeospatial characteristics.



To facilitate quick detection ofPPatterns, instead of exhaustively scanning all the queriedpathsin cache, we design a grid-based index for the PPatternDetection module. Based on these detected PPatterns,the Shortest Path Estimation module (see Steps (5)-(8)) constructscandidate paths for the new query and chooses thebest (shortest) one. In this component, if a PPattern perfectlymatches the query, we immediately return it to the user;otherwise, the server is asked to compute the unmatched path segments between the PPattern and the query (seeSteps (6)-(7)). Because the unmatched segments are usuallyonly a smaller part of the original query, the server only processesa “smaller subquery”, with a reduced workload. Oncewe return the estimated path to the user, the Cache Managementmodule is triggered to determine which queried-pathsin cache should be evicted if the cache is full. An importantpart of this module is a new cache replacement policy whichtakes into account the unique characteristics of road networks.

In this paper, we provide a new framework for reusing the previously cached query results as well as an effective algorithm for improving the query evaluation on the server.

CONCLUSION:

We projected a scheme, that is, Path Planning by Caching, to respond a new path planning query with quick response by efficiently caching and reusing the chronological queried-paths. Dissimilar the conventional cache-based path planning systems, everyplace a queried-path in cache is used only when it matches perfectly with the new query, PPC leverages the partially matched cached queries to answer part(s) of a new query. As a result, the server only needs to calculate the unmatched fragment, thus significantly dipping the overall system workload.

REFERENCES:

[1] H. Mahmud, A. M. Amin, M. E. Ali, and T. Hashem, "Shared Execution of Path Queries on Road Networks," Clinical Orthopaedics and Related Research, vol. abs/1210.6746, 2012.

[2] L. Zammit, M. Attard, and K. Scerri, "Bayesian Hierarchical Modelling of Traffic Flow - With Application to Malta's Road Network," in International IEEE Conference on Intelligent Transportation Systems, 2013, pp. 1376–1381.

[3] S. Jung and S. Pramanik, "An Efficient Path Computation Model for Hierarchically Structured Topographical Road Maps," IEEE Transactions on Knowledge and Data Engineering, vol. 14, no. 5, pp. 1029–1046, 2002.

[4] E. W. Dijkstra, "A Note on Two Problems in Connexion with Graphs," Numerische Mathematik, vol. 1, no. 1, pp. 269–271, 1959.

[5] U. Zwick, "Exact and approximate distances in graphs – a survey," in Algorithms – ESA 2001, 2001, vol. 2161, pp. 33–48.

[6] A. V. Goldberg and C. Silverstein, "Implementations of Dijkstra's Algorithm Based on Multi-Level Buckets," in Network Optimization,

[7] P. Hart, N. Nilsson, and B. Raphael, "A Formal Basis for the Heuristic Determination of Minimum Cost Paths," IEEE Transactions on Systems Science and Cybernetics, vol. 4, no. 2, pp. 100–107, 1967.

[8] A. V. Goldberg and C. Harrelson, "Computing the Shortest Path: A Search Meets Graph Theory," in ACM Symposium on Discrete Algorithms, 2005.

[9] R. Gutman, "Reach-Based Routing: A New Approach to Shortest Path Algorithms Optimized for Road Networks," in Workshop on Algorithm Engineering and Experiments, 2004.

[10] A. V. Goldberg, H. Kaplan, and R. F. Werneck, "Reach for A*: Efficient Point-to-Point Shortest Path Algorithms," in Workshop on Algorithm Engineering and Experiments, 2006, pp. 129–143.

[11] S. Jung and S. Pramanik, "An Efficient Path Computation Model for Hierarchically Structured Topographical Road Maps," IEEE Transactions on Knowledge and Data Engineering, vol. 14, no. 5, pp. 1029–1046, 2002.

[12] R. Goldman, N. Shivakumar, S. Venkatasubramanian, and H. Garcia-Molina, "Proximity Search in Aatabases," in International Conference on Very Large Data Bases, 1998, pp. 26–37.

[13] N. Jing, Y.-W. Huang, and E. A. Rundensteiner, "Hierarchical Optimization of Optimal Path Finding for Transportation Applications," in ACM Conference on Information and Knowledge Management, 1996.

[14] N. Jing, Y. wu Huang, and E. A. Rundensteiner, "Hierarchical Encoded Path Views for Path Query Processing: An Optimal Model and its Performance Evaluation," IEEE Transactions on Knowledge and Data Engineering, vol. 10, pp. 409–432, 1998.

[15] U. Demiryurek, F. Banaei-Kashani, C. Shahabi, and A. Ranganathan, "Online Computation of Fastest Path in Time-Dependent Spatial Networks," in International Conference on Advances in Spatial and Temporal Databases, 2011.

[16] H. Gonzalez, J. Han, X. Li, M. Myslinska, and J. P. Sondag, "Adaptive Fastest Path Computation on a Road Network: a Traffic Mining Approach," in International Conference on Very Large Data Bases, 2007.

[17] J. R. Thomsen, M. L. Yiu, and C. S. Jensen, "Effective caching of shortest paths for location-based services," in ACM International Conference on Management of Data, 2012.

[18] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, Introduction to Algorithms, Third Edition, 2009.

[19] E. Markatos, "On Caching Search Engine Query Results," Computer Communications, vol. 24, no. 2, pp. 137–143, 2001.