International Journal of Research
Available at https://edupediapublications.org/journals

e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 04 Issue 14
November 2017

PPC: A New Cache Replacement System For Efficient Path Planning On Roads

1. Rumaiya Tarannum, 2.Ms.B.Jyothi, 3.Dr.V.Janaki

1. PG Scholar, Department of CSE, Vaagdevi College of Engineering,Bollikunta,Warangal, Telangana. Mail id:farhan0036@gmail.com

2. Assistant Professor Department of CSE, Vaagdevi College of Engineering,Bollikunta, Warangal ,Telangana. Mail id:Jyothi.mtech10@gmail.com

3.Professor,HOD Department of CSE , Vaagdevi College of Engineering, Bollikunta, Warangal, Telangana.

ABSTRACT

Owing to the wide availability of the global positioning system (GPS) and digital mapping of roads, road network navigationservices have become a basic application on many mobile devices. Path planning, a fundamental function of road network navigationservices, finds a route between the specified start location and destination. The efficiency of this path planning function is critical formobile users on roads due to various dynamic scenarios, such as a sudden change in driving direction, unexpected traffic conditions,lost or unstable GPS signals, and so on. In these scenarios, the path planning service needs to be delivered in a timely fashion. In thispaper, we propose a system, namely, Path Planning by Caching (PPC), to answer a new path planning query in real time by efficientlycaching and reusing historical queried-paths. Unlike the conventional cache-based path planning systems, where a queried-path incache is used only when it matches perfectly with the new query, PPC leverages the partially matched queries to answer part(s) of thenew query. As a result, the server only needs to compute the unmatched path segments, thus significantly reducing the overall systemworkload. Comprehensive experimentation on a real road network database shows that our system outperforms the state-of-the-artpath planning techniques by reducing 32 percent of the computation latency on average.

1 INTRODUCTION

WITH the advance of the global positioning system(GPS) and the popularity of mobile devices, we havewitnessed a migration of the conventional Internet-basedon-line navigation services (e.g., Mapquest) onto mobileplatforms (e.g., Google Map). In mobile navigation services,on-road path planning is a basic function that finds a routebetween a queried start location and a destination. While onroads, a path planning query may be issued due to dynamicfactors in various scenarios, such as a sudden change in driving direction, unexpected traffic conditions, or lost ofGPS signals. In these scenarios, path planning needs to bedelivered in a timely fashion. The requirement of timelinessis even more challenging when an overwhelming numberof path planning queries is submitted to the server, e.g.,during peak hours. As the response time is critical to usersatisfaction with personal navigation services, it is a mandatefor the server to efficiently handle the heavy workloadof path planning requests. To meet this need, we propose a

system, namely, Path Planning by Caching (PPC), that aimsto answer a new path planning query efficiently by cachingand reusing historically queried paths (queried-paths inshort). Unlike conventional cache-based path planning systemswhere a cached query is returned only when it

matches completely with a new query, PPC leverages partiallymatched queried-paths in cache to answer part(s)of the new query. As a result, the server only needs to computethe unmatched path segments, thus significantlyreducing the overall system workload.

II RELATED WORK

In this section, we review the related works in the researchlines of path planning, shortest path caching and cachemanagement, which are highly relevant to our study.

2.1 Path Planning

The Dijkstra algorithm [4], [5] has been widely used for pathplanning [6] by computing the shortest distance between twopoints on a road network. Many algorithms such as A_ [7],ATL [8] have been established to improve its performance byexploring geographical constraints as heuristics. Gutman [9]propose a reach-based approach for computing the shortestpaths. An improved version [10] adds shortcut arcs to reducevertices from being visited and uses partial trees to reducethe preprocessing time. This work further combines the benefitsof the reach-based and ATL approaches to reduce thenumber of vertex visits and the search space. The experimentshows that the hybrid approach provides a superior result interms of reducing query processing time.Jung and Pramanik [11] propose the HiTi graph model tostructure a large road network model. HiTi aims to reducethe search space for the shortest path computation. WhileHiTi achieves high performance on road weight updatesand reduces storage overheads, it incurs higher computationcosts when computing the shortest paths than thHEPV and the Hub Indexing methods [12], [13], [14]. Tocompute time-dependent fast paths, Demiryurek et al. [15]propose the B-TDFP algorithm by leveraging backwardsearches to reduce the search space. It adopts an area-levelpartition scheme which utilizes a road hierarchy to balanceeach area. However, a user may prefer a route withbetter driving experience to the shortest path. Thus,Gonzalez et al. propose an adaptive fast path algorithm.

Because SPC has to score each sub-path in the shortestpath, the time complexity is high. Assuming that a shortestpath contains n nodes, a shortest path contains ðn _ ðn_1ÞÞ=2 sub-paths. The time complexity for scoring a shortestpath is Oðn2Þ. In the above study, each query is answeredindependently. However, when queries in the currentrequest pool share similar properties, they may be processedas a group. Thus, Mahmud et al. [1] propose a groupbasedapproach to accelerate the processing by calculatingthe similarity among a group of queries and send the commonpart as a query to the server. Therefore, only dissimilarsegments for each query are answered by the server individually.However, this work does not explore any cache

mechanism in the system.

2.3 Cache Management

Caching techniques have been employed to alleviate theworkload of web searches. Since cache size is limited, cachereplacement policies have been a subject of research. Thecache replacement policy aims to improve the hit ratio andreduce access latency. Markatos et al. conduct experimentsto analyze classical cache replacement approaches on realquery logs from the EXCITE search engine . Three importantobservations are described as follows. First, a small

number of queries are frequently re-used. By preservingresults of these queries in cache, the system is able to respondto the users without incurring time-consuming computations.Second, while a larger cache size implies a higher hitratio, significant overheads may be incurred for cache maintenance.Third, static cache replacement has better performancewhen the cache size is small, and vice versa fordynamic cache replacement.

3 PATTERN DETECTION

To detect the best PPatterns, an idea is to calculate the estimationdistance based on each cached path andselect the cached path with the shortest distance. However,it faces several challenges. Firstly, the distance estimation inrequires the server to compute the unshared segments(i.e., SDPðs0; aÞ; SDPðb; t0Þ). Therefore, it incurs significantcomputation to exhaustively examine all cached paths.Secondly, such an exhaustive operation implicitly assumesthat each cached path is a PPattern

candidate to the query.However, this is not always true. For example, a path in
Manhattan does not contribute to a query in London.While we assume that users may accept an approximatepath if its error is within a certain tolerable range, exhaustiveinspection cannot not be sure that the path with theminimal error is found until all paths are inspected. Toaddress these challenges, we aim to narrow down theinspection scope to only good candidates.

Probabilistic Model for PPattern Detection

The coherency property of road networks indicates that twopaths are very likely to share segments while source nodes(and their destination nodes, respectively) are close to eachother. This property has been used in many applicationsfor various purposes, e.g., efficient trajectory lookupsas the common segments among multiple paths are queriedonly once [1]. Notice that this property is mainly attributedto the locality of the path source and destination nodes. Weargue that, for two queries, if they satisfy certain spatial constraints,their shortest distance paths are very likely to bethe PPatterns to the other.

IV CONCLUSION

In this paper, we propose a system, namely, Path Planningby Caching, to answer a new path planning query with rapidresponse by efficiently caching and reusing the historicalqueried-paths. Unlike the conventional cache-based pathplanning systems, where a queried-path in cache is usedonly when it matches perfectly with the new query, PPCleverages the partially matched cached queries to answerpart(s) of a new query. As a result, the server only needs tocompute the unmatched segments, thus significantly reducingthe overall system workload. Comprehensive experimentationon a real road network database shows that oursystem outperforms the state-of-the-art path planning techniquesby reducing 32 percent of the computational latencyon average.

REFERENCES

[1] H. Mahmud, A. M. Amin, M. E. Ali, and T. Hashem, "Shared executionof path queries on road networks," Clinical OrthopaedicsRelated Res., vol. abs/1210.6746, 2012.

[2] L. Zammit, M. Attard, and K. Scerri, "Bayesian hierarchicalmodelling of traffic flow - With application to Malta's roadnetwork," in Proc. Int. IEEE Conf. Intell. Transp. Syst., 2013,pp. 1376–1381.

[3] S. Jung and S. Pramanik, "An efficient path computation modelfor hierarchically structured topographical road maps," IEEETrans. Knowl. Data Eng., vol. 14, no. 5, pp. 1029–1046, Sep. 2002.

[4] E. W. Dijkstra, "A note on two problems in connexion withgraphs," Num. Math., vol. 1, no. 1, pp. 269–271, 1959.

[5] U. Zwick, "Exact and approximate distances in graphs – asurvey," in Proc. 9th Annu. Eur. Symp. Algorithms, 2001, vol. 2161,pp. 33–48.

[6] A. V. Goldberg and C. Silverstein, "Implementations of Dijkstra'salgorithm based on multi-level buckets," Network Optimization,vol. 450, pp. 292–327, 1997.

[7] P. Hart, N. Nilsson, and B. Raphael, "A formal basis for the heuristicdetermination of minimum cost paths," IEEE Trans. Syst. Sci.Cybern., vol. SSC-4, no. 2, pp. 100–107, Jul. 1968.

[8] A. V. Goldberg and C. Harrelson, "Computing the shortest path:A search meets graph theory," in Proc. ACM Symp. Discr. Algorithms,
2005, pp. 156–165.

[9] R. Gutman, "Reach-based routing: A new approach to shortestpath algorithms optimized for road networks," in Proc. WorkshopAlgorithm Eng. Experiments, 2004, pp. 100–111.

[10] A. V. Goldberg, H. Kaplan, and R. F. Werneck, "Reach for A*: Efficientpoint-to-point shortest path algorithms," in Proc. WorkshopAlgorithm Eng. Experiments, 2006, pp. 129–143.

[11] S. Jung and S. Pramanik, "An efficient path computation modelfor hierarchically structured topographical road maps," IEEETrans. Knowl. Data Eng., vol. 14, no. 5, pp. 1029–1046, Sep. 2002.

[12] R. Goldman, N. Shivakumar, S. Venkatasubramanian, and H.Garcia-Molina, "Proximity search in aatabases," in Proc. Int. Conf.Very Large Data Bases, 1998, pp. 26–37.

[13] N. Jing, Y.-W.Huang, and E. A. Rundensteiner, "Hierarchicaloptimization of optimal path finding for transportationapplications," in Proc. ACM Conf. Inf. Knowl.Manage., 1996,
pp. 261–268.

[14] N. Jing, Y. wu Huang, and E. A. Rundensteiner, "Hierarchicalencoded path views for path query processing: An optimal modeland its performance evaluation," IEEE Trans. Knowl. Data Eng.,vol. 10, no. 3, pp. 409–432, May/Jun. 1998.

[15] U. Demiryurek, F. Banaei-Kashani, C. Shahabi, and A. Ranganathan,"Online computation of fastest path in time-dependent spatialnetworks," in Proc. 12th Int. Conf. Adv. Spatial TemporalDatabases, 2011, pp. 92–111.

[16] H. Gonzalez, J. Han, X. Li, M. Myslinska, and J. P. Sondag,"Adaptive fastest path computation on a road network: A trafficmining

approach," in Proc. 33rd Int. Conf. Very Large Data Bases,2007, pp. 794–805.

[17] J. R. Thomsen, M. L. Yiu, and C. S. Jensen, "Effective caching ofshortest paths for location-based services," in Proc. ACM SIGMODInt. Conf. Manage. Data, 2012, pp. 313–324.

[18] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, Introduction
to Algorithms, 3rd ed. Cambridge, MA, USA: MIT Press 2009.

[19] E. Markatos, "On caching search engine query results," Comput.Commun., vol. 24, no. 2, pp. 137–143, 2001.

[20] R. Ozcan, I. S. Altingovde, and O. Ulusoy, "A cost-aware strategyfor query result caching in web search engines," in Proc. Adv. Inf.Retrieval, 2009, vol. 5478, pp. 628–636.